ORNL/TM-SCALE-6.3.0

SCALE 6.3.0 User Manual

W. A. Wieselquist
R. A. Lefebvre, Editors

Dec. 2021

OAK RIDGE National Laboratory

ORNL IS MANAGED BY UT–BATTELLE, LLC, FOR THE U.S. DEPARTMENT OF ENERGY

ORNL/TM-SCALE-6.3.0

Nuclear Energy and Fuel Cycle Division

SCALE 6.3.0 USER MANUAL

W. A. Wieselquist
R. A. Lefebvre, Editors

Date Published: Dec. 2021

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831-6283
managed by
UT-Battelle, LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

ABSTRACT

SCALE is a comprehensive modeling and simulation suite for nuclear safety analysis and design developed and maintained by Oak Ridge National Laboratory under contract with the U.S. Nuclear Regulatory Commission, U.S. Department of Energy, and the National Nuclear Security Administration to perform reactor physics, criticality safety, radiation shielding, and spent fuel characterization for nuclear facilities and transportation/storage package designs.

Visit the SCALE website for additional information.

1. INTRODUCTION

W. A. Wieselquist

The SCALE code system includes verified and validated tools for criticality safety, reactor physics, radiation shielding, radioactive source characterization, and sensitivity and uncertainty analysis. SCALE is developed, maintained, tested, and managed by Oak Ridge National Laboratory (ORNL) and may be obtained through the Radiation Safety Information Computational Center (RSICC).

This manual documents version 6.3 of the SCALE code system, with initial release of 6.3.0 in 2022. The previous 6.2 series had its last maintenance release, 6.2.4, in 2020. This section describes the key features in 6.3 relative to 6.2 and details on how to run SCALE.

Maintenance releases of SCALE 6.3 (6.3.1, 6.3.2, etc.) will be made as needed to fix any issues discovered. Historically, there has been approximately one maintenance release per year. Note that once users have been granted a license from RSICC, any subsequent maintenance releases may be obtained free of charge directly from the SCALE team by sending an email to scalehelp@ornl.gov.

The next major release of SCALE with additional features will be SCALE 7.0 in ~2025.

The SCALE manual is included with installation; however, an online version is also available. The online version is new as of SCALE 6.3 and is in a convenient format for users, containing detailed lists of changes as well as any issues discovered in a particular version, starting with SCALE 6.3.0.

Instructions on installing or building SCALE, testing the configuration with the sample problems are available in Sect. 12.

1.1 ORGANIZATION

SCALE's top-level applications have been historically called "sequences" because of their original design, which called specific modules in sequence to solve specific problems. With the development of SCALE 6.3, we have found it more useful to think of SCALE as a set of products. A product is potentially more than one sequence and attempts to logically group capabilities around a particular application area. This is useful both from the standpoint of helping a user identify which sequences they should use and from a development point of view, as domain experts are assigned to each product. See Table 1.1.1 for the relationships between end user applications, SCALE products, and sequences.

Table 1.1.1: Relationship between end-user applications, SCALE products, and sequences

End user Application Area	Product	Example Sequences
Reactor Physics	Polaris	`polaris`

	TRITON	`t-depl`
		`t5-depl`
		`t5-depl`
		`t-depl-1d`

Criticality Safety	CSAS	`csas5` `csas6`
	VADER	`vader`

continues on next page

Table 1.1.1 – continued from previous page

End user Application Area	Product	Example Sequences
Spent Fuel Inventory	ORIGAMI	`origami`
	ORIGEN	`arp` `origen`
Activation and Decay	ORIGEN	`couple` `origen`
Radiation Shielding	MAVRIC	`mavric`

Sensitivity and Uncertainty	TSUNAMI	`tsunami-3d-k5`
		`tsunami-3d-k6`
		`tsunami-ip`
		`tsar`
		`tsurfer`
		`tsunami-1d`
		`tsunami-2d`

	Sampler	`sampler`

The reconceptualization of sequences as products also facilitates management and discussion of certain key SCALE components, such as nuclear data. Table 1.1.2 briefly describes each product and contains links to the relevant section(s) in the manual.

Table 1.1.2: SCALE Products

Product	Purpose	Reference
Fulcrum	SCALE Graphical User Interface	Sect. 12.3.1
CSAS	Solve Monte-Carlo (and deterministic) criticality problems	Sect. 2.1
MAVRIC	Solve Monte-Carlo fixed source radiation transport with automatic variance reduction for difficult shielding tallies	Sect. 4.1
ORIGAMI	Generate fast LWR spent fuel inventory predictions	Sect. 5.4
ORIGEN	General depletion	Sect. 5.1.5
Polaris	Easy-to-use lattice physics	Sect. 3.2
Sampler	General Uncertainty Propagation	Sect. 6.4
TSUNAMI	Sensitivity and experiment selection analysis	Sect. 6.1, Sect. 6.2
VADER	Trending analysis	Sect. 6.5
TRITON	Solve coupled neutron transport and depletion problems for nuclear reactor systems	Sect. 3.1
XSProc	Accurate multi-group cross section self-shielding methods	Sect. 7.1
DATA	Verified and validated SCALE fundamental nuclear data	Sect. 10.1
AMPX	Transform ENDF-formatted data into SCALE libraries	external
OMNIBUS	Flexible user interface for HPC Applications	external

1.2 SCALE 6.3 UPDATES BY PRODUCT

The new features relative to SCALE 6.2 are described below, organized in terms of products as shown in Table 1.1.2.

1.2.1 FULCRUM

The major new feature in Fulcrum is the 3D visualization of geometry, including results overlay capability.

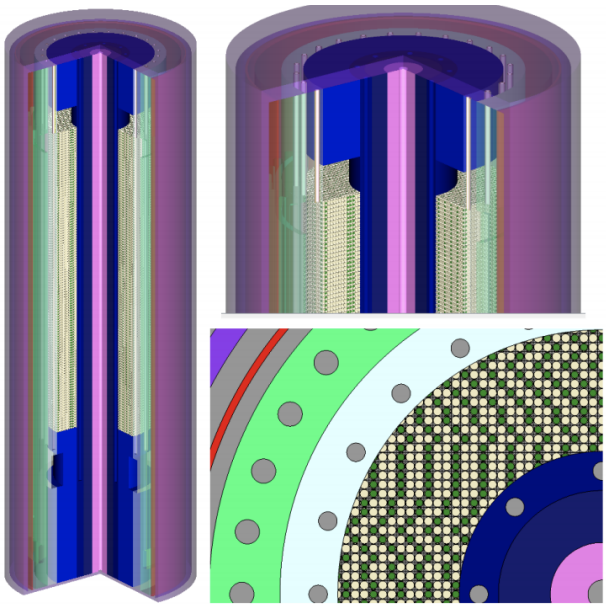


Fig. 1.2.1: Fulcrum visualization of geometry including new 3D.

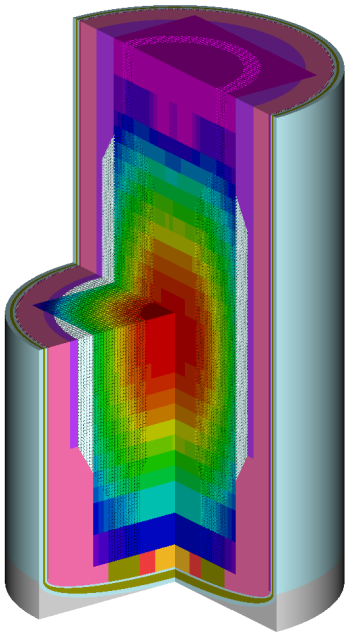


Fig. 1.2.2: Fulcrum visualization of 3D reactor geometry with flux overlay.

1.2.2 CSAS

The key new CSAS feature in 6.3 is the ability to call Shift for neutron transport. The same CSAS user interface is available for both CSAS-KENO and CSAS-Shift, and the majority of KENO features are supported by Shift. See Sect. 2.2 for details.

- CSAS-KENO now allows specification of the upper thermal scattering cutoff energy. A value above the default 10 eV may be necessary for high-temperature graphite–moderated systems. See Sect. 8.1.7.2.17 for details.
- CSAS now supports reusable energy and spatial grid definitions in the `definitions` block, especially useful for producing *.3dmap files for visualization in Fulcrum. See Sect. 2.1.4.2.1 for details.
- Shift can randomly place spheres within volumes, also useful for tristructural isotropic (TRISO) packing in pebbles. See Sect. 2.2.4.1.3 for details.

1.2.3 VADER

VADER is the modernization of the trending analysis code USLStats previously externally distributed with SCALE. VADER uses the SON input format introduced in SCALE 6.2. See Sect. 6.5 for details.

1.2.4 MAVRIC

The key new MAVRIC feature in 6.3 is the ability to call Shift for neutron and gamma transport. Append `-shift` to the sequence name to use Shift as in the example below.

```
=mavric-shift
Leakage spectrum of Cf-252 through a heavy water sphere
v7.1-28n19g

read composition
    d2o          1 0.99286 293.0 end
    h2o          1 0.00714 293.0 end
    polyethylene 2 0.882   293.0 end
    boron        2 0.118   293.0 end
    iron         3 1.0     293.0 end
    orconcrete   4 1.0     293.0 end
end composition

'------------------------------------------------------------------------------
' Geometry Block
'------------------------------------------------------------------------------
read geometry
    global unit 1
        sphere    10   0.5
        sphere    11  15.0
        sphere    21   2.0  origin x=75.0
        cuboid    41  650.0 -650  500 -500   2300 -200
        cuboid    42  750.0 -750  600 -600   2400 -300
        media  0 1   10
        media  1 1   11 -10
        media  0 1   21          vol=33.510322
        media  0 1   41 -11 -21
        media  4 1   42 -41
    boundary 42
end geometry

'------------------------------------------------------------------------------
' Definitions Block
'------------------------------------------------------------------------------
read definitions
    distribution 1
```

(continues on next page)

(continued from previous page)

```
        title="Cf-252 neutrons, Watt spectrum a=1.025 MeV and  b=2.926/MeV"
        special="wattSpectrum"
        parameters 1.025 2.926 end
    end distribution
end definitions

'------------------------------------------------------------------------------
' Sources Block
'   Cf-252 neutrons, Watt fission spectrum model
'   strength set so that total unattenuated flux at detector would be 1
'      strength = 4*pi*(75)^2
'------------------------------------------------------------------------------
read sources
    src 1
        title="Cf-252 neutrons, Watt fission spectrum, using a=1.025 and b=2.926"
        neutrons  strength=70685.834704
        sphere 0.1
        eDistributionID=1
    end src
end sources

'------------------------------------------------------------------------------
' Tallies Block
'------------------------------------------------------------------------------
read tallies
    regionTally 3
        title="example region tally"
        neutron
        unit=1 region=3
    end regionTally
end tallies

'------------------------------------------------------------------------------
' Parameters Block
'------------------------------------------------------------------------------
read parameters
    randomSeed=8655745262010033
    batches=10
    neutrons  noPhotons
    fissionMult=0  secondaryMult=0

'   speed things up for the sample problem
    perBatch=20000
end parameters

end data
end
```

Note: Some features in MAVRIC using the default Monaco transport engine are not yet available in Shift.

1.2.5 TRITON

TRITON was updated with a host of new capabilities in SCALE 6.3.

- TRITON can now call the new Shift Monte Carlo code instead of KENO. Shift was designed with parallelism and robustness as the highest priorities. With Shift enabled, few-group macroscopic cross sections on 3D hex and Cartesian meshes can be generated, similar to but more general than existing capabilities in TRITON-NEWT and Polaris. See Sect. 3.1.3.3.15 for details.
- Shift can randomly place spheres within volumes, which is available in both CSAS-Shift and TRITON-Shift. See Sect. 2.2.4.1.3 for details.

- A new flow block has been added to the TRITON timetable, intended for simulating molten salt reactors. See Sect. 3.1.3.3.11 for details.

1.2.6 POLARIS

The main Polaris improvement for 6.3 is a noticeable decrease in runtime, by a factor of 3 to 5 for almost all use cases. This was enabled through numerous performance-related refactors of the geometry, transport, and self-shielding components.

In addition, the following updates were made.

- Additional accident-tolerant fuel compositions (see Sect. 3.2.5.9) and new dopant properties (see Sect. 3.2.5.11).
- Polaris can now generate ORIGEN library files (*.f33) (see Sect. 3.2.7.4).
- Gamma detector modeling was added.
- Output file was improved.

See Sect. 3.2.2 for details on the updates.

1.2.7 ORIGEN

Extensive improvements to ORIGEN were made for SCALE 6.3, including the following.

- Modernized library construction integrated into the ORIGEN sequence instead of external the COUPLE sequence (see Sect. 5.1.5.2.21 for details).
- Sensitivity analysis capability (see Sect. 5.1.5.2.28 for details).
- OBIWAN command line utility (see Sect. 5.2 for details).

1.2.8 ORIGAMI

There were minimal changes to ORIGAMI for SCALE 6.3; however, the ORIGEN reactor library data has been refreshed and includes higher burnups and enrichments (see Sect. 5.3 for details).

1.2.9 XSPROC

The main changes to XSProc for SCALE 6.3 comprise adjustments to the default self-shielding parameters to reduce biases at high temperature and high burnup, as well as to improve the Bondarenko method accuracy for fuel models with multiple radial temperature zones. The input below demonstrates a lattice cell with multiple temperature rings using the fast Bondarenko-based self-shielding method (Bonami).

```
=csas6 parm=(bonami)
mg-keno with bonami for nonuniform temperature
v7.1-56
read comp
   uo2  1  den=10.97  1.0 1200.0  92235 5.0  92238 95.0 end
   uo2  2  den=10.97  1.0 1100.0  92235 5.0  92238 95.0 end
   uo2  3  den=10.97  1.0 1000.0  92235 5.0  92238 95.0 end
   uo2  4  den=10.97  1.0  900.0  92235 5.0  92238 95.0 end
   uo2  5  den=10.97  1.0  800.0  92235 5.0  92238 95.0 end
   uo2  6  den=10.97  1.0  700.0  92235 5.0  92238 95.0 end
   uo2  7  den=10.97  1.0  600.0  92235 5.0  92238 95.0 end
helium  8   1.0 600.0 end
 zirc4  9   1.0 600.0 end
   h2o 10  den=0.661  1.0  600.0 end
```

(continues on next page)

(continued from previous page)

```
end comp

read celldata
latticecell squarepitch
  pitch=1.4427  10
  fuelr=0.177473 1
  fuelr=0.250985 2
  fuelr=0.307393 3
  fuelr=0.354946 4
  fuelr=0.396842 5
  fuelr=0.434719 6
  fuelr=0.469550 7
  gapr=0.479100  8
  cladr=0.546400 9 end
end celldata

read param
 gen=200
 nsk=100
 npg=400000
 sig=0.0001
 far=yes
end param

read geometry
global unit 1
  cylinder 1 0.177473 1.0 -1.0
  cylinder 2 0.250985 1.0 -1.0
  cylinder 3 0.307393 1.0 -1.0
  cylinder 4 0.354946 1.0 -1.0
  cylinder 5 0.396842 1.0 -1.0
  cylinder 6 0.434719 1.0 -1.0
  cylinder 7 0.469550 1.0 -1.0
  cylinder 8 0.479100 1.0 -1.0
  cylinder 9 0.546400 1.0 -1.0
  cuboid  10 6p0.72135
  media   1  1  1
  media   2  1  2 -1
  media   3  1  3 -2
  media   4  1  4 -3
  media   5  1  5 -4
  media   6  1  6 -5
  media   7  1  7 -6
  media   8  1  8 -7
  media   9  1  9 -8
  media  10  1 10 -9
  boundary 10
end geometry

read bounds
all=mirror
end bounds

end data
end
```

1.2.10 DATA

One of the main data updates for SCALE 6.3 is the inclusion of additional ENDF/B-VIII.0 data resources and the removal of ENDF/B-VII.0 data. New data resources include the following:

- New ice and other compounds such as reactor-grade graphite present in ENDF/B-VIII.0.
- Continuous-energy (CE) cross section data based on ENDF/B-VII.1 and ENDF/B-VIII.0.
- New multigroup (MG) libraries for fast-spectrum and thermal-spectrum systems (reactivity and depletion results closer to higher fidelity CE results).

See Sect. 10.1 for details.

Note that all 3D Monte Carlo codes in SCALE (TRITON, CSAS, MAVRIC) support both CE and MG methods. MG data and methods are faster but more approximate solutions.

1.2.11 SAMPLER

Sampler includes considerable updates to help users understand the causes of uncertainty in their simulations. Updates include the following:

- New sensitivity metrics for understanding nuclear data responsible for the majority of uncertainty (see Sect. 6.4.2.5 for details).
- Ability to analyze uncertainty due to delayed neutron data using the new `perturb_kinetics` option.
- A new analysis block which includes ability to calculate correlation coefficients between arbitrary outputs (see Sect. 6.4.3.7 for details).

1.2.12 TSUNAMI

For SCALE 6.3, the following enhancements were made to TSUNAMI.

- New Shift-based iterated fission probability (IFP) method with better performance than previous KENO-based IFP and Clutch methods.
- Improved performance of TSUNAMI-IP similarity and uncertainty calculations.
- New general HDF5 format for the sensitivity coefficients (see Sect. 6.3.4.1.3 for details).

1.2.13 AMPX

AMPX continues to be included in SCALE 6.3 and has been used exclusively to generate all libraries described in Sect. 10.1. The major enhancements were modernization related—to increase robustness of generated CE and MG libraries and to enable reading the new international GNDS data format. AMPX is now an open-source code system; the newest features are available by contacting the AMPX team.

1.2.14 OMNIBUS

Omnibus is the new frontend for the high-performance Shift Monte Carlo and Denovo deterministic transport codes that enables cutting-edge execution of Shift on GPU and Hybrid GPU/CPU platforms. As this capability evolves rapidly, please contact the authors of [IntroPJE+16] for details on getting the latest version of this frontend and manual [IntroJED+20].

1.3 USING SCALE

SCALE sequences have been incrementally developed over several decades, with the primary goals of robustness, accuracy, and ease-of-use. One side effect of this evolution can be seen in our user interfaces: the text-based input that drives a calculation. There is no standard SCALE input. For example, TRITON, CSAS, and MAVRIC are similar in their style, which predates SCALE 6.2, when numerous new user interfaces were introduced.

For example:

- The Polaris sequence for lattice physics was modeled after the brevity and conciseness of CASMO and has a distinct syntax and output.
- The ORIGEN code for general depletion and decay uses the SON syntax introduced in SCALE 6.2 as a significant upgrade from FIDO, but it keeps the same basic input structure and has a more general feel compared to Polaris.
- The Sampler code for uncertainty propagation was created with a syntax originally intended to be the upgrade to the TRITON, CSAS, and MAVRIC style, but it was superseded by SON.
- The shell sequence, which can perform a limited set of common file system operations across platforms.

The SCALE input file is quite flexible in that it can contain numerous sequence inputs executed sequentially. For example, in the input below, we irradiate a milligram of iron in a beginning-of-life pressurized water reactor spectrum for one day and then decay for nine days.

```
=shell
cp ${DATA}/arplibs/w17_e50.f33 f33
end

=origen
case{
  lib{ file=f33 pos=1 }
  mat{ iso=[Fe=0.001] units=GRAMS }
  flux=[1e14  0] %neutrons/cm^2-s
  time=[   1 10] %days
}
end
```

The first *=shell* sequence copies a specific ORIGEN library from the SCALE data directory to the temporary working directory. ORIGEN can then find this file to load one-group cross sections for this test irradiation.

1.3.1 RUNNING SCALE FROM FULCRUM

The most convenient way to run SCALE from a desktop is by launching Fulcrum. The Fulcrum executable is provided in the *bin* directory where SCALE was installed (e.g., `C:\SCALE-6.3.0\bin\Fulcrum.exe`). Fulcrum includes an online help document to assist users with its many features, and it includes links to the user manual and primers.

1.3.2 RUNNING SCALE FROM THE COMMAND LINE

Using the command line, SCALE can be executed using the *scalerte* command from the *bin* directory inside the SCALE installation (e.g., `C:\SCALE-6.3.0\bin\scalerte`). Your directory may differ based on the installation. Assuming the location of *scalerte* is known, a SCALE input can be run simply as

```
scalerte -m my.inp
```

with the *-m* option requesting SCALE to output status updates to the terminal. See Sect. 12 for details.

1.4 USER GUIDANCE AND TECHNICAL ASSISTANCE

This SCALE manual serves as the primary reference for SCALE users. The fundamental theory and all code options are documented herein. Several SCALE primers are available to serve as step-by-step guides for new users performing common calculations using the GUIs. SCALE training courses are presented several weeks each year, during which users can interact directly with the software developers and expert users from ORNL. Additional technical information on SCALE can be found at http://scale.ornl.gov—including training course schedules, a link to an online user forum, newsletters, benchmark reports, and downloads. Technical assistance is also provided via email at scalehelp@ornl.gov.

1.5 CODE AVAILABILITY

The SCALE code system is packaged and distributed by the RSICC and is also distributed through the Organization for Economic Cooperation and Development (OECD) Nuclear Energy Agency (NEA) Data Bank in France and the Research Organization for Information Science and Technology (RIST) in Japan.

1.6 HISTORY

The SCALE code system dates back to 1969, when ORNL began providing the transportation package certification staff at the US Atomic Energy Commission (AEC) with computational support in the use of the new KENO code. KENO was used to perform criticality safety assessments with the statistical Monte Carlo method. From 1969 to 1976, the AEC certification staff relied on ORNL personnel to assist them in the correct use of codes and data for criticality, shielding, and heat transfer analyses of transportation packages. However, the certification staff learned that occasional users had difficulty becoming proficient in performing the calculations often needed for an independent safety review. Thus, shortly after the certification staff was moved to the US Nuclear Regulatory Commission (NRC), the NRC proposed development of an easy-to-use analysis system that provided the technical capabilities of the individual modules with which they were familiar. With this proposal, the concept of SCALE as a comprehensive modeling and simulation suite for nuclear safety analysis and design was born. The NRC staff provided ORNL with some general development criteria for SCALE: (1) focus on applications related to nuclear fuel facilities and package designs, (2) use well-established computer codes and data libraries, (3) design an input format for the occasional or novice user, (4) prepare standard analysis sequences (control modules) to automate the use of multiple codes (functional modules) and data to perform a system analysis, and (5) provide complete documentation and public availability. With these criteria, the ORNL staff laid out the framework for the SCALE system and began development efforts. The initial version of SCALE (Version 0) was distributed in July 1980. Although the capabilities of the system continue to evolve, the philosophy established with the initial release still serves as the foundation of SCALE 6.3, more than 40 years later. In July 1980, the initial version of SCALE was made available to the Radiation Safety Information Computational Center (RSICC) at ORNL. This system was packaged and released by RSICC as CCC-288/SCALE 0. Subsequent additions and modifications resulted in the following releases: CCC-424/SCALE1 in 1981; CCC-450/SCALE 2 in 1983; CCC-466/SCALE 3 in 1985; CCC-545/SCALE 4.0 in 1990; SCALE 4.1 in 1992; SCALE 4.2 in 1994; SCALE 4.3 in 1995; SCALE 4.4 in 1998; SCALE 4.4a in 2000; CCC-725/SCALE 5 in 2004; CCC-732/SCALE 5.1 in 2006; CCC-750/SCALE 6.0 in 2009; CCC-785/SCALE 6.1 in 2011; CCC-834/SCALE 6.2 in 2016.

1.7 ACKNOWLEDGEMENTS

Most team members are credited for their authorship of the sections in this manual that correspond to their work. A few individuals have been essential to the development and maintenance of SCALE but are not credited by authorship. These individuals include B. Taylor, J. Batson, M. Henley, S. Poarch, B. Bevard, L. Aloisi, D. Bowen, and R. Grove.

We also acknowledge the support of Dr. A. Chambers and the DOE/NCSP and D. Algama, D. Barto, L. Kyriazidis, and H. Esmaili of the NRC.

2. CRITICALITY SAFETY

Introduction by B. T. Rearden [1] and K. B. Bekar

SCALE provides a suite of computational tools for criticality safety analysis primarily based on the Monte Carlo codes KENO and Shift for eigenvalue neutronics calculations. [CritSafetyGPJD+11], [CritSafetyPJE+16].

Two variants of KENO provide identical solution capabilities with different geometry packages. KENO V.a uses a simple and efficient geometry package sufficient for modeling many systems of interest to criticality safety and reactor physics analysts. KENO-VI uses the SCALE Generalized Geometry Package, which provides a quadratic-based geometry system with much greater flexibility in problem modeling but with slower runtimes. Both versions of KENO perform eigenvalue calculations for neutron transport primarily to calculate multiplication factors (*keff*) and flux distributions of fissile systems in both continuous-energy and multigroup modes. KENO's grid geometry capability extends region-based features for accumulating data for source or biasing parameter specifications, as well as for tallying results from a calculation for visualization or communication of data into or out of a calculation.

Shift, an advanced Monte Carlo code specifically designed for efficient parallel and GPU executions for high-performance computers, provides both eigenvalue and fixed-source Monte Carlo transport capabilities as well as hybrid capabilities for variance reduction methods with the Denovo deterministic transport solver. [CritSafetyPJE+16], [CritSafetyESSC10]. Shift supports different geometry engines including the Oak Ridge Adaptable Nested Geometry Engine (ORANGE) designed to provide particle transport capabilities on both KENO V.a and KENO-VI geometries as well as geometry visualization capabilities in the Fulcrum user interface. Shift with both versions of KENO geometries performs eigenvalue calculations in both continuous-energy and multigroup modes. Shift supports most widely used primary capabilities available with KENO codes.

Capabilities with both KENO codes and Shift code are typically accessed through the integrated SCALE sequences described below. Criticality safety analysts may also be interested in the sensitivity and uncertainty analysis techniques that can be applied for code and data validation as described elsewhere in this document.

[1] Formerly with Oak Ridge National Laboratory.

Criticality Safety Analysis Sequences

The Criticality Safety Analysis Sequences (CSAS) with KENO V.a (CSAS5), KENO-VI (CSAS6), and Shift (CSAS5-Shift, CSAS6-Shift) provide a reliable, efficient means of performing *keff* calculations for systems routinely encountered in engineering practice. The CSAS sequences implement XSProc to process material input and provide a temperature and resonance-corrected cross section library based on the physical characteristics of the problem being analyzed. If a continuous energy cross section library is specified, no resonance processing is needed, and the continuous energy cross sections are used directly in KENO, with temperature corrections provided as the cross sections are loaded.

CSAS sequences with 3D Monte Carlo transport capabilities currently available in SCALE 6.3 are listed in Table 2.1. The transport module run by each sequence, and the geometry engine used by each transport module, are also provided in this table. Note that the sequence names CSAS25 and CSAS26—similarly, CSAS25-Shift and CSAS26-Shift—are only the alias names of the CSAS sequences, and they were added to the SCALE repository for backward compatibility purposes.

Table 2.1: Valid CSAS sequences available in SCALE 6.3

Sequence Name	Transport Module	Geometry Engine
CSAS5, CSAS25	KENO V.a	KENO V.a
CSAS6, CSAS26	KENO-VI	SGGP [1]
CSAS5-Shift, CSAS25-Shift	Shift with KENO V.a geometry	ORANGE [2]
CSAS6-Shift, CSAS26-Shift	Shift with KENO-VI geometry	ORANGE

[1] SGGP is SCALE General Geometry Package

[2] ORANGE is a new C++ geometry package, the Oak Ridge Adaptable Nested Geometry Engine, has been designed to provide particle transport capabilities on both KENO V.a and KENO-VI geometries in SCALE sequences as well as geometry visualization capabilities in the Fulcrum user interface.

For continuous energy calculations, reaction rate tallies can be requested within the CSAS input, and for multigroup calculations, reaction rate calculations are performed using the KENO Module for Activity-Reaction Rate Tabulation (KMART) post-processing tools. A conversion tool is provided to up-convert KENO V.a input to KENO-VI either as a direct KENO input (K5toK6) or, more commonly, as a CSAS sequence (C5toC6). Note that these capabilities are only available with CSAS5 and CSAS6 sequences.

The CSAS5 search capability available in previous SCALE versions is no longer supported by the CSAS sequences in SCALE 6.3. Research is being conducted to support the equivalent search capabilities in a more robust modernized code framework for the next SCALE release.

CSAS sequences support parallel execution of KENO V.a, KENO-VI, and Shift transport modules. When running on multiple cores, Shift performance is always better than KENO codes since its design was targeted for high-performance computers.

Criticality Accident Alarm System Analysis with KENO and MAVRIC

Criticality accident alarm systems (CAAS) safety analyses modeling presents challenges because the analysis consists of a criticality problem and a deep-penetration shielding problem [CritSafetyPPJ09]. Modern codes are typically optimized to handle one of these types of problems, but not both. The two problems also differ in size-the criticality problem depends on materials relatively close to the fissionable materials, whereas the shielding problem can cover a much larger range.

CAAS analysis can be performed using the CSAS6 criticality sequence and the MAVRIC shielding sequence. First, the fission distribution (in space and energy) is determined via CSAS6. This information is collected on a grid geometry that overlies the physical geometry model and is saved as a Monaco mesh source file. The mesh source is then used as the source term in MAVRIC. The absolute source strength is set by the user to the total number of fissions (based on the total power released) during the criticality excursion. MAVRIC can be optimized to calculate a specific detector response at one location or to calculate multiple responses/locations with roughly the same relative uncertainty. See Sect. 4.2 for further details.

Note: The Sourcerer sequence is no longer supported in SCALE 6.3 because it depends on several legacy components not supported in SCALE 6.3. The equivalent capability will be designed as another start data type in CSAS sequences for the next SCALE release.

Note: DEVC sequence is a deprecated capability in SCALE 6.3.

2.1 CSAS: CONTROL MODULE FOR ENHANCED CRITICALITY SAFETY ANALYSIS SEQUENCES WITH KENO

K. B. Bekar, *L. M. Petrie* [1], *S. Goluoglu* [1], *D. F. Hollenbach* [1] and *N. F. Landers* [1]

The **C**riticality **S**afety **A**nalysis **S**equences with KENO codes provide reliable and efficient means of performing k_{eff} calculations for systems that are routinely encountered in engineering practice. Two CSAS sequence implementations, CSAS5 and CSAS6, with two variants of KENO codes, KENO V.a and KENO-VI, provide identical solution capabilities with different geometry packages. In the multigroup calculation mode, CSAS uses XSProc to process the cross sections for temperature corrections and problem-dependent resonance self-shielding and calculates the k_{eff} of a three-dimensional (3D) system model. If the continuous-energy calculation mode is selected no resonance processing is needed and the continuous-energy cross sections are used directly in KENO codes, with temperature corrections provided as the cross sections are loaded. The geometric modeling capabilities available in KENO codes coupled with the automated cross-section processing within the control sequences allow complex, 3D systems to be easily analyzed.

The CSAS5 search capability available in previous SCALE versions is no longer supported by the CSAS5 sequence in SCALE 6.3.

In SCALE 6.3, CSAS5 and CSAS6 support two new sequence data blocks, definitions and tallies data, to allow flexible definition and output control of mesh tallies. The mesh responses neutron flux, fission rate, and fission source can now be requested multiple times on different spatial and energy grids in the same calculation.

[1] Formerly with Oak Ridge National Laboratory.

2.1.1 ACKNOWLEDGMENTS

CSAS5 and its related Criticality Safety Analysis sequences are based on the old CSAS2 control module (no longer in SCALE) and the KENO V.a functional module described in Sect. 8.1. Therefore, special acknowledgment is made to J. A. Bucholz, R. M. Westfall, and J. R. Knight who developed CSAS2. G. E. Whitesides is acknowledged for his contributions through early versions of KENO. Appreciation is expressed to C. V. Parks and S. M. Bowman for their guidance in developing CSAS5.

2.1.2 INTRODUCTION

Criticality Safety Analysis Sequence with KENO V.a (CSAS5) and KENO-VI (CSAS6) provide reliable and efficient means of performing k_{eff} calculations for systems that are routinely encountered in engineering practice, especially in the calculation of k_{eff} of three-dimensional (3D) system models. CSAS5 and CSAS6 implement XSProc to process material input and provide a temperature and resonance-corrected cross-section library based on the physical characteristics of the problem being analyzed. If a continuous energy cross-section library is specified, no resonance processing is needed and the continuous energy cross sections are used directly in KENO codes, with temperature corrections provided as the cross sections are loaded.

The search capability available in the CSAS5S in previous SCALE versions is no longer supported by the CSAS5 in SCALE 6.3. This capability was excluded when doing modernization work for CSAS sequences in SCALE 6.2 and permanently disabled in SCALE 6.3 due to the inconsistencies between the legacy code implementation and the modern CSAS framework. Research is being continued to support equivalent search capabilities in a more robust modernized code framework for the next SCALE release.

In SCALE 6.3, CSAS5 and CSAS6 support two new sequence data blocks, definitions and tallies data, to allow flexible definition and output control of mesh tallies. The mesh responses neutron flux, fission rate, and fission source can now be requested multiple times on different spatial and energy grids in the same

calculation. This capability helps users efficiently manage computational resources when collecting detailed information, depending on their requirements. For example, fission density can be tallied in a very fine spatial mesh in a few energy groups while performing calculations in high resolution with SCALE's very fine group multigroup library (1597 energy groups), or fission density can be tallied on multiple spatial fine meshes rather than using a large global fine mesh to keep the runtime and memory footprint of the calculation at reasonable levels.

2.1.3 SEQUENCE CAPABILITIES

In the CSAS sequence framework, SCALE data handling is automated as much as possible. CSAS and many other SCALE sequences apply a standardized procedure to provide appropriate number densities and cross sections for the calculation. XSProc is responsible for reading the standard composition data and other engineering-type specifications, including volume fraction or percent theoretical density, temperature, and isotopic distribution as well as the unit cell data. XSProc then generates number densities and related information, prepares geometry data for resonance self-shielding and flux-weighting cell calculations, if needed, and (if needed) provides problem-dependent multigroup cross-section processing. Sequences that execute KENO codes include a KENO Data Processor to read and check the KENO data. When the data checking has been completed, the control sequence executes XSProc to prepare a resonance-corrected microscopic cross-section library in the AMPX working library format if a multigroup library has been selected.

For each unit cell specified as being cell-weighted, XSProc performs the necessary calculations and produces a cell-weighted microscopic cross-section library. KENO codes may be executed to calculate the k_{eff} or neutron multiplication factor using the cross-section library that was prepared by the control sequence.

Computational capabilities available in KENO codes—including the determination of k-effective, neutron lifetime, generation time, energy-dependent leakages, energy- and region-dependent absorptions, fissions, the system mean-free-path, the region-dependent mean-free-path, average neutron energy, flux densities, fission densities, reaction rate tallies, mesh tallies, source convergence diagnostics, problem-dependent continuous-energy temperature treatments, parallel calculations, restart capabilities, and many more—are also provided by the CSAS5 sequence. Details of each capability, their input methods, and output edits are provided in Sect. 8.1 of this document and will not be repeated here.

2.1.3.1 Multigroup limitations

The CSAS control module was developed to use simple input data and prepare problem-dependent cross sections for use in calculating the effective neutron multiplication factor of a 3D system using KENO codes, KENO V.a and KENO-VI. An attempt was made to make the system as general as possible within the constraints of the standardized methods chosen to be used in SCALE. Standardized methods of data input were adopted to allow easy data entry and for quality assurance purposes. Some of the limitations of the CSAS multigroup sequences are a result of using preprocessed multigroup cross sections. Inherent limitations in multigroup CSAS calculations are as follows:

1. Two-dimensional (2D) effects such as fuel rods in assemblies where some positions are filled with control rod guide tubes, burnable poison rods and/or fuel rods of different enrichments. The cross sections are processed as if the rods are in an infinite lattice of identical rods. If the user inputs a Dancoff factor for the cell (such as one computed by MCDancoff), XSProc can produce an infinite lattice cell, which reproduces that Dancoff. This can mitigate some two dimensional lattice effects

2.1.3.2 Continuous energy limitations

When continuous energy KENO calculations are desired, none of the resonance processing capabilities of XSProc are applicable or needed. The continuous energy cross sections are directly used in KENO. An existing multigroup input file can easily be converted to a continuous energy input file by simply specifying the continuous energy library. In this case, all cell data is ignored. However, the following limitations exist:

1. If CELLMIX is defined in the cell data, the problem will not run in the continuous energy mode. CELLMIX implies new mixture cross sections are generated using XSDRNPM-calculated cell fluxes and therefore is not applicable in the continuous energy mode.
2. Only VACUUM, MIRROR, PERIODIC, and WHITE boundary conditions are allowed. Material-specific albedos, e.g., WATER, CARBON, POLY, etc., are for multigroup only.
3. Problems with DOUBLEHET cell data are not allowed as they inherently utilize CELLMIX feature.

2.1.4 INPUT DATA GUIDE

This section describes the input data required for the CSAS with KENO transport codes. A typical CSAS input, shown in Example 2.1.1, starts with the sequence identifier always preceded by the = sign (=CSAS5 and =CSAS6), and it is followed by the problem title. Then, a cross section library name is specified, and all these entries are followed by several data blocks each starting with READ data_block and ending with END data_block.

Example 2.1.1: A typical CSAS sequence input

```
=sequence_identifier parm=(parm_options)
problem title

' ----- XSProc data
' cross section library name (REQUIRED)
ce_v7.1

' List of material specifications in standard SCALE format (REQUIRED)
read composition
...
end composition

' Specify data for resonance processing (OPTIONAL)
read celldata
...
end celldata

' ---- New CSAS sequence data blocks

' Used to define energy bounds and grid geometries for
' the tallies defined in tallies data block
' (REQUIRED if tallies data block exists)
read definitions
...
end definitions

' Used to define tallies in a more robust way (OPTIONAL)
read tallies
...
end tallies

' ---- KENO transport data
' Specify the problem geometry (REQUIRED)
read geometry
```

(continues on next page)

(continued from previous page)

```
...
end geometry

' Other input data blocks (OPTIONAL)
```

The input data for the CSAS sequence are composed of three broad categories of data, as shown in Example 2.1.1. The first is XSProc data, including Standard Composition Specification Data and Unit Cell Geometry Specification Data. This first category specifies the cross section library and then defines the composition of each mixture and optionally unit cell geometry that may be used to process the cross sections. This data block is necessary for the CSAS sequence.

Note: Sequence implementation determines the calculation (transport) mode automatically, either as multigroup or continuous-energy, by testing the cross section library whose name has been entered.

Warning: Continuous-energy mode does not process data entered in *celldata data* block(s).

The second category of data, the CSAS sequence input data, includes two new data blocks, **definitions data** and **tallies data**, for flexible tally definitions. These new blocks available in all CSAS sequences in SCALE 6.3 currently provide only accumulation of neutron flux, fission rate, and neutrons produced from fission on **different** energy and spatial grids. Although similar to capabilities activated with old-style KENO parameter input methods (with `GFX`, `CDS`, and `FIS` as described in Sect. 8.1.3.3), these input methods do not allow tallying the requested quantities on different energy and spatial grids. This limitation is relaxed with the new CSAS input blocks.

The third category of data, the KENO input data, is used to specify the geometric and boundary conditions that represent the physical 3D configuration of a KENO problem.

CSAS ensures data consistency among these three category of data. For example, it verifies that mixture numbers used in the KENO **geometry data** block must correspond to those defined either in the **composition data** or **celldata data** blocks. Note that in multigroup mode, a unique mixture number can be specified in the **celldata data** block by `CELLMIX=` if the cell is cell-weighted.

Note: As depicted in Example 2.1.1, a successful CSAS calculation requires at least a problem title and cross section library definitions, followed by **composition data** and **geometry data**. Depending on the requirements of the problem, other optional data blocks can be activated. Following CSAS5 input demonstrates this minimal requirement.

```
=csas5
sample problem 14  u metal cylinder in an annulus
ce_v7.1

read comp
   uranium  1 den=18.69 1 300  92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp

read geom
   global unit 1
      cylinder  1 1 8.89 10.109 0.0 orig 5.0799 0.0
```

(continues on next page)

(continued from previous page)

```
      cylinder  0 1 13.97 10.109 0.0
      cylinder  1 1 19.05 10.109 0.0
end geom
end data
end
```

Unlike the CSAS5 and CSAS6 versions in previous SCALE releases, in SCALE 6.3, user can enter all data blocks in any order in both CSAS5 and CSAS6 inputs. Following CSAS5 input illustrates this input flexibility.

```
=csas5
sample problem 14  u metal cylinder in an annulus
ce_v7.1

read geom
   global unit 1
      cylinder  1 1 8.89 10.109 0.0 orig 5.0799 0.0
      cylinder  0 1 13.97 10.109 0.0
      cylinder  1 1 19.05 10.109 0.0
end geom

read comp
   uranium  1 den=18.69 1 300  92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp

end data
end
```

All data are entered in free form, allowing alphanumeric data, floating-point data, and integer data to be entered in an unstructured manner. Up to 252 columns of data entry per line are allowed. Data can usually start or end in any column with a few exceptions. As an example, the word END beginning in column 1 and followed by two blank spaces or a new line will end the problem and any data following will be ignored. Each data entry must be followed by one or more blanks to terminate the data entry. For numeric data, either a comma or a blank can be used to terminate each data entry. Integers may be entered for floating-point values. For example, 10 will be interpreted as 10.0. Imbedded blanks are not allowed within a data entry unless an E precedes a single blank as in an unsigned exponent in a floating-point number. For example, 1.0E 4 would be correctly interpreted as 1.0×10^4.

The word "END" is a special data item. An "END" may have a name or label associated with it (e.g., "END DATA"). The name or label associated with an "END" is separated from the "END" by a single blank and is a maximum of 12 characters long. *At least two blanks or a new line MUST follow every labeled and unlabeled ``END``. It is the user's responsibility to ensure compliance with this restriction. Failure to observe this restriction can result in the use of incorrect or incomplete data without the benefit of warning or error messages.*

Multiple entries of the same data value can be achieved by specifying the number of times the data value is to be entered, followed by either `R`, `\*`, or `$`, followed by the data value to be repeated. Imbedded blanks are not allowed between the number of repeats and the repeat flag. For example, `5R12`, `5*12`, `5$12`, or `5R 12`, etc., will enter five successive 12's in the input data. Multiple zeros can be specified as nZ where n is the number of zeroes to be entered.

The purpose of this section is to define the input data in discrete subsections relating to a particular type of data. Tables of the input data are included in each subsection, and the entries are described in more detail in the appropriate sections.

Resonance-corrected cross sections are generated using the appropriate boundary conditions for the unit cell description (i.e., void for the outer surface of a single unit, white for the outer surface of an infinite array of cylinders). As many unit cells as needed may be specified in a problem. A unit cell is cell-weighted by using the keyword "CELLMIX=" followed by a unique user specified mixture number in the unit cell data.

To check the input data without actually processing the cross sections and without performing transport calculations, the sequence parameter options `PARM=CHECK` or `PARM=CHK` should be entered, as shown below.

```
=CSAS5 PARM=CHK

=CSAS6 PARM=CHK
```

This will cause the input data for CSAS to be checked and appropriate error messages to be printed. If plots are specified in the data, they will be printed. This feature allows the user to debug and verify the input data while using a minimum of computer time.

2.1.4.1 XSProc data

The XSProc reads the standard composition specification data and the unit cell geometry specifications. It then produces the mixing table and unit cell information necessary for processing the cross sections if needed. Sect. 7 of this manual provides a detailed description of the input data and processing options. CSAS sequences are responsible for passing data such that mixing table and problem-dependent cross sections from XSProc calculations are conveyed to the transport calculations. Note that reported elapsed time in each transport calculation does not include the time required to process and prepare multigroup cross section data. When running the transport module concurrently on multiple cores, these data are broadcasted to all instances of the transport module running on each computational node.

In contrast, in continuous-energy mode, only mixing table data generated from XSProc utilities are passed to the transport module. In addition to this, the defined continuous-energy data library is first verified by the utilities that exist in XSProc, and then temperature correction is applied to each nuclide data using the defined temperatures when loading these data from disk by each instance of the transport module. Therefore, elapsed time reported at the end of each transport module calculation also includes the time spent on temperature correction and data loading.

In multigroup mode, the XSProc calculation path for each unit cell is always determined with the following hierarchy to prepare the problem-dependent multigroup cross section data:

- All non-fissionable cells are processed with BONAMI by default, and this cannot be changed.
- All fissionable cells are processed with CENTRM by default, and this can be overridden by defining a cross section processing option with the sequence parameter option (PARM=BONAMI or PARM=2REGION).
- Double-het cells defined in the **celldata data** block are **always** processed with CENTRM, and this cannot be changed.

CSAS sequences always print a cross section processing summary of the cells used/defined in the problem. This can be seen in Sect. 2.1.5.3. See Sect. 7 of this manual for detailed description of these unit cell processing options.

2.1.4.2 CSAS input data

Two new data blocks, **definitions data** and **tallies data**, are currently supported by CSAS sequences to provide flexible tally definitions. These new data blocks are currently available to define the mesh responses flux, fission_density, and fission_source on different spatial and energy grids. This section introduces these two new data blocks and discusses the limitations with some details.

Definitions data

The **definitions data** input block allows (1) multiple spatial grids to be defined using the **gridGeometry data** blocks inside the definitions data block, and (2) multiple energy grids to be defined using the **energyBounds data** inside the **definitions data** block. The syntax for defining a **gridGeometry** inside a definitions block is the same as defining a standalone grid at the root level of input (i.e., KENO's **gridgeometry data** block). The syntax for defining energyBounds is already used for defining energy grids in the MAVRIC sequence. See Sect. 8.2 for further details.

In addition to specifying a list of individual energy group boundaries, equal-width energy bins and equal-width lethargy bins can be requested. Moreover, any available SCALE library structure can be requested (e.g., n56 for the group structure from SCALE's 56-group neutron library). CSAS5 allows using combinations of some of these input methods, as shown in the example given in Example 2.1.2.

Note: As shown in example given Example 2.1.2 `READ` keyword is not required when defining energy boundaries with **energyBounds** data block. This may show differences from one CSAS sequence to another.

Example 2.1.2: Typical spatial and energy grid specifications in the **definitions data** block

```
READ DEFINITIONS

read grid 1
   xlinear 30 -10 70
   ylinear 10 -20 60
   zlinear 50 -30 40
end grid

read grid 2
   numxcells=10 xmin=-18.5 xmax=+68.5
   numycells=25 ymin=-28.5 ymax=+58.5
   numzcells=10 zmin=-38.5 zmax=+48.5
end grid

'user specified energy grid
read energyBounds 1
   bounds 2e7 0.625 1e-5 end
end energyBounds

'user specified energy grid using equal-energy bins'
energyBounds 2
   linear=10 1e-5 2e7
end energyBounds

'user specified energy grid using equal-lethargy bins'
energyBounds 3
   logarithmic=10 1e-5 2e7
end energyBounds

'SCALE 56-group neutron structure with additional energy points'
energyBounds 10
```

(continues on next page)

(continued from previous page)

```
   n56
   bounds 1.1  0.11 0.011 0.0011 end
energyBounds

END DEFINITIONS
```

In continuous-energy mode, a special DEFAULT keyword allows modification of the default energy group structure that was previously defined with the NGP parameter and/or the KENO **energy data** block.

Note: The default energy group structure is currently acquired from the SCALE 252-group neutron library. This may be overridden by defining data with the NGP parameter, data in the KENO **energy data** block, or data in the **definitions data** block entered with DEFAULT keyword.

Caution: CSAS does not allow using **definitions data** block together with KENO NGP parameter and/or KENO **energy data** block.

Caution: CSAS does not allow using **definitions data** block together with KENO FIS, GFX, CDS, and MSH parameters.

In multigroup mode, **DEFAULT** energy boundaries are always obtained from the multigroup library used by KENO codes in the neutron transport calculation, and this cannot be overriden by a **DEFAULT** energy boundaries specification in the definitions data block. In other words, an energyBounds **DEFAULT** is not permitted in multigroup mode.

Warning: In multigroup mode, **DEFAULT** energy boundaries are always acquired from the library used by the KENO V.a transport, and it cannot be changed.

Caution: In multigroup calculations, energy points of the user-defined energy boundaries must be a subset of the energy points of the energy structure obtained from the multigroup library used by KENO transport. Otherwise, execution will be terminated and an appropriate error message is displayed.

The sample definitions data block given in Example 2.1.3 defines an energy grid labeled 1 and an energy grid labeled DEFAULT in the definitions data block.

Example 2.1.3: Definitions data block with DEFAULT energyBounds specification

```
READ DEFINITIONS

'user defined energy grid 1
  read energyBounds 1
     bounds 2e7 0.625 1e-5 end
```

(continues on next page)

(continued from previous page)

```
  end energyBounds

'user defined default energy grid
  energyBounds DEFAULT
     bounds    2e7 8.2e5 2.0e4 1.05e2 5.0 0.65 0.15 0.04 1.e-5 end
  end energyBounds

END DEFINITIONS
```

When using this definitions block in continuous-energy mode, KENO codes read DEFAULT energy boundaries from the definitions data and utilizes these data in all tally calculations (`energyBounds DEFAULT` overrides the current default that is acquired from the SCALE 252-group library) if requested otherwise in the tallies block for the supported mesh responses. The two energy boundaries read from definitions data are printed in KENO's energy boundaries edit in the output as shown in Fig. 2.1.1.

```
    ******************************************************************************************************************************
    ***                                                                                                                        ***
    ***                                          energy boundaries                                                             ***
    ***                                                                                                                        ***
    ***     2 sets of energy group boundaries are either specified or internally setup for this problem                        ***
    ***                                                                                                                        ***
    ******************************************************************************************************************************

                             **** Energy group boundaries utilized in this problem ****

 Energy Boundaries DEFAULT
     title: Acquired from the specification in 'read definition` block
      group   energy (eV)
     ------  ------------
          1   2.00000E+07
          2   8.20000E+05
          3   2.00000E+04
          4   1.05000E+02
          5   5.00000E+00
          6   6.50000E-01
          7   1.50000E-01
          8   4.00000E-02
              1.00000E-05
     ------  ------------

 Energy Boundaries 1
      group   energy (eV)
     ------  ------------
          1   2.00000E+07
          2   6.25000E-01
              1.00000E-05
     ------  ------------


***** warning ***** keno message number k6-410 follows:
energy boundaries marked as DEFAULT will be used in all tally calculations if otherwise not requested.
```

Fig. 2.1.1: Sample energy boundaries output edit when running CSAS with the above definitions data in the continuous-energy mode.

Unlike continuous-energy mode, when the data defined in the sample definition block given above are processed in mutigroup mode, reading the energy boundaries `DEFAULT` from the definitions data is ignored, and the user is notified with a warning message, as shown in Fig. 2.1.2. However, the calculation is terminated because the energy boundaries given with energy identifier 1 does not conform to the default energy boundaries acquired from the library used by KENO transport (in this test case, the SCALE 28-group neutron and 19-group gamma library was used). The corresponding error message is also shown in Fig. 2.1.2 printed to the output just before the code termination.

```
***** warning ***** csas message number cs-400 follows:
ignore reading energy group boundaries from definitions data that are marked as DEFAULT since this is a multi group
calculation.

...

***** error ***** keno message number k5-388 follows:
inconsistent energy intervals. energy boundaries specified in definitions data with the identification number  1
must be a sub set of the DEFAULT energy boundaries acquired from the multi group library used by transport.


***** warning ***** keno message number k5-378 follows:
in multi group mode, DEFAULT energy boundaries used for tallying are always obtained from the multi group library used by
transport.
```

Fig. 2.1.2: CSAS terminates execution with an error message when the definitions data given above are used in multigroup mode.

Tallies data

The new tallies data input allows mesh responses to be requested using any energy grid and/or spatial grid from the definitions block. Three response types shown in Table 2.1.1 were added as mesh tally options for CSAS. Note that the same responses can also be activated by `GFX`, `FIS`, and `CDS`, but only using the default energy boundaries.

Table 2.1.1: Mesh tallies available with **tallies data** block.

Description	Old KENO input method to activate the same tally	New response name in t
Neutron flux averaged over mesh volumes	`GFX`	`flux`
Fission rates per voxel volume	`FIS`	`fission_density`
Neutron production per voxel volume	`CDS`	`fission_source`

Note: Either input method (parameter input or tallies data) can be used to request the mesh tallies described in Table 2.1.1. It is recommended to request mesh tallies using the new response names (`flux`, `fission_density`, `fission_source`) with the tallies data block rather than the old-style parameter inputs (`GFX`, `FIS`, `CDS`) with the limited energy and spatial grid options.

A typical mesh tally input block is given in Example 2.1.4. Each spatial and energy grid used by each mesh tally must be defined in the definitions data block. Note that, as shown in Example 2.1.4, the same mesh response can be defined multiple times using different spatial and energy grids.

Example 2.1.4: Typical mesh tally specifications in tallies data

```
READ TALLIES

   read mesh 1
      response=FLUX
      grid=1
      energy= 1
   end mesh

   read mesh 2
```

(continues on next page)

(continued from previous page)

```
      response=FISSION_DENSITY
      grid=2
      energy=2
   end mesh

   read mesh 3
      response=FISSION_SOURCE
      grid=3
      energy=default
   end mesh

   read mesh 13
      response=FISSION_SOURCE
      grid=3
      energy=2
   end mesh

END TALLIES
```

The KENO codes in SCALE 6.3 support multiple sets of energy group boundaries for tallying purposes. A data container was designed to store all energy boundaries that are either set up by KENO for some internal use or specified by the user. Note that multiple sets of energy boundaries can be defined only by using the new definitions data block available in CSAS and TRITON sequences. In continuous-energy mode, KENO with the NGP parameter or data in energy data block provides only a single set of energy boundaries, and these always override KENO's default energy group boundaries used in all tallies.

After processing data entered in the definitions and tallies data blocks, KENO codes print the summary of all corresponding definitions in energy boundaries, grid definitions, and tally definitions output edits. The following sample input can be used to demonstrate the new output edits in KENO codes with continuous-energy mode:

```
read definitions

 read gridgeometry 11
    numxcells=2 numycells=2 numzcells=2
    xmin=-0.73 xmax=0.73
    ymin=-0.73 ymax=0.73
    zmin=0 zmax=10.0
 end gridgeometry

 read gridgeometry 12
    numxcells=2 numycells=2 numzcells=8
    xmin=-0.73 xmax=0.73
    ymin=-0.73 ymax=0.73
    zmin=0 zmax=10.0
 end gridgeometry

read gridgeometry 13
    numxcells=4 numycells=2 numzcells=4
    xmin=-0.73 xmax=0.73
    ymin=-0.73 ymax=0.73
    zmin=0 zmax=10.0
 end gridgeometry

 read energyBounds 12
   title "ebounds is a sub-set of 8 group MG test library"
   bounds
               2.00000E+07
               1.05000E+02
               5.00000E+00
               1.00000E-05
```

(continues on next page)

(continued from previous page)

```
   end
 end energyBounds

 read energyBounds DEFAULT
   title "SCALE 8 group test library structure"
   bounds
             2.00000E+07
             8.20000E+05
             2.00000E+04
             1.05000E+02
             5.00000E+00
             6.50000E-01
             1.50000E-01
             4.00000E-02
             1.00000E-05
   end
 end energyBounds

end definitions

read tallies
read mesh 1
  energy=DEFAULT
  grid=11
  response=FLUX
end mesh

read mesh 2
  energy=DEFAULT
  grid=12
  response=FLUX
end mesh

read mesh 3
  energy=12
  grid=13
  response=FLUX
end mesh

read mesh 100
  energy=12
  grid=12
  response=FISSION_DENSITY
end mesh

read mesh 200
  energy=DEFAULT
  grid=13
  response=FISSION_DENSITY
end mesh

read mesh 1000
  energy=12
  grid=11
  response=FISSION_SOURCE
end mesh

read mesh 1080
  energy=DEFAULT
  grid=13
  response=FISSION_SOURCE
end mesh

end tallies
```

The energy boundaries output edit depicted in Fig. 2.1.3 summarizes the data stored in the energy boundaries

data container. For the above sample problem, two sets of energy group boundaries are read from the definitions data and stored in the data container.

```
    ************************************************************************************************************************
    ***                                                                                                                  ***
    ***                                          energy boundaries                                                       ***
    ***                                                                                                                  ***
    ***     2 sets of energy group boundaries are either specified or internally setup for this problem                  ***
    ***                                                                                                                  ***
    ************************************************************************************************************************

                              **** Energy group boundaries utilized in this problem ****

 Energy Boundaries DEFAULT
     title: Acquired from the specification in 'read definition` block
      group   energy (eV)
     ------  ------------
          1   2.00000E+07
          2   8.20000E+05
          3   2.00000E+04
          4   1.05000E+02
          5   5.00000E+00
          6   6.50000E-01
          7   1.50000E-01
          8   4.00000E-02
              1.00000E-05
     ------  ------------

 Energy Boundaries 1
      group   energy (eV)
     ------  ------------
          1   2.00000E+07
          2   6.25000E-01
              1.00000E-05
     ------  ------------


***** warning ***** keno message number k5-410 follows:
energy boundaries marked as DEFAULT will be used in all tally calculations if otherwise not requested.
```

Fig. 2.1.3: Energy boundaries edit in KENO output

Another edit that was added to KENO's output is the grid definitions edit, which summarizes the mesh grids that were either defined by the user or automatically constructed by KENO for Shannon entropy tallies. The grid definitions output edit corresponds to the above provided sample input and is shown in Fig. 2.1.4. Note that Fig. 2.1.4 shows only a part of the mesh tallies output edit.

```
  ***                                                                                                                  ***
  ***                                          grid definitions                                                        ***
  ***                                                                                                                  ***
  ***     4 grid geometries have been defined (either read from user input or constructed for internal use)            ***
  ***                                                                                                                  ***
  ********************************************************************************************************************

                              **** Grid geometries utilized in this problem ****

Grid Geometry: 11
    title:
    Plane Summary
      x:    2 cells from -7.30001E-01 to  7.30001E-01
      y:    2 cells from -7.30001E-01 to  7.30001E-01
      z:    2 cells from -1.00000E-05 to  1.00000E+01
    Total number of cells: 8

                        x-planes              y-planes              z-planes
    -----  --------------------  --------------------  --------------------
        1  -7.30000729999998E-01  -7.30000729999998E-01  -9.99999997475243E-06
        2   0.00000000000000E+00   0.00000000000000E+00   5.00000000000000E+00
        3   7.30000729999998E-01   7.30000729999998E-01   1.00000100000000E+01
    -----  --------------------  --------------------  --------------------

Grid Geometry: 12
    title:
    Plane Summary
      x:    2 cells from -7.30001E-01 to  7.30001E-01
      y:    2 cells from -7.30001E-01 to  7.30001E-01
      z:    8 cells from -1.00000E-05 to  1.00000E+01
    Total number of cells: 32

                        x-planes              y-planes              z-planes
    -----  --------------------  --------------------  --------------------
        1  -7.30000729999998E-01  -7.30000729999998E-01  -9.99999997475243E-06
        2   0.00000000000000E+00   0.00000000000000E+00   1.25000000000000E+00
        3   7.30000729999998E-01   7.30000729999998E-01   2.50000000000000E+00
        4                                                 3.75000000000000E+00
        5                                                 5.00000000000000E+00
        6                                                 6.25000000000000E+00
        7                                                 7.50000000000000E+00
        8                                                 8.75000000000000E+00
        9                                                 1.00000100000000E+01
    -----  --------------------  --------------------  --------------------

Grid Geometry: 13
    title:
    Plane Summary
      x:    4 cells from -7.30001E-01 to  7.30001E-01
      y:    2 cells from -7.30001E-01 to  7.30001E-01
      z:    4 cells from -1.00000E-05 to  1.00000E+01
    Total number of cells: 32

                        x-planes              y-planes              z-planes
    -----  --------------------  --------------------  --------------------
        1  -7.30000729999998E-01  -7.30000729999998E-01  -9.99999997475243E-06
        2  -3.65000000000000E-01   0.00000000000000E+00   2.50000000000000E+00
        3   0.00000000000000E+00   7.30000729999998E-01   5.00000000000000E+00
        4   3.65000000000000E-01                          7.50000000000000E+00
        5   7.30000729999998E-01                          1.00000100000000E+01
    -----  --------------------  --------------------  --------------------

Grid Geometry: 10001
    title: Default 5 x 5 x 5 Cartesian mesh which overlays the entire geometry
    Plane Summary
      x:    5 cells from -7.30001E-01 to  7.30001E-01
      y:    5 cells from -7.30001E-01 to  7.30001E-01
      z:    5 cells from -1.00000E-06 to  1.00000E+01
    Total number of cells: 125

                        x-planes              y-planes              z-planes
    -----  --------------------  --------------------  --------------------
        1  -7.30000999999997E-01  -7.30000999999997E-01  -9.99999997475243E-07
        2  -4.38000599999998E-01  -4.38000599999998E-01   2.00000120000000E+00
        3  -1.46000199999999E-01  -1.46000199999999E-01   4.00000339999999E+00
        4   1.46000200000000E-01   1.46000200000000E-01   6.00000559999999E+00
```

Fig. 2.1.4: Grid definitions edit in KENO output

The tally definitions output edit summarizes the specifications of tallies defined in tallies block. Currently, only mesh tally edits are supported, and this is shown in Fig. 2.1.5 for the above sample input.

```
   ************************************************************************************************************************
   ***                                                                                                                  ***
   ***                                          tally definitions                                                       ***
   ***                                                                                                                  ***
   ***     7 mesh tallies have been defined                                                                             ***
   ***                                                                                                                  ***
   ************************************************************************************************************************

                            **** Mesh tallies defined for this problem ****


mesh tally 1
    response  : flux
    grid id   : 11
    energy id : Default
    output    : csas5_CE_multiple_mesh_tallies.meshtally_1_flux.3dmap

mesh tally 2
    response  : flux
    grid id   : 12
    energy id : Default
    output    : csas5_CE_multiple_mesh_tallies.meshtally_2_flux.3dmap

mesh tally 3
    response  : flux
    grid id   : 13
    energy id : 12
    output    : csas5_CE_multiple_mesh_tallies.meshtally_3_flux.3dmap

mesh tally 100
    response  : fission_density
    grid id   : 12
    energy id : 12
    output    : csas5_CE_multiple_mesh_tallies.meshtally_100_fission_density.3dmap

mesh tally 200
    response  : fission_density
    grid id   : 13
    energy id : Default
    output    : csas5_CE_multiple_mesh_tallies.meshtally_200_fission_density.3dmap

mesh tally 1000
    response  : fission_source
    grid id   : 11
    energy id : 12
    output    : csas5_CE_multiple_mesh_tallies.meshtally_1000_fission_source.3dmap

mesh tally 1080
    response  : fission_source
    grid id   : 13
    energy id : Default
    output    : csas5_CE_multiple_mesh_tallies.meshtally_1080_fission_source.3dmap
```

Fig. 2.1.5: Tally definitions edit in KENO output

After the calculations have been completed for all the requested tallies, KENO also prints another output table that summarizes the mesh tallies, as shown in Fig. 2.1.6. Other than the mesh tally input specifications, the mesh tallies output edit also summarizes the intervals of the energy and spatial grids used in tally calculations and approximate memory allocation required to compute and write this tally to 3dmap output file. Note that Fig. 2.1.6 shows only a part of the mesh tallies output edit.

```
                                                    **** mesh tallies ****
7 mesh tallies computed for this problem

 mesh tally 1
     response         : flux
     grid id          : 11
     energy id        : Default
     memory allocated : 0.001 MB
     output           : csas6_CE_multiple_mesh_tallies.meshtally_1_flux.3dmap

     energy boundaries:
          group   energy (eV)
         ------  ------------
              1   2.00000E+07
              2   8.20000E+05
              3   2.00000E+04
              .              .
              6   6.50000E-01
              7   1.50000E-01
              8   4.00000E-02
                  1.00000E-05
         ------  ------------

     grid summary:
           x:    2 cells from -7.30001E-01 to  7.30001E-01
           y:    2 cells from -7.30001E-01 to  7.30001E-01
           z:    2 cells from -1.00000E-05 to  1.00000E+01
         Total number of cells: 8

 mesh tally 2
     response         : flux
     grid id          : 12
     energy id        : Default
     memory allocated : 0.004 MB
     output           : csas6_CE_multiple_mesh_tallies.meshtally_2_flux.3dmap

     energy boundaries:
          group   energy (eV)
         ------  ------------
              1   2.00000E+07
              2   8.20000E+05
              3   2.00000E+04
              .              .
              6   6.50000E-01
              7   1.50000E-01
              8   4.00000E-02
                  1.00000E-05
         ------  ------------

     grid summary:
           x:    2 cells from -7.30001E-01 to  7.30001E-01
           y:    2 cells from -7.30001E-01 to  7.30001E-01
           z:    8 cells from -1.00000E-05 to  1.00000E+01
         Total number of cells: 32

 mesh tally 3
     response         : flux
     grid id          : 13
     energy id        : 12
     memory allocated : 0.001 MB
     output           : csas6_CE_multiple_mesh_tallies.meshtally_3_flux.3dmap
```

Fig. 2.1.6: Mesh tallies edit in KENO output

See the relevant subsections in Sect. 8.1.5 for further details for all these output edits.

Mesh tally output files

Depending on the user input specifications, the naming of the mesh tally 3dmap output files show some variations. Table 2.1.2 lists the 3dmap output filenames for each response type if only a single tally was requested for each response type. And, Table 2.1.3 lists the 3dmap output filenames for each response type if multiple mesh tallies are requested with the same response type. In such a case, the output filenames are updated with the keyword meshtally followed by the mesh id (mesh identifier used to define each mesh in tallies data).

Table 2.1.2: Mesh tally 3dmap file naming when a single response is requested

response	3dmap file name
`flux`	`${BASENAME}.flux.3dmap`
`fission_density`	`${BASENAME}.fission_density.3dmap`
`fission_source`	`${BASENAME}.fission_source.3dmap`

Note: Mesh tallies activated with old-style input method (using the `GFX`, `CDS`, and `FIS` parameters) also use the definitions for 3dmap file naming given in Table 2.1.2.

Table 2.1.3: Mesh tally 3dmap file naming when a response is requested multiple times

response	3dmap file name
`flux`	`${BASENAME}.meshtally_${MESHID}_flux.3dmap`
`fission_density`	`${BASENAME}.meshtally_${MESHID}_fission_density.3dmap`
`fission_source`	`${BASENAME}.meshtally_${MESHID}_fission_source.3dmap`

2.1.4.3 KENO data

Table 2.1.4 contains the outline for the KENO input. A typical KENO input is divided into 13 data blocks. A brief outline of commonly used data blocks is shown in Table 2.1.4. Note that parameter data must precede all other KENO data blocks when running standalone KENO codes; however, this is not applied to the KENO calculations performed as part of each CSAS sequence. As described in above sections, a minimal CSAS input always requires **geometry data**, and KENO data blocks listed in Table 2.1.4 can be defined in any order.

Information on all KENO input is provided in Sect. 8.1 of this document and will not be repeated here.

Table 2.1.4: Outline of KENO data

Type of data	**Starting flag**	**Comments**	**Termination flag**
Parameters	READ PARAMETER	Enter desired parameter data	END PARAMETER
Geometry	READ GEOMETRY [1]	Enter desired geometry data Always required by CSAS	END GEOMETRY
Array data	READ ARRAY [1]	Enter desired array data	END ARRAY

continues on next page

Table 2.1.4 – continued from previous page

Type of data	Starting flag	Comments	Termination flag
Boundary conditions	READ BOUNDS	Enter desired boundary conditions	END BOUNDS
Volume data	READ VOLUME	Enter desired volume data (KENO-VI only)	END VOLUME
Energy group boundaries	READ ENERGY	Enter desired neutron energy group boundaries	END ENERGY
Start data or initial source	READ START	Enter desired start data	END START
Plot data	READ PLOT	Enter desired plot data	END PLOT
Grid geometry data	READ GRID	Enter desired mesh data	END GRID
Reaction	READ REACTION	Enter desire reaction tallies (CE mode only)	END REACTION
KENO data terminus	END DATA	Enter to signal the end of all KENO. data	
[1] Geometry input is different for both KENO V.a and KENO-VI See Sect. 8.1.3.4 for further details.			

Note: Unlike standalone KENO calculations, each KENO data block can be entered in any order when running KENO codes as part of CSAS sequence.

2.1.5 DESCRIPTION OF OUTPUT

This section contains a brief description and explanation of the CSAS sequence. As CSAS was designed as a SCALE control module/sequence its own output is minimal. To avoid duplicate output edits, it suppresses the output from KENO Data processor except a few diagnostic and warning messages while processing the KENO data blocks. Because the KENO Data processor and KENO codes produce the same output edits for some input data, capturing both output sections and keeping printing them may result in duplicate information in the output sections for those input data.

CSAS always captures the XSProc and KENO outputs and prints them in the code output. Because these output sections are described and their details are discussed in Sect. 8.1.5 and Sect. 7.1.1 and relevant XSProc sections, they will not be described in this section.

When CSAS is run with PARM=CHECK, only outputs from KENO Data processor and XSProc input processor are shown in the code output.

The sample output sections presented in this section were from one of the calculations performed by CSAS5. Here, only CSAS5 examples are given to prevent repetition because CSAS6 prints the same tables in the same format with the same content.

2.1.5.1 Program verification information

After the header page, program verification information is printed that lists the name of the program and the revision number. The job name, date, and time of execution are also printed as shown in Fig. 2.1.7. This information may be used for quality assurance purposes.

```
                                        program verification information
                                      code system: SCALE   version: 6.3.0

                     program:  csas6
                     version:  6.3.0
                    username:  5kq
                    hostname:  kbb-mac

           date of execution:  2021-10-11
           time of execution:  11:02:39
```

Fig. 2.1.7: Sample program verification table.

2.1.5.2 Mixture table

The first table printed by CSAS codes lists the compositions read and processed from the data entered in the **composition data** block. Basically, this table echos what user defined in the **composition data** block; data for each mixture are printed. First the mixture number, density, and temperature are printed, followed by a table of the nuclides which make up the mixture. This table contains the following data: mixture ID number, nuclide ZA number, atom density and temperature. A sample mixture table is shown in Fig. 2.1.8.

```
 Mixture table for component comps at line:5, column:1
Mixture = 1 with density(g/cc) = 18.7600 and temperature(K) = 300.00
        Entry      Mixture      Nuclide     Density (atoms/b-cm)     Temperature (K)
            1            1        92234             4.827163e-04              300.00
            2            1        92235             4.479718e-02              300.00
            3            1        92236             9.572338e-05              300.00
            4            1        92238             2.657677e-03              300.00
        Entry      Mixture      Nuclide     Density (atoms/b-cm)     Temperature (K)
 End of mixture table for component comps at line:5, column:1
```

Fig. 2.1.8: An example of a mixture table.

Note: This output table prints only all mixtures read and processed from the composition data block. Any mixture defined with `CELLMIX` in celldata block is not printed here.

Note: The mixing table printed in KENO output may not reflect the mixture properties listed in this output table. Any mixture which is defined in composition data block but not used in KENO transport process will not be printed in KENO mixing table data edits. KENO also prints the mixture data defined with `CELLMIX` or defined in **Double-het** cell treatment in KENO mixing table data edits in the output. See Sect. 8.1.3.10 for further details about the KENO mixing data.

2.1.5.3 Cross section processing summary

In multigroup mode, cross section processing calculation path with XSProc show some differences depending on the type of the unit cells being processed and/or desired calculation methodology defined by user as discussed in Sect. 2.1.4.1. CSAS sequences summarize which of the XSProc calculation path is used when processing the unit cells in XSProc in the output.

A typical cross section processing summary table printed by a CSAS sequence in the code output is shown in Fig. 2.1.9.

The first record printed in this table is the multigroup cross section library which will be used in the calculations. This is followed by the cross section processing summary of the unit cells for this problem. This table includes the total number of unit cells being processed, and the number of unit cells processed with CENTRM and BONAMI calculations path. The last record printed in this table is the total elapsed time to process the XSProc data and build all the unit cells for the subsequent XSProc calculations.

```
Loaded MG Library: /ornldev/code/Scale/S63/INSTALL/G_63_b1500/data/scale.rev05.xn252v7.1

-------------------------------------------------------------------------------------------------
 Cross section processing summary of the cells:
-------------------------------------------------------------------------------------------------

 Total 7 cells are being processed:
      1 *default cells* (non-fissionable) processed with BONAMI.
      6 cells specified in CELLDATA block processed with CENTRM (sequence default option).

 NOTE: *default cells* are the cells not specified in CELLDATA block (They are processed as
       infinite homogeneous cells).

-------------------------------------------------------------------------------------------------

SequenceInputBuilder prepared 7 unit cells in 0.25 seconds.
```

Fig. 2.1.9: Summary of cross section processing.

Caution: CSAS sequence always creates a unit cell for all the mixtures defined in the composition data block and stores them in a cell container. Then, XSProc cross section processing is applied to all the unit cells stored in the cell container regardless of whether they are used in KENO transport calculation. Performing cross section processing for the unused mixtures, especially fissile mixtures, might waste the allocated computational resources for this calculation.

2.1.6 WARNING AND ERROR MESSAGES

CSAS sequence contains two types of warning and error messages. The first type of messages are from XSProc and SCALE sequence implementation which are common to many of the SCALE sequences. The second type of messages are mainly from the KENO Data processor as part of CSAS sequence implementation, and identified by CS- followed by a number. The details of the messages from KENO Data processor can be seen in Sect. 8.1.6.

Warning messages appear when a possible error is encountered. It is the responsibility of the user to verify whether the data are correct when a warning message is encountered. The functional modules, XSProc and KENO, activated by CSAS sequences will be executed if no error messages are generated and a warning message has been generated.

When an error is recognized, an error message is written and an error flag is set so the functional modules will not be activated. the code stops immediately if the error is too severe to allow continuation of input. However, it will continue to read and check the data if it is able. When the data reading is completed, execution is terminated if an error flag was set when the data were being processed. If the error flag has not been set, execution continues. When error messages are present in the output, the user should focus on the first error message, because subsequent messages may have been caused by the error that generated the first message.

The messages listed below complement the messages, which are from KENO Data processor, listed in KENO manual section, Sect. 8.1.6.

> CS-21 A UNIT NUMBER WAS ENTERED FOR THE CROSS-SECTION LIBRARY. (LIB= IN PARAMETER DATA.) THE DEFAULT VALUE SHOULD BE USED IN ORDER TO UTILIZE THE CROSS SECTIONS GENERATED BY CSAS. MAKE CERTAIN THE CORRECT CROSS-SECTION LIBRARY IS BEING USED.

This message is from subroutine CPARAM. It indicates that a value has been entered for the cross-section library in the KENO V.a parameter data. The cross-section library created by the analytical sequence should be used. *MAKE CERTAIN THAT THE CORRECT CROSS SECTIONS ARE BEING USED.*

> CS-55 *** ERRORS WERE ENCOUNTERED IN PROCESSING THE KENO DATA. EXECUTION IS IMPOSSIBLE. ***

This message from subroutine SASSY is printed if errors were found in the KENO input data for CSAS. When the data reading and checking have been completed, the problem will terminate without executing. Check the printout to locate the errors responsible for this message.

> CS-62 *** ERROR *** MIXTURE _____ IN THE GEOMETRY WAS NOT CREATED IN THE STANDARD COMPOSITIONS SPECIFICATION DATA.

This message from subroutine MIXCHK indicates that a mixture specified in the KENO geometry was not created in the standard composition data.

> CS-68 *** ERROR *** AN INPUT DATA ERROR HAS BEEN ENCOUNTERED IN THE _____ DATA ENTERED FOR THIS PROBLEM.

This message from the main program, CSAS, is printed if the subroutine library routine LRDERR returns a value of "TRUE," indicating that a reading error has been encountered in the "KENO PARAMETER" data. The appropriate data type is printed in the message. Locate the unnumbered message stating "ERROR IN INPUT. CARD IMAGE PRINTED ON NEXT LINE". Correct the data and resubmit the problem.

> CS-69 ***ERROR*** MIXTURE _____ IS AN INAPPROPRIATE MIXTURE NUMBER FOR USE IN THE KENO GEOMETRY DATA BECAUSE IT IS A COMPONENT OF THE CELL-WEIGHTED MIXTURE CREATED BY XSDRNPM.

This message from subroutine CMXCHK indicates that a mixture that is a component of a cell-weighted mixture has been used in the KENO geometry data.

> CS-100 *** ERROR *** THIS PROBLEM WILL NOT BE RUN BECAUSE ERRORS WERE ENCOUNTERED IN THE INPUT DATA.

This self-explanatory message indicates that an error occurred in input processing. User should examine the printout to locate the error or errors in the input data. Correct them and resubmit the problem.

2.1.7 SAMPLE PROBLEMS

This section contains example problems to demonstrate some of capabilities available in CSAS with KENO codes. A brief problem description and the associated input data for multigroup mode of calculation are included for each problem. The same sample problems may be executed in the continuous energy mode by changing the library name from `v7.1-252` to `ce_v7.1`. The complete list of libraries distributed with SCALE is provided in the Nuclear Data Libraries chapter.

2.1.7.1 CSAS5 sample problems

This section contains sample problems to demonstrate some of the options available in CSAS5. Note that sample problem 8 does not run in continuous-energy mode because they use CELLMIX or DOUBLEHET cell type.

CSAS5 sample problem 1: k_{eff} calculation

The purpose of this problem is to calculate the k-effective of a system. This problem is the same as the KENO V.a sample problem 12 in Appendix B except the cross-section library and KENO V.a mixing table are prepared by CSAS. The problem represents a critical experiment consisting of a composite array [CSAS5Tho64, CSAS5Tho73] of four highly-enriched (93.2%) uranium metal cylinders having a density of 18.76 g/cc and four 5.0677-L Plexiglas containers filled with uranyl nitrate solution. The uranium metal cylinders have a radius of 5.748 cm and a height of 10.765 cm. The uranyl nitrate solution has a specific gravity of 1.555 and contains 415 g of uranium per liter. The ID of the Plexiglas bottle is 19.05 cm and the inside height is 17.78 cm. The Plexiglas is 0.635 cm thick. The center-to-center spacing between the metal units is 13.18 cm in the Y direction and 13.45 cm in the Z direction. The center-to-center spacing between the solution units is 21.75 cm in the Y direction and 20.48 cm in the Z direction. The spacing between the Y-Z plane that passes through the centers of the metal units and the Y-Z plane that passes through the centers of the solution units is 17.465 cm in the X direction.

The metal units in this experiment are designated in Table II of [CSAS5Tho64] as cylinder index 11 and reflector index 1. A photograph of the experiment, Fig. 9 in [CSAS5Tho73], is given in Fig. 2.1.10.

```
=csas5   parm=(centrm)
sample problem set up 4aqueous 4 metal in csas5
v7.1-252
read composition
  uranium        1 0.985 300.  92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  solution
     mix=2
     rho[uo2(no3)2]= 415. 92235 92.6 92238 5.9 92234 1.0  92236 0.5
     molar[hno3]=9.783-3
     temperature=300
  end solution
  plexiglass     3 end
end composition
read param
```

(continues on next page)

(continued from previous page)

```
  flx=yes fdn=yes nub=yes htm=no
end param
read geom
  unit 1
    com='uranyl nitrate solution in a plexiglas container'
    cylinder  2 1 9.525 2p8.89
    cylinder  3 1 10.16 2p9.525
    cuboid  0 1 4p10.875 2p10.24
  unit 2
    com='uranium metal cylinder'
    cylinder  1 1 5.748 2p5.3825
    cuboid  0 1 4p6.59 2p6.225
  unit 3
    com='1x2x2 array of solution units'
    array 1 3*0.0
  unit 4
    com='1x2x2 array of metal units padded to match solution array'
    array 2 3*0.0
    replicate 0 1 2*0.0 2*8.57 2*8.03 1
  global unit 5
    array 3 3*0.0
end geom
read array
  ara=1 nux=1 nuy=2 nuz=2 fill f1 end fill
  ara=2 nux=1 nuy=2 nuz=2 fill f2 end fill
  gbl=3 ara=3 nux=2 nuy=1 nuz=1
  com='composite array of solution and metal units'
  fill 4 3 end fill
end array
end data
end
```

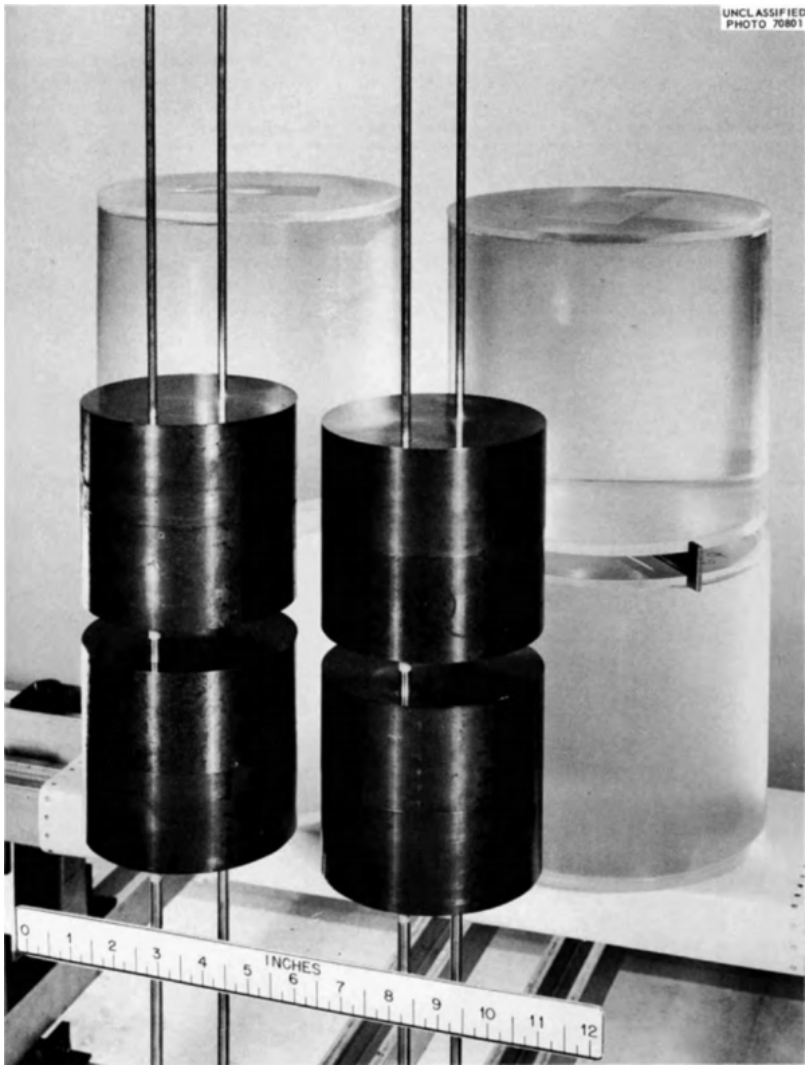


Fig. 2.1.10: Critical assembly of four solution units and four metal units.

CSAS5 Sample problem 8: k_∞ for a pebble bed fuel

This problem demonstrates setting up a fuel pebble from a pebble bed reactor, and calculating its k_∞. The pebble consists of a fuel grain of UO_2 0.025 cm in radius, coated with 0.003 cm of pyrolytic carbon, a further coat of 0.0035 cm thick silicon carbide, with a final coat of 0.004 cm thick pyrolytic carbon. 15000 grains are packed with graphite into an internal fuel sphere of 2.5 cm radius clad with a 0.5 cm thick covering of carbon and surrounded by helium. The fuel is 8.2% enriched ^{235}U. The pebbles are stacked into an infinite square pitched array with a pitch of 6 cm.

This problem uses DOUBLEHET cell type, which is applicable only in the multigroup mode of KENO calculations. Therefore, the continuous energy version of this problem will end with an error message.

```
=csas5            parm=(centrm)
infinite array of pebbles on a square pitch
v7.1-252
read composition
' fuel kernel
```

(continues on next page)

(continued from previous page)

```
  u-238  1 0 2.12877e-2 293.6 end
  u-235  1 0 1.92585e-3 293.6 end
  o      1 0 4.64272e-2 293.6 end
' inner pyro carbon
  c      3 0 9.52621e-2 293.6 end
' silicon carbide
  c      4 0 4.77240e-2 293.6 end
  si     4 0 4.77240e-2 293.6 end
' outer pyro carbon
  c      5 0 9.52621e-2 293.6 end
' graphite matrix
  c      6 0 8.77414e-2 293.6 end
' carbon pebble outer coating
  c      7 0 8.77414e-2 293.6 end
  he-3    8 0 3.71220e-11 293.6 end
  he-4    8 0 2.65156e-5 293.6 end
end composition
read celldata
  doublehet fuelmix=10 end
   gfr=0.025  1 coatt=0.004 3 coatt=0.0035 4 coatt=0.004 5
   matrix=6 numpar=15000 end grain
   centrm data
     ixprt=1 isn=8 nprt=2
   end centrm
  pebble sphsquarep right_bdy=white hpitch=3.0 8 fuelr=2.5 cladr=3.0 7 end
  centrm data
    ixprt=1 isn=8 nprt=2
  end centrm
end celldata
read param
  gen=210 npg=1000 htm=no
end param
read bounds
  all=mirror
end bounds
read geom
  global unit 1
    sphere   10 1 2.5
    sphere    7 1 3.0
    cuboid    8 1 6p3.0
end geom
end data
end
```

2.1.7.2 CSAS6 sample problems

This section contains sample problems to demonstrate some of the options available in CSAS6. A brief problem description and the associated input data for multigroup mode of calculation are included for each problem. The same sample problems may be executed in continuous-energy mode by changing the library name to an continuous-energy library. See Appendix A (Sect. 2.3) for additional examples.

CSAS6 Sample problem 1: Aluminum 30 Degree Pipe Angle Intersection

The purpose of this problem is to calculate the k-effective of a system composed of intersecting aluminum pipes, in the shape of a Y, filled with a 5% enriched UO_2F_2 solution. The UO_2F_2 solution at 299 K contains 907.0 gm/l of uranium, no excess acid, and has a specific gravity of 2.0289 gm/cm^3. The assembly is composed of a 212.1 cm long vertical pipe and a second pipe that intersects the vertical pipe 76.7 cm from the outside bottom at an angle of 29.26 degrees with the upper vertical pipe. Both pipes have 13.95 cm inner diameters and 14.11 cm outer diameters. The vertical pipe is open on the top and 1.3 cm thick on the bottom. The Y-leg pipe, in the YZ-plane, is 126.04 cm in length with the sealed end 0.64 cm thick. The assembly is filled with solution to a height 129.5 cm above the outside bottom of the vertical pipe. From the point where

the pipes intersect, the assembly is surrounded by water 37.0 cm in the ±X directions, 100 cm in the +Y direction, -37 cm in the -Y direction, to the top of the assembly in the +Z direction, and -99.6 cm in the -Z direction.

Fig. 2.1.11: Critical assembly of UO_2F_2 solution in a 30°-Y aluminum pipe.

```
=csas6
sample problem 1  Y-30, 5%uo2f2, 907.0g/l, 128.2, soln. ht.
v7.1-252
read comp
  solution
    mix=1
    rho[uo2f2]=907.0 92235 5.0 92238 95.0
    density=?
    temperature=299.0
  end solution
  al        2 1.0 end
  h2o       3 1.0 end
end comp
read parameters
  flx=yes fdn=yes far=yes pgm=yes plt=yes
end parameters
read start
  nst=6 tfx=0.0 tfy=0.0 tfz=0.0 lnu=1000
end start
read geometry
  global
  unit 1
    com='30 deg y'
    cylinder 10  13.95  135.4 -75.4
    cylinder 20  14.11  135.4 -76.7
    cylinder 30  13.95  125.4   0.0   rotate a2=-29.26
    cylinder 40  14.11  126.04  0.0   rotate a2=-29.26
    cuboid   50  2p37.0 100. -37.0 52.8 -75.4
    cuboid   60  2p37.0 100. -37.0 135.4 -99.6
    media  1  1 10  50
    media  2  1 20 -10 -30
    media  1  1 30  50 -10
    media  2  1 40 -30 -20
    media  0  1 10 -50
    media  0  1 30 -50 -10
    media  3  1 60 -20 -40 -10
    boundary  60
end geometry
read volume
    type=random  batches=1000
end volume
read plot
  scr=yes  lpi=10
  ttl='y-z slice at x=0.0  through centerline of both pipes'
  xul=0.0  yul=-39.0  zul=137.0
  xlr=0.0  ylr=105.0  zlr=-105.0
  vax=1    wdn=-1
  nax=400  end plt0
  ttl='x-y slice at z=26.0  slightly above point of separation'
  xul=-40.0 yul=105.0  zul=26.0
  xlr=+40.0 ylr=-40.0  zlr=26.0
  uax=1     vdn=-1
  nax=400 end plt1
  ttl='x-y slice at z=75.0  well above point of separation'
  xul=-40.0 yul=105.0  zul=75.0
  xlr=+40.0 ylr=-40.0  zlr=75.0
  uax=1     vdn=-1
  nax=400 end plt2
end plot
end data
end
```

CSAS6 Sample problem 2: Plexiglas Cross

The purpose of this problem is to calculate the k-effective of a system composed of intersecting Plexiglas pipes, in the shape of a cross, filled with a 5% enriched UO_2F_2 solution. The room temperature UO_2F_2 solution contains 896.1 gm/l of uranium, no excess acid, and has a specific gravity of 2.015 gm/cm^3. The pipes have a 13.335 cm inner diameter and 16.19 cm outer diameter. The vertical pipe is open on the top and 3.17 cm thick on the bottom. The horizontal pipe ends are 3.17 thick. The vertical pipe is 210.19 cm in length and filled with solution to a height of 117.2 cm. The two horizontal legs, positioned in the XZ-plane, intersect the vertical pipe 91.44 cm from the outside bottom at an 89 degree angle with the upper section of the pipe. Each horizontal is 91.44 cm in length and filled with the above specified UO_2F_2 solution. A water reflector surrounding the solution filled pipes extends out from the point where the pipes intersect 111.76 cm in the ±X directions, 20.64 cm in the ±Y directions, 29.03 cm in the +Z direction, and -118.428 cm in the -Z direction.

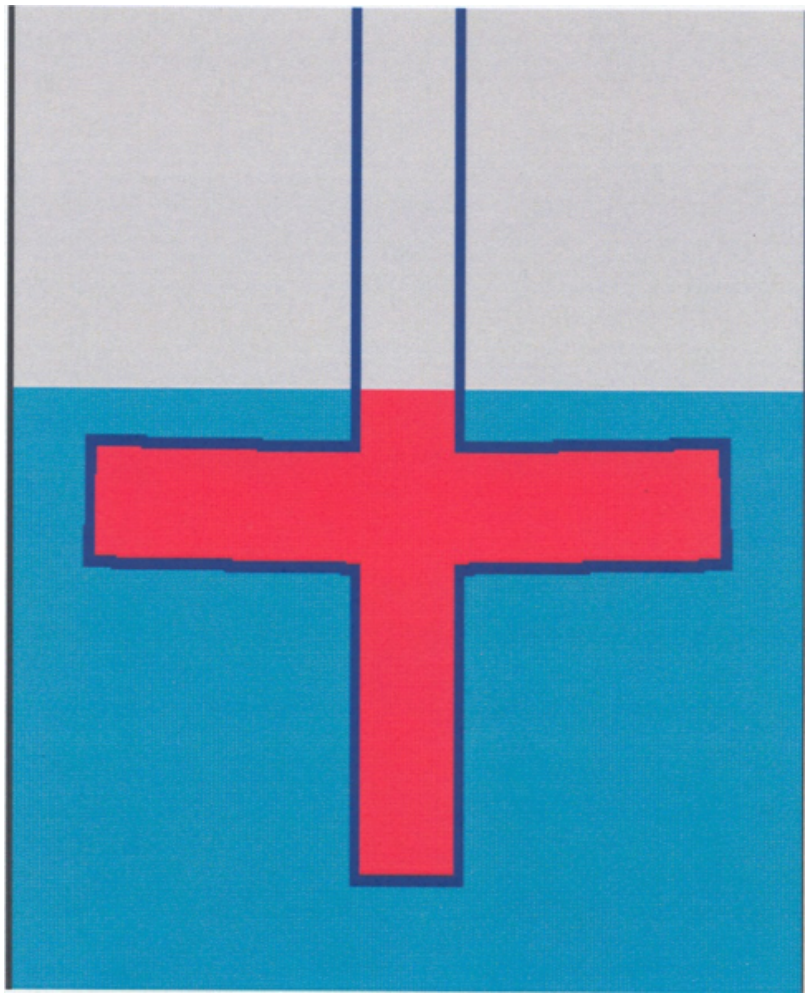


Fig. 2.1.12: Critical assembly of UO_2F_2 solution in a Plexiglas cross.

```
=csas6
sample problem 2  89-cross, 5% uo2f2 soln, plexiglass pipes, h2o refl.
v7.1-252
read comp
  solution
```

(continues on next page)

(continued from previous page)

```
    mix=1
    rho[uo2f2]=896.1 92235 5.0 92238 95.0
    density=?
    temperature=298.0
  end solution
  plexiglass 3 1.0 end
  h2o        2 1.0 end
end comp
read param
  plt=yes
end param
read geom
  global unit 1
    cylinder  10  13.335 28.93 -88.27
    cylinder  20  13.335 121.92 -88.27
    cylinder  30  16.19 121.92 -91.44
    cylinder  40  13.335 88.27 0.0 rotate a1=90. a2=89.
    cylinder  50  16.19  91.44 0.0 rotate a1=90. a2=89.
    cylinder  60  13.335 88.27 0.0 rotate a1=-90. a2=89.
    cylinder  70  16.19  91.44 0.0 rotate a1=-90. a2=89.
    cuboid    80  2p111.74 2p20.64 29.03 -118.428
    cuboid    90  2p111.74 2p40.64 121.92 -118.428
    media 1 1 10
    media 0 1 20 -10
    media 3 1 30 -10 -20 -50 -70
    media 1 1 40 -10 -20
    media 3 1 50 -40 -10 -20
    media 1 1 60 -10 -20
    media 3 1 70 -60 -10 -20 -50
    media 2 1 80 -10 -20 -30 -40 -50 -60 -70
    media 0 1 90 -20 -30 -80
    boundary  90
end geom
read volume
   type=trace
end volume
read start
  nst=6  tfx=0. tfy=0. tfz=0. lnu=1000
end start
read plot
  scr=yes  lpi=10
  ttl=' x-z slice at y=0.0 '
  xul=-113. yul=0. zul=  48.
  xlr= 113. ylr=0. zlr=-120.
  uax=1.0   wdn=-1.0
  nax=400  end plt0
  ttl=' y-z slce at x=0.0 '
  xul=0.    yul=-42. zul= 122.
  xlr=0.    ylr= 42. zlr=-120.
  vax=1.0   wdn=-1.0
  nax=400  end plt1
  ttl=' x-y slice at z=0.0 '
  xul=-113.0 yul= 42. zul=0.
  xlr= 113.0 ylr=-42. zlr=0.
  uax=1.0    vdn=-1.0
  nax=400  end plt2
end plot
end data
end
```

CSAS6 Sample problem 3: Sphere

This problem models an assembly consisting of a 93.2% enriched bare uranium sphere, 8.741 cm in radius, having a density of 18.76 gm/cm^3. Problem 3 models the assembly as a single bare sphere. The second problem models the assembly as a hemisphere with mirror reflection on the flat surface. The next three problems model the sphere using chords. This set of four problems is designed to illustrate the use of multiple chords in a problem.

```
=csas6
sample problem 3  bare 93.2% enriched uranium sphere
v7.1-252
read comp
  uranium  1 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp
read geometry
  global unit 1
    sphere 10  8.741
    cuboid   20  6p8.741
    media  1 1 10      vol=2797.5121
    media  0 1 20 -10  vol=2545.3424
    boundary 20
end geometry
end data
end
```

CSAS6 Sample problem 4: Sphere Models Using Chords and Mirror Albedos

This problem models an assembly consisting of a 93.2% enriched bare uranium sphere, 8.741 cm in radius, having a density of 18.76 gm/cm^3. The problem models the assembly as a hemisphere with mirror reflection on the flat surface.

```
=csas6
sample problem 4  bare 93.2% U sphere, hemisphere w/ mirror albedo
v7.1-252
read comp
  uranium  1 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp
read geometry
  global unit 1
    sphere 10  8.741  chord +x=0.0
    cuboid   20  8.741 0.0  8.741 -8.741 8.741 -8.741
    media  1 1 10      vol=2797.5121
    media  0 1 20 -10  vol=2545.3424
    boundary 20
end geometry
read bounds
  -xb=mirror
end bounds
end data
end
```

CSAS6 Sample problem 5: Sphere Models Using Chords and Mirror Albedos

This problem models an assembly consisting of a 93.2% enriched bare uranium sphere, 8.741 cm in radius, having a density of 18.76 gm/cm^3. The problem models the assembly as a quarter sphere with mirror reflection on the two flat surfaces.

```
=csas6
sample problem 5  bare 93.2% U sphere, quarter sphere w/ mirror albedo
v7.1-252
```

(continues on next page)

(continued from previous page)

```
read comp
  uranium  1 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp
read geometry
  global unit 1
    sphere 10  8.741  chord +x=0.0  chord  +y=0.0
    cuboid   20  8.741 0.0 8.741 0.0 8.741 -8.741
    media  1 1 10      vol=2797.5121
    media  0 1 20 -10  vol=2545.3424
    boundary 20
end geometry
read bounds
  -xy=mirror
end bounds
end data
end
```

CSAS6 Sample problem 6: Sphere Models Using Chords and Mirror Albedos (Eighth Sphere)

This problem models an assembly consisting of a 93.2% enriched bare uranium sphere, 8.741 cm in radius, having a density of 18.76 gm/cm^3. The problem models the assembly as an eighth sphere with mirror reflection on the three flat surfaces.

```
=csas6
sample problem 6  bare 93.2% U sphere, eighth sphere w/ mirror albedo
v7.1-252
read comp
  uranium  1 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp
read geometry
  global unit 1
    sphere 10  8.741  chord +x=0.0  chord  +y=0.0  chord +z=0.0
    cuboid   20  8.741 0.0 8.741 0.0 8.741 0.0
    media  1 1 10      vol=2797.5121
    media  0 1 20 -10  vol=2545.3424
    boundary 20
end geometry
read bounds
  -fc=mirror
end bounds
end data
end
```

CSAS6 Sample problem 7: Grotesque without the Diaphragm

The purpose of this problem is to calculate the k_{eff} of a system composed of eight enriched uranium units placed on a diaphragm, with an irregularly shaped centerpiece positioned in the center hole of the diaphragm [CSAS5Mih99]. The assembly and centerpiece are shown in Fig. 2.1.13, which is Fig. 4 from [CSAS5Mih99]. The eight units consist of an approximate parallelepiped with an irregular top, a parallelepiped, and six cylinders of various sizes. The centerpiece, which penetrates the hole in the diaphragm, consists of a cylinder topped by a parallelepiped topped by a hemisphere. The diaphragm is not modeled in this example.

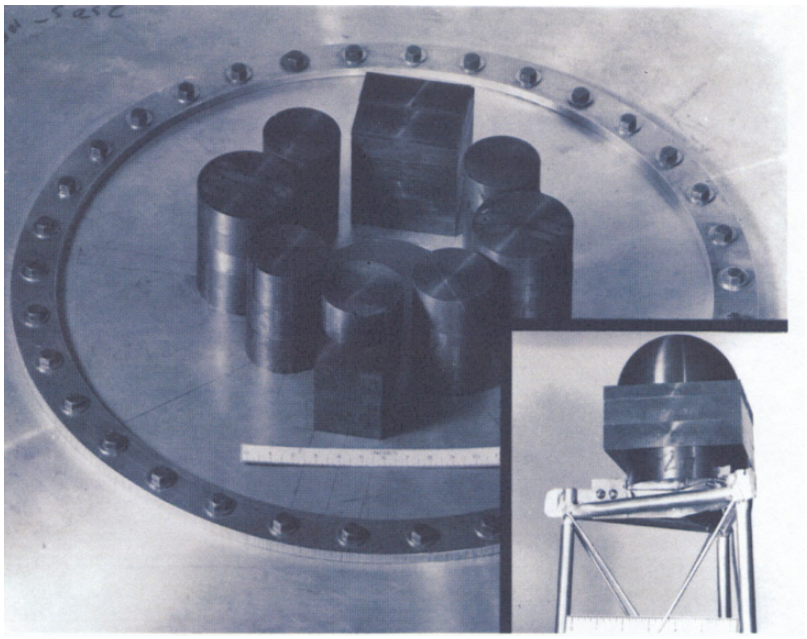


Fig. 2.1.13: Grotesque experimental setup.

```
=csas6
sample problem 7  keno-vi grotesque w/o diaphragm, ornl/csd/tm-220
v7.1-252
read comp
  uranium   1 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium   2 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium   3 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium   4 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium   5 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium   6 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium   7 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium   8 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium   9 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium  10 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium  11 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium  12 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium  13 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  uranium  14 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp
read param
  pgm=yes plt=yes
end param
read geom
  global unit 1
'*** one through three is item 1 in drawing 84-10649 ornl/csd/tm-220 ***
'one    top piece of item 1
  cuboid 10  2p6.3515 1.2685 -3.8115 13.377 13.058 origin y=-17.464 z=0.15 rotate a2=-1.35
'two    middle piece of item 1
  cuboid 20  2p6.3515 6.3515 -3.8115 13.058 11.155 origin y=-17.464 z=0.15 rotate a2=-1.35
'three  bottom piece of item 1
  cuboid 30  4p6.3515 11.155 0. origin y=-17.464 z=0.15 rotate a2=-1.35
'*** four is item 2 in drawing 84-10649 ornl/csd/tm-220 ***
  cylinder 40 4.555 12.918 0.        origin x=-12.176 y=-9.343 z=0.111 rotate a1=-52.5 a2=-1.400
'*** five is item 3 in drawing 84-10649 ornl/csd/tm-220 ***
  cylinder 50 5.761 13.475 0.        origin x=-16.333 y=1.681 z=0.174 rotate a1=83.5 a2=+1.173
'*** six is item 4 in drawing 84-10649 ornl/csd/tm-220 ***
  cylinder 60 4.5525 12.969 0.    origin x=-9.539 y=11.168 z=0.156 rotate a1=40.5 a2=+1.970
```

(continues on next page)

(continued from previous page)

```
'*** seven and eight are item 5 in drawing 84-10649 ornl/csd/tm-220 ***
'seven
  cuboid  70  2p3.81 8.13 -4.573 8.91 0. origin y=15.698 z=0.290 rotate a2=+2.58
'eight
  cylinder 80 4.573 13.229 8.91   origin y=15.698 z=0.290 rotate a2=+2.58
'*** nine is item 6 in drawing 84-10649 ornl/csd/tm-220 ***
  cylinder 90 4.5545 12.974 0.    origin x=9.854 y=10.964 z=0.134 rotate a1=-42.0 a2=+1.680
'*** ten is item 7 in drawing 84-10649 ornl/csd/tm-220 ***
  cylinder 100 5.7495 13.475 0.    origin x=16.388 y=1.434 z=0.140 rotate a1=-86.0 a2=+1.400
'*** eleven is item 8 in drawing 84-10649 ornl/csd/tm-220 ***
  cylinder 110 4.5565 12.954 0.    origin x=12.029 y=-9.398 z=0.087 rotate a1=38.0 a2=-1.100
'*12 through 14 is the centerpiece in drawing 84-10649 ornl/csd/tm-220
'twelve
  cylinder 120 5.757 2.690 0.      origin x=-0.593 y=-0.593 z=-1.753
'thirteen
  cuboid 130 4p6.35 5.718 0.       origin z=0.937
'fourteen
  sphere    140 6.082 chord +z=0.   origin x=-0.268 y=0.268 z=6.655
'*** fifteen is the system boundary ***
'fifteen
  cuboid   150 4p25.0 15.0 -2.0
  media  1 1 +10       vol=20.58546556
  media  2 1 +20 -10   vol=245.678420867
  media  3 1 +30 -20   vol=1800.040061395
  media  4 1 +40       vol=842.019046637
  media  5 1 +50       vol=1404.99376489
  media  6 1 +60       vol=844.415646269
  media  7 1 +70       vol=862.4600226
  media  8 1 +80 -70   vol=283.749744681
  media  9 1 +90       vol=845.483582679
  media 10 1 +100      vol=1399.390119093
  media 11 1 +110      vol=844.921798001
  media 12 1 +120 -130 vol=280.088070346
  media 13 1 +130      vol=922.25622
  media 14 1 +140 -130 vol=471.191948666
  media 0 1 150 -10 -20 -30 -40 -50 -60 -70 -80 -90 -100
            -110 -120 -130 -140  vol=31432.726088316
  boundary 150
end geom
read plot
  scr=yes  lpi=10
           clr= 1 255   0   0
                2   0   0 205
                3   0 229 238
                4   0 238   0
                5 205 205   0
                6 255 121 121
                7 145  44 238
                8 150 150 150
                9 240 200 220
               10   0 191 255
               11 224 255 255
               12   0 128  64
               13 255 202 149
               14 255   0 128
             end color
  ttl='grotesque x-y slice at z=0.5'
  xul=-25.5 yul= 25.5 zul=0.5
  xlr= 25.5 ylr=-25.5 zlr=.5
  uax=1     vdn=-1  nax=800 end
  ttl='grotesque x-y slice at z=2.0'
  xul=-25.5 yul= 25.5 zul=2
  xlr= 25.5 ylr=-25.5 zlr=2 end
  ttl='grotesque x-y slice at z=9.5'
  xul=-25.5 yul= 25.5 zul=9.5
```

(continues on next page)

(continued from previous page)

```
  xlr= 25.5 ylr=-25.5 zlr=9.5 end
  ttl='grotesque y-z slice at x=-0.593'
  xul=-.593 yul=-25.5 zul=15.5
  xlr=-.593 ylr= 25.5 zlr=-3.5
  uax=0     vax=1
  vdn=0     wdn=-1 nax=800  end
  ttl='grotesque x-z slice at y=0.0'
  xul=-25.5 yul=0.0  zul=15.5
  xlr= 25.5 ylr=0.0  zlr=-3.5
  uax=1     vax=0    wax=0
  udn=0     vdn=0    wdn=-1 nax=800  end
  ttl='grotesque x-z slice at y=12.125'
  xul=-25.5 yul=12.125  zul=15.5
  xlr= 25.5 ylr=12.125  zlr=-3.5
  uax=1     vax=0       wax=0
  udn=0     vdn=0       wdn=-1 nax=800  end
  ttl='grotesque x-z slice at y=-12.000'
  xul=-25.5 yul=-12.000  zul=15.5
  xlr= 25.5 ylr=-12.000  zlr=-3.5
  uax=1     vax=0        wax=0
  udn=0     vdn=0        wdn=-1 nax=800  end
end plot
end data
end
```

CSAS6 Sample problem 8 Infinite Array of MOX and UO2 Assemblies

The purpose of this problem is to calculate the k_{eff} of a system composed of an infinite array of MOX assemblies interspersed between UO_2 assemblies. Both assembly types contain 331 pins in a hexagonal lattice with a pin pitch of 1.275 cm and an assembly pitch of 23.60 cm as shown in Fig. 2.1.14. The moderator is borated water at 306°C having a density of 0.71533 gm/cc and composed of 99.94 wt % H_2O and 0.06 wt % natural boron. Each fuel rod is 355 cm in length, has a radius of 0.3860 cm, 0.722-cm-thick Zr cladding with no gap, and is at a temperature of 754°C.

The UO_2 fuel consists of 4.4 wt % ^{235}U and 95.6 wt % ^{238}U at a density of 8.7922 gm/cc. The UO_2 fuel also contains 9.4581E-9 atoms/b-cm of ^{135}Xe and 7.3667E-8 atoms/b-cm of ^{149}Sm.

The MOX fuel consists of 96.38 wt % UO_2 and 3.62 wt % PuO_2 at a density of 8.8182 gm/cc. The UO_2 fuel is composed of 2.0 wt % ^{235}U and 98.0 wt % ^{238}U. The PuO_2 fuel is composed of 93.0 wt % ^{239}Pu, 6.0 wt % ^{240}Pu- and 1.0 wt % ^{241}Pu. The MOX fuel also contains 9.4581E-9 atoms/b-cm of ^{135}Xe and 7.3667E-8 atoms/b-cm of ^{149}Sm.

These two assemblies are placed so they represent an infinite array in the X and Y dimensions as shown in Fig. 2.1.15. There is 20 cm of water above and below fuel assemblies. This problem uses CENTRM/PMC as the resolved resonance processor cross section. Since an infinite array cannot be explicitly modeled, a section of the array is modeled and the X and Y sides have mirror reflection.

```
=csas6        parm=(centrm)
sample problem 8 - VVER inf. array - MOX & UO2 Assemblies
v7.1-252
read comp
'  UO2 Fuel
    uo2     1 den=8.7922 1.0 1027 92235 4.4  92238 95.6 end
    xe-135  1 0 9.4581E-09 1027 end
    sm-149  1 0 7.3667E-08 1027 end
'  MOX Fuel
    uo2     2 den=8.8182 0.9638 1027  92235  2.0 92238 98.0 end
    puo2    2 den=8.8182 0.0362 1027  94239 93.0 94240  6.0 94241  1.0 end
    xe-135  2 0 9.4581E-09 1027 end
```

(continues on next page)

(continued from previous page)

```
    sm-149  2 0 7.3667E-08 1027 end
'  Cladding for UO2 fuel
    zr      3 den=6.4073 1.0  579  end
'  Moderator for UO2 fuel
    h2o     4 den=0.71533 0.9994 579 end
    boron   4 den=0.71533 0.0006 579 end
'  Cladding for MOX fuel
    zr      5 den=6.4073 1.0  579  end
'  Moderator for MOX fuel
    h2o     6 den=0.71533 0.9994 579 end
    boron   6 den=0.71533 0.0006 579 end
'  Moderator for vacant units
    h2o     7 den=0.71533 0.9994 579 end
    boron   7 den=0.71533 0.0006 579 end
end comp
read celldata
  latticecell triangpitch pitch=1.2750 4 fueld=0.7720 1 cladd=0.9164 3 end
  latticecell triangpitch pitch=1.2750 6 fueld=0.7720 2 cladd=0.9164 5 end
'  more data  dab=500  end more
end celldata
read param
  gen=203  npg=1000
end param
read bounds
  all=mirror zfc=void
end bounds
read geom
  unit   1
   com='UO2 Fuel Rod'
   cylinder 10 0.3860 355.0 0.0
   cylinder 20 0.4582 355.0 0.0
   hexprism 30 0.6375 355.0 0.0
   media 1 1 10
   media 3 1 20 -10
   media 4 1 30 -20
   boundary 30
  unit  2
   com='Vacant(water filled) hex'
   hexprism 10 0.6375 355.0 0.0
   media 7 1 10
   boundary 10
  unit  3
   com='Vacant(water filled) hex'
   hexprism 10 0.6375 355.0 0.0
   media 7 1 10
   boundary 10
  unit   4
   com='MOX Fuel Rod'
   cylinder 10 0.3860 355.0 0.0
   cylinder 20 0.4582 355.0 0.0
   hexprism 30 0.6375 355.0 0.0
   media 2 1 10
   media 5 1 20 -10
   media 6 1 30 -20
   boundary 30
  global unit 5
   rhexprism 10 11.800 355.0 0.0
   rhexprism 20 11.800 355.0 0.0 origin y=23.6
   rhexprism 30 11.800 355.0 0.0 origin x=20.4382 y=11.8
   rhexprism 40 11.800 355.0 0.0 origin x=20.4382 y=35.4
   cuboid   50 20.4382 0.0 35.4 0.0 375.0 -20.0
   array 1  10 -20 -30 -40 place 12 12 1  0.0       0.0  0.0
   array 2  20 -10 -30 -40 place 12 12 1  0.0      23.6  0.0
   array 2  30 -10 -20 -40 place 12 12 1  20.4382  11.8  0.0
   array 1  40 -10 -20 -30 place 12 12 1  20.4382  35.4  0.0
```

(continues on next page)

(continued from previous page)

```
   media 4 1 50 -10 -20 -30 -40
   boundary 50
end geom
read array
  ara=1 typ=shexagonal nux=23 nuy=23 nuz=1
    fill
   3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
    3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3 3 3
   3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3 3
    3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3 3
   3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3
    3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3
   3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3
    3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3
   3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3
    3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3
   3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3
    3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3
   3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3
    3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3
   3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3
    3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3 3
   3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3 3
    3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3 3 3
   3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
    end fill
  ara=2 typ=shexagonal nux=23 nuy=23 nuz=1
    fill
   2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
    2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2
   2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2
    2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2
   2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2
    2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2
   2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2
    2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2
   2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2
    2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2
   2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2
    2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2
   2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2
    2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2
   2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2
    2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2
   2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2
    2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2
   2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2
    2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2
   2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
    end fill
end array
read plot
  lpi=10 scr=yes
  ttl='VVER assembly x-y x-section'
  xul=-0.1  yul=35.5  zul=10
  xlr=20.6  ylr=-0.1  zlr=10
  uax=1     vdn=-1.0
  nax=640   pic=mat   end plt1
end plot
```

(continues on next page)

(continued from previous page)

```
read volume
    type=random  batches=1000
end volume
end data
end
```

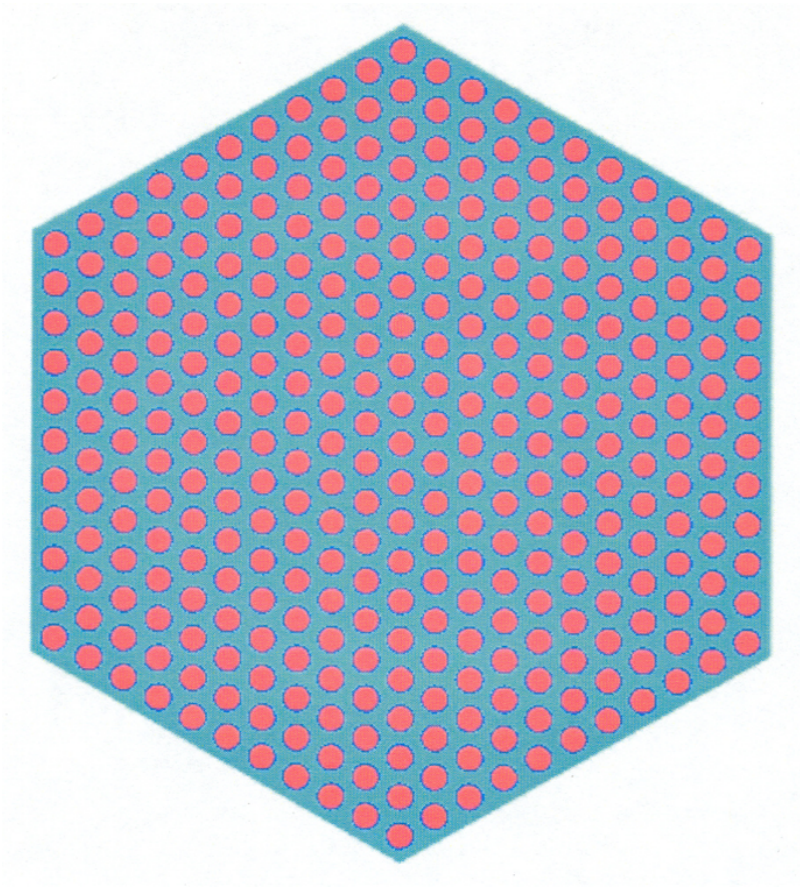


Fig. 2.1.14: MOX or UO_2 hexagonal assembly.

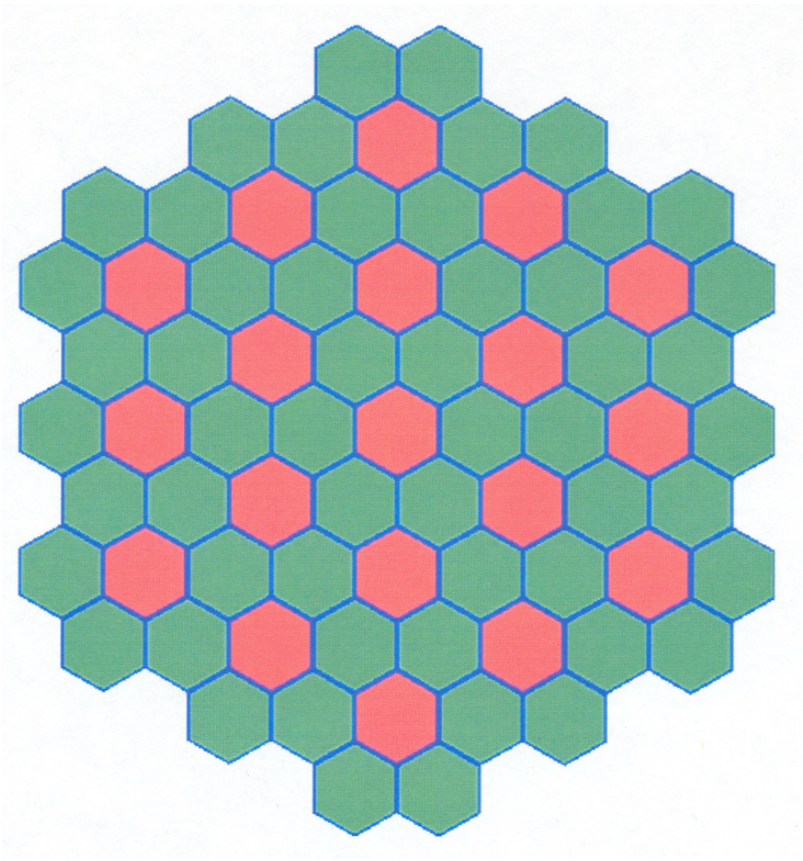


Fig. 2.1.15: Infinite array of MOX assemblies interspersed between UO_2 assemblies.

2.2 CSAS-SHIFT: CRITICALITY SAFETY ANALYSIS SEQUENCE WITH SHIFT

K. B. Bekar, *G. Davidson*, *B. Langley*, *B.J. Marshall*

The CSAS-Shift sequence integrates the Shift advanced Monte Carlo solver into the CSAS framework as an alternative to the KENO transport solvers to perform reliable and efficient eigenvalue calculations for criticality safety and reactor physics analysis. It supports both KENO V.a and KENO-VI geometries and provides most widely used KENO capabilities available in the CSAS5 and CSAS6 sequences for both multigroup and continuous-energy transport modes. The highly scalable Shift Monte Carlo solver enables faster solutions when running on multiple cores, and it shows better performance to achieve the same level of accuracy compared to the CSAS sequences with the KENO codes.

2.2.1 INTRODUCTION

The CSAS-Shift sequence was designed to provide the modeling and simulation capabilities required for criticality safety and reactor physics analysis through the Shift Monte Carlo solver.

Shift is a massively parallel Monte Carlo radiation transport package in the Exnihilo radiation transport code suite [CSAS-ShiftPJE+16], [CSAS-ShiftESSC10]. Shift was developed including features such as support for both fixed-source and eigenvalue Monte Carlo transport capabilities with multiple geometry and physics engines, hybrid capabilities for variance reduction methods, and advanced parallel decompositions to scale

well from laptops to small computing clusters to advanced supercomputers. Shift supports different geometry engines, including the Oak Ridge Adaptable Nested Geometry Engine (ORANGE) designed to provide particle transport capabilities on both KENO V.a and KENO-VI geometries as well as geometry visualization capabilities in the Fulcrum user interface. Shift with both versions of KENO geometries performs eigenvalue calculations in both continuous-energy and multigroup modes. Shift supports most widely used primary capabilities available with the KENO codes and provides some unique capabilities with ORANGE such as modeling randomly packed media and efficient parallel calculations for volume estimates.

CSAS-Shift provides all capabilities for both multigroup and continuous-energy transport modes. Like the CSAS5 and CSAS6 implementation, in the multigroup calculation mode, CSAS-Shift sequences automate the processing of the cross sections for temperature corrections and problem-dependent resonance self-shielding for utilization in multigroup neutron transport calculations using SCALE's cross section processing module, XSProc. If continuous-energy calculation mode is selected, no resonance processing is needed, and the continuous-energy cross sections are used directly in the Shift code, with temperature corrections provided as the cross sections are loaded.

CSAS-Shift with the highly scalable Shift solver enables some unique capabilities and faster solutions when running on multiple cores, and it shows better performance to achieve the same level of accuracy compared to the CSAS sequences with the KENO codes. CSAS-Shift input requirements, supported and unsupported capabilities, and input and output details are described in the following sections.

2.2.2 CSAS-SHIFT INPUT REQUIREMENTS

CSAS-Shift's design is aimed to make a smooth transition between KENO codes to the Shift transport code. Therefore, the original input data layout available in CSAS sequence with the KENO transport codes was kept the same for the CSAS sequence with Shift transport. CSAS-Shift uses the same CSAS5 and CSAS6 inputs, the only input modification that should be required is changing the sequence name by appending `-shift` to the sequence name, as shown in Example 2.2.1.

Like CSAS sequences with KENO codes, CSAS-Shift sequences are named with the KENO geometry that they support: CSAS5-Shift for the models with KENO V.a geometry, and CSAS6-Shift for the models with KENO-VI geometry.

Example 2.2.1: CSAS sequence inputs with KENO and Shift transport

```
=csas5
Godiva sphere
ce_v7.1

read composition
   u-234       1 0 0.000491995 300   end
   u-235       1 0 0.0449996 300   end
   u-238       1 0 0.002498 300   end
end composition

read parameter
   html=no
end parameter

read geometry
   sphere 1 1   8.741
end geometry

end data
end
```

(continues on next page)

(continued from previous page)

```
=csas5-shift
Godiva sphere
ce_v7.1

read composition
   u-234       1 0 0.000491995 300   end
   u-235       1 0 0.0449996 300   end
   u-238       1 0 0.002498 300   end
end composition

read parameter
   html=no
end parameter

read geometry
   sphere 1 1   8.741
end geometry

end data
end
```

Warning: CSAS-Shift sequence implementation may ask the user for some minor input updates for a successful calculation.

2.2.3 SEQUENCE CAPABILITIES

In the CSAS-Shift sequence framework, SCALE data handling is automated as much as possible. Similar to many other SCALE sequences, CSAS-Shift also applies a standardized procedure to provide appropriate number densities and cross sections for the calculation. XSProc is responsible for reading the standard composition data and other engineering-type specifications—including volume fraction or percent theoretical density, temperature, and isotopic distribution, as well as the unit cell data. XSProc then generates number densities and related information, prepares geometry data for resonance self-shielding and flux-weighting cell calculations, and (if needed) provides problem-dependent multigroup cross section processing.

Sequences that execute Shift transport include a data processor named ExnihiloInputBuilder to read and check the KENO data. This data processor processes the KENO data and creates a ParameterList input used by Shift to construct the problem and perform transport calculations. When the data checking has been completed, the CSAS-Shift sequence executes XSProc to prepare a resonance-corrected macroscopic cross section library in the AMPX working library format for the subsequent Shift transport calculation if a multigroup library has been selected.

Similar to CSAS sequences with KENO transport, the CSAS-Shift sequence supports both CELLMIX and Double-het capabilities. For each unit cell specified as being cell-weighted, XSProc performs the necessary calculations and produces a cell-weighted macroscopic cross-section library. Shift may be executed to calculate the k_{eff} or neutron multiplication factor using the cross section library that was prepared by the control sequence.

Computational capabilities available in CSAS sequences with KENO codes—including the determination of k-effective, flux densities, fission densities, mesh tallies, Shannon entropy tally, problem-dependent continuous-energy temperature treatments, parallel calculations, and many more—are also provided by the CSAS-Shift sequence.

CSAS-Shift also supports two new CSAS sequence data blocks, definitions and tallies data, to allow flexible definition and output control of mesh tallies. The mesh responses neutron flux, fission rate, and fission

source can now be requested multiple times on different spatial and energy grids in the same calculation. This capability helps users efficiently manage computational resources when collecting detailed information, depending on their requirements.

Criticality safety tools in SCALE attain some unique capabilities provided by Shift with the new geometry engine ORANGE, such as parallel volume estimates for KENO-VI geometric regions and modeling randomly packed media, which is enabled by implementing a random-packing algorithm to place spherical particles within simple bounding geometries. This capability allows constructing tristructural isotropic (TRISO) particle models for advanced reactor modeling and simulation activities. See Sect. 2.2.4.1.3 for further details.

Details for some of these capabilities, their input methods, and output edits are provided in the following sections.

2.2.3.1 Multigroup limitations

Some of the limitations of the CSAS-Shift multigroup sequences are a result of using preprocessed multigroup cross sections. Inherent limitations in multigroup CSAS-Shift calculations are as follows:

1. Spatial effects such as fuel rods in assemblies where some positions are filled with control rod guide tubes, burnable poison rods, and/or fuel rods of different enrichments. The cross sections are processed as if the rods are in an infinite lattice of identical rods. If the user inputs a Dancoff factor for the cell (such as one computed by MCDancoff), XSProc can produce an infinite lattice cell which reproduces that Dancoff. This can mitigate some spatial lattice effects.

2.2.3.2 Continuous-energy limitations

The continuous-energy cross sections are directly used in Shift. An existing multigroup input file can easily be converted to a continuous-energy input file by simply specifying the continuous-energy library. In this case, all cell data is ignored. However, the following limitations exist:

1. If CELLMIX is defined in the cell data, the problem will not run in continuous-energy mode. CELLMIX implies new mixture cross sections are generated using XSDRNPM-calculated cell fluxes; therefore it is not applicable in continuous-energy mode.
2. Problems with DOUBLEHET cell data are not allowed, as they inherently utilize the CELLMIX feature.

2.2.3.3 Unsupported Capabilities

Although Shift Monte Carlo code was designed with several advanced capabilities, it does not currently support some of the unique features available in KENO codes. Therefore, CSAS-Shift does not provide some of the capabilities available in CSAS5 and CSAS6 sequences.

The missing capabilities are mostly considered as the outdated features or those seldomly used by CSAS users in their analysis. The equivalent capabilities will be activated in the Shift transport code in the next SCALE release, depending on the need basis.

Table 2.2.1 summarizes the capabilities currently supported by CSAS with KENO codes but not supported by CSAS-Shift sequences.

Table 2.2.1: Unsupported CSAS capabilities

Capability	Input method(s) to activate the capability	Comments
Adjoint transport	parameter data `ADJ`	Adjoint transport capability is not available in Shift
Prompt-only *nu*	parameter `PNU`	Shift does not support using prompt neutron spectrum only in continuous-energy mode
Use unionized mixture cross section	parameter `UUM`	Shift does not support KENO-like mechanism to store cross sections on a material-based unionized energy grid for a faster cross section lookup in continuous-energy mode Although this method benefits for faster runtimes for some KENO models, storing all data may require prohibitively large amount of memory for problem with a large number of materials. Different approaches are being developed in Shift transport, and some experimental implementation is available in CSAS-Shift. See Table 2.2.4 for more details.
Matrix k_{eff}	parameters `MKP`, `CKP`, `FMP MKU`, `CKU`, `FMU MKH`, `CKH`, `FMH MKA`, `CKA`, `FMA HAL`, `HHL`, `FMA`	An alternative k-eff calculation method available in KENO codes are not supported by Shift.
Start data types 2, 3, 4, 5 and 9	`start data` block `NST`= 2, 3, 4, 5, and 9	Start data types 2 - 5 have not been implemented because GLOBAL ARRAY of KENO V.a geometry is not supported by ORANGE geometry engine used by Shift transport. Start type 9 designed to read starting distribution from a mesh source file is not currently supported by CSAS-Shift.
Biasing or weighting data	`bias data` block	KENO-like biasing capability is not currently supported by Shift transport.
Periodic and White Albedo boundary conditions	`bounds data` block	Shift transport does not currently support PERIODIC and WHITE boundary conditions for both KENO V.a and KENO-VI geometries.

continues on next page

Table 2.2.1 – continued from previous page

Capability	Input method(s) to activate the capability	Comments
Differential Albedos	`bounds data` block, `PAX` in `parameter data` block	Material specific albedos available with KENO multigroup transport is not supported by Shift multigroup transport
GLOBAL ARRAY instead of GLOBAL UNIT in KENO V.a geometry	`geometry data` block	Shift with ORANGE geometry engine does not currently support GLOBAL ARRAYs instead of the GLOBAL UNIT in KENO V.a geometry.
LOOP construct in array data	`array data` block	LOOP construct in array data input block is not supported by ORANGE geometry engine as part of Shift code.
Volume calculation (random sampling)	`type=RANDOM` in `volume data` block	Random volume estimates for KENO-VI geometry is not available in ORANGE geometry engine used by Shift transport
Accumulate mesh fluxes	parameter `MFX`	Shift does not support to tally mesh fluxes which are averaged over the region volumes in each mesh voxel.
Compute and print mean free paths	parameter `MFP`	This capability is not currently implemented in Shift.
Region-dependent fissions and absorptions	parameters `FAR` and `GAS`	Although these tallies are available in Shift transport they are not currently implemented in CSAS-Shift.
Mixture-dependent reaction tallies	`reaction data` block	Although these tallies are available in Shift transport they are not currently implemented in CSAS-Shift.
Time controlled termination	parameter `TME`	Shift does not have job termination capability controlled by the user-defined time limit.
Terminate execution on user signal	by creating a file named `stop_keno` in the working directory	CSAS-Shift does not support this capability
Restart capability	parameters `RES`, `BEG`, `APP`, `RST`, `WRS`	Restart capability is not available in Shift

continues on next page

Table 2.2.1 – continued from previous page

Capability	Input method(s) to activate the capability	Comments
Print particle tracks	parameter TRK	Although Shift has its own mechanisms to print information about each particle history, this capability is not fully integrated in CSAS-Shift sequence. It will be available in next releases.
Problem Characterization Output Edit	always ON	This capability is not currently implemented in CSAS-Shift.
Frequency Distributions	always ON	This capability is not currently implemented in CSAS-Shift
Plots of avg. k-eff and Shannon entropy	always ON	CSAS-Shift does not plot the average k-eff by generations run, the k-eff by generations skipped, and Shannon entropy per generation. Instead of the char-plots, CSAS-Shift creates Ptolemy plots and stores them in a dedicated plot directory. Fulcrum may be used to visualize these plots. See Sect. 2.2.6.1.12
Summary of Source Convergence Diagnostics	parameter SCD	CSAS-Shift does not perform the posterior entropy tests available in KENO. Instead, result of a single test performed by Shift is captured and printed in the relevant output section.
Print capabilities for mixed cross section	parameters AMX, XS1, XS2, PKI P1D, XAP, XSL	Currently, no capability is available to print the cross sections used by Shift transport.
Flux moments and angular flux calc.	parameters TFM, PMF, PMM	CSAS-Shift does not support any of these capabilities.
Print starting points	PSP parameter in start data	This capability is not currently implemented in CSAS-Shift
Plot capability	parameter PLT plot data block	Old-style plotting capabilities available in KENO codes are not supported. Fulcrum can be used for geometry visualization.
HTML output	parameter HTM	Old-style HTML-based output method is not supported.

2.2.4 INPUT DATA GUIDE

This section describes the input data required for the CSAS-Shift sequence. A typical CSAS6-Shift input, shown in Example 2.2.2, starts with the sequence identifier always preceded by the = sign, and it is followed by the problem title. Then, a cross section library name is specified, and all these entries are followed by several data blocks each starting with *READ data_block* and ending with *END data_block*.

Example 2.2.2: A typical CSAS-Shift sequence input

```
=CSAS6-Shift parm=(parm_options)
problem title

' ----- XSProc data
' cross section library name (REQUIRED)
ce_v7.1

' List of material specifications in standard SCALE format (REQUIRED)
read composition
   ...
end composition

' Specify data for resonance processing (OPTIONAL)
read celldata
   ...
end celldata

' ---- New CSAS sequence data blocks

' Used to define energy bounds and grid geometries for
' the tallies defined in tallies data block
' (REQUIRED if tallies data block exists)
read definitions
   ...
end definitions

' Used to define tallies in a more robust way (OPTIONAL)
read tallies
   ...
end tallies

' ---- KENO transport data
' Specify the problem geometry (REQUIRED)
read geometry
   ...
end geometry

' Other input data blocks (OPTIONAL)
```

Because CSAS-Shift uses the same input data used by CSAS5 and CSAS6, details of the input data blocks, compositions, celldata, definitions, and tallies will not be repeated here, and they can be seen in Sect. 2.1.4. Data blocks in the KENO transport data category will be discussed in the following section.

2.2.4.1 KENO input data in CSAS-Shift

Table 2.2.2 presents the lists of the KENO input data blocks supported by CSAS-Shift sequence. The input method in some data blocks may show some minor differences between the CSAS-KENO and CSAS-Shift sequences. Similarly, some capabilities provided by each input block also have some differences. All these details are discussed in this section. KENO input data blocks, that are `reaction data`, `bias data`, `mixt data`, and `plot data`, are not currently supported by the CSAS-Shift sequences.

Table 2.2.2: Summary of KENO input data blocks available in CSAS-Shift sequences.

Data block	**Status**	**Comments**
Parameters*	Supported	See Sect. 2.2.4.1.1 for more details.
Geometry	Supported	See Sect. 2.2.4.1.2 for more details.
Array data	Supported	
Boundary conditions	Supported	See Sect. 2.2.4.1.2 for more details.
Volume data	Supported	See Sect. 2.2.4.1.2 for more details.
Energy boundaries	Supported	
Start data	Supported	See Sect. 2.2.4.1.4 for more details.
Grid geometry data	Supported	
Plot data	Not available	The data in this block will be ignored in the calculations
Bias data	Not available	Execution will be terminated.
Reaction data	Not available	The data in this block will be ignored in the calculations
Mixt data	Not available	The data in this block will be ignored in the calculations
* Must precede all other data blocks in this table.		

Similar to CSAS5 and CSAS6, `geometry data` is the only KENO data block required to perform Shift transport calculation as part of the CSAS-Shift sequence. Other data blocks are optional, and the same default values listed in various locations in Sect. 8.1 are also applied to the data in each data block in CSAS-Shift. Note that parameter data must precede all other KENO data blocks if it is entered.

When CSAS-Shift is run with a user input including `bias data` block, the execution will be terminated with the error message given in Example 2.2.3. Note that CSAS-Shift ignores the data entered in the unsupported `plot data`, `mixt data`, and `reaction data` blocks and continues the calculation. User is notified with a warning message as shown in Example 2.2.4.

Example 2.2.3: Error message printed by CSAS-Shift output when `biasing data` is found in user input.

```
***Error: Failed to run ExnihiloModule with assertion:
-----------------------------------------------
 These input cards are unsupported by Shift.
-----------------------------------------------
 They must be removed from the input to run.
-----------------------------------------------
 line: 19   column: 1   biasing
-----------------------------------------------


 ^^^ at /ornldev/code/Scale/S63/Source/packages/Module/Exnihilo/InputProcessorBase.cpp:193
```

Example 2.2.4: A typical warning message printed by CSAS-Shift output when an unsupported data block is found in user input.

```
===========================================================================================
Input Warnings:
===========================================================================================

***Warning: Plot block found. This is not currently supported by Shift and will be ignored for now.
```

Note: CSAS5-Shift and CSAS6-Shift also support PARM=CHECK or PARM=CHK sequence parm options. This will allow checking the input data without performing cross section calculation as well as Shift transport calculations.

=CSAS6-SHIFT PARM=CHK

Parameter data

The KENO parameter data block in both CSAS5 and CSAS6 sequences provides many control parameters to activate the capabilities available in KENO transport for the problem being run. CSAS-Shift supports only a subset of these parameters, as listed in Table 2.2.3. Detailed description of these parameter can be seen in Sect. 8.1.3.3.

Parameters entered in the `parameter data` input block are processed by CSAS-Shift sequence implementation, and then the ParameterList input is updated to accordingly activate/deactivate the equivalent capabilities with Shift transport if the asking feature is currently supported by Shift. CSAS-Shift usually ignores the unsupported parameters by notifying the user with a warning message, and then it continues the calculation. For some specific parameters, code can terminate the execution and ask the user to remove this parameter from the input and rerun the code for a successful calculation.

Caution: CSAS-Shift notifies users of the unsupported parameters with a warning message before **Numeric and Logical Parameters** edit in the output, and then it ignores this parameter. It is the user's responsibility to examine which input parameter is ignored in the current calculation.

Note: CSAS-Shift defaults the value of a parameter, which is currently supported but not defined in the `parameter data` input block, to the KENO default. In other words, both CSAS and CSAS-Shift use the same defaults for the same parameters.

Table 2.2.3: Summary of KENO parameters currently supported by CSAS-Shift

PARAMETERS:		Format: READ PARAM *parameter_data* END PARAM See Sect. 8.1.3.3 for details.			
KEY	**DE-FAULT**	**DEFINITION**	**KEY**	**DE-FAULT**	**DEFINITION**
RND=	given	random number	THC=	10.0	thermal energy cutoff (eV)
SIG=	0.0	deviation limit	DBL=	0.4	DBRC lower energy cutoff (eV)
WTA=	0.5	average weight	DBH=	210	DBRC upper energy cutoff (eV)
WTL=	1/WTH	Russian Roulette weight	MSH=	0.0	mesh size of the cubic mesh
GEN=	203	number of generations	DBR=	0	use DBRC for scattering
NPG=	1000	number per generation	DBX=	2	Doppler Broadening method
NSK=	3	generations skipped	CET=	0	CE TSUNAMI calculation mode
NGP=	252	number of energy groups for tallying	CFP=	-1	number of latent generations for CE- SUNAMI
PTB=	YES	use probability tables	SCD=	YES	fission source convergence diag.
FLX=	NO	collect and print region fluxes	CDS=	NO	accumulate neutron production
FDN=	YES	fission densities	FIS=	NO	fission rate mesh tally
NUB=	YES	neutrons per fission	GFX=	NO	compute grid fluxes
FST=	NO	print F*(r) 3dmap	CGD=	NO	use mesh for CLUTCH F*(r) calc.
RUN=	YES	execute problem	TRK=	NO	NOT FULLY IMPLEMENTED

Table 2.2.4: Summary of parameters only available in CSAS-Shift

PARAMETERS:		
KEY	**DEFAULT | DEFINITION**	
PN_ORDER=	5	Legendre polynomial order
DOUBLE_INDEXING=	0.0	Accelerate xsec calculation using double
THINNING_TOLERANCE=	0.0001	Tolerance to use thinning the unionized x

In CSAS5 and CSAS6, users can control the number of scattering angles in multigroup calculations by entering the SCT parameter in the KENO `mixing data` block. The similar capability in CSAS-Shift was provided by adding a new parameter, PN_ORDER=, to the `parameter data` block because the `mixing data` block is not supported by CSAS-Shift sequences. Its default is set as 5.

Geometry data

CSAS-Shift with ORANGE geometry supports all KENO V.a and KENO-VI geometry capabilities except the following:

- Using `GLOBAL ARRAY` instead of `GLOBAL UNIT`. Further details for this capability are available in KENO V.a, see Sect. 8.1.3.4 and Sect. 8.1.4.6
- `LOOP` construct in `array data` block
- Material-specific albedo boundary conditions
- PERIODIC and WHITE albedo boundary conditions
- Random volume estimates for KENO-VI geometry (`TYPE=random` option in `volume data`)

CSAS-Shift can perform volume calculations with the stochastic ray-tracing method concurrently on the replicated domain on multiple cores.

Random geometry

Another new capability CSAS-Shift provides is the modeling of random packed media. The primary features that require input specification for this capability are the regular array and stochastic placement of spherical geometry without clipping along the region boundary, where the available region boundaries are spheres, cuboids, and cylinders.

To implement these new features, a new data block named `randomgeom` was added to SCALE. The `randomgeom` block is composed of `randommix` specifications, which are then composed of key/value pairs. The basic structure of the `randomgeom` block and then `randommix` specifications is as follows:

Example 2.2.5: New randomgeom data block in CSAS-Shift

```
read randomgeom

    RANDOMMIX = ID
       TYPE=random/cubic/fcc
       UNITS= U1 U2 ... UN end
       PFS= pf1 pf2 ... pfN end
       CLIP= yes/no
       SEED= int
    end RANDOMMIX

end randomgeom
```

with

- randommix keyword with ID number or name
- random/cubic/face-centered cubic array options
- list of existing unit number(s) in geometry
- fraction of volume occupied by units `U1 ... UN`
- boundary clipping on or off
- control random seed for random placement

Similar to the array block, this input block requires that the unit geometry specified as part of a randommix in the units list must exist in the geometry and must have a spherical outer boundary. The PF input, specified with the PFS keyword, must be the same length as the units list, and the sum of the PFS list must be less than 1.0. In practice, the actual limit to the total PF depends on the size and number of the units specified by the user, so the requirement for PFS to be less than 1.0 is just a simple check. The TYPE keyword specifies the placement methodology; TYPE=random will call the stochastic placement algorithm, whereas TYPE=cubic will place the unit geometry in a simple cubic structure (one sphere in a cuboid), and TYPE=fcc will place the unit geometry in a face-centered cubic (FCC) structure. For all type options, the CLIP keyword will control the clipping of geometry along the boundary. The SEED keyword is an integer used to determine the random seed used for placement of the stochastic geometry and for choosing which particle to place in the BCC and FCC lattice structure.

A "fill" material—the interstitial media surrounding the random spherical geometry units—is not present in the randomgeom block. The fill media is handled in the unit specification within the KENO geometry data block in the input file. As in a normal array in KENO geometry, the randommix specified in the randomgeom block is used to fill a region in a unit. However, unlike in the array specification, when using a randommix in a unit, the user must also specify a media filling that region. This feature allows the same randommix to be used in multiple different units with varying fill materials. The basic unit format along with a randommix on the media record itself is as follows:

```
read randomgeom
   randommix=ID
      ...
   end randommix
end randomgeom

read geometry
   ...

   unit U
      surfaces ...
      media ...
      media F biasID surfaces randommix=ID
      boundary S
   ...

end geometry
```

In the sample case shown in Example 2.2.6, a single pebble is filled with a single TRISO particle type. Fig. 2.2.1 shows how TRISO particles are placed in a single pebble with the randomgeom capability. The individual location of particles is written in the Shift Hierarchical Data Format (HDF5) output file, so these locations can be used for verification or other purposes as needed.

Example 2.2.6: One TRISO type in a pebble

```
=csas6-shift
...

read geometry
   unit 1
      com='kernel1'
      sphere 1 2.50e-02
      sphere 2 3.40e-02
      sphere 3 3.80e-02
      sphere 4 4.15e-02
      sphere 5 4.55e-02
```

(continues on next page)

(continued from previous page)

```
      media 100 1 1
      media 101 1 2 -1
      media 102 1 3 -2
      media 103 1 4 -3
      media 104 1 5 -4
      boundary 5
   global unit 10
      com='pebble'
      sphere 1 2.5
      sphere 2 3.0
      cuboid 3 6p5.0
      media 101 1 1 RANDOMMIX='trisos'
      media 106 1 2 -1
      media 0 1 3 -2
      boundary 3
end geometry
read randomgeom
   randommix = 'trisos'
      type= random
      units= 1 end
      pfs= 0.05 end
      clip= no
      seed= 0
    end randommix
end randomgeom

end data
end
```

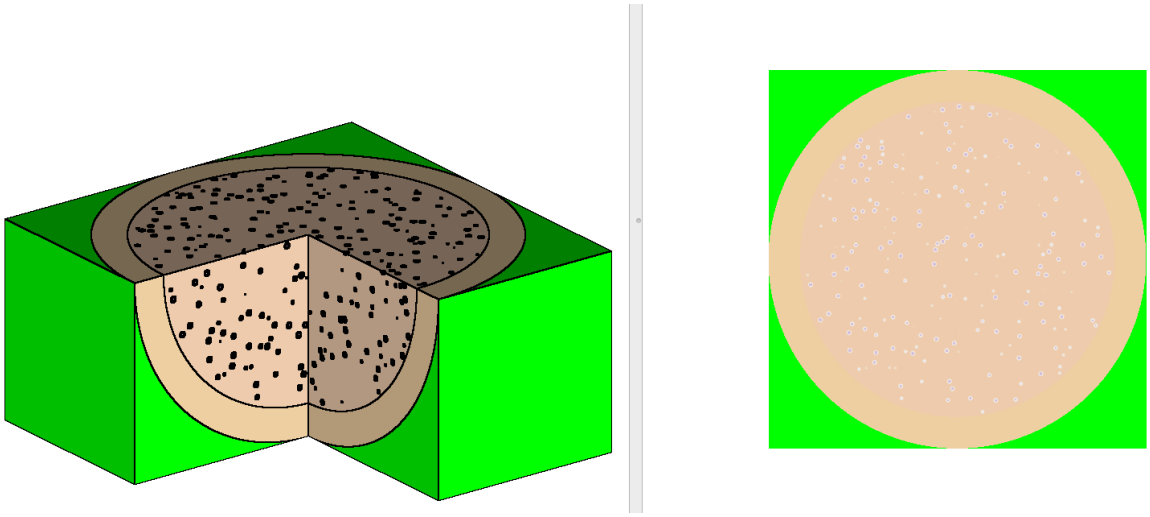


Fig. 2.2.1: Visualization of 3D and 2D cuts of the single TRISO model.

Start data

CSAS-Shift supports only START types 0, 1, 6, 7, and 8. CSAS-Shift start data implementation does not currently support the PSP option, which is used to print source positions sampled by the Shift transport.

Implementations for start types 0, 1, 7, and 8 in CSAS-Shift are are similar to those in CSAS5 and CSAS6. However, there are some minor differences in start type 6.

In KENO start type 6 implementation, the following rules are applied when selecting the starting points (see Sect. 8.1.4.8 and Sect. 8.1.3.3 for more details).

- Start NPG initial fission neutrons at first-NPG starting points defined by start type 6 data if NPG < LNU. Remaining starting points beyond NPG will be discarded.
- Start NPG initial fission neutrons at LNU starting points defined by start type 6 data if NPG = LNU.
- Start LNU initial fission neutrons at the starting points defined by start type 6 data, then randomly select the remaining fission source points (NPG-LNU) from these starting points if NPG > LNU.

where LNU, a start type 6 data parameter, is the total number of starting points specified in the start data block; and NPG, a parameter in the parameter data block, is the number of neutrons per generations.

Unlike KENO, the CSAS-Shift input processor does not follow the above rules when selecting positions for the initial fission neutrons. It calculates the probability of each point being selected and passes all starting points with this information to the Shift module. The Shift module always samples NPG initial fission source points using these data.

For example, the KENO code processes the following input and then samples the initial fission points.

```
=csas6 parm=bonami
Godiva test problem
test-8grp

read composition
   u-234       1 0 0.000491995 300   end
   u-235       1 0 0.0449996 300   end
   u-238       1 0 0.002498 300   end
end composition
read parameter
   htm=no gen=10 npg=15
end parameter

read geometry
   global unit 1
      sphere 1 8.67
      media 1 1 1
      boundary 1
end geometry

read start
   nst=6 ps6=yes psp=yes
   tfx=1.0 tfy=1.0 tfz=1.0 lnu=5
   tfx=2.0 tfy=2.0 tfz=2.0 lnu=20
end start
end data
end
```

The summary of the sampling process is printed in KENO output, as shown in Fig. 2.2.2. KENO first starts 5 neutrons at (1.0, 1.0, 1.0) and the remaining 10 neutrons at (2.0, 2.0, 2.0) since LNU=20 > NPG=15.

```
start type 6 was used.

neutrons started in non-fissile mixtures will use the fission spectrum for mixture     1

neutrons were started at the following positions.

         neutrons    1 to    5 were started at x=    1.00000E+00  y=    1.00000E+00  z=    1.00000E+00
        relative to the global coordinate system.

         neutrons    6 to   15 were started at x=    2.00000E+00  y=    2.00000E+00  z=    2.00000E+00
        relative to the global coordinate system.
```

Fig. 2.2.2: Start type 6 output printed by CSAS6.

However, CSAS-Shift creates a probability distribution from the defined start type 6 points and samples starting positions for NPG=15 particles using this distribution, as shown in Fig. 2.2.3.

```
===============================================================================================
 Starting Source Distribution
===============================================================================================

Starting source type 6 was used for the initial neutron distribution.

Neutrons started in non-fissile mixtures will use the Watt-fission spectrum in start type 6.

***Warning: PSP option is not currently supported in start type 6 on line:24 column:11

Neutrons were started at the following positions relative to the global coordinate system.

- Neutrons were started at ( 1.000000 , 1.000000 , 1.000000 ) with 25% probability.
- Neutrons were started at ( 2.000000 , 2.000000 , 2.000000 ) with 75% probability.
```

Fig. 2.2.3: Start type 6 output printed by CSAS-Shift.

2.2.5 NOTES TO CSAS-SHIFT USERS

In Monte Carlo calculation, the variance of the eigenvalue (k-eff) at each generation is calculated as a sample variance, which is the quantity obtained by assuming no correlation over the generations. However, there is a correlation among the fission sources over generations since the deviation of fission source at a generation from its equilibrium distribution is transferred to the following generations. To resolve this issue, KENO codes use an iterative method to estimate the real variance. [CritSafetyUMN97]

This methodology is not available in Shift transport (i.e., Shift does not estimate the real variances). However, the CSAS-Shift sequence uses this methodology as a post-processing stage to estimate the real variances. Therefore, Shift's reported uncertainty values (which are sample variance) in the message file are different from the CSAS-Shift sequence reported values (real variance) in the output file. It is highly recommended that k-eff uncertainties reported in the output file be used in the analyses where criticality is being assessed.

Caution: k-eff uncertainties printed in message and output files show differences. It is highly recommended that *k-eff uncertainties reported in the output file* be used in the analyses where criticality is being assessed.

2.2.6 CSAS-SHIFT OUTPUT

The CSAS-Shift sequence output is similar to the CSAS5 and CSAS6 outputs, except the output section dedicated to the transport module. See Sect. 2.1.5 for the layout of the output, mixture table edit, and cross section processing summary edits for a typical CSAS sequence.

This section contains a brief description of the output section dedicated to the Shift transport module. This section provides representative samples of the output format. The actual data contained in this section are not necessarily consistent with results computed by the current version of CSAS-Shift.

2.2.6.1 Shift output

The output layout of the Shift transport module is generally similar to the typical KENO output edits printed in the CSAS sequence output. However, some output edits are printed in a very different format, and these are discussed in this section.

Warning and error messages show stylistic differences compared to the traditional CSAS and KENO messages, and their details are not documented in this manual.

Program verification information

Program verification information Fig. 2.2.4 is printed after the header page. It lists the name of the program, the date the load module was created, the library that contains the load module, the computer code name from the configuration control table, and the revision number. The job name, date, and time of execution are also printed. This information may be used for quality assurance purposes.

```
*****************************************************************************************************************
*****************************************************************************************************************
*****************************************************************************************************************
*****                                                                                                       *****
*****                                     program verification information                                  *****
*****                                                                                                       *****
*****                                  code system: SCALE   version: 6.3.0                                  *****
*****                                                                                                       *****
*****************************************************************************************************************
*****************************************************************************************************************
*****                                                                                                       *****
*****                                                                                                       *****
*****                    program:  shift                                                                    *****
*****                                                                                                       *****
*****                    version:  6.3                                                                      *****
*****                                                                                                       *****
*****                   username:  5kq                                                                      *****
*****                                                                                                       *****
*****                   hostname:  kbb-mac                                                                  *****
*****                                                                                                       *****
*****                                                                                                       *****
*****          date of execution:  2021-10-17                                                               *****
*****                                                                                                       *****
*****          time of execution:  19:43:27                                                                 *****
*****                                                                                                       *****
*****************************************************************************************************************
*****************************************************************************************************************
*****************************************************************************************************************
```

Fig. 2.2.4: Sample program verification table.

General problem information

A general problem information output edit, shown in Fig. 2.2.5, follows the program verification information table. This table is printed by all SCALE Shift sequence implementations. After printing the title given in the input, it summarizes some high-level information for the physics setup of the Shift code.

```
Loaded MG Library: /ornldev/code/Scale/S63/INSTALL/G_63_b1129/data/scale.rev05.xn252v7.1

=============================================================================================================================
 General Problem Information
=============================================================================================================================

 - Problem-Title       Short description title for the input problem       sample problem 18   1f27  critical experiment
 - Problem-Mode        Forward / Hybrid / KCode                            KCode
 - Transport-Mode      Continuous-Energy / Multi-Group                     Multi-Group
 - Particle-Type       Neutron / Photon / Neutron-Photon transport         Neutron
 - Std-Comp-Path       Path to the SCALE Standard Composition Library      /ornldev/code/Scale/S63/INSTALL/G_63_b1129/data/scale.rev40.sclib
 - MC-Lib-Path         Library path used for Monte Carlo transport         /ornldev/code/Scale/S63/INSTALL/G_63_b1129/data/scale.rev05.xn252v7.1
 - MC-Num-Groups       Number of energy groups (N+P) for Monte Carlo       252
```

Fig. 2.2.5: Sample general problem information table.

Input Warnings

CSAS-Shift captures the warning messages emitted from the ExnihiloInputBuilder when processing KENO data for the Shift transport. All these stacked warning messages are printed in the input warnings output table as shown in Fig. 2.2.6.

```
=============================================================================================================================
 Input Warnings:
=============================================================================================================================

***Warning: Plot block found. This is not currently supported by Shift and will be ignored for now.

***Warning: input parameter ( amx = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( cka = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( cku = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( far = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( fma = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( fmh = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( fmu = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( htm = no   ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( mka = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( mku = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( pax = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( pgm = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( plt = yes  ) will not be used because it is not currently supported by Shift.

***Warning: input parameter ( pwt = yes  ) will not be used because it is not currently supported by Shift.

 - Number of Warning Messages: 15
```

Fig. 2.2.6: Sample input warnings table.

Tables of parameter data

The CSAS-Shift parameter edits list both numeric and logical parameters in the same table. In each table row, the name of the KENO parameter, its short description, the current value of the parameter, and its input method are printed. If the parameter value has been entered by user in the KENO parameter data block, the input method is printed as *(input *)*. Otherwise, the input method is printed as *(default)*. The user should always verify that the parameter data block was entered as desired. An example of the parameters table is shown in Fig. 2.2.7.

```
================================================================================================================================
 Numeric and Logical Parameters
================================================================================================================================

 - cds                 Collect CADIS fissions                                 OFF              ( default )
 - cet                 CE-TSUNAMI sensitivity method                          NONE             ( default )
 - cfp                 Number of latent gens in CE-TSUNAMI                    5                ( default )
 - cgd                 Use F*(r) Mesh                                         NO               ( default )
 - dbh                 High Energy Cutoff (in eV) for DBRC                    210              ( default )
 - dbl                 Low Energy Cutoff (in eV) for DBRC                     0.4              ( default )
 - dbr                 DBRC isotope selection                                 OFF              ( default )
 - dbx                 Doppler Broadening Method                              1D-AND-2D        ( default )
 - double_indexing     Accelerate xsec calculation using double indexing      OFF              ( default )
 - fdn                 Compute fission densities                              ON               ( input * )
 - fis                 Fission rate mesh tally                                OFF              ( default )
 - flx                 Print fluxes                                           ON               ( input * )
 - fst                 Print F*(r) mesh values                                NO               ( default )
 - gen                 Number of generations                                  103              ( input * )
 - gfx                 Grid flux, averaged over voxel volume                  OFF              ( default )
 - mfx                 Compute mesh fluxes                                    OFF              ( default )
 - msh                 Mesh size for mesh flux tally                          0.0              ( default )
 - ngp                 Number of energy groups for tallying                   252              ( default )
 - npg                 Number per generation                                  1000             ( input * )
 - nsk                 Number of generations to be skipped                    3                ( default )
 - nub                 Print fission densities                                ON               ( input * )
 - pn_order            Legendre polynomial order                              5                ( default )
 - ptb                 Use probability tables                                 ON               ( default )
 - rnd                 Starting random number                                 f12c09ed2195     ( input * )
 - run                 Execute problem after checking data                    ON               ( default )
 - scd                 Mesh based source convergence diagnostics              ON               ( default )
 - sig                 Cut off standard deviation                             0.0              ( default )
 - thc                 CE Thermal energy cutoff (in eV)                       10               ( default )
 - thinning_tolerance  Tolerance to use thinning the unionized xsec grid      0.0001           ( default )
 - trk                 Print tracking information                             OFF              ( default )
 - wta                 Default value of weight average                        0.5              ( default )
 - wtl                 Weight low for russian roulette                        0.333333         ( default )
```

Fig. 2.2.7: Sample table of numeric and logical parameter data.

Energy boundaries data

The CSAS-Shift implementation supports multiple sets of energy boundaries specifications for some of the tallies. This can be done by using the definitions data block as described in the CSAS manual Sect. 2.1.4. However, it prints only the default energy group bounds in the energy boundaries data edit, as illustrated in Fig. 2.2.8. Energy group boundaries used for each mesh tally will be printed in the mesh tallies output edit.

```
================================================================================================================================
 Upper Energy(eV) Boundaries of 252 Group Structure
================================================================================================================================

  Group     E(eV)       Group     E(eV)       Group     E(eV)       Group     E(eV)       Group     E(eV)
     1    2.00000e+07     52    5.00000e+04    103    6.75000e+01    154    6.00000e+00    205    9.75000e-01
     2    1.73300e+07     53    4.50000e+04    104    6.50000e+01    155    5.40000e+00    206    9.50000e-01
     3    1.56800e+07     54    3.00000e+04    105    6.30000e+01    156    5.00000e+00    207    9.25000e-01
     4    1.45500e+07     55    2.00000e+04    106    6.10000e+01    157    4.70000e+00    208    9.00000e-01
     5    1.38400e+07     56    1.70000e+04    107    5.80000e+01    158    4.10000e+00    209    8.50000e-01
     6    1.28400e+07     57    1.30000e+04    108    5.34000e+01    159    3.73000e+00    210    8.00000e-01
     7    1.00000e+07     58    9.50000e+03    109    5.06000e+01    160    3.50000e+00    211    7.50000e-01
     8    8.18700e+06     59    8.03000e+03    110    4.83000e+01    161    3.20000e+00    212    7.00000e-01
     9    6.43400e+06     60    5.70000e+03    111    4.52000e+01    162    3.10000e+00    213    6.50000e-01
    10    4.80000e+06     61    3.90000e+03    112    4.40000e+01    163    3.00000e+00    214    6.25000e-01
    11    4.30400e+06     62    3.74000e+03    113    4.24000e+01    164    2.97000e+00    215    6.00000e-01
    12    3.00000e+06     63    3.00000e+03    114    4.10000e+01    165    2.87000e+00    216    5.50000e-01
    13    2.47900e+06     64    2.50000e+03    115    3.96000e+01    166    2.77000e+00    217    5.00000e-01
    14    2.35400e+06     65    2.25000e+03    116    3.91000e+01    167    2.67000e+00    218    4.50000e-01

    ...
```

Fig. 2.2.8: Sample energy boundaries edit.

Mixing table data edits

CSAS-Shift uses the same format and contents as those described for KENO codes in Sect. 8.1.3.10 for the mixing table data edits. In this table, the mixture number, density, and temperature are first printed, followed by a table of the nuclides which make up the mixture. This table contains the following data: nuclide ID number, nuclide mixture ID number, atom density, weight fraction of nuclide in mixture, ZA number, atomic weight, temperature, and nuclide title. Mixture temperature is the same as the nuclides' temperatures for the multigroup calculations, but it may show some differences in continuous-energy calculations. See Sect. 8.1.3.10 for details.

A sample mixing table data edit is shown in Fig. 2.2.9 for a multigroup calculation.

```
=================================================================================================================================
 Mixing Table
=================================================================================================================================

 mixture =     1          density(g/cc) = 1.55523e+00          temperature(K) =    300.00
    nuclide    nucmix   atom-dens.   wgt. frac.     za      awt         temp          nuclide title
       1001         1   5.77865e-02  6.21826e-02    1001     1.0078      300.00     h_h2o 1 fast: h1 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:50 2018
       1002         1   6.64621e-06  1.42926e-05    1002     2.0141      300.00     d_d2o 11 fast: h2 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:13 2018
       7014         1   2.12317e-03  3.17441e-02    7014    14.0031      300.00     n14 725 endf-7 rel1 rev7 mod0 Sun Apr  8 12:12:08 2018
       7015         1   7.75657e-06  1.24228e-04    7015    15.0001      300.00     n15 728 endf-7 rel1 rev7 mod0 Sun Apr  8 12:12:09 2018
       8016         1   3.73205e-02  6.37360e-01    8016    15.9949      300.00     o16 825 endf-7 rel1 rev7 mod4 Sun Apr  8 12:12:27 2018
       8017         1   1.42163e-05  2.58030e-04    8017    16.9991      300.00     o17 828 endf-7 rel1 rev7 mod0 Sun Apr  8 12:12:27 2018
       8018         1   7.66934e-05  1.47389e-03    8018    17.9992      300.00     Injected O-18 zero cross sections
      92234         1   1.06784e-05  2.66842e-03   92234   234.0410      300.00     u234 9225 endf-7 rel1 rev7 mod2 Sun Apr  8 12:14:12 2018
      92235         1   9.84603e-04  2.47096e-01   92235   235.0439      300.00     u235 9228 endf-7 rel1 rev7 mod7 Sun Apr  8 12:14:13 2018
      92236         1   5.29387e-06  1.33421e-03   92236   236.0456      300.00     u236 9231 endf-7 rel1 rev7 mod1 Sun Apr  8 12:14:14 2018
      92238         1   6.19415e-05  1.57437e-02   92238   238.0508      300.00     u238 9237 endf-7 rel1 rev7 mod5 Sun Apr  8 12:14:18 2018

 mixture =     2          density(g/cc) = 1.18000e+00          temperature(K) =    293.00
    nuclide    nucmix   atom-dens.   wgt. frac.     za      awt         temp          nuclide title
       1001         2   5.67766e-02  8.05238e-02    1001     1.0078      293.00     h_h2o 1 fast: h1 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:50 2018
       1002         2   6.53006e-06  1.85083e-05    1002     2.0141      293.00     d_d2o 11 fast: h2 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:13 2018
       6000         2   3.54895e-02  5.99840e-01    6000    12.0107      293.00     c 600 endf-7 rel1 rev7 mod0 Sun Apr  8 12:10:10 2018
       8016         2   1.41613e-02  3.18752e-01    8016    15.9949      293.00     o16 825 endf-7 rel1 rev7 mod4 Sun Apr  8 12:12:27 2018
       8017         2   5.39440e-06  1.29044e-04    8017    16.9991      293.00     o17 828 endf-7 rel1 rev7 mod0 Sun Apr  8 12:12:27 2018
       8018         2   2.91013e-05  7.37111e-04    8018    17.9992      293.00     Injected O-18 zero cross sections

 mixture =     3          density(g/cc) = 8.99998e-01          temperature(K) =    293.00
    nuclide    nucmix   atom-dens.   wgt. frac.     za      awt         temp          nuclide title
    9001001         3   7.99137e-02  1.48599e-01    1001     1.0078      293.00     h_ch2 37 fast: h1 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:49 2018
       6000         3   3.84201e-02  8.51401e-01    6000    12.0107      293.00     c 600 endf-7 rel1 rev7 mod0 Sun Apr  8 12:10:10 2018
```

Fig. 2.2.9: Sample mixing table edit.

Geometry data edits

CSAS-Shift captures the output edit from ORANGE and prints these data as the overview of the geometry. Its format is completely different from the traditional KENO geometry output format but includes more descriptive sections for each geometry piece, as shown in Fig. 2.2.10.

```
=============================================================================================================================
 Geometry Essentials
=============================================================================================================================


************************************************
"unit 1" (<inp>:14)
************************************************
:Type:          masked unit
:# Cells:       2 (offset = 0)
:# Surfaces:    3 (offset = 0)
:Bounding box: ``{-8.6 -8.6 -8.6 to 8.6 8.6 8.6}``

======= = ==== ============================================================
Cell    Z Name Surface logic
======= = ==== ============================================================
0       B bnds 0 ~ 1 & 2 ~ & ~
               (from ``<inp>:16``)
1       M 1.1  0 ~ 1 & 2 ~ &
               (from ``<inp>:16``)
======= = ==== ============================================================

Cells with reentrant surface tracking: "bnds"

======= ========== ============================================================
Surface Name       Description
======= ========== ============================================================
0                  Cyl z: r=8.6
        reg@16.coz (from ``<inp>:16``)
1                  Plane: z=-8.6
        reg@16.mz  (from ``<inp>:16``)
2                  Plane: z=8.6
        reg@16.pz  (from ``<inp>:16``)
======= ========== ============================================================

==== ==========================================================================
Cell Fill
==== ==========================================================================
bnds ---
1.1  Material 0
==== ==========================================================================

Reflecting boundaries: reg@16.coz
```

Fig. 2.2.10: Sample ORANGE geometry output edit.

Volume information

Volume tables for both KENO V.a and KENO-VI geometries are always printed by CSAS-Shift using the KENO-style volume editing format and cannot be suppressed. KENO V.a and KENO-VI volume tables show some differences, and all these details are described in KENO manual. See Sect. 8.1.5.17 for further details.

A sample volume output edit for KENO-VI geometry printed by CSAS-Shift is shown in Fig. 2.2.11.

```
=================================================================================================================================
 Volumes for those units utilized in this problem
=================================================================================================================================

                    volumes not specified in the input were set to -1.0

                                  geometry
                unit     uses     region     mixture        total region volume (cm**3)

                  1        27        1           1          1.34944e+05 +/- 2.69031e+01
                                     2           0          1.58911e+03 +/- 1.76928e+01
                                     3           2          3.02505e+04 +/- 5.06464e+01
                                     4           0          1.14128e+06 +/- 8.68818e+01

                  2         1        5           0          0.00000e+00

                  3         1        2           3          2.27367e+05 +/- 8.59460e+01
                                     3           3          2.52236e+05 +/- 1.05032e+02
                                     4           3          2.78492e+05 +/- 8.39155e+01
                                     5           3          3.05954e+05 +/- 2.36072e+02
                                     6           3          3.62926e+05 +/- 1.82181e+02


                    mixture       total mixture volume (cm**3)          total mixture mass (gm)

                        0         1.14287e+06 +/- 8.68239e+01
                        1         1.34944e+05 +/- 2.69031e+01            2.09869e+05 +/- 4.18403e+01
                        2         3.02505e+04 +/- 5.06464e+01            3.56955e+04 +/- 5.97627e+01
                        3         1.42697e+06 +/- 1.57770e+02            1.28427e+06 +/- 1.41993e+02
                                  -------------                          -------------
                                  2.73504e+06                            1.52984e+06

 The maximum order of error associated with the calculated volumes is    1.1134 percent.
```

Fig. 2.2.11: Sample volume output edit.

Initial source edits

A summary table is always printed for start types 0, 1, 6, 7, and 8. The table format is the same for both KENO V.a and KENO-VI geometries. Fig. 2.2.12 illustrates typical starting data for start type 0. The parameter used in this example was `NST=0`.

```
=================================================================================================================================
 Starting Source Distribution
=================================================================================================================================

Starting source type 0 was used for the initial neutron distribution.

Neutrons were started with a flat distribution in a cuboid defined by:

         xmin = -70.590000 xmax = 70.590000 ymin = -70.590000 ymax = 70.590000 zmin = -68.610000 zmax = 68.610000
```

Fig. 2.2.12: Example of start data.

K-effectives by generation

At the completion of each generation, CSAS-Shift prints the k_{eff} for that generation and associated information obtained from the Shift transport module. An example of this printout is given in Fig. 2.2.13.

k-effectives by Generation

generation	generation k-effective	average k-effective	avg k-eff deviation	generation entropy
1	1.01336e+00	1.00000e+00	0.00000e+00	5.89961e+00
2	9.70182e-01	1.00000e+00	0.00000e+00	5.89779e+00
3	1.02343e+00	1.02343e+00	0.00000e+00	5.72883e+00
4	9.79642e-01	1.00153e+00	2.18918e-02	5.84274e+00
5	9.96666e-01	9.99911e-01	1.27430e-02	5.74560e+00
6	9.50861e-01	9.87649e-01	1.52170e-02	5.73809e+00
7	1.05027e+00	9.94360e-01	6.44694e-02	5.72844e+00
8	9.83610e-01	9.92210e-01	3.81502e-02	5.63436e+00
9	9.87795e-01	9.91474e-01	2.53610e-02	5.59095e+00
10	9.83392e-01	9.90319e-01	1.97590e-02	5.75625e+00
11	1.07302e+00	1.00066e+00	2.03331e-02	5.79948e+00
12	1.06752e+00	1.00809e+00	2.02116e-02	5.73203e+00
13	9.51821e-01	1.00246e+00	1.59043e-02	5.78473e+00
14	1.02666e+00	1.00466e+00	1.44316e-02	5.73784e+00
15	9.79247e-01	1.00254e+00	1.32584e-02	5.68546e+00
16	9.89899e-01	1.00157e+00	1.21493e-02	5.73697e+00
17	1.00517e+00	1.00183e+00	1.11792e-02	5.63913e+00
18	9.56220e-01	9.98786e-01	1.08530e-02	5.81476e+00
19	9.78524e-01	9.97520e-01	1.01939e-02	5.66365e+00
20	1.01753e+00	9.98697e-01	9.61749e-03	5.71381e+00
21	1.08211e+00	1.00333e+00	1.02847e-02	5.60681e+00
22	9.58422e-01	1.00097e+00	1.00133e-02	5.68453e+00
23	1.01848e+00	1.00184e+00	9.51646e-03	5.69092e+00
24	1.01482e+00	1.00246e+00	9.05146e-03	5.66359e+00
25	1.03469e+00	1.00393e+00	8.74571e-03	5.70811e+00
26	9.78416e-01	1.00282e+00	8.41911e-03	5.68900e+00
27	1.02084e+00	1.00357e+00	8.08287e-03	5.72107e+00
28	1.00972e+00	1.00381e+00	7.74301e-03	5.64845e+00
29	9.99763e-01	1.00366e+00	7.42860e-03	5.75952e+00
30	1.04161e+00	1.00506e+00	7.28510e-03	5.63108e+00
31	1.06577e+00	1.00723e+00	7.36234e-03	5.65349e+00
32	1.01665e+00	1.00756e+00	7.10251e-03	5.60440e+00
33	9.86369e-01	1.00685e+00	6.89210e-03	5.63754e+00
34	9.91391e-01	1.00635e+00	6.67834e-03	5.77750e+00
35	9.00973e-01	1.00306e+00	7.30001e-03	5.68364e+00
36	1.04272e+00	1.00426e+00	7.17616e-03	5.68296e+00
37	1.07232e+00	1.00626e+00	7.25488e-03	5.70924e+00
38	1.03503e+00	1.00708e+00	7.08899e-03	5.73101e+00
39	1.04942e+00	1.00826e+00	6.98901e-03	5.63885e+00
40	9.66479e-01	1.00713e+00	6.89061e-03	5.61902e+00
41	9.97690e-01	1.00688e+00	6.70665e-03	5.68703e+00
42	1.00626e+00	1.00687e+00	6.52779e-03	5.71207e+00
43	9.80604e-01	1.00621e+00	6.39379e-03	5.68552e+00
44	9.80350e-01	1.00558e+00	6.26537e-03	5.66305e+00
45	1.02343e+00	1.00600e+00	6.12615e-03	5.72912e+00
46	1.00280e+00	1.00593e+00	5.97898e-03	5.57660e+00
47	1.05123e+00	1.00696e+00	5.93264e-03	5.78946e+00
48	9.51343e-01	1.00572e+00	5.93253e-03	5.70625e+00
49	1.04804e+00	1.00664e+00	5.87497e-03	5.74601e+00
50	9.61190e-01	1.00568e+00	5.83022e-03	5.66760e+00
51	1.01157e+00	1.00580e+00	5.70620e-03	5.57502e+00
52	1.01977e+00	1.00608e+00	5.59364e-03	5.70075e+00
53	1.01518e+00	1.00627e+00	5.48144e-03	5.65813e+00
54	9.81417e-01	1.00578e+00	5.39365e-03	5.68064e+00
55	9.96837e-01	1.00561e+00	5.28974e-03	5.60448e+00
56	1.02665e+00	1.00600e+00	5.20278e-03	5.66613e+00
57	1.00231e+00	1.00594e+00	5.10415e-03	5.73515e+00
58	1.03411e+00	1.00645e+00	5.03584e-03	5.70948e+00
59	1.00668e+00	1.00645e+00	4.94343e-03	5.77343e+00
60	9.92644e-01	1.00621e+00	4.86061e-03	5.78771e+00
61	1.01020e+00	1.00628e+00	4.77509e-03	5.88953e+00
62	9.79943e-01	1.00583e+00	4.71397e-03	5.85620e+00
63	9.26348e-01	1.00451e+00	4.82531e-03	5.79497e+00
64	1.00356e+00	1.00449e+00	4.74423e-03	5.73009e+00
65	1.02184e+00	1.00477e+00	4.67448e-03	5.85977e+00
66	1.01753e+00	1.00497e+00	4.60307e-03	5.70199e+00
67	9.45218e-01	1.00404e+00	4.62768e-03	5.67147e+00
68	1.02752e+00	1.00440e+00	4.56956e-03	5.61904e+00
69	1.00384e+00	1.00439e+00	4.49872e-03	5.65525e+00
70	9.82875e-01	1.00407e+00	4.44202e-03	5.65685e+00
71	9.44534e-01	1.00320e+00	4.46457e-03	5.76498e+00
72	9.75537e-01	1.00280e+00	4.41719e-03	5.76101e+00
73	9.72098e-01	1.00236e+00	4.37539e-03	5.72013e+00
74	9.93805e-01	1.00224e+00	4.31416e-03	5.80465e+00
75	9.49435e-01	1.00150e+00	4.31751e-03	5.75048e+00
76	9.93615e-01	1.00139e+00	4.25853e-03	5.89058e+00
77	1.05888e+00	1.00217e+00	4.27298e-03	5.74990e+00
78	9.54240e-01	1.00153e+00	4.26433e-03	5.78041e+00
79	1.08583e+00	1.00264e+00	4.44126e-03	5.73331e+00
80	1.04125e+00	1.00314e+00	4.32690e-03	5.64544e+00
81	1.00142e+00	1.00312e+00	4.27040e-03	5.64528e+00
82	9.99268e-01	1.00307e+00	4.21558e-03	5.59989e+00
83	1.09075e+00	1.00417e+00	4.30735e-03	5.70087e+00
84	9.89121e-01	1.00398e+00	4.25733e-03	5.67111e+00
85	1.05128e+00	1.00456e+00	4.38935e-03	5.56920e+00
86	1.02962e+00	1.00486e+00	4.33205e-03	5.68235e+00
87	1.00378e+00	1.00485e+00	4.27712e-03	5.62192e+00
88	9.94941e-01	1.00473e+00	4.21844e-03	5.79162e+00
89	1.02692e+00	1.00499e+00	4.17515e-03	5.76907e+00
90	1.04299e+00	1.00543e+00	4.13663e-03	5.67881e+00
91	1.05452e+00	1.00598e+00	4.16318e-03	5.79737e+00
92	1.00499e+00	1.00597e+00	4.11383e-03	5.72353e+00
93	1.08052e+00	1.00680e+00	4.26667e-03	5.71142e+00
94	9.78412e-01	1.00649e+00	4.23170e-03	5.79313e+00
95	1.05817e+00	1.00705e+00	4.33538e-03	5.67985e+00
96	9.83257e-01	1.00680e+00	4.31484e-03	5.78564e+00
97	1.02705e+00	1.00701e+00	4.30225e-03	5.73712e+00
98	9.98824e-01	1.00692e+00	4.26205e-03	5.67090e+00
99	9.75794e-01	1.00660e+00	4.21266e-03	5.76754e+00
100	1.03116e+00	1.00685e+00	4.17041e-03	5.83370e+00
101	1.02928e+00	1.00708e+00	4.11451e-03	5.83643e+00
102	9.75829e-01	1.00677e+00	4.06466e-03	5.70025e+00
103	9.53697e-01	1.00624e+00	4.02878e-03	5.78531e+00

*** Execution terminated due to completion of the specified number of generations.

Fig. 2.2.13: Example of k-effectives and source entropy by generation.

The data printed include (1) the generation number, (2) the k-effective calculated for the generation, (3) the average value of k-effective through the current generation (excluding the `nskip`-1 generations), (4) the deviation associated with the average k-effective, and (5) Shannon entropy for the generation.

After the last generation, a message is printed to indicate why execution was terminated. If matrix k-effectives were calculated, then this is followed by a message stating the method used to determine the matrix k-effective. The user should examine this portion of the printed results to ensure that the two methods of calculating k-effective are in acceptable agreement and to verify that the average value of k-effective has become relatively stable. If the k-effectives appear to be oscillating or drifting significantly, then the user should consider rerunning the problem with a larger number of histories per generation.

Note: k-eff values from Shift calculation are always printed to the standard output (and `.msg` file). There is no user option to suppress this.

Warning: k-eff uncertainties printed in the message file always show differences compared to those printed in this output edit. Users should use the results printed in this output edit in their analysis. See Sect. 2.2.5 for more details.

Final k-effective edit

The final k-effective edit prints the average k-effective, its associated deviation, and the limits of k-effective for the 67, 95, and 99% confidence intervals. The number of histories used in calculating the average k-effective is also printed. This is done by skipping various numbers of generations. The user should carefully examine the final k-effective edit to determine whether the average k-effective is relatively stable. If a noticeable drift is apparent as the number of initial generations skipped increases, then it may indicate a problem in converging the source. If this appears to be the case, the problem should be rerun with a better initial source distribution and should be run for sufficient number of generations so that the average k-effective becomes stable. The final k-effective edit is printed as shown in Fig. 2.2.14.

```
================================================================================================================================
 Final k-eff Edits
================================================================================================================================

 no. of initial                                                                                                    deviation of
  generations      average                          67 percent           95 percent           99 percent       number of   variance
    skipped      k-effective    deviation      confidence interval  confidence interval  confidence interval  histories  (per cent)

       3          1.00624 + or - 0.00403       1.00221 to 1.01026   0.99818 to 1.01429   0.99415 to 1.01832     100000     11.8824
       4          1.00650 + or - 0.00402       1.00248 to 1.01052   0.99846 to 1.01455   0.99444 to 1.01857      99000     12.1658
       5          1.00661 + or - 0.00407       1.00253 to 1.01068   0.99846 to 1.01475   0.99439 to 1.01882      98000     12.0722
       6          1.00718 + or - 0.00404       1.00314 to 1.01121   0.99911 to 1.01525   0.99507 to 1.01928      97000     12.4957
       7          1.00673 + or - 0.00408       1.00265 to 1.01081   0.99858 to 1.01488   0.99450 to 1.01896      96000     12.5019
       8          1.00697 + or - 0.00410       1.00288 to 1.01107   0.99878 to 1.01517   0.99468 to 1.01926      95000     12.6312
       9          1.00718 + or - 0.00417       1.00301 to 1.01135   0.99883 to 1.01552   0.99466 to 1.01970      94000     12.4203
      10          1.00743 + or - 0.00425       1.00318 to 1.01169   0.99893 to 1.01594   0.99468 to 1.02019      93000     12.2083
      11          1.00672 + or - 0.00434       1.00238 to 1.01106   0.99803 to 1.01541   0.99369 to 1.01975      92000     11.8026
      12          1.00605 + or - 0.00430       1.00175 to 1.01036   0.99745 to 1.01466   0.99314 to 1.01896      91000     12.1895
      17          1.00695 + or - 0.00442       1.00253 to 1.01137   0.99811 to 1.01579   0.99369 to 1.02021      86000     12.8294
      22          1.00747 + or - 0.00459       1.00288 to 1.01207   0.99828 to 1.01666   0.99369 to 1.02125      81000     12.9479
      27          1.00708 + or - 0.00486       1.00222 to 1.01194   0.99736 to 1.01680   0.99250 to 1.02166      76000     13.0425
      32          1.00570 + or - 0.00580       0.99990 to 1.01150   0.99410 to 1.01729   0.98830 to 1.02309      71000     10.3142
      37          1.00622 + or - 0.00656       0.99966 to 1.01278   0.99310 to 1.01934   0.98655 to 1.02590      66000      7.4123
      42          1.00583 + or - 0.00707       0.99876 to 1.01291   0.99169 to 1.01998   0.98461 to 1.02705      61000      7.4243
      47          1.00567 + or - 0.00769       0.99798 to 1.01336   0.99029 to 1.02105   0.98259 to 1.02874      56000      7.3990
      52          1.00638 + or - 0.00819       0.99820 to 1.01457   0.99001 to 1.02275   0.98182 to 1.03094      51000      7.7281
      57          1.00659 + or - 0.00920       0.99739 to 1.01579   0.98819 to 1.02499   0.97899 to 1.03419      46000      7.4012
      62          1.00682 + or - 0.01043       0.99639 to 1.01725   0.98596 to 1.02767   0.97553 to 1.03810      41000      7.1202
      67          1.01014 + or - 0.01071       0.99943 to 1.02085   0.98872 to 1.03156   0.97800 to 1.04228      36000      7.3212
      72          1.01389 + or - 0.01037       1.00352 to 1.02427   0.99315 to 1.03464   0.98277 to 1.04502      31000      9.4548
      77          1.01780 + or - 0.00868       1.00912 to 1.02649   1.00043 to 1.03517   0.99175 to 1.04386      26000     17.3340
      82          1.01814 + or - 0.01076       1.00737 to 1.02890   0.99661 to 1.03967   0.98585 to 1.05043      21000     14.5223
      87          1.01352 + or - 0.01279       1.00073 to 1.02631   0.98794 to 1.03910   0.97515 to 1.05189      16000     13.9137
      92          1.00836 + or - 0.01687       0.99149 to 1.02523   0.97462 to 1.04211   0.95775 to 1.05898      11000     18.1435
      97          0.99410 + or - 0.02331       0.97079 to 1.01741   0.94748 to 1.04072   0.92417 to 1.06402       6000     12.5970
```

Fig. 2.2.14: Example of the final k-effective edit.

Plot of average k-effective by generations run and by generations skipped

ASCII character plots of the average k-effective versus the number of generations run, and the average k-effective versus the number of generations skipped, are not printed by CSAS-Shift in the code output. Instead, two Ptolemy plot files are created and copied into the plots directory in `${OUTDIR}`. The name of the plot files and their final destinations are printed in the output, followed by the final k-effective edit as illustrated in Fig. 2.2.15. These plot files can be loaded and visualized by Fulcrum.

```
================================================================================================================================================
 Plot of average k-effective by generations run
================================================================================================================================================

 Plot file, k6smp18_keff_by_generations00000.ptp, copied to k6smp18.plots directory.


================================================================================================================================================
 Plot of average k-effective by generations skipped
================================================================================================================================================

 Plot file, k6smp18_keff_by_generations_skipped00000.ptp, copied to k6smp18.plots directory.

 ***** k-effective satisfies the chi**2 test for normality at the 95 % level
```

Fig. 2.2.15: Information about the average k-eff plot files.

Shannon Entropy Diagnostics

CSAS-Shift does not perform any posterior entropy tests like those available in KENO codes. Instead, it captures diagnostic test results performed by Shift and prints its details in the Shannon entropy diagnostics output edit, as shown in Fig. 2.2.16

```
================================================================================================================================================
 Shannon Entropy Diagnostics
================================================================================================================================================


Inactive cycles 5.84208 +- 0.0566254
Active cycles 5.71327 +- 0.00709912
First half of active cycles 5.69563 +- 0.00867071
Last half of active cycles 5.73091 +- 0.0107595

The difference between the first and and second half of the active cycles (delta / sigma = 2.55283) is just outside the 95% confidence interval

The first cycle to cross over one sigma of the final entropy was cycle 2, which was before the first active cycle 3.


================================================================================================================================================
 Plot of Shannon Entropy by generation
================================================================================================================================================

 Plot file, k6smp18_shannon_entropy00000.ptp, copied to k6smp18.plots directory.
```

Fig. 2.2.16: Sample Shannon Entropy Diagnostics edit.

Fission densities

The fission density edit is optional. CSAS-Shift prints the neutron production density and the fission density for each geometry region if parameters `FDN=YES` and `NUB=YES` are specified in the parameter data (these are the default values). If `NUB=NO` is specified but `FDN=YES`, then only the production density will be given. An example of the fission density edit is shown in Fig. 2.2.17

```
================================================================================================================================================
 Fission densities
================================================================================================================================================

                                 production   percent        total          fission     percent        total
            unit      region      density     deviation    productions       density     deviation     fissions

               1           1    7.457e-06     0.337       1.006e+00       3.055e-06     0.338       4.123e-01
                           2    0.000e+00     0.000       0.000e+00       0.000e+00     0.000       0.000e+00
                           3    0.000e+00     0.000       0.000e+00       0.000e+00     0.000       0.000e+00
                           4    0.000e+00     0.000       0.000e+00       0.000e+00     0.000       0.000e+00

               2           1    0.000e+00     0.000       0.000e+00       0.000e+00     0.000       0.000e+00

       global unit

               3           1    0.000e+00     0.000       0.000e+00       0.000e+00     0.000       0.000e+00
                           2    0.000e+00     0.000       0.000e+00       0.000e+00     0.000       0.000e+00
                           3    0.000e+00     0.000       0.000e+00       0.000e+00     0.000       0.000e+00
                           4    0.000e+00     0.000       0.000e+00       0.000e+00     0.000       0.000e+00
                           5    0.000e+00     0.000       0.000e+00       0.000e+00     0.000       0.000e+00
```

Fig. 2.2.17: Example of the fission density edit.

Flux Edit

Printing the fluxes is optional; they are printed only if `FLX=YES` is specified in the parameter data. The fluxes are printed for each unit and each geometry region in the unit for every energy group. A sample of a flux edit is given in Fig. 2.2.18.

```
================================================================================================================================================
 Region-averaged fluxes
================================================================================================================================================


 fluxes for Unit        1
             region   1           region   2           region   3           region   4
          (vol= 1.349e+05)     (vol= 1.589e+03)     (vol= 3.025e+04)     (vol= 1.141e+06)

 group      flux     percent     flux     percent     flux     percent     flux     percent
                    deviation            deviation            deviation            deviation
    1    0.000e+00     0.00    0.000e+00     0.00    0.000e+00     0.00    0.000e+00     0.00
    2    0.000e+00     0.00    0.000e+00     0.00    0.000e+00     0.00    0.000e+00     0.00
    3    0.000e+00     0.00    0.000e+00     0.00    0.000e+00     0.00    0.000e+00     0.00
    4    0.000e+00     0.00    0.000e+00     0.00    0.000e+00     0.00    0.000e+00     0.00
    5    3.487e-09    42.14    0.000e+00     0.00    1.408e-09    56.31    7.627e-10    58.70
    6    4.778e-08    12.00    2.444e-08    34.82    3.109e-08    17.49    1.909e-08    15.67
    7    1.752e-07     6.36    1.060e-07    20.19    9.802e-08     8.81    6.165e-08     8.98
    8    6.012e-07     3.45    4.316e-07     8.55    4.054e-07     4.30    2.428e-07     4.39
    9    1.896e-06     1.96    1.198e-06     5.46    1.180e-06     2.49    7.124e-07     2.54
   10    1.061e-06     2.41    8.872e-07    10.09    6.585e-07     3.18    4.109e-07     3.37
   11    4.944e-06     1.11    3.796e-06     3.63    3.037e-06     1.51    1.868e-06     1.58
   12    3.701e-06     1.22    3.088e-06     5.39    2.332e-06     1.72    1.415e-06     1.80
   13    1.098e-06     2.22    9.462e-07    10.47    6.830e-07     3.17    4.175e-07     3.37
   14    4.812e-06     1.03    3.669e-06     3.88    3.090e-06     1.47    1.870e-06     1.56
   15    3.955e-06     1.06    2.976e-06     5.22    2.440e-06     1.57    1.526e-06     1.73
   16    1.345e-06     1.72    9.825e-07    10.18    8.321e-07     2.70    5.039e-07     3.01
   17    5.973e-07     2.53    3.400e-07     9.67    3.337e-07     4.07    2.034e-07     4.68

  ...

  233    4.050e-07     0.99    9.136e-07     6.64    1.207e-06     1.33    1.623e-06     1.64
  234    4.642e-07     0.92    1.032e-06     5.49    1.501e-06     1.21    1.990e-06     1.50
  235    4.949e-07     0.88    1.516e-06     8.86    1.822e-06     1.14    2.362e-06     1.40
  236    5.319e-07     0.84    1.720e-06     5.63    2.147e-06     1.07    2.836e-06     1.32
  237    5.328e-07     0.82    1.677e-06     4.35    2.404e-06     1.05    3.195e-06     1.28
  238    2.380e-07     1.06    9.535e-07    11.21    1.183e-06     1.25    1.588e-06     1.68
  239    6.052e-07     0.77    2.017e-06     3.99    3.432e-06     1.01    4.573e-06     1.20
  240    5.751e-08     1.66    2.083e-07     9.44    3.773e-07     1.82    5.196e-07     2.93
  241    4.387e-08     1.94    2.283e-07    10.77    3.080e-07     2.03    3.800e-07     3.47
  242    1.243e-08     3.09    6.073e-08    19.95    9.531e-08     3.13    1.230e-07     5.88
  243    1.003e-08     3.33    4.426e-08    19.64    7.809e-08     3.32    1.192e-07     6.33
  244    3.385e-09     4.89    1.081e-08    32.35    2.960e-08     4.79    4.465e-08    10.23
  245    2.980e-09     5.32    2.261e-08    43.67    2.569e-08     5.27    3.575e-08    10.85
  246    2.592e-09     5.97    9.218e-09    43.54    2.011e-08     5.99    3.561e-08    13.35
  247    1.045e-09     8.14    7.708e-09    52.68    1.013e-08     8.12    1.441e-08    19.03
  248    5.284e-10    10.70    4.193e-09    65.11    6.511e-09     9.83    1.374e-08    20.79
  249    6.021e-10    10.18    5.586e-10   100.00    5.982e-09     9.44    8.711e-09    22.27
  250    3.741e-10    12.38    9.800e-10   100.00    4.185e-09    10.27    3.842e-09    26.72
  251    2.356e-10    13.15    0.000e+00     0.00    2.691e-09    12.32    5.131e-09    33.98
  252    9.219e-12    49.83    0.000e+00     0.00    1.113e-10    33.58    0.000e+00     0.00
 TOTL    8.971e-05     0.21    7.499e-05     1.04    7.256e-05     0.34    6.145e-05     0.41

 fluxes for Unit        2
             region   1
          (vol= 0.000e+00)

 group      flux     percent
                    deviation
    1    0.000e+00     0.00
    2    0.000e+00     0.00
    3    0.000e+00     0.00
    4    0.000e+00     0.00

    ...
```

Fig. 2.2.18: An example of a flux edit.

Final results table

The final results table contains the best-estimate system k-effective with one standard deviation, the number of warning and error messages generated during code execution, and a final statement on the convergence of the χ^2 test results as shown in Fig. 2.2.19.

```
================================================================================================================================

        ********************************************************************************************************************
        ***                                                                                                                  ***
        ***   sample problem 18   1f27  critical experiment                                                                   ***
        ***                                                                                                                  ***
        ********************************************************************************************************************
        ***                                                                                                                  ***
        ***                                ******    final results table    ******                                         ***
        ***                                                                                                                  ***
        ***        best estimate system k-eff                              1.006505 + or - 0.004020                          ***
        ***                                                                                                                  ***
        ***        number of warning messages                             15                                                ***
        ***                                                                                                                  ***
        ***        k-effective satisfies the chi**2 test for normality at the 95 % level                                     ***
        ***                                                                                                                  ***
        ***                                                                                                                  ***
        ********************************************************************************************************************
        ********************************************************************************************************************
```

Fig. 2.2.19: An example of the final results table.

Final timing report table

The final timing report table summarizes the time elapsed for input processing, cell processing (for multigroup mode), cross section processing (for multigroup mode), the entire transport process (Shift transport), and post-processing performed by the CSAS-Shift sequence after obtaining all results from the Shift transport calculation. A sample timing report obtained for a multigroup calculation is shown in Fig. 2.2.20.

```
================================================================================================================================
 Final timing report (time in minutes)
================================================================================================================================

  Number of MPI processes                          1

  Input processing                           0.00635
  Cell processing                            0.00224
  Cross section processing (XSProc)          0.30361
  Transport (Shift MC transport)             0.43600
  Post-processing/Output edits               0.00018
  -------------------------------------------------
  Sequence total time                        0.74837

================================================================================================================================

================================================================================================================================
```

Fig. 2.2.20: An example of the final results table.

2.3 EXAMPLE APPLICATIONS OF CSAS6

Several example uses of CSAS6 are shown in this section for a variety of applications.

2.3.1 RUN KENO-VI USING CSAS6

CSAS6 creates a microscopic working format library and a mixing table that is passed to KENO-VI. The library is created using XSProc to process the cross section data in the resolved resonance regions of the isotopes contained in the library. CSAS6 then executes KENO-VI, which calculates k_{eff} for the problem. The following examples are for using the multigroup mode of calculation for KENO-VI. Using the continuous energy mode can be accomplished by simply changing the library name to one of the continuous energy libraries.

EXAMPLE 1. CSAS6 – Determine the k_{eff} of a system.

Consider a problem consisting of eight uranium metal cylinders that are 93.2% wt enriched, having a density of 18.76 g/cm^3. The cylinders are arranged in a $2 \times 2 \times 2$ array. Each has a radius of 5.748 cm and a height of 10.765 cm. The center-to-center spacing in the horizontal (X-Y) plane is 13.74 cm and the vertical center-to-center spacing is 13.01 cm. Because the cross section processing will be done assuming an infinite homogeneous medium and no cell mixtures are used, there is no unit cell data. The input data for this problem follow.

```
=csas6
set up 2c8 in csas6
v7.1-252
read comp
uranium  1 den=18.76 1 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp
read parameters  flx=yes fdn=yes far=yes  end parameters
read geometry
unit 1
cylinder 10  5.748 5.3825 -5.3825
cuboid   20  6.87 -6.87 6.87 -6.87 6.505 -6.505
media 1 1 10
media 0 1 20 -10
boundary 20
global unit 2
cuboid 10 4p13.74   2p13.010
array  1  10  place 1 1 1 -6.87 -6.87 -6.505
boundary 10
end geometry
read array
gbl=1 ara=1 nux=2 nuy=2 nuz=2  fill  f1 end fill
end array
end data
end
```

EXAMPLE 2. CSAS6 – Determine the k_{eff} of an array of fuel pellets in a UO_2F_2 solution.

Consider a 60 cm inside diameter cylindrical tank filled with 5.0%-enriched UO_2 fuel rods and 5.0%-enriched UO_2F_2 solution at 295 gm/liter. A $51 \times 51 \times 1$ array of fuel rods is centered on the bottom of the tank. The fuel rods are 366 cm long, 0.45 cm in radius, clad with 0.01-cm-thick Al, and at a pitch of 1.5 cm. The fuel rods sit on the bottom of the container and the container and solution rise 5.0 cm above the top of the rods. The container is 10 cm thick in the side and bottom and open at the top. Determine the k_{eff} of the system. Input data for this problem follow.

```
=csas6
uo2 pins in a uo2f2 solution
```

(continues on next page)

(continued from previous page)

```
v7.1-252
read comp
uo2        1 0.95 300 92235 5.0 92238 95.0 end
al         2 1.0  300 end
solnuo2f2 3 295 0.0 1.0 300 92235 5.0 92238 95.0 end
al         4 1.0  300 end
solnuo2f2 5 295 0.0 1.0 300 92235 5.0 92238 95.0 end
end comp
read celldata
latticecell squarepitch pitch=1.50 3 fueld=0.9  1 cladd=0.94 2 end
end celldata
read geom
unit  1
com='fuel pin'
cylinder 10  0.45  2p183.0
cylinder 20  0.47 2p183.1
cuboid     30  4p0.75 2p183.1
media 1 1 10
media 2 1 20 -10
media 3 1 30 -20 -10
boundary 30
global unit  2
com='fuel pins and solution in tank'
cuboid    10  4p38.25  2p183.1
cylinder  20  60.0  188.1 -183.1
cylinder  30  70.0  188.1 -193.1
array  1  10  place 26 26 1 3*0.0
media  5  1  20 -10
media  4  1  30 -20
boundary  30
end geom
read array
ara=1 nux=51 nuy=51 nuz=1 fill f1 end fill
end array
end data
end
```

2.3.2 RUN KENO-VI CONTAINING CELL-WEIGHTED MIXTURES

CSAS6 creates a microscopic working format library and a mixing table that is passed to KENO-VI. The microscopic cross sections of the nuclides used in the unit cell geometry description are cell-weighted by specifying CELLMIX= followed by a unique mixture number. This mixture number utilizes the cell-weighted cross sections that represent the heterogeneous system. CSAS6 executes KENO-VI and calculates k_{eff} for the problem.

EXAMPLE 1. CSAS6 – Calculate the k_{eff} of an array of fuel assemblies using cell-weighted cross sections.

Consider the $4 \times 4 \times 1$ array of fuel assemblies in a square aluminum cask described in Sect. 2.3.1, Example 2. Calculate the k_{eff} of the system by using the cell-weighted mixture 200 to represent the fuel pins in the fuel assembly. Note that mixtures 1, 2, and 3, representing UO_2, zirconium, and water, respectively, are used in the unit cell description. Cell-weighting is applied to the microscopic cross sections that are used in the cell, making them incorrect for use elsewhere. Because water is used both inside the cell and between the fuel assemblies, an additional mixture, mixture 6, has been added to represent the water between the fuel assemblies. The input data for this problem follow.

```
=csas6
square fuel cask example using homogeneous mockup
v7.1-252
read comp
uo2  1 den=9.21 1.0 293. 92235 2.35 92238 97.65 end
```

(continues on next page)

(continued from previous page)

```
zr   2 1 end
h2o  3 1 end
b4c  4 0.367 end
al   4 0.636 end
al   5 1 end
h2o  6 1 end
end comp
read celldata
latticecell squarepitch pitch=1.3 3 fueld=0.8 1 cladd=0.94 2 cellmix=200 end
end celldata
read param  far=yes gen=253 end param
read geom
unit  2
com='fuel assembly'
cuboid  10 4p11.05  2p183.07
cuboid  20 4p11.70  2p183.72
cuboid  30 4p12.20  2p184.22
media  200  1  10
media  4  1  20  -10
media  6  1  30  -20  -10
boundary  30
global  unit 3
com='fuel cask containing 4x4 array of assemblies'
cuboid  10  4p48.8  2p184.22
cuboid  20  4p58.8  2p194.22
array  1 10  place 1 1 1 -36.6 -36.6 0.0
media  5  1  20 -10
boundary  20
end geom
read array
ara=1 nux=4 nuy=4 nuz=1 fill f2 end fill
end array
end data
end
```

EXAMPLE 2. CSAS6 – Determine the k_{eff} of an array of fuel pellets in a UO_2F_2 solution using cell-weighted cross sections.

This is the same problem as described in Sect. 2.3.1 Example 2. However, the rods and solutions have been replaced with a cell-weighted mixture 50. Determine the k_{eff} of the container. Input data for this problem follow.

```
=csas6
uo2 pins in a uo2f2 solution, cell-weighted mixture
v7.1-252
read comp
uo2        1 0.95 300 92235 5.0 92238 95.0 end
al         2 1.0  300 end
solnuo2f2 3 295 0.0 1.0 300 92235 5.0 92238 95.0 end
al         4 1.0  300 end
solnuo2f2 5 295 0.0 1.0 300 92235 5.0 92238 95.0 end
end comp
read celldata
latticecell squarepitch pitch=1.50 3 fueld=0.9 1 cladd=0.94 2 cellmix=50 end
end celldata
read geom
global unit  2
com='fuel pins and solution in tank'
cuboid    10  4p38.25  2p183.1
cylinder  20  60.0  188.1 -183.1
cylinder  30  70.0  188.1 -193.1
media 50  1  10
media  5  1  20 -10
```

(continues on next page)

(continued from previous page)

```
media  4  1  30 -20
boundary  30
end geom
end data
end
```

2.3.3 RUN KENO-VI CONTAINING MULTIPLE UNIT CELLS

CSAS6 can create a microscopic working format library and a mixing table that contains more than one unit cell. Each unit cell is explicitly defined in the CELLDATA section of the standard composition data. Materials may appear in only one unit cell. All materials in the standard composition that are not contained in a unit cell are processed assuming infinite homogeneous media. CSAS6 passes the created working library to KENO-VI which calculates k_{eff} for the problem.

EXAMPLE 1. CSAS6 – Calculate the k_{eff} of a system using two unit cell descriptions.

Consider an infinite XY-array composed of two types of fuel assemblies in a checkerboard pattern moderated by water. Each assembly consists of a 17 × 17 × 1 array of zirconium-clad, enriched UO_2 fuel pins in a square pitched array. In one array the uranium is 3.5%-enriched and in the other array the uranium is 2.9%-enriched. The UO_2 has a density of 9.21 g/cm^3. The pin diameter is 0.8 cm and is 366 cm long. The clad is 0.07 cm thick, and the pitch is 1.3 cm. Each fuel bundle is contained in a 0.65-cm-thick Boral sheath. The bundles are separated by an edge-to-edge spacing of 1.0 cm. The water and zirconium is input in the standard composition data once for every unit cell in which it appears because a material may appear in only one unit cell. Determine the k_{eff} of the infinite array. Note that periodic boundary conditions are required to obtain an infinite checkerboard array. Input data for this problem follow.

```
=csas6
2 square fuel assemblies example in an infinite lattice of assemblies
v7.1-252
read comp
uo2  1  den=9.21 1.0 293. 92235 3.5 92238 96.5 end
zr   2  1 end
h2o  3  1 end
uo2  4  den=9.21 1.0 293. 92235 2.9 92238 97.1 end
zr   5  1 end
h2o  6  1 end
b4c  7  0.367 end
al   7  0.636 end
end comp
read celldata
latticecell squarepitch pitch=1.3 3 fueld=0.8 1 cladd=0.94 2 end
latticecell squarepitch pitch=1.3 6 fueld=0.8 4 cladd=0.94 5 end
end celldata
read param far=yes gen=253 end param
read geom
unit  1
com='3.5 w% fuel pin'
cylinder 10  0.4  2p183.0
cylinder 20  0.47 2p183.07
cuboid   30  4p0.65 2p183.07
media 1 1 10
media 2 1 20 -10
media 3 1 30 -20 -10
boundary 30
unit  2
com='3.5 w% fuel assembly'
cuboid  10  4p11.05  2p183.07
cuboid  20  4p11.7   2p183.72
cuboid  30  4p12.2   2p184.22
```

(continues on next page)

(continued from previous page)

```
array  1 10  place 9 9 1 3*0.0
media  7  1  20 -10
media  3  1  20 -20 -20
boundary  30
unit  3
com='2.9 w% fuel pin'
cylinder 10  0.4  2p183.0
cylinder 20  0.47 2p183.07
cuboid   30  4p0.65 2p183.07
media 4 1 10
media 5 1 20 -10
media 6 1 30 -20 -10
boundary 30
unit   4
com='2.9 w% fuel assembly'
cuboid  10  4p11.05  2p183.07
cuboid  20  4p11.7   2p183.72
cuboid  30  4p12.2   2p184.22
array  2 10  place 9 9 1 3*0.0
media  7  1  20 -10
media  6  1  20 -20 -20
boundary  30
global  unit 5
com='fuel cask containing 4x4 array of assemblies'
cuboid   10  4p24.4  2p184.22
array  3 10  place  1 1 1 -12.2  -12.2  0.0
boundary 10
end geom
read array
ara=1 nux=17 nuy=17 nuz=1 fill f1 end fill
ara=2 nux=17 nuy=17 nuz=1 fill f3 end fill
gbl=3 ara=3 nux=2 nuy=2 nuz=1 fill 2 4 4 2 end fill
end array
read bounds xyf=periodic end bounds
end data
end
```

EXAMPLE 2. CSAS6 – Calculate the k_{eff} of a system using two unit cell descriptions and cell-weighted mixtures.

Consider a problem in which a stainless steel cylinder with an inner diameter of 56 cm and an inside height of 91 cm is filled with pellets of UO_2 in borated water. The steel is 0.125 cm thick. The spherical 2.57%-enriched UO_2 pellets have a diameter of 1.07 cm and are arranged in a triangular pitch array with a pitch of 1.13 cm. The spherical 2.96%-enriched UO_2 pellets have a diameter of 1.07 cm and are arranged in a triangular pitch array with a pitch of 1.12 cm. The cylindrical tank is filled half full of the 2.96% pellets in borated water, and the remainder is filled with the 2.57%-enriched pellets in borated water.

Mixture 100 is the cell-weighted mixture containing the 2.57%-enriched uranium pellets and mixture 200 is the cell-weighted mixture containing the 2.96%-enriched uranium pellets. Determine the k_{eff} of this system. Input data for this problem follow.

```
=csas6
2.57% and 2.96% enr uo2 pellets in 3500 ppm borated water
v7.1-252
read comp
uo2    1 0.925  283 92235 2.57 92238 97.43 end
h2o    2 1.0 283 end
atombacid 2 2.0017-2 3 5000 1 1001 3 8016 3 1.0 283 end
uo2    3 0.925  283 92235 2.96 92238 97.04 end
h2o    4 1.0 283 end
atombacid 4 2.0017-2 3 5000 1 1001 3 8016 3 1.0 283 end
```

(continues on next page)

(continued from previous page)

```
ss304 5 1.0 283 end
end comp
read celldata
latticecell  sphtriangp pitch=1.13 2 fueld=1.07 1 cellmix=100 end
latticecell  sphtriangp pitch=1.13 4 fueld=1.07 3 cellmix=200 end
end celldata
read param  flx=yes   end param
read geom
global unit 1
cylinder 10 38.0 45.5 0.0
cylinder 20 38.0 91.0 0.0
cylinder 30 38.125 91.0 -0.125
media 100 1 10
media 200 1 20 -10
media   5 1 30 -20
boundary 30
end geom
end data
end
```

2.4 DEVC: DENOVO EIGENVALUE CALCULATION

Douglas E. Peplow and Cihangir Celik

2.4.1 INTRODUCTION

The DEVC (Denovo EigenValue Calculation) sequence is an interface to the Denovo discrete-ordinates package [DEVCESSC10] for calculating criticality eigenvalue problems. This sequence reads an input file very similar to a CSAS6 input file [DEVCGPJD+11] that contains an extra block of input for describing the Denovo mesh grid and calculational parameters. Many of the subroutines are shared from the MAVRIC routines that interface with Denovo for fixed-source calculations.

This manual assumes that the user is familiar with the discrete-ordinates method for radiation transport and the Denovo package. DEVC provides an easy way for users to modify existing CSAS6 inputs and use them to run Denovo. The DEVC sequence also provides a way to create mesh geometry for Denovo from the combinatorial solid geometry description used by KENO-VI.

The steps in the DEVC sequence are listed in Table 2.4.1.

Table 2.4.1: Steps in DEVC for an input file named *input*.inp

Step	Module/Task	Creates file	To stop after
0	Check user input		
1	Self-shielding (celldata/cellm ix) calculations		
2	Produces optional *.png plots		
	Produces optional *.3mdap files (to visualize grid in MeshFileViewer)		`parm=check`
3	Creates AMPX cross sections for the "real" materials	`ft02f001`	`parm=cross`
4	Creates Denovo binary stream input file and the macromaterial table file	`xkba_b.inp` `input.mmt`	`parm=input`
5	Runs Denovo to compute k_{eff} and either the fluxes or the fission source	`input.dff` or `input.dso`	

The DEVC sequence uses KENO-VI geometry. Users can specify what output Denovo will generate: fluxes by space and energy in a binary *.dff (Denovo flux file) file or the space-only fission source distribution in a binary *.dso (Denovo spatial output) file. The eigenvalue is printed in the main output text file.

Some of the more common KENO starting source types are supported. Other starting source types may be added or extended to all of the different array types in the future. Currently, starting sources are not sent to Denovo because the Arnoldi solver does not use it. This may change in the future.

2.4.2 SEQUENCE INPUT

The input file for a DEVC calculation looks similar to a CSAS6 input file, as shown in Table 2.4.2. The major difference is that the parameter block contains information for the Denovo calculation, not the KENO Monte Carlo calculation. A macromaterial block is used to describe how the KENO-VI materials are mapped onto the Denovo mesh grid. Only multi-group cross-section libraries can be used with Denovo.

Table 2.4.2: Input file for a DEVC calculation (and differences with a CSAS6 input file, where black text is the same as CSAS6 and green text is new for DEVC sequence)

`=devc parm=…`
`Title for the problem`
`library`
	Parm: check, cross, input
	Multi-group cross-section library name

`read xxx` `end xxx`	Standard CSAS6 blocks used by DEVC: composition, celldata, geometry, array, plot and gridGeometry.
`read bounds` `end bounds`	Bounds are used if the boundary of the global unit is a cuboid.
`read start` `end start`	Most KENO start types are supported for some array types. Currently, Denovo does not make use of a starting source.
`read xxx` `end xxx`	Not used by DEVC: volume, x1ds, energy, biasing, importance, reactions, and search

`read parameters`
`gridGeometry 7`
`…`
`end gridGeometry`
	Denovo discrete-ordinates parameters
	Mesh grid – list of planes in each dimension

`quadType=2`
`polarsperoct=4`
`azimuthsperoct=4`
`ktolerance=1.0e-5`
`…`
	List of quadrature order, Legendre order, upscatter, and other eigenvalue calculation parameters

`xmin=0` `…`	Boundary conditions – these override conditions listed in the bounds block
`fissionSource` `end parameters`	Save fission source instead of fluxes

`read macromaterial`
`mmSubCell=3`
`mmTolerance=0.001`
`end macromaterial`
	How to create materials for each voxel in the mesh grid

`read xxx` `end xxx`	
`end data` `end`	

2.4.2.1 Parameters block

This block contains the parameters for the Denovo eigenvalue calculation, the grid geometry, and the macromaterials. Boundary conditions listed in the parameters block will override those listed in the bounds block (using CSAS6 syntax). Table 2.4.3 lists the Denovo calculation parameters and their default values, and Table 2.4.4 lists the keywords for the setting the boundary conditions and file saving options. The grid geometry is defined in a sub-block in the parameters block, or the keyword `"gridGeometryID=\ *n*\ "` can be used to point to a grid geometry defined in its own input block.

Table 2.4.3: Denovo parameters in the parameters block

keyword	type	default	restrictions/comments
read parameters			
discretization=	integer	4	0-diamond difference, 1-DD with flux fix-up:lin0, 2-theta-weighted DD, 3-linear discontinuous finite element, 4-step characteristics, 5-trilinear discontinuous finite element
quadType=	integer	2	0-level symmetric, 1-Gauss-Legendre product, 2-QR
legendre=	integer	3*	P_L, L=highest Legendre polynomial, L=0,1,2,3,… *default is to use min(the highest available in the data,3)
tportcorrection=	integer	1	transport correction: 0-none, 1-diagonal, 2-Cesaro *P_2 or higher is required for Cesaro
upScatter=	integer	1	upscatter iterations: 0-none, 1-yes, 2-ignore
xblocks=	integer	1	parallel calcs - how many divisions in x
yblocks=	integer	1	parallel calcs - how many divisions in y
zblocks=	integer	1	parallel calcs - how many commication layers in z
numSets=	integer	1	parallel calcs - how many energy sets
partUpscatter=	integer	1	partition upscatter (0-no, 1-yes)
quadrature=	integer	8	level symmetric S_N quadrature, N=2, 4, 6, 8, 10, 12, 14, 16
polarsPerOct=	integer	3	Gauss-Legendre product quadrature or QR
azimuthsPerOct=	integer	3	Gauss-Legendre product quadrature or QR
maxIters=	integer	1000	maximum number of iterations
diagnostics=	integer	0	0-no diganostics, 1-all diagnostics
output=	integer	0	0-no ouput, 1-all output
krylovSpaceSize=	integer	25	size in memory for Krylov space
tolerance=	double	1.E-04	tolerance used in convergence test
krylovType=	integer	0	0-GMRES, 1-BiCGStab
eigenSolver=	integer	1	0-power iteration, 1-Arnoldi, 2-shifted inverse
multiGSolover=	integer	1	0-Gauss-Seidel, 1-Krylov
withinGSolver=	integer	0	0-Krylov, 1-residual Krylov, 2-source iteration
mgSettings=	integer	0	0-user supplied, 1-automatic by Denovo
upGroupSolver=	integer	0	0-same as within-group solver, 1-Krylov, 2-residual Krylov, 3-source iteration, 4-single source iteration
acceleration=	integer	0	0-none, 1-two grid
maxItersMG=	integer	1000	maximum number of iterations
toleranceMG=	double	1.E-04	tolerance used in convergence test
keff=	double	1	starting guess of k
kTolerance=	double	1.E-05	tolerance on k
diagnosticLevel=	integer	0	0-off, 1-on
eigenSettings=	integer	0	0-user supplied, 1-automatic by Denovo
l2Tolerance=	double	0.01	tolerance on the L-2 norm
linfTolerance=	double	0.001	tolerance on L-inf
powerIterAcc=	integer	0	0-none, 1-rebalance, 2-RQI
arnoldiKSpace=	integer	25	
arnoldiRestarts=	integer	100	
calcMoments=	integer	1	0-off, 1-on
end parameters			

Table 2.4.4: Boundary conditions and what type of file to save

keyword	type	default	restrictions/comments
read parameters			
xmin=	integer	0	boundary conditions on the six faces:
xmax=	integer	0	0 - vacuum
ymin=	integer	0	1 - reflective
ymax=	integer	0	2- periodic
zmin=	integer	0	3 - white
zmax=	integer	0	(these will override the "read bounds" block if specified here.)
fluxes			save space/energy fluxes (default)
fissionSource			save the space-only fission source
end parameters			

2.4.2.2 Grid geometry block

Grid geometries ("`gridGeometry` *id*") require an identification number and then a description of a three-dimensional rectangular mesh by specifying the bounding planes of the cells in each of the *x*, *y*, and *z* dimensions. The keyword "`xPlanes ... end`" can be used to list plane values (in any order). The keyword "`xLinear` *n a b*" can be used to specify *n* cells between *a* and *b*. The keywords "`xPlanes`" and "`xLinear`" can be used together and multiple times — they will simply add planes to any already defined for that dimension. Any duplicate planes will be removed. Similar keywords are used for the *y* and *z* dimensions.

When using multiple instances of the keywords *``Linear`` and *``Planes`` for a given dimension, duplicates should be removed from the final list. In some cases, double precision math will leave two planes that are nearly identical but not removed (e.g., 6.0 and 5.9999999). To prevent this, a default tolerance is set to remove planes that are within 10^{-6} cm of each other. The user is free to change this by using the keyword "`tolerance=`" and specifying something else. Note that the tolerance can be reset to a different value in between each use of *``Linear`` or *``Planes``.

The keyword "`make3dmap`" for a particular grid geometry definition will create a file called "*outputName*.grid*id*.3dmap", which can be visualized using the Java Mesh File Viewer. These files will contain crude geometry information (unit, region, material) that corresponds to the center of each voxel.

Keywords for the grid geometry block are listed in Table 2.4.5.

Table 2.4.5: Grid geometry input keywords

keyword	type	restrictions/comments
read parameters		
gridGeometry *id*		covers a cuboid where all points are in defined regions
title=	character	contained in double quotes (title="yada yada yada")
x planes … end	real array	a list of *x* plane values (order not important)
x Linear *n min max*	int real real	adds *n*+1 *x* planes (*n* cells) evenly from min to max
y planes … end	real array	a list of *y* plane values (order not important)
y Linear *n min max*	int real real	adds *n*+1 *y* planes (*n* cells) evenly from min to max
z planes … end	real array	a list of *z* plane values (order not important)
z Linear *n min max*	int real real	adds *n*+1 *z* planes (*n* cells) evenly from min to max
tolerance=	real	for removing duplicate planes from *planes or *Linear
make3dmap		makes a *.3dmap file showing the grid geometry
xdivide=	integer	once all xplanes are entered, further divide them by this
ydivide=	integer	once all yplanes are entered, further divide them by this
zdivide=	integer	once all zplanes are entered, further divide them by this
end gridGeometry		
gridGeometryID=	integer	Either define a grid geoemtry in a sub block or refer to a grid geometry defined in its own block
end parameters		

2.4.2.3 Macromaterial block

In order to get more accurate solutions from a coarse-mesh discrete-ordinates calculation, Denovo can represent the material in each voxel of the mesh as a volume-weighted mixture of the real materials in the problem. When constructing the Denovo input, DEVC can estimate the volume fraction taken by each real material in each voxel by a sampling method. The user can specify parameters for how to sample the geometry. Note that finer sampling makes more accurate estimates of the material fraction but requires more setup time to create the Denovo input. Users should understand how the macromaterials are sampled and consider that when constructing a mesh grid. This is especially important for geometries that contain arrays. Careful consideration should be given when overlaying a mesh on a geometry that contains arrays of arrays.

Because the list of macromaterials could become large, the user can also specify a tolerance for how close two different macromaterials can be to be considered the same, thereby reducing the total number of macromaterials. The macromaterial tolerance, "`mmTolerance=`", is used for creating a different macromaterial from the ones already created by looking at the infinity norm between two macromaterials.

The number of macromaterials does not appreciably impact Denovo run time or memory requirements.

Keywords for the macromaterial block are listed Table 2.4.6. Two different sampling methods are available — ray tracing [DEVCIPE+09] with the keyword `mmRayTest` and point testing [DEVCJoh13] with the keyword `mmPointTest`.

Table 2.4.6: Macromaterial block input

keyword	type	length	default	restrictions/comments
read macromaterial				
mmSubCell=	integer		1	rays per dimension to throw at each voxel
mmTolerance=	real		0.01	smallest volume fraction for macromaterial
mmSubCells	integer	6		rays per dimension to throw (x:ny,nz; y:nx,nz; z: nx,ny)
mmPointTest				use recursive bisection point testing method
mmRayTest				use ray tracing method
mmRTSpeed				optimize ray-tracing method for speed
mmRTMemory				optimize ray-tracing method for memory conservation
end macromaterial				

Ray tracing

This method estimates the volume of different materials in the Denovo mesh grid elements by throwing rays through the KENO-VI geometry and computing the average track lengths through the each material. Rays are traced in all three dimensions to better estimate the volume fractions of materials within each voxel. The `mmSubCell` parameter controls how many rays to trace in each voxel in each dimension. For example, if `mmSubCell`= n, then when tracing rays in the z dimension, each column of voxels uses a set of $n \times n$ rays starting uniformly spaced in the x and y dimensions. With rays being cast from all three orthogonal directions, then a total of $3n^2$ rays are used to sample each voxel. One can think of subcells as an equally spaced sub-mesh with a single ray positioned at each center. The number of subcells in each direction, and hence the number of rays, can be explicitly given with `mmSubCells ny nz nx nz nx ny end` keyword for rays parallel to the x axis, y axis, and z axis. Fig. 2.4.1 shows different subcell configurations (in two dimensions) for a given voxel.

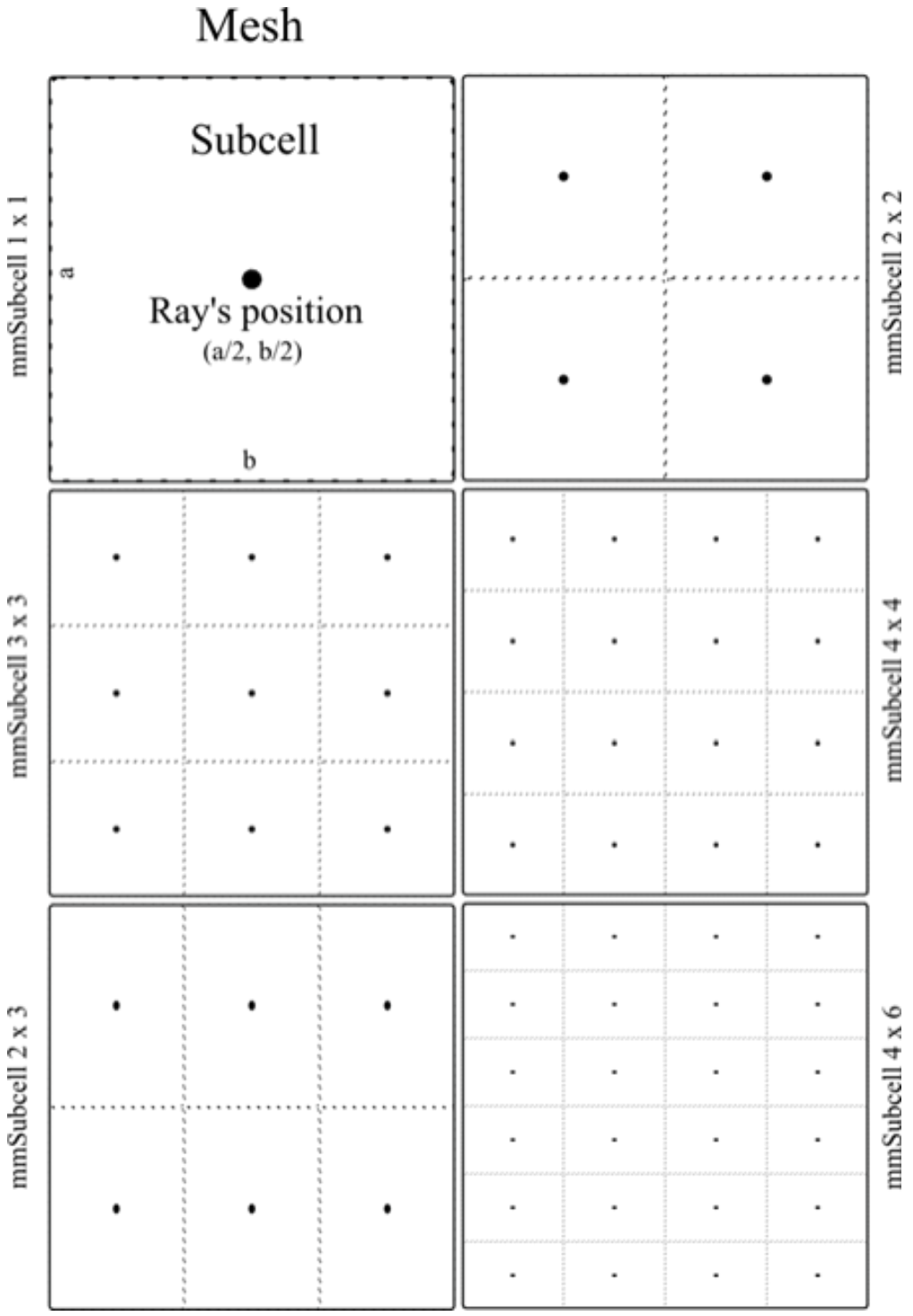


Fig. 2.4.1: Ray positions within a voxel with different mmSubCells parameters.

Ray tracing is a more robust method compared to the simple point testing method used in previous versions of SCALE/MAVRIC; however, it requires more memory than point testing. Ray tracing gives more accurate estimates of volume fractions because track lengths across a voxel give more information than a series of test points. Ray tracing is also much faster than point testing because the particle tracking routines are optimized

for quickly determining lists of materials and distance along a given ray.

Ray tracing operates on the grid geometry supplied by the user and shoots rays in all three directions starting from the lower bounds of the mesh grid. An example of arbitrary assembly geometry is shown in Fig. 2.4.2. A ray consists of a number of steps that corresponds to crossing a different material along the path of the ray. Ratios of each step's length to the voxel length in the ray's direction determine the material volume fraction of that step in that voxel, and summation of the same material volume fractions gives the material volume fraction of that material in that voxel. Ray tracing through a single voxel that contains a fuel pin is illustrated in Fig. 2.4.3.

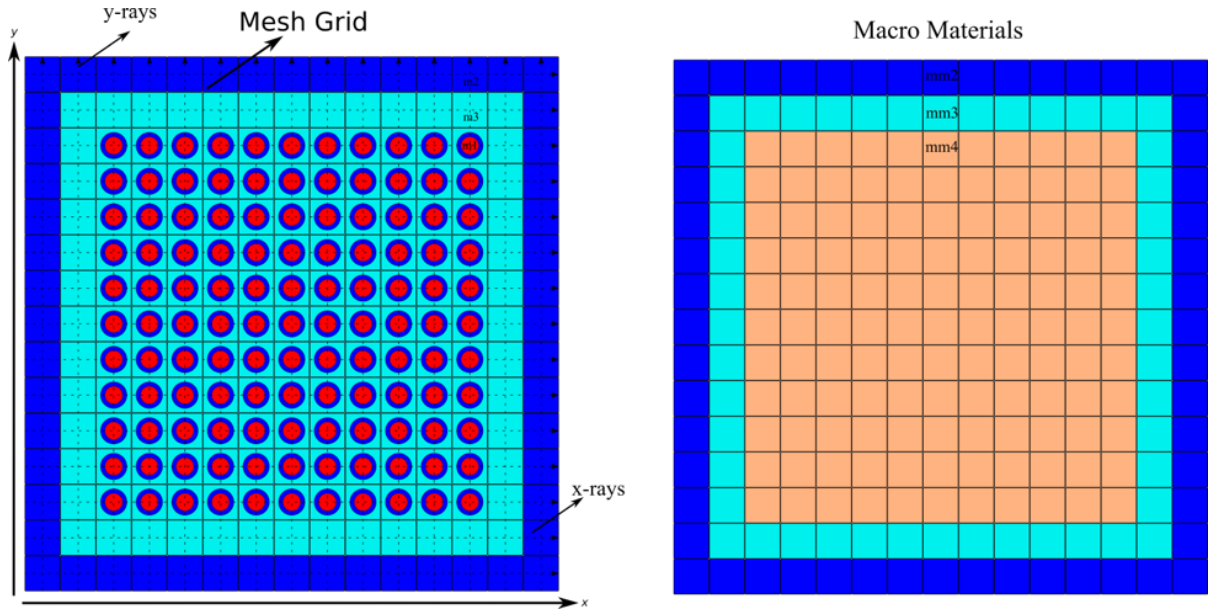


Fig. 2.4.2: Geometry model (left) and the Denovo representation (right) of an assembly using macromaterials determined by ray tracing.

The final constructed macromaterials for this model are also shown in Fig. 2.4.2. Voxels that contain only a single material are assigned that original material number in the constructed macromaterials. For the voxels that contain a fuel pin with three different materials, the result is a new macromaterial consisting of the volume weighted fractions of each original material.

After the rays are shot in all three directions, the material volume fractions are updated and macromaterials are created by using these material volume fractions. Material volume fraction calculations for a single voxel, as shown in Fig. 2.4.3, are given by

$$F_m = \sum_{d=x,y,z} \sum_{r=1}^{N_r} \sum_{s=1}^{N_s} \begin{cases} \frac{L_{d,r,s}}{L_d}, & m_s == m \\ 0, & \text{otherwise} \end{cases} \quad \text{and} \quad V_m = \frac{F_m}{\sum_{n=1}^{N_m} F_n}, \tag{2.4.1}$$

where F_{m} = sampled fraction of material m in the voxel,

d = direction of the rays (x, y, z),

r = ray number,

N_{r} = total number of rays in the voxel for direction of d,

s = step number,

N_s = total number of steps for ray r in the voxel for direction of d,

$L_{d,r,s}$ = length of the steps s for ray r in the voxel for direction of d,

$L_{d,}$ = length of the voxel along direction of d,

m_s = material of step s,

m = material number,

N_m = total number of materials in the voxel, and

V_m = volume fraction of material m in the voxel.

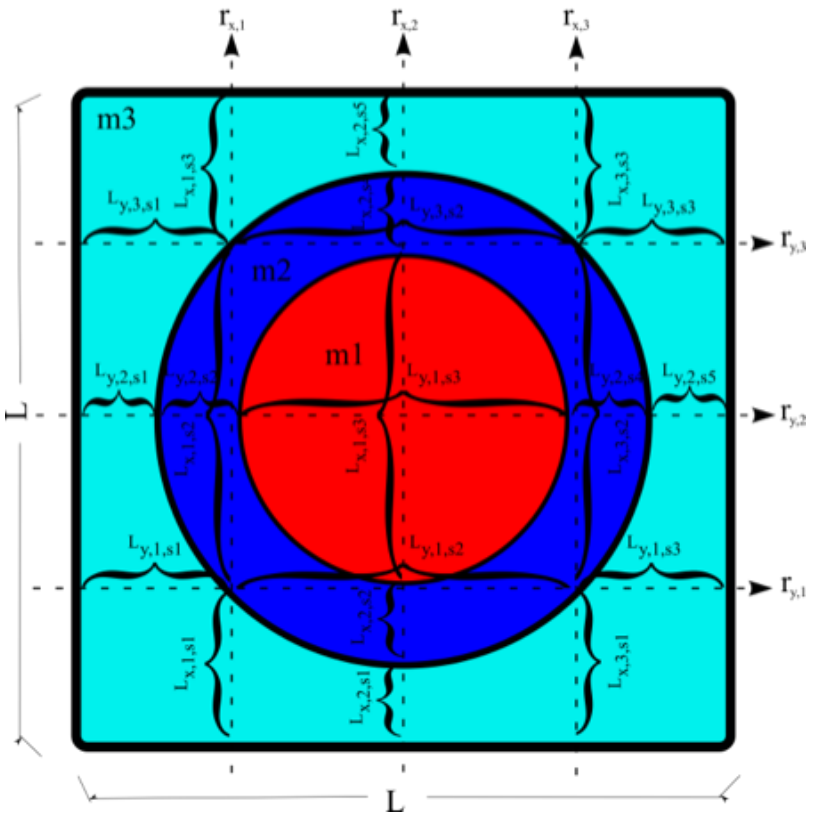


Fig. 2.4.3: Ray tracing (in two dimensions) through a voxel.

Point testing

The recursive bisection method uses a series of point tests to determine the macromaterial fractions. For a given voxel, the material at the center is compared to the material at the eight corners. If they are all the same, the entire volume is considered to be made of that material. If different, the volume is divided into two in each dimension. Each subvolume is tested, and the method is then applied to the subvolumes that are not of a single material. When the ratio of the volume of the tested region to the original voxel becomes less than a user-specified tolerance (in the range of 10^{-1} to 10^{-4}), then further subdivision and testing are stopped. This is illustrated in Fig. 2.4.4.

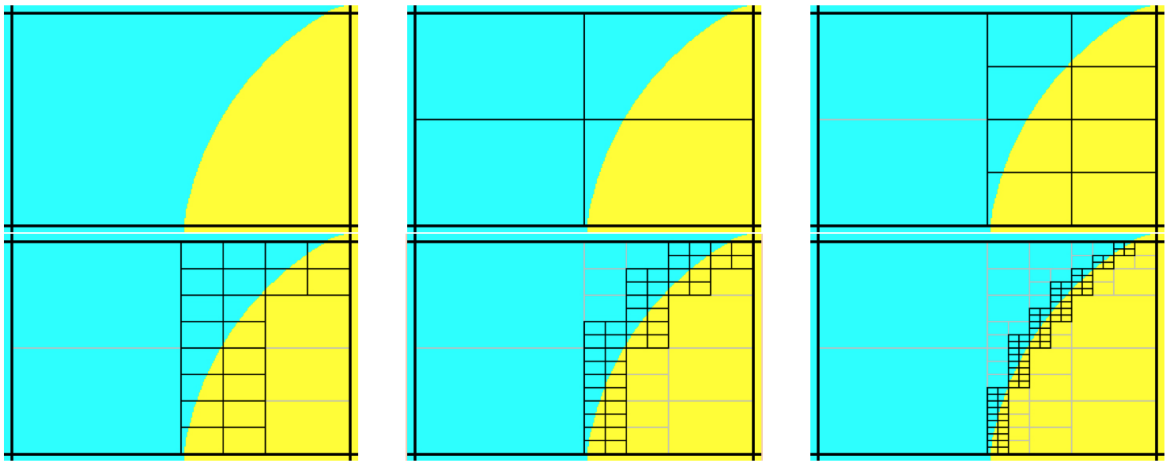


Fig. 2.4.4: Progression of the recursive bisection method (from upper left to lower right).

In point testing, the keyword "`mmTolerance=`*f*" is interpreted to be where *f* is the smallest fraction of the voxel volume to consider. This same tolerance *f* is also used to limit the number of macromaterials. Before a new macromaterial is created, if one already exists where the fraction of each actual material matches to within the given tolerance, then the existing material will be used. If using only a single point at the center of each voxel, use "`mmTolerance=1`". The `mmSubCell` keyword is not used in point testing.

Example

Fig. 2.4.5 shows an example of a cask geometry with two types of spent fuel (yellows), steel (blue), resin (green), and other metals (gray). When the Denovo geometry is set up by testing only the center of each mesh cell, the curved surfaces are not well represented (upper right). By applying the ray-tracing method and defining a new material made of partial fractions of the original materials, an improved Denovo model can be made. In the lower left of the figure, the Denovo model was constructed using one ray (in each dimension) per voxel and a tolerance of 0.1. This gives 20 new materials that are a mixture of the original 13 actual materials and void. With `mmSubCells=3` and an `mmTolerance=0.01`, 139 macromaterials are created.

A macromaterial table listing the fractions of each macromaterial is saved to a file called "*outputName*.mmt", where *outputName* is the name the user chose for his or her output file. This file can be used by the Mesh File Viewer to display the macromaterials as mixtures of the actual materials, as seen in lower row of Fig. 2.4.5. See the Mesh File Viewer help pages for more information on how to use colormap files and macromaterial tables.

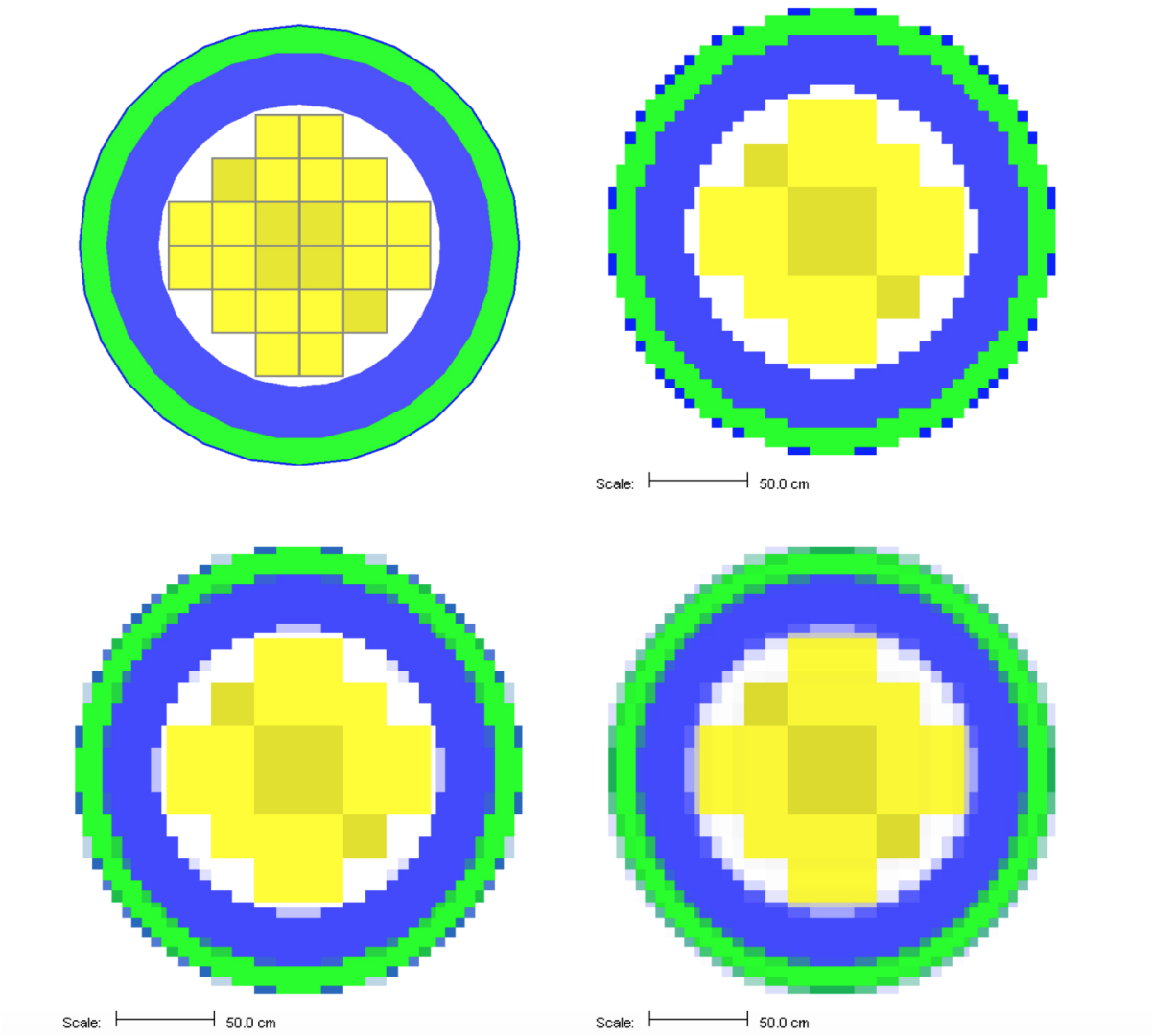


Fig. 2.4.5: Cask geometry model (upper left) and the Denovo representation using (upper right) cell center testing. Representations using macromaterials determined by ray tracing are shown for (lower left) mmSubCell=1/mmTolerance=0.1 and (lower right) mmSubCell=3/mmTolerance=0.01.

2.4.2.4 Starting sources block

The default KENO-VI starting source is "flat over the volume specified by the unrotated, untranslated geometry record specified in the first position of the global unit boundary record in fissile material only". For DEVC, the default starting source strength is uniform in the fissile voxels contained within the bounding box of the global unit (uniform density). If macromaterials are used, the amounts in each voxel are volume averaged between fissile and non-fissile materials. Table 2.4.7 and Table 2.4.8 describe the starting sources available in the DEVC interface to Denovo.

Table 2.4.7: Starting source types (within the fissile areas of the listed shape)

	KENO-VI start type	DEVC
nst=0	the first surface of boundary (default)	The bounding box of global unit
nst=0	Within boundary of global array having a reflector key set to false	Not supported
nst=0	A cuboid defined by XSM, XSP, YSM, YSP, ZSM, and ZSP	Supported
nst=1	A cuboid defined by XSM, XSP, YSM, YSP, ZSM, and ZSP with cosine distributions	Supported
nst=2	Arbitrary fraction (FCT) in element NXS, NYS, NZS of the global array with the remainder in a cuboid defined by XSM, XSP, YSM, YSP, ZSM, and ZSP with cosine distributions	Supported for some array types (see Table 2.4.8)
nst=3	At the location TFX, TFY, TFZ in the element NXS, NYS, NZS of the global array	Supported for some array types (see Table 2.4.8)
nst=4	At the location TFX, TFY, TFZ in units NBX of the global array	Supported
nst=5	Across units NBX in the global array	Not supported
nst=6	List of points TFX, TFY, TFZ in global coordinates	Limited to 1 point
nst=6	List of points TFX, TFY, TFZ in element NXS, NYS, NZS of the global array	Limited to 1 point and only for some array types (see Table 2.4.8)
nst=7	Flat distributions in X and Y with [1-$\cos^2$(z)] in Z for a cuboid defined by XSM, XSP, YSM, YSP, ZSM, and ZSP	Supported
nst=8	Flat distributions in X and Y with a segmented distribution in Z for a cuboid defined by XSM, XSP, YSM, YSP, ZSM, and ZSP	Not supported
nst=9	Use a mesh source lite file	Not supported

Table 2.4.8: Supported array types for starting sources

no arrays	supported
cuboid	supported
hexagonal	supported
shexagonal	no
rhexagonal	no
dodecahedral	no

The starting source initialized in Denovo is always a volumetric source—DEVC does not create point sources (which would activate the first collision option in Denovo).

2.4.3 SEQUENCE OUTPUT

The main text output file consists of the output from the cross-section processing codes and Denovo. The user should examine the output and pay attention to any warnings or errors. Table 2.4.9 lists the files generated during the DEVC sequence.

Table 2.4.9: Files created by DEVC for an input file named *input*.inp

Filename		**Viewer**	**Description**
Output Summary			
	input.out		main text output file, contains k_{eff}
	input.msg		messages file
Denovo			
	xkba_b.inp	V[a]	input file for Denovo - if this file is renamed to have a *.dsi extension (Denovo simple input), it is viewable in the Mesh File Viewer
	ft02f001		AMPX formatted cross sections for Denovo
	input.mmt	V	macromaterial table, use with *.dso or *.dsi file
	input.dff	V	Denovo fission fluxes
	input.dso	V	Denovo fission source distribution
[a]V - can be displayed with the Mesh File Viewer.			

2.4.3.1 Using the mesh file viewer

The Mesh File Viewer is a Java utility shipped with SCALE that is used for viewing mesh tallies from Monaco, as well as importance maps and mesh-based sources in MAVRIC. The Mesh File Viewer can be used with DEVC to view the Denovo input file (*.dsi, showing the starting source), the Denovo output fission source distribution (*.dso) file, or the Denovo output flux (*.dff) file.

With any Denovo file, material information for each voxel can be displayed. Users can use the graphical user interface to select colors for each material in the DEVC input and save them to a colormap text file (*.cmp) for later use. For example, Fig. 2.4.6 shows a cask model that has 13 materials with the default colors assigned by the viewer. A better color map (uranium is yellow, steel is blue, etc.) and an image using that color map are also shown in Fig. 2.4.6.

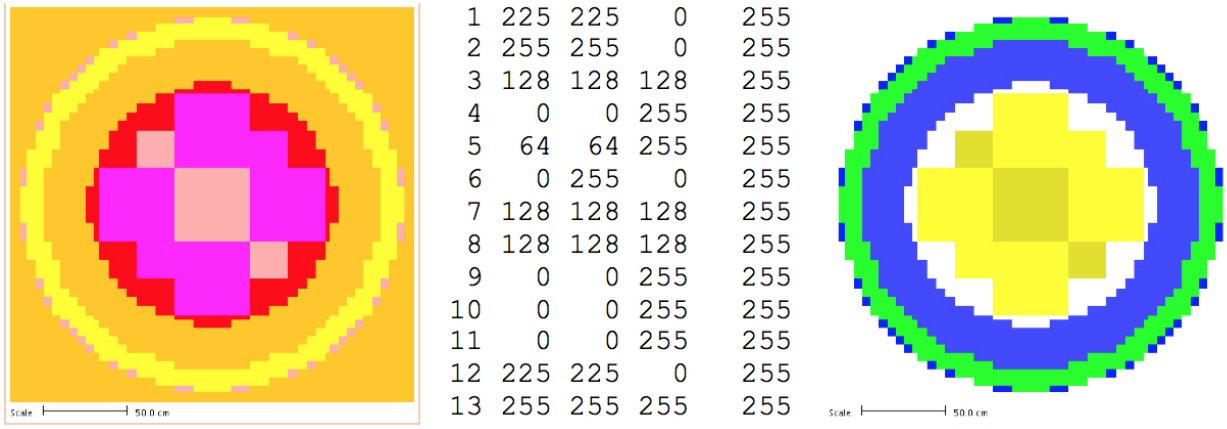


Fig. 2.4.6: Viewing material information with the Mesh File Viewer.

If the Denovo input was made using the mixed macromaterials, that can also be shown with the Mesh File Viewer. When one of the mesh files is loaded and the geometry is displayed (e.g., the materials for a cask model shown in Fig. 2.4.7), there will be many values since each macromaterial is treated by the Java viewer as a separate material. Redefining the color map using the GUI is not realistic.

A colormap needs to be defined for the original materials of the problem. For this example, there are 13 original materials, but when using macromaterials, 73 macromaterials are used in the discrete-ordinates model (as listed in the *.mmt file). The colormap file should only contain values for the 13 original materials. When loading the colormap, a corresponding *.mmt file can also be loaded. This will display mixed colors for the macromaterials, as shown in Fig. 2.4.7.

The final result is an image where the colors for the different values of the geometry attribute (which in the above example is material) are mixed in the ratio of the macromaterial definition.

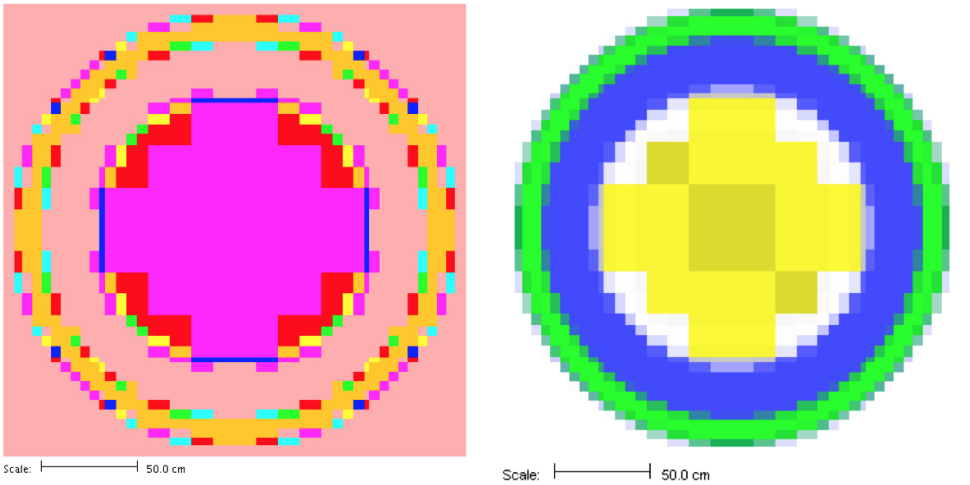


Fig. 2.4.7: Viewing material information from a Denovo input file.

2.4.3.2 Viewing the starting source

The Java MeshFileViewer (scale/cmds/meshview) can be used to visualize the Denovo binary stream input file to show materials and starting source strengths. An example is shown in Fig. 2.4.8.

Material Numbering

The cross sections produced by the SCALE cross-section mixing module ICE typically have a number of materials equal to the maximum material ID specified by the user in the "read composition" block. For a user input that contains materials 1, 3, and 5, the GIP-formatted cross-section file will contain five entries. For each lattice cell calculation that uses a cellmix= parameter, one more entry is contained in the GIP file. Hence, the GIP file does not contain the user's value from the "cellmix=" parameter but instead numbers the cellmix materials sequentially starting from 1+maximum(material ID). The Denovo geometry input and macroMixTable file produced by DEVC are modified to match the GIP cross-section file. When viewing the Denovo binary input file with the Mesh File Viewer, the GIP numbering for materials will be seen.

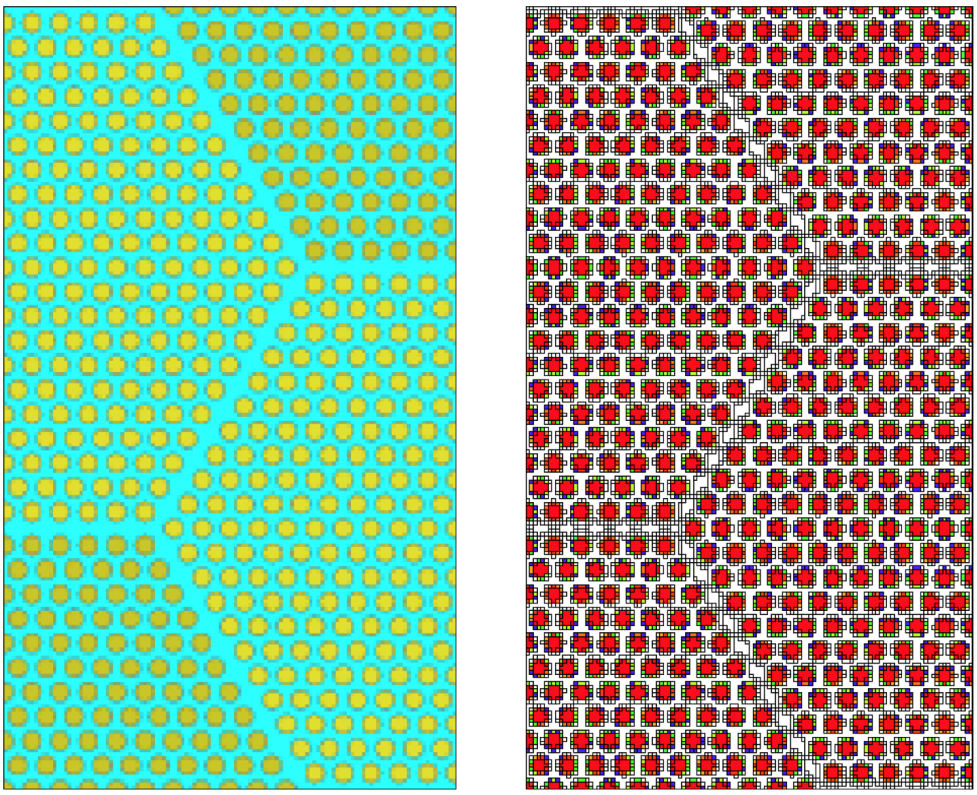


Fig. 2.4.8: Fine-mesh version of sample problem 8, showing the materials using macromaterial blending of UO_2, MOX, clad, and water (left) and the starting source strength limited to cells with fissionable material (right).

2.4.4 SAMPLE PROBLEMS

Sample problems have been made that correspond to the eight CSAS6 sample problems. In each problem, the KENO parameters block was commented out, a parameters block containing Denovo calculation parameters was added, and a grid geometry block was added. These sample problems use a coarse discretization and loose tolerances to obtain a short runtime. Users will typically use much finer discretization (mesh, quadrature) and higher fidelity parameter settings for real eigenvalue calculations. The voxelized geometry and starting source distribution are shown below in Fig. 2.4.9.

Results for the sample problems are displayed in Table 2.4.10. The sample problems used QR 1/1, a P_0 scattering expansion, a k tolerance of 0.001 and coarse meshes for speed. The higher fidelity runs used finer spatial meshes, default parameters of QR 2/2, P_0 scattering expansion, and the default *k* tolerance (10:sup:-*5*). Results for the longer-time CSAS6 and higher fidelity Denovo calculations are shown in Fig. 2.4.10.

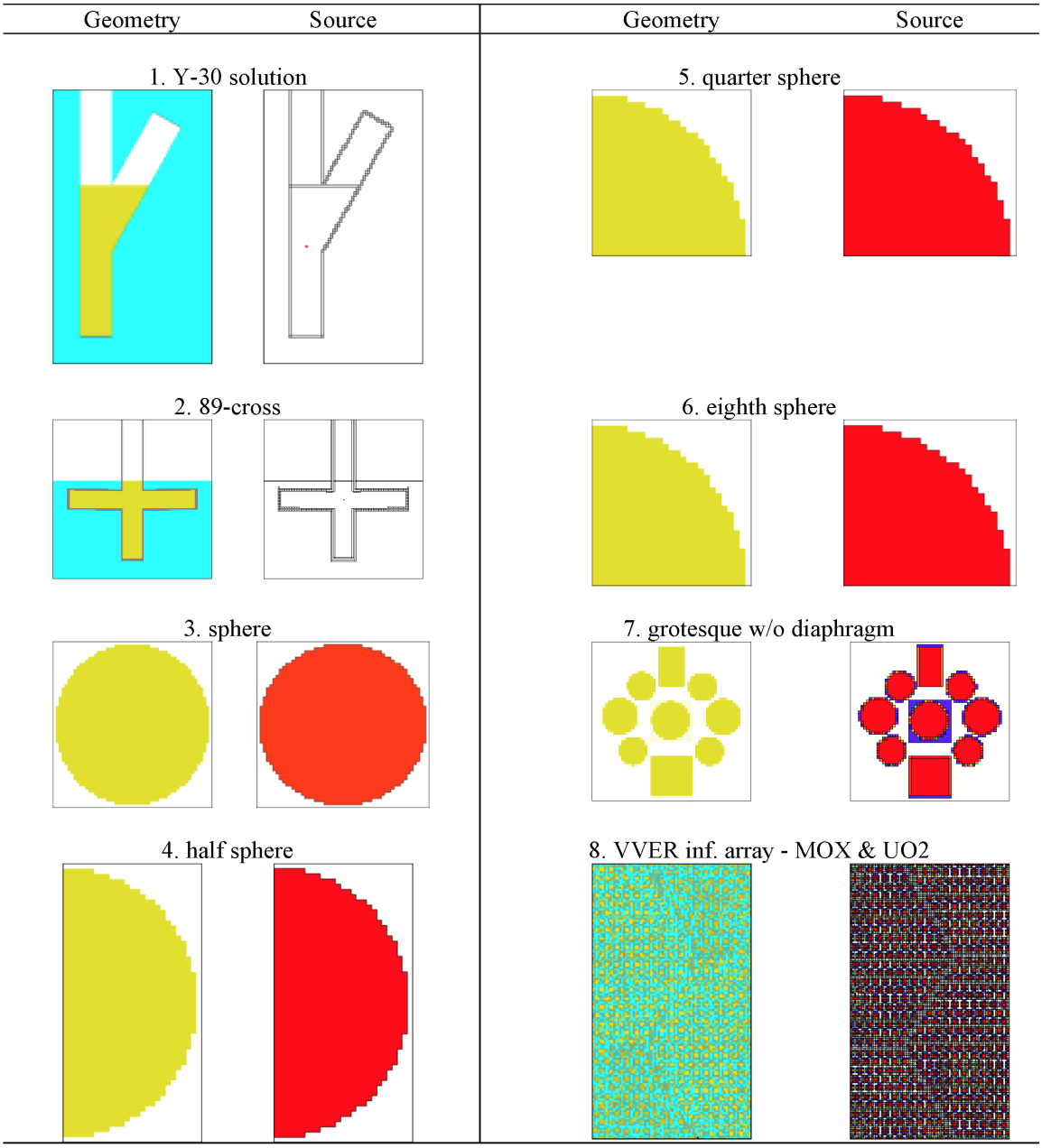


Fig. 2.4.9: Denovo geometry (left) and starting source distribution (right) for the sample problems.

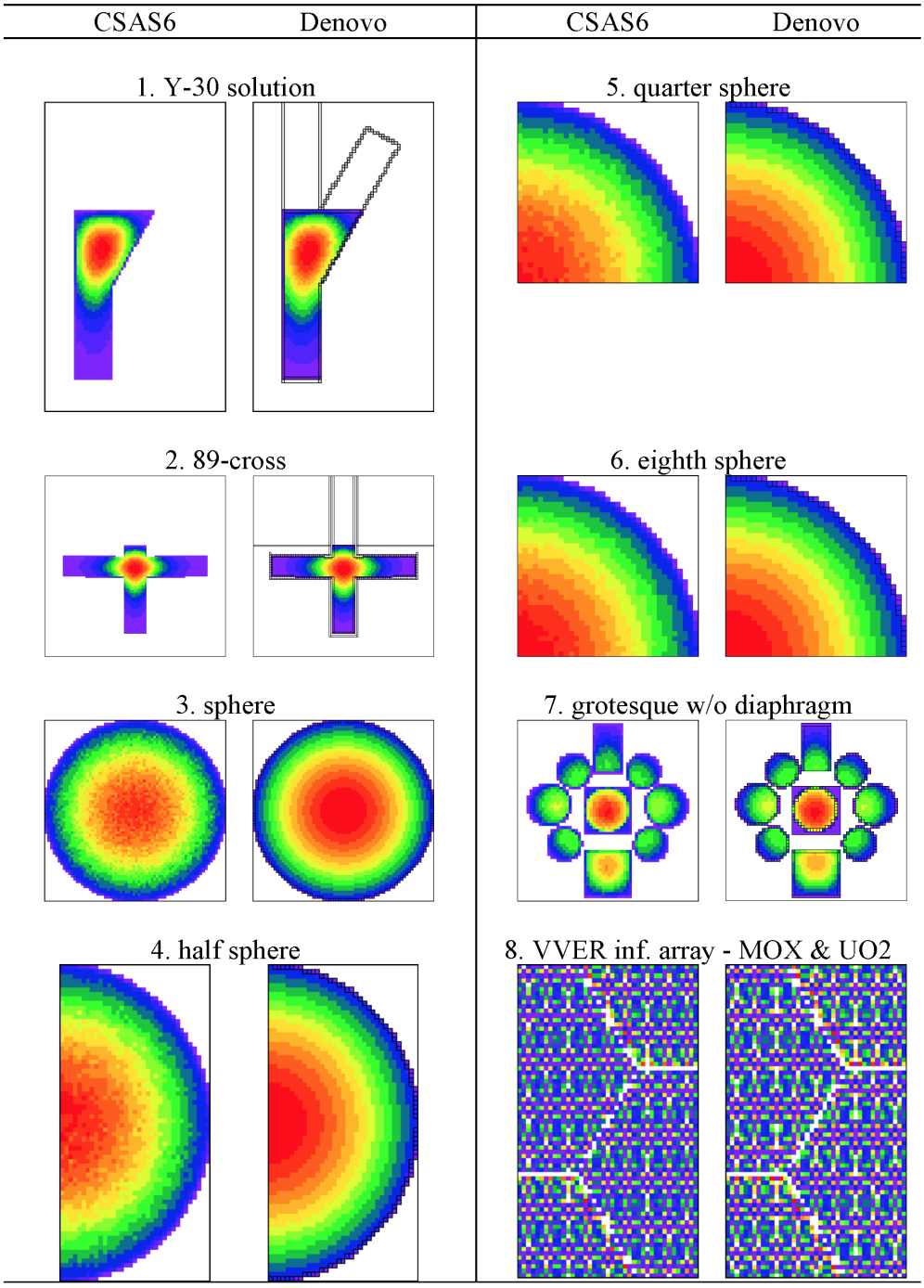


Fig. 2.4.10: Fission source distributions computed by CSAS6 (left) and Denovo (right).

Table 2.4.10: Sample problem results

	CSAS6						Denovo using DEVC			
	Short time			Long time			Sample problem		High Fidelity	
	k_{eff}	uncert	min	k_{eff}	uncert	min	k_{eff}	min	k_{eff}	min
1	1.0048	0.0023	1.94	1.00362	0.00018	380.18	0.889	10.97	1.00220	334.13
2	1.0048	0.0016	1.80	1.00421	0.00017	377.52	0.859	6.96	0.99816	245.34
3	0.9997	0.0017	0.45	0.99831	0.00018	178.69	0.982	3.30	0.99412	75.00
4	1.0013	0.0014	0.53	0.99841	0.00018	106.02	0.982	2.01	0.99412	30.91
5	0.9987	0.0019	0.46	0.99869	0.00020	63.18	0.982	1.40	0.99413	16.78
6	0.9998	0.0019	0.53	0.99830	0.00019	42.03	0.982	1.21	0.99412	8.98
7	0.9955	0.0017	1.43	0.99776	0.00019	101.56	0.882	8.98	0.99988	38.19
8	1.2668	0.0013	2.22	1.26784	0.00013	279.69	1.198	5.70	1.22295	204.08

2.5 KMART5 AND KMART6: POSTPROCESSORS FOR KENO V.A AND KENO-VI

K. B. Bekar and *L. M. Petrie* [1]

KMART5 and KMART6 (**K**eno **M**odule for **A**ctivity-**R**eaction Rate **T**abulation) are modules whose primary purpose is to postprocess a KENO V.a or KENO-VI restart file with the corresponding working cross-section library to generate nuclide activity tables. It also allows collapsing and printing fluxes calculated by KENO. The KENO problem must have a mixing table, must calculate the fluxes, and must write a restart file containing the calculated data. KMART calculations are generally imbedded within a CSAS5 or CSAS6 input file immediately following the CSAS input as a stacked input case.

[1] Formerly with Oak Ridge National Laboratory

2.5.1 INTRODUCTION

KMART5 and KMART6 (**K**eno **M**odule for **A**ctivity-**R**eaction Rate **T**abulation) are modules whose primary purpose is to postprocess a KENO V.a or KENO-VI restart file with the corresponding working cross-section library to generate nuclide activity tables. It also allows collapsing and printing fluxes calculated by KENO. The KENO problem must have a mixing table, must calculate the fluxes, and must write a restart file containing the calculated data.

2.5.2 KMART INPUT DATA

Input data for KMART is read into the program using free form blocked input similar to KENO. The data blocks are started with a READ *BLOCK NAME* and ended with an END *BLOCK NAME*. There are three data blocks that KMART can read. The first data block is named INITIAL, and the input starts with the keywords READ INITIAL. There are ten possible keyworded entries in this block that may be entered in any order.

Keyword	Variable	Description
PRTVOLS	PRINT_VOLUMES	A flag to cause the volumes calculated by KENO to be printed by KMART. Off by default, turn on by entering.
KUNIT=	KUNIT	The logical unit number of the KENO restart file. The default is 35
FNI=	RESTART	*Mode_in* extra field in the input restart file name [restart_*mode_in*.keno_input] and [restart_*mode_in*.keno_calculated]. The default is an empty field.
XUNIT=	XUNIT	The logical unit number of the cross-section library. The default is 4
ACTBY-GRP	ACTIVITIES_BY_GROUP	A flag which turns on printing activities by group. If the fluxes are collapsed, the activities will be by broad group, otherwise they will be by fine group. Off by default, turn on by entering
RRPVOL	REACTION_RATES_PER_UNIT_VOLUME	A flag causing the activities to be printed per unit volume rather than integrated over the volume of the region. Off by default, turn on by entering
KENO3D	NK3D K3DFILE	Unit number on which to write data for plotting with KENO3D. File name of the KENO input file (minus the trailing extension). The plot data file will be named K3DFILE.kmt.
NOPRINT	PRINT_RESULTS	A flag allowing suppressing printing results.
FLUXBIN	FLUX_BIN	A flag to turn on generating a collapsed cross section file for TRITON.
WUNIT	WGTD	The logical unit number on which to write an AMPX weighted library of the 1-D neutron cross sections.

A sample data block is given below.

```
READ INITIAL KUNIT=35 XUNIT=4 END INITIAL
```

One of the next two blocks is required, but both can be specified if desired. If both are entered, either one can be first. The next data block specified is named ACTIVITY, and the input starts with the keywords READ ACTIVITY. It contains the data specifying which activities are to be calculated. The activities are specified by pairs of numbers giving the nuclide identifier and the reaction type identifier desired. A list of reaction types, also known as MT numbers, can be found in Appendix A of the XSLib chapter. These pairs are repeated until all the desired activities have been specified. If the nuclides are identified by the SCALE scheme, then the nuclide can be specified most explicitly by following the nuclide by the keyword MIX= and the mixture number the nuclide is in. By specifying a mixture of zero the activity will be calculated for each region in which the nuclide occurs. If the nuclide specifies a natural element identifier (1000*Z) and individual isotopes occur on the cross-section library, the isotope activities will be summed to produce the total activity for the element. If the nuclide is a special nuclide, i.e., identified with a prefix id times a million + a ZA, then MIX= must be specified as a mixture the nuclide occurs in. The data pair is described below.

Keyword	Variable	Description
	NUCLIDE	The nuclide identification number on the cross-section library for this activity request.
MIX	MIXTURE	Mixture number of the nuclide for this activity request. This is an optional entry, and defaults to 0.
	REACTION	The reaction type identifier for this activity request.

If no activities are desired, then the block can be omitted. A sample block is given below.

```
READ ACTIVITY 92235 18 92235 27 92235 1452 END ACTIVITY
```

The other input block is named COLLAPSE, and starts with the keywords READ COLLAPSE. There are two keyword entries that may be input in this block. A flux factor to normalize the fluxes by can be specified. It defaults to 1. The last fine group in the current broad group is the other entry. The broad groups are specified sequentially starting with group one. If the flux factor is specified more than once, the last value given is used. The data is specified as below.

Keyword	Variable	Description
FACTOR	FACTOR	A flux multiplier used to scale the fluxes before printing (default 1.0).
LASTG=	LAST_GROUP	The last fine group to be included in the current broad group. The broad groups are input sequentially starting with group one.

If no collapsed fluxes are desired, then the block can be omitted. A sample block is given below.

```
READ COLLAPSE FACTOR 1.0 LASTG=10 LASTG=20 LASTG=30 LASTG=56
 END COLLAPSE
```

2.5.3 KMART SAMPLE INPUT

Sample input data for KMART5 is given in Example 2.5.1.

Example 2.5.1: KMART5 sample input.

```
=kmart5
read initial
  kunit=35 xunit=4 actbygrp rrpvol  keno3d  40  kmart5
end initial
read collapse
  lastg=39  lastg=214  lastg=252
end collapse
read activity
  92234 18
  92234 27
  92235 18
  92235 27
  92236 18
  92236 27
  92238 18
  92238 27
   1001 27
```

(continues on next page)

(continued from previous page)

```
   8016 27
   9019 27
end activity
end
```

2.6 C5TOC6: INPUT FILE CONVERSION PROGRAMS FOR CSAS

K. Bekar

2.6.1 INTRODUCTION

Program C5TOC6 can be used to automatically convert a CSAS5 input file to a CSAS6 input file. This functionality can be useful when converting a KENO V.a validation sequence to a KENO-VI validation sequence. It removes the problem of introducing a mistake or inadvertently changing the data when remodeling a geometry to the KENO-VI format. For some cases, however, the converted model may be a very inefficient KENO-VI model.

2.6.2 DESCRIPTION AND INPUT GUIDE

C5TOC6 is a utility program that can be used to automatically convert a CSAS5 input file to a CSAS6 input file. For C5TOC6, the "=CSAS*BB*" record in the input stream (where the *BB* is 5, 25, or 2x) is replaced by "=C5TOC6 PARM=CSAS5." The output file is named _geom*nnnnnnn* where *nnnnnnn* is a unique 7-digit number.

Important: The conversion makes no attempt to optimize the output file, so it almost surely will be inefficient in its use of storage, and in its use of bodies. This can lead to models that are very inefficient in their running time also.

The input/output (I/O) units for C5TOC6 are given below.

Unit Number	Function
5	CSAS5 input file
6	Output
7	Input file generated for CSAS6)

A sample C5TOC6 sample input file and corresponding converted CSAS6 input file are shown in Example 2.6.1 and Example 2.6.2, respectively.

Example 2.6.1: Sample C5TOC6 problem.

```
=c5toc6    parm=csas25
sample problem 1  case 2c8 bare
v7.1-252
read composition
   uranium  1 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end composition
read parameters
   flx=yes fdn=yes far=yes htm=no
end parameters
read geometry
```

(continues on next page)

(continued from previous page)

```
   unit 1
     cylinder 1 1 5.748 5.3825 -5.3825
     cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
end geometry
read array
   nux=2 nuy=2 nuz=2  fill f1 end fill
end array
end data
end
```

Example 2.6.2: Sample converted CSAS6 output file.

```
=csas26
sample problem 1  case 2c8 bare
v7-252
read composition
     uranium  1 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  end composition
read parameters
flx=yes fdn=yes far=yes htm=no
end parameters
read geometry
unit          1
cylinder   1
              5.748000000E+00  5.382500000E+00 -5.382500000E+00
            origin   x=  0.000000000E+00   y=  0.000000000E+00
media     1    1       1
                   vol=  8.938968621E+03
cuboid     2
              6.870000000E+00 -6.870000000E+00  6.870000000E+00
             -6.870000000E+00  6.505000000E+00 -6.505000000E+00
media     0    1       2  -1
                   vol=  1.071004479E+04
boundary     2
global unit         2
cuboid 1
              2.748000000E+01  0.000000000E+00  2.748000000E+01
              0.000000000E+00  2.602000000E+01  0.000000000E+00
array      1  1
               place 1 1 1   6.870000000E+00  6.870000000E+00  6.505000000E+00
boundary 1
end geometry
read array
nux=2 nuy=2 nuz=2  fill f1 end fill
end array
end data
end
```

3. REACTOR PHYSICS

Introduction by M. A. Jessee and F. Bostelmann

SCALE supports a wide range of reactor physics analysis capabilities. SCALE reactor physics calculations couple neutron transport calculations with ORIGEN to simulate the time-dependent transmutation of various materials of interest. The two reactor physics analysis tools within SCALE are TRITON and Polaris. TRITON is SCALE's modular reactor physics sequence for a wide variety of system types; Polaris is SCALE's light water reactor (LWR) lattice physics sequence.

The primary function of TRITON is to simulate the time-dependent evolution of nuclide inventories of a reactor system through a series of multigroup transport calculations and depletion/decay calculations. Before each multigroup transport calculation, TRITON executes the XSProc module to calculate multigroup cross sections for each user-designated depletion material. The XSProc calculation is performed based on the time-dependent material compositions and the user-defined cross section processing definitions. TRITON provides maximum modeling flexibility, supporting the full range of cross section processing options in XSProc along with support for four different multigroup transport modules available in SCALE. These transport modules include the following:

- XSDRN: one-dimensional discrete ordinates (S_N) transport module for modeling simple slab, cylindrical, and spherical geometries,
- NEWT: two-dimensional (2D) S_N polygon-mesh deterministic transport module with arbitrary geometry model definitions, and
- KENO-V.a and KENO-VI and Shift: three-dimensional Monte Carlo transport modules with arbitrary geometry model definitions.

In addition to the multigroup-based calculation sequences, TRITON supports continuous-energy (CE) transport calculations with KENO-V.a and KENO-VI and Shift. For each depletion material, the CE Monte Carlo calculation tallies energy-integrated nuclide-dependent reaction rates to couple the transport solution to the ORIGEN depletion calculation. Both the multigroup- and CE-based depletion calculations are parallelizable and can run on an arbitrary number of processors.

TRITON provides easy-to-use input options to define the time-dependent reactor condition, including power history, material temperatures, and material compositions. TRITON also provides the option to perform lattice physics calculations, with input options to define branch calculations, homogenization edits, few-group energy structures, and assembly discontinuity factors. The homogenized few-group cross sections are archived onto auxiliary data files for subsequent in-reactor core calculations.

TRITON generates several data files for follow-on SCALE analysis. It creates the ORIGEN binary concentration file (.f71 extension) that stores all of the material inventories at each depletion/decay step. This file can be used as input to ORIGEN, ORIGAMI, and MAVRIC to support spent fuel characterization and shielding analysis. TRITON also creates the ORIGEN binary library file (.f33 extension) that stores the ORIGEN transition matrix for each depletion material at each depletion/decay step. The ORIGEN library files can be used as input to ORIGEN or ORIGAMI for rapid assessment of spent fuel inventories. Finally, TRITON also generates the aforementioned lattice physics few-group cross section archive (xfile016) for reactor core calculations.

In addition to the TRITON reactor physics sequence, SCALE supports an alternative easy-to-use LWR depletion sequence for generating lattice physics data for full-core reactor calculations. The Polaris lattice

physics sequence couples 2D multigroup transport calculations with ORIGEN depletion to simulate LWR fuel assemblies. Polaris provides easy-to-use input definitions for defining the primary inputs necessary for lattice physics analysis, namely the pin and lattice geometry, the material compositions, and specifications for the power history and branch calculations. Polaris supports systematic input defaults for analysis of both fuel and reflector models for both pressurized water reactor (PWR) and boiling water reactor (BWR) geometries. It uses a novel approach for cross section processing called the Embedded Self Shielding Method (ESSM). The hallmark feature of ESSM is that the self-shielding calculation is performed on the 2D fuel assembly geometry, eliminating the need for user-defined cross section processing definitions. The ESSM calculation and the keff calculation are performed with a deterministic transport module based on the Method of Characteristics (MOC) approach. The Polaris sequence generates the lattice physics archive file (.t16 and .x16 extension), the ORIGEN binary concentration file (.f71 extension), the ORIGEN binary library file (.f33 extension), and a geometry plot file (.png extension).

3.1 TRITON: A MULTIPURPOSE TRANSPORT, DEPLETION, AND SENSITIVITY AND UNCERTAINTY ANALYSIS MODULE

F. Bostelmann, M. A. Jessee, D. Wiarda, K. T. Clarno, U. Mertyurek, K. Bekar

ABSTRACT

The TRITON computer code is a multipurpose SCALE control sequence for transport, depletion, and sensitivity and uncertainty analysis. TRITON automates the processing of cross sections, the neutron transport calculations for one-, two-, and three-dimensional (1D, 2D, and 3D) configurations, and the depletion calculations to estimate the neutron flux, mixture-wise powers, isotopic concentrations, source terms, decay heat and other quantities as well as few-group homogenized cross sections for nodal core calculations as a function of burnup.

TRITON can be used in combination with any one of SCALE's neutron transport kernels. Deterministic multigroup transport calculations for 1D and 2D geometries are performed using XSDRNPM and NEWT, respectively. The application of the Monte Carlo codes KENO V.a, KENO-VI, and Shift enables depletion calculations of 3D geometries in either multigroup or in continuous-energy mode. In MG mode, TRITON automates the preparation of problem-dependent MG cross sections for use in MG neutron transport calculations using SCALE's cross section processing module XSProc. The depletion calculations are performed by the ORIGEN depletion module.

The SAMS module is used to determine the sensitivity of the calculated value of responses to the nuclear data used in the calculation as a function of nuclide, reaction type, and energy. The uncertainty in the calculated value of the response, resulting from uncertainties in the basic nuclear data used in the calculation, is estimated using energy-dependent cross section covariance matrices. The implicit effects of the cross section processing calculations are also treated.

3.1.1 INTRODUCTION

TRITON (Transport Rigor Implemented with Time-dependent Operation for Neutronic depletion) is a multipurpose SCALE control sequence for transport and depletion analysis for reactor physics applications. By calling the appropriate SCALE modules, TRITON automates the processing of cross sections, the neutron transport calculations for one-, two-, and three-dimensional (1D, 2D, and 3D) configurations, and the depletion calculations to estimate the neutron flux, mixture-wise powers, isotopic concentrations, source terms, decay heat and other quantities as a function of burnup. An overview can be found in [TRITONDB11].

The choice of the neutron transport kernel determines whether TRITON is run in multi-group (MG) or in continuous-energy (CE) mode. TRITON can be used in combination with any one of SCALE's neutron

transport kernels. Deterministic MG transport calculations for 1D and 2D geometries are performed using XSDRNPM and NEWT, respectively. The application of the Monte Carlo codes KENO V.a, KENO-VI, and Shift enables depletion calculations of 3D geometries in either MG or in CE mode. In MG mode, TRITON automates the preparation of problem-dependent MG cross sections for use by the MG neutron transport kernels (see Fig. 3.1.1). Nodal data for use in nodal core simulations can be generated with the TRITON sequence that uses the NEWT deterministic transport code and with the TRITON sequences using the Shift Monte Carlo code.

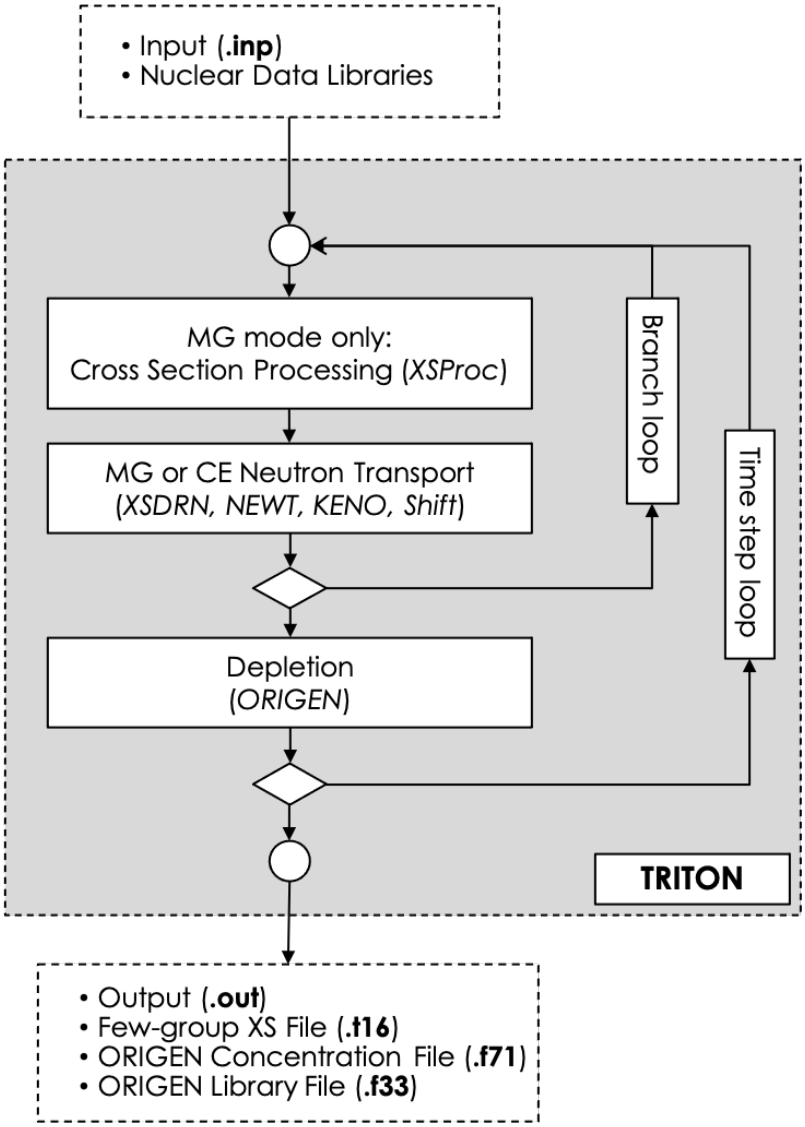


Fig. 3.1.1: General flowchart of the TRITON reactor physics sequence.

The SAMS module is used to determine the sensitivity of the calculated value of the response to the nuclear data used in the calculation as a function of nuclide, reaction type, and energy. The uncertainty in the calculated value of the response, resulting from uncertainties in the basic nuclear data used in the calculation, is estimated using energy-dependent cross section covariance matrices. The implicit effects of the cross section processing calculations are predicted using SENLIB and BONAMIST.

As a SCALE control module, TRITON automates execution of SCALE functional modules and manages data

transfer and input/output processes for multiple analysis sequences. Each of TRITON's eleven calculational sequences is provided in Table 3.1.1, which lists the sequence name keyword, the sequence description, and the function modules invoked within each sequence. The method for cross section processing is selected using a separate "*parm*=" keyword, which is described in more detail in the next section.

Table 3.1.1: Overview of TRITON sequences.

Sequence keyword	**Primary SCALE modules**	**parm= options**	**Sequence function**
Cross section processing sequences			
=T-XSEC	XSProc	bonami centrm[a] xslevel=1/2/3/4	Preparation of multigroup (MG) cross section library.
Transport sequences			
=T-XSDRN	XSProc, XSDRNPM	bonami centrm[a] xslevel=1/2/3/4 weight[b]	1D MG deterministic transport calculation.
=T-NEWT	XSProc, NEWT		2D MG deterministic transport calculation.
Depletion sequences			
=T-DEPL-1D	XSProc, XSDRNPM, ORIGEN, OPUS	bonami centrm xslevel=1/2/3*a*/4 addnux=0/1/2[a]/3/4 weight[b]	1D MG deterministic transport, coupled with ORIGEN depletion.
=T-DEPL	XSProc, NEWT, ORIGEN, OPUS		2D MG deterministic transport, coupled with ORIGEN depletion.
=T5-DEPL	XSProc[c] KENO-V.a, ORIGEN, OPUS		3D, Monte Carlo transport (KENO-V.a), coupled with ORIGEN depletion.
=T6-DEPL	XSProc[c] KENOVI, ORIGEN, OPUS		3D, Monte Carlo transport (KENO-VI), coupled with ORIGEN depletion.
=T5-DEPL-SHIFT	XSProc[c] Shift, ORIGEN, OPUS		3D, Monte Carlo transport (Shift, coupled with ORIGEN depletion.
=T6-DEPL-SHIFT	XSProc[c] Shift, ORIGEN, OPUS		3D, Monte Carlo transport (Shift), coupled with ORIGEN depletion.

[a]Default parm value.
[b]parm=weight is used to generate a broad group cross section library. This parm option is only available for the T-DEPL sequence.
[c]T5-DEPL and T6-DEPL are also available in CE-mode, which does not invoke XSProc for cross section processing.

3.1.2 OVERVIEW OF TRITON SEQUENCES

The TRITON control module supports eleven calculational sequences, each with its own design and applications. Each of these sequences is described in the following subsections.

The first subsection covers the basic cross section processing sequence T-XSEC. The T-XSEC sequence prepares problem-dependent multigroup cross sections for subsequent transport analysis. The second subsection covers TRITON's transport analysis sequences, while the third subsection discusses TRITON's depletion analysis sequences.

3.1.2.1 Cross section processing sequence (T-XSEC)

The T-XSEC sequence provides the ability to prepare a problem-dependent multigroup cross section library using SCALE cross section processing modules to appropriately account for spatial and energy self-shielding effects. The problem-dependent cross section library contains microscopic cross sections for each nuclide for each material composition defined in the TRITON input. SCALE provides several unit cell types (e.g., a lattice of pins, an infinite medium, a multiregion problem, or a doubly heterogeneous cell) to correct the cross sections for spatial and energy self-shielding. Multiple cell calculations can be used in the same calculation. The calculation of multigroup cross sections is executed by XSProc).

3.1.2.2 Transport sequences (T-XSDRN, T-NEWT)

The TRITON transport sequences build upon the cross section processing sequence by automating a transport calculation after cross section processing. Both 1D and 2D discrete-ordinates transport calculations can be performed using XSDRNPM and NEWT, respectively. The T-XSDRN sequence calls XSDRNPM for transport analysis in slab, sphere, or cylindrical geometries, while the T-NEWT sequence calls NEWT for analyses in 2D *xy*-geometries. In addition to the input necessary for cross section processing, an XSDRN or NEWT input model is also required. The XSDRN model input is discussed in Appendix A of TRITON; the NEWT model input requirements are described in the NEWT chapter. Similar capabilities and applications for KENO-V.a and KENO-VI are handled through the CSAS5 and CSAS6 sequences, respectively.

3.1.2.3 Depletion sequences (T-DEPL, T-DEPL-1D, T5-DEPL, T6-DEPL, T5-DEPL-SHIFT, T6-DEPL-SHIFT)

The TRITON depletion sequences build upon the transport sequences by automating depletion/decay calculations after the transport calculations for each material designated for depletion. One or more materials in the model can be designated for depletion. Each designated material is depleted using region-averaged reaction rates, accounting for all regions in the model associated with a given depletion material. The TRITON depletion calculation procedure is described further in the next subsection. TRITON automates the various computational processes-cross section processing, transport, and depletion-over a series of depletion and decay intervals supplied by the user. The depletion procedure is discussed in Sect. 3.1.2.3.1. The 2D TRITON depletion sequence (T-DEPL), which uses NEWT for the transport calculations and the 3D TRITON depletion sequences which use Shift in CE mode for the transport calculations (T5-DEPL-Shift, T6-DEPL-SHIFT) provide the capability to generate lattice-physics data for nodal core calculations.

Within TRITON depletion calculations, TRITON invokes the ORIGEN depletion module for the time-dependent transmutation of each user-defined material. TRITON provides ORIGEN the neutron flux space-energy distribution, the multigroup cross sections, material concentrations, and material volumes. ORIGEN performs the flux normalization, cross section collapse, and multi-material depletion/decay operations to determine new isotopic concentrations for the next calculation.

Predictor-corrector depletion process

For all depletion sequences, TRITON automates cross section processing, transport, and depletion calculations over a series of depletion-decay intervals supplied by the user. A **depletion interval** represents a time interval in which the model power level is assumed constant. A depletion model that exhibits various power level changes will require multiple depletion intervals to accurately model the changes in power. Each depletion interval can be followed by a decay calculation over a user-specified **decay interval**.

Within a given depletion interval (e.g., an LWR operating at constant power for a 12-month fuel cycle), the isotope concentrations of different depletion materials change, which induces changes in the problem-dependent multigroup cross sections (through spatial and energy self-shielding effects) as well as the neutron flux distribution, leading to different power distributions and transmutation rates in depletion materials. This requires TRITON to represent each depletion interval as a series of smaller time intervals in which cross section processing and transport solutions are recomputed to accurately model these time-dependent effects. A **depletion subinterval** represents a time interval in which TRITON performs cross section processing and transport calculations to determine cross sections and flux distributions used in the depletion calculations. All depletion subintervals for a given depletion interval have the same length-for example, one 12-month depletion interval can be represented as a series of 12 one-month depletion subintervals, or as 6 two-month depletion subintervals. Alternatively, the 12-month depletion interval can be modeled as two consecutive 6-month depletion intervals, each one having a different number of subintervals. Therefore the formulation of a **depletion scheme** in TRITON is highly flexible. A depletion scheme is the set of user-defined depletion and decay intervals with associated power levels and number of subintervals.

Caution: TRITON does not provide automated means to determine the appropriate depletion scheme for a given application. The user must determine the accurate depletion scheme specific to his or her application.

TRITON uses a predictor-corrector approach to process the user-defined depletion scheme. The predictor-corrector approach performs cross section processing and transport calculations based on anticipated isotope concentrations at the *midpoint* of a depletion subinterval. Depletion calculations are then performed over the full subinterval using cross sections and flux distributions predicted at the midpoint. Depletion calculations are then extended to the midpoint of the next subinterval (possibly through a decay interval and into a new depletion interval), followed by cross section processing and transport calculations at the new midpoint. The iterative process is repeated until all depletion subintervals are processed. In order to start the calculation, a "bootstrap case" is required using initial isotope concentrations for the initial cross section processing and transport calculation. The bootstrap calculation is used to determine the anticipated isotope concentrations at the midpoint of the first depletion subinterval.

The predictor-corrector approach is best explained by an example. Fig. 3.1.2 illustrates the predictor-corrector process for a hypothetical depletion scheme with two depletion intervals. The first depletion interval contains two subintervals, followed by a decay interval. The second depletion interval contains one subinterval and is also followed by a decay interval. In Fig. 3.1.2, cross section processing and transport calculations are represented by the 'T' label, and depletion calculations are represented by the 'D' label. For this example, four sets of calculations would be necessary: one for each of the three depletion subintervals, and one for the initial "bootstrap case." These calculations are represented in the following eight steps.

- **Step 1**
  - T_0: Cross section processing and transport calculation using initial (i.e., time-zero) isotope concentrations.

- **Step 2**
  - D_1: Depletion calculation from time-zero to the midpoint of the first depletion subinterval. The dashed horizontal arrow denotes a "predictor" depletion step.
- **Step 3**
  - T_1: Cross section processing and transport calculation at the midpoint of the first depletion subinterval.
- **Step 4**
  - D_1: Depletion calculation for the first depletion subinterval. The solid horizontal arrow across the subinterval denotes a "corrector" depletion step. *Corrector steps use cross sections and flux distribution computed at the subinterval midpoint.* This is represented by a solid arrow from T_1 to D_1.
  - D_2: Predictor depletion calculation for the second depletion subinterval. *Predictor steps use cross sections and flux distribution computed at the***previous***subinterval midpoint.* This is represented as the dashed arrow from T_1 to D_2.
- **Step 5**
  - T_2: Cross section processing and transport calculation at the midpoint of the second depletion subinterval.
- **Step 6:**
  - D_2: Corrector depletion calculation for the second depletion subinterval, followed by the decay calculation at the end of the first depletion interval.
  - D_3: Predictor depletion calculation for the third depletion subinterval. The third depletion subinterval is the first and only subinterval associated with the second depletion interval.
- **Step 7**
  - T_3: Cross section processing and transport calculation at the midpoint of the third depletion subinterval.
- **Step 8**
  - D_3: Corrector depletion calculation for the third depletion subinterval. This calculation is followed by a second decay calculation.

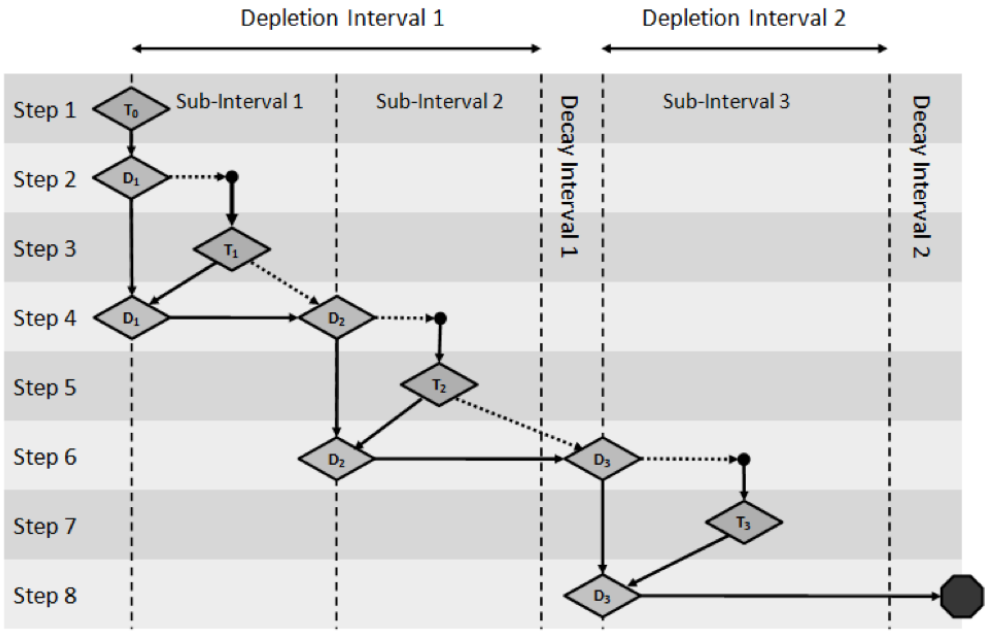


Fig. 3.1.2: Predictor/corrector depletion algorithm used by TRITON.

The depletion calculations are performed by ORIGEN and span either the first half of a subinterval (predictor step) or the full subinterval (corrector step). ORIGEN performs these depletion calculations and possible decay calculations over a series of smaller time intervals. The **ORIGEN time intervals** are automatically determined by TRITON depending on the length of the depletion subinterval and decay interval. Additionally, TRITON will automatically adjust the number of subintervals per depletion interval if the time length of the user-defined subinterval is large (i.e., >400 days). TRITON writes the utilized depletion scheme near the top of the output file. The depletion scheme output edit is further described in Sect. 3.1.5.4.1.

Lattice physics analysis

The 2D depletion sequence (T-DEPL) may be used to generate lattice physics data for subsequent core analysis calculations using core simulator software. Core simulators typically employ few-group nodal diffusion theory for neutronic calculations, coupled with other calculation methods for thermal hydraulics, fuel performance, and plant operation (e.g., soluble boron letdown or control rod movement). Core simulation requires the use of pretabulated **lattice physics data** for the neutronic calculations-that is, few-group homogenized cross sections, with appropriate discontinuity factors, pin powers, and kinetic parameters, functionalized in terms of burnup and other system conditions such as fuel temperature and moderator density.

To support lattice physics database preparation, the NEWT transport module contains flexible input options to define the few-group energy structure, spatial homogenization regions, and discontinuity factors. After the transport calculation at the midpoint of each depletion subinterval, NEWT computes the lattice physics data and stores this data on a temporary file. TRITON reads the temporary file and archives the lattice physics data onto a separate database file. In addition, the *T-DEPL* sequence supports branch calculations in which perturbations may be applied to certain system conditions such as fuel temperatures and moderator density. TRITON automates the cross section processing and transport calculations for each branch condition at the midpoint of the depletion subinterval. NEWT computes the lattice physics data for the branch calculations, and TRITON archives this data onto the lattice physics database file.

The TRITON input options for branch calculations are described in Sect. 3.1.3.3.2, and the file format of the lattice physics database is provided in the Appendix B of TRITON.

Note: The TRITON input options for branch calculations are designed to be highly flexible to support a large range of core analyses; therefore, TRITON does not provide automated means to determine the branch calculations. The user must determine the necessary branch calculations for his or her core analysis and be knowledgeable of the capabilities and limitations of the cross section treatment of the core simulator. The TRITON Lattice Physics Primer has been developed to provide guidance on appropriate TRITON branch calculations for LWR core analysis (NUREG/CR-7041) and in "Cross Section Generation Guidelines for TRACE-PARCS" (NUREG/CR-7164).

3.1.3 INPUT DESCRIPTION

TRITON input is free-form and keyword based, similar in form to many other modules in SCALE. With a few exceptions, the following formatting rules apply:

- Data is limited to 255 columns but may wrap into as many lines as are needed.
- Comment lines start with a tick mark (') in the first column of a line and may be placed anywhere in the input.
- The keyword-based input is case insensitive.
- TRITON input is organized into blocks of data. Each data block begins with *read blockname* and terminates with *end blockname*.
- Blocks of data may appear in any order. Each block of data may appear only once in the input.
- Input can be redirected from an auxiliary file by using the open angle bracket (<) and the name of the file-for example, *</path/to/auxiliary_input_file*.

The first three lines of input and the last line of the input are unique. The first line of input contains the TRITON sequence name along with parameter specifications, e.g., *parm=centrm*. The second line contains the case title (up to 80 characters), and the third line contains the cross section library identifier. The last line of the input contains the end keyword and terminates the input file. An example TRITON input is as follows:

```
=t-xsec         parm=(centrm,check)
TRITON Input Example
V7-252
...
end
```

In this example, the first line of input declares this calculation to use the T-XSEC sequence. The name of the sequence is preceded by the "=" sign. After the sequence name, two parameter options are specified. Parameters are optional. If specified, the keyword *parm=* must precede the parameter options. Multiple parameter options can be provided in a comma-separated list enclosed in parentheses. In this example, the *centrm* option specifies the CENTRM-based discrete-ordinates sequence is used by default. The *check* option implies that TRITON will read all input and ensure that no input errors are present, without running additional calculations. The second input line provides the case title: *TRITON Input Example*. The third input line provides the cross section library: *V7-252*. This example input file is terminated at the *end* keyword. The *end* keyword must appear by itself at the beginning of the final line of the input file.

The TRITON input section is organized by sequences. The first section summarizes the input requirements for the cross section processing sequence T-XSEC, which includes discussion of the *COMPOSITION* and

CELLDATA block. The second section summarizes the input requirements for the TRITON transport sequences T-XSDRN and T-NEWT. The *XSDRN MODEL* block is described in Appendix B of TRITON. The third section summarizes the input for TRITON depletion sequences: T-DEPL-1D, T-DEPL, T5-DEPL, and T6-DEPL. The depletion sequence input section includes discussion of the *DEPLETION, BURNDATA, TIMETABLE, BRANCH,* and *OPUS* blocks.

The input requirements for the depletion sequences and the S/U sequences build upon the input requirements for the cross section processing sequence and the transport sequences, so the user should be familiar with these first two sections. However, the input requirements for the depletion and S/U sequences are independent, so the user can skip over these sections as needed.

The fifth and sixth section of the input description is dedicated to two TRITON-specific blocks of data to simplify model development and output control: the *ALIAS* block and the *KEEP_OUTPUT* block, respectively. The final section describes TRITON control parameters used in the *parm*= specification.

3.1.3.1 Cross section processing

An example input structure for a cross section processing sequence calculation is the following:

```
=t-xsec parm=(options)
title-goes-here
xslib-goes-here
read alias
[List of user-defined aliases (optional)]
end alias
read comp
[List of material specifications (standard SCALE format)]
end comp
read celldata
[Unit cell specifications for self-shielding calculation (optional)]
end celldata
end
```

In this input, the title can be any descriptive title, and the cross section library *x-sect_lib_name* can be any multigroup SCALE cross section library (or continuous-energy library if KENO is used). The three blocks of data highlighted in red-*ALIAS, COMPOSITION,* and *CELLDATA*-**must** appear in the order shown above. However, the *ALIAS* and *CELLDATA* blocks are optional. If the *ALIAS* block is not used, the *COMPOSITION* block follows the cross section library line. If the *CELLDATA* block is not used, the input is terminated after the *COMPOSITION* block.

The input requirements for the *ALIAS* block are deferred to Sect. 3.1.3.4 as the *ALIAS* block impacts many different blocks of data for all TRITON sequences. The *COMPOSITION* block is used to define material compositions and temperatures. The *CELLDATA* block is used to specify unit cell calculations used to generate problem-dependent multigroup cross sections. The input requirements for the *COMPOSITION* and *CELLDATA* blocks are comprehensively described in the XSProc manual and are not repeated here. Fig. 3.1.3 shows an example input for a cross section processing calculation. In this input file, cross section processing calculations are performed for two different square-pitched UO_2 fuel pins surrounded by Zircaloy-4 cladding and borated H_2O moderator. The first fuel pin (material 1) is 2.5% enriched in ^{235}U. The second fuel pin (material 4) is 4.5% enriched in ^{235}U. These materials are used in two separate unit cell definitions in the *CELLDATA* block.

```
=t-xsec
Simple T-XSEC input
v7.1-252n
read composition
'cell 1
  uo2   1 den=9.550 1 750 92235 2.5 92238 97.5 end
  zirc4 2 1 600 end
  h2o   3 den=0.75 1 550 end
  boron 3 den=0.75 600e-6 550 end
'cell 2
  uo2   4 den=9.550 1 750 92235 4.5 92238 95.5 end
  zirc4 5 1 600 end
  h2o   6 den=0.75 1 550 end
  boron 6 den=0.75 600e-6 550 end
end composition
read celldata
   latticecell squarepitch fueld=0.95 1 cladd=1.05 2 pitch=1.4 3 end
   latticecell squarepitch fueld=0.95 4 cladd=1.05 5 pitch=1.4 6 end
end celldata
end
```

Fig. 3.1.3: Example T-XSEC input.

One key observation in this example is the duplicate definitions for the clad material (materials 2 and 5) and the moderator material (materials 3 and 6). For practical use in subsequent transport calculations, only four material compositions need to be defined: one each for the different fuel pin enrichments and one definition each for the clad and moderator material compositions. However, as described in the XSProc manual, the same material identifier cannot be used in multiple unit cell definitions. Because this example requires two separate unit cell definitions to appropriately generate cross sections for each fuel pin enrichment, duplicate definitions are required for the clad and moderator compositions. The unique mixture number input requirement can lead to many duplicate definitions of clad and moderator materials, depending on model complexity. To simplify model development, duplicate material compositions and similar unit cell definitions can be defined simultaneously through the use of aliases. The *ALIAS* block is discussed further in Sect. 3.1.3.4.

Combined two-region and S_N cross section processing

It is possible to use the both the CENTRM-based two-region method and the CENTRM-based S_N method within the same input file. Fig. 3.1.4 shows a modified input file of the previous example in which the first unit cell uses S_N cross section processing and the second unit cell uses two-region cross section processing. Each unit cell contains a *centrmdata* keyword specification after the *latticecell* specification. The *centrmdata* specification contains a set of additional keyword specifications used to identify the S_N and the two-region options in CENTRM.

The input centrmdata npxs=1 end centrmdata instructs TRITON to use S_N cross section processing, whereas the input centrmdata npxs=5 end centrmdata instructs TRITON to use two-region cross section processing. These keyword options are described in detail in the XSProc manual. The default cross section option for TRITON is S_N; therefore, the first *centrmdata* specification is not needed (but still acceptable). If *parm=centrm* was specified, the first *centrmdata* specification would not be needed (but still acceptable), whereas the second *centrmdata* specification would be required to activate the two-region option. Conversely,

if *parm=2region* was specified, the second *centrmdata* specification is not needed (but still acceptable), whereas the first *centrmdata* specification would be required to activate the S_N option.

The *centrmdata* specifications may also be applied to other unit cell types (e.g., multiregion); however, the two-region method is only valid for specific unit cell configurations described in the XSProc manual. The user should determine the applicability of the two-region method by comparing calculation results with continuous-energy calculations or multigroup calculations using the CENTRM-based S_N method.

```
=t-xsec
Simple T-XSEC input
v7.1-252n
read composition
'cell 1
  uo2   1 den=9.550 1 750 92235 2.5 92238 97.5 end
  zirc4 2 1 600 end
  h2o   3 den=0.75 1 550 end
  boron 3 den=0.75 600e-6 550 end
'cell 2
  uo2   4 den=9.550 1 750 92235 4.5 92238 95.5 end
  zirc4 5 1 600 end
  h2o   6 den=0.75 1 550 end
  boron 6 den=0.75 600e-6 550 end
end composition
read celldata
   latticecell squarepitch fueld=0.95 1 cladd=1.05 2 pitch=1.4 3 end
   centrmdata npxs=1 end centrmdata
   latticecell squarepitch fueld=0.95 4 cladd=1.05 5 pitch=1.4 6 end
   centrmdata npxs=5 end centrmdata
end celldata
end
```

Fig. 3.1.4: T-XSEC input with multiple cross section processing options.

User-defined Dancoff factors

Like other SCALE calculations, TRITON uses Dancoff factors as part of its cross section processing calculations. The user can specify Dancoff factors for various materials by using the *centrmdata* specification and the *dan2pitch* keyword. Here is an example.

```
read celldata
   latticecell squarepitch fueld=0.95 1 cladd=1.05 2 pitch=1.4 3 end
   centrmdata dan2pitch=0.51 end centrmdata
   latticecell squarepitch fueld=0.95 4 cladd=1.05 5 pitch=1.4 6 end
   centrmdata dan2pitch=0.65 end centrmdata
end celldata
```

In this example, fuel materials 1 and 4 were assigned a Dancoff factor of 0.51 and 0.65, respectively. These Dancoff factor values can be computed using the SCALE MCDANCOFF sequence. Only one *dan2pitch* keyword is allowed for a given *centrmdata* specification.

3.1.3.2 Transport sequences

An example input structure for a transport sequence calculation is the following:

```
=t-newt (or =t-xsdrn) parm=(options)
title-goes-here
xslib-goes-here
read alias
[List of user-defined aliases (optional)]
end alias
read comp
[List of material specifications (standard SCALE format)]
end comp
read celldata
[Unit cell specifications for self-shielding calculation (optional)]
end celldata
read keep_output
[keep output options (optional)]
end keep_output
read model
[specification of XSDRN or NEWT model]
end model
end
```

The *MODEL* block contains a full transport model input description and is required for both the T-NEWT and T-XSDRN sequences. *The MODEL block must be the last block of data in the input file.* The *MODEL* block provides the physical layout of the configuration for which the transport calculation is to be performed, along with general control parameters. The nature of data embedded within the MODEL block depends on the sequence selected. For the T-NEWT sequence, the *MODEL* block contains a complete NEWT input listing. NEWT input is fully described in the NEWT chapter and is not repeated here. For the T-XSDRN sequence, the *MODEL* block is described in the Appendix B of TRITON. Sample problems for both the T-NEWT and T-XSDRN sequences are provided in Sect. 3.1.6. The optional *KEEP_OUTPUT* block is described in Sect. 3.1.3.4.7.

3.1.3.3 Depletion sequences input

An example input structure for a depletion calculation is provided in the following:

```
=t-depl (or =t-depl-1d or =t5-depl or =t6-depl) parm=(options)
title-goes-here
xslib-goes-here
read alias
[List of user-defined aliases (optional)]
end alias
read comp
[List of material specifications (standard SCALE format)]
end comp
read celldata
[Unit cell specifications for self-shielding calculation (optional)]
end celldata
read keep
[keep output options (optional)]
end keep
read burndata
[information about specific power, depletion/decay time and intervals]
end burndata
read depletion
[material depletion specifications]
end depletion
read branch
[branch calculation specifications (optional, t-depl only)]
end branch
```

(continues on next page)

(continued from previous page)

```
read timetable
[time-dependent parameter specifications (optional)]
end timetable
read opus
[opus specification (optional)]
end opus
read model
[specification of XSDRN, NEWT, KENO-V.a or KENO-VI model]
end model
end
```

The TRITON depletion sequences support the following data blocks: the *DEPLETION, BURNDATA, OPUS, BRANCH, and TIMETABLE* data blocks. These data blocks, along with the *KEEP_OUTPUT* block, may appear only once, in any order, and must follow the *COMPOSITION* and *CELLDATA* blocks and must precede the *MODEL* block. The *DEPLETION* and *BURNDATA* blocks are always required for depletion calculations.

The *MODEL* block contains a full transport model input description and is required for all depletion sequences. For the *T-DEPL* sequence, the *MODEL* block contains a complete NEWT input listing. NEWT input is fully described in NEWT chapter and is not repeated here. For the *T-DEPL-1D* sequence, the *MODEL* block is described in Appendix A of TRITON. For *T5-DEPL* and *T6-DEPL* sequences, the *MODEL* block contains input for KENO V.a and KENO-VI, respectively. The details of KENO V.a and KENO-VI input formats are described in the KENO V.a and KENO-VI chapters and are not repeated here. To use the Monte Carlo code Shift instead of KENO V.a or KENO-VI, only the sequence name has to be changed from *T5-DEPL* to *T5-DEPL-SHIFT*, or from *T6-DEPL-SHIFT* to *T6-DEPL* sequences, respectively. Note that the KENO geometry description ends with an additional *END DATA* before *END MODEL*.

TRITON reads the *MODEL* block at the beginning of the sequence to process the input and save data to appropriate data in memory (or on a restart file for KENO). Reading the *MODEL* block at the beginning of the sequence allows TRITON to check all transport module data and to terminate immediately if errors are found in the model input. When the transport module is eventually invoked by the sequence, TRITON uses the processed data in memory (or reads it from the restart file), allowing for transport iterations (XSDRN, NEWT) or neutron histories (KENO, Shift) to begin immediately, eliminating the need for recalculation of geometry data each time the transport module is invoked.

BURNDATA block

The *BURNDATA* data block allows specification of the **depletion scheme** for the model and is used only by the depletion sequences of TRITON for which this block is required. As described in Sect. 3.1.2.3.1, the depletion scheme consists of a series of **depletion intervals**—time intervals of constant power operation—which may be partitioned into many **depletion subintervals**—intervals over which cross section processing and transport calculations are performed to update cross sections and flux distributions used in the depletion calculation. Moreover, depletion intervals may be optionally followed by a **decay interval**—a time interval for zero-power decay.

The depletion intervals that define the depletion scheme are specified in the *BURNDATA* block in chronological order within the *BURNDATA* block, with the following format.

```
READ burndata
power=P burn=B down=D nlib=N end
power=P burn=B down=D nlib=N end
END burndata
```

where

P = average specific power in the basis material(s), in megawatts per metric tonne of initial heavy metal (MW/MTHM) (typically MW/MTU for uranium-only models);

B = length of depletion interval in days;

D = length of decay interval in days following the depletion interval (optional, default = 0.0);

N = number of depletion subintervals for the depletion interval (optional, default = 1).

The average specific power is provided for the basis material(s). In other words, localized power distributions are uniformly scaled accordingly in the transport solution such that the average power in the basis material(s) matches the power specified in input. By default, the basis consists of all materials in the model, so that local powers are scaled to obtain a problem-wide average power matching the power specified in input. The basis can be set as a single material or set of materials in the *DEPLETION* data block. The *DEPLETION* data block is described in Sect. 3.1.3.3.4.

Each depletion interval specification must be terminated by an end keyword. As many depletion intervals as necessary may be entered to model the depletion scheme. The number of depletion subintervals can be used to refine the temporal discretization to force more cross section processing and transport calculations per depletion interval, as discussed in Sect. 3.1.2.3.1.

An example of a *BURNDATA* block is shown below. The example case contains three depletion intervals, with the first interval at power 26.54 MW/MTHM in the basis materials (the basis is defined in the *DEPLETION* block), for an interval of 121 days. This is followed by a second depletion interval at power 38.01 MW/MTHM for 201.5 days and then 30 days of zero-power operation. In the third depletion interval, the basis materials are depleted at a 31.44 MW/MTHM power level for 386.25 days, followed by 5 years (1826.25 days) of decay. In this model, three, two, and one depletion subintervals are used for the first, second, and third depletion intervals, respectively.

Example 3.1.1: Example *BURNDATA* block input.

```
READ burndata
power=26.54 burn=121.0 nlib=3 end
power=38.01 burn=201.5 down=30 nlib=2 end
power=31.44 burn=386.25 down=1826.25 end
END burndata
```

While at least one depletion interval was required in TRITON input files up to SCALE 6.2, TRITON in SCALE 6.3 permits the specification of a depletion step of 0 days or the omission of the depletion step. A TRITON calculation without a depletion step enables a neutron transport-only calculation as in the CSAS sequence, but with TRITON default settings and with the additional TRITON output files (f71, f33, etc.). A 0 day interval is only permitted in the first and only *BURNDATA* entry. The following two examples are equivalent and cause a neutron transport calculation only at t=0.

```
READ burndata
power=26.54 burn=0 end
END burndata

READ burndata
power=26.54 end
END burndata
```

BRANCH block

The T-DEPL sequence in TRITON supports the ability to perform branch calculations during depletion calculations. Branch calculations are not supported for the 3D depletion sequences, nor are branch calculations supported for problems that require doubly heterogeneous cross section processing. A branch calculation is a recalculation of cross section processing and transport calculations with one or more of a limited set of input parameters modified. These calculations are performed at the same location in the depletion scheme as in the nominal cross section processing and transport calculations-that is, at t = 0 and at the midpoint of the depletion subintervals (see Sect. 3.1.2.3.1 for more details on the TRITON predictor-corrector depletion scheme). Branch calculations allow for the quantification of changes in system responses of interest (eigenvalue, pin powers, homogenized few-group cross sections, and kinetic parameters) due to changes in system parameters. TRITON saves the responses of interest for the nominal and each perturbed (branch) state, for each evaluation within the TRITON depletion scheme. These responses of interest-in particular, homogenized cross sections-may be subsequently extracted for use in nodal core simulation calculations.

Branch calculations represent a branch from the primary depletion scheme at each depletion subinterval. With branching enabled, selected properties or conditions (fuel temperature, moderator temperature, moderator density, soluble boron concentration, and control rod insertion, or any combination thereof) can be varied from the reference state for as many branches as are desired. Depletion calculations, however, are performed for reference-state conditions only. Fig. 3.1.1 illustrates the branch loop during a T-DEPL sequence calculation. Although not technically a branch state, the reference state is considered to be branch 0 for numbering purposes within TRITON. For each branch calculation >0, TRITON updates the appropriate parameters and re-executes the cross section processing and transport calculations. Responses of interest are saved to a database file (i.e., the txtfile16 file) for both the nominal and perturbed-state conditions, and TRITON reverts to cross sections and fluxes from the reference branch 0 to proceed with the depletion calculation. The process repeats following each depletion subinterval, until all depletion subintervals are simulated. Responses of interest are added to the database file for all branches at each depletion subinterval.

Branch perturbations may be applied to any of the following five parameters: fuel temperature, moderator temperature, moderator density, moderator soluble boron concentration, and control rod insertion. These properties may be varied individually or simultaneously. Branch calculations are specified in the TRITON *BRANCH* data block. The *BRANCH* data block has the following form.

```
READ branch
define deftype   I1 I2 ... In end
...
tf=fueltemp tm=modtemp dm=moddens sb=boronconc cr=inout end
...
END branch
```

where

deftype = ‘fuel,’ ‘mod,’ ‘crout’, or ‘crin’,

Ii = list of materials associated with type definition *deftype*,

fueltemp = branch fuel temperature (K),

modtemp = branch moderator temperature (K),

moddens = branch moderator density (g/cm^3),

boronconc = soluble boron concentrations (ppm),

inout = control rod/blade state (out = 0, in = 1).

The type definitions must come first within the *BRANCH* block, and at least one definition is always required. The 'fuel' type definition is used to specify which of the problem materials are considered to be fuel during branch calculations; similarly, the 'mod' type definition specifies the material or materials that are to be considered moderator. The 'crout' definition specifies the materials that are in place in the transport model when control structures are withdrawn, while the 'crin' definition specifies the materials that are present in the transport model when a control structure is inserted. The 'fuel' definition must be present if any fuel temperature branches are performed. The 'mod' type definition must be present whenever moderator temperature, moderator density, or soluble boron branches are performed. Both the 'crout' and 'crin' definitions must be present if control rod branches are requested. Definitions may not be repeated-for example, 'define fuel' may occur only once.

Type definitions are followed by branch specifications. For each branch, one or more branch specifications may be given; if a particular property is omitted, then the reference conditions of the original model and material specifications are used. **The first branch specification must describe the nominal conditions,** and all parameters must be specified for this branch. Each branch specification can optionally define up to five branch keywords before terminating with the *end* keyword. The five branch keywords are as follows.

tf = fuel temperature (K),

tm = moderator temperature (K),

dm = moderator density (g/cm^3),

sb = soluble boron concentration (ppm boron), and

cr = control rod state (out = 0, in = 1).

The format of a *BRANCH* block is best illustrated by an example. Fig. 3.1.5 shows a complete branch data block for a five-branch calculation, with embedded descriptions of each branch. Note that there are six entries; the first branch is the reference or branch 0 state.

In this example, materials 11 and 12 are specified as 'fuel', and fuel temperature perturbations will be applied to only these materials. The nominal temperature for both materials is determined from the branch 0 input (901 K). The nominal fuel temperature must be the same for all materials in the definition and must be consistent with the initial standard composition input. Similarly, materials 13 and 14 are defined as the moderator materials. The temperature (559 K), density (0.76 g/cm^3), and soluble boron concentrations (655 ppm) for the reference state must be identical to those of the initial material specifications and must be identical for all materials defined as moderator.

```
READ branch
define fuel 11 12 end
define mod 13 14  end
define crout 20 21 end
define crin  30 31 end
'reference state
tf=901 tm=559 dm=.4 cr=0 sb=655 end
'moderator density branch
dm=.80 cr=0 end
'moderator density + control rod insertion branch
dm=.80 cr=1 end
'moderator density + soluble boron (low) branch
sb=20 dm=.8 end
'moderator density + soluble boron (high) branch
sb=1300 dm=.8 end
'moderator density + fuel temperature branch
tf=559 dm=0.8 end
END branch
```

Fig. 3.1.5: Example BRANCH block input.

In a reactor core, when a control structure (rod, blade, etc.) is withdrawn, the volume occupied by the structure is replaced by something else. Thus, in a branch calculation with rod insertion and withdrawal, the material(s) present for both states must be specified. If the reference condition is defined as control rods withdrawn (i.e., cr = 0), the NEWT geometry model must contain the materials defined by 'crout'. For a control rod insertion branch (cr = 1), TRITON exchanges the materials specified in the 'crin' definition (30, 31) with corresponding materials in the 'crout' definition (20, 21). Conversely, if the reference condition is defined as control rods inserted (i.e., cr = 1), the NEWT geometry model must contain the materials defined by 'crin'. For a control rod withdrawal branch (cr = 0), TRITON exchanges the materials specified in the 'crout' definition with corresponding materials in the 'crin' definition. For this reason, unique material numbers must be paired between crin and crout definitions. For example, consider a zirc-clad B_4C control rod inserted during a control rod insertion branch, with materials 30 and 31 representing the clad and rod materials, respectively. In the withdrawn position, both the clad and poison materials are replaced by the moderator. To have consistent definitions of 'crin' and 'crout', two moderator materials must be defined for the withdrawn state: one corresponding to the clad material and one corresponding to the rod material.

As mentioned earlier, only one condition keyword is required per branch, but all five may be used. However, the reference state (branch 0) entry must specify all five conditions. For subsequent branches, when a specific branch state is not specified, the reference state is used. In the above example, the first entry, branch zero, specifies the reference state with a fuel temperature of 901 K, moderator temperature of 559 K, moderator density of 0.4 g/cm^3, control rod withdrawn, and a soluble boron concentration of 655 ppm. The second entry (branch 1) specifies a moderator density of 0.80 g/cm^3 and the control rod state as withdrawn. Since the reference state is for a withdrawn control rod, the statement cr = 0 is redundant (but completely acceptable). The next branch is identical to the previous branch, except that in this case the control rod is inserted. For

both cases, reference fuel and moderator temperatures were used. In the following branch, the soluble boron concentration is changed to 20 ppm, and the moderator density is again set to a value of 0.8 g/cm^3. In fact, this moderator density is applied to all five branches. Along with the moderator density change, the soluble boron concentration is changed to 1300 ppm for the next branch. And finally, in the last branch, in addition to the moderator density change, the fuel temperature is changed to 559 K. For this case, reference conditions are used for boron concentration, moderator temperature, and control rod state.

Note that TRITON compares the reference values of fuel temperature, moderator temperature, moderator density, and soluble boron concentration with the data entered in the *COMPOSITION* block. TRITON prints warning messages if the data in the *COMPOSITION* block and *BRANCH* block are inconsistent. Also note that each branch calculation is independent of other branch calculations. Thus, the order in which branch calculations are computed is not important.

Branch calculations are usually requested for lattice physics analysis, where the objective is to generate a database of few-group homogenized cross sections for nodal core calculations. Thus, BRANCH blocks are used in tandem with the NEWT's *COLLAPSE, HOMOGENIZATION,* and *ADF* blocks. With these blocks of data, TRITON will archive lattice physics data-few-group homogenized cross sections, assembly discontinuity factors (ADFs), homogenized kinetic parameters, pin powers, and form factors-to a binary file called xfile016 in the SCALE temporary working directory. An auxiliary text-formatted data file called txtfile16 is also created in the SCALE temporary working directory. This file format is documented in Sect. 3.1.7.1.

BRANCH block with user-defined Dancoff factors

As previously mentioned in Sect. 3.1.3.1.2, TRITON uses Dancoff factors as part of its cross section processing calculations. Dancoff factors play an important role in characterizing spatial self-shielding effects. The XSProc module computes the Dancoff factors based on the *CELLDATA* input. For a square-pitched lattice cell example, Dancoff factors are computed by DANCOFF by assuming that the fuel pin is within an infinite lattice of identical fuel pins. The assumption of an infinite uniform lattice of fuel pins may lead to inaccurate Dancoff factors for certain configurations such as BWR assembly designs, leading to inappropriate problem-dependent multigroup cross sections. Moreover, the Dancoff factors may change significantly for certain branch conditions, such as changing the in-channel moderator density in a BWR assembly.

The TRITON BRANCH block allows the user to specify material-dependent Dancoff factors for various branch conditions. Branch-specific Dancoff factors may be utilized by defining a new set of material-dependent Dancoff factors using the *d2pset* type definition. The set of Dancoff factors may be included in a branch specification by using the *d2p=* keyword. The *BRANCH* block now has the following format.

```
READ branch
define deftype   I1 I2 ... In end
define d2pset id M1 D1 M2 D2 ... Mn Dn end
...
tf=fueltemp tm=modtemp dm=moddens sb=boronconc cr=inout d2p=d2pID end
...
END branch
```

In the type definition section, the *d2pset* keyword is followed by a positive integer identifier, which is subsequently followed by pairs of material identifiers and their user-defined Dancoff factor value. Multiple material/Dancoff pairs may be entered for a particular set definition, as long as the material identifiers are unique. Multiple set definitions are allowed, as long as the set identifiers are unique.

The *d2p=* keyword in the branch specification can be assigned to any set identifier defined in the branch definition section. If *d2p=* is utilized, the material/Dancoff pairs in the set definition are applied for the

given branch condition. The values *d2p=0* and *d2p=-1* have special meaning. If *d2p=* is set to 0, the material/Dancoff pairs defined in the *CELLDATA* block are utilized. If *d2p=* is set to -1, the default MIPLIB-computed Dancoff factors will be utilized, even if material/Dancoff pairs are defined in the *CELLDATA* block using the *dan2pitch* keyword available there. The nominal (branch 0) condition **must** use the material/Dancoff pairs (if defined) in the *CELLDATA* block; therefore, the first branch specification **must** not set the *d2p* keyword to anything other than zero. (Note: *d2p=0* need not be defined for the first branch condition since this is always the case.)

In Fig. 3.1.6, the *BRANCH* block from the previous example has been modified to use branch-specific Dancoff factors. In this example, the nominal branch defines the reference moderator density to be 0.4 g/cm^3, and five branches use a higher moderator density of 0.8 g/cm^3. The Dancoff factors for the higher moderator density condition are different from the reference moderator density. To account for the different Dancoff factors at the higher moderator density condition, a set of material/Dancoff pairs are defined with the set identifier of 400. In the set, fuel material 11 has a Dancoff factor of 0.4, and fuel material 12 has a Dancoff factor of 0.5. The set of Dancoff factors is used for the five branch states through the specification of the *d2p=* keyword to 400.

```
READ branch
define fuel 11 12 end
define mod 13 14  end
define crout 20 21 end
define crin  30 31 end
define d2pset 400 11 0.4 12 0.5 end
'reference state
tf=901 tm=559 dm=.4 cr=0 sb=655 end
'moderator density branch
dm=.80 cr=0 d2p=400 end
'moderator density + control rod insertion branch
dm=.80 cr=1 d2p=400 end
'moderator density + soluble boron (low) branch
sb=20 dm=.8 d2p=400 end
'moderator density + soluble boron (high) branch
sb=1300 dm=.8 d2p=400 end
'moderator density + fuel temperature branch
tf=559 dm=0.8 d2p=400 end
END branch
```

Fig. 3.1.6: Example BRANCH block input with Dancoff factors.

DEPLETION block

The *DEPLETION* block, used by the four depletion sequences, is simple in concept but performs four important functions. First, this block specifies the materials for which depletion calculations are to be performed. In general, it is desirable to perform depletion calculations only for fuel and target materials of interest. Calculating the depletion of gas gaps, cladding, moderator, or coolant is usually of little value unless the material contains components that will be significantly depleted with burnup. Additionally, it is not usually desirable to deplete soluble poisons in reactor coolants. Therefore, the *DEPLETION* block requires that the user specify the materials to be depleted. There are no defaults; hence, the block is required for all depletion sequences.

The second function of the *DEPLETION* block is to specify the basis to which the model power is normalized. In general, the average time-dependent power to which an irradiated object is exposed is known. For example, an LWR fuel assembly discharged from a reactor is known to have operated at certain power levels for one or more time periods. The individual pins in the assembly will have varying power levels depending on position and assembly design. In such a case, the basis for the input power is the full assembly. Fluxes computed in the transport solution will be normalized by TRITON based on reaction rates and energies **in all problem materials** (depleted and nondepleted materials) such that the assembly-wide power will match the power given in *BURNDATA* block. However, it is often the case in radiochemical assay analysis that the burnup history of a specific pin is known and isotopic concentrations for that pin are desired. It is still necessary to model the full assembly in order to properly characterize the fluxes in that pin. In such a case, it would be advantageous to specify the operating history for the pin instead of the full assembly. When this is done, the average specific power of the full assembly will be different from that of the pin and will be computed automatically based on power distributions calculated for the assembly. In other words, powers for other materials in the assembly will be normalized such that the power in the pin of interest matches that specified in the *BURNDATA* block. The material of that pin becomes the *basis* for power normalization.

Sect. 3.1.3.3.5 below describes the general format of the *DEPLETION* block that is available to all four depletion sequences. The third function of the *DEPLETION* block is an optional function used to specify ORIGEN solver options and ORIGEN depletion mode for each depletion material. These options are further described in Sect. Sect. 3.1.3.3.6. The fourth function of the *DEPLETION* block is to define optional deletion instructions used to simplify cross section processing using the ASSIGN function. Special provisions have been made in the 1D and 2D depletion sequence (T-DEPL-1D and T-DEPL) to reduce the number of cross section processing calculations in order to decrease calculation run-time. The ASSIGN functionality is further described in Sect. Sect. 3.1.3.3.7.

Basic DEPLETION block format

The basic format of the DEPLETION block is as follows:

```
READ depletion M1 M2 M3... Mn END depletion
```

where M_i represents the SCALE material numbers for materials to be depleted. As discussed above, the *DEPLETION* block can also be used to specify the basis for the input power. Power normalization is accomplished by prefixing the material number(s) with a negative sign (-). For example, consider a problem in which materials 1, 2, and 3 are being depleted, but the power for material 1 is known. The *DEPLETION* block for this case is

```
READ depletion
  -1 2 3
END depletion
```

In this case, powers for all materials will be normalized such that the power in material 1 matches the input power specification in the *BURNDATA* block.

Note that multiple materials can be used as a power basis. Consider a fuel assembly with three fuel types represented by materials 1, 2, and 3, and also containing cladding as material 4 and water as material 5. The following illustrates multiple ways that the power basis for this assembly might be specified and describes the effect of each specification.

- The assembly-averaged power is normalized to match the input specific power. Power generated by moderator and clad is included but they are not depleted.

```
READ depletion
  1 2 3
END depletion
```

- The assembly-averaged power is normalized such that the power of material 1 matches the input specific power.

```
READ depletion
  -1 2 3
END depletion
```

- The assembly-averaged power is normalized such that the average power in materials 1 and 2 matches the input specific power.

```
READ depletion
  -1 -2 3
END depletion
```

- The assembly-averaged power is normalized to match input specific powers. TRITON will attempt to do depletion in cladding and moderator materials too. (Note that cladding and moderator materials should be depleted using the deplete-by-flux option described in the next subsection).

```
READ depletion
  1 2 3 4 5
END depletion
```

- The assembly-averaged power is normalized such that the average power in materials 1–3 matches the input specific power. This is not the same as the normalizing specification for an assembly average, because it neglects contributions of, for example, (n, γ) sources in moderator and cladding materials.

```
READ depletion
  -1 -2 -3
END depletion
```

ORIGEN depletion options

ORIGEN provides two input options for the flux used in the depletion calculation: direct specification of fluxes (i.e., deplete by flux) or indirect specification of fluxes in terms of power (i.e., deplete by power). The ORIGEN depletion is based on a known flux; however, it is more often the case that one knows the specific power in a depletion region rather than the actual flux. When ORIGEN is used in deplete-by-power mode, ORIGEN will internally determine the corresponding flux from the input-specific power and internal tables of fission and capture energy releases for the nuclides present and the macroscopic cross sections of those nuclides. Additionally, at each ORIGEN time interval, ORIGEN recalculates the material power density as nuclide inventories change. Hence, the deplete-by-power mode will result in a time-varying flux, whereas the

deplete-by-flux mode will result in a constant flux over the calculation time interval. Since reactors typically operate at a constant (or nearly so) power level, with varying local fluxes, the deplete-by-power option is closer to reality. However, the choice of approach is generally not an issue. Significant differences between calculation results between the two depletion modes would indicate that the TRITON depletion subintervals are too large.

By default, all TRITON depletion materials use the deplete-by-power mode. However, there exist some circumstances where deplete-by-flux is more appropriate. In deplete-by-power mode, ORIGEN will often halt when an attempt to maintain constant power results in a large change in flux between ORIGEN time intervals. Large changes in flux can occur in media where isotope contents are changing rapidly with time, such as in a gadolinium-bearing burnable absorber rod, where gadolinium is being rapidly depleted with time. Another circumstance pertains to activation analysis of nonfuel materials. The flux for these materials is typically governed by external power sources (i.e., fuel materials located elsewhere in the problem domain) rather than by internal power sources. Therefore, the deplete-by-flux option is recommended for these materials.

TRITON provides the option to specify deplete-by-flux mode for selected depletion materials using a modified form of the depletion specification:

```
READ depletion M1 M2 M3...Mi-1 flux Mi Mi+1... Mn END depletion
```

Materials preceding the *flux* keyword are depleted using the deplete-by-power mode; materials following the flux keyword are depleted using the deplete-by-flux mode. For example, consider a problem in which materials 1–6 are to be depleted, but materials 3 and 4 represent nonfuel materials that do not contribute significantly to the total power and are therefore to be depleted assuming constant flux. The *DEPLETION* block for this situation could be specified as follows.

```
READ depletion 1 2 5 6 flux 3 4 END depletion
```

The DEPLETION block also supports the specification of the ORIGEN calculation method. The default option is solver=matrex, which represents the matrix exponential option. The other option is solver=cram, which represents the new CRAM solver option in ORIGEN. An example depletion specification for the cram solver is as follows.

```
READ depletion 1 2 5 6 flux 3 4 solver=cram END depletion
```

Cross section processing simplification using ASSIGN

When depleting a large number of fuel materials, considerable time may be spent in the cross section processing calculations prior to the multigroup transport calculation. Fuel assembly designs may require 20-200 unique depletion materials across the different fuel pins in the assembly. In such cases, an assembly model may require hours of run-time for each pin-wise cross section processing calculation in order to perform a 10-minute transport solution.

Although highly rigorous, such a cross section processing process is extremely burdensome for depletion calculations, especially if branch calculations are requested. To reduce run-time, the 2D depletion sequence (*T-DEPL*) provides the option to group depletion materials together such that they are tracked independently in the depletion calculation but use a common set of microscopic cross sections. The microscopic cross sections for a given depletion group are computed using the average composition of all the depletion materials within the group. Typically, this grouping is applied to fuel pins of identical initial composition. Although the nuclide number densities of such pins will diverge with burnup as a function of location within an assembly, the cross sections of these pins are well represented by a single pin cell calculation with an average composition representative of all these pins.

Although the material grouping option introduces approximations in the cross section processing calculations, which in turn affects the transport and depletion calculations, internal investigations have shown that solution accuracy can be maintained for a wide range of assembly designs while significantly improving the run-time.

The alternate format of the *DEPLETION* block for simplified cross section processing is as follows.

```
READ depletion M1 M2 M3... Mz END
assign N1 Ma Mb ... Mx end
assign N2 Mf Mg ... My end
...
assign Nn Mj Mk ... Mz end
END depletion
```

Similar to the basic format, each material designated for depletion (M_i) is listed after *READ* depletion and before the *END* keyword. Each designated depletion material must be present in the 2D NEWT model. After the first *END* keyword, the alternate format contains a list of material "assignments" used to simplify cross section processing for a group of depletion materials. The material assignments begin with the *assign* keyword and terminate with the *end* keyword. After the *assign* keyword, a unique representative material identifier (N_j) is defined. The representative material is associated with the group of depletion materials that immediately follows in the *assign* definition. The representative material identifier is used in the *COMPOSITION* and *CELLDATA* blocks to define the initial composition, temperature, and cell definition for the group of depletion materials. Thus, the *assign* definitions in *TRITON* are currently constrained such that each depletion material group must have the same initial composition. After the last *assign* definition, the depletion block is terminated with *END depletion*.

Only depletion materials may be assigned to representative materials. The group of depletion materials assigned to a particular representative material must **not** appear in the *COMPOSITION* and *CELLDATA* blocks.

The use of material assignments is best illustrated by an example. Fig. 3.1.7 shows a complete T-DEPL input that uses material assignments. A 2D plot of the model is shown in Fig. 3.1.8. In this example, two fuel materials are defined as materials 1 and 2 in the *COMPOSITON* block. In the *DEPLETION* block, the list of depletion materials includes materials 1, 20, 30, and 40. Depletion materials 20, 30, and 40 are "assigned" to representative material 2. Material 2 does not appear in the depletion list or the transport model; materials 20, 30, and 40 do. But only material 2 is defined in the *COMPOSITION* and *CELLDATA* blocks. In the transport model, four units are defined, one for each material. An array is used to place each unit in its own location.

The initial calculation uses material 2 to define the compositions of materials 20, 30, and 40, since all are initially identical. Microscopic cross sections computed for material 2 are used for each of the three assigned depletion materials during the transport calculation and the depletion calculation. After the first depletion calculation, materials 20, 30, and 40 will have different isotopic concentrations because of different locations in the nonsymmetric transport model. At this time, the number densities in each of these three materials are averaged and used to update the concentration of representative material 2. A new set of cell calculations will be performed for materials 1 and 2; this will be followed by a transport calculation that uses the microscopic cross sections for material 2 along with local nuclide number densities for materials 20, 30, and 40 to calculate new and unique macroscopic cross sections for each. The transport and subsequent depletion calculation are then run. The iterative process will continue until all depletion steps have been completed.

```
=t-depl
example of assigns
v7-238
READ comp
' 2.3 w/o
  u-235 1 0 5.2968e-04 900 end
  u-238 1 0 2.2208e-02 900 end
  o-16  1 0 4.5491e-02 900 end
' 3.6 w/o
  u-235 2 0 8.2904e-04 900 end
  u-238 2 0 2.1907e-02 900 end
  o-16  2 0 4.5497e-02 900 end
'Clad nat. zr per spec.
  zirc4  101 den=6.53 1 559 end
  zirc4  102 den=6.53 1 559 end
' lwtr mod
  h2o 201  den=0.457 1.0000   559 end
  h2o 202  den=0.457 1.0000   559 end
END comp
READ celldata
  latticecell squarepitch pitch=1.63 201 fuelr=0.529 1 cladr=0.615 101 end
  latticecell squarepitch pitch=1.63 202 fuelr=0.529 2 cladr=0.615 102 end
END celldata
READ depletion 1 20 30 40 END
          assign 2 20 30 40 END
END depletion
READ burndata
  power=25.0 burn=300 nlib=5  end
END burndata
READ model
simple 2x2 array
READ parm
  sn=4 drawit=yes echo=yes collapse=yes epsilon=1e-3 cmfd=yes xycmfd=2
END parm
READ materials
  mix=1   pn=1 end
  mix=20   pn=1 end
  mix=30   pn=1 end
  mix=40   pn=1 end
  mix=101  pn=1  com='Zirc4' end
  mix=201  pn=2  com='H2O' end
END materials
READ geom
  unit 1
  cuboid 1 1.63 0.0000 1.63 0.0000
  cylinder 2 0.615 origin x=0.815 y=0.815
  cylinder 3 0.529 origin x=0.815 y=0.815
  media 201 1 1 -2
  media 101 1 2 -3
  media   1 1 3
  boundary 1 3 3
  unit 2
  cuboid 1 1.63 0.0000 1.63 0.0000
  cylinder 2 0.615 origin x=0.815 y=0.815
  cylinder 3 0.529 origin x=0.815 y=0.815
  media 201 1 1 -2
  media 101 1 2 -3
  media  20 1 3
  boundary 1 3 3
  unit 3
  cuboid 1 1.63 0.0000 1.63 0.0000
  cylinder 2 0.615 origin x=0.815 y=0.815
  cylinder 3 0.529 origin x=0.815 y=0.815
  media 201 1 1 -2
  media 101 1 2 -3
  media  30 1 3
  boundary 1 3 3
  unit 4
  cuboid 1 1.63 0.0000 1.63 0.0000
  cylinder 2 0.615 origin x=0.815 y=0.815
  cylinder 3 0.529 origin x=0.815 y=0.815
  media 201 1 1 -2
  media 101 1 2 -3
  media  40 1 3
  boundary 1 3 3
  global unit 50
  cuboid 1 3.26  0.0  3.5  0.0
  array 1 1 place 1 1 0 0
  media 201 1 1
  boundary 1  6  6
END geom
READ array
  ara=1 nux=2 nuy=2 pinpow=yes fill   1   2
                                      3   4 end fill
END array
READ bounds
  all=refl
END bounds
END model
END
```

Fig. 3.1.7: Example input with material assignments.

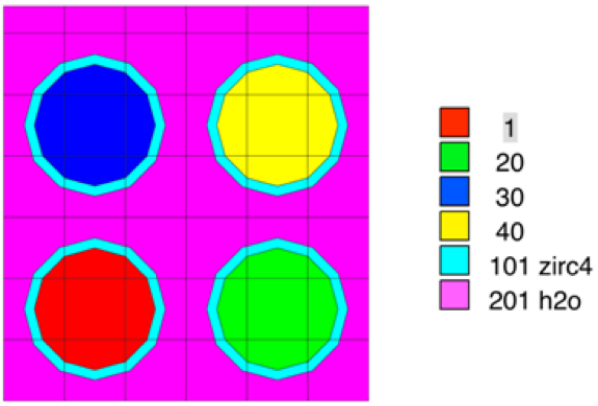


Fig. 3.1.8: Example 2D model plot of material assignments.

The use of assignments can make a considerable difference in run-time performance with minimal sacrifice in accuracy. The above example ran 1.8 times faster using the assignment of three similar pins to one initial specification. A larger BWR calculation, in which 41 pin positions were depleted independently, was run in an assessment of the accuracy of the method. Using this approach, the simplified representation ran 20 depletion steps in 20% of the time required for the explicitly modeled cells. Fig. 3.1.9 shows a comparison of the eigenvalues using the simplified (with assignments) and explicit (without assignments) models. Also shown on the figure is the percent difference between the approximate and explicit models. For this model, the error in k_{eff} remains well below 0.05%.

Note that one can combine depletion mode control with material assignments, as follows.

```
READ depletion 1 2 5 6
          flux 3 4 end
         assign 11 1 2 end
         assign 12 3 4 end
         assign 13 5 6 end
END depletion
```

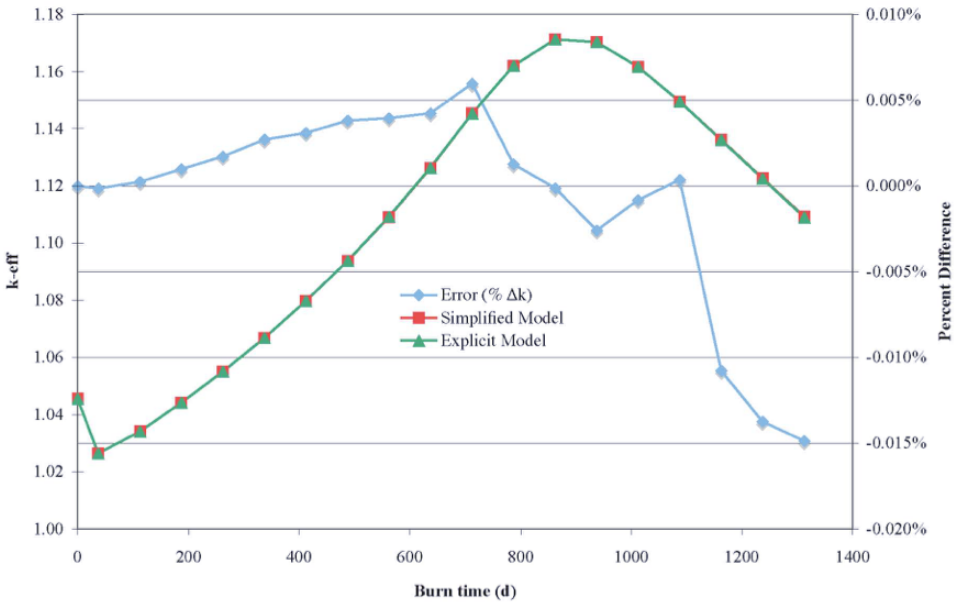


Fig. 3.1.9: Eigenvalue comparison of simplified cross section processing example.

TIMETABLE block

In many depletion analyses, material properties can change due to influences outside the depletion process (e.g., boron letdown in pressurized water reactors [PWRs], the insertion or removal of poisons during or between fuel cycles, or changes in temperatures of materials with time). The *TIMETABLE* block has been provided to allow modification of properties during a depletion calculation. Timetables may be entered for any material or for selected nuclides within a material and allow changes in number densities or temperatures. Timetables may also be entered to swap a material in and out of the geometry during depletion. Continuous feed and/or removal to/from mixtures during depletion can be enabled for analysis of systems with flowing fuel.

The *TIMETABLE* block takes the following general format.

```
READ timetable
[time dependent specifications for a given material]
[time dependent specifications for a given material]
[time dependent specifications for a given material]
END timetable
```

Four different material specifications are allowed to modify temperature, density, swap materials, or fractional removal/continuous feed of nuclides from/to materials.

Temperature and density

Temperature timetable entries are specified in the format

```
temperature I t1 K1 t2 K2 t3 K3...tC KC end
```

where

I = material ID number;

t_i = time (days) in calculation where temperature K_i is set, i = 1 to C;

K_i = temperature (in K) of specified materials at time t_i, i = 1 to C;

C = number of time steps.

Density entries have an analogous specification, with the addition of a couple of extra terms:

```
density  I  M N1 N2 N3 ... NM   t1 D1 t2 D2 t3 D3...tC DC end
```

where

I = material ID number;

M = number of nuclides to which this change is applied;

N_i = nuclide ID for the i_{th} nuclide in the list, i = 1 to M;

t_j = time (days) in calculation where density multiplier D_j is set, i = 1 to C;

D_j = density multiplier (fractional change) of specified nuclides at time t_j, i = 1 to C;

C = number of time steps.

In both formats, time and data (temperature or density multiplier) must be entered in pairs. Note that density changes may be applied to specific nuclides, while for temperature the change must be applied to all nuclides within the material simultaneously. If M (the number of nuclides for which the density is to be modified) is specified as 0 and no nuclide IDs are entered, then the timetable values are applied to all nuclides in the material.

Note that timetable entries are specified at distinct times in the calculation. These times are measured relative to the beginning of the calculation and are *continuous* (as opposed to *BURNDATA* entries, which give burn times or down times in increments per depletion interval). The initial timetable entry should always begin at t=0 days. To allow for time-dependent changes in properties, *TRITON applies linear interpolation between data pairs.* To hold a parameter constant over a time interval, that parameter should be specified at the same value at both the beginning and the end of this time interval.

The application of timetable entries is best illustrated by example. Consider the depletion scheme described by the following *BURNDATA* block which contains three depletion intervals:

```
READ burndata
power=26.54 burn=121.0 nlib=3 end
power=38.01 burn=201.5 down=30 nlib=2 end
power=31.44 burn=386.25 down=1826.25 end
END burndata
```

Assume that the moderator, material 3, has temperatures and boron concentrations that vary over the three depletion intervals in the following manner:

Interval	Boron concentration (ppm)		Temperature (K)
	BOC	EOC	
1	1000	100	615
2	1250	130	685
3	980	100	610

```
READ timetable
temperature 3
0.0     615
121.0   615
121.01  685
322.5   685
352.5   610
738.75  610 end
density 3 2 5010 5011
0.0     1.0
121.0   0.1
121.01  1.25
322.5   0.13
352.5   0.98
738.75  0.1 end
END timetable
```

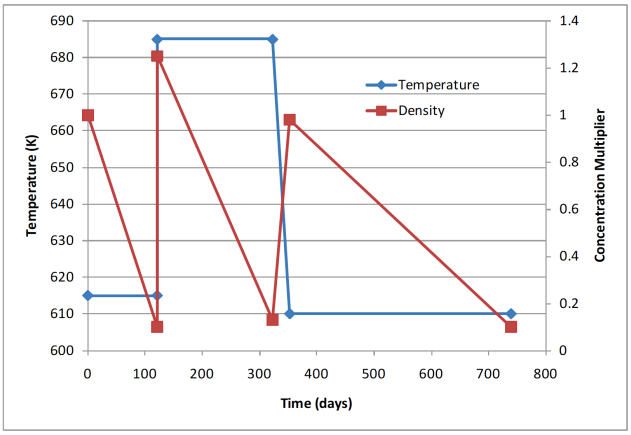


Fig. 3.1.10: Example temperature and density *TIMETABLE* block input.

Note: It is important to note that time-dependent changes to temperatures and number densities are not applied continuously over the depletion calculation, but instead are applied only at the times at which cross section processing and transport calculations are performed - that is, the midpoint of the depletion subintervals. *The user must determine the accurate depletion scheme specific to his or her application to accurately model time-dependent changes in system properties.*

Density timetable specifications can be used to effectively exchange compositions of a single material. One may construct a compound material comprised of two distinct materials at their design densities; a timetable specification can be used to set the density multiplier to 1.0 for the nuclides initially present and to use a multiplier of 0.0 for all nuclides in materials that are not intended to be present at time zero. The timetable can then affect the exchange by changing the multipliers from 0 to 1, and from 1 to 0, at the time of the material exchange. One must bear in mind that timetable processing within TRITON performs linear interpolation between time points; if the exchange is intended to occur at a specific moment in time, then the timetable should be set up with the exchange occurring within a very short period. Moreover, it is important to note that material exchanges for two materials that have common nuclides are more difficult to model. For example, a B_4C absorber material and borated H_2O moderator material both contain boron nuclides in common. In order to exchange the B_4C absorber material and the borated H_2O moderator material, the carbon, oxygen, and hydrogen density multipliers would be 0 or 1, but the boron density multipliers would need to be derived from the boron concentrations in both materials.

Material swap

Material exchange timetables offer another option to users to exchange one material with another material during depletion calculations.

Material exchange timetable has a similar format to temperature timetables:

```
swap I1 I2 t1 S1 t2 S2 t3 S3...tC SC end
```

where

I1 = first material ID

I2 = second material ID

t_j = time (days) in calculation where swap ID is set, i = 1 to C;

S_j = swap value 0/1 at time t_j, i = 1 to C;

C = number of time steps.

The first two entries in the timetable specify the material IDs for swap materials. The remaining entries are entered in pairs: the first pair value is a time value, the second pair value is either "0" or "1". "0" instructs TRITON to model the swap materials as defined in the nominal model. "1" instructs TRITON to swap the materials (swap every *I1* for every *I2* and swap every *I2* for every *I1*). The swap state persists until the next time entry in the timetable. For the last time entry in the timetable, the swap state persists for the duration of the calculation. For example:

```
read timetable
  swap 5 6  0 0  100 1  200 0 end
read timetable
```

- Do not perform the material swap on the interval [0, 100],
- Perform the material swap on the interval [100, 200], and
- Return to the nominal state at time 200 days until the duration of the calculation.

Depending on the *BURNDATA* specification, there may be one or more depletion/decay steps between timetable entries. Moreover, for accurate depletion modeling, material exchanges must not occur during a depletion subinterval. If a material exchange occurs during a depletion interval, TRITON will subdivide the depletion subinterval at the time of the material exchange. Extending the example above, assume the *BURNDATA* block is as follows:

```
read timetable
  swap 5 6  0 0  100 1  200 0 end
read timetable
read burndata
 power=40 burn=300 nlib=4 end
end burndata
```

Without the material exchange table, the depletion subintervals are [0, 75], [75, 150], [150, 225], and [225, 300]. With the material exchange table, the subintervals are:

- [0, 75] – Swap value is 0
- [75, 100] – Swap value is 0
- [100–150] – Swap value is 1, i.e. materials 5 and 6 swap definitions

- [150–200] – Swap value is 1, i.e. materials 5 and 6 swap definitions
- [200–225] – Swap value is 0, materials 5 and 6 return to their original definitions
- [225–300] – Swap value is 0

As a limitation of the material exchange timetable, if a depletion material is removed from the geometry, the isotope concentrations at the time of removal are stored in-memory, and then reused upon re-entry into the geometry. In other words, the depletion material does not undergo radioactive decay for the period of time outside the problem geometry.

Material flow

To allow modeling of systems with flowing fuel, TRITON offers a *FLOW* block which permits fractional removal and continuous feed from/to mixture. When this block is requested, TRITON makes use of ORIGEN's capability for feed and removal from/to mixtures, including the decay of nuclides removed from the system. Detailed information about the feed and removal to simulate flowing fuel can be found in [TRITONVBWF20], [TRITONBPW17], [TRITONBPBR17].

Eq. (5.1.1) from the ORIGEN part of the manual (Sect. 5.1.3) is as follows when explicitly adding a removal rate $\lambda_{i,rem}$ and acknowledging the feed rate S_i:

$$\frac{dN_i}{\mathrm{dt}} = \sum_{j \neq i} (l_{\mathrm{ij}} \lambda_j + f_{\mathrm{ij}} \sigma_j \Phi) N_j(t) - (\lambda_i + \lambda_{i,rem} + \sigma_i \Phi) N_i(t) + S_i(t) \tag{3.1.1}$$

where

- N_i = amount of nuclide i (atoms),
- λ_i = decay constant of nuclide i (1/s),
- $\lambda_{i,rem}$ = **removal constant** defining the continuous removal of nuclide i (1/s),
- l_{ij} = fractional yield of nuclide i from decay of nuclide j,
- σ_i = spectrum-averaged removal cross section for nuclide i (barn),
- f_{ij} = fractional yield of nuclide i from neutron-induced removal of nuclide j,
- Φ = angle- and energy-integrated time-dependent neutron flux (neutrons/cm^2-s), and
- S_i = time-dependent **source/feed term** (atoms/s).

The *FLOW* block can consider two scenarios: (1) continuous feed into a mixture and (2) fractional removal from one mixture into another. In the case of the continuous feed, the feed rate (source) must be specified by the user. In the case of the fractional removal of a nuclide from material 1 to material 2, the removal rate (removal constant) for material 1 is specified. The removed nuclides from material 1 are then considered as source for material 2.

The following *FLOW* block enables continuous feed of nuclides into a mixture:

```
read timetable
  flow
    to I2
    type continuous_feed
    units [pers or gpers]
    nuclides [N1 N2 ... NM] end
    rates    [R1 R2 ... RM] end
    time       [t1 t2 ... tC] end
    multiplier [f1 f2 ... fC] end
  end flow
end timetable
```

The following *FLOW* block enables fractional removal of nuclides from one mixture to another:

```
read timetable
  flow
    from I1 to I2
    type fractional_removal
    units [pers or gpers]
    nuclides [N1 N2 ... NM] end
    rates    [R1 R2 ... RM] end
    time       [t1 t2 ... tC] end
    multiplier [f1 f2 ... fC] end
  end flow
end timetable
```

The input parameters are as follows:

- I = mixture ID
- *from (to)* = identifier from which mixture nuclides are removed (from I1) and to which mixture they are added (to I2)
- *type* = type of addition or removal of nuclides from/to mixture:
  - *fractional_removal* requires *from* and *to*
  - *continuous_feed* requires *to* and does not permit *from*
- *units* = *pers* or *gpers* unit of the flow rate constant used:
  - for flow blocks with flow *from* a mixture *to* another mixture: pers (1/second)
  - only flow *to* a mixture: gpers (grams/second)
- M = number of nuclides to which change is applied
- N_i = nuclide ID for the i_{th} nuclide in the list, i = 1 to M
- R_i = rates for the i_{th} nuclide in the list, i = 1 to M
- t_j = time (days) in calculation where multiplier f_j is set, i = 1 to C, t1 must be 0.0
- f_j = multiplier to vary specified flow rate (based off the initial flow rate) at time t_j, i = 1 to C
- C = number of time steps.

If nuclides are transferred from mixture 1 into mixture 2, but mixture 2 is not part of the system (e.g., contained in a loop outside the reactor), then the decay of the nuclides in this mixture can still be considered by specifying keyword *decayonly* in the *DEPLETION* block.

The following is an example in which xenon is transfered from mixture 1 to mixture 2, and the decay of nuclides in mixture 2 is enabled in the *DEPLETION* block:

```
read depletion
-1 decayonly 2
end depletion

read timetable
  flow
   from 1
   to 2
   type continuous_feed
   units pers
```

(continues on next page)

(continued from previous page)

```
   nuclides    xe end
   rates     2e-2 end
   time       0.0 end
   multiplier 1.0 end
 end flow
end timetable
```

OPUS block

The OPUS module of SCALE is fully documented in the OPUS chapter of the SCALE manual. OPUS provides the ability to extract specific data from ORIGEN output libraries, perform unit conversions, and generate plot data for post-calculation analysis. In essence, OPUS is an ORIGEN post-processor that provides data in the desired form for a desired subset of nuclides. TRITON by default calls OPUS to extract nuclide concentrations for selected nuclides for all depletion materials and for the most important nuclides. TRITON provides the capability to specify the full set of OPUS commands to tailor OPUS calculations to obtain specific information. TRITON allows a stacked set of OPUS calculations in order to retrieve selected data for selected nuclides.

The content of the *OPUS* block is based on standard OPUS input parameters, as described in the OPUS chapter; the details of OPUS control and use are not repeated here. However, additional input is necessary to support TRITON operations with OPUS, because TRITON enables additional capabilities beyond those provided for in standard OPUS input. For example, OPUS is designed to process the output file from a single ORIGEN calculation; because ORIGEN is a point depletion solver, the output represents data from a single material. TRITON is typically used to perform multiple depletion calculations at each depletion step-one calculation for each material being depleted. Hence, multiple OPUS calculations are needed to obtain results from multiple materials. The OPUS calculations are performed automatically by TRITON but require the user to specify the materials for which OPUS processing is desired. Additionally, TRITON supports stacked OPUS cases within the *READ OPUS* data block; hence, keywords are introduced to separate stacked cases.

There are two alternatives available to SCALE users that are complimentary to the OPUS block within TRITON. First, standalone OPUS case(s) can be used to post process the ORIGEN binary concentration file (.f71 extension). This file is automatically saved in the output directory with the file name `${OUTBASENAME}.f71`. (e.g. if the input file is reactor.inp, the concentration file is saved in the output directory as reactor.f71) Second, the user may also open the concentration file within Fulcrum to enable similar post-processing capabilities.

Selection of materials for OPUS processing

Beyond standard OPUS input keywords (see OPUS chapter), TRITON reads a *matl*= keyword to allow specification of ID number(s) for the material(s) in the problem for which outputs are desired. The *matl*=. . . *end* input keyword accepts one or more materials from the *DEPLETION* data block for which OPUS processing is desired. If omitted, OPUS processing will be performed for all materials in the *DEPLETION* block. For example, consider the following *DEPLETION* and *OPUS* data blocks:

```
READ depletion 1 2 3 4 5 6 END depletion
READ opus
units=gram symnuc=u-234 u-235 u-236 u-238 pu-238 pu-239
pu-240 pu-241 pu-242 pu-243 np-237 end
time=year
END opus
```

In this example, OPUS processing will be performed for all depletion materials, 1–6. Adding a subset of materials using the *matl*= keyword, for example.

```
READ depletion 1 2 3 4 5 6 END depletion
READ opus
units=gram symnuc=u-234 u-235 u-236 u-238 pu-238 pu-239
pu-240 pu-241 pu-242 pu-243 np-237 end
time=year
matl=1 2 3 end
END opus
```

will result in OPUS calculations for materials 1, 2, and 3 only.

Although ORIGEN calculations are performed only for individual materials, TRITON provides the capability of combining the results of all or a subset of all depletion materials to get a multimaterial average set of ORIGEN responses. TRITON provides two special ID numbers for combining material results. Specification of material ID 0 will return system-averaged results for the entire set of depletion materials (typically, all fuel elements in a depletion model). Specification of material -1 returns the average of only those materials with ID > 0 present in the *matl*= list. Again, this is best illustrated by example. Specification of the data blocks

```
READ depletion 1 2 3 4 5 6 END depletion
READ opus
units=gram symnuc=u-234 u-235 u-236 u-238 pu-238 pu-239
pu-240 pu-241 pu-242 pu-243 np-237 end
time=year
matl=1 2 3 0 -1 end
END opus
```

will result in five OPUS calculations and five sets of results-one for each of materials 1, 2, and 3, one for the average of materials 1–6 (due to input of material ID 0), and one for the average of materials 1–3 (due to input of material ID –1).

Specification of stacked OPUS cases

In a given calculation, multiple output units may be desired (e.g., grams, curies, and watts), or multiple time scales (e.g., seconds and years), or a combination of these or other parameters. TRITON provides the ability to stack inputs such that multiple cases may be run within a single TRITON calculation. In order to stack cases, the keywords *new case* are entered in the input stream. Any parameters following these keywords are used to define a new OPUS case.

There is no limit on the number of stacked cases that may be input; however, the *matl*= specification may be used only in the first case. OPUS calculations are run for each of the materials in this list, for all cases.

Consider a depletion calculation where gadolinium pins are present in the assembly design. One may wish to determine the quantities of gadolinium nuclides from the initial poison rods (tracked as a light element by ORIGEN within TRITON) and from fission (tracked as a fission product by ORIGEN). One may also need masses of selected actinides as well as the total (α,n) reaction rate. Fig. 3.1.11 shows how the *new case* keyword set is used to define unique OPUS calculations. In this example, the *new case* keywords are shown in upper case and on a line by themselves, but this has been done only for readability. The text may be entered in lower case and on the same line as other keywords. Note, however, that the *matl*= specification is given only in the first case. All OPUS calculations will be performed for materials 1, 2, and 3 and for the average of these three materials.

```
READ opus
  matl=1 2 3 -1 end
  units=grams  time=year  typarms=nucl  libtyp=lite
  title= Masses of gadolinium from poison rods
  symnuc=gd-154 gd-155 gd-156 end
NEW CASE
  units=grams  time=year  typarms=nucl  libtyp=fiss
  title=Masses of gadolinium fission products
  symnuc=gd-154 gd-155 gd-156 end
NEW CASE
  units=grams  time=year  typarms=nucl  libtype=act
  title=Masses of actinides
  symnuc=u-232 u-233 u-234 u-235 u-236 u-238 np-236 np-237 pu-236
       pu-238 pu-239 pu-240 pu-241 pu-242 pu-243 pu-244 am-241
       am-242m am-243 cm-242 cm-243 cm-244 cm-245 cm-246 cm-247
       cm-248 ra-226 ra-228 ac-227 th-229 th-230 th-232 cf-252 end
NEW CASE
  libtype=all  time=year
  title=alpha,n emission rate
  typarams=anspec units=particles
END OPUS
```

Fig. 3.1.11: Example OPUS block input.

FGXS block

The 3D TRITON depletion sequences which use Shift in CE mode for the transport calculations (T5-DEPL-Shift, T6-DEPL-SHIFT) provide the capability to generate lattice-physics data for nodal core calculations based on complex 3D geometries. The use of CE data avoids any approximations due to MG cross section processing and it allows the generation of nodal data in arbitrary group structures.

The generation of nodal data in a TRITON-Shift input is requested through the *FGXS* block. The general input format of this block is as follows:

```
read fgxs
    energy       id=<NUM> <E0> <E1> ... <EN> <EN+1> end
    <tallytype>  id=<NUM> [options]
    shape <type> id=<NUM> [dimensions]
  [ mesh <type>  id=<NUM> [definition] ]
end fgxs
```

- **id**: Integer identifier of the requested nodal data set. Multiple sets of data can be requested, and the id allows the association of parameters to one specific set.
- **energy**: Energy group boundaries [eV] in increasing order, including the uppermost and lowermost boundaries. For an N-group structure, N+1 boundaries need to be provided.
- **<tallytype>**: The type of requested tally. The only supported type is currently *tallyset t16* to request the full set of nodal data.
- **shape**: Either the name of one of the permitted cell shapes [*cuboid*, *rhexprism*] or keyword *global* to request data on a mesh over the entire geometry. The use of *global* requires a *mesh* record. All dimensions are provided in cm.

- **mesh**: Gobal mesh definition in case of a *global* shape. See details below.

Global energy and tallytype definition

If multiple sets of nodal are requested within one *FGXS* block, and if the same type of tally in the same energy group structure is desired for *all* tallies, a global *energy* definition and the *tallytype* can be defined. The energy and tallytype id can be set to 0 (*id=0*) and is then valid for all other defined tallies. The energy and tallytype definition can then be skipped for the individual tallies (see Example 5 below).

Shapes

The supported shapes follow the following input syntax:

1. **cuboid**:

```
shape cuboid id=<NUM> <xmax> <xmin> <ymax> <ymin> <zmax> <zmin>
```

2. **rhexprism**:

```
shape rhexprism id=<NUM> hpitch=<hpitch> <zmax> <zmin> origin x=<x0> y=<y0>
```

3. **global**:

```
shape global id=<NUM>
```

with the following parameters:

- *<xmax>, <xmin>, <ymax>, <ymin>, <zmax>, <zmin>*: minimum and maximum x-, y-, or z-coordinates [cm]
- *hpitch=<p>*: half-pitch [cm] in case of the hexagon
- *origin x=<x0> y=<y0>*: coordinates in the global unit coordinate system for the center of the defined shape [cm]

Meshes

Global meshes are superimposed over the entire geometry. The half pitches of the mesh needs to be provided, the origin of the mesh (the origin of the central mesh cell), and the axial discretization.

At this point, only square lattices and **rotated** hexagonal lattices are supported:

1. **square**:

```
mesh square id=<NUM> hpitch=<p> origin x=<x0> y=<y0> dz <z1> ... <zM> end
```

2. **hexagonal**

```
mesh hexagonal id=<NUM> hpitch=<hpitch> origin x=<x0> y=<y0> dz <z1> ... <zM> end
```

with the following parameters:

- *hpitch=<p>*: half-pitch of the lattice [cm]
- *origin x=<x0> y=<y0>*: <x0> and <y0> are coordinates in the global unit coordinate system in which to place the center of a lattice element [cm]. Lattice elements are repeated in the negative-x, negative-y, positive-x, and positive-y directions to fill the global unit.

- *dz <z1> ... <zM>*: definition of the relative axial mesh widths. Each <*zi*> value must be positive, and the sum of all <*zi*> values must be 1.0. M entries in the dz-list will create M axial zones. The axial zone boundaries is determined by the relative mesh widths and the bottom and top axial boundaries of the global unit.

Output

On overview of the requested nodal data is provided in the output file:

```
================================================================================================
Nodal FGXS Tally Summary
================================================================================================

  Tally ID=30
    tally spatial type: MESH
    tally boundary: RHEXPRISM
    energy grid: 2e+07,0.625,1e-05
    volume mesh inscribed radius: 0.500000
    volume mesh center: -0.288675,-0.5
    volume mesh planes z: -1e-06,0.9,1
    volume tallies: absorption,fission,flux,kappa_sigma,nu_delayed_fission,nu_fission,transfer_1n,transfer_2n,
↪transfer_3n,transfer_4n
    volume tallies on isotopes (nuclide id/reaction mt): 1001/18, ... , 96245/1018
    fine energy grid: 2e+07,... , 1e-05
    fine energy grid volume tallies: flux,absorption,fission,transfer_1n,transfer_2n,transfer_3n,transfer_4n
    volume tally name (shift): mesh_nodal_tally_t16_30
    num tally energy groups: 2
    num tally mesh rings: 2
    num tally fine energy groups: 1000

  Tally ID=40
   ...
```

The t16 output file names are composed of the ${BASENAME} of the input file, the tally id, the mesh ids in case of global meshes, and they have the ending t16.

Example 1:

The following block requests nodal data in a cuboid within the interval *x=[2, -5]*, *y=[8, -5]*, *z=[2.5, 0]*.

```
read fgxs
  energy          id=1 1e-5 0.625 20E6 end
  tallyset t16    id=1
  shape cuboid    id=1 2 -5 8 -5 2.5 0
end fgxs
```

One output file with name *$ {BASENAME}.id1.t16* will be generated.

Example 2:

The following block requests nodal data in a rotated hexagon with half pitch of 1.25 cm, in the axial zone *z=[2,-2]*. The center of the hexagon is located at *(x,y,z) = (10.0, 0.5, 0.0)*.

```
read fgxs
  energy          id=2 1e-5 0.625 20E6 end
  tallyset t16    id=2
  shape rhexprism id=2 1.25 2.0 -2.0 origin x=10 y=0.5
end fgxs
```

One output file with name *$ {BASENAME}.id2.t16* will be generated.

Example 3:

The following block requests nodal data in a square lattice that stretches over the entire geometry. The half pitch of the square lattice is 0.8 cm and the origin is at (x,y) = (0,0). Nodal data is requested for 2 axial zones of equal height.

```
read fgxs
  energy          id=3 1e-10 0.625 20e6 end
  tallyset t16    id=3
  shape global    id=3
  mesh square     id=3 hpitch=0.8 origin x=0 y=0 dz 0.5 0.5 end
end fgxs
```

Multiple files named *$ {BASENAME}.id3-<x>-<y>-<z>.t16* will be generated.

Example 4:

The following block requests nodal data in a rotated hexagonal lattice that stretches over the entire geometry. The half pitch of the hexagonal lattice is 0.5 cm and the origin is at (x,y) = (-0.288675,-0.5). Nodal data is requested in two axial zones, the lower zone stretching over 90% of the geometry, the upper zone covering the upper 10% of the geometry.

```
read fgxs
  energy         id=4  1E-5 0.625 20E6 end
  tallyset t16   id=4
  shape global   id=4
  mesh hexagonal id=4 hpitch=0.5 origin x=-0.288675 y=-0.5 dz 0.9 0.1 end
end fgxs
```

Multiple files named *$ {BASENAME}.id3-<ii>-<jj>-<kk>.t16* will be generated. The output files contains an overview of all file names with the corresponding (x,y)-origin of the cells and the axial range. A visual representation of the mesh superimposed on the geometry indicates the centrl cell and the cells for which nodal data was generated.

```
T16 file summary for hex mesh tally id=4
=========================================

Apothem (inner radius):        0.5 cm
Number of rings: 2
Grid size: 5

basename.id4-<II>-<JJ>-<KK>.t16              x              y           zmin           zmax
-------------------------------------------------------------------------------------------------------
      basename.id30-02-04-01.t16   -1.15470e+00    0.00000e+00   -1.00000e-06    9.00000e-01
      basename.id30-02-04-02.t16   -1.15470e+00    0.00000e+00    9.00000e-01    1.00000e+00
      basename.id30-03-03-01.t16   -2.88675e-01   -5.00000e-01   -1.00000e-06    9.00000e-01
      basename.id30-03-03-02.t16   -2.88675e-01   -5.00000e-01    9.00000e-01    1.00000e+00
      basename.id30-03-04-01.t16   -2.88675e-01    5.00000e-01   -1.00000e-06    9.00000e-01
      basename.id30-03-04-02.t16   -2.88675e-01    5.00000e-01    9.00000e-01    1.00000e+00
      basename.id30-04-02-01.t16    5.77350e-01   -1.00000e+00   -1.00000e-06    9.00000e-01
      basename.id30-04-02-02.t16    5.77350e-01   -1.00000e+00    9.00000e-01    1.00000e+00
      basename.id30-04-03-01.t16    5.77350e-01    0.00000e+00   -1.00000e-06    9.00000e-01
      basename.id30-04-03-02.t16    5.77350e-01    0.00000e+00    9.00000e-01    1.00000e+00
      basename.id30-04-04-01.t16    5.77350e-01    1.00000e+00   -1.00000e-06    9.00000e-01
      basename.id30-04-04-02.t16    5.77350e-01    1.00000e+00    9.00000e-01    1.00000e+00


                          _____
                         /     \
                    _____/ 05    \
                   /     \ 05    /
              _____/ 04    \_____/
             /     \ 05    /     \
        _____/ 03    \_____/ 05    \
```

(continues on next page)

(continued from previous page)

```
        /     \ 05    /     \ 04    /
  _____/ 02    \_____/*04****\_____/
 /     \ 05    /     \*04****/     \
/ 01    \_____/*03****\_____/ 05    \
\ 05    /     \*04****/     \ 03    /
 \_____/*02****\_____/*04****\_____/
 /     \*04****/     \*03****/     \
/ 01    \_____/#03####\_____/ 05    \
\ 04    /     \#03####/     \ 02    /
 \_____/ 02    \_____/*04****\_____/
 /     \ 03    /     \*02****/     \
/ 01    \_____/ 03    \_____/ 05    \
\ 03    /     \ 02    /     \ 01    /
 \_____/ 02    \_____/ 04    \_____/
 /     \ 02    /     \ 01    /
/ 01    \_____/ 03    \_____/
\ 02    /     \ 01    /
 \_____/ 02    \_____/
 /     \ 01    /
/ 01    \_____/
\ 01    /
 \_____/

#### mesh origin cell
**** tally cells
<II> cell column/x-index
<JJ> cell row/y-index
```

Example 5:

The following block requests three sets of nodal data for different cuboids. The energy group definition and the tallyset are only defined once and apply to all defined tallies.

```
read fgxs
  energy       id=0 1e-5 0.625 20E6 end
  tallyset t16 id=0
  shape cuboid id=10 1.0 -1.0 5.0 -5.0 1.0 0.1
  shape cuboid id=20 2.0 -2.0 6.0 -6.0 2.5 1.2
  shape cuboid id=30 3.0 -3.0 7.0 -7.0 5.0 2.0
end fgxs
```

TALLIES and DEFINITIONS block

Since SCALE 6.3, a *TALLIES* input block and a *DEFINITIONS* input block can be used with CSAS-Shift and TRITON-Shift for flexible definition and output control of mesh tallies. The mesh responses for neutron flux, the neutrons produced from fission, and the fission rate (new since 6.3) can be requested on different spatial and energy grids. The syntax is very similar to the corresponding blocks used in the MAVRIC sequence.

The input parameters previously used to request these responses (i.e., *gfx*, *cds*, and *fis*) are still permitted in SCALE 6.3, but it is recommended to use the new input. The new input further permits the mesh tally responses based on different spatial grids and energy grids within one calculation, whereas previously only one spatial and energy grid was permitted per calculation.

DEFINITIONS BLOCK

The new definitions input block allows multiple spatial grids to be defined using the *gridGeometry* keyword and multiple energy grids to be defined using the *energyBounds* keyword. The syntax for defining a *gridGeometry* inside a definitions block is the same as defining a standalone grid at the root level of input (see Sect. 8.1.3.14). The syntax for defining *energyBounds* is already used for defining energy grids in the MAVRIC sequence (see Table 8.2.12).

In addition to specifying a list of individual energy group boundaries, equal-width energy bins and equal-width lethargy bins can also be requested. In continuous-energy mode, a special *default* keyword allows modification of the default energy group structure previously defined with the *NGP* parameter and/or the standalone energy block. For example, TRITON-Shift can use these *energyBounds* to tally flux and cross sections for depletion calculations.

The following examples demonstrate how the *DEFINITIONS* block in TRITON is used.

```
read definitions

  read grid 1
      xlinear 30 -10 70
      ylinear 10 -20 60
      zlinear 50 -30 40
  end grid

  read grid 2
      numxcells=10 xmin=-18.5 xmax=+68.5
      numycells=25 ymin=-28.5 ymax=+58.5
      numzcells=10 zmin=-38.5 zmax=+48.5
  end grid

' global energy grid, default is 252g in CE mode, but can be modified here
  energyBounds default
      bounds 2e7 0.625 1e-5 end
  end energyBounds

' user specified energy grid
  energyBounds 1
      bounds 2e7 0.625 1e-5 end
  end energyBounds

' user specified energy grid using equal-energy bins
  energyBounds 2
      linear=10 1e-5 2e7
  end energyBounds

' user specified energy grid using equal-lethargy bins
  energyBounds 3
      logarithmic=10 1e-5 2e7
  end energyBounds

end definitions
```

TALLIES BLOCK

The new tallies input block allows mesh responses to be requested using any energy grid and/or spatial grid from the definitions block. TRITON-Shift also allows specifying a time input array to enable or disable these mesh tally responses for specific depletion steps. The last depletion step may be conveniently requested using the special *LAST* keyword, and the special *ALL* keyword may be used to request the tally for every depletion step.

Description	SCALE 6.2 parameter name	SCALE 6.3 parameter name
neutron flux	GFX	flux
fission rate	FIS	fission_density
neutrons produced from fission	CDS	fission_source

The following examples demonstrate how the *TALLIES* block in TRITON is used.

```
read tallies

  read mesh 1
      response = FLUX
      grid     = 1
      time       0 LAST end
      energy   = 1
  end mesh

  read mesh 2
      response = FISSION_DENSITY
      grid     = 2
      time       0 1 2 end
      energy   = 2
  end mesh

  read mesh 3
      response = FISSION_SOURCE
      grid     = 3
      time       ALL end
      energy   = default
  end mesh

end tallies
```

3.1.3.4 *ALIAS* block

The optional *ALIAS* block may be used to simplify model development within TRITON by defining a set of material numbers that will be inserted in place of the alias when that alias is used in subsequent data blocks. Aliases function as variables for which a user-defined set of materials are inserted; they are identified by a dollar character ($) preceding a single-word alphanumeric label. The *ALIAS* block is used to preprocess an input, creating a new, modified input deck with all alias variable substitutions included. TRITON then processes the modified input deck before proceeding with the calculation.

The use of an alias variable is best illustrated by a brief example. Assume that the alias `$fuel` is defined as materials 1, 2, and 3, and `$mod` as materials 4, 5, and 6. (The input format for defining aliases is described below.) The user wishes to create three identical sets of materials and use them in three identical pin cell specifications. In the *COMPOSITION* data block, specifications could be written in the following form

```
h2o $mod den=0.6616 1.0 595 end
wtpt-boron $mod 0.6616 1 5000 100 655e-6 595 end
```

TRITON would create a modified input with the alias expanded as follows:

```
uo2  1 den=10.29 0.9322 920 92235 3.0 92238 97.0 end
uo2  2 den=10.29 0.9322 920 92235 3.0 92238 97.0 end
uo2  3 den=10.29 0.9322 920 92235 3.0 92238 97.0 end
h2o 4 den=0.6616 1.0 595 end
h2o 5 den=0.6616 1.0 595 end
h2o 6 den=0.6616 1.0 595 end
wtpt-boron  4 0.6616 1 5000 100 655e-6 595 end
wtpt-boron  5 0.6616 1 5000 100 655e-6 595 end
wtpt-boron  6 0.6616 1 5000 100 655e-6 595 end
```

Similarly, if the alias were used in the *CELLDATA* block as

```
latticecell squarepitch pitch=1.26 $mod fuelr=0.4095 $fuel end
```

then TRITON would expand the aliases to

```
latticecell squarepitch pitch=1.26 4 fuelr=0.4095 1 end
latticecell squarepitch pitch=1.26 5 fuelr=0.4095 2 end
latticecell squarepitch pitch=1.26 6 fuelr=0.4095 3 end
```

In a depletion calculation, one may wish to deplete a large number of fuel rods independently because of different geometric locations in a fuel assembly. Even though each fuel rod may have the same initial composition, each must be specified as a unique material composition in order to be depleted independently. Furthermore, multiple cell specifications must all use unique material identifiers for each cell component. Thus, if one desired to deplete 25 fuel materials in a fuel/clad/moderator pin cell, one would need to set up material composition definitions for 25 fuels, 25 moderators, and 25 clads. Then one would need to provide 25 pin cell specifications. By using aliases, one need only specify the material identifiers corresponding to each alias and then provide only one material composition specification for each alias type, and then one pin cell specification. TRITON will automatically expand the aliases and create a revised input with all materials and cell specifications explicitly defined.

Note: Note that although this will simplify the pin cell input in the CELLDATA, 25 pin cell calculations would still be required. The number of pin cell calculations can be reduced by using the ASSIGN function described in Sect. 3.1.3.3.7.

The purpose of the *ALIAS* block is to define a set of alias variables to be used in subsequent data blocks. The *ALIAS* block is optional, but aliases may not be used in other blocks if an *ALIAS* block is not present to define the aliases. An *ALIAS* block may contain as many aliases as desired. Each alias specification consists of three parts: the alias name, consisting of a dollar sign followed by up to 11 alphanumeric characters with no embedded spaces; the material number or numbers; and an *end* keyword. Material numbers may be entered in any order and may be separated by spaces or commas (or both). Material numbers may also be separated by a dash (-), but this represents an inclusive list. In other words, a material specification of 1-3 (or 1 - 1) indicates materials 1, 2, and 3. The example *ALIAS* block below illustrates the various means for assigning a set of materials for an alias definition.

```
read alias
$fueltype1 1 2 3 end
$fueltype2 4,5,6, 31-33 end
$clad1 21,22,23 end
$clad2 24 25 26 34-36 end
$mod1 11    -   13 end
$mod2 14-16, 37-39 end
end alias
```

The *ALIAS* block simply serves to assign material identifiers to specific variables, and the variables are used in subsequent data blocks. The same material identifier can be used in more than one alias if desired. As indicated earlier, TRITON will preprocess any input deck containing an *ALIAS* block and replace instances of alias variables with the appropriate material identifiers. The following subsections describe how aliases are implemented in TRITON's various input blocks, as the form of alias variable substitution is block dependent. Aliases are processed only in these input blocks; aliases used in other blocks will result in an error.

COMPOSITION *block aliases*

The *COMPOSITION* block uses aliases to create multiple copies of each standard composition specification, replacing the alias variable with each material identifier associated with the alias definition. For example, consider the following alias definition in an *ALIAS* block:

```
read alias
$fuel 1 2 10 end
end alias
```

and the standard composition specification:

```
uo2 $fuel den=10.045 1 800 92235 2.5 92238 97.5 end
```

A modified TRITON input would be created with the standard composition specification replaced by

```
uo2  1 den=10.045 1 800 92235 2.5 92238 97.5 end
uo2  2 den=10.045 1 800 92235 2.5 92238 97.5 end
uo2 10 den=10.045 1 800 92235 2.5 92238 97.5 end
```

CELLDATA *block aliases*

CELLDATA block *latticecell* specifications typically contain more than one material; therefore, multiple aliases are permitted in each cell specification. However, this constrains the set of aliases used in the cell specification to have the same number of material identifiers associated with it.

Consider the *ALIAS* block:

```
read alias
$fuel 1-3 10 end
$clad 4,5,6,11 end
$mod 7 8-9 12 end
end alias
```

All three aliases contain four materials each. One could then create a single cell specification that uses one or more of these alias variables, such as

```
latticecell squarepitch pitch=1.26 $mod fuelr=0.41 $fuel cladr=0.50 $clad end
```

This would result in the following alias expansion by TRITON:

```
latticecell squarepitch pitch=1.26  7 fuelr=0.41  1 cladr=0.50  4 end
latticecell squarepitch pitch=1.26  8 fuelr=0.41  2 cladr=0.50  5 end
latticecell squarepitch pitch=1.26  9 fuelr=0.41  3 cladr=0.50  6 end
latticecell squarepitch pitch=1.26 12 fuelr=0.41 10 cladr=0.50 11 end
```

Material identifiers are substituted according to their position in the alias definition (i.e., the first substitution will use the first material associated with each alias, and the second expansion will use the second material associated with each alias, etc.)

Material numbers should not be entered manually in a cell specification; for example,

```
latticecell triangpitch pitch=1.26 $mod fuelr=0.4095 1 end
```

TRITON would allow this to occur and would create a set of cell specifications as follows:

```
latticecell triangpitch pitch=1.26 2 fuelr=0.4095 1 end
latticecell triangpitch pitch=1.26 3 fuelr=0.4095 1 end
```

where $mod was defined as materials 2 and 3. However, SCALE does not allow the same material identifier to occur in two different cell specifications, and the fact that material 1 occurs in two different cell specifications would result in TRITON ending with an error. *Note that alias expansions for* **multiregion** *and* **doublehet** *cell specifications are not supported. Also note that TRITON will not copy* **centrmdata** *and* **moredata** *specifications that follow a cell specification that uses an alias variable.*

DEPLETION *block aliases*

Aliases in the TRITON *DEPLETION* are simply replaced by the set of materials associated with the alias. For example, the *ALIAS* block

```
read alias
$fuel 1 2 10 end
end alias
```

and DEPLETION block

```
read depletion 7 8 9 $fuel end depletion
```

would be expanded to

```
read depletion 7 8 9 1 2 10 end depletion
```

Aliases may be mixed with actual material numbers in the depletion block, along with the flux and assign keywords. *However, the negative sign-used to define the basis for power normalization-cannot precede an alias definition.*

TIMETABLE *block aliases*

TIMETABLE block alias expansion is similar to that of the *COMPOSITION* block: TRITON will create a new timetable entry for each material associated with the alias used in the *TIMETABLE* definition. For the *TIMETABLE* block below, using the alias *$ allmod*, unique timetables will be created for each material identifier associated with this alias variable.

Note: Note that alias expansion of **density** timetable entries is not yet supported.

```
read timetable
temperature $allmod
0.0    615
121.0  615
121.01 685
322.5  685
352.5  610
738.75 610 end
end timetable
```

BRANCH *block aliases*

Aliases may be used within the *define* keyword definitions of the *BRANCH* block. Aliases are simply replaced by the list of materials associated with the alias, as is done for the *DEPLETIO*N block. Hence,

```
read alias
$fuel 1 2 10 end
end alias
```

used with

```
read branch
define fuel $fuel end
md=0.75 tm=559 tf=880 sb=0.0 cr=0 end
tf=1600 end
end branch
```

would be expanded to

```
read branch
define fuel 1 2 10 end
md=0.75 tm=559 tf=880 sb=0.0 cr=0 end
tf=1600 end
end branch
```

NEWT MATERIAL *block aliases*

The *MATERIAL* block within the NEWT model section of a TRITON input can also use aliases. As with *COMPOSITION* and *TIMETABLE* entries, TRITON will create a new material specification for each material represented by an alias. For the sample material block below, using the alias *$ fuel*, unique material block entries will be created for each material associated with the alias variable.

```
read materials
mix=$fuel pn=1  com="3.25 wo uo2 fuel" end
mix=21    pn=1  com="zirc cladding"    end
mix=31    pn=1  com="water"            end
end materials
```

If an alias were defined as

```
$fuel 10 11 12 end
```

then the *MATERIAL* block would be expanded to

```
read materials
mix=10 pn=1  com="3.25 wo uo2 fuel" end
mix=11 pn=1  com="3.25 wo uo2 fuel" end
mix=12 pn=1  com="3.25 wo uo2 fuel" end
mix=21    pn=1  com="zirc cladding"    end
mix=31    pn=1  com="water"            end
end materials
```

KEEP_OUTPUT *block*

When performing depletion calculations for a number of different materials, TRITON output can become quite voluminous. Often, much of that output is not needed for calculations that seek only eigenvalues, sources, or concentrations as a function of irradiation history. TRITON provides the ability to trim output to only those portions for which output is desired. Output produced directly by the TRITON module is always provided and cannot be disabled, but output from any other code in the sequence can be automatically removed from the output listing. Retaining certain output is accomplished using the *KEEP_OUTPUT* data block.

The *KEEP_OUTPUT* data block provides the ability to preserve only selected outputs. The format of this data block is

```
read keep_output
module_1 module_1 ... module_i ... module_N
end keep_output
```

where `module_i` represents any valid module name from the list of modules invoked by TRITON, as listed here:

xsproc xsdrn newt kenova kenovi couple origen

Without the *KEEP_OUTPUT* data block, the output from the neutron transport kernel (xsdrn, newt, kenova, kenovi) is retained and the output of all other modules (xsproc, couple, origen) is suppressed. Note that the output of SAMS and OPUS is not controlled by this block; the output of these modules is always retained.

If a *KEEP_OUTPUT* data block is included, then only the output of the specified modules is retained. If the output of the neutron transport kernel is desired, then the corresponding module has to be listed since the above mentioned default is no longer applicable. A *KEEP_OUTPUT* data block without any module name can be specified to suppress the output of all modules.

When using the TRITON-Shift sequence, the generation of Shift's HDF5 output files can be controlled through a time array in the *KEEP_OUTPUT* output. The special keywords "ALL" for all depletion steps and "LAST" for the last depletion step are supported.

Examples:

1. Only the output of the neutron transport kernel is retained:

The input does not contain a KEEP_OUTPUT data block.

2. Only the XSProc output is retained; the output of the neutron transport kernel is suppressed:

```
read keep_output
  xsproc
end keep_output
```

3. The output of both XSProc and the neutron transport kernel KENO-VI is retained:

```
read keep_output
  xsproc kenovi
end keep_output
```

4. The output of all modules is suppressed:

```
read keep_output
end keep_output
```

5. Shift's HDF5 output files are requested for the first, second and last depletion step:

```
read keep_output
  shift 0 1 LAST end
end keep_output
```

3.1.3.5 TRITON control parameters

TRITON supports the following of control parameter options:

parm= CHECK, CENTRM, 2REGION, XSLEVEL=N, WEIGHT, WEIGHT=N, ADDNUX=N, INFDCUTOFF=X, CXM=N, MAXDAYS=N

If an invalid control parameter option is specified, including misspelled keywords, an error message will be generated and execution terminated. TRITON also provides the ability to nest several control parameter keywords together; to combine keywords (where appropriate), a list may be entered, enclosed in parentheses, and separated by commas. For example, to specify CHECK, 2REGION, and ADDNUX=1 at the same time, input would begin with

```
=t-depl  parm=(check, 2region,addnux=1)
```

The following subsections provide more detail on each of the control parameters listed above.

Check mode: parm=check

Specification of *parm=check* will request that TRITON read all input and ensure that no input errors are present, without running additional calculations. In this mode, all input is set up as if a full calculation will be run, but the sequence exits without any functional module execution. The check mode is useful for debugging or obtaining processed standard composition data, without actually running a calculation. It can also be used to generate plot files for embedded NEWT and KENO inputs for additional review and checking of input specifications. Of course, some errors may be uncovered only by dynamically executing the functional modules; hence, there are rare occasions where a *parm=check* run will complete with no errors but will fail when run outside of check mode as the problem begins to run.

Multigroup cross section processing options

The most common use of *parm=* sequence control is in the selection of an alternate multigroup cross section processing mode.

By default, XSProc enables both the BONAMI and CENTRM modules for cross section processing. BONAMI-only XSProc calculations can be performed using the control parameter *parm=bonami*.

TRITON also supports the control parameter *parm=(xslevel=N)*. The *xslevel* parameter option initializes various CENTRM options for the XSProc calculations. The *xslevel* option is equivalent to initializing all unit cell calculations with the following *centrmdata* specifications:

```
parm=(xslevel=1):
centrmdata
  npxs=5 nfst=0 nthr=3 nmf6=-1 alump=0.3 demin=0.125 pmc_omit=1 pmc_dilute=5.0e5
end centrmdata
parm=(xslevel=2):
  centrmdata npxs=5 nfst=0 nthr=3 nmf6=-1 end centrmdata
```

(continues on next page)

(continued from previous page)

```
parm=(xslevel=3):
   centrmdata alump=0.3 demin=0.125 pmc_omit=1 pmc_dilute=5.0e5 end centrmdata
parm=(xslevel=4):
   [no centrmdata statement]
```

The option *parm=(xslevel=4)* is equivalent to *parm=centrm*. The option *parm=(xslevel=3)* is the default for depletion sequences and is equivalent to *parm=centrm* but with some minor approximations to decrease run time. The option *parm=(xslevel=2)* is equivalent to *parm=2region* for all sequences.

Note that the *xslevel=1* and *xslevel=3* options have additional specifications for keywords *alump*, *demin*, *pmc_omit*, and *pmc_dilute*. These keywords are further discussed in the XSProc chapter. The additional keyword specifications are used to decrease run-time for the CENTRM and PMC calculations. Internal investigations have shown that the approximations introduced by the additional keyword specifications have minimal impact on solution accuracy for a wide range of calculations. Therefore the additional keyword specifications are used by default for depletion calculations, where several CENTRM and PMC calculations are necessary. The additional keyword values are not used by default for nondepletion calculations to be consistent with the SCALE CSAS5 and CSAS6 criticality sequences.

***Creating a broad group library:* parm=weight, parm=(weight=N)**

Used in tandem with the TRITON T-NEWT sequence, the specification *parm=weight* extends the sequence by setting up and executing the MALOCS2 module to generate a weighted broad-group cross-library (AMPX master format). The spectrum generated in the NEWT transport calculation is used as the weighting function for the collapse. Additionally, the broad-group library energy structure is defined by the NEWT *COLLASPE* block.

The *parm=weight* option uses the problem-averaged flux spectrum for the weighting function in the collapse. The problem may be a simple pin cell or a full assembly. However, there may be cases where the flux in a specific region or material is most appropriate for the spectral collapse. TRITON allows identification of a specific material from which the collapsing spectrum should be used. When specified in the form *parm=(weight=N)*, the average flux determined for material N is used in place of the total domain spectrum to perform the collapse.

TRITON sample problem 1 (Sect. 3.1.6.1) provides an example of the use of T-NEWT to produce a new broad-group library. Note that the broad-group library produced in this calculation will reside in the SCALE temporary working directory with the name *newxnlib* at the end of the calculation. If the library will be needed for future calculations, the user should use a shell script to copy the library back to a more permanent location, and perhaps give it a more meaningful name. In sample problem 1, the SCALE 252-group master library is collapsed to 56 energy groups.

The process for creating a broad-group master library is also supported in the 2D depletion sequence T-DEPL. When *parm=weight* or *parm=(weight=N)* is specified in a depletion calculation, the input cross section library must be one of the SCALE 238-group or 252-group libraries, which will automatically be collapsed to the SCALE 49-group or 56-group structure, respectively. An initial fine group calculation is performed for the input configuration, and the flux from the solution is used to create the broad group library. The initial calculation is then repeated with the new broad group library, followed by the remainder of the depletion calculation. *Note that for lattice physics calculations, the NEWT* COLLAPSE *block will be based on the 49-group (or 56-group) energy structure, not the fine group structure.*

It is important to note that the 252-group library contains intermediate resonance parameters and other data that cannot be accurately collapsed into 56-group data with the collapsing procedures available in MALOCS2.

These parameters are important for bonami-only cross section processing calculations, i.e., *parm=bonami*. Therefore, the *parm=centrm* option is recommended for follow-on application of the collapsed 56-group collapsed library. The 238-group and 49-group libraries do not contain intermediate resonance parameter data, and bonami-only processing is available, provided that this cross section processing option and group structure is suitable for the intended application.

Inclusion of additional nuclides for depletion: **parm=(addnux=N)**

For depletion calculations, it is important to add trace quantities (1×10^{-20} at/b-cm) of certain nuclides to the inventories of depletion materials in order to accurately track the nuclides' impact on cross section processing and transport calculations as a function of burnup. By default, TRITON automatically adds to all fuel materials trace quantities of a set of nuclides that have been determined to be important in the characterization of spent fuel. TRITON recognizes fuel materials as any material containing quantities of heavy metals (Z > 89) in the standard composition specification.

TRITON provides user control of the set of nuclides added to a fuel material through the *parm=(addnux=N)* control parameter, where N is an integer value. For N = 0, no nuclides are added, which is generally a very poor approximation and should only be used when the ramifications are fully understood. For N = 1, a bare minimum set of 15 nuclides (actinides) are added; this will generate improved number density estimates for actinides in low-burnup fuels but will not update cross sections for fission products of primary importance. Again, use of this option is discouraged unless it addresses special modeling needs. For N = 2, the default setting for the TRITON depletion sequences, 95 nuclides are added. N = 3 and N = 4 add 231 and 388 nuclides, respectively. Note that in previous versions of TRITON, N = 2 would add 64 nuclides. The set of 64 nuclides is still supported by specifiying *parm=(addnux=-2)* in the input. The default in the SCALE 6.1 release remains *parm=(addnux=2).* Table 3.1.2 through Table 3.1.6 list the set of nuclides added in trace quantities for each value of *addnux*.

Table 3.1.2: Additional nuclides added in trace quantities for *parm=(addnux=1)*

	^{234}U	^{235}U	^{236}U
^{238}U	^{237}Np	^{238}Pu	^{239}Pu
^{240}Pu	^{241}Pu	^{242}Pu	^{241}Am
^{242}Am	^{243}Am	^{242}Cm	^{243}Cm
*15 nuclides total.			

Table 3.1.3: Additional nuclides added in trace quantities for *parm=(addnux= -2)*

^{1}H	^{10}B	^{11}B	
^{14}N	^{16}O	^{83}Kr	^{93}Nb
^{94}Zr	^{95}Mo	^{99}Tc	^{103}Rh
^{105}Rh	^{106}Ru	^{109}Ag	^{126}Sn
^{135}I	^{131}Xe	^{135}Xe	^{133}Cs
^{134}Cs	^{135}Cs	^{137}Cs	^{143}Pr
^{144}Ce	^{143}Nd	^{145}Nd	^{146}Nd
^{147}Nd	^{147}Pm	^{148}Pm	^{149}Pm
^{148}Nd	^{147}Sm	^{149}Sm	^{150}Sm

continues on next page

Table 3.1.3 – continued from previous page

^{151}Sm	^{152}Sm	^{151}Eu	^{153}Eu
^{154}Eu	^{155}Eu	^{152}Gd	^{154}Gd
^{155}Gd	^{156}Gd	^{157}Gd	^{158}Gd
^{160}Gd	^{244}Cm		
*49 additional nuclides in addition to the 15 nuclides added in addnux=1, for a total of 64.			

Table 3.1.4: Additional nuclides added in trace quantities for *parm=(addnux=2)*

^{91}Zr	^{93}Zr	^{95}Zr	^{96}Zr
^{95}Nb	^{97}Mo	^{98}Mo	^{99}Mo
^{100}Mo	^{101}Ru	^{102}Ru	^{103}Ru
^{104}Ru	^{105}Pd	^{107}Pd	^{108}Pd
^{113}Cd	^{115}In	^{127}I	^{129}I
^{133}Xe	^{139}La	^{140}Ba	^{141}Ce
^{142}Ce	^{143}Ce	^{141}Pr	^{144}Nd
^{153}Sm	^{156}Eu	^{242m}Am	
*31 additional nuclides in addition to the 15 nuclides in Table 3.1.2 and 49 nuclides in Table 3.1.3, for a total of 95.			

Table 3.1.5: Additional nuclides added in trace quantities for *parm=(addnux=3)*

^{72}Ge	^{73}Ge	^{74}Ge	^{76}Ge
^{75}As	^{79}Br	^{76}Se	^{77}Se
^{78}Se	^{80}Se	^{82}Se	^{81}Br
^{80}Kr	^{82}Kr	^{84}Kr	^{85}Kr
^{86}Kr	^{85}Rb	^{86}Rb	^{87}Rb
^{84}Sr	^{86}Sr	^{87}Sr	^{88}Sr
^{89}Sr	^{90}Sr	^{89}Y	^{90}Y
^{91}Y	^{90}Zr	^{92}Zr	^{92}Mo
^{94}Mo	^{96}Mo	^{94}Nb	^{96}Ru
^{98}Ru	^{99}Ru	^{100}Ru	^{105}Ru
^{102}Pd	^{104}Pd	^{106}Pd	^{110}Pd
^{107}Ag	^{111}Ag	^{106}Cd	^{108}Cd
^{110}Cd	^{111}Cd	^{112}Cd	^{114}Cd
^{115m}Cd	^{116}Cd	^{140}Ce	^{113}In
^{140}La	^{112}Sn	^{114}Sn	^{115}Sn
^{116}Sn	^{117}Sn	^{118}Sn	^{119}Sn
^{120}Sn	^{122}Sn	^{123}Sn	^{124}Sn
^{125}Sn	^{121}Sb	^{123}Sb	^{124}Sb
^{125}Sb	^{126}Sb	^{120}Te	^{122}Te
^{123}Te	^{124}Te	^{125}Te	^{126}Te
^{127m}Te	^{128}Te	^{129m}Te	^{130}Te
^{132}Te	^{130}I	^{131}I	^{124}Xe

continues on next page

Table 3.1.5 – continued from previous page

^{126}Xe	^{128}Xe	^{129}Xe	^{130}Xe
^{132}Xe	^{134}Xe	^{136}Xe	^{134}Ba
^{135}Ba	^{136}Ba	^{137}Ba	^{138}Ba
^{136}Cs	^{142}Pr	^{142}Nd	^{150}Nd
^{151}Pm	^{144}Sm	^{148}Sm	^{154}Sm
^{152}Eu	^{157}Eu	^{232}U	^{233}U
^{159}Tb	^{160}Tb	^{160}Dy	^{161}Dy
^{162}Dy	^{163}Dy	^{164}Dy	^{165}Ho
^{166}Er	^{167}Er	^{175}Lu	^{176}Lu
^{181}Ta	^{182}W	^{183}W	^{184}W
^{186}W	^{185}Re	^{187}Re	^{197}Au
^{231}Pa	^{233}Pa	^{230}Th	^{232}Th

*136 additional nuclides in addition to the 15 nuclides in Table 3.1.2, 49 nuclides in Table 3.1.3, and 31 nuclides in Table 3.1.4, for a total of 231.

Table 3.1.6: Additional nuclides added in trace quantities for *parm=(addnux=4)*

^{2}H	^{3}H	^{3}He	^{4}He
^{6}Li	^{7}Li	^{7}Be	^{9}Be
^{15}N	^{17}O	^{19}F	^{23}Na
^{24}Mg	^{25}Mg	^{26}Mg	^{27}Al
^{28}Si	^{29}Si	^{30}Si	^{31}P
^{32}S	^{33}S	^{34}S	^{36}S
^{35}Cl	^{37}Cl	^{36}Ar	^{38}Ar
^{40}Ar	^{39}K	^{40}K	^{41}K
^{40}Ca	^{42}Ca	^{43}Ca	^{44}Ca
^{46}Ca	^{48}Ca	^{45}Sc	^{46}Ti
^{47}Ti	^{48}Ti	^{49}Ti	^{50}Ti
^{50}Cr	^{52}Cr	^{53}Cr	^{54}Cr
^{55}Mn	^{54}Fe	^{56}Fe	^{57}Fe
^{58}Fe	^{58}Co	^{58m}Co	^{59}Co
^{58}Ni	^{59}Ni	^{60}Ni	^{61}Ni
^{62}Ni	^{64}Ni	^{63}Cu	^{65}Cu
^{70}Ge	^{69}Ga	^{71}Ga	^{74}As
^{74}Se	^{79}Se	^{78}Kr	^{110m}Ag
^{113}Sn	^{123}Xe	^{130}Ba	^{132}Ba
^{133}Ba	^{136}Ce	^{138}Ce	^{139}Ce
^{138}La	^{148m}Pm	^{153}Gd	^{156}Dy
^{158}Dy	^{166m}Ho	^{162}Er	^{164}Er
^{168}Er	^{170}Er	^{174}Hf	^{176}Hf
^{177}Hf	^{178}Hf	^{179}Hf	^{180}Hf
^{182}Ta	^{191}Ir	^{193}Ir	^{196}Hg
^{198}Hg	^{199}Hg	^{200}Hg	^{201}Hg
^{202}Hg	^{204}Hg	^{204}Pb	^{206}Pb

continues on next page

Table 3.1.6 – continued from previous page

^{207}Pb	^{208}Pb	^{209}Bi	^{223}Ra
^{224}Ra	^{225}Ra	^{225}Ac	^{226}Ac
^{227}Ac	^{226}Ra	^{227}Th	^{228}Th
^{229}Th	^{233}Th	^{234}Th	^{232}Pa
^{235}Np	^{236}Np	^{238}Np	^{239}Np
^{237}U	^{239}U	^{240}U	^{241}U
^{236}Pu	^{237}Pu	^{243}Pu	^{244}Pu
^{246}Pu	^{244}Am	^{244m}Am	^{241}Cm
^{245}Cm	^{246}Cm	^{247}Cm	^{248}Cm
^{249}Cm	^{250}Cm	^{249}Bk	^{250}Bk
^{249}Cf	^{250}Cf	^{251}Cf	^{252}Cf
^{253}Cf	^{254}Cf	^{253}Es	^{254}Es
^{255}Es			

*158 additional nuclides in addition to the 15 nuclides in Table 3.1.2, 49 nuclides in Table 3.1.3, 30 nuclides in Table 3.1.4, and 136 nuclides in Table 3.1.5, for a total of 388.

Few-group reaction cross section calculation control for continuous energy depletion: **parm=(cxm=N)**

In continuous energy depletion calculations, few group reaction cross sections are computed by KENO directly rather than using a post-processing approach that TRITON uses for multigroup mode. In addition to these region averaged multigroup reaction cross sections, KENO also provides problem-dependent region-averaged multigroup fluxes to TRITON that will be used by COUPLE to generate one-group cross section library for each depletion material.

Option *parm=(cxm=N)* is used to setup continuous-energy depletion calculation with different modes of calculation, which tells KENO the details of the tallying process for the reaction cross sections and mixture fluxes. Available calculations modes and their descriptions are presented in Table 3.1.7.

Table 3.1.7: *cxm* values and their descriptions.

cxm	**cross sections**		**flux**	**description**
	reactions	number of energy groups	number of energy groups	
1	All	NGP	NGP	KENO uses default NGP-group energy group boundaries to generate region-averaged reaction cross sections for all available reactions of the nuclides in each depletion mixture. KENO also computes region-averaged multigroup fluxes using the default NGP-group energy bins. multigroup fluxes using the default NGP-group energy bins.
2	Transmutation (MT=16-18, 102-125)	NGP	NGP	KENO uses default NGP-group energy group boundaries to generate region-averaged reaction cross sections for only transmutation reactions of the nuclides in each depletion mixture. KENO also computes region-averaged multigroup fluxes using the default NGP-group energy bins. multigroup fluxes using the default NGP-group energy bins.

continues on next page

Table 3.1.7 – continued from previous page

3	All	1	NGP	KENO uses 1-group energy group boundaries to generate region-averaged reaction cross sections for all available reactions of the nuclides in each depletion mixture. KENO also computes region-averaged multigroup fluxes using the default NGP-group energy bins. multigroup fluxes using the default NGP-group energy bins.
4 (default)	Transmutation (MT=16-18, 102-125)	1	NGP	KENO uses 1-group energy group boundaries to generate region-averaged reaction cross sections for only transmutation reactions of the nuclides in each depletion mixture. KENO also computes region-averaged multigroup fluxes using the default NGP-group energy bins. multigroup fluxes using the default NGP-group energy bins.

Note: The energy group structure in KENO and associated number of energy groups, NGP, should be consistent with those from the ORIGEN library used in the problem.

Infinite dilution cutoff control: **parm=(infdcutoff=X)**

The addition of nuclides to depletion materials as described in the previous section can lead to increased run-times for CENTRM-based XSProc calculations. However, many nuclides (e.g., low-density nuclides) are effectively infinitely dilute and can be treated as such and omitted from the expensive point-wise cross section collapse operation. For the option *parm=(infdcutoff=sigma0)* sequence option, XSProc will compute an effective background microscopic cross section for each nuclide. If the computed background cross section is greater than the cutoff value *sigma0*, recommended as 5×10^5 barns, then the nuclide is considered infinitely dilute and the infinitely dilute multigroup cross section is utilized from the cross section library.

In general, a *sigma0* cutoff value of 5×10^5 barns will be acceptable for most applications. However, TRITON and the centrmdata card in the *CELLDATA* block provide a means for the user to control the cutoff value. The cutoff value may be assigned in either of two ways. A single global value may be assigned to all cells using the TRITON *parm=* specifier with the keyword *infdcutoff*, for example, *parm=(infdcutoff=1e10)*. Addition of the specifier with a value of 1×10^{10} will set the cutoff value to 1×10^{10} for all cells in the problem, which is generally appropriate for most calculations. However, a provision is made to specify a unique cutoff value to each cell using the *pmc_dilute* keyword in a *centrmdata* specification. An example of this is shown in the description of *parm=xslevel* in Sect. 3.1.3.5.2.

The default value of *sigma0* depends on the sequence and cross section processing option. For nondepletion sequences that use *parm=centrm*, the default is 0. The default value of 0 instructs PMC to include all nuclides for PMC processing. For depletion sequences that use *parm=centrm* or for any sequence that uses *parm=2region*, the default value is 5×10^5 barns.

Override of the maximum number of days per depletion subinterval: **PARM=(MAXDAYS=N)**

TRITON is set to limit ORIGEN time intervals to no more than 40 days to avoid potential numerical error that would be introduced if depletion were performed over a long time interval. For depletion subintervals of more than 400 days (10 time intervals of 40 days), TRITON will automatically increase the number of depletion subintervals in a depletion interval. The depletion subinterval is based on a rule of thumb for ORIGEN depletion. However, the rule breaks down when burning at very low powers for extended time intervals. Thus, TRITON allows the user to override the default behavior by specifying a new value for the maximum number of days per ORIGEN time interval. A 100-day limit per ORIGEN time interval may be set using *parm=(maxdays=100)*. In overriding the default behavior, the user must be aware of any potential errors introduced in the approximation.

3.1.4 OUTPUT FILES CREATED BY TRITON

TRITON produces a variety of output files that may be of use in related calculations. Of those files, only certain files are copied back to the return directory: the TRITON output file (.out); plot files generated by NEWT, KENO, or OPUS (.plt); SAMS sensitivity data files (.sdf), in the case of an S/U calculation; ORIGEN binary concentration files (.f71) and HTML-formatted output (.html), where available. The TRITON output file is a concatenated listing of outputs from TRITON and all modules for which output is kept. Other files of potential interest are not copied, and the user should be aware of these files and their names so that they may be retrieved using a SHELL script after TRITON execution is complete. The following subsections list those files and their purposes.

3.1.4.1 Standard composition restart files

At the end of all depletion calculations, standard composition files are automatically produced for each material, listing the nuclides and number densities of the materials at the time the transport calculation (i.e., XSDRN, NEWT, KENO) is performed. Only nuclides for which cross section data are available in the master cross section library are saved in these files. Files are saved using the file naming convention StdCmpMix*NNNNN*, where *NNNNN* is the material identifier. The file contains compositions at the final time of the calculation. Additional files are saved with the file naming convention StdCmpMix*NNNNN_MMMMM*, where *MMMMM* is an index to a particular time step in the depletion calculation. For example, if a calculation is completed with materials 1 and 40 for two depletion steps, then the following files will be created in the temporary working directory.

```
StdCmpMix00001_00000 (t=0)
StdCmpMix00001_00001 (midpoint of 1st depletion step)
StdCmpMix00001_00002 (midpoint of 2nd depletion step)
StdCmpMix00001_00003 (final compositions, end of 2nd depletion step)
StdCmpMix00001 (same as StdCmpMix00001_00003)
StdCmpMix00040_00000 (t=0)
StdCmpMix00040_00001 (midpoint of 1st depletion step)
StdCmpMix00040_00002 (midpoint of 2nd depletion step)
StdCmpMix00040_00003 (final compositions, end of 2nd depletion step)
StdCmpMix00040 (same as StdCmpMix00040_00003)
```

The contents of these files will be a standard composition description of each material by atomic contents-that is, SCALE standard nuclide IDs (e.g., U-235), number density, and temperature (using the temperature of the original material). Using SCALE's external file read capability, these outputs may be automatically included in a follow-on calculation that relies on depleted/decayed number densities. TRITON sample problem 7 (Sect. 3.1.6.6) provides an example of the use of these restart files.

These files are not automatically copied back to the output directory. Users can use a shell block to manually copy back the files, for example:

```
read shell
  cp ${TMPDIR}/StdCmpMix* ${OUTDIR}
end shell
```

Important: Standard composition restart files should be used only for follow-on criticality or shielding calculations.

3.1.4.2 Lattice physics parameters

During T-DEPL depletion calculations that use branch states and homogenization, a database of few-group cross sections is saved for each branch state and at each depletion step containing homogenized cross section data and other lattice physics parameters (e.g., discontinuity factors, pin power peaking factors, diffusion coefficients, etc.). The *xfile016* file is intended for post-processing, to be read and written in the desired format for subsequent nodal diffusion core simulator calculations. The *xfile016* file is a binary-formatted file, which is described in detail in Appendix A of TRITON. An auxiliary text-formatted database file (*txtfile16*) is also created that contains the same data as the binary-formatted file.

3.1.4.3 ORIGEN binary library files (.f33)

During depletion calculations, ORIGEN binary library files are created to archive cross sections for each depletion material at each depletion subinterval. These files can be used in future depletion calculations in ORIGEN, ORIGAMI, and ARP. For each depletion material, the ORIGEN binary library file is named *$ {BASENAME}.mixNNNNNN.f33*, where *NNNNN* is the material number for each depleted material. Additionally, the system-average cross section file is saved with the name *$ {BASENAME}.system.f33*. All f33-files are automatically copied back into the output directory.

Note that in SCALE versions up to 6.2, these files were named *ft33f001_mixNNNNN* and *f33f001_cmbined*. They had to be copied manually from the temporary working directory *$ {TMPDIR}* into the output directory through a shell block following the TRITON input. Since SCALE 6.3, the files are automatically copied back and have more intuitive names.

3.1.4.4 ORIGEN binary concentration file (.f71)

During depletion calculations, TRITON creates the ORIGEN binary concentration file (.f71). This file is created in the temporary directory as *ft71f001* and is copied back at the end of the SCALE calculation to the return directory with the name ${OUTBASENAME}.f71. TRITON archives computed concentrations for each depletion material at the beginning and end of each depletion subinterval or decay interval. These files include concentrations and also decay heat term, photon and neutron data, and other quantities or interest computed by ORIGEN. These data may be post-processed by the OPUS module.

The .f71 file contains concentrations for each individual material, and it also contains the combined concentrations of the individual material results (i.e., the net response for the entire system). The TRITON output contains an index of the contents of this file (see Sect. 3.1.5.4.5).

3.1.4.5 Binary mesh response files (.3dmap)

If mesh responses are requested, the corresponding binary 3dmap files are generated. If the TRITON-Shift sequences with the definitions and tallies block was used to request mesh responses, the output files follow the following syntax:

- flux: *$ {BASENAME}.flux_time$ {STEP}_3dmap*
- fission_density: *$ {BASENAME}.fission_density_time$ {STEP}.3dmap*
- fission_source: *$ {BASENAME}.fission_source_time$ {STEP}.3dmap*

3.1.4.6 Binary Shift output file (.h5)

If the TRITON-Shift sequence was used, detailed results of the Shift calculation can be found in Shift's hdf5 output file. The HDF5 filename follow the following syntax: *$ {basename}_time$ {step}.shift-output.h5*. The generation of these files for only selected depletion steps can be controled through the *KEEP_OUTPUT* block.

3.1.5 OUTPUT DESCRIPTION

This section contains a brief description and explanation of TRITON output. As with any SCALE module, TRITON output begins with the SCALE header, the job information, the input file, and the program verification information. These outputs are common to all SCALE modules. Likewise, all SCALE calculations report a run-time summary at the end of the output file.

3.1.5.1 Control parameter edit

When TRITON control parameters are specified using the parm= command (see Sect. 3.1.3.5), all specified parameters are echoed following the above output, with an explanation of the meaning of the parameter, as shown below. If no parameters are specified, no edit is provided.

```
The following TRITON control parameters were requested:

   WEIGHT - Weighted collapsed master library
            option selected for t-newt calculation, based
            on system-averaged flux.
   ADDNUX - specifies the set of additional nuclides added
            in trace quantities for depletion
            calculations. Set 1 was selected.
            See TRITON manual for more information.
```

3.1.5.2 T-XSEC output

The T-XSEC sequence performs only cross section processing functions. The XSProc output is written to the output file as the calculation proceeds.

3.1.5.3 T-NEWT and T-XSDRN output

By default, the T-NEWT and T-XSDRN outputs include only the NEWT and XSDRN output respectively. The XSProc output can be included by using the *KEEP_OUTPUT* block (see Sect. 3.1.3.4.7).

3.1.5.4 Depletion sequence output

The output of TRITON depletion sequences contains several depletion edits. The edits are described in the following subsections. These output edits are written to the output file in the order in which they are computed during the calculation.

Burnup history summary (all depletion sequences)

TRITON generates the burnup history summary table after processing the *BURNDATA* block. An example of this table is as follows:

```
*******************************************************************************************
Based on the supplied burnup history, triton will use the following time history to perform
depletion calculations.  This breakdown has been calculated so as to permit burn steps of
no more than 40 days, and decay times using the rule of threes with a maximum first decay
period of no more than 75 days.
*******************************************************************************************

  7 Time-dependent libraries will be created

Sub-Interval   Depletion   Sub-interval    Specific      Burn Length  Decay Length   Library Burnup
     No.       Interval     in interval  Power(MW/MTIHM)     (d)          (d)           (MWd/MTIHM)
-------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------
        0     ****Initial Bootstrap Calculation****                                    0.00000E+00
        1          1                1          37.883      42.500         0.000          8.05014e+02
        2          1                2          37.883      42.500        15.000          2.41504e+03
        3          2                1          32.215      45.000         0.000          3.94489e+03
        4          2                2          32.215      45.000        50.000          5.39457e+03
-------------------------------------------------------------------------------------------------
NOTE: Library Burnup is the cumulative burnup computed at the midpoint of the depletion sub-interval.
      Specific Power and Library Burnup depend on basis mixture(s) selected in DEPLETION block.
-------------------------------------------------------------------------------------------------
```

This table shows the results of a burnup history using one depletion interval with 5 depletion subintervals. Column 1 is the cumulative depletion subinterval number. Column 2 is the depletion interval number, and column 3 is the depletion subinterval number within the current depletion interval. Columns 4–6 echo the specific power, depletion interval, and decay interval specified in the *BURNDATA* block. The final column shows the cumulative burnup at the midpoint of each depletion subinterval.

Embedded transport model output

The output from the initial transport calculation follows the burnup history edit. The output edits for NEWT, XSDRN, KENO-V.a, and KENO-VI are described in their respective manual sections.

System mass balance table

After the initial transport calculation output, a summary of system mass information is printed, an example of which is provided as follows.

```
*****************************************************************************
** System total mass is  1.8684e+01 grams heavy metal per unit length.       **
** Masses will be normalized by a factor of  5.3521e+04 cm to obtain a total  **
** system mass of  1.0000e+06 g of heavy metal                                **
*****************************************************************************

    Mix   Heavy Metal   Normalized HM  Fractional HM  Heavy Metal      Mixture
    No.   Mass (g/cm)     Mass (g)      Mass (---)     Dens. (g/cc)   Dens. (g/cc)   Depletion Mode
      1    1.868420e+01  1.000000e+06    1.000000     9.177679e+00   1.041200e+01   Depleted by power
     25    0.000000e+00  0.000000e+00    0.000000     0.000000e+00   6.440000e+00   Not depleted
     26    1.951123e-16  1.044264e-11    0.000000     5.964321e-17   6.801399e-01   Depleted by flux
 System    1.868420e+01  1.000000e+06    1.000000     2.942208e+00   4.746240e+00
```

This table provides mass and density data for each material used in the transport model. Column 1 provides the material identifier, and columns 5 and 6 provide the material density and material heavy metal density, respectively, in units of grams per cubic centimeter. Heavy metal mass is determined from masses of all nuclides with an atomic number greater than 89. The final column provides the depletion mode for each material (see Sect. 3.1.3.3.6). Column 2 provides the "prenormalized" heavy metal mass of each material. The units for this mass value depend on the transport model. For 2D *xy* NEWT models, the units are grams per centimeter since there is no *z*- dimension in the model. Similarly, the units are grams per centimeter for 1D cylinder XSDRN models, grams per square centimeter for 1D slab XSDRN models, and grams for 1D spherical XSDRN models and 3D KENO models. The total prenormalized heavy metal mass is printed in the final row of the table as well as in the table header. The heavy metal mass is normalized such that a total system mass of 1 MTHM is present. The volume scaling factor used to normalize the system mass is also printed in the table banner. The units of the volume scaling factor depend on the transport model. Column 3 prints the normalized material heavy metal mass in units of grams, which is equal to the prenormalized material heavy metal mass in column 2 multiplied by the volume scaling factor in the table header. The total normalized mass is printed in the final row and also in the table header. The fourth column shows the fractional heavy metal mass of all materials, which is equal to the normalized heavy metal mass in column 3, divided by the total normalized system heavy metal mass in the table header.

Power balance tables

As the TRITON calculation proceeds, the results of the cross section processing and transport calculations are used to calculate fluxes and powers in each mixture. The output segments listed in the next two tables show the results for the first calculation based on the initial mixture compositions. The total power (column 2) represents the mixture-specific power in units of MW/MTHM of initial **system** mass. The fractional power (column 3) is equal to the total power for a mixture divided by the total system power. The mixture power (column 4) represents mixture-specific power in units of MW/MTHM of initial **mixture** mass. The mixture power is equal to the total power of the mixture divided by the fractional heavy metal mass of the mixture, which is provided in the system mass balance table (Sect. 3.1.5.4.3). Column 5 presents the burnup of the individual mixture. If a mixture does not contain heavy metal, then "N/A" is printed in the mixture power and burnup columns. Columns 6 and 7 show the mixture thermal and total flux values, respectively, in units of neutrons/cm^2-sec. The thermal flux is determined by integrating multigroup flux values for energy groups below 0.625 eV.

If the specific power is normalized to the total system power, the summation of the mixture powers in column 1 should match the input specification in the *BURNDATA* block (in the example given here, 37.883 MW/MTHM):

```
--- Mixture powers for depletion pass no.   1 (MW/MITHM) ---
     Time =    21.25 days (   0.058 y), Burnup =    0.805     GWd/MTIHM, Transport k=  1.2783

                 Total    Fractional  Mixture      Mixture     Mixture       Mixture
      Mixture    Power      Power      Power       Burnup      Thermal Flux  Total Flux
       Number (MW/MTIHM)    (---)   (MW/MTIHM)   (GWd/MTIHM)   n/(cm^2*sec)  n/(cm^2*sec)
          1      37.799    0.99779     37.799         0.803    3.2106e+13    3.1724e+14
         25       0.041    0.00109     N/A           N/A       3.4431e+13    3.1682e+14
         26       0.042    0.00112     N/A           N/A       3.5114e+13    3.1832e+14
      Total      37.883    1.00000

  NOTE:    Total Power is the Mixture Power per 1 metric ton of HM of the initial  system mass.
         Mixture Power is the Mixture Power per 1 metric ton of HM of the initial mixture mass.
         Mixture Burnup is the Mixture Burnup per 1 metric ton of HM of the initial mixture mass.
         Mixture Thermal Flux determined using 0.625 eV cutoff: Groups 214 through 252.
 ------------------------------------------------------------
```

If the specific power is normalized to the power to one or more specific mixtures, the mixture powers are slightly different. For the case above, if depletion was performed with input power normalized to mixture 1, then mixture 1 has the input-specified power (37.883 MW/MTHM), and the power in the remainder of the model mixtures is normalized according to this basis mixture:

```
--- Mixture powers for depletion pass no.   1 (MW/MITHM) ---
     Time =    21.25 days (   0.058 y), Burnup =    0.805     GWd/MTIHM, Transport k=  1.2783

                 Total    Fractional  Mixture      Mixture     Mixture       Mixture
       Mixture   Power      Power      Power       Burnup      Thermal Flux  Total Flux
        Number (MW/MTIHM)    (---)   (MW/MTIHM)   (GWd/MTIHM)   n/(cm^2*sec)  n/(cm^2*sec)
           1  *   37.883    0.99779     37.883         0.805    3.2176e+13    3.1795e+14
          25       0.041    0.00109     N/A           N/A       3.4507e+13    3.1753e+14
          26       0.042    0.00112     N/A           N/A       3.5191e+13    3.1903e+14
       Total      37.967    1.00000

      * - Power normalized to this mixture.

  NOTE:    Total Power is the Mixture Power per 1 metric ton of HM of the initial  system mass.
         Mixture Power is the Mixture Power per 1 metric ton of HM of the initial mixture mass.
         Mixture Burnup is the Mixture Burnup per 1 metric ton of HM of the initial mixture mass.
         Mixture Thermal Flux determined using 0.625 eV cutoff: Groups 214 through 252.
 ---------------------------------------------------------------
```

Note: Note that the above two power balance tables refer to results at the time of the neutron transport calculation, i.e. the middle of a depletion subinterval.

Additionally, *after* every depletion subinterval, a summary of the mixture-wise power, flux, fluence, burnup, and initial heavy metal is provided. This is the result from ORIGEN, i.e. results noramlized to 1 ton of initial heavy metal.

```
end-of-step summary at time =    42.500  days (    0.116  y), system-average burnup* =    1.606  GWd/MTIHM

mixture        power         flux      fluence      burnup*    initialhm
    (-)         (MW)    (n/cm2-s)      (n/cm2)  (MWd/MTIHM)      (MTIHM)
------- ------------ ------------ ------------ ------------ ------------
      1  3.77957e+01  3.16389e+14  1.16178e+21  1.60632e+03  1.00000e+00

* Burnup is only calculated for mixtures with initial HM mass fraction greater than 1e-6.
```

ORIGEN binary concentration file listing

After all depletion calculations are completed, TRITON creates an ORIGEN binary concentration file (.f71) with isotopic concentrations for each depletion material. The order and content of the .f71 file is provided in the TRITON output. An example of this edit is shown below. For each depletion material, the output gives the location in the file, the ORIGEN time interval number, the depletion interval time in days, the cumulative time in years, and a title. After all materials are added to the library, the system average of all libraries and the average of all fuel mixtures are computed and added to the library. The file *case* numbers correspond to the mixture ids. Special cases are 0, -1, and -2 as indicated in the table below.

```
File ft71f001 is the ORIGEN binary concentration file, containing concentrations for
  - each of the 2 depletion mixtures,
  - a set for the sum of all depletion mixtures,
  - a set for the sum of selected mixtures (from optional opus block),
  - a set for the sum of all fuel mixtures
for always 7 time steps.
```

(continues on next page)

(continued from previous page)

```
Isotopic data locations are listed according to the following table.

Position  Time Step  Cycle Time (d)  Cumulative Time (y)
     1          0       0.0000e+00       0.0000e+00       Depletion mixture no. 1 (ft71 case=1)
     2          1       4.2500e+01       1.1636e-01
     3          2       4.2500e+01       2.3272e-01
     4          3       1.5000e+01       2.7379e-01
     5          4       4.5000e+01       3.9699e-01
     6          5       4.5000e+01       5.2019e-01
     7          6       5.0000e+01       6.5708e-01
     8          0       0.0000e+00       0.0000e+00       Depletion mixture no. 26 (ft71 case=26)
     9          1       4.2500e+01       1.1636e-01
    10          2       4.2500e+01       2.3272e-01
    11          3       1.5000e+01       2.7379e-01
    12          4       4.5000e+01       3.9699e-01
    13          5       4.5000e+01       5.2019e-01
    14          6       5.0000e+01       6.5708e-01
    15          0       0.0000e+00       0.0000e+00       Weighted sum of concentrations of all depleted␣
↪mixtures (ft71 case=0)
    16          1       4.2500e+01       1.1636e-01
    17          2       4.2500e+01       2.3272e-01
    18          3       1.5000e+01       2.7379e-01
    19          4       4.5000e+01       3.9699e-01
    20          5       4.5000e+01       5.2019e-01
    21          6       5.0000e+01       6.5708e-01
    22          0       0.0000e+00       0.0000e+00       Weighted sum of concentrations for selected mixtures␣
↪(ft71 case=-1)
    23          1       4.2500e+01       1.1636e-01
    24          2       4.2500e+01       2.3272e-01
    25          3       1.5000e+01       2.7379e-01
    26          4       4.5000e+01       3.9699e-01
    27          5       4.5000e+01       5.2019e-01
    28          6       5.0000e+01       6.5708e-01
    29          0       0.0000e+00       0.0000e+00       Weighted sum of concentrations for fuel mixtures␣
↪(ft71 case=-2)
    30          1       4.2500e+01       1.1636e-01
    31          2       4.2500e+01       2.3272e-01
    32          3       1.5000e+01       2.7379e-01
    33          4       4.5000e+01       3.9699e-01
    34          5       4.5000e+01       5.2019e-01
    35          6       5.0000e+01       6.5708e-01
```

The requested OPUS output edits follow this .f71 file summary edit.

3.1.6 TRITON SAMPLE CASES

This section provides descriptions of the 13 TRITON sample problems included with SCALE. Note that all of these problems (along with all other SCALE sample problems) are typically executed in the initial SCALE installation to test the performance of various codes and options, for validation of the installation process. Because of the number of problems that are executed, these sample problems are adjusted to run as fast as possible so that all test problems may be completed in relatively short order. To accomplish this, crude modeling approximations (reduced convergence, few histories, simplified cross section processing, low-order quadrature and scattering approximations, coarse computational grids, reduced numbers of libraries per depletion cycle, etc.) may be used. Hence, although these problems provide guidance in setting up and executing TRITON problems, it is generally a good idea to review all control settings to ensure sufficient accuracy in one's own calculations.

Additional TRITON input files for several reactor types can be generated with the SCALE/ORIGEN Library Generator (SLIG). The SLIB documentation is available as Appendix B of the ORIGEN chapter.

3.1.6.1 TRITON sample problem 1: triton1.inp

Sample problem triton1.inp is an example of a T-NEWT transport calculation sequence. Input begins (as with all SCALE sequences) with a title card and cross section library specification; this calculation is performed using the 252-group ENDF/B-7.1 library. After the library specification, three materials are defined in the composition block, followed by a cell specification and the NEWT transport model.

This example includes an axial height of 37.1 cm and will therefore do a buckled calculation based on this height. The geometric model consists of a simple pin cell, with cylindrical fuel and clad regions inside a square moderator region, with a 6×6 base grid. The NEWT *BOUNDS* block specifies that periodic boundary conditions are used for this model.

This simple problem also demonstrates the use of TRITON's automatic cross section collapse capability-*parm=weight*. For *T-NEWT* calculations, TRITON uses the NEWT *COLLAPSE* block to define the broad-group energy structure. For this sample problem, the cross sections are collapsed to a 56-group format. The new broad-group library will be identified as filename *newxnlib* in the temporary working directory, which can be used in follow-up SCALE calculations.

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t-newt sequence
'  ** v7-252 group library
'  ** centrm cross section processing (default for t-newt calculations)
'  ** parm=weight option for the t-newt sequence, which uses the NEWT collapse block to specify a 252 -> 56␣
↪group collapse.
'  ** latticecell cross section processing option
=t-newt        parm=weight
Buckled pin-cell transport calculation
v7-252
read comp
  u-234 1 0 6.74213e-6 296.15 end
  u-235 1 0 7.65322e-4 296.15 end
  u-236 1 0 3.68820e-6 296.15 end
  u-238 1 0 2.20912e-2 296.15 end
  o     1 0 4.57338e-2 296.15 end
  b-10  1 0 3.64042e-9 296.15 end
  b-11  1 0 1.46531e-8 296.15 end
  cr    25 0 6.67242e-5 296.15 end
  fe    25 0 1.25922e-4 296.15 end
  sn    25 0 4.17642e-4 296.15 end
  o     25 0 2.63724e-4 296.15 end
  zr    25 0 3.78392e-2 296.15 end
  h     26 0 6.68559e-2 296.15 end
  o     26 0 3.34279e-2 296.15 end
end comp
read celldata
  latticecell squarepitch pitch=1.2600 26 fuelr=0.4095 1 cladr=0.4750 25 end
end celldata
read model
238 group solution
read parm
  dz=37.1
end parm
read materials
  mix=1  com="3.0 enriched fuel, pin location 1" end
  mix=25 com="cladding" end
  mix=26 com="water" end
end materials
read geom
  global unit 1
    cylinder 10 0.4095
    cylinder 20 0.4750
    cuboid   30 4p0.63
```

(continues on next page)

(continued from previous page)

```
    media  1 1 10
    media 25 1 20 -10
    media 26 1 30 -20
  boundary 30 6 6
end geom
read collapse
  8r1 2r2 3 3r4 5 5r6 6r7 2r8 3r9 4r10 4r11 12 13 10r14 3r15 16 6r17
  3r18 18r19 2r20 6r21 22 3r23 24 7r25 26 16r27 2r28 11r29 30 31 14r32
  33 2r34 35 3r36 35r37 5r38 7r39 11r40 4r41 2r42 43 44 3r45 2r46 2r47 2r48
  2r49 2r50 51 52 2r53 54 3r55 10r56

'  OLD 238G collapse to 49G
'  7r1 2 3 2r4 5 6 7 8 8 8r9 14r10 6r11 10r12 13 7r14 11r15 12r16 30r17 16r18 2r19
'  6r20 3r21 6r22 14r23 3r24 5r25 4r26 5r27 5r28 5r29 10r30 5r31 32 33 34 2r35
'  36 37 38 2r39 2r40 3r41 2r42 43 44 45 46 47 3r48 9r49 end collapse
read bounds
all=periodic
end bounds
end model
end
```

3.1.6.2 TRITON sample problem 2: triton2.inp

Sample problem triton2.inp is an example of a T-XSDRN transport calculation sequence. In this case, the parameter specification *parm=2region* instructs TRITON to perform cross section processing using the CENTRM-based two-region option in place of the default CENTRM-based S_N option (see Sect. 3.1.2.1). As in sample problem 1, a simple square-pitched pin cell is modeled but in this case using an XSDRN model block rather than the NEWT model block. The moderator radius was defined in order to preserve the volume of the moderator region.

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t-xsdrn sequence
'  ** v7-252 group library
'  ** 2region cross section processing
'  ** latticecell cross section processing option
=t-xsdrn           parm=2region
pin-cell model with MOX
v7-252
read comp
' Fuel
  u-234 1 0  2.5952E-7 900  end
  pu-238 1 0 4.6610E-5 900  end
  pu-241 1 0 1.7491E-4 900  end
  pu-242 1 0 1.3201E-4 900  end
  o-16  1 0  4.6586E-2 900  end
  pu-240 1 0 4.8255E-4 900  end
  pu-239 1 0 1.0156E-3 900  end
  u-235 1 0  5.4287E-5 900  end
  u-238 1 0  2.1387E-2 900  end
' zirc
  zr-90 2 0 3.8657E-2 620 end
  fe    2 0 1.3345E-4 620 end
  cr    2 0 6.8254E-5 620 end
' h2o
  h-1  3 0 4.8414E-2 575  end
  o-16 3 0 2.4213E-2 575  end
  b-10 3 0 4.7896E-6 575  end
  b-11 3 0 1.9424E-5 575  end
end comp
read cell
  latticecell squarepitch pitch=1.3127 3 fueld=0.8200 1 cladd=0.9500 2 end
end cell
```

(continues on next page)

(continued from previous page)

```
read model
pin-cell model with MOX
read parm
  sn=16
end parm
read materials
  mix=1 com='fuel' end
  mix=2 com='clad' end
  mix=3 com='moderator' end
end materials
read geom
  geom=cylinder
  rightBC=white
  zoneIDs 1 2 3 end zoneids
  zoneDimensions 0.41 0.475 0.7406117 end zoneDimensions
  zoneIntervals 3r10 end zoneIntervals
end geom
end model
end
```

3.1.6.3 TRITON sample problem 3: triton3.inp

Sample problem 3 illustrates the input format for a T-DEPL-1D depletion calculation. In this case, a single square-pitched pin-cell model is depleted, where the fuel composition is comprised of UO_2 fuel homogenized with aluminum and B_4C. Although this is not representative of real fuel, it does allow one to observe the effect of boron depletion during burnup; results will show an increasing multiplication factor as boron is depleted, followed by a decreasing eigenvalue after the fuel depletion becomes the dominant contributor to reactivity change. Three depletion intervals are specified with the same power and no decay intervals. Two depletion subintervals are specified for the first two depletion intervals, with only one depletion subinterval for the final depletion interval. Note that this may be insufficient to capture the effect of boron depletion early in life; fewer depletion subintervals are used here only to reduce the run-time for this sample problem. In this model, power is normalized such that material 1 has a power density of 21.22 MW/MTHM (or MT/MTU for UO_2 fuel), and OPUS output is requested for 35 nuclides. The problem is run using the addnux=3 option set to add trace quantities of 231 nuclides to depletion materials.

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t-depl-1d sequence
'  ** v7-252 group library
'  ** Sn centrm cross section processing (default for t-depl-1d calculations)
'  ** latticecell cross section processing option
'  ** parm=addnux=3 option to add 231 nuclides to fuel material
'  ** deplete-by-constant power
'  ** mixture power normalization
'  ** opus block
=t-depl-1d        parm=(addnux=3)
Infinite lattice depletion model for a single pincell.
v7-252
read comp
' Fuel/AL2O3-B4C
  uo2   1 den=10.045 1 841 92234 0.022 92235 2.453 92236 0.011 92238 97.514 end
  b-10  1  0  8.5900E-4   841.0 end
  b-11  1  0  3.4400E-3   841.0 end
  c     1  0  1.0700E-3   841.0 end
  al    1  0  3.9000E-2   841.0 end
' Clad
  wtptzirc 4 6.44 4 40000 97.91 26000 0.5 50116 0.86 50120 0.73 1.0 620  end
' Moderator
  h2o  5 den=0.7573  1  557 end
end comp
```

(continues on next page)

(continued from previous page)

```
read celldata
  latticecell squarepitch   pitch=1.4732  5 fuelr=0.47250 1 cladr=0.5588  4  end
end celldata
read depletion
  -1
end depletion
read burndata
  power=21.220 burn=750 down=0 nlib=2 end
  power=21.220 burn=750 down=0 nlib=2 end
  power=21.220 burn=375 down=0 nlib=1 end
end burndata
read opus
  units=gram
  symnuc=u-234 u-235 u-236 u-238 pu-238 pu-239
  pu-240 pu-241 pu-242 pu-243 np-237
  cs-133 cs-134 cs-135 cs-137 nd-143 nd-144 nd-145 nd-146
  nd-148 nd-150 pm-147 sm-147 sm-148 sm-149 sm-150 sm-151
  sm-152 eu-153 sm-154 eu-154 gd-154 eu-155 gd-155 o-16 end
  matl=0 1  end
end opus
read model
Infinite-lattice pin model (one-fourth)
read parm
  sn=16
end parm
read materials
  mix=1      com='fuel' end
  mix=4      com='clad' end
  mix=5 pn=2 com='water' end
end materials
read geom
  geom=cylinder
  rightBC=white
  zoneIDs 1 4 5 end zoneids
  zoneDimensions 0.47250 0.5588 0.83116409 end zoneDimensions
  zoneIntervals 3r10 end zoneIntervals
end geom
end model
end
```

TRITON sample problem 4: triton4.inp

Sample problem *triton4.inp* performs a large-scale depletion calculation for a one-fourth PWR assembly, taking advantage of symmetry to reduce the problem size. The same fuel material is used in each fuel rod, which will result in assembly-averaged isotopic compositions for all fuel rods. If one wanted to obtain an isotopic estimate for one or more unique fuel rod locations, then different materials would be specified for different rod positions. Even though all fuel is identical at the beginning of life, unique materials must be specified if one desires to perform tracking of the unique response of each unique fuel position.

The problem parameter specification *parm=(weight)* instructs TRITON to perform an automated cross section library collapse. For library collapse automation within depletion calculations (see Sect. 3.1.3.5.3), TRITON will perform a single 252-group calculation at t = 0 to generate the 56-group cross section library. TRITON will restart the depletion calculation at t = 0 using the broad-group library after it is created. Because *parm=weight* is specified, the *NEWT COLLAPSE* block must comply with the 56-group energy structure and not the 252-group energy structure. The *COLLAPSE* block input is slightly different for the library collapse automation for *T-NEWT* calculations, where the *NEWT COLLAPSE* block must comply with the 252-group energy structure.

Problem 4 also uses a timetable to specify boron letdown in the moderator. The initially specified boron concentration in the *COMP* (or *COMPOSITION*) data block is multiplied by a density multiplier at the

time of each cross section processing and transport calculation (i.e., the midpoint of depletion subinterval). Linear interpolation is performed between values on the timetable to obtain the multiplier for a given time. Typically a multiplier of 1.0 is used for t = 0, and the beginning-of-life boron concentration is input in the *COMPOSITION* block, but this example demonstrates that this is not necessary. For this calculation, a 500 ppm boron concentration is specified in the standard composition description, and a concentration of (500 ppm)*(1.832), or 916 ppm, would be used in the t = 0 transport calculation.

Problem 4 is also an example of a lattice physics calculation for a full fuel assembly. The NEWT model employs coarse-mesh finite-difference acceleration, whole-assembly homogenization, 2-energy-group collapse, and a pin-power print, and computes assembly discontinuity factors. Although this sample problem will create the cross section database file for core calculations, this sample problem does not contain branching calculations, nor do lattice physics calculations typically use boron letdown curves. Additional guidance for TRITON lattice physics calculations can be found in the lattice physics primer.

Because only one fuel material is used, only one cell specification is necessary. If multiple fuel materials were used, then a corresponding cell specification would be required for each fuel, with a unique clad and moderator material identifier for each cell. To apply boron letdown properly, the moderator present in each cell specification would need to have the same letdown curve specified. Hence, a letdown timetable would need to be specified for each moderator (even if the moderators are not all used in the NEWT *model* block). If multiple fuel materials are used, requiring corresponding multiple clad, moderation, cell, and timetable specifications, the use of an *alias* specification can simplify input. Aliases are described in Sect. 3.1.3.4; sample problems triton6.inp (Sect. 3.1.6.5), triton8.inp (Sect. 3.1.6.7), and triton12.inp (Sect. 3.1.6.10) demonstrate the use of aliases.

This case also illustrates the use of stacked OPUS cases within a single TRITON input file. Here, an OPUS calculation is requested to obtain the mass in grams of 26 actinides and fission products for material 1 and for the entire system; since material 1 is the entire set of depletion materials, the system output will be identical to the material 1 output. A second OPUS calculation is also specified, which requests a ranked output of the top 20 nuclides in terms of decay heat (in watts).), TRITON will perform a single 252-group calculation at t = 0 to generate the 56-group cross section library. TRITON will restart the depletion calculation at t = 0 using the broad-group library after it is created. Because *parm=weight* is specified, the *NEWT COLLAPSE* block must comply with the 56-group energy structure and not the 252-group energy structure. The *COLLAPSE* block input is slightly different for the library collapse automation for *T-NEWT* calculations, where the *NEWT COLLAPSE* block must comply with the 252-group energy structure.

Problem 4 also uses a timetable to specify boron letdown in the moderator. The initially specified boron concentration in the *COMP* (or *COMPOSITION*) data block is multiplied by a density multiplier at the time of each cross section processing and transport calculation (i.e., the midpoint of depletion subinterval). Linear interpolation is performed between values on the timetable to obtain the multiplier for a given time. Typically a multiplier of 1.0 is used for t = 0, and the beginning-of-life boron concentration is input in the *COMPOSITION* block, but this example demonstrates that this is not necessary. For this calculation, a 500 ppm boron concentration is specified in the standard composition description, and a concentration of (500 ppm)*(1.832), or 916 ppm, would be used in the t = 0 transport calculation.

Problem 4 is also an example of a lattice physics calculation for a full fuel assembly. The NEWT model employs coarse-mesh finite-difference acceleration, whole-assembly homogenization, 2-energy-group collapse, and a pin-power print, and computes assembly discontinuity factors. Although this sample problem will create the cross section database file for core calculations, this sample problem does not contain branching calculations, nor do lattice physics calculations typically use boron letdown curves. Additional guidance for TRITON lattice physics calculations can be found in the lattice physics primer.

Because only one fuel material is used, only one cell specification is necessary. If multiple fuel materials were used, then a corresponding cell specification would be required for each fuel, with a unique clad and moderator material identifier for each cell. To apply boron letdown properly, the moderator present in each cell specification would need to have the same letdown curve specified. Hence, a letdown timetable would need to be specified for each moderator (even if the moderators are not all used in the NEWT *model* block). If multiple fuel materials are used, requiring corresponding multiple clad, moderation, cell, and timetable specifications, the use of an *alias* specification can simplify input. Aliases are described in Sect. 3.1.3.4; sample problems triton6.inp (Sect. 3.1.6.5), triton8.inp (Sect. 3.1.6.7), and triton12.inp (Sect. 3.1.6.10) demonstrate the use of aliases.

This case also illustrates the use of stacked OPUS cases within a single TRITON input file. Here, an OPUS calculation is requested to obtain the mass in grams of 26 actinides and fission products for material 1 and for the entire system; since material 1 is the entire set of depletion materials, the system output will be identical to the material 1 output. A second OPUS calculation is also specified, which requests a ranked output of the top 20 nuclides in terms of decay heat (in watts).

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t-depl sequence
'  ** v7-252 group library
'  ** 2region cross section processing
'  ** parm=weight option for the t-depl sequence, which uses builtin 49-group collapse
'  ** latticecell cross section processing option
'  ** deplete-by-constant power
'  ** system power normalization
'  ** timetable block using density multiplier
'  ** opus block defining multiple plots
=t-depl      parm=(2region,weight)
Large scale 2-D depletion model with a boron letdown curve
v7-252
read comp
  uo2       1 den=10.412 1 900 92234 0.04 92235 4.11  92238 95.85 end
  wtptzirc 25 6.44 4 40000 97.91 26000 0.5 50116 0.86 50120 0.73 1.0 600 end
  h2o      26 den=0.6798 1 593 end
  wtptbor  26 0.6798 1 5000 100 500e-6 593 end
end comp
read celldata
  latticecell squarepitch pitch=1.2600 26 fuelr=0.4025 1 cladr=0.4750 25 end
end celldata
read depletion
  1
end depletion
read timetable
  densmult 26 2 5010 5011
    0.0    1.832
    106    1.419
    205    1.033
    306    0.641
    385    0.611
    473    1.797
    592    1.371
    704    0.973
    817    0.568
    875    0.362 end
end timetable

read burndata
  power=37.883 burn=385 down=88  nlib=1 end
  power=32.215 burn=402 down=158 nlib=1 end
end burndata
read opus
  units=gram
```

(continues on next page)

(continued from previous page)

```
  symnuc=u-234 u-235 u-236 u-238 pu-238 pu-239
    pu-240 pu-241 pu-242 np-237 am-241 am-243 cm-242 cm-243
    cs-134 cs-137 nd-143 nd-144 nd-145 nd-146 cm-244 cm-245
    cm-246 cm-247 ru-106 am-242m end
  matl=0 1  end
newcase
  units=watts sort=yes nrank=20 time=years
end opus
read model
One-fourth fuel assembly
read parm
  drawit=yes cmfd=yes xycmfd=0 echo=yes collapse=yes sn=4 inners=3 outers=200 epsilon=1e-3
end parm
read materials
  mix=1  com='4.11 wt % enriched fuel' end
  mix=25 com='cladding' end
  mix=26 com='water' end
end materials
read collapse
  40r1 16r2
end collapse
read homog
  500 whole_assm 1 25 26 end
end homog
read adf
  1 500 n=10.71 e=10.71 end adf
read geom
' unit 25 is a right-half water hole
  unit 25
    cylinder 10 .4500 chord +x=0.0
    cylinder 20 .4950 chord +x=0.0
    cuboid 30 0.63 0.0 0.63 -0.63
    media 26 1 10
    media 25 1 20 -10
    media 26 1 30 -20
  boundary 30 2 4
' unit 45 is top-half water hole
  unit 45
    cylinder 10 .4500 chord +y=0.0
    cylinder 20 .4950 chord +y=0.0
    cuboid 30 0.63 -0.63 0.63 0.0
    media 26 1 10
    media 25 1 20 -10
    media 26 1 30 -20
  boundary 30 4 2

' unit 46 is a 1/4 water hole
  unit 46
    cylinder 10 .4500 chord +x=0 chord +y=0
    cylinder 20 .495 chord +x=0 chord +y=0
    cuboid 30 0.63 0. 0.63 0.
    media 26 1 10
    media 25 1 20 -10
    media 26 1 30 -20
  boundary 30 2 2
' unit 1 is a full material #1 rod
  unit 1
    cylinder 10 .4025
    cylinder 20 .4950
    cuboid 30 0.63 -0.63 0.63 -0.63
    media 1 1 10
    media 25 1 20 -10
    media 26 1 30 -20
  boundary 30 4 4
' unit 2 is a top-half material #1 rod
```

(continues on next page)

(continued from previous page)

```
  unit 2
    cylinder 10 .4025 chord +y=0
    cylinder 20 .4950 chord +y=0
    cuboid 30 0.63 -0.63 0.63 0.0
    media 1 1 10
    media 25 1 20 -10
    media 26 1 30 -20
  boundary 30 4 2
' unit 3 is a right-half material #1 rod
  unit 3
    cylinder 10 .4025 chord +x=0
    cylinder 20 .4950 chord +x=0
    cuboid 30 0.63 0.0 0.63 -0.63
    media 1 1 10
    media 25 1 20 -10
    media 26 1 30 -20
  boundary 30 2 4
  global unit 100
    cuboid 1 10.71 0.0 10.71 0.0
    array 10 1
    media 26 1 1
  boundary 1
end geom
read array
  ara=10 nux=9 nuy=9 pinpow=yes typ=cuboidal
  fill 46  2  2 45  2  2 45  2  2
        3  1  1  1  1  1  1  1  1
        3  1  1  1  1  1  1  1  1
       25  1  1  1  1  1  1  1  1
        3  1  1  1  1  1  1  1  1
        3  1  1  1  1  1  1  1  1
       25  1  1  1  1  1  1  1  1
        3  1  1  1  1  1  1  1  1
        3  1  1  1  1  1  1  1  1 end fill
end array
read bounds
  all=refl
end bounds
end model
end
```

3.1.6.4 TRITON sample problem 5: triton5.inp

Sample problem *triton5.inp* is similar to *triton4.inp*, except that it is a T5-DEPL case; thus, a KENO V.a transport model is used in place of the NEWT model of the earlier case. The KENO V.a model, although 3D, is axially uniform with reflecting boundary conditions, so it is effectively the same model as the 2D model of *triton4.inp*. Moreover, the KENO V.a model represents the full assembly rather than a one-fourth model. Hence, both cases will generate similar results. In the KENO model, only 300,000 neutron histories are retained, which is somewhat low to obtain good statistics on fluxes. The 238 ENDF/B-VII library is used for this sample problem compared to the 252 ENDF/B-VII.1 library utilized in *triton4.inp*.

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t-depl sequence
'  ** v7-252 group library
'  ** 2region cross section processing
'  ** latticecell cross section processing option
'  ** deplete-by-constant power
'  ** system power normalization
'  ** timetable block using density multiplier
=t5-depl              parm=2region
Large scale 2-D depletion model with boron density change.
V7-238
```

(continues on next page)

(continued from previous page)

```
read comp
  uo2       1 den=10.412 1 900 92234 0.04 92235 4.11  92238 95.85 end
  wtptzirc 25 6.44 4 40000 97.91 26000 0.5 50116 0.86 50120 0.73 1.0 600 end
  h2o      26 den=0.6798 1 593 end
  wtptbor  26 0.6798 1 5000 100 500e-6 593 end
end comp
read celldata
  latticecell squarepitch pitch=1.2600 26 fuelr=0.4025 1 cladr=0.4750 25 end
end celldata
read depletion
  1
end depletion
read timetable
  densmult 26 2 5010 5011
    0.0    1.832
    106    1.419
    205    1.033
    306    0.641
    385    0.611
    473    1.797
    592    1.371
    704    0.973
    817    0.568
    875    0.362 end
end timetable
read burndata
  power=37.883   burn=385  down=88 nlib=1 end
  power=32.215   burn=402  down=158 nlib=1 end
end burndata
read model
read parm
  cfx=yes gen=620 nsk=20 npg=500 plt=no htm=no
end parm

read geom
' unit 2 is a water hole
  unit 2
    cylinder 26 1 .4500 10.0 0.0
    cylinder 25 1 .4950 10.0 0.0
    cuboid 26   1 0.63 -0.63 0.63 -0.63 10.0 0.0
' unit 1 is a material #1 rod
  unit 1
    cylinder 1  1 .4025  10.0 0.0
    cylinder 25 1 .4950 10.0 0.0
    cuboid 26   1 0.63 -0.63 0.63 -0.63 10.0 0.0
  global unit 100
    array 10 0.0 0.0 0.0
end geom
read array
ara=10 nux=17 nuy=17 nuz=1 typ=cuboidal
 fill 17r1
      17r1
      8r1 2 8r1
      17r1
      17r1
      8r1 2 8r1
      17r1
      17r1
      2r1 2 2r1 2 2r1 2 2r1 2 2r1 2 2r1
      17r1
      17r1
      8r1 2 8r1
      17r1
      17r1
      8r1 2 8r1
```

(continues on next page)

(continued from previous page)

```
      17r1
      17r1 end fill
end array
read bounds
  all=refl
end bounds
end data
end model
end
```

3.1.6.5 TRITON sample problem 6: triton6.inp

Sample problem *triton6.inp* performs T-DEPL depletion in a pin cell model; however, the pin is discretized into five equal-volume rings of fuel. Thus, CENTRM-based S_N cross section processing is necessary to capture the radial burnup of the pin cell. A *multiregion* cell specification is given to allow specification of the varying radii for the fuel regions. Because the multiregion cell is cylindrical, the moderator volume is represented in terms of a radius that corresponds to the volume associated with the pin pitch. The right boundary condition for the cell is set to *white*; this is important, as the default right boundary condition for a multiregion cylinder is vacuum. In this case, addnux=1 is also requested in the parameter specification, simply for a faster (but less accurate) calculation. Material aliases are used to simplify input. The calculation is performed with the 238 ENDF/B-VII library. The TRITON *TIMETABLE* block is used to demonstrate time-dependent temperature changes to the moderator material.

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t-depl sequence
'  ** v7-252 group library
'  ** centrm cross section processing
'  ** multiregion cross section processing option
'  ** deplete-by-constant power
'  ** parm=addnux=1 option to add 15 nuclides to fuel material
'  ** system power normalization
'  ** timetable block using temperature change
'  ** alias block definition
'  ** opus block
=t-depl           parm=(centrm,addnux=1)
Pin-cell depleted in rings
v7-252

read alias
  $fuel 1-5 end
end alias

read comp
  uo2 $fuel den=9.459 1 829.0 92234 0.027 92235 3.038 92236 0.014 92238 96.921 end
  wtptzirc  10 6.44 4 40000 97.91 26000 0.5 50116 0.86 50120 0.73 1.0 620  end
  h2o 11 den=0.7575 1 557 end
  wtptbor 11 0.7575 1 5000 100 654e-6 557 end
end comp

read celldata
  multiregion cylindrical right=white end
    1 0.16425
    2 0.28449
    3 0.36727
    4 0.43456
    5 0.49275
   10 0.55880
   11 .83120 end zone
end celldata
```

(continues on next page)

(continued from previous page)

```
read depletion
  $fuel
end depletion

read timetable
  temperature 11
'   cycle 1
      0.0 557.0
    306.0 557.0
'   cycle 2
    377.0 540.0
    838.1 557.0 end
end timetable

read burndata
  power=27.24 burn=306.0 down=71   nlib=1 end
  power=34.57 burn=461.1 down=1870 nlib=1 end
end burndata
read opus
  units=gram symnuc=u-235 u-238 pu-239 pu-241 nd-148 end matl=0 1 2 3 4 5 end
end opus
read model
Infinite lattice PWR pin cell
read parm
  drawit=yes prtbroad=yes epsilon=1e-3 soln=b1 converg=matl
end parm
read materials
  mix=$fuel com='3.038 wt % enriched fuel' end
  mix=10 pn=0  com='cladding' end
  mix=11 com='water' end
end materials
read geom
  global unit 1
    cylinder 1 .16425 chord +x=0 chord +y=0
    cylinder 2 .28449 chord +x=0 chord +y=0
    cylinder 3 .36727 chord +x=0 chord +y=0
    cylinder 4 .43456 chord +x=0 chord +y=0
    cylinder 5 .49275 chord +x=0 chord +y=0
    cylinder 20 .5588 chord +x=0 chord +y=0
    cuboid   30 0.7366 0.0 0.7366 0.0
    media 1 1 1
    media 2 1 2 -1
    media 3 1 3 -2
    media 4 1 4 -3
    media 5 1 5 -4
    media 10 1 20 -5
    media 11 1 30 -20
  boundary 30 4 4
end geom
read bounds
  all=refl
end bounds
end model
end
```

3.1.6.6 TRITON sample problem 7: triton7.inp

Sample problem triton7.inp is an example of a T-DEPL depletion calculation for a full PWR fuel assembly model. Depletion is performed on the basis of material 7, which is located in a single fuel pin for which destructive assay measurements were performed. All other fuel is modeled as a single (average) material, material 1. The parameter specification *parm=(2region,addnux=1,weight)* was chosen to reduce the run-time of the sample problem.

This sample problem also demonstrates the use of TRITON's standard composition restart files and SCALE external file reading capabilities to represent the time-dependent behavior of an assembly in which burnable poisons are removed after the first cycle of operation.

This problem consists of two TRITON 2D depletion cases. In the first case, the full assembly model contains borosilicate glass burnable poison rods (BPRs), material 4, which are included in the list of materials to be depleted. The calculation is run for the entirety of the first operational cycle, which included a 40-day mid-cycle decay interval. The model also includes a 64-day decay interval after the end of the operational cycle. When this calculation is completed, TRITON creates in the temporary working directory a standard composition file for each material containing the isotopic inventories for each depletion material at the end of the 64-day decay interval. The second TRITON calculation reads the standard composition specifications for materials 1 and 7 as part of the input to provide the fuel state for the second calculation. In the second model, the BPRs are removed and replaced with the moderator in the embedded NEWT model. The initial depletion calculation uses the 252 ENDF/B-VII.1 library. With the *parm=(...,weight)* option, a 56 group library is created in the temporary directory called *newxnlib*. This library is used for the second *T-NEWT* calculation.

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t-depl sequence
'  ** v7-252 group library
'  ** 2region cross section processing
'  ** latticecell cross section processing option
'  ** deplete-by-constant flux
'  ** parm=addnux=1 option to add 15 nuclides to fuel material
'  ** mixture power normalization
'  ** timetable block using density multiplier
'  ** composition restart files.
'  ** weight used to collapse library for reuse in restart calculation
=t-depl      parm=(2region,addnux=1,weight)
ASSEMBLY model with BPRs with depletion
v7-252

read comp
 uo2      1 den=9.550 1 743  92234 0.023 92235 2.561 92236 0.013 92238 97.403 end
 wtptzirc 2 6.44 4 40000 97.91 26000 0.5 50116 0.86 50120 0.73 1.0 620  end
 h2o      3 den=0.7544 1 559 end
 wtptbor  3 0.7544 1 5000 100  652.5e-6 559 end
 wtptbpr  4 2.081 6 8016 53.58 11000 2.82 13027 1.758 14000 37.63 19000 0.33  5000 3.882 1 559 end
 wtptair  5 0.00129 2 7000 78.0 8016 22.0  1 559.0 end
 ss304    6 1 559.0 end
 uo2      7 den=9.550 1 743  92234 0.023 92235 2.561 92236 0.013 92238 97.403 end
 wtptzirc 8 6.44 4 40000 97.91 26000 0.5 50118 0.64 50120 0.95 1 595 end
 h2o      9 den=0.7544 1 559 end
 wtptbor  9 0.7544 1 5000 100  652.5e-6 559 end
end comp

read celldata
 latticecell squarepitch pitch=1.43 3 fueld=0.9484 1 cladd=1.0719 2 end
 latticecell squarepitch pitch=1.43 9 fueld=0.9484 7 cladd=1.0719 8 end
end celldata

read depletion
 1 -7 flux 4
```

(continues on next page)

(continued from previous page)

```
end depletion

read timetable
 density 3 2 5010 5011
     0.00 1.000
   243.5  1.000
   283.5  0.379
   527.0  0.379 end
 density 9 2 5010 5011
     0.00 1.000
   243.5  1.000
   283.5  0.379
   527.0  0.379 end
end timetable

read burndata
  power=20.86 burn=243.5 down=40.0  nlib=1 end
  power=20.15 burn=243.5 down=64.0  nlib=1 end
end burndata

read model
ASSEMBLY model with BPRs with depletion

read parm
  drawit=yes inners=2 epsilon=-5e-2 cmfd=1 xycmfd=0 echo=yes solntype=b1 timed=yes
end parm

read materials
  mix=6  pn=1  com="SS-304 - BPR clad" end
  mix=5  pn=1  com="air in BPRs" end
  mix=4  pn=1  com="borosilicate glass" end
  mix=3  pn=2  com="water" end
  mix=2  pn=1  com="cladding" end
  mix=1  pn=1  com="2.561 wt % enriched fuel " end
  mix=7  pn=1  com="rod N-9" end
end materials

read geom
  global unit 10
    cuboid 13 10.725 0.0 10.725 0.0
    array 101 13  place 1 1 -0.715 -0.715
    media 3 1 13
    boundary 13 15 15
  unit 1
    cuboid 13 1.43 0.0 1.43 0.0
    cylinder 12 0.53595 origin x=0.715 y=0.715
    cylinder 11 0.4742 origin x=0.715 y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 1 1 11
    boundary 13 2 2
  unit 2
    cuboid 13 1.43 0.0 1.43 0.0
    cylinder 14 0.28385 origin x=0.715 y=0.715
    cylinder 15 0.30035 origin x=0.715 y=0.715
    cylinder 16 0.50865 origin x=0.715 y=0.715
    cylinder 17 0.55755 origin x=0.715 y=0.715
    media 3 1 13 -17
    media 6 1 17 -16
    media 4 1 16 -15
    media 6 1 15 -14
    media 5 1 14
    boundary 13 2 2
  unit 3
    cuboid 13 1.43 0.0 1.43 0.0
```

(continues on next page)

(continued from previous page)

```
    cylinder 12 0.6934 origin x=0.715 y=0.715
    cylinder 11 0.6502 origin x=0.715 y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 3 1 11
    boundary 13 2 2
  unit 4
    cuboid 13 1.43 0.715 1.43 0.0
    cylinder 12 0.53595 origin x=0.715 y=0.715 chord +x=0.715
    cylinder 11 0.4742 origin x=0.715 y=0.715 chord +x=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 1 1 11
    boundary 13 1 2
  unit 5
    cuboid 13 1.43 0.0 1.43 0.715
    cylinder 12 0.53595 origin x=0.715 y=0.715 chord +y=0.715
    cylinder 11 0.4742 origin x=0.715 y=0.715 chord +y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 1 1 11
    boundary 13 2 1
  unit 6
    cuboid 13 1.43 0.715 1.43 0.0
    cylinder 12 0.6934 origin x=0.715 y=0.715 chord +x=0.715
    cylinder 11 0.6502 origin x=0.715 y=0.715 chord +x=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 3 1 11
    boundary 13 1 2
  unit 7
    cuboid 13 1.43 0.0 1.43 0.715
    cylinder 12 0.6934 origin x=0.715 y=0.715 chord +y=0.715
    cylinder 11 0.6502 origin x=0.715 y=0.715 chord +y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 3 1 11
    boundary 13 2 1
  unit 8
    cuboid 13 1.43 0.715 1.43 0.715
    cylinder 12 0.6934 origin x=0.715 y=0.715 chord +x=0.715 chord +y=0.715
    cylinder 11 0.6502 origin x=0.715 y=0.715 chord +x=0.715 chord +y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 3 1 11
    boundary 13 1 1
  unit 9
    cuboid 13 1.43 0.0 1.43 0.0
    cylinder 12 0.53595 origin x=0.715 y=0.715
    cylinder 11 0.4742 origin x=0.715 y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 7 1 11
    boundary 13 2 2
end geom

read array
ara=101 nux=8 nuy=8 typ=cuboidal fill
8 5 5 5 7 5 5 5
4 1 1 1 1 1 1 1
4 1 1 1 1 2 1 1
4 1 1 3 1 1 1 1
6 1 1 1 1 1 1 1
4 9 2 1 1 2 1 1
4 1 1 1 1 1 1 1
```

(continues on next page)

(continued from previous page)

```
4 1 1 1 1 1 1 1 end fill
end array
end model
end
```

```
=t-newt      parm=(2region)
ASSEMBLY model without BPRs
newxnlib

read comp
<StdCmpMix00001
  wtptzirc 2 6.44 4 40000 97.91 26000 0.5 50116 0.86 50120 0.73 1.0 620  end
  h2o      3 den=0.7544 1 559 end
  wtptbor  3 0.7544 1 5000 100  652.5e-6 559 end
<StdCmpMix00007
  wtptzirc 8 6.44 4 40000 97.91 26000 0.5 50118 0.64 50120 0.95 1 595 end
  h2o      9 den=0.7544 1 559 end
  wtptbor  9 0.7544 1 5000 100    652.5e-6 559 end
end comp

read celldata
   latticecell squarepitch pitch=1.43 3 fueld=0.9484 1 cladd=1.0719 2 end
   latticecell squarepitch pitch=1.43 9 fueld=0.9484 7 cladd=1.0719 8 end
end celldata

read model
ASSEMBLY model without BPRs

read parm
  drawit=yes inners=2 epsilon=-5e-2 cmfd=1 xycmfd=0 echo=yes solntype=b1 timed=yes
end parm

read materials
  mix=3  pn=2  com="water" end
  mix=2  pn=1  com="cladding" end
  mix=1  pn=1  com="2.561 wt % enriched fuel " end
  mix=7  pn=1  com="rod N-9" end
end materials

read geom
  global unit 10
    cuboid 13 10.725 0.0 10.725 0.0
    array 101 13  place 1 1 -0.715 -0.715
    media 3 1 13
    boundary 13 15 15
  unit 1
    cuboid 13 1.43 0.0 1.43 0.0
    cylinder 12 0.53595 origin x=0.715 y=0.715
    cylinder 11 0.4742 origin x=0.715 y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 1 1 11
    boundary 13 2 2
  unit 3
    cuboid 13 1.43 0.0 1.43 0.0
    cylinder 12 0.6934 origin x=0.715 y=0.715
    cylinder 11 0.6502 origin x=0.715 y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 3 1 11
    boundary 13 2 2
  unit 4
    cuboid 13 1.43 0.715 1.43 0.0
    cylinder 12 0.53595 origin x=0.715 y=0.715 chord +x=0.715
```

(continues on next page)

(continued from previous page)

```
    cylinder 11 0.4742 origin x=0.715 y=0.715 chord +x=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 1 1 11
    boundary 13 1 2
  unit 5
    cuboid 13 1.43 0.0 1.43 0.715
    cylinder 12 0.53595 origin x=0.715 y=0.715 chord +y=0.715
    cylinder 11 0.4742 origin x=0.715 y=0.715 chord +y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 1 1 11
    boundary 13 2 1
  unit 6
    cuboid 13 1.43 0.715 1.43 0.0
    cylinder 12 0.6934 origin x=0.715 y=0.715 chord +x=0.715
    cylinder 11 0.6502 origin x=0.715 y=0.715 chord +x=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 3 1 11
    boundary 13 1 2
  unit 7
    cuboid 13 1.43 0.0 1.43 0.715
    cylinder 12 0.6934 origin x=0.715 y=0.715 chord +y=0.715
    cylinder 11 0.6502 origin x=0.715 y=0.715 chord +y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 3 1 11
    boundary 13 2 1
  unit 8
    cuboid 13 1.43 0.715 1.43 0.715
    cylinder 12 0.6934 origin x=0.715 y=0.715 chord +x=0.715 chord +y=0.715
    cylinder 11 0.6502 origin x=0.715 y=0.715 chord +x=0.715 chord +y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 3 1 11
    boundary 13 1 1
  unit 9
    cuboid 13 1.43 0.0 1.43 0.0
    cylinder 12 0.53595 origin x=0.715 y=0.715
    cylinder 11 0.4742 origin x=0.715 y=0.715
    media 3 1 13 -12
    media 2 1 12 -11
    media 7 1 11
    boundary 13 2 2
end geom

read array
ara=101 nux=8 nuy=8 typ=cuboidal fill
8 5 5 5 7 5 5 5
4 1 1 1 1 1 1 1
4 1 1 1 1 3 1 1
4 1 1 3 1 1 1 1
6 1 1 1 1 1 1 1
4 9 3 1 1 3 1 1
4 1 1 1 1 1 1 1
4 1 1 1 1 1 1 1 end fill
end array
end model
end
```

3.1.6.7 TRITON sample problem 8: triton8.inp

Sample problem *triton8.inp* is an example of TRITON's simplified cross section processing scheme in a BWR-like configuration. It uses the T-DEPL extended format for the *DEPLETION* block to allow material assignments to be made to reduce the number of cross section processing calculations required in a multimaterial depletion model. In this sample problem, the lower-left quadrant of a 6×6 fuel assembly is modeled (see Fig. 3.1.12). The fuel pin layout is as follows: one 2.3% ^{235}U-enriched fuel pin in the southwest corner (red pin), one-fourth of a water rod in the northeast corner, five 3.6% ^{235}U-enriched fuel pins (2 green, 1 yellow, and 2 blue pins), and two gadolinium-bearing pins each modeled with three fuel rings. (Gadolinium-bearing pins are typically modeled with multiple fuel rings due to the strong spatial dependence of the flux.) Due to diagonal symmetry, only seven depletion material regions need to be defined: the red, green, yellow, and blue pins along with three regions for the gadolinium-bearing pins. Although seven depletion materials are defined, only three cell specifications are used: one for the gadolinium-bearing pin cell and one each for the 2.3% and 3.6% ^{235}U-enriched pin cells. This model makes extensive use of aliases. The sequence is run with *parm=(addnux=0,...)*, which includes no extra nuclides, for an accelerated solution; however, this is an extremely poor approximation unless important nuclides are manually specified. This example also illustrates the use of the keyword *flux* in the *DEPLETION* data block to force flux-based ORIGEN calculations in place of power-based calculation for all three rings of the gadolinium-bearing fuel pins. Finally, this case uses the *parm=(...,weight,...)* directive to request the automatic collapse of the input 238 ENDF/B-VII library to a 49 group library (collapsed using the 238-group system-averaged flux), which is then used for depletion calculations.

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t-depl sequence
'  ** v7-252 group library
'  ** centrm cross section processing
'  ** parm=weight option for the t-depl sequence, which uses builtin 49-group collapse
'  ** latticecell cross section processing option
'  ** multi-region cross section processing option
'  ** deplete-by-constant flux and by constant power
'  ** parm=addnux=0 option to add 0 nuclides to fuel material
'  ** system power normalization
'  ** depletion assignments
'  ** alias block
=t-depl    parm=(addnux=0,weight)
BWR-like depletion model with lattice physics calculations
v7-252
read alias
 $fuel23 1 end
 $fuel36 2-4 end
 $2fuels 1,2 end
 $2clads 401-402 end
 $2mods 201-202 end
 $fuelg 5-7 end
 $h2osolid 399 end
 $allfuels 1-7 end
end alias

read comp
' 2.3 w/o
  u-234 1 0 4.7008e-06 900 end
  u-235 1 0 5.2968e-04 900 end
  u-236 1 0 3.4083e-06 900 end
  u-238 1 0 2.2208e-02 900 end
  o-16  1 0 4.5491e-02 900 end
' 3.6 w/o
  u-234 2 0 7.5720e-06 900 end
  u-235 2 0 8.2904e-04 900 end
  u-236 2 0 5.1701e-06 900 end
```

(continues on next page)

(continued from previous page)

```
  u-238 2 0 2.1907e-02 900 end
  o-16  2 0 4.5497e-02 900 end
' Gd Pin
  u-234 $fuelg 0 5.8824e-06 900 end
  u-235 $fuelg 0 6.5057e-04 900 end
  u-236 $fuelg 0 4.1028e-06 900 end
  u-238 $fuelg 0 2.0759e-02 900 end
  o-16  $fuelg 0 4.5095e-02 900 end
  gd-154 $fuelg 0 3.2253e-05 900 end
  gd-155 $fuelg 0 2.2141e-04 900 end
  gd-156 $fuelg 0 3.0778e-04 900 end
  gd-157 $fuelg 0 2.3576e-04 900 end
  gd-158 $fuelg 0 3.7393e-04 900 end
  gd-160 $fuelg 0 3.3200e-04 900 end
' Clad nat. zr per spec.
  zirc4  $2clads den=6.53 1 559 end
  zirc4  409 den=6.53 1 559 end
' lwtr mod    den (g/cc) (1-void) tmp(K)
  h2o $2mods  den=0.457 1.0000   559 end
  h2o 209  den=0.457 1.0000   559 end
  h2o 399  den=0.737 1.0000   559 end
end comp

read celldata
  latticecell squarepitch pitch=1.63 $2mods fuelr=0.529 $2fuels cladr=0.615 $2clads end
  multiregion cylindrical right_bdy=white end
    7  0.37405950
    6  0.45812740
    5  0.52900000
  409  0.61500000
  209  0.91962900 end zone
end celldata
read depletion
  $fuel23 $fuel36 flux $fuelg end
  assign 1 $fuel23 end
  assign 2 $fuel36 end
end depletion
read burndata
  power=25.0 burn=300  end
end burndata
read model
BWR-like model with lattice physics calculations
read parm
  soln=b1 echo=yes drawit=yes sn=4 collapse=yes epsilon=1e-3 cmfd=yes xycmfd=3
end parm
read materials
  mix=$allfuels pn=0 end
  mix=401       pn=0  com='Zirc4' end
  mix=201       pn=0  com='H2O(void)' end
  mix=399       pn=0  com='H2O(solid)' end
end materials
read adf
  1 500 w=0.0 s=0.0
end adf
read collapse
  30r1 19r2
end collapse
read hmog
  500 PSZ 1 2 3 4 5 6 7 401 399 201 end
end hmog

read geom
  unit 001
    cuboid 1 1.63 0.0000 1.63 0.0000
    cylinder 2 0.615 origin x=0.815 y=0.815
```

(continues on next page)

(continued from previous page)

```
    cylinder 3 0.529 origin x=0.815 y=0.815
    media 201 1 1 -2
    media 401 1 2 -3
    media 001 1 3
  boundary 1 3 3
  unit 002
    cuboid 1 1.63 0.0000 1.63 0.0000
    cylinder 2 0.615 origin x=0.815 y=0.815
    cylinder 3 0.529 origin x=0.815 y=0.815
    media 201 1 1 -2
    media 401 1 2 -3
    media 002 1 3
  boundary 1 3 3
  unit 003
    cuboid 1 1.63 0.0000 1.63 0.0000
    cylinder 2 0.615 origin x=0.815 y=0.815
    cylinder 3 0.529 origin x=0.815 y=0.815
    media 201 1 1 -2
    media 401 1 2 -3
    media 003 1 3
  boundary 1 3 3
  unit 004
    cuboid 1 1.63 0.0000 1.63 0.0000
    cylinder 2 0.615 origin x=0.815 y=0.815
    cylinder 3 0.529 origin x=0.815 y=0.815
    media 201 1 1 -2
    media 401 1 2 -3
    media 004 1 3
  boundary 1 3 3
  unit 005
    cuboid 1 1.6300 0.0000 1.6300 0.0000
    cylinder 2 0.6150 origin x=0.8150 y=0.8150
    cylinder  3 0.52900000 origin x=0.8150 y=0.8150
    cylinder  4 0.45812740 origin x=0.8150 y=0.8150
    cylinder  5 0.37405950 origin x=0.8150 y=0.8150
    media 201 1 1 -2
    media 401 1 2 -3
    media 005 1 003   -4
    media 006 1 004   -5
    media 007 1 005
  boundary 1 3 3
' water channels
  unit 121
    cuboid 1 1.63 0.0000 1.63 0.0000
    cylinder 2 1.6 origin x=1.63 y=1.63
    chord -x=1.63 chord -y=1.63 sides=16
    cylinder 3 1.5 origin x=1.63 y=1.63
    chord -x=1.63 chord -y=1.63 sides=16
    media 201 1 1 -2
    media 401 1 2 -3
    media 399 1 3
  boundary 1 3 3
  global unit 50
    cuboid 1 5.99 1.10  5.99 1.1
    cuboid 2 5.99 0.846 5.99 0.846
    cuboid 3 5.99 0.00  5.99 0.00
    array 1 1 place 1 1 1.1 1.1
    media 399 1 3 -2
    media 401 1 2 -1
  boundary 3 12 12
end geom
  ara=1 nux=3 nuy=3 pinpow=yes fill   1   2   3
                                  2   4   5
                                  3   5   121  end fill
end array
```

(continues on next page)

(continued from previous page)

```
end model
end
```

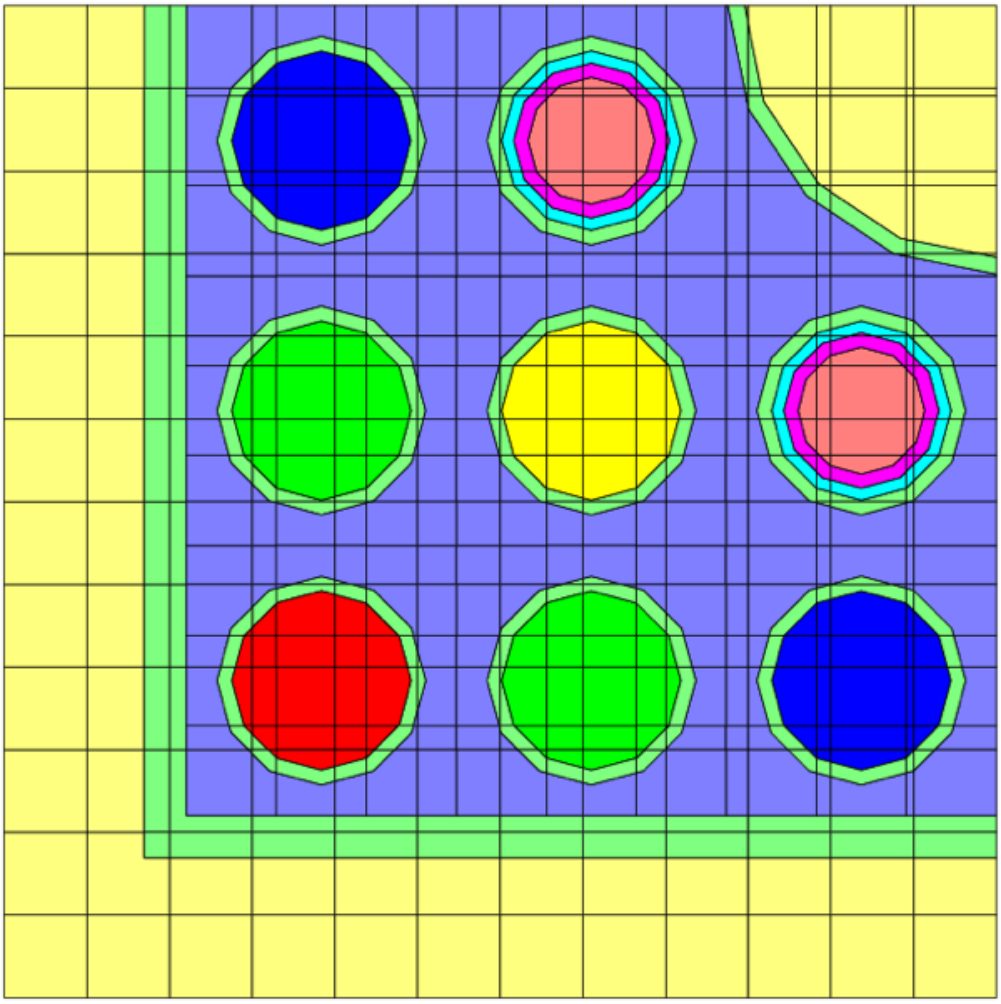


Fig. 3.1.12: BWR-like assembly design for triton8.

3.1.6.8 TRITON sample problem 10: triton10.inp

Sample problem *triton10.inp* performs NEWT-based depletion of a one-fourth symmetric assembly model. The primary intent of this sample problem is to test and to demonstrate the use of branches and archival of computed cross sections for a depletion case. The model includes two branch calculations-in addition to the nominal condition-that demonstrate the different perturbation outputs available in the *BRANCH* block. A two-group collapse is requested in the NEWT input, along with homogenization over all materials. (Note that the parameter specification *parm=(addnux=1)* is only used to reduce run-time of the sample problem.) At the end of the calculation, the binary file "xfile016" and text file "txtfile16" will exist in the temporary working directory and will contain all lattice physics parameters for all branches at all depletion states. These files are often copied back from the SCALE temporary working directory to another more permanent directory for subsequent post-processing.

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t-depl sequence
'  ** v7-56 group library
'  ** centrm cross section processing
'  ** latticecell cross section processing option
'  ** deplete-by-constant power
'  ** parm=addnux=1 option to add 15 nuclides to fuel material
'  ** system power normalization
'  ** branch block to generate few-group cross sections.
=t-depl parm=(addnux=1)
1/4 assembly model
v7-56
read comp
  uo2      1 0.95 923 92235 3.0 92238 97.0 end
  zirc2    2 1 595 end
  h2o      3 den=0.7135 1      579 end
  boron    3 den=0.7135 600e-6 579 end
  n        4 den=0.00125 1 595 end
  zirc2    5 1 579 end
  h2o      6 den=0.7135 1      579 end
  boron    6 den=0.7135 600e-6 579 end
  h2o      7 den=0.7135 1      579 end
  boron    7 den=0.7135 600e-6 579 end
  zirc2    8 1 579 end
  b4c      9 den=2.52 1 579 end
end comp
read celldata
  latticecell squarepitch pitch=1.4300 3 fueld=0.9294 1 gapd=0.9484  4 cladd=1.0719 2 end
end celldata
read depletion
  1
end depletion
read burndata
  power=40.0  burn=75 end
end burndata
read branch
  define fuel 1 end
  define mod  3 end
  define crout 8 9 end
  define crin  6 7 end
  define d2pset 10 1 0.53 end
  define d2pset 20 1 0.93 end
  tf=923 dm=.7135 tm=579 cr=0 sb=600 end
  dm=0.01 sb=0 d2p=20 end
  cr=1 d2p=10 tf=300 end
end branch
read model
1/4 assembly model
read parm
  echo=yes drawit=no cmfd=1 xycmfd=0 inners=2 epsilon=-5e-5 outers=300
end parm

read materials
  mix=1 pn=0 com="fuel"  end
  mix=2 pn=0 com="clad" end
  mix=3 pn=0 com="water" end
  mix=4 pn=0 com="gap" end
  mix=5 pn=0 com="guide tube" end
  mix=6 pn=0 com="CRout-clad" end
  mix=7 pn=0 com="CRout-abs" end
  mix=8 pn=0 com="CRin-clad" end
  mix=9 pn=0 com="CRin-abs" end
end materials
read geom
  unit 1
    com='fuel rod'
```

(continues on next page)

(continued from previous page)

```
  cylinder 10 .4647
  cylinder 20 .4742
  cylinder 30 .53595
  cuboid   40  4p0.715
  media 1 1 10
  media 4 1 20 -10
  media 2 1 30 -20
  media 3 1 40 -30
  boundary  40 2 2
unit 5
  com='guide tube'
  cylinder 10 .45
  cylinder 20 .52
  cylinder 30 .6502
  cylinder 40 .6934
  cuboid   50  4p0.715
  media 7 1 10
  media 6 1 20 -10
  media 3 1 30 -20
  media 5 1 40 -30
  media 3 1 50 -40
  boundary  50 2 2
unit 11
  com='right half of fuel rod'
  cylinder 10 .4647   chord +x=0
  cylinder 20 .4742   chord +x=0
  cylinder 30 .53595  chord +x=0
  cuboid   40  0.715  0.0 2p0.715
  media 1 1 10
  media 4 1 20 -10
  media 2 1 30 -20
  media 3 1 40 -30
  boundary  40 1 2
unit 12
  com='top half of fuel rod'
  cylinder 10 .4647   chord +y=0
  cylinder 20 .4742   chord +y=0
  cylinder 30 .53595  chord +y=0
  cuboid   40  2p0.715 0.715 0.0
  media 1 1 10
  media 4 1 20 -10
  media 2 1 30 -20
  media 3 1 40 -30
  boundary  40 2 1
unit 51
  com='right half of guide tube'
  cylinder 10 .45   chord +x=0
  cylinder 20 .52   chord +x=0
  cylinder 30 .6502 chord +x=0
  cylinder 40 .6934 chord +x=0
  cuboid   50  0.715 0.0 2p0.715
  media 7 1 10
  media 6 1 20 -10
  media 3 1 30 -20
  media 5 1 40 -30
  media 3 1 50 -40
  boundary  50 1 2
unit 52
  com='top half of guide tube'
  cylinder 10 .45   chord +y=0
  cylinder 20 .52   chord +y=0
  cylinder 30 .6502 chord +y=0
  cylinder 40 .6934 chord +y=0
  cuboid   50  2p0.715 0.715 0.0
  media 7 1 10
```

(continues on next page)

(continued from previous page)

```
    media 6 1 20 -10
    media 3 1 30 -20
    media 5 1 40 -30
    media 3 1 50 -40
    boundary  50 2 1
  unit 53
    com='1/4 instrument tube'
    cylinder 10 .6502   chord +x=0 chord +y=0
    cylinder 20 .6934   chord +x=0 chord +y=0
    cuboid   40  0.715 0.0 0.715 0.0
    media 3 1 10
    media 5 1 20 -10
    media 3 1 40 -20
  boundary  40 1 1
global unit 10
    com='1/4 assembly'
    cuboid 10 10.725 0.0 10.725 0.0
    array 1 10 place 1 1 0 0
    media 3 1 10
  boundary 10 15 15
end geom
read coll
  25r1 19r2
end coll
read homog
  500 allmatl 1 2 3 4 5 6 7 8 9 end
end homog
read array
  ara=1 nux=8 nuy=8 typ=cuboidal pinpow=yes
  fill
    53  12 12 12 52 12 12 12
    11  1  1  1  1  1  1  1
    11  1  1  1  1  5  1  1
    11  1  1  5  1  1  1  1
    51  1  1  1  1  1  1  1
    11  1  5  1  1  5  1  1
    11  1  1  1  1  1  1  1
    11  1  1  1  1  1  1  1  end fill
end array
read bounds
  all=refl
end bounds
end model
end
```

3.1.6.9 TRITON sample problem 11: triton11.inp

Sample problem *triton11.inp* demonstrates the use of determining Dancoff factors for a BWR fuel assembly. The BWR assembly design contains a 7×7 array of fuel pins enclosed by a channel box (see Fig. 3.1.13). The in-channel moderator void fraction is 40%, and the bypass moderator void fraction is 0%. The input file contains an MCDANCOFF input file that calculates the Dancoff factors for each fuel pin (See the MCDANCOFF chapter). The MCDANCOFF input is essentially the equivalent of the KENO-VI model of the 2D assembly design. Following the MCDANCOFF input, the T-NEWT input is provided that shows how the computed Dancoff factors are inserted into the TRITON model. The Dancoff factors are inserted into the model via the *centrmdata* keyword entry in the *CELLDATA* block.

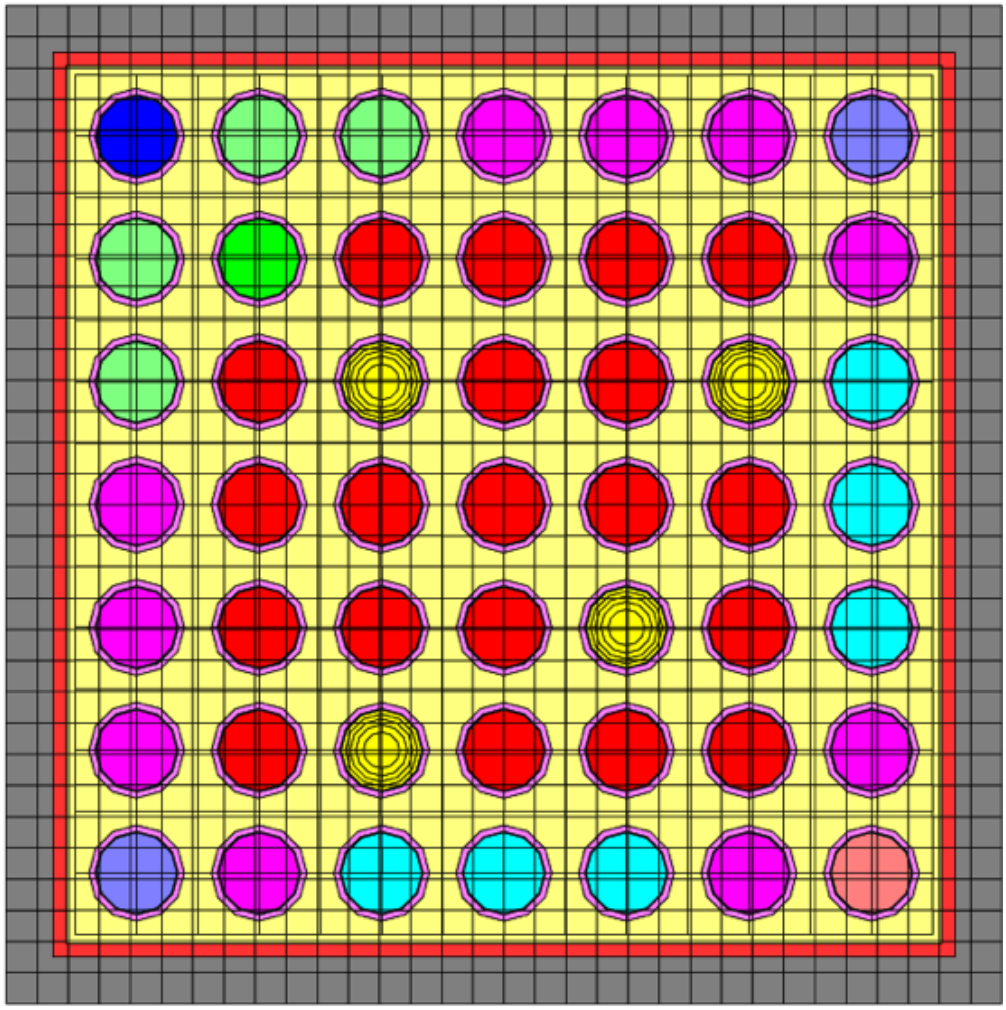


Fig. 3.1.13: triton11 BWR assembly design.

```
' THIS SAMPLE PROBLEM TESTS THE FOLLOWING:
'  ** mcdancoff sequence to compute problem dependent dancoffs.
=mcdancoff
PB CYCLE1
xn01
read comp
  uo2   1 den=10.42 0.99 900 92235 2.93 92234 0.0261 92236 0.0135 92238 97.0304 end
  uo2 201 den=10.42 0.99 900 92235 2.93 92234 0.0261 92236 0.0135 92238 97.0304 end
  uo2   2 den=10.42 0.99 900 92235 1.94 92234 0.0173 92236 0.0089 92238 98.0338 end
  uo2 202 den=10.42 0.99 900 92235 1.94 92234 0.0173 92236 0.0089 92238 98.0338 end
  uo2 212 den=10.42 0.99 900 92235 1.94 92234 0.0173 92236 0.0089 92238 98.0338 end
  uo2 203 den=10.42 0.99 900 92235 1.69 92234 0.0150 92236 0.0078 92238 98.2872 end
  uo2 213 den=10.42 0.99 900 92235 1.69 92234 0.0150 92236 0.0078 92238 98.2872 end
  uo2   4 den=10.42 0.99 900 92235 1.33 92234 0.0118 92236 0.0061 92238 98.6521 end
  uo2 500 den=10.29 0.97 900 92235 2.93 92234 0.0261 92236 0.0135 92238 97.0304 end
  gd2o3 500 den=10.29 0.03 900 end
  he 121 den=4.9559E-4 1 711.15 end
  zirc2 101 den=5.678 1 630 end
  h2o 111 den=0.4577 1 560 end
  zirc4 630 den=6.525 1 630 end
  h2o   620 den=0.738079 1 560  end
end comp
```

(continues on next page)

(continued from previous page)

```
read parm
  gen=100 npg=100 nsk=0 htm=no flx=yes fdn=yes run=yes
end parm
read geom
  unit   11
    com="corner rod 1.33% enr"
    cylinder 1   0.60579  2p1.0
    cylinder 2   0.62103  2p1.0
    cylinder 3   0.71501  2p1.0
    cuboid   4 4p0.9375   2p1.0
    media   4 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4
  unit   12
    com="edge rod 1.69% enr"
    cylinder 1   0.60579  2p1.0
    cylinder 2   0.62103  2p1.0
    cylinder 3   0.71501  2p1.0
    cuboid   4 4p0.9375   2p1.0
    media 203 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4
  unit   14
    com="edge rod 1.94% enr"
    cylinder 1   0.60579  2p1.0
    cylinder 2   0.62103  2p1.0
    cylinder 3   0.71501  2p1.0
    cuboid   4 4p0.9375   2p1.0
    media 202 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4
  unit   17
    com="corner rod 1.69% enr"
    cylinder 1   0.60579  2p1.0
    cylinder 2   0.62103  2p1.0
    cylinder 3   0.71501  2p1.0
    cuboid   4 4p0.9375   2p1.0
    media 213 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4
  unit   22
    com="interior rod 1.94% enr"
    cylinder 1   0.60579  2p1.0
    cylinder 2   0.62103  2p1.0
    cylinder 3   0.71501  2p1.0
    cuboid   4 4p0.9375   2p1.0
    media   2 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4
  unit   23
    com="interior rod 2.93% enr"
    cylinder 1   0.60579  2p1.0
    cylinder 2   0.62103  2p1.0
    cylinder 3   0.71501  2p1.0
    cuboid   4 4p0.9375   2p1.0
```

(continues on next page)

(continued from previous page)

```
    media   1 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4
  unit   33
    com="interior gad pin"
    cylinder 5   0.60579  2p1.0
    cylinder 6   0.62103  2p1.0
    cylinder 7   0.71501  2p1.0
    cuboid   8 4p0.9375   2p1.0
    media 500 1 5
    media 121 1 6 -5
    media 101 1 7 -6
    media 111 1 8 -7
    boundary    8
  unit   37
    com="edge rod 2.93% enr"
    cylinder 1   0.60579  2p1.0
    cylinder 2   0.62103  2p1.0
    cylinder 3   0.71501  2p1.0
    cuboid   4 4p0.9375   2p1.0
    media 201 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4
  unit   77
   com="corner rod 1.94% enr"
    cylinder 1   0.60579  2p1.0
    cylinder 2   0.62103  2p1.0
    cylinder 3   0.71501  2p1.0
    cuboid   4 4p0.9375   2p1.0
    media 212 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4
global unit 100
    cuboid  1   4p6.5625   2p1.0
    array 1 1  place 4 4 1  0.0 0.0 0.0
    cuboid  2   4p6.70306  2p1.0
    cuboid  3   4p6.90626  2p1.0
    cuboid  10  4p7.62     2p1.0
    media   111 1 2 -1
    media   630 1 3 -2
    media   620 1 10 -3
    boundary 10
end geom

read array
  ara=1 nux=7 nuy=7 nuz=1 typ=cuboidal
  fill
    17  14  37  37  37  14  77
    14  23  33  23  23  23  14
    14  23  23  23  33  23  37
    14  23  23  23  23  23  37
    12  23  33  23  23  33  37
    12  22  23  23  23  23  14
    11  12  12  14  14  14  17 end fill
end array
read bounds
  all=refl
end bounds
read start
```

(continues on next page)

(continued from previous page)

```
  dancoff  array  1  1  1  1  unit  17  region 1
  dancoff  array  1  1  2  1  unit  14  region 1
  dancoff  array  1  1  3  1  unit  14  region 1
  dancoff  array  1  1  4  1  unit  14  region 1
  dancoff  array  1  1  5  1  unit  12  region 1
  dancoff  array  1  1  6  1  unit  12  region 1
  dancoff  array  1  1  7  1  unit  11  region 1
  dancoff  array  1  2  1  1  unit  14  region 1
  dancoff  array  1  2  2  1  unit  23  region 1
  dancoff  array  1  2  3  1  unit  23  region 1
  dancoff  array  1  2  4  1  unit  23  region 1
  dancoff  array  1  2  5  1  unit  23  region 1
  dancoff  array  1  2  6  1  unit  22  region 1
  dancoff  array  1  3  1  1  unit  37  region 1
  dancoff  array  1  3  2  1  unit  33  region 1
  dancoff  array  1  3  3  1  unit  23  region 1
  dancoff  array  1  3  4  1  unit  23  region 1
  dancoff  array  1  3  5  1  unit  33  region 1
  dancoff  array  1  4  1  1  unit  37  region 1
  dancoff  array  1  4  2  1  unit  23  region 1
  dancoff  array  1  4  3  1  unit  23  region 1
  dancoff  array  1  4  4  1  unit  23  region 1
  dancoff  array  1  5  1  1  unit  37  region 1
  dancoff  array  1  5  2  1  unit  23  region 1
  dancoff  array  1  5  3  1  unit  33  region 1
  dancoff  array  1  6  1  1  unit  14  region 1
  dancoff  array  1  6  2  1  unit  23  region 1
  dancoff  array  1  7  1  1  unit  77  region 1
end start
end data
end
```

3.1.6.10 TRITON sample problem 12: triton12.inp

Sample problem triton12.inp illustrates the use of the TRITON-NEWT with a model with numerous mixtures and aliases. Sample problem 11 demonstrates how MCDANCOFF is used to compute fuel pin Dancoff factors for designs where nonuniform lattice effects play a critical role in cross section processing. The Dancoff factors are inserted into the follow-on T-NEWT model through the centrmdata keyword in the CELLDATA block. The output for the T-NEWT model in sample problem 11 provides an adjusted moderator pitch needed to preserve the user-specified Dancoff factor. As common in other sample problems, several input options were used to reduce the run-time—and therefore solution accuracy—of the sample problem.

```
=t-newt   parm=(check)
PB CYCLE1
v7-252
' Data taken from:
'        Benchmark for Uncertainty Analysis in Modeling (UAM)
'        for Design, Operation and Safety Analyses of LWRs,
'        Nuclear Energy Agency, 2007.
read alias
 $gadpin 500 end
 $clad   101 102 103 104 105 301 302 303 304 305 end
 $mod    111 112 113 114 115 311 312 313 314 315 end
 $gap    121 122 123 124 125 321 322 323 324 325 end
end alias
read comp
' 2.93% enriched fuel pin
  uo2   1 den=10.42 0.99 900 92235 2.93 92234 0.0261 92236 0.0135 92238 97.0304 end
  uo2 201 den=10.42 0.99 900 92235 2.93 92234 0.0261 92236 0.0135 92238 97.0304 end
' 1.94% enriched fuel pin
  uo2   2 den=10.42 0.99 900 92235 1.94 92234 0.0173 92236 0.0089 92238 98.0338 end
  uo2 202 den=10.42 0.99 900 92235 1.94 92234 0.0173 92236 0.0089 92238 98.0338 end
```

(continues on next page)

(continued from previous page)

```
  uo2 212 den=10.42 0.99 900 92235 1.94 92234 0.0173 92236 0.0089 92238 98.0338 end
' 1.69% enriched fuel pin
  uo2 203 den=10.42 0.99 900 92235 1.69 92234 0.0150 92236 0.0078 92238 98.2872 end
  uo2 213 den=10.42 0.99 900 92235 1.69 92234 0.0150 92236 0.0078 92238 98.2872 end
' 1.33% enriched fuel pin
  uo2   4 den=10.42 0.99 900 92235 1.33 92234 0.0118 92236 0.0061 92238 98.6521 end
' 3% Gd2O3 by weigh, 2.93% enriched fuel pin
  uo2   $gadpin den=10.29 0.97 900 92235 2.93 92234 0.0261 92236 0.0135 92238 97.0304 end
  gd2o3 $gadpin den=10.29 0.03 900 end
' gap/clad/moderator
  he    $gap  den=4.9559E-4 1 711.15  end
  zirc2 $clad den=5.678 1 630 end
  h2o   $mod    den=0.4577 1  560  end
' channel
  zirc4 630 den=6.525 1 630 end
' water in bypass
  h2o   620 den=0.738079 1 560  end
end comp
read celldata
  latticecell squarep pitch=1.87452 111 fuelr=0.60579   1 gapr=0.62103 121 cladr=0.71501 101 end
    centrmdata dan2pitch=0.504 end centrmdata
  latticecell squarep pitch=1.87452 112 fuelr=0.60579   2 gapr=0.62103 122 cladr=0.71501 102 end
    centrmdata dan2pitch=0.494 end centrmdata
  latticecell squarep pitch=1.87452 114 fuelr=0.60579   4 gapr=0.62103 124 cladr=0.71501 104 end
    centrmdata dan2pitch=0.362 end centrmdata
  latticecell squarep pitch=1.87452 311 fuelr=0.60579 201 gapr=0.62103 321 cladr=0.71501 301 end
    centrmdata dan2pitch=0.423 end centrmdata
  latticecell squarep pitch=1.87452 312 fuelr=0.60579 202 gapr=0.62103 322 cladr=0.71501 302 end
    centrmdata dan2pitch=0.423 end centrmdata
  latticecell squarep pitch=1.87452 313 fuelr=0.60579 203 gapr=0.62103 323 cladr=0.71501 303 end
    centrmdata dan2pitch=0.417 end centrmdata
  latticecell squarep pitch=1.87452 314 fuelr=0.60579 212 gapr=0.62103 324 cladr=0.71501 304 end
    centrmdata dan2pitch=0.359 end centrmdata
  latticecell squarep pitch=1.87452 315 fuelr=0.60579 213 gapr=0.62103 325 cladr=0.71501 305 end
    centrmdata dan2pitch=0.357 end centrmdata
  latticecell squarep pitch=1.87452 115 fuelr=0.60579 500 gapr=0.62103 125 cladr=0.71501 105 end
    centrmdata dan2pitch=0.506 end centrmdata
end celldata

read model
PB CYCLE1
read parm
 echo=yes timed=yes drawit=yes cmfd=1 epsilon=3e-3 inners=2 therm=yes therms=1 outers=9999 xycmfd=4
end parm
read materials
  mix=  1        pn=1  com='2.93% UO2'         end
  mix=  2        pn=1  com='1.94% UO2'         end
  mix=  4        pn=1  com='1.33% UO2'         end
  mix=$gadpin    pn=1  com='2.93% UO2 (3% Gd)' end
  mix=201        pn=1  com='2.93% UO2, edge'   end
  mix=202        pn=1  com='1.94% UO2, edge'   end
  mix=212        pn=1  com='1.94% UO2, corner' end
  mix=203        pn=1  com='1.69% UO2, edge'   end
  mix=213        pn=1  com='1.69% UO2, corner' end
  mix=111        pn=2  com='H2O(void)'         end
  mix=101        pn=1  com='Zirc2'             end
  mix=121        pn=1  com='Helium'            end
  mix=620        pn=2  com='H2O(solid)'        end
  mix=630        pn=1  com='Zirc4'             end
end materials
read geom
  unit   11
    com="corner rod 1.33% enr"
    cylinder 1   0.60579
    cylinder 2   0.62103
```

(continues on next page)

(continued from previous page)

```
  cylinder 3   0.71501
  cuboid   4 4p0.9375
  media   4 1 1
  media 121 1 2 -1
  media 101 1 3 -2
  media 111 1 4 -3
  boundary    4  2 2
unit   12
  com="edge rod 1.69% enr"
  cylinder 1   0.60579
  cylinder 2   0.62103
  cylinder 3   0.71501
  cuboid   4 4p0.9375
  media 203 1 1
  media 121 1 2 -1
  media 101 1 3 -2
  media 111 1 4 -3
  boundary    4  2 2
unit   14
  com="edge rod 1.94% enr"
  cylinder 1   0.60579
  cylinder 2   0.62103
  cylinder 3   0.71501
  cuboid   4 4p0.9375
  media 202 1 1
  media 121 1 2 -1
  media 101 1 3 -2
  media 111 1 4 -3
  boundary    4  2 2
unit   17
  com="corner rod 1.69% enr"
  cylinder 1   0.60579
  cylinder 2   0.62103
  cylinder 3   0.71501
  cuboid   4 4p0.9375
  media 213 1 1
  media 121 1 2 -1
  media 101 1 3 -2
  media 111 1 4 -3
  boundary    4  2 2
unit   22
  com="interior rod 1.94% enr"
  cylinder 1   0.60579
  cylinder 2   0.62103
  cylinder 3   0.71501
  cuboid   4 4p0.9375
  media   2 1 1
  media 121 1 2 -1
  media 101 1 3 -2
  media 111 1 4 -3
  boundary    4  2 2
unit   23
  com="interior rod 2.93% enr"
  cylinder 1   0.60579
  cylinder 2   0.62103
  cylinder 3   0.71501
  cuboid   4 4p0.9375
  media   1 1 1
  media 121 1 2 -1
  media 101 1 3 -2
  media 111 1 4 -3
  boundary    4  2 2
unit   33
  com="interior gad pin"
  cylinder 1   0.270917524
```

(continues on next page)

(continued from previous page)

```
    cylinder 2   0.383135237
    cylinder 3   0.469242916
    cylinder 4   0.541835048
    cylinder 5   0.60579
    cylinder 6   0.62103
    cylinder 7   0.71501
    cuboid   8 4p0.9375
    media 500 1 1
    media 500 1 2 -1
    media 500 1 3 -2
    media 500 1 4 -3
    media 500 1 5 -4
    media 121 1 6 -5
    media 101 1 7 -6
    media 111 1 8 -7
    boundary    8  2 2
  unit   37
    com="edge rod 2.93% enr"
    cylinder 1   0.60579
    cylinder 2   0.62103
    cylinder 3   0.71501
    cuboid   4 4p0.9375
    media 201 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4  2 2
  unit   77
   com="corner rod 1.94% enr"
    cylinder 1   0.60579
    cylinder 2   0.62103
    cylinder 3   0.71501
    cuboid   4 4p0.9375
    media 212 1 1
    media 121 1 2 -1
    media 101 1 3 -2
    media 111 1 4 -3
    boundary    4  2 2
global unit 100
   cuboid  1   4p6.70306
   array 1 1  place 4 4  0.0 0.0
   cuboid  2   4p6.90626
   cuboid  10  4p7.62
   media   111 1 1
   media   630 1 2 -1
   media   620 1 10 -2
   boundary 10 32 32
end geom
read array
  ara=1 nux=7 nuy=7 typ=cuboidal
  fill
    17  14  37  37  37  14  77
    14  23  33  23  23  23  14
    14  23  23  23  33  23  37
    14  23  23  23  23  23  37
    12  23  33  23  23  33  37
    12  22  23  23  23  23  14
    11  12  12  14  14  14  17 end fill
end array
read bounds
  all=refl
end bounds
end model
end
```

3.1.6.11 TRITON6 sample problem 1: triton6-1.inp

Sample problem *triton6-1.inp* is an example of KENO-VI-based depletion for an infinite lattice of cylinders fabricated with particulate TRISO fuel dispersed in a graphite matrix. This provides an example of the cross section processing specification of a doubly heterogeneous (*DOUBLEHET*) media and use of the resultant homogenized media in a depletion calculation.

```
' THIS SAMPLE PROBLEM TEST THE FOLLOWING:
'  ** t6-depl sequence
'  ** v7-252 group library
'  ** centrm cross section processing
'  ** double-heterogeneous cross section processing option
'  ** deplete-by-constant power and flux
'  ** system power normalization
=t6-depl      parm=centrm
Test case - infinite cylinder
v7-252
read comp
' fuel kernel
  u-238  101 0 1.72877e-2 293.6 end
  u-235  101 0 5.92585e-3 293.6 end
  o      101 0 4.64272e-2 293.6 end
  b-10   101 0 1.14694e-7 293.6 end
  b-11   101 0 4.64570e-7 293.6 end
' first coating
  c      102 0 5.26449e-2 293.6 end
' inner pyro carbon
  c      103 0 9.52621e-2 293.6 end
' silicon carbide
  c      104 0 4.77240e-2 293.6 end
  si     104 0 4.77240e-2 293.6 end
' outer pyro carbon
  c      105 0 9.52621e-2 293.6 end
' graphite matrix
  c      106 0 8.77414e-2 293.6 end
' carbon pebble outer coating
  c      107 0 8.77414e-2 293.6 end
  b-10   107 0 9.64977e-9 293.6 end
  b-11   107 0 3.90864e-8 293.6 end
  he-3    108 0 3.71220e-11 293.6 end
  he-4    108 0 2.65156e-5 293.6 end
end comp

read celldata
  doublehet  right_bdy=white fuelmix=10 end
    gfr=0.025    101
    coatt=0.009  102
    coatt=0.004  103
    coatt=0.0035 104
    coatt=0.004  105
    matrix=106 numpar=15000 end grain
  rod squarepitch right_bdy=white hpitch=3.0 108 fuelr=2.5 cladr=3.0 107 fuelh=365 end
end celldata
read depletion
  101 flux 107
end depletion
read burndata
  power=30 burn=600 down=30 nlib=1 end
end burndata
read model
read param
  npg=200 gen=350 nsk=100 htm=no
end param
read geometry
  global unit 1
```

(continues on next page)

(continued from previous page)

```
    cylinder 1 2.5 99 -99
    cylinder 2 3.0 99 -99
    cuboid   3 4p3.0 99 -99
    media    10    1       1
    media     107    1       2  -1
    media     108    1       3  -2
  boundary     3
end geometry
read bounds
  all=mirror
end bounds
end data
end model
end
```

3.1.7 APPENDICES

3.1.7.1 XSDRN Model Block Description

The model data block for *T-XSDRN* and *T-DEPL-1D* calculations allows specification of the 1D geometry model and various control parameters used in the transport solution. The XSDRN *MODEL* block input is arranged in blocks of data that are similar to the NEWT *MODEL* block input described in Sect. 9.2. The XSDRN model input starts with an optional 80-character title, followed by a *PARAMETER* block, and then the following three data blocks in any order: the *GEOMETRY* data block, the *MATERIAL* data block, and the optional *COLLAPSE* data block. If the *PARAMETER*, *GEOMETRY*, and *MATERIAL* data block are not specified, an error message is printed and the problem is terminated. Sample input files for *T-XSDRN* and *T-DEPL-1D* calculations are provided in TRITON sample problems 2 and 3, respectively.

XSDRN* PARAMETER *block

PARAMETER Block keyword = parm, para, parameter, or parameters

Valid *PARAMETER* block specifications are described below. For each keyword, allowable values are listed in parentheses, and the default is listed in brackets. Input that can take an arbitrary integer value is indicated by an IN; similarly, any parameter that can take an arbitrary real/floating point value is indicated by RN as the allowable value. SCALE read routines allow the input of integers for real numbers, and vice versa, and the number will be converted accordingly. The order of the parameters within the block is arbitrary and may be skipped if a default value is desired for that parameter. If a parameter is listed multiple times, the final specified value is used.

bf=(RN) - Buckling factor, equal to twice the extrapolation distance multiplier used to determine the zero point of the asymptotic flux. [1.420892]

collapse=(yes/no) - If collapse=yes is specified, a flux-weighted collapse is performed by material number; cross sections for each nuclide in each material in the problem are collapsed to a specified (or default) group structure based on the average flux in that material. If collapse=yes, TRITON will look for the *COLLAPSE* block; if not found, TRITON will generate cross sections based on the original group structure. [no]

deltay=(RN) - The first transverse dimension in centimeters used in a buckling correction to calculate

leakage normal to the principal calculation direction (i.e., the height of a slab or a cylinder).

deltaz=(RN) - The second transverse dimension in centimeters used in a buckling correction (i.e., the width of a slab).

difftreatment=(mg_1d_sigtr/mg_0d_diff/mg_0d_sigtr/1g_0d_sigtr) - Diffusion treatment option for transverse leakage corrections. The mg_1d_sigtr option uses zone-dependent transport cross-sections for the transverse leakage correction. The mg_0d_diff option uses flux-volume-weighted homogenized diffusion coefficients. The mg_0d_sigtr option uses flux-volume weighted homogenized transport cross-sections. The 1g_0d_sigtr option uses a one-group homogenized transport cross-section. [1g_0d_sigtr]

epsglobal=(RN) - Overall problem convergence criteria. [1.0e-6]

epsouter=(RN) - Scalar flux convergence criteria. [1.0e-6]

inners=(IN) - Maximum number of inner iterations in an energy group. [20]

outers=(IN) - Maximum number of outer iterations. [100]

prtflux=(yes/no) - Flag indicating whether or not scalar flux values are should be printed in problem output. [no]

prtangflux=(yes/no) - Flag indicating whether or not angular flux values should be printed in problem output. [no]

prtbalnc=(yes/no) - Flag indicating whether or not fine-group material balance tables should be printed in problem output. [no]

prtmxsec=(yes/no/1d) - Flag indicating whether or not material macroscopic cross sections should be printed in problem output. The 1D option indicates that 2D scattering tables are not to be printed. [no]

sn=(2/4/6/8/16/32) - Sn quadrature order for the transport calculations.

XSDRN *GEOMETRY* block

GEOMETRY Block keyword = geom, geometry

The *GEOMETRY* block is used to specify the geometry type (e.g., slab, cylinder, or sphere), the boundary conditions, the 1D material mesh (i.e., zone mesh), and the 1D spatial mesh used in the transport calculation. The order of the parameters entered in the *GEOMETRY* block is arbitrary and can be any of the following supported keyword specifications or keyword array specifications.

geom=(slab/cylinder/sphere) - Problem geometry. Keywords geometry=, ige=, and cyl for cylinder are also allowed. [slab]

leftbc=(vaccum/periodic/white/albedo/mirror) - Left-hand boundary condition. Keywords ibl=, vac for vacuum, refl for mirror, and reflected for mirror are also allowed. [mirror]

rightbc=(vaccum/periodic/white/albedo/mirror) - Right-hand boundary condition. Keywords ibr=, vac for vacuum, refl for mirror, and reflected for mirror are also allowed. [mirror]

left_albedo RN1 RN2 ... RNN **end left_albedo -** The left-hand boundary albedo values as a function of energy group. The left_albedo array is ignored if leftbc= is vacuum, periodic, white, or mirror. If the left_albedo array is omitted and leftbc=albedo, white boundary conditions are used. If the number of entries in the left_albedo array does not equal the number of energy groups in the cross-section library, an error message is printed and the problem is terminated.

right_albedo RN1 RN2 ... RNN **end right_albedo -** The right-hand boundary albedo values as a function of energy group. The right_albedo array is ignored if rightbc= is vacuum, periodic, white, or mirror. If the right_albedo array is omitted and rightbc=albedo, white boundary conditions are used. If the number of entries in the right_albedo array does not equal the number of energy groups in the cross-section library, an error message is printed and the problem is terminated.

zoneids IN1 IN2 ... INN **end zoneids -** Material composition number by zone. The number of entries in the zoneids array defines the number of zones for the problem. If the zoneids array is not defined, an error message is printed and the problem is terminated.

zonedimensions RN1 RN2 ... RNN **end zonedimensions -** The right-hand boundary for each zone is given in centimeters. Note that the left-hand boundary of the first zone is 0.0 and must not be entered. If the zonedimensions array is not defined or the number of entries does not equal the number of entries in the zoneids array, then an error message is printed and the problem is terminated.

zoneintervals IN1 IN2 ... INN **end zoneintervals -** Number of spatial mesh of constant width per each problem zone. If specified, the number of entries of the zonedimensions array must equal the number of entries in the zoneids array. Otherwise, an error message is printed and the problem is terminated.

mesh RN1 RN2 ... RNN **end mesh -** The right-hand boundary for each spatial mesh in centimeters. The spatial mesh is the discretization used in the transport calculation. Note that the left-hand boundary of the first spatial mesh is 0.0 and must not be entered. The zone boundaries in the zonedimensions array must be a subset of the spatial mesh boundaries in the mesh array. Otherwise, an error message is printed and the problem is terminated. The mesh array is optional and is not used if the zoneintervals array is specified. If neither the zoneintervals array nor the mesh array is specified, an error message is printed and the problem is terminated.

XSDRN* MATERIAL *block

MATERIAL Block keyword = matl, mat, material, materials

The *MATERIAL* block is used to specify the material numbers for each material used in the calculation in the order of scattering cross section to be used for each material. The format of the *MATERIAL* block is identical to the NEWT *MATERIAL* block that is described in detail in (Sect. 9.2). Although source and description specifications are allowed, these options are not used by XSDRN.

XSDRN* COLLAPSE *block

COLLAPSE Block keyword = collapse, coll

The *COLLAPSE* block is used to define the energy group collapsing operation to calculate broad group cross-section libraries using the XSDRN flux solution. The format of the *COLLAPSE* block is identical to the NEWT *COLLAPSE* block that is described in detail in Sect. 9.2.

3.1.7.2 Data Structure for Cross Section Database File xfile016

When branch calculations are performed, TRITON archives collapsed homogenized cross sections in an unformatted, direct-access FORTRAN file called *xfile016*. The contents and format of this file are described in this appendix.

TRITON uses a library of SCALE subroutines to read and write blocks of data to direct-access FORTRAN files. The SCALE subroutine library allows the blocks of data to have variable length, even though direct-access FORTRAN files have a fixed record length. The data blocks can be retrieved from the file at random, provided the block length and block starting record position are known. The block length is expressed in terms of 4-byte words. For example, a block of 3-group macroscopic cross sections that contained the total, fission, capture, chi, and nubar cross sections would have a block length of 15 (3×5), assuming that the cross sections are stored in single precision 4-byte format.

The *xfile016* file supports 11 different block types. The first seven block types appear only once in the file, each block type occupying one of the first seven record positions. The remaining four block types, types 8-11, are repeated for each branch, at each depletion step, starting at the eighth record position.

Branch-specific blocks, i.e., block types 8-11, are written in the following order, for N branch calculations over M depletion steps:

First (t=0) transport calculation, branch 0 (reference state)

First (t=0) transport calculation, branch 1

First (t=0) transport calculation, branch 2

...

First (t=0) transport calculation, branch N

Second transport calculation, branch 0 (reference state)

Second transport calculation, branch 1

Second transport calculation, branch 2

...

Second transport calculation, branch N

...

...

...

$(M + 1)^{th}$ transport calculation, branch 0 (reference state)

$(M + 1)^{th}$ transport calculation, branch 1

$(M + 1)^{th}$ transport calculation, branch 2

...

$(M + 1)^{th}$ transport calculation, branch N

Note that $(M + 1) \times (N + 1)$ sets are saved for M depletion steps and N branches. For each set, block types 8 and 9 are always written, whereas block types 10 and 11 are written only if pin data output was requested ($nx \neq 0$).

Block Type 1: block length data

Length: 13

Position: 1

Type: integer.

Data: datlen(13)

datlen(1) Length of block type 1 (this array), which is 13.

datlen(2) Number of blocks allocated for this file (1000). Currently not used.

datlen(3) Length of FORTRAN record for this file (512).

datlen(4) Length of block type 2: general dimensioning data.

datlen(5) Length of block type 3: depletion data.

datlen(6) Length of block type 4: branching data.

datlen(7) Length of block type 5: branching data for advanced branch block (not yet supported).

datlen(8) Length of block type 6: currently not used.

datlen(9) Length of block type 7: energy group boundaries.

datlen(10) Length of block type 8: cross sections and misc data.

datlen(11) Length of block type 9: corner discontinuity factors.

datlen(12) Length of block type 10: pin power factors.

datlen(13) Length of block type 11: groupwise form factors.

Block Type 2: general dimensioning data

Length: datlen(4)

Position: 2

Type: integer, unless specified otherwise

Data: brnchdepl, nobranch, nsets, igm, iftg, ndelay, nadf, ncdf, ipin, nxpin, nypin, ivers, adftype, branchflag

brnchdepl Number of depletion steps + 1.

nobranch Number of branches.

nsets Number of cross-section sets on library (typically 1).

igm Number of energy groups in collapsed cross sections.

iftg First thermal energy group (max upscatter group).

ndelay Number of delayed neutron precursor groups (6).

nadf Number of assembly discontinuity factors (ADFs).

ncdf Number of corner discontinuity factors (CDFs).

ipin Flag for pin data (0 = no pin data, 1 = pin data included).

nxpin Number of pins in x-direction (0 if ipin = 0).

nypin Number of pins in y-direction (0 if ipin = 0).

ivers Format version number. This appendix describes version 5 of the database structure.

adftype ADF type: (1= single-assembly, 2= reflector, 3= single-assembly on arbitrary grid lines).

branchflag (logical) TRUE for simple BRANCH block format, FALSE for advanced format.

Block Type 3: depletion data

Length: datlen(5)

Position: 3

Type: real

Data: burnup(brnchdepl), time(brnchdepl), power(brnchdepl), sysHMdens

burnup(brnchdepl) Burnup (GWd/MTHM) at each transport step.

time(brnchdepl) Cumulative cycle time (days) at each transport step.

power(brnchdepl) Specific power (MW/MTHM) at each transport step.

sysHMdens System heavy metal mass density (g/cm^3).

Block Type 4: branching data

Length: datlen(6)

Position: 4

Type: integer, unless specified otherwise

Data: fuelused, modused, crused, fuelcount, modcount, crcount, crref, tfref, tmref, mdref, sbref, fuelmix(fuelcount), modmix(modcount), crinmix(crcount), croutmix(crcount), crstate(nobranch), tfuel(nobranch), tmod(nobranch), dmod(nobranch), sboron(nobranch)

fuelused (logical) TRUE if fuel mixtures were specified for branches.

modused (logical) TRUE if moderator mixtures were specified for branches.

crused (logical) TRUE if control rod mixtures were specified for branches.

fuelcount Number of mixtures in fuel definition.

modcount Number of mixtures in moderator definition.

crcount Number of mixture pairs in control rod definition.

crref Reference control rod state (0/1).

tfref (real) Reference fuel temperature (K).

tmref (real) Reference moderator temperature (K).

mdref (real) Referenced moderator density (g/cm^3).

sbref (real) Reference soluble boron concentration (ppm).

fuelmix(fuelcount) Mixtures defined as fuel.

modmix(modcount) Mixtures defined as moderator.

crinmix(crcount) Mixtures defined for the control-rod in state.

croutmix(crcount) Mixtures defined for the control-rod out state.

crstate(nobranch) Control rod state (0=withdrawn/1=inserted) for each branch.

tfuel(nobranch) (real) Fuel temperature (K) for each branch.

tmod(nobranch) (real) Moderator temperature (K) for each branch.

dmod(nobranch) (real) Moderator density (g/cm^3) for each branch.

sboron(nobranch) (real) Soluble boron concentration (ppm) for each branch.

Block Type 5: advanced branching data

Length: datlen(7)

Position: 5

Type: integer

Data: Stores data for advanced branch block (not yet supported)

Block Type 6: currently not used

Block Type 7: energy group boundary data

Length: datlen(9)

Position: 7

Type: real

Data: ebnds(igm+1)

ebnds(igm+1) Energy group boundaries

Blocks 1–7 are written only once. Blocks 8 and 9 (plus 10 and 11 if pin power data is output) are written for each branch case at each depletion step.

Block Type 8: cross-section data

Length: datlen(10)

Position: 8 + (igm + 3) [i * (nobranch + 1) + j]) , i = 0,. . . , brnchdepl, j = 0,. . . , nobranch

Type: real

Data: {kinf(i), beta_eff(1:ndelay, i), lam_eff(1:ndelay, i) , y_i135(i), y_xe135(i), y_pm149(i), id(i), nden(i), aden(i), [sigt(i,j), siga(i,j), xemac(i,j), smmac(i,j), sigc(i,j), sigf(i,j), sign2n(i,j), sigtr(i,j), nusigf(i,j), kappaf(i,j), nu(i,j), chi(i,j), diffcoef(i,j), flux(i,j), sigselas(i,j), sig_xe(i,j), sig_sm (i,j), detfis(i,j), detflx(i,j), invvel(i,j), sigtr2(i,j), sigtr(i,j), [(adf(i,j,k), k=1,nadf),(0,k=nadf+1,12), (current(i,j,k), k=1,nadf),(0,k=nadf+1,12)], (sigs(i,j,k), k=1,igm), j=1,igm], i=1,nsets}

Data is saved for i = 1,nsets (number of homogenized regions):

kinf(i) k-infinity

beta_eff(1:ndelay,i) Approximate delayed neutron fractions.

lam_eff(1:ndelay,i) Approximate delayed neutron decay constants (sec^{-1}).

y_i135(i) Fission product yield for ^{135}I.

y_xe135(i) Fission product yield for ^{135}Xe.

y_pm149(i) Fission product yield for ^{149}Pm.

Data is saved for j = 1, igm (number of energy groups):

sigt(i,j) Total cross section (cm^{-1}).

siga(i,j) Effective absorption cross section (cm^{-1}).

xemac(i,j) Macroscopic ^{135}Xe cross section (cm^{-1})

smmac(i,j) Macroscopic ^{149}Sm cross section (cm^{-1}).

sigc(i,j) Capture cross section (cm^{-1}).

sigf(i,j) Fission cross section (cm^{-1}).

sign2n(i,j) Effective n2n cross section (cm^{-1}).

sigtr(i,j) Transport cross section (cm^{-1}), determined by outscatter approximation.

nusigf(i,j) Average total number of neutrons/fission $\times$ fission cross section (cm^{-1}).

kappaf(i,j) Energy released per capture × capture cross section + Energy released per fission × fission cross section (J/cm).

nu(i,j) Average total number of neutrons released per fission (delayed + prompt).

chi(i,j) Fission spectrum (delayed + prompt).

diffcoef(i,j) Diffusion coefficient (cm), 1 / (3 × sigtr(i,j)).

flux(i,j) Average flux (n/cm^2-sec).

sigselas(i,j) Total elastic scattering cross section (cm^{-1}).

sig_xe(i,j) Microscopic cross section for ^{135}Xe (barns).

sig_sm (i,j) Microscopic cross section for ^{149}Sm (barns).

detfis(i,j) Microscopic ^{235}U cross section at detector location (barns).

detflx(i,j) Average flux in detector mixture (n/cm^2-sec).

invvel(i,j) Inverse neutron velocity (sec/cm).

sigtr2(i,j) Transport cross section (cm^{-1}), determined by inscatter approximation.

sigtr(i,j) Transport cross section (cm^{-1}), determined by outscatter approximation.

adf(1:nadf,i,j) Assembly discontinuity factors for up to 12 faces.

current(1:nadf,i,j) Net current for up to 12 faces (n/cm^2-sec), adftype = 3 only.

sigs(i,j,k), k=1,igm Macroscopic scattering cross section, j k (cm^{-1}).

End of data saved for j = 1, igm

End of data saved for i = 1,nsets

Block Type 9: corner discontinuity factors

Length: datlen(11)

Position: 9 + (igm + 3) [i * (nobranch + 1) + j]) , i = 0,. . . , brnchdepl, j = 0,. . . , nobranch

Type: real

Data: ((cdf(i,j), i=1,ncdf), j=1,igm)

Data is saved for i = 1,ncdf (number of “corner” discontinuity factors):

Data is saved for j = 1, igm (number of energy groups):

cdf(i,j) Corner discontinuity factors

End of data saved for j = 1, igm

End of data saved for i = 1,ncdf

Block Type 10: pin power peaking factors

Length: datlen(12)

Position: 10 + (igm + 3) [i * (nobranch + 1) + j]) , i = 0,. . . , brnchdepl, j = 0,. . . , nobranch

Type: double precision

Data: ((ppf(i,j), i=1,nx), j=1,ny)

Data is saved for j = 1, ny (number of pins in y direction):

Data is saved for i = 1,nx (number of pins in x direction):

ppf(i,j) Pin power (peaking) factors

End of data saved for i = 1, nx

End of data saved for j = 1, ny

Block Type 11: group form factors

Length: datlen(13)

Position: 10 + k + (igm + 3) [i * (nobranch + 1) + j]) ,

k = 1,. . . , igm, i = 0,. . . , brnchdepl, j = 0,. . . , nobranch

Type: double precision

Data: ((gff(i,j), i=1,nx), j=1,ny)

Data is saved for j = 1,ny (number of pins in y direction):

Data is saved for j = 1, nx (number of pins in x direction):

gff(i,j,k) Groupwise form factors

End of data saved for i = 1, nx

End of data saved for j = 1, ny

NOTE: Block Type 11 is repeated igm times where igm is the number of energy groups.

It is recommended that code written to process *xfile016* include the SCALE subroutine library. Although possible to link in the appropriate files in the scalelib object library in SCALE, it may be more practical to copy the appropriate SCALE routines into a new FORTRAN code used in reading *xfile016*. All direct-access operations needed to operate on this file are contained in the file direct_access_M.f90 in the scale src/scalelib directory. This file has dependencies and requires the following additional subroutines, all in the `src/scalelib` directory, in order to compile:

Error_functions_M.f90

common_unit_C.f90

Vast_kind_param_M.f90

separator_character_M.f90

Y0trns_M.f90

f_exit.c

The single C routine can be eliminated by eliminating the call to f_exit in subroutine stop of Error_function.f90, e.g., change

```
if ( stopcode == 0 ) return
write(npr,'(1x,a,i10)') 'stop code ',stopcode
call f_exit(npr)

end subroutine stop
```

to

```
if ( stopcode == 0 ) return
write(npr,'(1x,a,i10)') 'stop code ',stopcode
write(standard_output,'(a)')npr
stop

end subroutine stop
```

Alternatively, one can utilize the module listed on the following pages, developed by Mr. Benjamin Collins of the University of Michigan, which includes all necessary coding wrapped into a single Fortran module. Although developed from SCALE 5.1 routines, the format of SCALE direct access does not change and this source should remain compatible with later versions of SCALE.

```
module direct_access
!     Module taken from SCALE 5.1 source code and modified to eliminate
!     dependencies to other scale modules
!     Ben Collins, Doctoral Candidate, University of Michigan

     implicit none

      private
      integer,private,parameter::number_of_units=99
      integer,private:: nblks(number_of_units),lblks(number_of_units),char_word(number_of_units)
      integer,private :: record_length
      integer, parameter :: dp = selected_real_kind(14)
      integer,public :: next(3), nexsav(3), nda
      character(len=1) :: separator='/'
! ***change separator character to backslash ('\') for Windows***
!      character(len=1) :: separator='\'
      public :: openda, xtenda, closda, inquir
      public :: reed
!
!
!  set chpwrd to 1 now so that everything is specified in characters rather than
!      in words when reading or writing character arrays
!
      integer,public,parameter:: chpwrd=1
!
    interface reed
      module procedure real_reed, integer_reed, dp_reed
    end interface
```

```
    contains
!
      subroutine openda ( nblk,lblk,type,nrr,nunit,optional_name )
!
      integer                   :: nblk,lblk,nrr,nunit
      real,dimension(lblk)      :: a
      character(len=1)          :: type
      character(len=*),optional :: optional_name
      character(len=16)         :: filnam
      character(len=512)        :: dsname
      character(10)             :: action
      logical                   :: lopen
      integer                   :: i, record_length
!
```

(continues on next page)

(continued from previous page)

```
      if ( nunit <= 0 .or. nunit >= 100 ) then
        stop 'da error - invalid unit number: program will terminate.'
      else
        inquire(unit=nunit,opened=lopen)
        if ( lopen ) then
          stop 'da error - unit already open: program will terminate.'
        end if
      end if
```

```
!
      inquire(iolength=record_length) a
      write(filnam,'(a,i3.3,a8)') 'xfile',nunit,' '
      if ( present(optional_name) ) filnam = optional_name
      if ( type == 'o' .or. type == 'w' ) then
        call fulnam(filnam,dsname)
        select case (type)
        case('o')
          action = 'read'
        case('w')
          action = 'readwrite'
        end select
        open(unit=nunit,access='direct',status='old',action=action, &
             form='unformatted',recl=record_length,file=dsname)
        nblks(nunit) = 999999
        lblks(nunit) = lblk
        inquire(unit=nunit,opened=lopen)
        if (.not.lopen) then
          stop 'da error - unable to open unit: program will terminate.'
        end if
      else
        nblks(nunit) = nblk
        lblks(nunit) = lblk
        open(unit=nunit,access='direct',status='replace', &
        form='unformatted',recl=record_length,file=filnam)
        inquire(unit=nunit,opened=lopen)
        if (.not.lopen) then
          stop 'da error - unable to open unit: program will terminate.'
        end if
      end if
      char_word(nunit) = record_length/lblk

      end subroutine openda
```

```
!

      subroutine closda ( nunit )
!
      integer:: nunit
      logical:: lopen
!
      inquire(unit=nunit,opened=lopen)
      if (lopen) close(unit=nunit)
      nblks(nunit) = 0
      lblks(nunit) = 0

      end subroutine closda

!

      subroutine real_reed ( x,lword,nunit,nrec )
!
      integer::lword,nunit,nrec
      real,dimension(lword)::x
      integer::lb,nb,nr,no,i,nl,j
```

(continues on next page)

(continued from previous page)

```
!
      call check_unit(nunit, lword)
      lb     = lblks(nunit)
      nb     = (lword+lb-1)/lb
      nr     = nrec
      no     = 1
      do i=1,nb
        if ( nr <= 0 .or. nr > nblks(nunit) ) then
          call print_rel_blk ( nunit, nr )
        end if
        nl     = min(no+lb-1,lword)
        read (nunit,rec=nr) (x(j),j=no,nl)
        nr     = nr + 1
        no     = nl + 1
      end do

      end subroutine real_reed
```

```
!

      subroutine integer_reed ( nnx,lword,nunit,nrec )
!
      integer::lword,nunit,nrec
      integer,dimension(lword)::nnx
      integer::lb,nb,nr,no,i,nl,j
!
      call check_unit(nunit, lword)
      lb     = lblks(nunit)
      nb     = (lword+lb-1)/lb
      nr     = nrec
      no     = 1
      do i=1,nb
        if ( nr <= 0 .or. nr > nblks(nunit) ) then
          call print_rel_blk ( nunit, nr )
        end if
        nl     = min(no+lb-1,lword)
        read (nunit,rec=nr) (nnx(j),j=no,nl)
        nr     = nr + 1
        no     = nl + 1
      end do

      end subroutine integer_reed
```

```
!

      subroutine dp_reed ( x,lword,nunit,nrec )
!
      integer::lword,nunit,nrec
      real(dp),dimension(:)::x
      integer::lb,nb,nr,no,i,nl,j,lwrd
!
      lwrd   = ubound(x,1)
      call check_unit(nunit, lwrd)
      lb     = lblks(nunit)/2
      nb     = (lwrd+lb-1)/lb
      nr     = nrec
      no     = 1
      do i=1,nb
        if ( nr <= 0 .or. nr > nblks(nunit) ) then
          call print_rel_blk ( nunit, nr )
        end if
        nl     = min(no+lb-1,lwrd)
        read (nunit,rec=nr) (x(j),j=no,nl)
        nr     = nr + 1
```

(continues on next page)

(continued from previous page)

```
        no     = nl + 1
      end do

      end subroutine dp_reed
```

```
!

      subroutine inquir ( nunit,nrec )
!
      integer::nunit,nrec
!
      inquire(unit=nunit,nextrec=nrec)

      end subroutine inquir

!

      subroutine xtenda ( mblk,nunit )
         integer::mblk,nunit
         integer::lblk
         lblk = lblks(nunit)
         nblks(nunit) = nblks(nunit) + mblk
      end subroutine xtenda

!

      subroutine check_unit(nunit, lword)

      integer :: nunit, lword
      logical :: lopen
      character(len=10)::access
!
      inquire(unit=nunit,opened=lopen,access=access)
      if (.not.lopen) then
        stop 'da error - unit not open: program will terminate.'
      else
        if ( lword <= 0 ) then
          stop 'da error - invalid block length: program will terminate.'
        end if
      end if

      end subroutine check_unit
```

```
!

      subroutine print_rel_blk ( unit, block )

      integer :: unit, block
      stop 'da error - relative block not in data set: program will terminate.'

      end subroutine print_rel_blk

        subroutine fulnam ( filnam, name )

!   routine to convert a simple file name to a full path

      character(len=*)   :: filnam
      character(len=512) :: data_path
      character(len=4)   :: data='DATA'
      character(len=512) :: current_path
      character(len=6)   :: curdir='PWD'
      character(len=16)  :: short_name
      character(len=512) :: name, data_path_name, current_path_name, full_path_name
      logical            :: exists, found
```

(continues on next page)

(continued from previous page)

```
      integer            :: n99=99, iostat

!   check if filnam already has path
      if (index(filnam(1:3),separator) > 0 ) then
         name = filnam
         return
      end if
!   get the scale data and tmpdir directory paths from environmental variables
      data_path          = ' '
      current_path       = ' '
      data_path_name     = filnam
      current_path_name  = filnam
      call getenv ( data, data_path )
      call getenv ( curdir, current_path )
```

```
!   construct the full path name for the dataset name
      if (    data_path /= ' ' )     data_path_name = (trim(data_path))//separator//filnam
      if ( current_path /= ' ' ) current_path_name  = (trim(current_path))//separator//filnam

!   if the dataset exists in the current directory (tmpdir), use it
!   otherwise, look for it in the data directory
      inquire (file=filnam,exist=exists)
      if ( exists ) then
            name = current_path_name
      else
!   check names constructed in script
          inquire (file='data_directory',exist=exists)
          found = .false.
          if ( exists ) then
             open(n99,status='old',form='formatted',file='data_directory')
             rewind n99
             do
                read (n99,*,iostat=iostat) short_name, full_path_name
                if ( iostat /= 0 ) exit
                if ( short_name == filnam ) then
                   name  = full_path_name
                   found = .true.
                end if
             end do
          end if
          close (n99)
          if ( found ) return
          inquire (file=data_path_name,exist=exists)
          if ( exists ) then
             name = data_path_name
          else
             name = current_path_name
          end if
      end if

      end subroutine fulnam

      end module direct_access
```

3.1.7.3 The Flexible Branch Block

In support of various projects, the "flexible branch block" was developed to enable a broader set of perturbations than are available in the typical TRITON branch capability. The typical branch block allows the user to define a single set of mixtures for 'fuel,' 'mod,' 'crout', or 'crin'. Having only four material set definitions limits user's ability to specify more complex perturbations that may be possible in some reactors, especially under transient conditions. The flexible branch block was developed such that the user can specify any number of material sets, and then apply separate perturbations to those sets. This capability, for example, enables specification of bypass flow density branches in BWRs in which the in-channel coolant and out-channel moderator can set to different densities in the same branch calculation.

The flexible branch block was developed in the SCALE 6.1 implementation of TRITON and was not modernized for SCALE 6.2. As a result, the flexible branch block is available in SCALE 6.1 and in the legacy mode in SCALE 6.2. The legacy mode can be accessed using *t-d* as the sequence name, rather than the more typical *t-depl*. The flexible branch block can be accessed using **branchblock** as the block name, rather than **branch** that is used for the typical branch block.

The following section of the manual explains the syntax of the **branchblock** and contains short examples of each element within the the **branchblock**. At the end of this section, a full example of a **branchblock** is provided so that users can gain an understanding of how to use all of the parts of the **branchblock** in order to define needed calculation branches.

SYNTAX:

```
read branchblock
  [block keyword specifications]
end branchblock
```

The advanced **branchblock** supports five different keyword specifications described below.

- **mixset** — used to define a set of mixtures which can be used in **swap** and **perturbset** definition,
- **systemchange** — used to define a system change to, temperatures, nuclide concentrations, and Dancoff factors,
- **swap** - used to define a set of mixtures to swap,
- **perturbset** — used to define a set of perturbations which apply the system changes defined by **systemchange** to a set of mixtures, and
- **branch** — used to define a branch calculation, composed of various **swaps** and **perturbsets**. Additional perturbations may also be defined.

Note: Several keywords in the*__branchblock__* are defined using strings. These strings must be must be delimited, i.e. starts and ends with an identifying marker. (Examples: `title="cold"`, `title=#hot Doppler#`, `title=!40%void!`, `title=(80%void)`). As shown in the following examples, the string can optionally start with open angle bracket < and end with a closing angle bracket > (Example: `title=<cold>`). All string-value inputs in the ***branchblock*** are delimited, alphanumeric strings with a maximum length of 80 characters. It is recommended that users choose a single type of delimiter, and then use that delimiter throughout the **branchblock**

---

systemchange

SYNTAX:

```
read branchblock
  [...]
  systemchange title
    [systemchange keyword specifications]
  end systemchange
  [...]
end branchblock
```

systemchange supports the following keyword specifications:

```
title
dancoff=(real value)
temperature=(real value)
dendiv N1 f1 N2 f2 end
denmult N1 f1 N2 f2 end
```

title is required string input and must follow **systemchange**. Only one title keyword may be specified. Multiple **systemchange** specifications are allowed, so each specification must have a unique **title**.

dancoff is optional and is used to set a dancoff factor value in the interval [0,1]. Only one dancoff specification is allowed and can appear anywhere in the **systemchange** specification following the title.

temperature is optional and is used to set a system temperature in Kelvin. It must be nonnegative. Only one temperature specification is allowed and can appear anywhere in the **systemchange** specification following the title.

dendiv and **denmult** are keyword arrays used to define nuclide concentration dividers and multipliers respectively. The arrays must be terminated with the **end** keyword. Each array is defined by a series of nuclide/factor pairs where nuclide is the ZZZAAA identifier and factor is either a multiply or divide factor applied to that nuclide concentration (Note that the particular mixture for which the factor is applied is defined in the **perturb** specification described below). Multiply factors must be >=0. Divide factors must be >0. A nuclide identifier set to zero implies that the factor is applied to all nuclides that are not explicitly listed in the array. Multiple dendiv and denmult arrays are allowed and can appear anywhere in the **systemchange** specification following the title. TRITON applies the concentration factors in the order in which they are entered in the systemchange specification.

Multiple **systemchange** specifications are allowed in the branch block. They can appear in any order, but must have a unique title.

EXAMPLE:

Define a temperature change to 60 kelvin. (The temperature change will be applied to a set of mixtures defined in the **perturbset** specification defined later.)

```
systemchange <60C>
  temperature=333.15
end systemchange
```

swap

SYNTAX:

```
read branchblock
  [...]
  swap title
    [swap keyword specifications]
  end swap
  [...]
end branchblock
```

swap supports the following keyword specifications:

```
title
group1 [mixture specifications] end
group2 [mixture specifications] end
```

title is required string and must follow **swap**. Only one title keyword may be specified. Multiple **swap** specifications are allowed, so each specification must have a unique **title**.

group1 and **group2** are used to define a set of mixtures to exchange. **group1** must follow the **swap** title. **group2** must follow **group1**. Only one specification for each group is allowed and they must have the same number of mixtures.

The **group1** and **group2** keywords support the following keyword specifications:

```
mixture=(integer value)
mixtures I1 I2 ... IN end
mixset=(string value)
```

mixture is used to define a single mixture. **mixtures** is used to define an array of mixtures and is terminated with the **end** keyword. **mixset** is used to substitute a **mixset** specification defined elsewhere in the branchblock. Multiple **mixture**, **mixtures**, and **mixset** are allowed and can be placed in any order. TRITON will remove any duplicated mixture identifier, however each mixture must be defined in the model input.

EXAMPLES:

Exchange material 1 for 4.

```
swap <1 for 4>
  group1 mixture=1 end
  group2 mixtures 4 end end
end swap
```

Exchange a set of mixtures:

```
swap <RodInsertion>
  group1 mixset=<crout> end
  group2 mixset=<crin> end
end swap
```

branch

SYNTAX:

```
read branchblock
  [...]
  branch title
    [branch keyword specifications]
  end branch
  [...]
end branchblock
```

branch supports the following keyword specifications.

```
title
swap=(string value)
perturbset=(string value)
perturb [perturb specification] end
```

title is required string and must follow **branch**. Only one title keyword may be specified. Multiple **branch** specifications are allowed, so each specification must have a unique **title**.

swap is used to swap different sets of mixtures. The swap value is a string which is the title of a **swap** specification defined elsewhere in the branchblock. (The **swap** specification is described below). Multiple **swap** specifications are allowed and can appear anywhere in the **branch** specification following the title.

perturbset is used to apply a series of system perturbations. The perturbset value is a string which is the title of a **perturbset** specification defined elsewhere in the branchblock. (The **perturbset** specification is described below). Multiple **perturbset** specifications are allowed and can appear anywhere in the **branch** specification following the title.

perturb is used to apply a system perturbation that is not defined through the use of a **perturbset** specification. **perturb** specifications must terminate with the **end** keyword.

perturb supports the following keyword specifications.

```
change=(string value)
mixture=(integer value)
mixtures I1 I2 ... IN end
mixset=(string value)
```

change is a string which is the title of a **systemchange** specification defined elsewhere in the branchblock. Only one **change** specification is allowed and may appear anywhere in the **perturb** specification.

The system change is applied to a set of mixtures defined by the **mixture**, **mixtures**, and **mixset** specifications. Only one of each of these keywords is allowed (however all three may be used in the same **perturb** specification). **mixture**, **mixtures**, and **mixset** may be placed in any order. TRITON will remove any duplicated mixture, however each mixture must be defined in the model input. TRITON will perform **swap** and **perturb** operations in the order they appear in the input.

EXAMPLES:

Define a branch to charactize the rodded, cold-zero-power condition. This requires the use of mixture swap entitled <CRodIn> along with the perturbset definition <ColdMod> which perturbs all of the moderator mixtures to a cold temperature and density. The fuel mixtures (defined as <FuelMix>) must also be set to a temperature of 300K.

```
read branchblock
  [...]  (contains definitions for <CRodIn>, <FuelMix>, and <ColdMod>)
  branch <CZP,rodded>
    perturbset=<ColdMod> swap=<CRodIn>
    perturb change=<300K> mixset=<FuelMix> end
  end branch
  systemchange <300K> temperature=300 end systemchange
end branchblock
```

Define a branch to characterize the BWR instantaneous 100% void branch. This requires that:

- in-channel moderator mixtures (<ChannelMod>) are perturbed from 40% void to 100% void (defined by systemchange <40vf-100vf>).

- Water-rod moderator mixtures (<WaterRodMod>) are perturbed from 0% void to 5% void (<0vf-5vf>)
- Bypass moderator mixtures (<BypassMod>) are perturbed from 0% void to 3% void (<0vf-3vf>)
- Corner Rod Fuel mixture (mixture 1) dancoff factor changes (described by <100vf-cornerDF>)
- Edge Fuel Rod Mixtures (3,4,5,6,7,10) dancoff factor changes (described by <100vf-edgeDF>)

```
read branchblock
  [...]   (contains all other definitions)
  branch <100VF>
    perturb change=<40vf-100vf> mixset=<ChannelMod>  end
    perturb change=<0vf-5vf>    mixset=<WaterRodMod> end
    perturb change=<0vf-3vf>    mixset=<BypassMod> end
    perturb mixture=1 change=<100vf-cornerDF> end
    perturb mixtures 3 4 5 6 7 10 end change=<100vf-edgeDF> end
  end branch
end branchblock
```

mixset

mixset — used to define a set of mixtures used in **swap**, **perturbset**, and **perturb** specifications.

SYNTAX:

```
read branchblock
  [...]
  mixset title
    [mixset keyword specifications]
  end mixset
  [...]
end branchblock
```

mixset supports the following keyword specifications:

```
title
mixture=(integer value)
mixtures I1 I2 ... IN end
mixset=(string value)
```

title is required string and must follow **mixset**. Only one title keyword may be specified. Multiple **mixset** specifications are allowed, so each specification must have a unique **title**.

mixture is used to define a single mixture. **mixtures** is used to define an array of mixtures and is terminated with the **end** keyword. **mixset** is used to substitute a **mixset** specification defined elsewhere in the branchblock. Multiple **mixture**, **mixtures**, and **mixset** are allowed and can be placed in any order. TRITON will remove any duplicated mixture identifier, however each mixture must be defined in the model input. If **mixset** is used to, the mixture set must be *previously* defined in the branchblock.

EXAMPLE:

In previous example for 100% void fraction, define a mixture set to be used for the edge rod dancoff factor perturbation.

```
read branchblock
  [...]   !contains all other definitions
  branch <100VF>
    perturb change=<40vf-100vf> mixset=<ChannelMod>  end
    perturb change=<0vf-5vf>    mixset=<WaterRodMod> end
    perturb change=<0vf-3vf>    mixset=<BypassMod> end
```

(continues on next page)

(continued from previous page)

```
    perturb mixture=1 change=<100vf-cornerDF> end
    perturb change=<100vf-edgeDF> mixset=<edge-fuel> end
  end branch
  mixset <edge-fuel>
    mixtures 3 4 5 6 7 10 end
  end mixset
end branchblock
```

perturbset — used to define a set of system perturbations that can be used in **branch** specifications.

SYNTAX:

```
read branchblock
  [...]
  perturbset title
    [perturbset keyword specifications]
  end perturbset
  [...]
end branchblock
```

perturbset supports the following keyword specifications:

```
title
perturb [perturb specification] end
```

title is required string and must follow **perturbset**. Only one title keyword may be specified. Multiple **perturbset** specifications are allowed, so each specification must have a unique **title**.

After title, multiple **perturb** specifications can be used to defined a set of perturbations. The **perturbset** can then be used in **branch** specifications to simplify the **branch** input. TRITON will apply the perturbations in the order in which they appear in the **perturbset** specification.

EXAMPLE:

In previous example for 100% void fraction, define a perturbset for the moderator perturbations, and a separate perturbset for the fuel perturbations.

```
read branchblock
  [...]   !contains all other definitions
  perturbset <modChange>
    perturb change=<40vf-100vf> mixset=<ChannelMod>  end
    perturb change=<0vf-5vf>    mixset=<WaterRodMod> end
    perturb change=<0vf-3vf>    mixset=<BypassMod> end
  end perturbset
  branch <100VF> perturbset=<modChange> perturbset=<fuelChange>  end branch
  perturbset <fuelChange>
    perturb mixture=1 change=<100vf-cornerDF> end
    perturb change=<100vf-edgeDF> mixset=<edge-fuel> end
  end perturbset
  mixset <edge-fuel>
    mixtures 3 4 5 6 7 10 end
  end mixset
end branchblock
```

branchblock Full Example

Because the **branchblock** input is so flexible, it may be difficult for users to know where to begin. For that reason, we have provided a sample **branchblock** that is typical to a BWR analysis. In this example, the open and close parentheses are using instead of angle brackets. In the example provided, the **branchblock** is separated into ### different sections: definition of **mixsets**, definition of **systemchanges**, definition of **swaps**,

definition of **perturbsets** (which are composed of multiple **systemchanges**), and definition of **branches**. This example may appear complicated, but in essence, it is quite straightforward. First, all of the mixture IDs in the problem are defined into logical **mixsets**. Then, other large **mixsets** are composed of the individual **mixsets**. The first two **systemchanges**, (1/nom) and (1/liq), are a very important items. These **systemchanges** are density divisors that divide the number densities of a specified moderator mixture by the nominal or liquid density, making the resulting density 1.0. Then, **systemchanges** that are density multipliers are specified as the actual density, which make the **branchblock** much easier to read and understand. By using the special density divisors, an almost identical **branchblock** can be use for different nominal densities — only the density specified in (1/nom) needs to be modified for a different nominal density.

Following the **systemchanges**, a number of **perturbsets** are defined to make multiple perturbations to the collant or moderator density. For example, the (00%Void353) **perturbset** shown below makes six changes: (1) divide all coolant (in-channel) mixtures by the nominal density, then (2) multiple all coolant mixtures by the specified density, (3) divide all liquid water moderator (out-channel) features by the saturated liquid density, then (4) multiple all liquid water features by the specified density, and then, (5) and (6) change the Dancoff factors to their appropriate values corresponding to the coolant and moderator densities.

```
perturbset (00%Void353)
 perturb mixset=(coolant)    change=(1/nom)             end
 perturb mixset=(coolant)    change=(00V-353)           end
 perturb mixset=(solidmod)   change=(1/liq)             end
 perturb mixset=(solidmod)   change=(00V-353)           end
 perturb mixset=(cornerfuel) change=(00VCold-dfC0)      end
 perturb mixset=(edgefuel)   change=(00VCold-dfE0)      end
end perturbset
```

To end the file, all branch calculations are specified in a single block using the previously defined **perturbsets**. Note that unlike the typical **branch** block, the flexible **branchblock** does not need the first branch to correspond to the nominal conditions. It is important to note that in the *xfile016* and *txtfile16* files, the branch conditions (moderator density, temperature, soluble boron, and CR state) will not be listed correctly in the file header as they are for the typical **branch** block. When using the **branchblock** input, TRITON no longer knows the condition for any given branch, however, the branch order specified in the input file is maintained in the *xfile016* and *txtfile16* files.

Also note that in the example provided, no soluble boron changes have been specified (as this is a BWR example). However, soluble poisons (boron or other), are also fairly straightforward to specify using the density divisors and density multipliers.

BWR branchblock Example

```
read branchblock
  mixset (1f127E)      mixtures   701                 end   end mixset
  mixset (1f127C)      mixtures   702                 end   end mixset
  mixset (1f169C)      mixtures   703                 end   end mixset
  mixset (1f169E)      mixtures   704                 end   end mixset
  mixset (1f194)       mixtures   705                 end   end mixset
  mixset (1f194C)      mixtures   706                 end   end mixset
  mixset (1f194E)      mixtures   707                 end   end mixset
  mixset (1f279)       mixtures   708                 end   end mixset
  mixset (1f279E)      mixtures   709                 end   end mixset
  mixset (1f279gd40)   mixtures   710 711 712 713 714  end   end mixset

  mixset (gap)       mixtures   800 801 802 803 804 805 806 807 808 809 end   end mixset
  mixset (clad)      mixtures   825 826 827 828 829 830 831 832 833 834 end   end mixset
  mixset (coolant)   mixtures   850 851 852 853 854 855 856 857 858 859 end   end mixset
```

(continues on next page)

(continued from previous page)

```
  mixset (mod1)       mixtures   1001                    end   end mixset
  mixset (can)        mixtures   1004                    end   end mixset
  mixset (cbpois)     mixtures   1002                    end   end mixset
  mixset (cbstru)     mixtures   1003                    end   end mixset
  mixset (cbclad)     mixtures   1005                    end   end mixset

  mixset (cbpoisout)  mixtures   1012                    end   end mixset
  mixset (cbstruout)  mixtures   1013                    end   end mixset
  mixset (cbcladout)  mixtures   1015                    end   end mixset

  mixset (allfuel)     mixsets  (1f127E)
                                (1f127C)
                                (1f169C)
                                (1f169E)
                                (1f194)
                                (1f194C)
                                (1f194E)
                                (1f279)
                                (1f279E)
                                (1f279gd40)             end   end mixset

  mixset (cornerfuel)  mixsets  (1f127C)
                                (1f169C)
                                (1f194C)                end   end mixset

  mixset (edgefuel)    mixsets  (1f127E)
                                (1f169E)
                                (1f194E)
                                (1f279E)                end   end mixset
```

```
mixset (solidmod)    mixsets  (mod1) (cbpoisout) (cbstruout) (cbcladout)       end   end mixset
  mixset (crin)        mixsets  (cbpois) (cbstru) (cbclad)                       end   end mixset
  mixset (crout)       mixsets  (cbpoisout) (cbstruout) (cbcladout)              end   end mixset
  mixset (allmod)      mixsets  (coolant)  (solidmod)                            end   end mixset

  systemchange (1/nom)      dendiv 0 0.4573 end      end systemchange
  systemchange (1/liq)      dendiv 0 0.7373 end      end systemchange

  systemchange (00V)       denmult 0 0.7373 end      end systemchange
  systemchange (40V)       denmult 0 0.4573 end      end systemchange
  systemchange (70V)       denmult 0 0.2473 end      end systemchange
  systemchange (90V)       denmult 0 0.1073 end      end systemchange
  systemchange (100V)      denmult 0 0.0373 end      end systemchange
  systemchange (00V-293)   denmult 0 0.9982 end      end systemchange
  systemchange (00V-313)   denmult 0 0.9922 end      end systemchange
  systemchange (00V-333)   denmult 0 0.9837 end      end systemchange
  systemchange (00V-353)   denmult 0 0.9718 end      end systemchange

  systemchange (293.15K)   temperature= 293.15    end systemchange
  systemchange (313.15K)   temperature= 313.15    end systemchange
  systemchange (333.15K)   temperature= 333.15    end systemchange
  systemchange (353.15K)   temperature= 353.15    end systemchange
  systemchange (300.00K)   temperature= 300.00    end systemchange
  systemchange (500.00K)   temperature= 500.00    end systemchange
  systemchange (1500.00K)  temperature=1500.00    end systemchange
  systemchange (560.29K)   temperature= 560.29    end systemchange
  systemchange (948.45K)   temperature= 948.45    end systemchange
```

```
' 0% void, cold Dancoff Factors
  systemchange (00VCold-dfC0) dancoff=0.084       end systemchange
  systemchange (00VCold-dfE0) dancoff=0.125       end systemchange
' 0% void, Dancoff Factors
  systemchange (00V-dfC0)     dancoff=0.116       end systemchange
  systemchange (00V-dfE0)     dancoff=0.171       end systemchange
```

(continues on next page)

(continued from previous page)

```
' 40% void, Dancoff Factors
  systemchange (40V-dfCO)     dancoff=0.180       end systemchange
  systemchange (40V-dfEO)     dancoff=0.256       end systemchange
' 70% void, Dancoff Factors
  systemchange (70V-dfCO)     dancoff=0.281       end systemchange
  systemchange (70V-dfEO)     dancoff=0.376       end systemchange
' 90% void, Dancoff Factors
  systemchange (90V-dfCO)     dancoff=0.421       end systemchange
  systemchange (90V-dfEO)     dancoff=0.524       end systemchange

  swap (cr)
    group1 mixset=(crout) end
    group2 mixset=(crin)  end
  end swap

  perturbset (00%Void293)
    perturb mixset=(coolant)    change=(1/nom)             end
    perturb mixset=(coolant)    change=(00V-293)           end
    perturb mixset=(solidmod)   change=(1/liq)             end
    perturb mixset=(solidmod)   change=(00V-293)           end
    perturb mixset=(cornerfuel) change=(00VCold-dfCO)      end
    perturb mixset=(edgefuel)   change=(00VCold-dfEO)      end
  end perturbset
  perturbset (00%Void313)
    perturb mixset=(coolant)    change=(1/nom)             end
    perturb mixset=(coolant)    change=(00V-313)           end
    perturb mixset=(solidmod)   change=(1/liq)             end
    perturb mixset=(solidmod)   change=(00V-313)           end
    perturb mixset=(cornerfuel) change=(00VCold-dfCO)      end
    perturb mixset=(edgefuel)   change=(00VCold-dfEO)      end
  end perturbset
  perturbset (00%Void333)
    perturb mixset=(coolant)    change=(1/nom)             end
    perturb mixset=(coolant)    change=(00V-333)           end
    perturb mixset=(solidmod)   change=(1/liq)             end
    perturb mixset=(solidmod)   change=(00V-333)           end
    perturb mixset=(cornerfuel) change=(00VCold-dfCO)      end
    perturb mixset=(edgefuel)   change=(00VCold-dfEO)      end
  end perturbset
  perturbset (00%Void353)
    perturb mixset=(coolant)    change=(1/nom)             end
    perturb mixset=(coolant)    change=(00V-353)           end
    perturb mixset=(solidmod)   change=(1/liq)             end
    perturb mixset=(solidmod)   change=(00V-353)           end
    perturb mixset=(cornerfuel) change=(00VCold-dfCO)      end
    perturb mixset=(edgefuel)   change=(00VCold-dfEO)      end
  end perturbset
```

```
perturbset (00%Void)
  perturb mixset=(coolant)    change=(1/nom)             end
  perturb mixset=(coolant)    change=(00V)               end
  perturb mixset=(cornerfuel) change=(00V-dfCO)          end
  perturb mixset=(edgefuel)   change=(00V-dfEO)          end
end perturbset
perturbset (40%Void)
  perturb mixset=(coolant)    change=(1/nom)             end
  perturb mixset=(coolant)    change=(40V)               end
  perturb mixset=(cornerfuel) change=(40V-dfCO)          end
  perturb mixset=(edgefuel)   change=(40V-dfEO)          end
end perturbset
perturbset (70%Void)
  perturb mixset=(coolant)    change=(1/nom)             end
  perturb mixset=(coolant)    change=(70V)               end
  perturb mixset=(cornerfuel) change=(70V-dfCO)          end
```

(continues on next page)

(continued from previous page)

```
  perturb mixset=(edgefuel)   change=(70V-dfE0)          end
end perturbset
perturbset (90%Void)
  perturb mixset=(coolant)    change=(1/nom)             end
  perturb mixset=(coolant)    change=(90V)               end
  perturb mixset=(cornerfuel) change=(90V-dfC0)          end
  perturb mixset=(edgefuel)   change=(90V-dfE0)          end
end perturbset

perturbset (Tf=293.15)
  perturb mixset=(allfuel)     change=(293.15K)           end
end perturbset
perturbset (Tf=313.15)
  perturb mixset=(allfuel)     change=(313.15K)           end
end perturbset
perturbset (Tf=333.15)
  perturb mixset=(allfuel)     change=(333.15K)           end
end perturbset
perturbset (Tf=353.15)
  perturb mixset=(allfuel)     change=(353.15K)           end
end perturbset
perturbset (Tf=948.45)
  perturb mixset=(allfuel)     change=(948.45K)           end
end perturbset
perturbset (Tf=500.00)
  perturb mixset=(allfuel)     change=(500.00K)           end
end perturbset
perturbset (Tf=1500.00)
  perturb mixset=(allfuel)     change=(1500.00K)          end
end perturbset

perturbset (Tm=293.15)
  perturb mixset=(allmod)     change=(293.15K)           end
end perturbset
perturbset (Tm=313.15)
  perturb mixset=(allmod)     change=(313.15K)           end
end perturbset
perturbset (Tm=333.15)
  perturb mixset=(allmod)     change=(333.15K)           end
end perturbset
perturbset (Tm=353.15)
  perturb mixset=(allmod)     change=(353.15K)           end
end perturbset
perturbset (Tm=560.29)
  perturb mixset=(allmod)     change=(560.29K)           end
end perturbset
```

```
'          Name                    Void Frac     Fuel Temp     Mod Temp           CR Pos
  branch   (branch  1) perturbsets (00%Void)      (Tf=948.45)   (Tm=560.29)   end              end branch
  branch   (branch  2) perturbsets (40%Void)      (Tf=948.45)   (Tm=560.29)   end              end branch
  branch   (branch  3) perturbsets (70%Void)      (Tf=948.45)   (Tm=560.29)   end              end branch
  branch   (branch  4) perturbsets (90%Void)      (Tf=948.45)   (Tm=560.29)   end              end branch
  branch   (branch  5) perturbsets (00%Void)      (Tf=948.45)   (Tm=560.29)   end  swap=(cr)   end branch
  branch   (branch  6) perturbsets (40%Void)      (Tf=948.45)   (Tm=560.29)   end  swap=(cr)   end branch
  branch   (branch  7) perturbsets (70%Void)      (Tf=948.45)   (Tm=560.29)   end  swap=(cr)   end branch
  branch   (branch  8) perturbsets (90%Void)      (Tf=948.45)   (Tm=560.29)   end  swap=(cr)   end branch
  branch   (branch  9) perturbsets (00%Void)      (Tf=500.00)   (Tm=560.29)   end              end branch
  branch   (branch 10) perturbsets (40%Void)      (Tf=500.00)   (Tm=560.29)   end              end branch
  branch   (branch 11) perturbsets (70%Void)      (Tf=500.00)   (Tm=560.29)   end              end branch
  branch   (branch 12) perturbsets (90%Void)      (Tf=500.00)   (Tm=560.29)   end              end branch
  branch   (branch 13) perturbsets (00%Void)      (Tf=1500.00)  (Tm=560.29)   end              end branch
  branch   (branch 14) perturbsets (40%Void)      (Tf=1500.00)  (Tm=560.29)   end              end branch
  branch   (branch 15) perturbsets (70%Void)      (Tf=1500.00)  (Tm=560.29)   end              end branch
  branch   (branch 16) perturbsets (90%Void)      (Tf=1500.00)  (Tm=560.29)   end              end branch
```

(continues on next page)

(continued from previous page)

```
  branch  (branch 17) perturbsets (00%Void293)   (Tf=293.15)   (Tm=293.15)   end             end branch
  branch  (branch 18) perturbsets (00%Void313)   (Tf=313.15)   (Tm=313.15)   end             end branch
  branch  (branch 19) perturbsets (00%Void333)   (Tf=333.15)   (Tm=333.15)   end             end branch
  branch  (branch 20) perturbsets (00%Void353)   (Tf=353.15)   (Tm=353.15)   end             end branch
  branch  (branch 21) perturbsets (00%Void293)   (Tf=293.15)   (Tm=293.15)   end  swap=(cr)  end branch
  branch  (branch 22) perturbsets (00%Void313)   (Tf=313.15)   (Tm=313.15)   end  swap=(cr)  end branch
  branch  (branch 23) perturbsets (00%Void333)   (Tf=333.15)   (Tm=333.15)   end  swap=(cr)  end branch
  branch  (branch 24) perturbsets (00%Void353)   (Tf=353.15)   (Tm=353.15)   end  swap=(cr)  end branch
end branchblock
```

3.2 POLARIS - 2D LIGHT WATER REACTOR LATTICE PHYSICS MODULE

M. A. Jessee, W. A. Wieselquist, A. M. Holcomb, S. W. Hart, J. W. Bae, K. S. Kim, C. A. Gentry

ABSTRACT

Polaris is an easy-to-use light water reactor (LWR) lattice physics capability for SCALE. Originally released in SCALE 6.2, Polaris uses the Embedded Self Shielding Method (ESSM) for multigroup cross section processing and a transport solver based on the Method of Characteristics (MOC). The ESSM computes multigroup self-shielded cross sections using Bondarenko interpolation methods. The background cross section used in the interpolation is determined by iterative 2D MOC fixed-source transport calculations. Polaris is integrated with ORIGEN for depletion calculations. Each pin-or each radial subregion of the pin-is depleted based on the local power distribution. An optional critical spectrum calculation is incorporated into the depletion calculation and the output edits of few-group homogenized cross sections. Few-group cross sections are archived to the lattice physics archive (`.t16` and `.x16` extension) file, which can be used in subsequent core simulator calculations. Nuclide inventories are archived to the ORIGEN binary concentration file (`.f71` extension), which can be used in subsequent SCALE calculations for source term characterization or radition shielding. Depletion and decay data are archived to the ORIGEN binary library file (`.f33` extension), and can also be used in subsequent source term calculations.

ACKNOWLEDGMENTS

The authors express gratitude to the US Nuclear Regulatory Commission (NRC) for funding Polaris development. The author acknowledges Don Algama and Mourad Aissa of the NRC for their support of the project.

3.2.1 INTRODUCTION

Polaris was introduced in SCALE 6.2 to provide 2D lattice physics analysis capability for light water reactor (LWR) fuel designs. For multigroup cross section processing, Polaris uses the Embedded Self-Shielding Method (ESSM) [POLARISWK12]. Unlike SCALE multigroup sequences that use XSProc, ESSM does not require user-defined unit cell definitions. ESSM computes multigroup cross sections using Bondarenko interpolation methods. The background cross section used in the interpolation is determined by iterative 2D fixed source transport calculations. Both the ESSM fixed-source calculations and the keff calculation utilize a Method of Characteristics (MoC) transport solver developed in the Exnihilo computational package.

Polaris is integrated with ORIGEN for depletion calculations. Each pin-or each radial subregion of the pin-is depleted based on the local power distribution. An optional critical spectrum calculation is incorporated into the depletion calculation for computing depletion reaction rates and/or few-group homogenized cross sections. A complete description of the Polaris computational methods is provided in [POLARISJWM+21].

Polaris provides an easy-to-use input format to allow users to set up lattice models with minimal lines of input. All recognized Polaris commands are shown in Table 3.2.1. Note that many commands support short and

long forms. The allowed basic *Types* for input are described in Table 3.2.2. The special Polaris *TYPES* are shown in Table 3.2.3. The convention used in this manual is that basic types appear italicized and capitalized (*Type*), while special Polaris types appear in all caps (*TYPE*).

Table 3.2.1: Polaris commands.

card	*long command*	*short command(s)*
system	system	sys
geometry	geometry	geom
composition	composition	comp
property	property	prop
material	material	mat
burnup	bu or dbu	-
power	power	pow
options	option	opt
time	t or dt	-
state	state	-
branch block	branch	-
pin geometry component	pin	-
assembly pin map	pinmap	-
assembly channel	channel	-
assembly half gap	hgap	-
channel box	box	-
shield	shield	-
deplete	deplete	-
slab geometry component	slab	-
power basis materials	basis	-
assembly inserts	insert	-
assembly control elements	control	-
water cross geometry	cross	-
displacement maps	dxmap (dymap)	-
spatial meshing	mesh	-
detector tallies	detector	-
operating histories	history	-
restart cumulative burnup	bui (ti)	-

Table 3.2.2: Basic Types in Polaris input.

basic type	**description**	**examples**	**incorrect examples**
Word	starts with a character A-Z or a-z and includes characters, numbers, underscores	uox bor_water_500ppm FUEL	uox_enr5.1 316SS uox-3.1

continues on next page

Table 3.2.2 – continued from previous page

basic type	description	examples	incorrect examples
Int	integer	17 92235 2 565	31.4 uox
Bool	boolean/logical	yes false	TRUE No
Real	any number	565 10.257 1.5e-6	yes bor_water
String	a single or double quoted string	"INFMED" "Includes spaces" 'NONE'	Includes spaces
Value	any non-word		*Int*|*Bool*|*Real*|*String*

Table 3.2.3: Special Polaris Types.

Polaris type	description	variants
STYPE	system type	*PWR* | *BWR*
GTYPE	geometry type	*ASSM* | *REFL*
CTYPE	composition type	*NUM* | *WT* | *FORM* | *CONC* | *LW* | *UOX* | *ENRU* | *UN* | *USI*
PTYPE	property type	*SOLP* | *DOPANT*
ETYPE	control element type	*RODLET*|*BLADE*
OTYPE	option type	*KEFF* | *BOND* | *ESSM* | *CRITS PEC* | *FG* | *DEPL* | *RUN* | *PRINT* | *GAMMA* | *GEOM* | *DATA*

The Polaris input supports a very flexible input scheme that allows some elements to be suppressed for better readability. With *key=value* type input, when the standard order of keys is used, the keys may be suppressed. Consider the following input specification as an example.

```
geometry GNAME : ASSM npins=Int ppitch=Real [sym=<FULL>|SE]
```

The geometry card requires a geometry name (GNAME) in the first group, then a geometry type (*GTYPE*) which is *ASSM* here indicating an assembly geometry. The remaining arguments have keys: "npins" with an integer value, "ppitch" with a real value, and the optional "sym" with either *FULL* or *SE* values (optional arguments are always shown in square brackets: [sym=<*FULL*>|*SE*]. The default value is in angle brackets: <FULL>. The pipe "|" shows an *or* relation (i.e., *FULL* or *SE* is an acceptable value). With the flexible input processing, the following inputs are all valid and identical.

```
geometry FuelNode : ASSM npins=15 ppitch=1.43 sym=FULL
geometry FuelNode : ASSM 15 1.43 FULL
geometry FuelNode : ASSM sym=FULL ppitch=1.43 npins=15
geometry FuelNode : ASSM 15 1.43
geometry FuelNode ASSM 15 1.43
```

The group separator ":" is suppressed in the last variant. This is possible in any situation where (1) the group is implicitly terminated by running out of arguments or (2) the next type does not match the expected type in the current group. For example, consider the hgap card:

```
hgap [ d ] [: M ]
```

In this card, d and M are values (without keys) defined as *Real* and material name (MNAME), respectively. The following form would automatically bypass the *Real* value, which allows a default, and set the interassembly gap material name as COOL.2.

```
hgap COOL.2
```

3.2.2 SCALE 6.3 POLARIS INPUT UPDATES

For the release of SCALE 6.2.3, several new input cards were implemented into Polaris to model boiling water reactor (BWR) geometries and neutron/gamma detectors, which requires a gamma transport calculation. Moreover, improvements to existing input cards were implemented, along with the ability to specify time-dependent state properties and the ability to specify one or more depletion histories. This section describes the new and modified input cards that are included in the Polaris input format for SCALE 6.3, which are accessible as part of the release of SCALE 6.2.3.

To maximize backwards compatibility for input files developed with the original SCALE 6.2.0 release, the new and modified input cards are not available *by default* with SCALE 6.2.3. The new and modified input cards were activated if the input file begins with =polaris_6.3 rather than =polaris. The suffix "_6.3" is an indicator to the Polaris input processor to use the SCALE 6.3 input format. For the SCALE 6.3 release, the original input cards supported in the SCALE 6.2 input format will be available if the input file begins with =polaris_6.2, with the new SCALE 6.3 defaults being used for inputs with =polaris.

The new input cards to model BWR geometries include:

- **cross** – define the interior water cross geometry of SVEA assembly designs;
- **dxmap** (or **dymap**) – define displacement maps that indicate that translation of the pin center in the x- (or y-) direction;
- **control <BLADE>** – define the control blade geometry;
- **mesh** – define advanced spatial meshing options for different materials; and
- **option <GEOM>** – define geometry tolerances, advance meshing options, and plotting options.

The modified input cards to model BWR geometries include:

- **pin** – define circular and square-based geometry zones, as well as arbitrarily sized pins, e.g. size=1.5 water rod in some 9x9 BWR lattice designs; and
- **box** – define channel box geometry with arbitrary number of zones and cutout regions.

For neutron/gamma detector modeling, there is a new **detector** card and an addition to the existing **option <FG>** card to enable output to the few-group cross section output (T16) file.

To control the gamma calculation, an **option <GAMMA>** has been added.

The new input cards for time-dependent modeling include:

- **history** – define one or more operating histories in the input file; and

- **bui** (or **ti**) – define restart cumulative burnup (or time) values.

The modified input cards for time-dependent modeling include:

- **state** – define one or more time-independent or time-dependent state properties;
- **bu** (or **t**) – define cumulative burnup (or time) values; and
- **dbu** (or **dt**) – define incremental burnup (or time) values.

Example input files are included in the ${SCALE}/regression/input directory:

- polaris.6.3.atrium9x9.inp and polaris.6.3.atrium10x10.inp – prototypic ATRIUM models;
- polaris.6.3.blade1.inp and polaris.6.3.blade2.inp – control <BLADE> examples;
- polaris.6.3.ge7x7.inp through polaris.6.3.ge10x10.inp – prototypic GE models;
- polaris.6.3.svea100.inp and polaris.6.3.svea64.inp – prototypic SVEA models; and
- polarisHistory.inp: history example.

3.2.3 SETUP

The cards in this section generally appear at the beginning of an input file. Note that the manual is organized with each card starting a new page. This is especially convenient when printing a few cards across different sections.

3.2.3.1 title - case title lines

title Line_1 Line_2 ... Line_i ... Line_N

param	type	name	details	default
Line_i	*String*	line	used in output file headers	"DEFAULT TITLE"

Examples:

```
title "Westinghouse 15x15"

title "Westinghouse 15x15"
       "Condition: Hot Full Power"
       "Date: 10/18/2012"
```

Comments:

The **title** card gives a title to this Polaris case, which appears as a descriptive header on the output file. The additional lines may be used to document a subcase or to embed additional information in the output file in an orderly way (e.g., author, date, project identifier).

The **title** card is optional.

See also:

lib

3.2.3.2 library - nuclear data libraries

lib [mg=*String*]

param	type	name	details	default
mg	*String*	multigroup library	multigroup cross section library	"fine_n"

Examples:

```
% a name of library in the DATA directory
% use SCALE 252g ENDF/B-VII.1 library
lib "fine_therm"   % Alias for the V7.1 252g-neutron 47g-gamma library

% use SCALE 56g ENDF/B-VII.1 library
lib mg="broad_lwr"   % Alias for the V7.1 56g-neutron 19g-gamma library

% a name of a local library in the temporary working directory
% (useful in SAMPLER calculations)
lib "perturbed_xs_library"

% fully specified path
lib "C:\scale6.3\data\scale.rev05.xn252v7.1"
```

Comments:

The **lib** card specifies the multigroup library location. See SCALE's FileNameAliases.txt file in the installation directory for up-to-date library aliases for the fine and broad group libraries provided in SCALE's data directory. Only the 252-group and the 56-group cross section libraries are recommended to be used in Polaris. Full specification of the file path is acceptable, as in the final example shown above.

The **lib** card is optional.

See also:

title

3.2.4 GEOMETRY

The highest level structures in the model are named and defined with a **geometry** card. The general outline for a geometry definition is shown below. Two types of geometry are currently supported, *ASSM* for pressurized water reactor (PWR) or boiling water reactor (BWR) assemblies with fuel elements in a square-pitch, and *REFL* for an assembly-adjacent reflector.

geom GNAME : GTYPE *arguments*

param	type	name	details	default
GNAME	*Word*	geometry name		
GTYPE	-	geometry type		
	ASSM	assembly	see **pin** & **pinmap**	
	REFL	reflector	see **slab**	
arguments	-	remaining arguments	depends on *GTYPE*	

The control element geometry is also enumerated with types, as shown below. To model PWR-type rod cluster control assemblies (RCCAs), the *RODLET* element type is used in conjunction with **pin** definitions. In future releases of Polaris, other control element types, such as BWR-type control blades will be supported..

control INAME : ETYPE *arguments*

param	type	name	details	default
INAME	*Word*	control element name		
ETYPE	-	control element type		
	RODLET	PWR-type RCCA	requires PINIDs	
arguments	-	remaining arguments	depends on *ETYPE*	

3.2.4.1 geometry<ASSM> – assembly

geom GNAME : *ASSM*
npins=*Int*
ppitch=*Real*
[sym=<*FULL*>|*SE*]

param	type	name	details	default
GNAME	*Word*	assembly name		
GTYPE	*ASSM*			
npins	*Int*	number of pins	on each side of the assembly	
ppitch	*Real*	pin pitch	units: cm	
sym	FULL|SE	symmetry	assembly symmetry, FULL: no symmetry, SE: south-east quarter	FULL

Examples:

```
% simple pincell
geom MyPin : ASSM 1 1.5

% 17x17 Westinghouse with 1.26 cm pin pitch in quarter symmetry
geom FuelNode : ASSM 17 1.26 sym=SE
```

Comments:

Examples:

% simple pincell

geom MyPin : ASSM 1 1.5

% 17x17 Westinghouse with 1.26 cm pin pitch in quarter symmetry

geom FuelNode : ASSM 17 1.26 sym=SE

Comments:

The assembly geometry describes the basic elements of an assembly. The **pin** and **pinmap** cards are required to finalize the assembly geometry. The **hgap** card specifies the interassembly half gap, and the **channel** specifies the channel material for the assembly.

See also:

pinmap, **pin, hgap, box, channel, control**, **insert**

3.2.4.2 geometry<REFL> - reflector

geom GNAME : *REFL*
thick=*REAL*

param	type	name	details	default
GNAME	*Word*	reflector name		
GTYPE	*REFL*			
thick	*Real*	thickness	units: cm	

Examples:

```
% defines a 20 cm reflector
geom ReflectorNode : REFL 20.0
```

Comments:

The reflector geometry describes the basic elements of a simple slab-type reflector. The **slab** card can be used to define geometric dimensions and mesh for the reflector geometry.

See also:

slab

3.2.4.3 channel - coolant channel

channel [M_{chan} =MCLASS]

param	type	name	details	default
M_{chan}	MCLASS	material class	initializes materials in outermost **pin** zone	*
*By default, M_{chan} will be set to COOL by "system PWR" and "system BWR." Otherwise, M_{chan} is required.				

Examples:

```
% define the channel material class to be COOL
channel COOL
```

Comments:

The **channel** card is used to set the default channel material class for the outermost region of each pin, typically containing reactor coolant. See the **material** card for a description of material classes.

See also:

pin, material, geometry<ASSM>

3.2.4.4 hgap - half distance between assemblies

hgap [d_E d_N d_W d_S]
[: M_E M_N M_W M_S]
[: nf_E nf_N nf_W nf_S]
[: nd_E nd_N nd_W nd_S]

param	type	name	details	default
d_i	*Real*	list of widths *with i*: *E:* east *N:* north *W:* west *S*: south	accepts 1, 2, or 4 values E: all hgaps are same E+N: d_E =d:sub: *S* and d_N =d_W units: cm	0.0
M_i	MNAME	list of material names	requires same # as d_i	*
meshing options				
nf_i	*Int*	list of number of faces per pin	requires same # as d_i	2
nd_i	*Int*	list of number of divisions	requires same # as d_i	1

*By default, **hgap** material will be set to COOL.1 by "system PWR." For "system BWR," the east and south **hgap** materials will be set to MOD.2, and the west and north **hgap** materials will be set to MOD.1. Otherwise **hgap** material is required.

Examples:

```
% defines a 17x17 Westinghouse assembly with 1.26 cm pin pitch
% with 0.04 cm half-gap filled with material COOL.1
geom w17x17 : ASSM 17 1.26 sym=SE
hgap 0.04 COOL.1

% defines a GE 7x7 assembly with 1.88 cm pin pitch
```

(continues on next page)

(continued from previous page)

```
%   0.48 cm narrow gap on east and south edge
%   0.95 cm wide gap on north and west edge
%   narrow gap mesh is 3
%   wide gap mesh is 4
%   faces per pin is 2 for both narrow and wide gap
geom ge7x7 : ASSM 7 1.88
hgap 0.48 0.95 : MOD.1 MOD.1 : 2 2 : 3 4
```

Comments:

The **hgap** specifies the outermost geometry region in an assembly. If a channel **box** exists, then **hgap** specifies the material and mesh from outer channel box edge to the problem boundary. Otherwise**, hgap** specifies the material and mesh from the edge of the fuel array to the problem boundary. In both cases, **hgap** represents the half-distance between adjacent assemblies for single assembly calculations. Fig. 3.2.1 shows some of the hgap meshing options. Referring to the south edge of the assembly, the number of faces per pin refers to the extra cells introduced by "splitting" the pin cell boundary, and the number of divisions refers to extra horizontal lines dividing half gap into smaller width cells.

See also:

pinmap, **control**, **insert**, **geometry<ASSM>, channel, box**

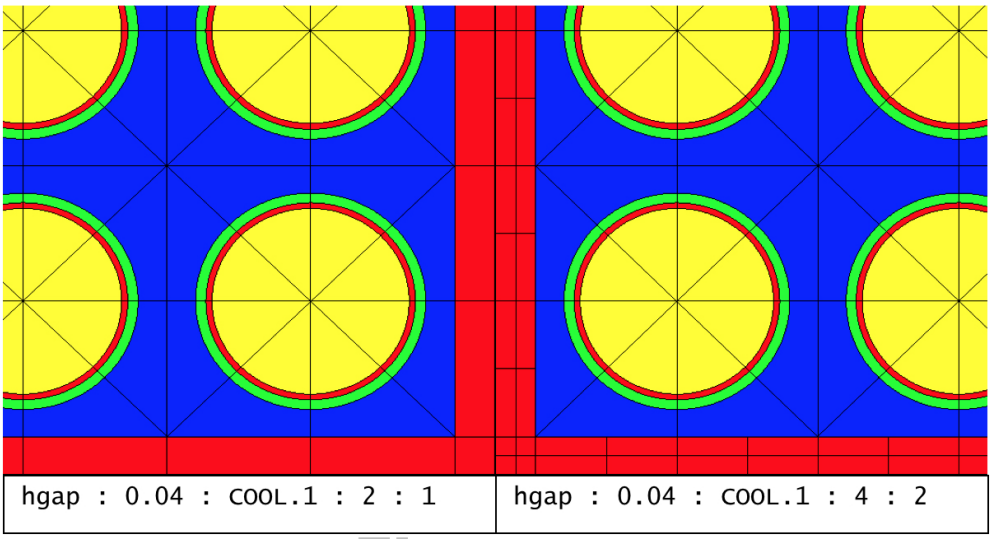


Fig. 3.2.1: Interassembly half gap meshing variants.

3.2.4.5 box - channel box geometry

box thick=*Real* [rad=*Real*] [hspan=*Real*] [Mbox=MNAME] [cothick=*Real*] [cobtm=*Real*] [cotop=*Real*]

[: t_2 t_3 … t_i … t_N
[: a_2 a_3 … a_i … a_N
[: b_2 b_3 … b_i … b_N
[: M_2 M_3 … M_i … M_N

[: r_2 r_3 ... r_i ... r_{N+1}]]]]]

param	type	name	details	default
thick	*Real*	nominal thickness (cm)	must be > 0.0	
rad	*Real*	inner corner radius (cm)	must be ≥ 0.0	0.0
hspan	*Real*	half inner span (cm)	———-See comments——-	
Mbox	*MNAME*	box material		*
cothick	*Real*	interior cutout region thickness (cm)	thick > cothick ≥ 0.0	0.0
cobtm	*Real*	distance from box centerline to bottom of interior cutout region (cm)	hspan-rad > cobtm ≥ 0.0	0.0
cotop	*Real*	distance from box centerline to top of interior cutout region (cm)	cobtm ≥ cotop ≥ 0.0	cobtm
options for additional box zones				
t_i	*Real*	zone thickness (cm)	must be ≥ 0	
a_i	*Real*	distance from box centerline to bottom of zone cutout region (cm)	——See comments——	
b_i	*Real*	distance from box centerline to top of zone cutout region (cm)	——See comments——	
M_i	*MNAME*	zone material	——See comments——	
ri	*Real*	zone inner corner radius (cm)	——See comments——	

*By default, **box** material will be set to CAN.1 by "system BWR." Otherwise **box** material is required.

Examples:

```
% simple
box 0.2

% rounded corner, rad 0.9
box 0.2 0.9
```

(continues on next page)

(continued from previous page)

```
% rounded corner and user-defined inner span
box 0.2 0.9 6.7

% two zones
box 0.2 0.9 6.7
  : 0.2
  : 4.0
  : 4.3
```

Comments:

The **box** specifies the channel box geometry that surrounds the **pinmap**. The three primary dimensions of the channel box are the thickness (thick), the inner corner radius (rad), and the half inner span (hspan). Several additional dimensions for both **box** and **cross** are defined with respect to the channel box center. The channel box center is not to be confused with the lattice center: the former is the centroid of the inner channel box square boundary and the latter will depend on the wide and narrow gap dimensions provided on the **hgap** card. By default, the half inner span is equal to the half pin pitch multiplied by the number of pins on each side of the assembly (see npins and ppitch on the **geometry<ASSM>** card). If a **cross** card is applied, the default half inner span is increased by the half width of the interior cross buffer region (see hwidth on the **cross** card).

Additional channel box zones can be specified on the **box** card. The additional zones are useful for defining thick corner regions of the channel box. Each additional zone must have a user-defined thickness (t_i, i = 2 to N). Note that the starting index begins at "2" rather than "1" because the zone 1 thickness has already been defined by the "thick" input field.

"Cutout regions" may be defined in which a portion of the channel box zone is replaced by the corresponding **hgap** material along the horizontal and vertical centerlines of the channel box. The cutout region is defined by the distance from the channel box centerline to the bottom additional channel box zone (a_i) and the top of the channel box zone (b_i). The values of a_i and b_i determine the size of trapezoidal cutout region centered along each face of the channel box. The b_i value must be greater than or equal to the a_i value. The a_i value must be greater than or equal to the previous zone's b_i value, i.e., b_{i-1}. By default, a_2 and b_2 are zero. If only M cutout regions are specified for N additional zones, i.e., M < N, both a_i and b_i is set to b_M for i = M+1 to N.

Additional zones can also have a different inner corner radius ($r_2 \ldots r_N$). The outer corner radius of the last zone may also be specified (r_{N+1}). By default, r_2 is zero if rad is zero. If rad is greater than zero, the default value of r_2 is rad+thick. Similar rules apply for determining the default corner radii for additional zones if they are omitted in the input specification.

Additional zones can also have a different material (M_i). By default, M_2 is M_{box}. If additional materials are omitted in the input, the default value of M_i is M_{i-1} for i = 3 to N.

The spatial mesh along each face of the channel box will be determined by the nf values specified on the hgap card.

The four examples listed above are displayed in Fig. 3.2.2. For additional examples, see the polaris.6.3 regression input files described at the beginning of Sect. 3.2.2.

See also:

geometry<ASSM>, hgap, cross

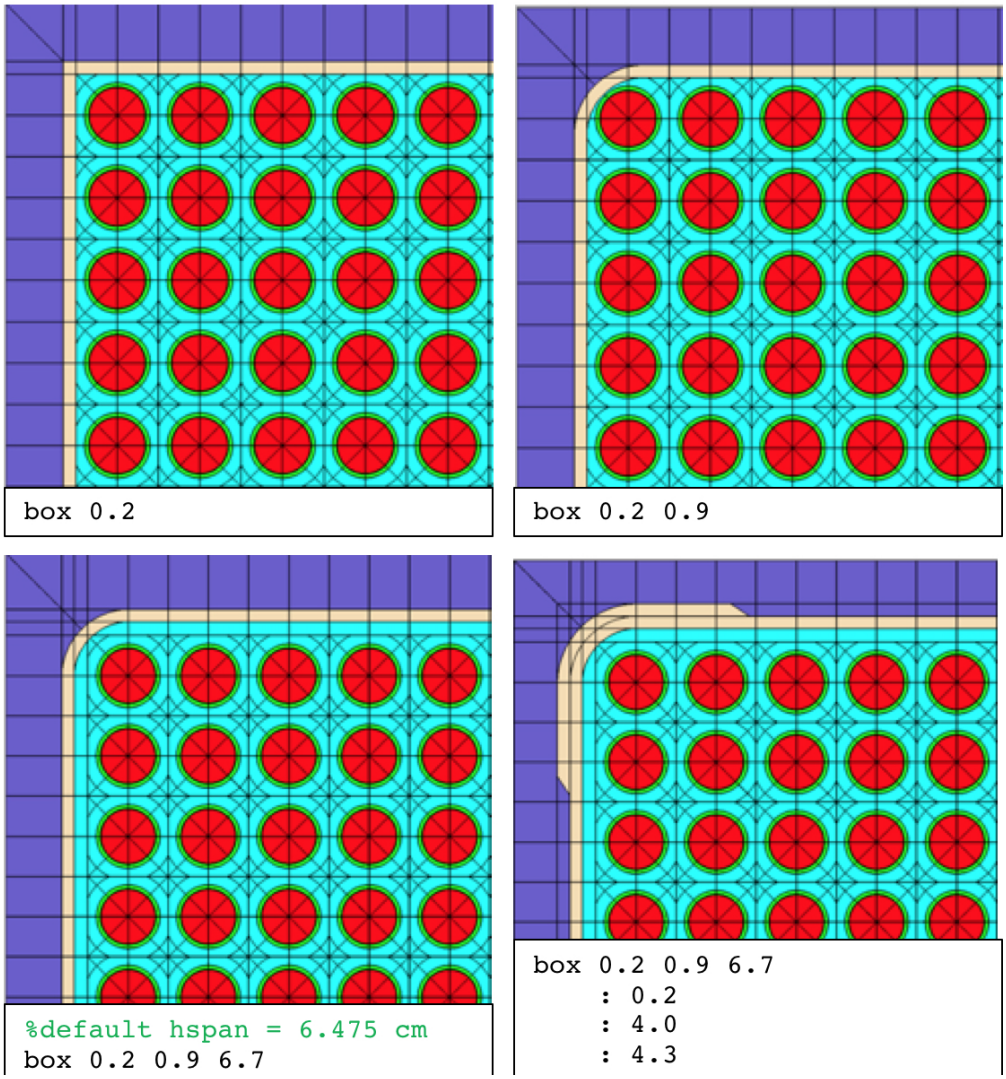


Fig. 3.2.2: Box card examples.

3.2.4.6 pin - pincell comprised of nested geometry zones of variable shape

pin PINID [size=*Real*]

: r_1 r_2 ... r_i ... r_N

: M_1 M_2 ... M_i ... M_N [M_{out}]

[: S_1 S_2 ... S_i ... S_N]

param	type	name	details	default
PINID	*Word|Int*	pin identifier		
size	*Real*	pin size multiplication factor	must be ≥ 1.0	1.0

continues on next page

Table 3.2.13 – continued from previous page

r_i	*Real*	zone interior radius (cm)	r_1 must be > 0. Additional zones must be > r_{i-1}	
M_i	*MNAME*	zone material	.1 added if given MCLASS, e.g., FUEL-FUEL.1	
M_{out}	*MNAME*	outermost zone material	.1 added if given MCLASS	*
S_i		CIR or SQR or SQR(X)	circular zone or square zone with optional corner radius, X ≥ 0.0	CIR X=0.0

*If not specified, the material class MCLASS is taken from the **channel** card (M_{chan}) and set to the first member of that class, "M_{chan}.1." For example if M_{chan}="COOL," then M_{out}= "COOL.1."

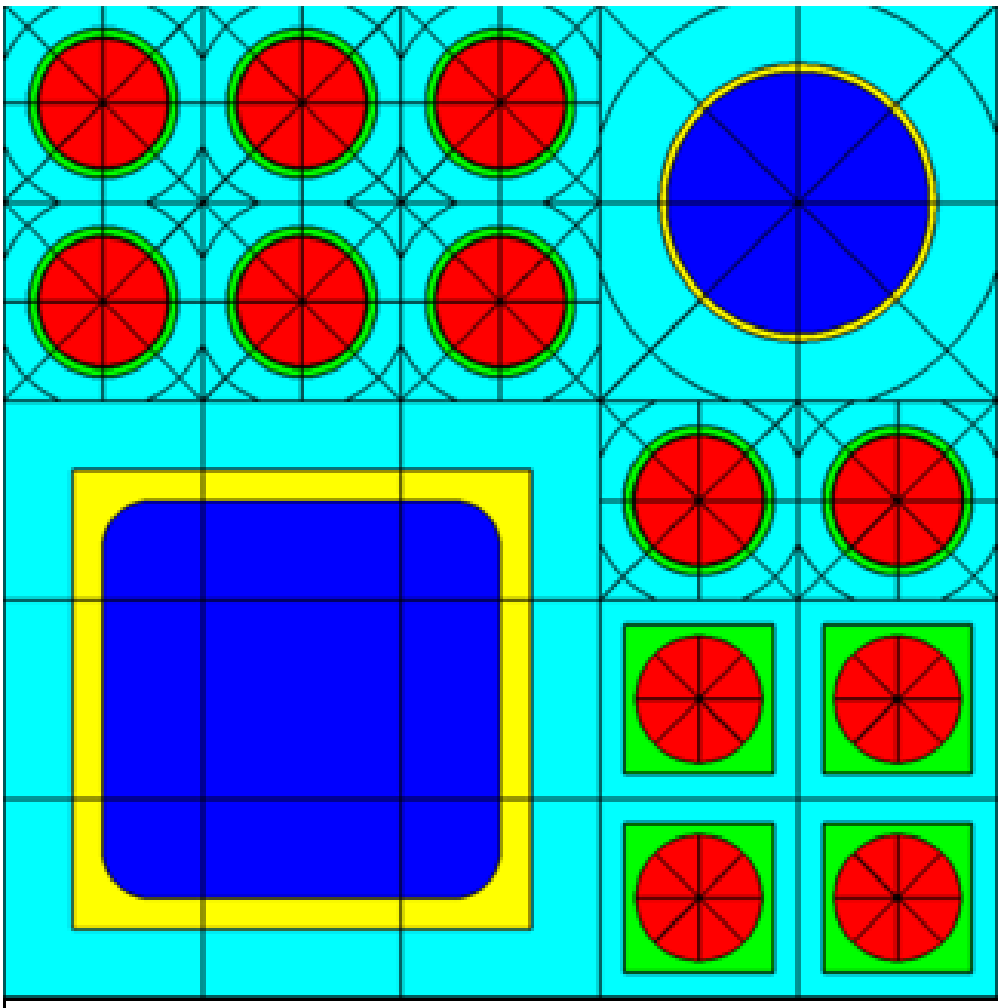
Examples:

```
%standard fuel pin
pin 1 : 0.4096 0.418 0.475 : FUEL.1 GAP.1 CLAD.1

%2x2 water rod
pin W 2.0 : 1.6   1.7 : MOD.1 TUBE.1

%3x3 square water box (ATRIUM)
pin W 3.0 : 1.68 1.75 : MOD.1 TUBE.1 : SQR SQR

%noninteger size water rod (GE9x9)
pin W 1.76 : 1.16 1.259 : MOD.1 TUBE.1 COOL.2
```

Comments:

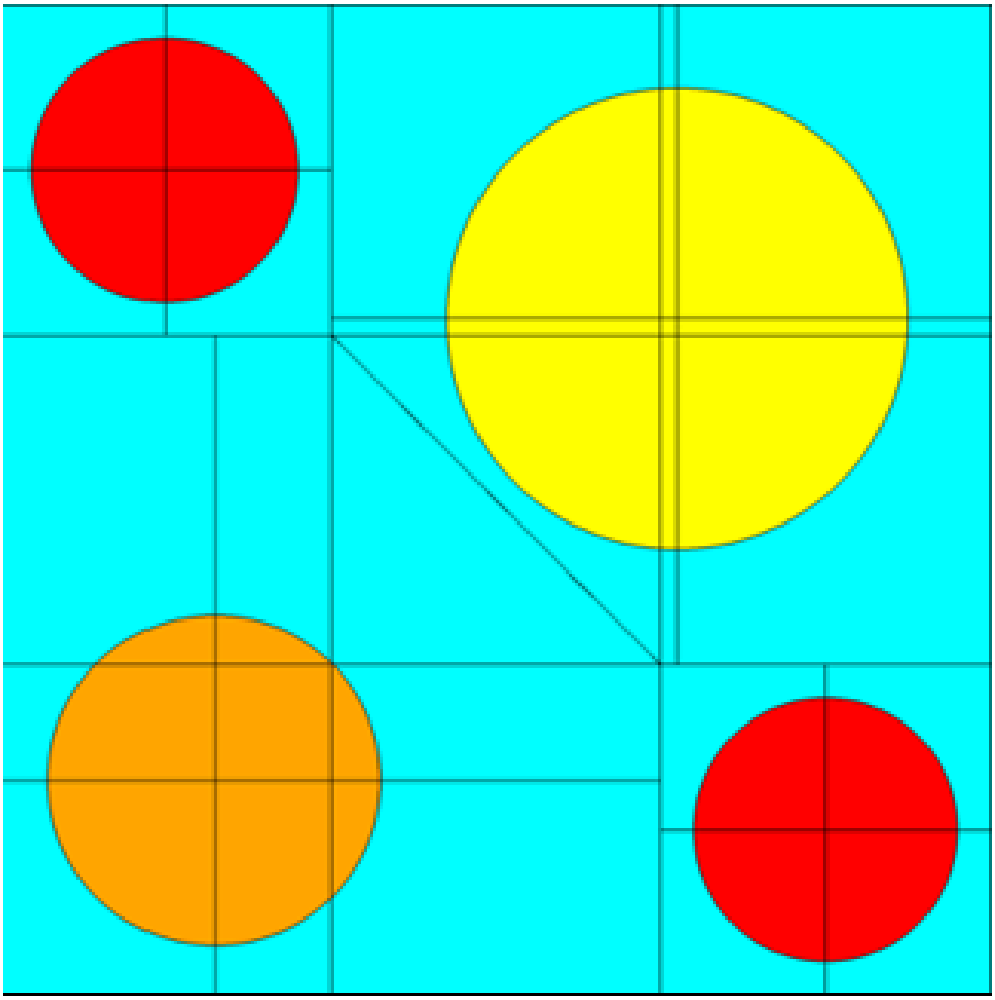
The **pin** card is one of the basic building blocks of the assembly model. **pin** and **slab** are the only geometry components which allows an integer (*Int*) identifier as well as a *Word*-all other geometric identifiers use *Word*. Note that the materials are required, except for the last M_{out}, which can be used to overwrite the material given by a **channel** for the outermost region in the pincell. The **pin** geometry is constructed from the inside out, using either circle zones (defined by the radius) or square zones (defined by the half-width, and optional corner radius). Different examples of pin geometries are displayed in Fig. 3.2.3. All meshing options for the **pin** are provided through the **mesh** card.

If the pin size is an integer value, the pin consumes a size × size subarray in the **pinmap** (e.g. 1 × 1, 2 ×times` 3, etc). If the pin size is noninteger, the pin consumes a *ceil*(size) × *ceil*(size) subarray in the **pinmap**. *ceil(x)* represents the ceiling function to round the value of x to the nearest integer greater than or equal to x. For size equal to 1.3, each instance of the pin will consume a 2 × 2 subarray in the **pinmap**. Each instance of a noninteger-sized pin must share a location with another instance of a noninteger-sized pin, but not necessarily the same pin. The shared location must be set to "_" in the **pinmap**. The identification of the shared location is necessary to determine the center of each pin. The pin center is at a distance of size*half pitch*sqrt(2) from the opposite corner of the shared location, along the diagonal of the pin boundary. An example of an integer-sized pins is displayed in Fig. 3.2.3. An example of noninteger-sized pins is displayed in Fig. 3.2.4.



```
geom manual_ex : ASSM 5 1.26
pin A : 0.4096 0.418 0.475 : FUEL.1 GAP.1 CLAD.1
pin B 2 : 0.83  0.88 : MOD.1    TUBE.1
                     : CIR      CIR
pin C 3 : 1.26  1.46  : MOD.1    TUBE.1
                      : SQR(0.3) SQR
pin D : 0.4    0.475 : FUEL.1 CLAD.1
                     : CIR    SQR

pinmap A A A B B
       A A A B B
       C C C A A
       C C C D D
       C C C D D
```

Fig. 3.2.3: Pin examples with different shape geometries



```
geom manual_ex : ASSM 3 1.0
pin A 1.0 : 0.4 : FUEL.1
pin B 1.3 : 0.5 : FUEL.2
pin C 1.9 : 0.7 : FUEL.3
pinmap
A C C
B _ C
B B A
```

Fig. 3.2.4: Pin examples with noninteger pin size.

For additional examples, see the polaris.6.3 regression input files described at the beginning of Sect. 3.2.2

See also:

slab, pinmap, channel, mesh

3.2.4.7 pinmap - pin layout

pinmap PINID_1
PINID_2 ...
PINID_i ... PINID_N

param	type	name	details	default
PINID_i	*Word|Int*	list of pin identifiers	supports full, quarter, or octant symmetry quarter: assumes south-east (SE) octant: assumes south-by-so utheast (SSE)	

Examples:

```
%Westinghouse 17x17 pinmap in octant symmetry
pinmap
  2
  1 1
  1 1 1
  3 1 1 3
  1 1 1 1 1
  1 1 1 1 1 3
  3 1 1 3 1 1 1
  1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1

%Westinghouse 17x17 pinmap in quarter symmetry
pinmap
  2 1 1 3 1 1 3 1 1
  1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1
  3 1 1 3 1 1 3 1 1
  1 1 1 1 1 1 1 1 1
  1 1 1 1 1 3 1 1 1
  3 1 1 3 1 1 1 1 1
  1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1

%Westinghouse 17x17 pinmap in full
pinmap
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 3 1 1 3 1 1 3 1 1 1 1 1
  1 1 1 3 1 1 1 1 1 1 1 1 1 3 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 3 1 1 3 1 1 3 1 1 3 1 1 3 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 3 1 1 3 1 1 2 1 1 3 1 1 3 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 3 1 1 3 1 1 3 1 1 3 1 1 3 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 3 1 1 1 1 1 1 1 1 1 3 1 1 1
  1 1 1 1 1 3 1 1 3 1 1 3 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

(continues on next page)

(continued from previous page)

```
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

%large central 2x2 water rod in 6x6 assembly
%pinmap must show adjacent Ws
pin W size=2 : 0.8
             : COOL
pinmap
F F F F F F
F F F F F F
F F W W F F
F F W W F F
F F F F F F
F F F F F F
```

Comments:

The **pinmap** card defines the layout of pin cells in the assembly. The symmetry is determined by the number of pin identifiers given on the card and must not be more general than the symmetry option given on the assembly **geometry** card (i.e., do not define a full pin map for a *sym*=SE assembly model). If the **pin** has a large size specifier, *size>1*, then the pinmap must reflect that with those pins occurring in blocks of size × size.

See also:

pin, control, insert

3.2.4.8 control<RODLET> - RCCA-type layout

control INAME: *RODLET*

PINID$_1$

PINID$_2$ ...

PINID$_i$ ... PINID$_N$

param	type	name	details	default
INAME	*Word*	insert name		
ETYPE	*RODLET*			
PINID$_i$	*Word|Int*	list of pin identifiers	same format as **pinmap** "_" indicate empty locations	

Examples:

```
% B4C control rods
mat GAS.1  : FILLGAS
mat CLAD.1 : ZIRC4
mat MOD.1  : LW
mat TUBE.1 : SS304
mat CNTL.1 : B4C

pin B : 0.214  0.231 0.241 0.427 0.437 0.484 0.561 0.602
      : GAS    TUBE  GAS   CNTL  GAS   TUBE  MOD   CLAD

control  BankD  : RODLET
```

(continues on next page)

(continued from previous page)

```
_
_ _
_ _ _
B _ _ B
_ _ _ _ _
_ _ _ _ _ B
B _ _ B _ _ _
_ _ _ _ _ _ _ _
_ _ _ _ _ _ _ _ _
```

Comments:

Note that control elements and inserts share the INAME identifiers, so an insert and a control element cannot have the same name. Different control rod banks may be included in a single input file using more than one **control** card with unique INAMEs. The main difference between the inserts defined by **control** element and **insert** cards is that by default, *control element materials are not depleted,* whereas *insert materials are depleted*.

The outer dimensions of the tube must be included in the **pin** card that is inserted.

See also:

pinmap, **control**, **insert, state**

3.2.4.9 control<BLADE> - BWR control blade

control INAME : BLADE hwgthck=*Real* sththck=*Real* cslnth=*Real*
[sthmat=*MNAME*] [csmat=*MNAME*] [hcsthck=*Real*] [wgcrv=*Real*]
: ID_1 ID_2 … ID_N
: L_1 L_2 … L_N
[: N_1 N_2 … N_N]

param	type	name	details	default
hwgthck	*Real*	half blade wing thickness (cm)	must be >0	
sththck	*Real*	sheath thickness (cm)	must be >=0	
cslnth	*Real*	central support length (cm)	must be >=hwgthck	
sthmat	*MNAME*	sheath material		STRUCT.1*
csmat	*MNAME*	central support material		STRUCT.1*
hcsthck	*Real*	half central support thickness (cm)	must be >0	hwgthck
wgcrv	*Real*	wing tip radius (cm)	must be >=0	0
ID_i `	*Word|Int*	pin or slab identifier	——- See comments ———	

continues on next page

Table 3.2.16 – continued from previous page

L_i	*Real*	length of section i	——- See comments ———-	
N_i	*Real*	# of pins or slabs in section i	——- See comments ———-	
*Default values for shtmat and csmat are set by "system BWR." If "system BWR" is omitted, the material definitions are required.				

Comments:

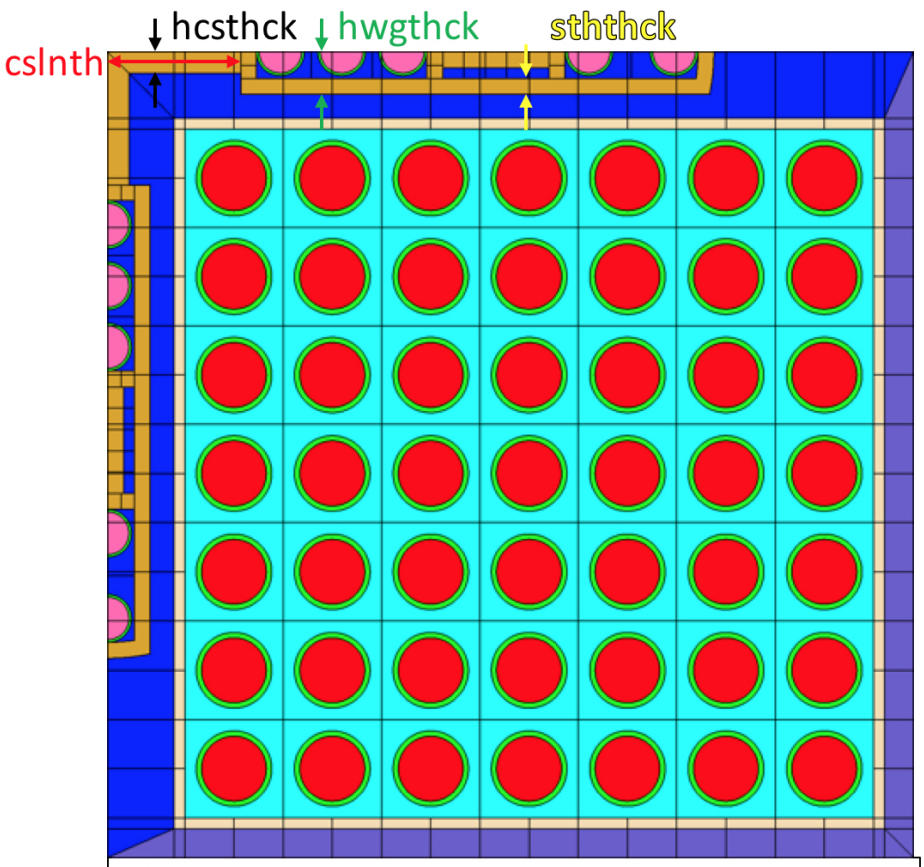
The **blade** card defines a control blade geometry. The control blade identifier (INAME) can be used to insert the control blade using **state** or **add** statements to define histories or branches respectively. INAME=yes inserts the control blade into the northwest corner of the lattice.

The control blade geometry is described with reference to the blade wing on the northern edge of the lattice in Fig. 3.2.5. The blade wing on the west edge is a reflection of the northern edge wing along the diagonal symmetry line that extends from the northwest corner to the southeast corner of the lattice.

The two primary regions of the blade are the central support and the active blade wing. The central support has a length (cslnth), half width (hcsthck), and material (csmat). The central support half width is the vertical distance between the north face of the lattice and the south boundary of the central support. The central support length is the horizontal distance from west face of the lattice to the east boundary of the central support. See Fig. 3.2.5 for details.

The active portion of the blade wing begins at the east boundary of the central support. The active portion has a half width (hwgthck), sheath thickness (shthck), sheath material (sthmat), and wing tip radius (wgcrv). The half width is the vertical distance between the north face of the lattice and the southern boundary of the active blade wing, including the sheath. The wing tip radius can be any nonnegative number. If the radius is zero, the wing tip is a straight edge.

The active portion of the blade wing is subdivided into sections. Each section has a length (L_i), and identifier associated with a pin or slab (ID_i), and the number of pins or slabs for each section (N_i). The list of section lengths and section identifiers is required and must have consistent list lengths. The final list for number of pin/slabs per section is optional. If omitted, the default number of pin or slabs per section is one.




```
pin A : 0.4 0.45 : CNTL.1 CLAD.1 MOD.1
slab S : 0.5 : STRUCT.1 : 2 : 1
slab Z : 0.3 0.2 : STRUCT.1 MOD.1
control blade1 : BLADE 0.8 0.3 2.5
                   hcsthck=0.4 wgcrv=4.0
                 : S   A   S   Z   S   A
                 : 0.3 3.2 0.3 2.0 0.3 2.5
                 : 1   3   1   5   1   2

state blade1 : in=yes
```

Fig. 3.2.5: Control blade example.

If there is only one **pin** in a pin section, the pin is placed in the section center. If there are multiple pins, the first and last pin are positioned flush against the west and east section boundary respectively, and the interior pins are uniformly spaced between the two edge pins. Slab sections are built from the blade centerline in the vertical direction towards the interior sheath boundary. Each slab zone has width equal to the section length. Each slab zone can be subdivided in the vertical direction by the zone nx parameter on the **slab** card. The slab zone can be subdivided in the horizontal direction by the product of the zone ny parameter and the number of slabs in the blade wing section.

The blade is further subdivided by the **mesh** nf/nd settings for **hgap** material in the north and west bypass region, typically MOD.1 for models that include **system BWR.**

See also: pin, slab, mesh, system BWR

3.2.4.10 insert - insert layout

insert INAME:

PINID$_1$

PINID$_2$ ...

PINID$_i$ ... PINID$_N$

param	type	name	details	default
INAME	*Word*	insert name		
PINID$_i$	*Word|Int*	list of pin identifiers	same format as **pinmap** "_" indicate empty locations	

Examples:

```
%pyrex inserts
pin P : 0.214  0.231 0.241 0.427 0.437 0.484 0.561 0.602
      : GAS    TUBE  GAS   BP.3  GAS   TUBE  COOL  CLAD

insert PyrexInserts :
  _
  _ _
  _ _ _
  P _ _ P
  _ _ _ _ _
  _ _ _ _ _ P
  P _ _ P _ _ _
  _ _ _ _ _ _ _ _
  _ _ _ _ _ _ _ _ _
```

Comments:

The **insert** card defines a set of pins to be used to model inserts such as WABA. When the **insert** is "in," the insert pins replace *overlapping regions* of the pins defined on the assembly **pinmap**. An underscore (_) is used to indicate locations without inserts. See the notes on the **control<RODLET>** card for additional guidelines.

See also:

pinmap, **control**, **insert**

3.2.4.11 slab - slab geometry

slab [SLABID]

: t_1 t_2 ... t_i ... t_N

: M_1 M_2 ... M_i ... M_N

[: nx_1 nx_2 ... nx_i ... nx_N]

[: ny_1 ny_2 ... ny_i ... ny_N]

param	type	name	details	default
SLABID	*Word*	slab geometry identifier		reflector GNAME
t_i	*Real*	list of slab thicknesses	units: cm	
M_i	MNAME	list of material names		
meshing options				
nx_i	*Int*	list of number of x-divisions		1
ny_i	*Int*	list of number of y-divisions		1

Examples:

```
% a reflector definition
% 2.22 cm of baffle
% 15 cm of moderator
geom ReflectorNode : REFL 17.22
  slab 2.22 15 : BAFFLE.1  MOD.1
```

Comments:

The **slab** card may be used to define three things: (1) the materials and thicknesses of a reflector initiated on a **geometry** card, (2) slabs in a control blade, and (3) spacer grids. If the first argument identifier is *not present,* then the first purpose of describing the various material thicknesses in a reflector is assumed. The meshing options allow each material slab to be spatially refined in x and y, increasing the number of cells in the transport problem. The meshing option for the number of x divisions creates the equivalent of additional "sub-slabs" in each user-defined slab thickness. The y-divisions create additional cells vertically. The default of one y-division corresponds to the entire assembly.

See also:

geometry<REFL>

3.2.4.12 cross - cross geometry

cross hwidth=*Real* lthick=*Real*

[Mcross=*MNAME*] [row=*Int*] [Min=*MNAME*] [ld=*Int*] [Mout=*MNAME*]

[: x_1 x_2 ... x_N]

[: y_1 y_2 ... y_N]

[: yin_1 yin_2 ... yin_N]

[: nx_1 nx_2 ... nx_{N-1}]

[: ny_1 ny_2 ... ny_{N-1}]

param	*type*	*Name*	*details*	*default*
hwidth	*Real*	half width of interior cross region (cm)	must be >=0	
lthick	*Real*	cross liner thickness (cm)	must be >=0	
Mcross	*MNAME*	cross liner material		CAN.1*
row	*Int*	row to insert cross	must be >1 and < npins	npins/2
Min	*MNAME*	cross interior material		MOD.2*
ld	*Int*	number of liner divisions		1
Mout	*MNAME*	cross outer fill material		COOL.1*
x_i	*Real*	x coordinate of vertex i	------- See comments ----------	
y_i	*Real*	y coordinate of vertex i	------- See comments ----------	
yin_i	*Real*	y “inner” coordinate of vertex i	------- See comments ----------	
nx_i	*Int*	# of x subdivisions b/w vertices [i,i+1]	must be >0	1
ny_i	*Int*	# of y subdivisions b/w vertices [i,i+1]	must be >0	1

*Default values for Mcross, Min, and Mout set by “system BWR.” If “system BWR” is omitted, the material definitions are required.

Comments:

The cross card performs two tasks. First, it subdivides the pinmap into four subarrays, optionally adding a horizontal and vertical gap between the subarrays. The row parameter is uses to subdivide the pinmap. If the pinmap is 10×10 and row=5, each of the four subarrays is 5×5. If the pinmap is 10×10 and row=4, the northwest subarray is 4 ×times`6, the southwest subarray is 6×4, and the southeast subarray is 6×6. The hwidth parameter controls the half-spacing of the horizontal and vertical gap in between the subarrays. The hwidth parameter must be ≥ 0.0 and if hwidth is > 0.0, the gap is filled with material M_{out} (default is COOL.2 with system BWR).

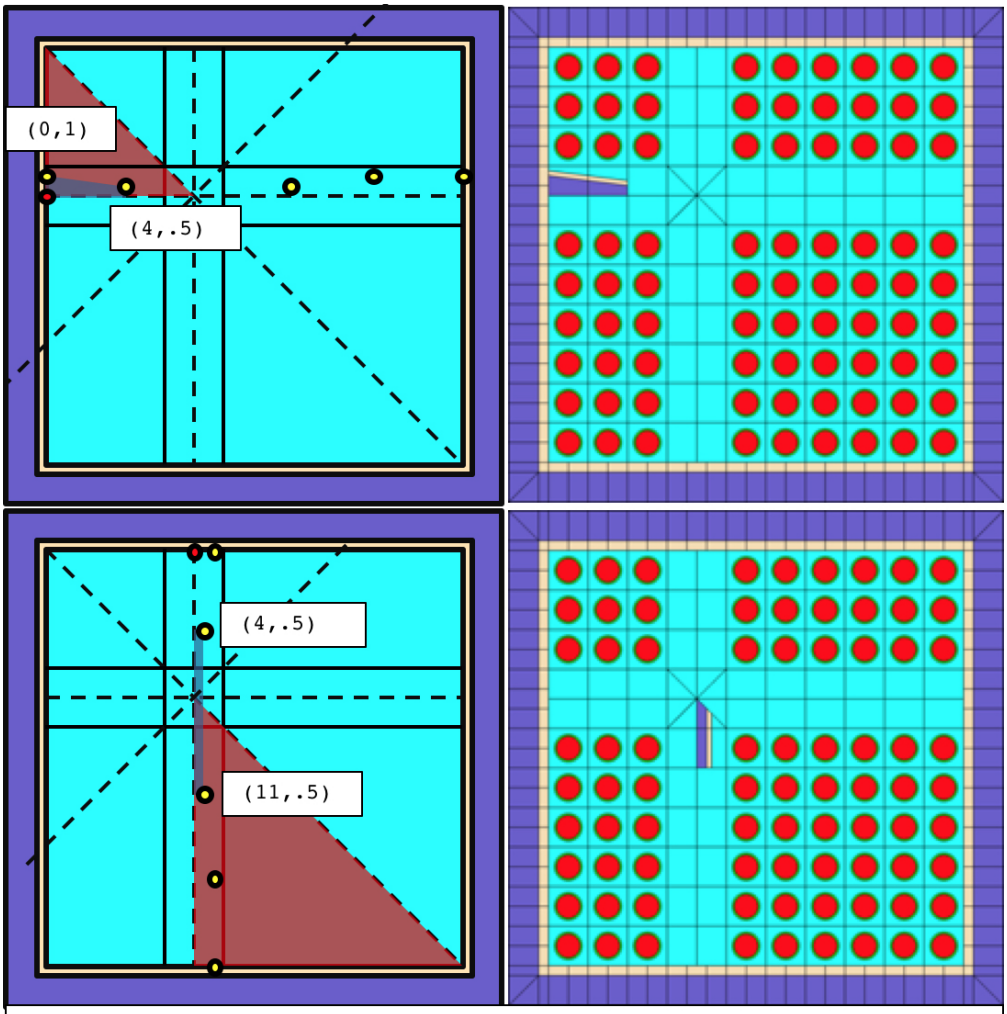
The second task is the insertion of the cross structure into the lattice geometry. The process is described with reference to the example in Fig. Fig. 3.2.6. In the example, the **pinmap** is 9×9 and row=3, hwidth=1.5, and hspan=10.5. The top left plot contains the four following lines:

1. the line in the center of the vertical cross gap,
2. the line in the center of the horizontal cross gap,
3. the diagonal line from the northwest (NW) channel box corner to the southeast (SE) corner, and

4. the diagonal line, perpendicular to line 3, passing through the intersection of line 1 and line 2.

These four lines intersect and form 8 separate regions, i.e., octants, within the channel box interior. The intersection point, i.e., cross center, is not necessarily equal to the box center as shown in this example. In the top left plot, the red triangle represents the WNW octant. In the bottom left plot, the red triangle represents the SSE octant.

```
% Centered cruciform flow channel
cross   0.625  0.08  5
: -6.86 -6.53 -5.20 -4.81 -3.77 -3.39 -2.05  0.0  2.05  3.39  3.77  4.81  5.20  6.53  6.86
: 0.625  0.0   0.0   0.24  0.24  0.0   0.0  2.05  0.0   0.0   0.24  0.24  0.0   0.0   0.625
```




```
geom manual_ex : ASSM 9 2.0
hgap 1.5
box 0.5 0.0 10.5
cross 1.5 0.25 row=3 : 0 4   11 13 21
                     : 1 .5 .5 1  1
```

Fig. 3.2.6: Construction of the BWR cross geometry (full example shown later).

The cross structure is defined be a series of vertices (x_i, y_i). Shown as yellow points in the top left plot, the **cross** vertices are defined based on an origin displayed as the red point, which is the intersection of the inner

west edge of the channel box and the horizontal line in that passes through the cross center.

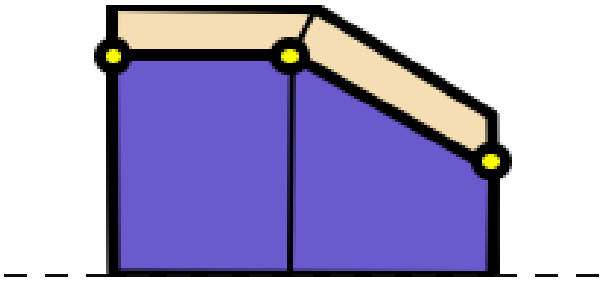
The top plots demonstrate how Polaris inserts a section of the cross into the WNW octant. In the top left plot, the blue polygon is constructed based on the first two vertices defined on the **cross** card: (0.0,1.0) and (4.0,0.5). The intersection of the blue polygon and red polygon is inserted into the lattice and filled with cross interior material (M_{in}). The liner is then inserted **above** the blue polygon, padded by the liner thickness (lthick), and clipped by WNW red polygon if needed.

Similarly, the bottom plots demonstrate insertion into the SSE octant. For SSE insertions, the origin and the **cross** vertices are rotated 90 degrees about the cross center. The blue polygon is constructed from the second and third vertices on the **cross** card: (4.0,0.5) and (11,0.5). The intersection of the blue polygon and red polygon is inserted into the lattice and filled with cross interior material (M_{in}). The liner is then inserted **above** the blue polygon, padded by the liner thickness (lthick), and clipped by SSE red polygon if needed.

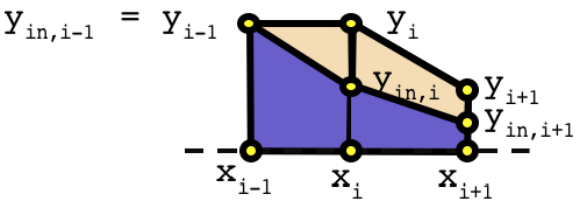
For each consecutive set of **cross** vertices, Polaris inserts a polygonal region into each of the 8 octants. The **cross** vertices are entered in the input as an x-values list followed by a y-values list of the same length. The coordinate system of the x- and y- lists is displayed in the top left plot of Fig. 3.2.6. The coordinate system is transformed based on the following rules for each octant:

- WNW, ENE: no transform,
- NNW, SSW: reflected across the diagonal line from NW to SE channel box corners,
- NNE, SSE: rotated 90 degrees about the cross center, and
- ESE, WSW: reflected across the line in the center of the horizontal cross gap.

The cross liner is inserted above the cross vertex values. The liner has a uniform thickness (lthick) and uniform material (M_{cross}). The uniform liner thickness is constructed with a miter joint at each cross vertex as shown in the following diagram:



After the set of y-values on the **cross** card, an optional list of interior y-values can be specified. The length of the interior y-values list must be equal to the length of the x- and y- lists. The optional interior y-values list is used to split the polygons into two material regions as shown in the following diagram:

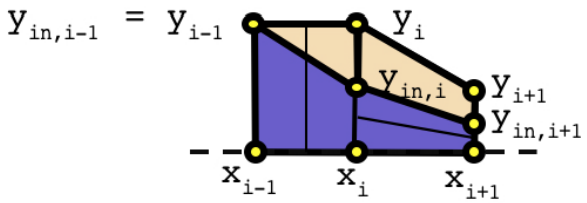


For the left-hand polygon, $y_{in,i-1}$ is the same as y_{i-1}, but $y_{in,i}$ is less than y_i. In this scenario, the trapezoid region is filled with M_{in}, and the triangular region is filled with M_{cross}. Similarly for the right-hand polygon,

the lower trapezoid is filled with M_{in} and the upper trapezoid is filled with M_{cross}. Note that the uniform liner above the y-values is not shown for simplicity.

The interior y-list values can be specified in one of two ways. First, a positive value may be entered that is greater than or equal to zero and less than or equal to the corresponding y-value. Second, a negative value may be entered that represents the relative distance of the interior y-value below the corresponding y-value. Note the Polaris input processor interprets "-0" different than "0". "-0" implies that the internal y-value is the same as the y-value. "0" implies that the internal y-value is zero. If the interior y-list is omitted, the default for all internal y-values is "-0", i.e., the polygon regions defined are completely filled with M_{in}.

In addition to the interior y-values list, two additional lists can be used to refine the spatial mesh in the x- and y- directions. The list of nx- and ny- values subdivide the polygon regions along the x- and y- directions respectively. Both lists must have one less entry than the x-, y-, and y_{in}- lists. If omitted, the default values for both the nx- and ny- lists are 1, i.e., no additional spatial refinement is applied to the polygon regions. For refinement in the y-direction, only the M_{in} material is refined. The following diagram shows nx=2 refinement for the left polygon and ny=2 refinement for the right polygon:

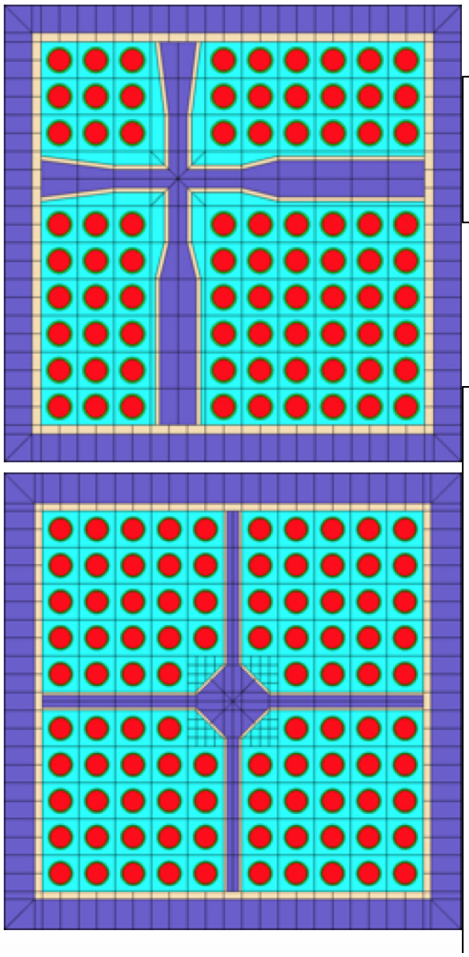


The full cross example from Fig. 3.2.6 is displayed in the top left plot of Fig. 3.2.7. The bottom plot shows a centered cross structure with a diamond water box and empty pins surrounding the water box.

For additional examples, see the polaris.6.3 regression input files described at the beginning of Sect. 3.2.2.

See also:

box, system BWR, pinmap



```
geom manual_ex : ASSM 9 2.0
hgap 1.5
box 0.5 0.0 10.5
cross 1.5 0.25 row=3 : 0 4  11 13 21
                     : 1 .5 .5 1  1
```

```
geom svea : ASSM 10 1.2
hgap 1.0
box 0.252
cross 0.3 0.1
   : .0 5.1 6.3 7.5 12.6
   : 0.2 0.2 1.4 0.2 0.2
%fuel pin
pin 1 : 0.35 0.4 : FUEL.1 CLAD.1
%empty pin
pin E : 0.6 : COOL.1 : SQR
mesh COOL.1 nx=4 ny=4
pinmap
1
1 1
1 1 1
1 1 1 1
1 1 1 1 E
1 1 1 1 E E
1 1 1 1 1 1 1
1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1
```

Fig. 3.2.7: Additional cross examples.

3.2.4.13 dxmap and dymap - pin-by-pin displacement maps

dxmap d_1 d_2 ... d_i ... d_N
dymap d_1 d_2 ... d_i ... d_N

param	type	details	default
d_i	*Real*	pin center displacement value in the x or y direction (cm)	0.0

Comments:

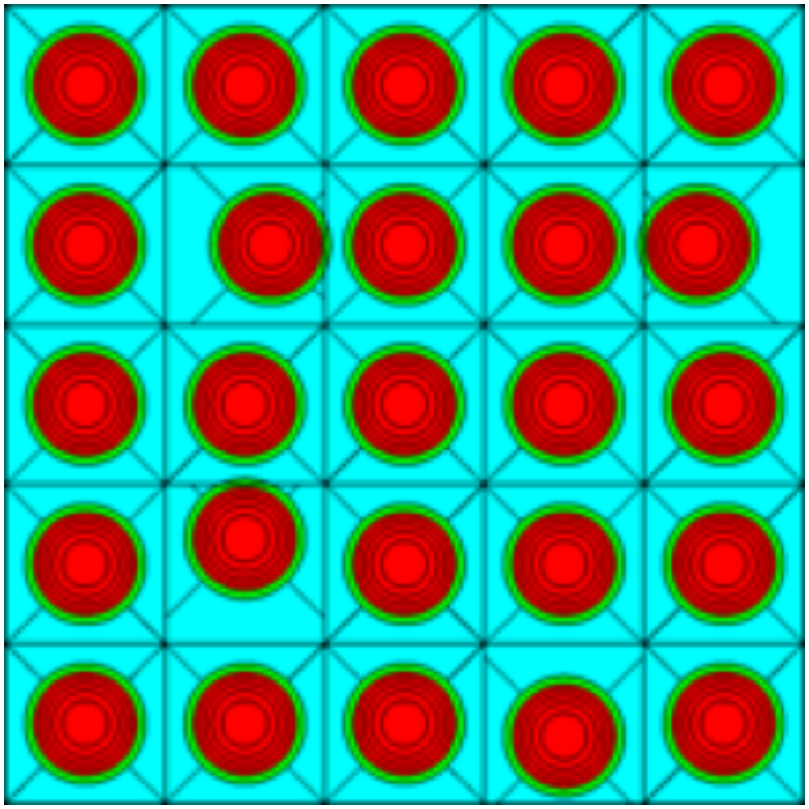
The dxmap and dymap cards displace pins from their natural position in the geometry (see comments for the pin card). If displacement maps are required, both the dxmap and dymap must be specified in the input and they must have the same length. However, the length of the displacement maps does not have to equal

of length of the pinmap if the displacement maps have reduced symmetry. For integer-sized pins greater than 1, the displacement value should be entered in the northwest corner element of the size × size subarray. For noninteger-sized pins, the displacement value should be in the corner element opposite of shared corner location. Note the following symmetry restrictions for the displacement maps:

- dy_i value must be zero on a horizontal symmetry line for an odd × odd pinmap,
- dx_i value must be zero on a vertical symmetry line for an odd × odd pinmap, and
- dx_i must equal dy_i for an element on a diagonal symmetry line.

Examples:

```
dxmap
  0.0  0.0  0.0  0.0  0.0
  0.0  0.2  0.0  0.0 -0.2
  0.0  0.0  0.0  0.0  0.0
  0.0  0.0  0.0  0.0  0.0
  0.0  0.0  0.0  0.0  0.0
dymap
  0.0  0.0  0.0  0.0  0.0
  0.0  0.0  0.0  0.0  0.0
  0.0  0.0  0.0  0.0  0.0
  0.0  0.2  0.0  0.0  0.0
  0.0  0.0  0.0 -0.1  0.0
```



See also:

pinmap

3.2.4.14 mesh - advanced material dependent meshing options

mesh MSPEC : [m=*Real*] [nx=*Int*] [ny=*Int*] [mx=*Real*] [my=*Real*]
[nr=*Int*] [ns=*Int*] [mr=*Real*] [ms=*Real*]
[nf=*Int*] [nd=*Int*] [mf=*Real*] [md=*Real*]

param	**type**	**name**	**details**	**default**
MSPEC	*MCLASS* |*MNAME*	material identifier		
nx	*Int*	# of x divisions	must be >0	MeshNumX*
ny	*Int*	# of y divisions	must be >0	MeshNumY*
nr	*Int*	# of radial rings	must be >0	MeshNumRings*
ns	*Int*	# of radial sectors	must be nonzero	MeshNumSectors*
nf	*Int*	# of faces/pin	must be >0	2**
nd	*Int*	# of gap divisions	must be >0	1
meshing multipliers				
m	*Real*	global multiplier	must be >0	1.0
mx	*Real*	x divisions	must be >0	1.0
my	*Real*	y divisions	must be >0	1.0
mr	*Real*	radial rings	must be >0	1.0
ms	*Real*	radial sectors	must be >0	1.0
mf	*Real*	faces/pin	must be >0	1.0
md	*Real*	gap divisions	must be >0	1.0

* The global mesh default values are set on the **option <GEOM>** card using the parameter name in the table above.

** Default number of faces per pin is 1. Default is 2 for **system BWR** or **system PWR.**

Examples:

```
mesh COOL : nr=3 ns=4 nx=2 ny=2 %coolant mesh: 3 ring, 4 sectors, 2 in x and y

mesh MOD.1 : nf=2 nd=4          %mesh used for wide gap (MOD.1): nf=2 nd=4

mesh MOD.2 : nf=2 nd=3          %mesh used for narrow gap (MOD.2): nf=2 nd=3

mesh FUEL : mr=2.0              %double the fuel radial mesh

mesh FUEL.2 : m=3.0             %triple all mesh values for FUEL.2

mesh CLAD : ms=0.5              %coarsen the clad sector mesh by a factor of 1/2.
```

Comments:

Polaris supports three different mesh types: 1) cylindrical mesh for CIR shapes in the pin card, 2) Cartesian mesh for SQR shapes in the pin card, and 3) a special Cartesian mesh for the region external to the pinmap region. As shown in the examples above, the mesh card is used to define, refine, or coarsen the mesh parameters for one or more of the mesh types associated with a given material class or material name. The default values for mesh parameters are defined through the option<GEOM> card and the system card. The

default values on the option<GEOM> card are nr=1, ns=1, nx=1, ny=1, nf=1, nd=1, and MeshMult=1.0. The “MeshMult” multiplier from the option<GEOM> is a global mesh multiplier applied in conjunction with any material-specific multiplier (see option<GEOM> example for details). If system BWR or system PWR is applied, new default values include nf=2, ns=8, and nr=2 (only for the channel material class). If the final mesh value is noninteger, Polaris rounds down to determine the applied value.

See also: pin, system, option<GEOM>

3.2.4.15 detector - insert a detector geometry

det DNAME : PINID at=PINID|GNAME loc=*String*
: rx=RSPEC fmat=*MNAME* rmat=*MNAME*

param	*type*	*Name*	*details*	*default*
DNAME	*Word*	detector name		*none*
PINID	*Word|Int*	detector geometry	currently only pin geometry is supported	*none*
at	*PINID|GNAME*	parent geometry	detector may be inserted either into an existing pin or assembly geometry	*none*
loc	*String*	location within parent	location is either "CENTER": center (only valid for placing within a pin geometry) "SE": southeast corner (only valid for placing within an assembly geometry)	*none*
rx	*RSPEC*	reaction specifier	has the form Units(Inc,Type) "Units": reaction units either "R" for a reaction rate or "E" for an energy deposition "Inc": incident particle type either "N" for a neutron or "G" for a gamma/photon "Type": reaction type either "CAP" for capture, "FIS" for fission, "ABS" for absorption (FIS+CAP), or “TOT” for total (gamma/photon only)	*none*
fmat	*MNAME*	flux material	material whose flux is used to calculate the reation rate (must exist in the detector geometry)	*none*
rmat	*MNAME*	reaction material	material whose reaction cross sections are to be used instead of fmat (must **not** exist in the detector geometry)	*fmat*

Examples:

```
% declare detector materials
mat DET.2 : SS304 %transmitter
```

(continues on next page)

(continued from previous page)

```
mat DET.3 : Al 2.7 %active material matrix (with u235 dopant not modeled)

% declare detector geometry
% - transmitting wire
% - active material
% - sheath
pin PD : 0.03 0.04 0.0787
: DET.2 DET.3 TUBE.1

%u235 for reaction rate material
mat DET.1 : detu 1e-5
   comp detu : WT u235=99.9 u238=0.1

% place detector D1, described by pin PD
% into instrument tube pin.IT, at the center
% with signal proportional to fission in material DET.1
% but with flux inside material matrix DET.3
det D1 : pin.PD at=pin.IT loc=CENTER
: R(n,FIS) fmat=DET.3 rmat=DET.1

% enable detector output in XFile16
opt FG DetectorEdit=D1
```

Comments:

The **detector** card is not intended to produce arbitrary reaction rates, but model a simple, discrete detector inserted into the geometry. The first line of detector input defines the detector and where it should be placed. The second line of input defines the "signal" the detector produces. The most complex piece of the input is the reaction specificer (RSPEC) which denotes the units (reaction rate or energy rate), incident particle, and type of reaction. For example, neutron absorption rate would be denoted "R(n,ABS)" and gamma energy deposition rate "E(g,CAP)".

The detector signal is collected for the material specified as the flux material, "fmat". In some cases, e.g. PWR fission detector modeling, it is desirable to model a reaction rate for fissile material that does not exist, e.g. a U-235 fission rate in an gaseous fission chamber that does not have trace amounts of U-235 present in the composition definition. The optional reaction material, "rmat", gives a way to specify this type of scenario. In this case the "fmat" would be a physical material in the model like the detector fill gas whereas the "rmat" would be another material (not present in the model) with composition and temperature defined appropriately.

3.2.5 MATERIALS

A material contains two main types of information: (1) the *composition*, or distribution of nuclides, and (2) *the properties* which include basic (required) properties like density and temperature, and as well as (optional) properties like soluble poison content, void, or grid spacer smearing. The composition is defined by a **composition** card. The basic specification for a material is shown below.

[**mat** MNAME : CNAME [dens=*Real*] [temp=*Real*] [: *properties*]]

argument	type	name	details	default
MNAME= MCLASS.MSUB	*Word. Int*	material name	used to reference this material	
CNAME	Word	composition name		

continues on next page

Table 3.2.21 – continued from previous page

dens	Real	density	basic property units: g/cm^3	composition reference density, if defined
temp	Real	temperature	basic property units: K	293
properties	Word=Value	properties	extra properties are defined with **property** cards	no extra properties

A material name has two parts, the material class, or MCLASS, and a member identifier, or MSUB. For example, FUEL.2 has an MCLASS=FUEL and an MSUB=2. **All properties are defined by MCLASS.** The composition referenced by CNAME is created with a **composition** card, as shown below.

[**comp** CNAME : CTYPE *arguments*]

Table 3.2.22: :class: longtable

argument	**type**	**name**	**details**	**default**
CNAME	*Word*	composition name	used to reference this composition later in materials and property definitions	
CTYPE	-	composition type		
	General Composition Constructors			
	NUM	number fraction		
	WT	weight fraction		
	FORM	Formula		
	CONC	Concentrations		
	Reactor Composition Constructors			
	LW	borated light water		
	UOX	uranium oxide fuel		
arguments	-	remaining arguments	depends on *CTYPE*	

Additional properties are defined with the **property** card, which defines the property PNAME for a material class MCLASS**.** The property type, *PTYPE,* determines the remaining arguments.

prop PNAME MCLASS : PTYPE *arguments*

argument	**type**	**name**	**details**	**default**
PNAME	*Word*	property name	used to reference this property	
PTYPE	*PTYPE*	property type		

continues on next page

Table 3.2.23 – continued from previous page

	SOLP	soluble poison	used to define soluble boron content	
arguments	-	remaining arguments	depends on *PTYPE*	

3.2.5.1 material - material initialization

mat MNAME : CNAME [dens=*Real*] [temp=*Real*]
[: p_1=val_1 p_2=val_2 ... p_i=val_i ... p_N=val_N]

argument	**type**	**name**	**details**	**default**
MNAME	*Word. Int*	material name	uses form MCLASS.MSUB	
CNAME	*Word*	composition name		
dens	*Real*	density	*basic property* units: g/cm^3	composition reference density
temp	*Real*	temperature	*basic property* units: K	293
p_i =val_i	PNAME=Value	properties	*additional properties*	0

Examples:

```
% define a gas gap material
mat GAP.1  : FILLGAS

% define a 3.5% enriched fuel material
comp uox_e350 : UOX 3.5
mat FUEL.1 : uox_e350 dens=10.257 temp=900

% define a cladding material
mat CLAD.1 : ZIRC4

% define a guide tube material
mat TUBE.1 : SS304

% define a control rod material
mat CNTL.1 : AIC
```

Comments:

Material properties may be set on either a **material** card or on a **state** card. If a temperature is specified rather than a density, the "temp=" key must be used to skip over the density argument.

See also:

state, composition, property

3.2.5.2 composition<NUM|WT> – general atom/wt fraction

comp CNAME: *NUM|WT*
[scale=<*PCT*>|*ABS*|*PPM*]
[norm=*Bool*]
[refdens=*Real*]
[structure=*Function*]
id_1=val_1 id_2=val_2 … id_i=val_i … id_N=val_N

---

param	*type*	*name*	*details*	*default*
CNAME	*Word*	composition name		
CTYPE	*NUM*	number fraction		
	WT	weight fraction		
scale	*PCT*|*ABS*|*PPM*	scaling factor	all values are divided by this factor PCT: percentage (divide by 100) PPM: parts per million (divide by 1e6) ABS: absolute (divide by 1)	PCT
norm	*Bool*	normalize	normalize values to 1	false
refdens	*Real*	reference density	default density for materials using this composition units: g/cm^3	*
structure	*Function*	structure	see Structure names	FREE
id_i=val_i	Word|Int=*Real*	id/value pairs	value<0 fills the remainder up to value see acceptable id forms below	

*The density property must be defined for each *material* either explicitly on the material card itself or implicitly through the "reference density" of the material's composition.

Examples:

```
% create a plutonium vector and then plutonium oxide
comp puvec : WT scale=PCT
  Pu238=1.2
  Pu239=63.3
  Pu240=21.0
  Pu241=8.6
  Pu242=5.9
comp puox : FORM puvec=1 O=2

% create an 85/10/5 Ag/In/Cd composition
```

(continues on next page)

(continued from previous page)

```
% using 10% In, 5% Cd,
% and filling the remainder up to 100% with Ag
comp aic : WT In=10 Cd=5 Ag=-100
```

Comments:

IDs in weight or number fraction-based compositions may be any of the following:

- nuclide IDs (*Int*), e.g., 92235,
- nuclide names (*Word*), e.g., U235 or u235,
- element Z numbers (*Int*), e.g., 92,
- element names (*Word*), e.g., U or u, or
- other composition names (CNAME).

See also:

composition<FORM>

3.2.5.3 composition<FORM> - general chemical formula

comp CNAME : *FORM*
[refdens=*Real*]
[structure=*Function*]
id_1=val_1 id_2=val_2 ... id_i=val_i ... id_N=val_N

param	type	name	details	default
CNAME	*Word*	composition name		
CTYPE	*FORM*	formula		
refdens	*Real*	reference density	default density for materials using this composition units: g/cm^3	*
structure	*Function*	structure	see Structure names	FREE
id_i =val_i	Word|Int=*Real*	id/value pairs	see acceptable id forms below values in atoms per molecule (e.g., H_2O is given as "H"=2 "O"=1)	

*The density property must be defined for each *material* either explicitly on the material card itself or implicitly through the "reference density" of the material's composition.

Examples:

```
% define Gd2O3 using element names
% (using elements implies natural abundances used in isotopics)
comp gd2o3 : FORM Gd=2 O=3

% define Gd2O3 using 100% Gd 155
comp gd2o3 : FORM Gd155=2 O=3

% define D2O using nuclide IDs and element names
comp d2o : FORM 1002=2 8000=3
comp d2o : FORM H2=2 O=3
```

Comments:

IDs in formula-based composition may be any of the following:

- nuclide IDs (*Int*), e.g., 92235,
- nuclide names (*Word*), e.g., U235 or u235,
- element Z numbers (*Int*), e.g., 92,
- element names (*Word*), e.g., U or u, or
- other composition names (CNAME).

See also:

composition<NUM|WT>, composition<CONC>

3.2.5.4 composition<CONC> - general number density

comp CNAME : *CONC* [refdens=*Real*]
id_1=val_1 id_2=val_2 … id_i=val_i … id_N=val_N

param	type	name	details	default
CNAME	*Word*	composition name		
CTYPE	*CONC*	concentration		
refdens	*Real*	reference density	default density for materials using this composition units: g/cm^3	**
id_i =val_i	Int=*Real*	id/value pairs	see acceptable id forms below **note: cannot use other CNAMEs for IDs in CONC input** units: #/barn-cm	

**A reference density is automatically calculated from concentrations input. If specified, it will simply scale up concentrations linearly.

```
% pyrex composition
comp pyrex_e125 : CONC
```

(continues on next page)

(continued from previous page)

```
         5010=9.63266E-04
         5011=3.90172E-03
         8016=4.67761E-02
        14028=1.81980E-02
        14029=9.24474E-04
        14030=6.10133E-04

% fuel composition
comp uox_e310_gd180 : CONC
        92234=3.18096E-06
        92235=3.90500E-04
        92236=1.79300E-06
        92238=2.10299E-02
        64152=3.35960E-06
        64154=3.66190E-05
        64155=2.48606E-04
        64156=3.43849E-04
        64157=2.62884E-04
        64158=4.17255E-04
        64160=3.67198E-04
         8016=4.53705E-02

% wet annular burnable absorber (WABA) composition
comp waba : CONC
         5010=2.98553E-03
         5011=1.21192E-02
         6000=3.77001E-03
         8016=5.85563E-02
        13027=3.90223E-02
```

Comments:

IDs in concentration-based composition may be any of the following:

- nuclide IDs (*Int*), e.g., 92235;
- nuclide names (*Word*), e.g., U235 or u235;
- element Z numbers (*Int*), e.g., 92;
- element names (*Word*), e.g., U or u; and
- SCALE-specific Nuclide IDs (*Int*), e.g., 3006000 (Only available for comp(CONC) card).

Other composition names (CNAME) *cannot* be used in a concentration definition. To easily ensure consistency of input when comparing codes, the composition input should be used in the concentrations described here. In all other cases, the other **composition** constructors are recommended because they are much simpler and easier to use.

See also:

composition<FORM>, composition<NUM|WT>

3.2.5.5 composition<LW> - borated light water

comp CNAME : *LW* [borppm=*Real*] [refdens=*Real*]

param	type	name	details	default
CNAME	*Word*	composition name		

continues on next page

Table 3.2.27 – continued from previous page

CTYPE	*LW*	light water		
borppm	*Real*	boron	parts per million by weight of natural boron (B) in light water (H_2 O)	*
refdens	*Real*	reference density	default density for materials using this composition units: g/cm^3	0.0

*The density property must be defined for each *material* either explicitly on the material card itself or implicitly through the "reference density" of the material's composition.

Examples:

```
% define a 600ppm boron moderator composition
comp mod_600ppm : LW 600

% same composition using FORM and WT
comp mod : FORM H=2 O=1
comp mod_600ppm : WT scale=PPM norm=yes
                  mod=1e6 B=600
```

Comments:

Internally, the borated light water composition is built from **composition<FORM>** and **composition<WT>** cards assuming natural boron. To use different boron isotopics such as depleted or enriched boron, the more general composition cards should be used.

See also:

composition<FORM>, composition<WT>

3.2.5.6 composition<UOX> -UO_2 fuel

comp CNAME : UOX enr=*Real* [bu=*Real*] [refdens=*Real*]

param	**type**	**name**	**details**	**default**
CNAME	*Word*	composition name		
CTYPE	*UOX*	uranium dioxide		
enr	*Real*	enrichment	U-235 wt. % (see **composition<ENRU>** for formula)	
bu	*Real*	burnup	only available for $0 \leq bu \leq 100$ units: GWd/MTU	0*
refdens	*Real*	reference density	default density for materials using this composition units: g/cm^3	**

*Note that the burnup parameter should not be specified. It is a future option to create a fixed, representative, burned fuel composition. The composition is interpolated using linear interpolation from an internal burnup- and enrichment-dependent data matrix.

**The density property must be defined for each *material* either explicitly on the material card itself or implicitly through the "reference density" of the material's composition.

Examples:

```
% define a 4.95% enriched fuel composition with a reference density
comp uox_495 : UOX 4.95 refdens=10.25

% same result as above
comp u_495 : ENRU 4.95
comp uox_495 : FORM u_495=1 O=2
```

Comments:

Internally, the UO_2 composition is built from **composition<FORM>** and **composition<ENRU>** cards.

See also:

composition<ENRU>, composition<FORM>

composition<USI> – U_2Si_3 fuel

comp CNAME : USI enr=*Real* [bu=*Real*] [refdens=*Real*]

param	**type**	**name**	**details**	**default**
CNAME	*Word*	composition name		
CTYPE	*USI*	uranium silicide		
enr	*Real*	enrichment	U-235 wt. % (see **composition<ENRU>** for formula)	
bu	*Real*	burnup	only available for $0 \leq bu \leq 100$ units: GWd/MTU	0*
refdens	*Real*	reference density	default density for materials using this composition units: g/cm^3	**

*Note that the burnup parameter should not be specified. It is a future option to create a fixed, representative, burned fuel composition. The composition is interpolated using linear interpolation from an internal burnup- and enrichment-dependent data matrix.

**The density property must be defined for each *material* either explicitly on the material card itself or implicitly through the "reference density" of the material's composition.

Examples:

```
% define a 4.95% enriched fuel composition with a reference density
comp usi_495 : USI 4.95 refdens=11.54
```

(continues on next page)

(continued from previous page)

```
% same result as above
comp u_495 : ENRU 4.95
comp usi_495 : FORM u_495=2 Si=3
```

Comments:

Internally, the U_2Si_3 composition is built from **composition<FORM>** and **composition<ENRU>** cards.

See also:

composition<ENRU>, composition<FORM>

3.2.5.7 composition<UN> - UN fuel

comp CNAME : UN enr=*Real* [n15enr=*Real*][bu=*Real*] [refdens=*Real*]

param	type	name	details	default
CNAME	*Word*	composition name		
CTYPE	*UN*	uranium nitride		
enr n15enr	*Real* *Real*	enrichment nitrogen enrichment	U-235 wt. % (see **composition<ENRU>** for formula) N-15 wt. %	100%*
bu	*Real*	burnup	only available for $0 \leq bu \leq 100$ units: GWd/MTU	0**
refdens	*Real*	reference density	default density for materials using this composition units: g/cm^3	***

* Natural nitrogen is 0.4 wt. % ^{15}N, but the reactivity penalty of ^{14}N warrants using the highest ^{15}N composition possible.

**Note that the burnup parameter should not be specified. It is a future option to create a fixed, representative, burned fuel composition. The composition is interpolated using linear interpolation from an internal burnup- and enrichment-dependent data matrix.

***The density property must be defined for each *material* either explicitly on the material card itself or implicitly through the "reference density" of the material's composition.

Examples:

```
% define a 4.95% enriched fuel composition with a reference density
comp usi_495 : UN 4.95 refdens=11.3

% same result as above
comp u_495  : ENRU 4.95
comp un_495 : FORM u_495=1 7015=1

% define a 3.25% enriched fuel composition with a reference density and
% specify natural 15N composition
Comp un_495 : UN 3.25 refdens=11.3 n15enr=0.4
```

Comments:

Internally, the uranium nitride composition is built from **composition<FORM>**, **composition<WT>**, and **composition<ENRU>** cards.

See also:

composition<ENRU>, composition<FORM>

3.2.5.8 composition<ENRU> – enriched uranium

comp CNAME : ENRU enr=*Real* [refdens=*Real*]

param	**type**	**name**	**details**	**default**
CNAME	*Word*	composition name		
CTYPE	*UOX*	uranium dioxide		
enr	*Real*	enrichment	*U-235 wt. %	
refdens	*Real*	reference density	default density for materials using this composition units: g/cm^3	**

*The following formula from [POLARISGod14] is used to determine the ^{234}U and ^{236}U wt% from the ^{235}U enrichment. Note that this formula is only valid for U-235 enrichments less than 10 wt%.

$w_{u234} = 0.007731*(enr)^{1.0837}$

$w_{u236} = 0.0046*enr$

$w_{u238} = 100 - w_{u234} - enr - w_{u236}$

**The density property must be defined for each *material* either explicitly on the material card itself or implicitly through the "reference density" of the material's composition.

Examples:

```
% 5% enriched metal fuel
comp umetal : ENRU 5
```

Comments:

This composition for enriched uranium is used internally to create uranium oxide, silicide, and nitride fuels using the **composition<UOX>**, **composition<USI>**, and **composition<UN>** cards.

See also:

composition<UOX>, **composition<USI>**, **composition<UN>**

3.2.5.9 composition library (pre-defined)

The Polaris composition library contains predefined compositions that may be used without a constructor by simply referencing the CNAME below. Each predefined library composition has a reference density, so it can be used directly on a material card.

Table 3.2.32: Standard molecular compositions

CNAME	*Description*
H2O	light water with structure=BOND(H2O)
B4C	Boron carbide burnable poison material
Er2O3	Erbium oxide burnable poison material
Gd2O3	Gadolinium oxide burnable poison material
SiC	Silicon carbide
ZrH	zirconium hydride alloy with structure=CRYS(orthorhombic_zrh)
Zr5H8	zirconium hydride alloy with structure=CRYS(cubic_zrh)
ZrH2	zirconium hydride alloy with structure=CRYS(tetragonal_zrh)
fillgas	Helium gas
Cr2O3	Chromium dioxide (chromia, Cr_2O_3)
Al2O3	Aluminum dioxide (alumina, Al_2O_3)
BeO	Beryllium dioxide (beryllia, BeO)

Table 3.2.33: Standard reactor mixtures and alloys

CNAME	*Description*
aic	Ag-In-Cd control rod absorber material
pyrex	Pyrex glass
zirc2	Zircaloy-2 clad material
zirc4	Zircaloy-4 clad material
ss304	Stainless Steel 304
ss316	Stainless Steel 316
inc718	Inconel 718
water	H2O with trace amount of boron
pyroc	Pyrolytic carbon, C with structure=CRYS(pyrolytic_c)
graphite	Graphite, C with structure=CRYS(hexagonal_c)

Table 3.2.34: Structure names.

structure	*Description*	*Nuclide* *Cross section ID**
BOND(H2O)	H,O in liquid water	H-1 → 1001 H-2 → 1002
FREE	atoms allowed to orient freely (no structure)	H-1 → 8001001 H-2 → 8001002
CRYS(orthorhombic_zrh)	Zr,H in zirconium hydride alloy with orthorhombic crystal structure (gamma phase)	H-1 → 7001001 Zr-90 → 1040090 Zr-91 → 1040091 Zr-92 → 1040092 Zr-93 → 1040093 Zr-94 → 1040094 Zr-95 → 1040095 Zr-95 → 1040096
CRYS(cubic_zrh)	Zr,H in zirconium hydride alloy with cubic crystal structure (delta phase)	H-1 → 7001001 Zr-90 → 1040090 Zr-91 → 1040091 Zr-92 → 1040092 Zr-93 → 1040093 Zr-94 → 1040094 Zr-95 → 1040095 Zr-95 → 1040096
CRYS(tetragonal_zrh)	Zr,H in zirconium hydride alloy with tetragonal crystal structure (epsilon phase)	H-1 → 7001001 Zr-90 → 1040090 Zr-91 → 1040091 Zr-92 →1040092 Zr-93 → 1040093 Zr-94 → 1040094 Zr-95 → 1040095 Zr-95 → 1040096
CRYS(pyrolytic_c)	C in pyrolytic crystal structure (graphite)	C → 3006000
CRYS(hexagonal_c)	C in pyrolytic carbon (additional covalent bonds compared to graphite)	C → 3006000
CRYS(hexagonal_beo)	Be,O in beryllium oxide with hexagonal structure	Be-9 → 5004009 O-16 → 5008016

Note: The cross section IDs can only be used on composition cards with the CONC variant to input number densities directly.

3.2.5.10 property<SOLP> - soluble poison by weight

prop PNAME M1 ... : SOLP poison [scale=<*PPM*>|*PCT*|*ABS*]

param	type	name	details	default
PNAME	*Word*	property name	property value p ≥ 0	
M1 ...	MCLASS	material class	one or more material classes to gain this property	
PTYPE	*SOLP*	soluble poison		
poison	CNAME	soluble poison composition name		
scale	PCT|ABS|PPM	scaling factor	all values are divided by this factor PCT: percentage (divide by 100) PPM: parts per million (divide by 1e6) ABS: absolute (divide by 1)	PPM

Examples:

```
% define a soluble boron property for moderator
% and coolant material classes
% using natural boron
prop boron MOD COOL : SOLP B

% investigate coolant crud/impurity activation
%  1. define a general impurity property to mix in coolant,
  comp crud : NUM Ni=12.7 Cr=2.3 Fe=-100 %mostly Fe
  prop impurity COOL : SOLP crud
% 2. create coolant material with 100ppm of crud
  mat COOL.1 : LW dens=0.75 : impurity=100
% 3. make sure to "deplete" coolant so crud gets activated
  deplete COOL=true
```

Comments:

None

See also:

pinmap, **control**, **insert**

3.2.5.11 property<DOPANT> - fuel dopant by weight

prop PNAME M1 . . . : DOPANT dopant [scale=<*PPM*>|*PCT*|*ABS*]

param	type	name	details	default
PNAME	*Word*	property name	property value $p \geq 0$	
M1 . . .	MCLASS	material class	one or more material classes to gain this property	
PTYPE	*DOPANT*	fuel dopant		
dopant	CNAME	Fuel dopant composition name		
scale	PCT|ABS|PPM	scaling factor	all values are divided by this factor PCT: percentage (divide by 100) PPM: parts per million (divide by 1e6) ABS: absolute (divide by 1)	PPM

Examples:

```
% define three dopants properties for the fuel
% using predefined Cr2O3, Al2O3, and BeO
% then dope (deferred definition) fuel with 0.3% chromia,
% 0.2% alumina, 0.1% beryllia
prop Cr2O3 FUEL: DOPANT Cr2O3
prop Al2O3 FUEL: DOPANT Al2O3
prop BeO   FUEL: DOPANT BeO
mat FUEL.1 : uox_e310  temp=565 : Cr2O3=1000

% can also set the values through system properties
% works for both PWR and BWR
=polaris
title "pincell with UOX fuel doped with Cr2O3, Al2O3, and BeO"
lib "broad_lwr"
sys PWR
geom wec17 : ASSM 1 1.26
comp uox_e310 : UOX 3.10
mat FUEL.1 : uox_e310  10.5 : cr2o3=3000 al2o3=2000 beo=1000
pin 1 : 0.4096 0.418 0.475 : FUEL.1 GAP CLAD
end
```

Comments:

None

See also:

solp

3.2.5.12 property<TWOPHASE> - density property used to control two phase mixtures

property PNAME MCLASS : TWOPHASE liqden=*Real* vapden=*Real*

param	type	name	details	default
PNAME	*String*	name for this property	Name with which to refer to this TWOPHASE property in other cards.	
MCLASS	*MCLASS*	material class(es) for which this property is valid	Generally COOL or MOD but can be any material.	
liqden vapden	*Real* *Real*	density of the liquid density of the vapor	In g/cm^3. In g/cm^3.	

Examples:

```
% define a TWOPHASE combination for moderator and coolant
prop vf MOD COOL : TWOPHASE 0.7 0.04
```

Comments:

The two-phase property is used to model materials that exist in two phases, such as water existing in a liquid or gas. This is typically used for water in BWR systems where the density of the water changes as one moves axially through the core. This property is defined by the liquid and vapor density. Later, in the state or history cards, a number between 0% and 100% is entered and the final density is calculated by interpolating between the liquid and vapor density values. The only requirement is that the liquid density must be greater than the vapor density.

3.2.5.13 deplete - material depletion and decay

deplete M_1=*Bool* M_2=*Bool* ... M_i=*Bool* ... M_N=*Bool*

param	type	name	details	default
M_i	MNAME|MCLASS	list of material names or material classes	use ALL for all materials	

Note: Only one deplete card is allowed in an input. ALL only applies in the first position.

Examples:

```
% turn on depletion/decay for two new materials
sys PWR
deplete MyMaterial=true MyOtherMaterial=true

% activate/deplete/decay every material
deplete ALL=true
```

(continues on next page)

(continued from previous page)

```
% impose strict conditions
sys PWR
deplete ALL=false FUEL=true CLAD=true
```

Comments:

The **deplete** card not only instructs Polaris to deplete a material, but also to solve the Bateman equations with ORIGEN for that material. Thus if the flux/power is zero, only materials that are flagged to "deplete" will undergo decay. The **deplete** card modifies the depletables included in a **system** card to avoid the situation in which "deplete MyMaterial=true" would make only MyMaterial depletable. Thus to completely re-specify the depletable materials, "ALL=false" should be used as the first argument. This is in contrast to the **basis** card, which completely specifies a new power basis.

See also:

material, shield, basis

3.2.5.14 basis - power basis materials

basis M_1=*Bool* M_2=*Bool* ... M_i=*Bool* ... M_N= *Bool*

param	type	name	details	default
M_i	MNAME|MCLASS	List of material names or material classes	use ALL for all materials	ALL

Note: Only one basis card is allowed per input. ALL is only allowed in the first position.

Examples:

```
% use only FUEL materials as the basis
basis ALL=no FUEL=YES

% Specify FUEL.3 as the basis
basis ALL=no FUEL.3=YES
```

Comments:

The **basis** card is used to specify the materials to use in power normalization. By default, the energy release from all materials is taken into account, including (n,gamma) reactions in structural materials such as cladding. It is not recommended to change the default of ALL in most situations. Exceptions include (1) when comparing results to other codes that only use fuel in the basis and (2) fixing the power in a specific pin is known-a material should be created only for that pin, and the power basis should be specified for that material only. The **basis** card overrides any power basis imposed by a **system** card. Thus it behaves differently than a **deplete** card, which is combined with depletable materials imposed by a system card.

See also:

material, shield, deplete

3.2.5.15 shield - cross section self-shielding expansion specification

shield M_1=*XTYPE* M_2=*XTYPE* ... M_i=*XTYPE* ... M_N= *XTYPE*

param	type	name	Details	default
M_i	MNAME| MCLASS	list of material names or material classes	use ALL for all materials	
XTYPE	N|P|R|S	self-shield ing expansion type	shield across various mesh elements N: no expansion P: pins R: rings (P implicit)	R

Note: Only one shield card is allowed per input. ALL is only allowed in the first position.

Examples:

```
%create a unique self-shielded FUEL cross sections in each pin
%consider all other materials to have a single self-shielded cross section
shield ALL=N FUEL=P

%assess effect of self-shielding each pin's cladding
shield CLAD=P

%re-specify self-shielding to be P by default, R for the FUEL
shield ALL=P FUEL=R
```

Comments:

The **shield** card controls how materials are internally expanded for self-shielding purposes. By default, Polaris expands all materials across pins and rings (R). For example, a fuel region defined on a **pin** card as having 10 rings will be expanded internally to have 10 different self-shielded cross sections. Because the R option also implicitly includes the P option, each instance of that pin will also get different cross sections.

When using specific systems (e.g., **system** *PWR*), this card is generally not needed. The **shield** card modifies the self-shielding options included in a **system** card. Thus, to completely re-specify the expansion, use "ALL=N" as the first argument. This is in contrast to the **basis** card, which completely specifies a new power basis.

See also:

material, deplete, system

3.2.6 STATE

The idea of a "state" or "statepoint" is a standard concept in lattice physics calculations. In Polaris, the concept of state is mostly tied to the values of material properties. The *base state* for a calculation is determined as follows:

1. The base state is initialized with any property values set on **material** cards.
2. The base state is updated with any **state** cards that apply to ALL.

3. The base state is updated with any other **state** cards, and the **power** card is used to set the base state power.

This sequence ensures that the state does not change, even if the order of inputs changes. A **time** or **burnup** card is then used to initiate a calculation as a function of time or burnup, thus producing a sequence of states. A **branch** block is used to perform branches off the base state at specific times or burnups.

single value mode:

state NAME : p_1=val_1 p_2=val_2 ... p_i=val_i ... p_N=val_N
[NAME : p_1=val_1 p_2=val_2 ... p_i=val_i ... p_N=val_N ...]

array mode with M burnup/time values, only in **history** *block:*

state NAME : p_1=val_1 [val_2 ... val_i ... val_M]
p_2=val_2 [val_2 ... val_i ... val_M]
...
p_i=val_1 ...
p_N=val_1 ...
[NAME : p_1=val_1 ...]

param	type	name	details	default
NAME	MNAME|MCLASS, or INAME, or GNAME	material name/class insert/cont rol geometry	use ALL for all materials in=yes|no pres=yes|no	
p_i	*PNAME*	property name		
val_i	*Value*	property value		

Examples:

```
state ALL : temp=293
     COOL : dens=1.0 boron=1100
    BPMAP : in=yes

read history
  %--------------------------------------%
  % cycle 1
  %--------------------------------------%
  pow 39
  state   ALL :  temp=600
         COOL : boron=900 850 700 600 400
                dens =0.6
         FUEL : temp =900 910 920 890 880
  dt                  50  50 100 100 200
  pow 0
  dt 70
```

(continues on next page)

(continued from previous page)

```
  %-------------------------------------%
  % cycle 2
  %-------------------------------------%
  state  BPMAP :    in= no
          COOL : boron=900 850 700 600 300
          FUEL : temp =900 910 920 890 880
  pow                   42  41  38  39  37
  dt                    50  50  50 100 250
  pow 0
  dt 80
end history
```

Comments:

The **state** card is used to specify properties for different materials, control maps, control blades, and insert maps. The **state** card can specify one or multiple properties simultaneously. The property specifications can either be a single value or multiple values, each value corresponding to a burnup/time step in the burnup/time input card (**bu|bui|dbu|t|ti|dt**).

The **state** card can be used outside a **history** block or inside a **history** block. Outside the history block, i.e., at the "root" input level, a single **state** card can initialize property values to a single value. If one or more power histories are defined at the root input level, the state properties are constant throughout the calculation. The state properties are only modified for **branch** calculations.

Time-dependent state properties are allowed through the **history** block. One or more power histories are allowed inside a history block. Each power history contains a **power** card (single value or array value) and a burnup/time card (**bu|bui|dbu|t|ti|dt**). Before the burnup/time card, a **state** card can be used to define one or more state properties. The property specifications are either a single value or an array of values that correspond to each burnup/time step. If a property is omitted in a **state** card, the value is defined based on the following precedence:

- the last value specified through the closest preceding **state** card in a given **history** block, or
- the value specified in the **state** card at the root input level, or
- the default property value.

By default, all control or insert geometries are not inserted (in=no). Material property defaults are defined on the **material** card or through the **system** card.

See also:

history, system

3.2.6.1 power - total power

pow [: p_1 p_2 ... p_i ... p_N]

param	type	name	details	default
p_i	Real	list of specific powers in W/g initial heavy metal		0

Examples:

```
% set power to 35 W/gIHM
power 35.0

% provide a power history
% must have same number of values as following time/burnup card
power : 35.0 40.0 45.0 45.0 40.0 5.0 0.0
time  : 10   20   30   40   50   60  70
```

Comments:

The **power** card specifies the total power of the basis materials specified by a **basis** card. The **power** value may be specified only once.

See also:

t, bu, history, basis, state<MNAME>

3.2.6.2 bu - initiate calculation with cumulative burnups

bu [b_1 b_2 ... b_i ... b_N]
[: units=<*GWD/MTIHM*>|*MWD/MTIHM*]

param	type	name	default
b_i	*Real*	list of burnups	0
units	GWD/MTIHM|MWD/MTIHM	burnup units	GWD/MTIHM

Examples:

```
% simple depletion case with constant power and absolute/cumulative burnups
power 40
bu     5 10 15 20 30 40 50 60 80

% using MWd/MTIHM units with variable power
% 40 W/gIHM for 0→5000 MWD/MTIHM, then 30 W/gIHM for 5000→10000 MWD/MTIHM
power  40    30
bu    5000 10000 MWD/MTIHM

% combine burnup and time cards
% 20 W/gIHM for 0→5 then 5→10 GWD/MTIHM steps, then
% 40 W/gIHM for a 5-day step then 30 W/gIHM for a 5-day step
power 20
bu     5 10 GWD/MTIHM
power 40 30
dt     5  5 DAYS
```

Comments:

The **bu** card initiates a calculation for a given sequence of cumulative/absolute burnups. A burnup or time card usually follows a **power** card, the two effectively specifying the power history. If multiple burnups are given, then the **power** card must have either a single power or a list of powers the same size as the list times. A value of 0 is implicit at the beginning of the first burnup list. Multiple burnup/time cards may be specified in an input. This can be convenient for switching units or changing from burnup-based to time-based depletion. Internal automatic substeps are always in effect unless modified with the **option<DEPL>** card.

See also:

t, dt, ti, bui, dbu, power, option<DEPL>, branch, deplete

3.2.6.3 bui - initiate calculation with cumulative burnups (with restart)

bui [b_1 b_2 … b_i … b_N]
[: units=<*GWD/MTIHM*>|*MWD/MTIHM*]

param	type	name	default
b_i	*Real*	list of burnups	0
units	GWD/MTIHM|MWD/MTIHM	burnup units	GWD/MTIHM

Examples:

```
power 30
  bui    5 10   %equivalent to: bu 5 10

  power 40
  bui    5 10   %equivalent to: bu 15 20
```

Comments:

The **bui** card initiates a calculation for a given sequence of cumulative burnups. If only one burnup list is provided, the **bui** card is identical to the **bu** card. For any subsequent burnup list, the **bui** card specifies cumulative burnups that restart at zero at the beginning of each list (see example above).

See also:

t, dt, ti, bu, dbu, power, option<DEPL>, branch, deplete

3.2.6.4 dbu - initiate calculation with incremental burnups

dbu [b_1 b_2 … b_i … b_N]
[: units=<*GWD/MTIHM*>|*MWD/MTIHM*]

param	type	name	details	default
b_i	*Real*	list of incremental burnups		0
units	GWD/MTIHM|MWD/MTIHM	burnup units		GWD/MTIHM

Examples:

```
% incremental burnups equivalent to
%   power 40
```

(continues on next page)

(continued from previous page)

```
%   bu 0 5 10 15 20 30 40 50 60 80
power 40
dbu 5 5 5 5 10 10 10 10 20
```

Comments:

The **dbu** card initiates a calculation for a given sequence of *incremental* burnups. Otherwise, it is identical to the **bu** card for specifying cumulative burnups.

See also:

t, dt, ti, bu, bui, power, option<DEPL>, branch, deplete

3.2.6.5 t - initiate calculation by cumulative time

t [t_1 t_2 ... t_i ... t_N]
[:units=*SECONDS*|*MINUTES*|*HOURS*|<*DAYS*>|*YEARS*]

param	type	name	details	default
t_i	*Real*	list of times	0	0
units	*SECONDS* | *MINUTES* | *HOURS* | *DAYS* | *YEARS*	time units		*DAYS*

Examples:

```
% burn with 40 W/gIHM for 300 days in 100-day increments
power 40
t 100 200 300

% simulate 2 cycles of time-dependent irradiation with shutdown cooling
% note that time defaults to DAYs
%
% cycle 1
power 40   30   30   30
t    100  200  300  400
power  0
t    415
%
% cycle 2
power 30   20   20   20
t    515  615  715  815
power  0
t    830
```

Comments:

The **t** card initiates a calculation for a given sequence of cumulative/absolute times. One of the time cards (**t**, **dt**, or **ti**) is required to model periods of decay in conjunction with **power** 0. Otherwise, the time card **t** is similar in functionality to the burnup **bu** card but with different units.

See also:

ti, dt, bu, bui, dbu, power, option<DEPL>, branch, deplete

3.2.6.6 ti - initiate calculation by cumulative time (with restart)

t [t_1 t_2 ... t_i ... t_N]
[:units=*SECONDS*|*MINUTES*|*HOURS*|<*DAYS*>|*YEARS*]

param	type	name	details	default
t_i	*Real*	list of times	0	0
units	*SECONDS*|*MINUTES*|*HOURS*|*DAYS*|*YEARS*	time units		DAYS

Example with t card:

```
% cycle 1
power 40
t 100 200 300 400

power  0
t    415

% cycle 2
power 30
t 515  615  715  815

power  0
t    830
```

Equivalent example with ti card:

```
% cycle 1
power 40
ti 100 200 300 400

power  0
ti    15

% cycle 2
power 30
ti    100 100 100 100

power  0
ti    15
```

Comments:

The **ti** card initiates a calculation for a given sequence of cumulative times. If only one time list is provided, the **ti** card is identical to the **t** card. For any subsequent time list, the **ti** card specifies cumulative times that restart at zero at the beginning of each list (see example above).

See also:

t, dt, bu, bui, dbu, power, option<DEPL>, branch, deplete

dt - initiate calculation by incremental time

t [t_1 t_2 ... t_i ... t_N]
[:units=*SECONDS*|*MINUTES*|*HOURS*|<*DAYS*>|*YEARS*]

param	type	name	details	default
t_i	*Real*	list of times		0
units	*SECONDS*|*MINUTES*|*HOURS*|*DAYS*|*YEARS*	time units		DAYS

Examples:

```
% burn with 40 W/gIHM for 300 days in 100-day increments equivalent to
%     power 40
%     t 100 200 300
power 40
dt 100 100 100

% decay for 30 minutes
power 0
dt 30 MINUTES
```

Comments:

The **dt** card is identical to the cumulative time card **t** except that the values given are incremental.

See also:

t, ti, bu, bui, dbu, power, option<DEPL>, branch, deplete

3.2.6.7 branch - instantaneous change

read branch [BNAME]
add ...
[**add** ...]
end branch

param	type	name	details	default
BNAME	*Word*	branch name		DEFAULT
allowed cards in branch block				
add	-	-	adds a list of states to branch on	

Examples:

```
% fuel temperature and boron branches (results in 7 total states)
read branch
  add FUEL : temp=800 1000 1200
  add COOL : boron=0 400 800 1400
end branch

% branch to different
% fuel temp/coolant temp/coolant density, synchronizing
% states (results in 3 total states)
read branch
  %            state 1    2    3
  add FUEL : temp=800  1000 1200
      COOL : temp=565   585  620
      COOL : dens=0.73 0.71 0.68
end branch
```

Comments:

The **branch** card initiates so-called "branch" calculations, i.e., instantaneous changes of state at specific burnups/times during the base depletion sequence of calculations. The syntax for the **add** card is identical to the **state** card except, instead of taking a list of different properties and their values, it takes *a single property* and a list of values. Note that a **time** or **burnup** card is not necessary-if not found, branches will be performed at every burnup/time specified in the base state. The initial state for *any* **branch** card is the base state as specified in the main file. This means **branch** cards have no knowledge of one another.

See also:

add, bu, t, title

3.2.6.8 history - time-dependent history

read history [HNAME]

power . . .

[state . . .]

bu|bui|dbu|t|ti|dt . . .

[power . . .

[state . . .]

bu|bui|dbu|t|ti|dt . . .]

. . .

end history

param	**type**	**name**	**details**	**default**
HNAME	*Word*	history name		DEFAULT
allowed cards in history block				
power	-	-	specific power	
state	-	-	state properties	
bu|bui|dbu|t|ti|dt	-	-	burnup or time	

Examples:

```
state ALL : temp=293
     COOL : dens=1.0 boron=1100
    BPMAP : in=yes

read history
  %-------------------------------------%
  % cycle 1
  %-------------------------------------%
  pow 39
  state   ALL :  temp=600
         COOL : boron=900 850 700 600 400
                dens =0.6
         FUEL : temp =900 910 920 890 880
  dt                   50  50 100 100 200
  pow 0
  dt 70
  %-------------------------------------%
  % cycle 2
  %-------------------------------------%
  state  BPMAP :    in= no
          COOL : boron=900 850 700 600 300
          FUEL : temp =900 910 920 890 880
  pow                   42  41  38  39  37
  dt                    50  50  50 100 250
  pow 0
  dt 80
end history
```

Comments:

The **history** card initiates a time-dependent calculation with user defined power history and time-dependent material or geometry properties. Each history block is independent from one another. Each history calculation generates an ORIGEN binary concentration file with the name *filename_hname.f71* where *filename* is the root name of the input file and *hname* is the name for the history. Similarly, if few-group cross section files are requested, the filenames are *filename.hname.t16*. If ORIGEN binary library files are requested, the names are *filename.hname.XXX.f33* where *XXX* is the name of the material (or system) for the corresponding library.

History calculations are also allowed in conjunction with branch calculation. At this time, all history calculations will perform all branch calculations defined in the input file. Selection of the burnup values for branch calculations is available on the **option <FG>** card.

See also:

state, branch, option<FG>, power, bu, bui, dbu, t, ti, dt

3.2.6.9 add<MNAME> - material branch

add MNAME|MCLASS

[incr=*Bool*]

[scale=<*ABS*>|*PCT*] :

PNAME=val_1 val_2 ... val_i ... val_N

[MNAME|MCLASS

...]

param	type	name	details	default
MNAME|MCLASS	-	material name or material class	use ALL for all materials	
incr	*Bool*	increment	values are added to reference value	false
scale	*ABS|PCT*	Scaling	scaling ABS: absolute units PCT: percentage units	*ABS*
PNAME	-	property name		
val_i	*Value*	list of property values		

Examples:

```
% fuel temperature branches using incremental
% changes from the base state of 900 K
state FUEL : temp=900
read branch
  add FUEL incr=true : temp=-200 -100 +100 +200 +500
end branch


% material properties may be varied together (synchronized)
% by chaining additional material/properties together
% the first block below results in 2 states
% the second is 6 states
read branch
  add FUEL : temp=900 1200
       FUEL : dens=10.4 10.3
       COOL : dens=0.7  0.65
end branch
read branch
  add FUEL : temp=900 1200
  add FUEL : dens=10.4 10.3
  add COOL : dens=0.7  0.65
end branch
```

Comments:

The **add** card is only valid inside a **branch** block. This version adds a set of branches for a specific material name (MNAME) or class (MCLASS). Branches are always with respect to the base state. Although similar to the **state** card, the **add** card has a single property name and a list of values. The **state** card has a list of *property=value* pairs.

See also:

material, state<MNAME>

3.2.6.10 add<INAME> - insert/control branch

add INAME : in=*Bool1*[*Bool2*]

Table 3.2.52: Options for control element insertion branch cases

param	type	name	details	default
INAME	-	insert name or control element name		
in	Bool	list of insertion states	"in=" is required	

Examples:

```
% branch to remove WABA inserts
state InsWABA4 : in=true
read branch
  add InsWABA4 : in=false
end branch

% synchronize rods in with material branches (5 states)
read branch
  add BankD : in=true false false false true
      FUEL  : temp=600 900 1200 2000 2000
end branch

% swap control banks
read branch
  add BankB : in=true  false false
      BankC : in=false true  false
      BankD : in=false false true
end branch
```

Comments:

This form of the **add** card is required to add branches to insert/remove control elements or inserts. Given that only two possible states exist, specifying "true false" will result in a calculation at the *other* state not specified by the *base state*.

See also:

insert, control, state<INAME>

3.2.6.11 add<GNAME> - geometry branch

add GNAME : pres=*Bool1*[*Bool2*]

param	type	name	details	default
GNAME	-	geometry name		
pres	Bool	list of geometry states	"pres=" is required	

Examples:

```
% perform a reflector calculation on a branch
state ReflectorNode : pres=no
read branch
  add ReflectorNode : pres=yes
end branch
```

Comments:

This form of the **add** card is required to add branches for new geometry, such as reflector calculations. Given that only two possible states exist, specifying "true false" will result in a calculation at the *other* state not specified by the *base state*.

See also:

geometry, state<GNAME>

3.2.7 OPTIONS

An extensive set of options is provided for manipulating the solvers and output. Most **option** cards support a *key*=*value* style of input, with reasonable defaults in place for all parameters.

3.2.7.1 option<KEFF> - eigenvalue

opt *KEFF* [key_1=val_1 key_2=val_2 ... key_i=val_i ... key_N=val_N]

key	*value type*	*details*	*default*
NumAzim	*Int*	number of azimuthal angles per octant	20
NumPolar	*Int*	number of polar angles per octant	3
PnOrder	*Int*	scattering order	2
RaySpacing	*Real*	spacing between MoC rays	0.04 cm
developer options (generally should not change)			
PolarScheme	*String*	polar quadrature “TY”: Tabuchi-Yamamoto quadrature “DECART”: DeCART quadrature “LO”: Leonard optimal quadrature “CACTUS”: Cactus quadrature	“TY”
EigSolver	*String*	eigenvalue solver type “CMFD” (single-assembly default) “POWER” (reflector default)	
UpscatterSolver	*String*	upscatter solver type “GAUSS_SEIDEL”, (default if EigSolver=”CMFD”) “BICGSTAB” (default if EigSolver=”POWER”) “GMRES”	
DownscatterSolver, UpscatterInnerSolver	*String*	within-group solver type “SOURCE”, (default if EigSolver=”CMFD”) “BICGSTAB” (default if EigSolver=”POWER”) “GMRES”	
EigMaxIterations	*Int*	maximum number of eigenvalue iterations	1000
DownscatterMaxIterations, UpscatterMaxIterations, UpscatterInnerMaxIterations	*Int*	maximum iterations for multigroup solves default is 1 for EigSolver=”CMFD” default is 1000 for EigSolver=”POWER”	
EigTolerance	*Real*	eigenvector L_2 residual tolerance	1.0
EigFissionTolerance	*Real*	eigenvector $L_¥$ relative fission error tolerance	3e-4
DownscatterTolerance, UpscatterTolerance, UpscatterInnerTolerance	*Real*	L_2 residual tolerance for multigroup solve	3e-6
EigVerbosity	*String*	msg file print level for eigenvalue iterations “NONE”, “LOW”, “HIGH”	“LOW”
DownscatterVerbosity, UpscatterVerbosity, UpscatterInnerVerbosity	*String*	msg file print level for multigroup iterations “NONE”, “LOW”, “HIGH”	“NONE”
CmfdRelaxation	*Real*	relaxation parameter for CMFD solution update	1.0
UpscatterSuperGroup	*Int*	number of energy groups in the inner iteration in the upscatter calculation	1
DownscatterSuperGroup	*Int*	number of energy groups in the inner iteration in the downscatter calculation	1
PartitionUpscatter	*Bool*	Partition problem into downscatter and upscatter energy regions default is true for EigSolver=”POWER” default is false for EigSolver="CMFD"	
ReverseRay	*Bool*	true – store attenuation coefficients from forward ray calculation and reuse in reverse ray calculation false – recompute attenuation coefficients for reverse ray	true
CyclicRay	*Bool*	true – continue particle tracks after reaching the boundary false – separate particle tracks	true
Accumulator	*String*	angle-to-moment flux calculation “SEGMENT”: during sweep (P_0 default) “SWEEP”: after sweep (P_N default)	
TransportCorrection	*Bool*	true: use P1 transport correction false: do not perform correction (Only applies for P_0 calculation)	true

Examples:

```
% change the MOC ray spacing for the
% eigenvalue calculation to 0.01 cm
opt KEFF RaySpacing=0.01

%P3 scattering
opt KEFF PnOrder=3
```

3.2.7.2 option<ESSM> - embedded self-shielding

opt *ESSM*

[key_1=val_1 key_2=val_2 ... key_i=val_i ... key_N=val_N]

[: MG_1=met_1 MG_2=met_2 ... MG_i=met_i ... MG_M=met_M]

[: E_1 E_2 ... E_i ... E_K]

key	*value type*	*details*	*default*
WithinGroupSolver	*String*	solver method "BICGSTAB": Bi-conjugate gradient stabilized "GMRES": general minimized residual "SOURCE": source iteration	"BICGSTAB"
MaxIterations	*Int*	maximum number of solver iterations	20
developer options (generally should not change)			
NumAzim	*Int*	number of azimuthal angles	8
NumPolar	*Int*	number of polar angles	3
RaySpacing	*Real*	spacing between MOC rays	0.02 cm
SolverTolerance	*Real*	solver tolerance	1e-4
EscapeXSConvergenceCriteria	*Real*		1e-3
ZeroEscapeXsTolerance	*Real*		1e-10
FluxMaxValue	*Real*		0.999999
LargeEscapeXs	*Real*		1e9
MinEnergy	*Real*	Full ESSM band lower energy bound in eV. The associated bounding energy group will have a lower boundary less than or equal to MinEnergy and an upper boundary greater than MinEnergy	1.25
MaxEnergy	*Real*	Full ESSM band upper energy bound in eV. The associated bounding energy group will have a lower boundary less than MaxEnergy and an upper boundary greater than or equal to MaxEnergy	50,000
PolarScheme	*String*	polar quadrature “TY”: Tabuchi-Yamamoto quadrature “DECART”: DeCART quadrature “LO”: Leonard optimal quadrature “CACTUS”: Cactus quadrature	DECART

param	type	details	default
MG_i	*String*	Material group name for ESSM group i. All materials in the model with a material ID which contains the material group name will have their ESSM escape cross-section calculations performed together. The method used to performed these calculations will be designated by met_i	ALL
met_i	*String*	The ESSM calculation method to be used by material group MG_i. Acceptable values are either "G" for full "Group-wise" treatment, or "I" for the Enhanced Neutron Current based tabular "Interpolation" approximation method.	G
E_i	*Real*	Energy values in units of eV which specify the energies to be used for tabular Interpolation. These values must be provided in ascending order.	0.1 1 10

Examples:

```
% change within group solver to use source iterations
opt ESSM WithinGroupSolver=SOURCE

% change all CLAD to use the interpolation
% self-shielding method (faster)
opt ESSM RaySpacing=0.08
         NumAzim=1
         NumPolar=1
         MinEnergy=1e3
         MaxEnergy=1e7
         : CLAD=I FUEL=G
         : 10 100 1000 10000
```

3.2.7.3 option<BOND> - Bondarenko search

opt *BOND* [key_1=val_1 key_2=val_2 … key_i=val_i … key_N=val_N]

key	value type	details	default
MaxIterations	*Int*	number of Bondarenko iterations (0 is disabled)	0*
ConvergenceXS	*String*	cross section used in Bondarenko convergence iterations "SIGA": absorption "SIGT": total	"SIGA"**
developer options (generally should not change)			
IterationCriteria	*Real*		1e-3
TempSearchMaxIterations	*Int*		20
TempSearchCriteria	*Real*		1e-8
TempSearchEqualTolerance	*Real*		1e-3

continues on next page

Table 3.2.55 – continued from previous page

key	value type	details	default
TempSearchMax-Power	*Real*		35.0
Sig0SearchMaxIterations	*Int*		20
Sig0SearchCriteria	*Real*		1e-8
Sig0SearchEqual Tolerance	*Real*		1e-3
Sig0SearchMaxPower	*Real*		35.0

* By setting the Bondarenko iterations >0, resonance interference effects may be taken into account. The default MaxIterations=0 effectively disables the Bondarenko resonance interference model.

**In SCALE, the transport cross section ("SIGT") has historically been used in Bondarenko iterations.

Examples:

```
% introduce Bondarenko iterations on total cross section
opt BOND
   MaxIterations=10
   ConvergenceXS="SIGT"
```

3.2.7.4 option<DEPL> - depletion

opt DEPL [key_1=val_1 key_2=val_2 ... key_i=val_i ... key_N=val_N]

key	value type.	details	default
TrackingSet	*String*	set of nuclides tracked in depletion calculations "None": set of nuclides present in user input "Complete": complete set of all nuclides available on ORIGEN data libraries	"Complete"
Solver	*String*	"MATREX": legacy ORIGEN solution method "CRAM": Chebyshev Rational Approximation Method	"MATREX"
Method	*String*	"PREDICTOR" "PREDICTOR_CORR ECTOR"	"PREDICTOR_CORR ECTOR"
StepRefinement	*Int*	divide the user input steps by this factor>0, i.e., refinement of 2 divides all steps by 2 (NOT ENABLED)	1
NumSubsteps	*Int*	Number of internal substeps for depletion calculations	4
DepleteMode	*String*	"BOSS" or "MOSS." The depletion power renormalization is done at the beginning of each substep (BOSS) or the middle of each substep (MOSS)	"BOSS"

continues on next page

Table 3.2.56 – continued from previous page

key	value type.	details	default
ArchiveF71	ALL|NONE	Archive all of materials on the .f71 file, or do not archive any isotope inventories	ALL
ArchiveF33	ALL |NONE|SYSTEM	Archive all of materials on the .f33 file, or no materials, or just the system average.	SYSTEM

Examples:

```
% Set the number of origen substeps per time steps to 2
% (may be useful for convergence studies)
opt DEPL
   NumSubsteps=2

% disable the addition of depletion nuclides to input materials
opt DEPL TrackingSet="NONE"

% use CRAM solver
opt DEPL Solver="CRAM"
```

3.2.7.5 option<CRITSPEC> - critical spectrum

opt *CRITSPEC* [key_1=val_1 key_2=val_2 ... key_i=val_i ... key_N=val_N]

key	value type	details	default
Mode	*String*	critical spectrum mode "SEARCH": search for critical mode (k-eff=1) "SPECIFIED": provide B2 below "NONE": do not use critical spectrum	"SEARCH"
B2	*Real*	value of critical buckling if Mode="SPECIFIED " units: cm^{-2}	0.0
Method	*String*	critical spectrum system "B1": solve the B_1 equations "P1": solve the P_1 equations	"P1"

Examples:

```
% enable critical buckling search using B1 equations for a buckling of 1e-3
opt CRITSPEC
  Mode="SPECIFIED"
  B2=1e-3
  Method="B1"
```

3.2.7.6 option<PRINT> - printing

opt *PRINT* [key_1=val_1 key_2=val_2 ... key_i=val_i ... key_N=val_N]

key	value type	details	default
XSSummary	*Bool*	print a cross section summary in the output file	yes

continues on next page

Table 3.2.58 – continued from previous page

key	value type	details	default
CritSpecSummary	*String*	print critical spectrum summary "NONE": no print out "BUCKLING": limited buckling info "SPECTRUM": full spectrum	"BUCKLING"
XFile16	*Bool*	output a TRITON xfile016 nodal data library	no
InputDataContainer InputPropertySummary	*Bool* *Bool*	print out the input data container print out the user defined properties	yes yes
InputCompositionSummary	*Bool*	print compostion card input summary	yes
InputMaterialSummary	*Bool*	print material card input summary	yes
LibrarySummary	*Bool*	print cross section library summary	no
MaterialSummary	*Bool*	print material summary at each statepoint	no
FgNodalSummary	*Bool*	print few-group nodal output summary	yes
DeplF71Summary	*Bool*	print .f71 archive summary	yes
DeplF33Summary	*Bool*	print .f33 archive summary	yes

Examples:

```
% print the xfile016
% if input file is polaris.inp, file name will be polaris.x16
opt PRINT XFile16=yes

% print summaries
opt PRINT XSSummary=yes
          CritSpecSummary="SPECTRUM"
          InputCompositionSummary=yes
          InputMaterialSummary=yes
          LibrarySummary=yes
          MaterialSummary=yes   % disabled for now
```

3.2.7.7 option<FG> - few-group cross section generation

opt *FG*

[AdjointMode=String InvVelMode=String DetectorEdit=DNAME]

[: b_1 b_2 … b_i … b_N]

[: E_1 E_2 … E_i … E_{N-1}]

param	type	details	default
AdjointMode	*String*	type of adjoint calculation to use in few-group data generation "INFMED": infinite medium adjoint "CRITICAL": critical spectrum adjoint "UNIFORM": uniform adjoint	"INFMED"
InvVelMode	*String*	weighting option for few-group inverse velocities "FORWARD": forward flux weighting "ADJOINT": adjoint flux weighting	"FORWARD"
DetectorEdit	*DNAME*	Name of the detector to use in XFile16 detector edits (see **detector** card)	*none*
b_i	*Real*	list of burnups to include in output few-group cross section database, e.g., XFile16 output units: GWd/MTHM	all burnups available
E_i	*Real*	note descending order and only N-1 divisions are needed for an N group structure E_0 is maximum energy (typically 2e7 eV) E_N is minimum (typically 1e-5 eV) units: eV	0.625 eV division (two groups)

Examples:

```
% enable a detector edit to the XFile16 based on detector D1
opt FG : DetectorEdit=D1

% enable the critical spectrum adjoint
opt FG AdjointMode="CRITICAL"

%only include 0,10,15,20 GWd/MTHM burnups in few-group outputs, including XFile16
opt FG : 0 10 15 20

%redefine group energy divisions for 3 groups with divisions at 10 and 0.625 eV
opt FG : : 10 0.625
```

3.2.7.8 option<RUN> - run time

opt *RUN* [key_1=val_1 key_2=val_2 ... key_i=val_i ... key_N=val_N]

key	value type	details	default
CheckOnly	*Bool*	check input and terminate	true
HomogenizeGrains	*Bool*	homogenize grains (DISABLED)	false

Examples:

```
% check input
opt RUN CheckOnly=true
```

3.2.7.9 option<GEOM> - geometry options

opt GEOM [key_1=val_1 key_2=val_2 ... key_i=val_i ... key_N=val_N]

key	value type	details	default
NumPlotRays	*Int*	Number of rays used to generate the geometry PNG file	1500
MeshMult	*Real*	Global mesh multiplier	1.0
MeshNumRings	*Int*	Default number of radial rings for circular pin zones	1
MeshNumSectors	*Int*	Default number of radial sectors for circular pin zones	1
MeshNumX	*Int*	Default number of Cartesian x subdivisions for square pin zone	1
MeshNumY	*Int*	Default number of Cartesian y subdivisions for square pin zone	1
LegacyChannelMe sh	*Bool*	Use legacy radial mesh approach in the pin outermost circular zone	false

Table 3.2.62: Developer options controlling geometry

AreaTol	*Real*		1e-6
RelTol	*Real*		5e-8
AbsThresh	*Real*		5e-10
PlaneHitTol	*Real*		1e-13
PlaneParallelTol	*Real*		1e-13
PolyAcceptTol	*Real*		5e-10
PolyHitTol	*Real*		1e-9
PolyPointTol	*Real*		1e-12
UnitPointTol	*Real*		0.0
UnitLookAheadPointTol	*Real*		1e-9

Examples:

```
% Make the png file smaller
opt GEOM NumPlotRays=1000

% Double the mesh everywhere
opt GEOM MeshMult=2.0

% Default the ring mesh to 3
opt Geom NumMeshRings=3
```

See also: mesh

3.2.7.10 option<GAMMA> - gamma transport calculation

opt *GAMMA* [key_1=val_1 key_2=val_2 ... key_i=val_i ... key_N=val_N]

key	*value type*	*details*	*default*
NumAzim	*Int*	number of azimuthal angles per octant	20
NumPolar	*Int*	number of polar angles per octant	3
RaySpacing	*Real*	spacing between MoC rays	0.04 cm
PnOrder	*Int*	scattering order	2
developer options (generally should not change)			
Solver	*String*	solver method "BICGSTAB": Bi-conjugate gradient stabilized "GMRES": general minimized residual "SOURCE": source iteration	"BICGSTAB"
SuperGroup	*Int*	number of energy groups in the inner iteration in the downscatter calculation	1
MaxIterations	*Int*	maximum number of solver iterations	20
Tolerance	*Real*	solver tolerance	3e-7
ReverseRay	*Bool*	true – store attenuation coefficients from forward ray calculation and reuse in reverse ray calculation false – recompute attenuation coefficients for reverse ray	true
CyclicRay	*Bool*	true – continue particle tracks after reaching the boundary false – separate particle tracks	true
Accumulator	*String*	angle-to-moment flux calculation “SEGMENT”: during sweep “SWEEP”: after sweep	“SWEEP”
Verbosity	*String*	msg file print level for multigroup iterations “NONE”, “LOW”, “HIGH”	“NONE”
PolarScheme	*String*	polar quadrature “TY”: Tabuchi-Yamamoto quadrature “DECART”: DeCART quadrature “LO”: Leonard optimal quadrature “CACTUS”: Cactus quadrature	“TY”

Examples:

```
% change the solver method to source iteration
opt GAMMA
  Solver="SOURCE"
```

3.2.7.11 option<DATA> – data libraries

opt *DATA* [key_1=val_1 key_2=val_2 ... key_i=val_i ... key_N=val_N]

key	value type	details	default
YieldResource	*String*	Fission yield data	yield.data
ReducedYield-Resource	*String*	Reduced fission yield data	lwr_yields.h5
DecayResource	*String*	ORIGEN decay data	de-cay.data
NuclideResource	*String*	Nuclide mass and abundance data	nu-clide.data
KineticsResource	*String*	Kinetics data	kinet-ics.h5
ReactionResource	*String*	JEFF reaction data for ORIGEN	reac-tion.data
TransitionDefini-tion	*String*	Master ORIGEN transition data	transi-tion.def

Examples:

```
% change the data resources to different values
% Can be aliases from FileNameAliases.txt or filenames
opt DATA
  KineticsResource="kinetics_Tuttle.h5"
  ReactionResource="n302.reaction.data"
```

3.2.8 SYSTEM

The **system** cards provide a way to initialize a set of defaults to simplify input and add robustness for a well-known and well-characterized system.

system *STYPE*

argument	type	name
STYPE		system type
	PWR	pressurized water reactor
	BWR	boiling water reac-tor

The **system** card performs the following actions:

1. defines a set of materials and properties, imposing standard names for the materials and properties;
2. warns user of potential mistakes; and

3. uses heuristics to modify *unspecified* mesh and solver options for robust results.

3.2.8.1 system<PWR> - pressurized water reactor

sys PWR

Definitions

geometry

type	*purpose*	*introduced*
channel	channel material	channel COOL

properties

PNAME	*purpose*	*introduced*
boron	soluble boron in ppm	prop boron COOL : SOLP B prop boron MOD : SOLP B
cr2o3	chromia as fuel dopant in ppm	prop cr2o3 FUEL : DOPANT Cr2O3
al2o3	alumina as fuel dopant in ppm	prop al2o3 FUEL : DOPANT Al2O3
beo	beryllia as fuel dopant in ppm	prop beo FUEL : DOPANT BeO

materials

MCLASS	*purpose*	*introduced*
COOL	light water in channel	mat COOL.1 : WATER dens=0.743 : boron=0 mat COOL.2 (same)
MOD	light water elsewhere (guide/instrument tubes)	mat MOD.1 : WATER dens=0.743 : boron=0 mat MOD.2 (same)
FUEL	fuel	*no default members*
CLAD	fuel cladding	mat CLAD.1 : ZIRC4
GRID	grid spacers (Zircaloy-4 and Inconel)	mat GRID.1 : ZIRC4 mat GRID.2 : INC718
TUBE	guide/instrument tubes	mat TUBE.1 : ZIRC4
STRUCT	structural material	mat STRUCT.1 : SS304
GAP	fuel/clad gap fill gas	mat GAP.1 : FILLGAS
CNTL	control elements	mat CNTL.1 : AIC mat CNTL.2 : B4C
BP	burnable poison	*no default members*

miscellaneous

type	*purpose*	*introduced*
deplete	depletion materials	deplete ALL=N FUEL=R BP=R
shield	material self-shielding	shield ALL=N FUEL=R BP=R CNTL=R

Description

Twelve reactor materials are initialized with compositions and densities from the predefined composition set. In most cases, all that remains is to define fuel materials, all material temperatures, and properties such as COOL/MOD soluble boron and density. Note that some rules are based on naming conventions. For example, burnable poisons (the material class BP) are declared to be depletable materials, whereas the CNTL (control elements) class of materials is not.

Examples:

```
% a complete input file for a PWR pincell model
=polaris
system PWR
geom MyPin : ASSM 1 1.5
comp f35 : UOX 3.5
mat FUEL.1 : f35 dens=10.25
pin 1 : 0.5 0.6 : FUEL CLAD
state ALL : temp=565
state MOD : dens=0.743
state COOL: dens=0.743
state ALL : boron=600
power 40
burn 0 0.1 0.2 0.5 1 2 5 10 15
     20 25 30 35 40 45 50 55 60
end
```

3.2.8.2 system<BWR> - boiling water reactor

sys BWR

Definitions

geometry

type	*purpose*	*introduced*
channel	channel material	channel COOL

properties

PNAME	*purpose*	*introduced*
boron	soluble boron in ppm	prop boron COOL : SOLP B prop boron MOD : SOLP B
cr2o3	chromia as fuel dopant in ppm	prop cr2o3 FUEL : DOPANT Cr2O3
al2o3	alumina as fuel dopant in ppm	prop al2o3 FUEL : DOPANT Al2O3
beo	beryllia as fuel dopant in ppm	prop beo FUEL : DOPANT BeO
void	void fraction in percent	prop void COOL : TWOPHASE 0.743 0.0353 prop void MOD : TWOPHASE 0.743 0.0353

materials

MCLASS	*purpose*	*introduced*
COOL	light water in channel	mat COOL.1 : WATER : boron=0 void=40 mat COOL.2 (same)
MOD	light water elsewhere (water rods/bypass)	mat MOD.1 : WATER : boron=0 void=0 mat MOD.2 (same)
FUEL	fuel	*no default members*
CLAD	fuel cladding	mat CLAD.1 : ZIRC2
TUBE	guide/instrument tubes	mat TUBE.1 : ZIRC2
STRUCT	structural material	mat STRUCT.1 : SS304
GAP	fuel/clad gap fill gas	mat GAP.1 : FILLGAS
CNTL	control elements	mat CNTL.1 : AIC mat CNTL.2 : B4C
CAN	channel box material	mat CAN.1 : ZIRC2

miscellaneous

type	*purpose*	*introduced*
deplete	depletion materials	deplete ALL=N FUEL=R
shield	material self-shielding	shield ALL=N FUEL=R CNTL=R

3.2.9 SAMPLE PROBLEMS

Within the SCALE distribution, 31 Polaris sample problems are provided to demonstrate the differences in calculation and geometry options, and 21 sample problems consider the Consortium for Advanced Simulation of Light Water Reactors (CASL) Virtual Environment for Reactor Applications (VERA) benchmark problems for pin cell and lattice configurations described in [POLARISRE13]. The VERA pin cell problems are identified as `polaris_1a.inp` through `polaris_1e.inp`. The VERA lattice problems are identified as `polaris_2a.inp` through `polaris_2p.inp`.

The remaining ten sample problems are described as follows:

- polaris_TMI1_Cycle1-2.inp – 15 × 15 PWR geometry model with branch block definition for lattice physics calculations
- polaris_bench_taka3_sf97-4_assm.inp, polaris_bench_taka3_sf97-4_pin.inp – Takahama UOX depletion benchmark for radiochemical assay NT3G24-SF97-4 described in [POLARISBS96].
- polaris.6.3_rca_pwr.inp – Additional Takahama UOX depletion benchmark with time-dependent boron concentraion defined in a **history** block.
- polaris_bwr10x10.inp, polaris_bwr7×7.inp – example BWR geometry models for 10 × 10 and 7 x 7 fuel.
- polaris_dv1a.inp – simple PWR pin cell depletion calculation.
- polaris_twophase.inp – simple BWR 8 x 8 fuel example to demonstrate the TWOPHASE property.
- polaris_dopants.inp – pin cell demonstrations of built-in dopant properties for LWR fuel.
- polarisHatchT1ASimple.inp - BWR model with detector model in narrow-narrow corner.

3.2.10 APPENDICES

3.2.10.1 SCALE 6.2 Polaris Input Format

For the release of SCALE 6.2.3, several new input cards were implemented into Polaris to model boiling water reactor (BWR) geometries and neutron/gamma detectors, which requires a gamma transport calculation. Moreover, improvements to existing input cards were implemented, along with the ability to specify time-dependent state properties and the ability to specify one or more depletion histories. This appendix documents the SCALE 6.2 input cards that were included in the Polaris input format which are accessible as part of the release of SCALE 6.3.

The old input cards are not available *by default* with SCALE 6.3, and should be considered deprecated. To maximize backwards compatibility for input files developed with the original SCALE 6.2.0 release, the old input options are available if the input file begins with =polaris_6.2.

The following input cards were introduced in SCALE 6.3 and are not available by default in SCALE 6.2:

- **cross** – define the interior water cross geometry of SVEA assembly designs;
- **dxmap** (or **dymap**) – define displacement maps that indicate that translation of the pin center in the x- (or y-) direction;
- **control <BLADE>** – define the control blade geometry;
- **mesh** – define advanced spatial meshing options for different materials; and

- **option <GEOM>** – define geometry tolerances, advance meshing options, and plotting options.
- **option <GAMMA>** – gamma transport calculation
- **detector** – insert a detector geometry
- **history** – time dependent history
- **bui [ti]** – initiate calculation with cumulative burnups [time] (with restart)
- **property<GRAIN>** – used to model stochastic media
- **state** – the generalized state input is preceded by the state<MNAME>, state<INAME>, state<GNAME> variants

box – channel box

box [thick=*Real*] [rad=*Real*] [icdist=*Real*] [xrad=*Real*] [xlen=*Real*] [Mbox=MNAME]

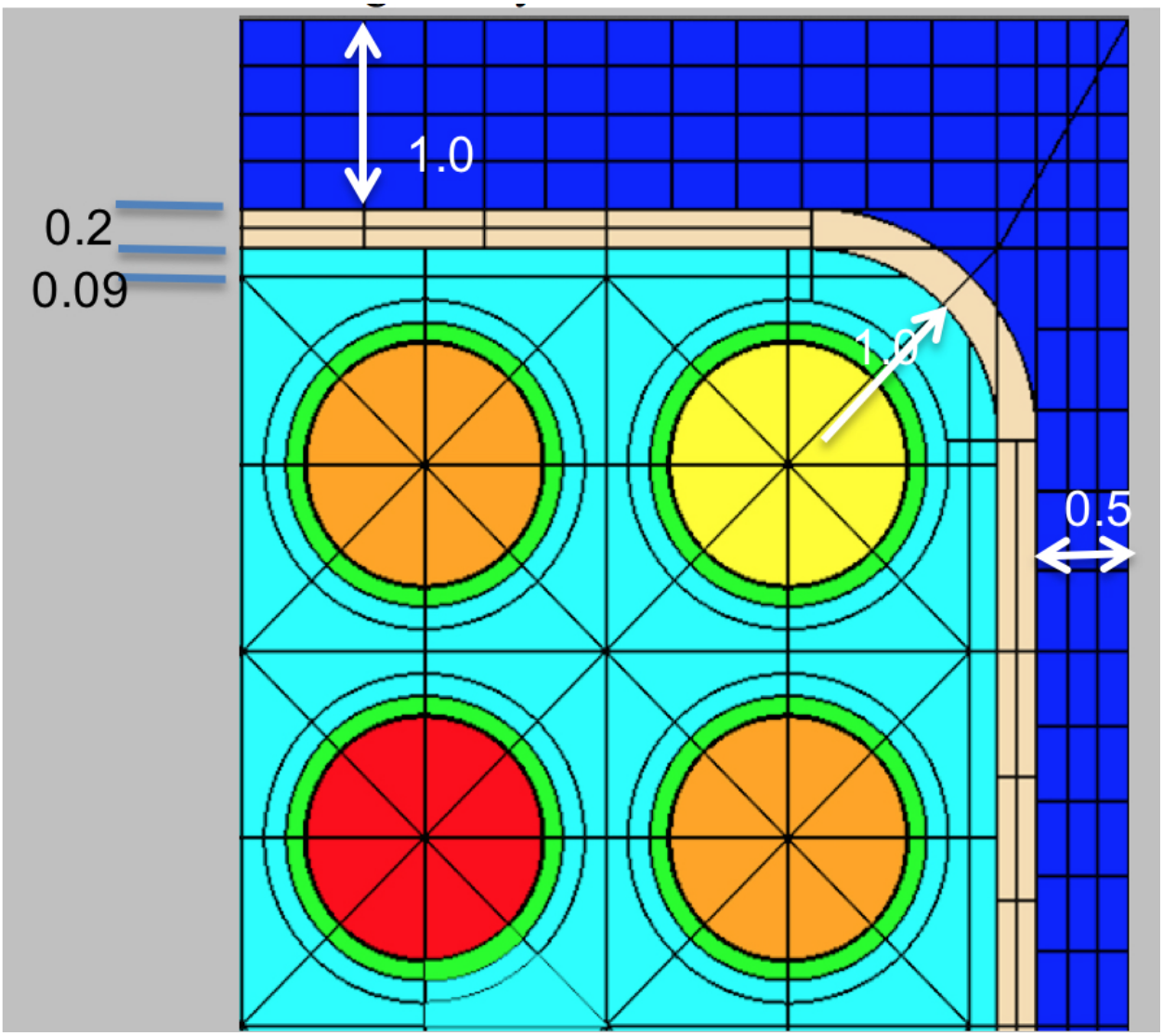
param	**type**	**name**	**details**	**default**
thick	*Real*	nominal thickness (cm)	must be >0	none
rad	*Real*	inner corner radius (cm)	must be >0, additional constraints listed below	none
icdist	*Real*	in-channel distance (cm)	must be >0	none
xrad	*Real*	extra corner thickness (cm)	must be >= 0	0
xlen	*Real*	extra corner length (cm)	excludes rounded corner length	0
M_{box}	MNAME	box material		*

*By default, **box** material will be set to CAN.1 by "system BWR." Otherwise **box** material is required.

```
% GE 7x7 assembly with 1.88 cm pin pitch
%   0.48 cm narrow gap
%   0.95 cm wide gap
system BWR
geom ge7x7 : ASSM 7 1.88
hgap 0.48 0.95

% Box geometry
%   0.2 thickness
%   0.97 inner corner radius
%   0.14 in-channel distance
box 0.2 0.97 0.14

% Same example, all variables
box 0.2 0.97 0.14 0 0 CAN.1
```

Comments:

The **box** specifies the channel box geometry and material that surround the array of fuel pins.



See also:

hgap

```
geom ge7x7 : ASSM 7 1.88
hgap 0.5 1.0
   : MOD.1 MOD.1
   : 2     2
   : 3     4
box 0.2 1.0 0.09
```

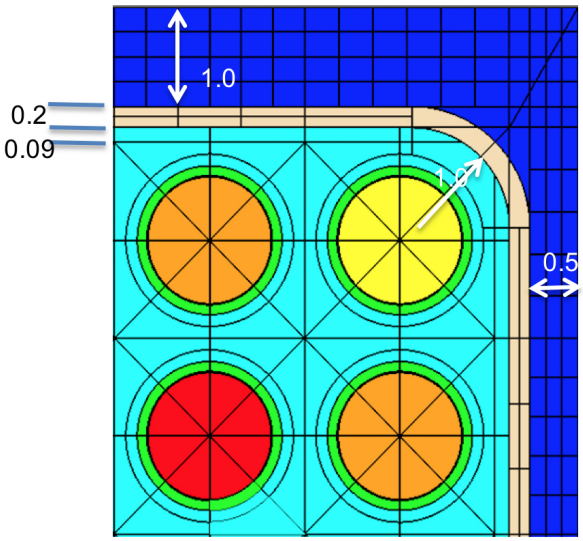


Fig. 3.2.8: Box geometry example (uniform thickness).

```
geom bwr10x10 : ASSM 10 1.3
hgap 0.5 1.0 : 2 2 : 3 4
box thick=0.2 rad=0.9
     icdist=0.2 xrad=0.1
     xlen=3.5
```

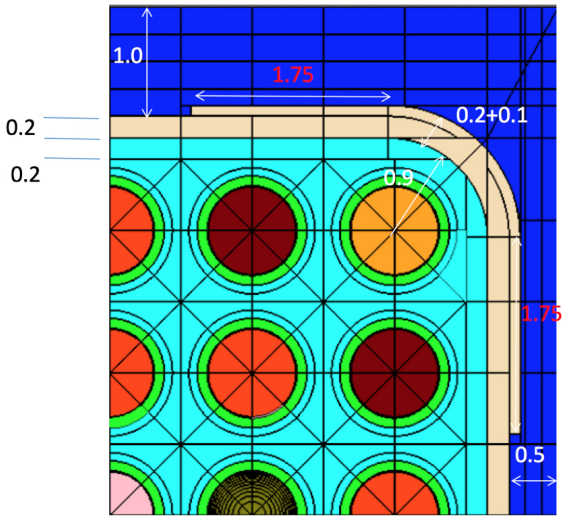


Fig. 3.2.9: Box geometry example (thick corners).

pin – pin or pincell

pin PINID [nsect=*Int*] [nring=*Int*] [size=*Int*]

: r_1 r_2 … r_i … r_N

: M_1 M_2 … M_i … M_N [M_{out}]

[: nr_1 nr_2 … nr_i … nr_N nr_{out}]

[: ns_1 ns_2 … ns_i … ns_N ns_{out}]

param	*type*	*name*	*details*	*default*
PINID	*Word|Int*	pin identifier		
size	*Int*	pin will be placed in a size x size pincell grid	used to create large water rods in CE PWRs and GE BWRs (see pinmap)	
r_i	*Real*	list of radii	for each radial zone from center out units: cm	
M_i	*MNAME*	list of pin materials	material in each radial zone .1 added if given MCLASS, e.g., FUEL→FUEL.1	
M_{out}	*MNAME*	material in outermost zone	material in outermost zone .1 added if given MCLASS, e.g., FUEL→FUEL.1	*
meshing options				
nsect	*Int*	number of sectors	azimuthal sections of the pin cell, value applies to all ns_i and ns_{out}	
nring	*Int*	number of rings	value applies to all nr_i and nr_{out}	
nr_i	*Int*	number of rings	number of rings in each radial zone	1
nr_{out}	*Int*	number of rings in outermost zone	number of rings in the outermost zone	1
ns_i	*Int*	number of sectors	number of sectors in each radial zone	1
ns_{out}	*Int*	number of sectors in outermost zone	number of sectors in the outermost zone	1

*If not specified, the material class MCLASS is taken from the **channel** card (M_{chan}) and set to the first member of that class, "M_{chan}.1." For example if M_{chan}="COOL," then M_{out}= "COOL.1."

Examples:

```
%standard fuel pin
pin 1 : 0.4096 0.418 0.475
      : FUEL   GAS   CLAD

%empty guide tube
pin E :        0.561 0.602
      :       COOL.1 CLAD.1

%pyrex
pin P : 0.214 0.231 0.241 0.427 0.437 0.484 0.561 0.602
      : GAS   TUBE  GAS   BP.3  GAS   TUBE  COOL  CLAD

%standard fuel pin with explicit material in the outermost region
pin 1 : 0.4096 0.418 0.475
      : FUEL   GAS   CLAD   COOL.6


%standard fuel pin with explicit ring and sector mesh
pin 1 : 0.4096 0.418 0.475
      : FUEL   GAS   CLAD
      : 5      1     1      0   %5 rings in fuel
      : 8      1     1      1   %8 sectors only in fuel
```

(continues on next page)

(continued from previous page)

```
%large central 2x2 water rod in 4x4 assembly
pin W size=2 : 0.8
             : COOL

%pinmap must show adjacent Ws
pinmap
F F F F
F W W F
F W W F
F F F F
```

Comments:

The **pin** card is one of the basic building blocks of the assembly model. It is the only geometry component which allows an integer (*Int*) identifier as well as a *Word*-all other geometric identifiers use *Word*. Note that the materials are required, except for the last M_{out}, which can be used to overwrite the material given by a **channel** for the outermost region in the pincell. The various pin cell meshing options are displayed in Fig. 3.2.10. Note that extra rings in the radial zones create equal area regions, whereas rings in the outermost region create equal distance divisions between the last radius and the pincell boundary. Extra sectors create additional azimuthal divisions. A negative value of sectors is allowed and can be used to rotate the sector mesh by a half angle, e.g., ns=4 looks like ⊗ and ns=-4 looks like ⊗.

The total number of cells used in the transport calculation is determined from both the number of rings and the number of sectors. With the MoC transport solver, the fidelity of the solution is also dictated by the number of azimuthal and polar angles and ray spacing. These parameters are changed on the **option<KEFF>** card.

Due to self-shielding and depletion, each cell could be modeled as a unique material with its own cross section data. However, this is prohibitively memory intensive and typically not necessary. The **shield** card provides the mechanism to control the additional self-shielded materials introduced.

See also:

pinmap, control, insert, channel, system, option<KEFF>, shield

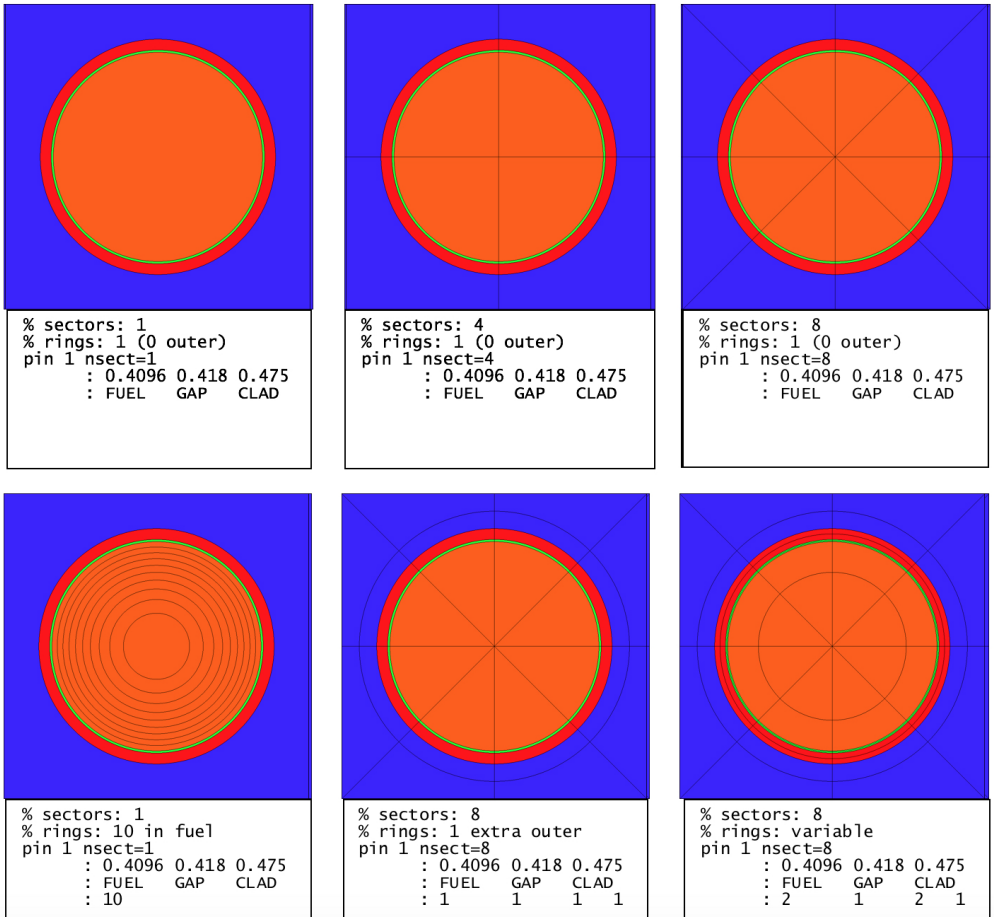


Fig. 3.2.10: Pincell meshing variants.

bu – initiate calculation with cumulative burnups

bu [units=<*GWD/MTIHM*>|*MWD/MTIHM*] : [b_1 b_2 … b_i … b_N]

param	Type	name	details	default
units	GWD/MTIHM| MWD/MTIHM	burnup units		GWD/MTIHM
b_i	*Real*	list of absolute burnups		0

Examples:

```
% simple depletion case with constant power and absolute/cumulative burnups
power 40
bu 0 5 10 15 20 30 40 50 60 80

% using MWd/MTIHM units with variable power
% 40 W/gIHM for 0→5000 MWD/MTIHM, then 30 W/gIHM for 5000→10000 MWD/MTIHM
power  40    30
bu MWD/MTIHM: 0 5000 10000
```

(continues on next page)

(continued from previous page)

```
% combine burn/time cards
% 20 W/gIHM for 0→5 then 5→10 GWD/MTIHM steps, then
% 40 W/gIHM for a 5-day step then 30 W/gIHM for a 5-day step
power 20
bu   GWD/MTIHM :  5 10 GWD/MTIHM
power 40 30
dt  DAYS :   5  5
```

Comments:

The **bu** card initiates a calculation for a given sequence of cumulative burnups. A burnup or time card usually follows a **power** card, the two effectively specifying the power history. If multiple burnups are given, then the **power** card must have either a single power or a list of powers the same size as the list times. A value of 0 is implicit at the beginning of the first burnup list. Multiple burnup/time cards may be specified in an input. This can be convenient for switching units or changing from burnup-based to time-based depletion. Internal automatic substeps are always in effect unless modified with the **option<DEPL>** card.

See also:

t, dt, dbu, power, option<DEPL>, branch, deplete

dbu – initiate calculation with incremental burnups

dbu [units=<*GWD/MTIHM*>|*MWD/MTIHM*] : [b_1 b_2 … b_i … b_N]

param	Type	name	details	default
units	GWD/MTIHM| MWD/MTIHM	burnup units		GWD/MTIHM
b_i	*Real*	list of incremental burnups		0

Examples:

```
% incremental burnups equivalent to
%   power 40
%   bu 0 5 10 15 20 30 40 50 60 80
power 40
dbu 5 5 5 5 10 10 10 10 20
```

Comments:

The **dbu** card initiates a calculation for a given sequence of *incremental* burnups. Otherwise, it is identical to the **bu** card for specifying cumulative burnups.

See also:

t, dt, bu, power, option<DEPL>, branch, deplete

t – initiate calculation by cumulative time

t [units=*SECONDS*|*MINUTES*|*HOURS*|<*DAYS*>|*YEARS*] : [t_1 t_2 … t_i … t_N]

param	Type	name	details	default
units	*SECONDS*| *MINUTES*| *HOURS*| <*DAYS*>| *YEARS*	time units		DAYS
t_i	*Real*	list of times	0	0

Examples:

```
% burn with 40 W/gIHM for 300 days in 100-day increments
power 40
t 100 200 300

% simulate 2 cycles of time-dependent irradiation with shutdown cooling
% note that time defaults to DAYs
%
% cycle 1
power 40   30   30   30
t    100  200  300  400
power  0
t    415
%
% cycle 2
power 30   20   20   20
t    515  615  715  815
power  0
t    830
```

Comments:

The **t** card initiates a calculation for a given sequence of cumulative/absolute times. One of the time cards (**t**, **dt**, or **ti**) is required to model periods of decay in conjunction with **power** 0. Otherwise, the time card **t** is similar in functionality to the burnup **bu** card but with different units.

See also:

dt, bu, dbu, power, option<DEPL>, branch, deplete

dt – initiate calculation by incremental time

dt [units=*SECONDS*|*MINUTES*|*HOURS*|<*DAYS*>|*YEARS*] : [t_1 t_2 … t_i … t_N]

param	Type	name	details	default
units	*SECONDS*| *MINUTES*| *HOURS*| <*DAYS*>| *YEARS*	time units		DAYS
t_i	*Real*	list of times	0	0

Examples:

```
% burn with 40 W/gIHM for 300 days in 100-day increments equivalent to
%     power 40
%     t 100 200 300
power 40
dt 100 100 100

% decay for 30 minutes
power 0
dt 30 MINUTES
```

Comments:

The **dt** card is identical to the cumulative time card **t** except that the values given are incremental.

See also:

t, bu, dbu, power, option<DEPL>, branch, deplete

option<ESSM> - embedded self-shielding

opt *ESSM*

[key_1=val_1 key_2=val_2 … key_i=val_i … key_N=val_N]

key	*value type*	*details*	*default*
WithinGroupSolver	*String*	solver method "BICGSTAB": Bi-conjugate gradient stabilized "GMRES": general minimized residual "SOURCE": source iteration	"BICGSTAB"
MaxIterations	*Int*	maximum number of solver iterations	20

developer options (generally should not change)

NumAzim	*Int*	number of azimuthal angles	8
NumPolar	*Int*	number of polar angles	3
RaySpacing	*Real*	spacing between MOC rays	0.02 cm
SolverTolerance	*Real*	solver tolerance	1e-4
EscapeXSConvergenceCriteria	*Real*		1e-3
ZeroEscapeXsTolerance	*Real*		1e-10
FluxMaxValue	*Real*		0.999999
LargeEscapeXs	*Real*		1e9
MinEnergy	*Real*	Full ESSM band lower energy bound in eV. The associated bounding energy group will have a lower boundary less than or equal to MinEnergy and an upper boundary greater than MinEnergy	1.25
MaxEnergy	*Real*	Full ESSM band upper energy bound in eV. The associated bounding energy group will have a lower boundary less than MaxEnergy and an upper boundary greater than or equal to MaxEnergy	50,000
PolarScheme	*String*	polar quadrature "TY": Tabuchi-Yamamoto quadrature "DECART": DeCART quadrature "LO": Leonard optimal quadrature "CACTUS": Cactus quadrature	DECART

Examples:

```
% change within group solver to use source iterations
opt ESSM WithinGroupSolver=SOURCE
```

option<FG> – few-group cross section generation

opt *FG*

[AdjointMode=String InvVelMode=String]

[: b_1 b_2 … b_i … b_N]

[: E_1 E_2 … E_i … E_{N-1}]

param	**type**	**details**	**default**
Adjoint-Mode	*String*	type of adjoint calculation to use in few-group data generation "INFMED": infinite medium adjoint "CRITICAL": critical spectrum adjoint "UNIFORM": uniform adjoint	"INFMED"
InvVelMode	*String*	weighting option for few-group inverse velocities "FORWARD": forward flux weighting "ADJOINT": adjoint flux weighting	"FORWARD"
b_i	*Real*	list of burnups to include in output few-group cross section database, e.g., XFile16 output units: GWd/MTHM	all burnups available
E_i	*Real*	note descending order and only N-1 divisions are needed for an N group structure E_0 is maximum energy (typically 2e7 eV) E_N is minimum (typically 1e-5 eV) units: eV	0.625 eV division (two groups)

Examples:

```
% enable the critical spectrum adjoint
opt FG AdjointMode="CRITICAL"

%only include 0,10,15,20 GWd/MTHM burnups in few-group outputs, including XFile16
opt FG : 0 10 15 20

%redefine group energy divisions for 3 groups with divisions at 10 and 0.625 eV
opt FG : : 10 0.625
```

state<MNAME> – material state

state MNAME|MCLASS :

p_1=val_1 p_2=val_2 … p_i=val_i … p_N=val_N

param	type	name	details	default
MNAME|MCLASS	-	material name or material class	use ALL for all materials	
p_i	PNAME	property name		
val_i	*Value*	property value		

Examples:

```
% reset to hot zero power conditions
state ALL  : temp=565
state COOL : dens=0.75


% set channel/bypass materials to different ppm boron
state COOL : boron=0
state BYP  : boron=600

% set all materials with a boron property
state ALL : boron=300
```

Comments:

The **state** card declares the *base state* for materials and the *base state* of possible control elements or insert elements.

See also: material, deplete, system

state<INAME> – insert/control state

state INAME : in=*Bool*

param	type	name	details	default
INAME	-	insert name or control element name		
in	Bool	insertion	"in=" is required	

Examples:

```
% insert bank D control rods
state BankD : in=true
% remove inserts named Ins6A
state Ins6A false
% perform reflector calculation
state ReflectorNode : in=true
```

Comments:

This form of the **state** card is required to insert any control element or inserts. By default, inserts and control elements are out when defined.

See also:

material, deplete, system

state<GNAME> – geometry state

state GNAME : pres=*Bool*

param	type	name	details	default
GNAME	-	geometry name		
pres	Bool	present	"pres=" is required	

Examples:

```
% disable reflector calculation even though
% reflector geometry is present
state ReflectorNode : pres=false
```

Comments:

The geometry version of the **state** card is used to declare which geometric elements are present in the system. The geometry version of the **add** card is useful for performing branch calculations for reflector nodes. Note that an assembly geometry must be present to perform a calculation.

See also:

geometry<REFL>, state<MNAME>, state<INAME>

4. RADIATION SHIELDING

Introduction by D. E. Peplow and B. T. Rearden

The Monaco with Automated Variance Reduction using Importance Calculations (MAVRIC) fixed-source radiation transport sequence is designed to apply the multigroup and continuous energy fixed-source Monte Carlo code Monaco to solve problems too challenging for standard, unbiased Monte Carlo methods.i The intention of the sequence is to calculate fluxes and dose rates with low uncertainties in reasonable times even for deep penetration problems. MAVRIC is based on the Consistent Adjoint Driven Importance Sampling (CADIS) methodology, which uses an importance map and biased source that are derived to work together. MAVRIC generates problem-dependent cross section data and then automatically performs a coarse mesh 3D discrete ordinates transport calculation using Denovo to determine the adjoint flux as a function of position and energy. MAVRIC then applies the information to optimize the shielding calculation in Monaco. In the Forward-Weighted CADIS (FW-CADIS) methodology, an additional Denovo calculation is performed to further optimize the Monaco model to obtain uniform uncertainties for multiple tally locations. Sources can be input using many spatial and energy distributions, or they can be imported from a previously generated mesh file. Responses can be generated from cross section reaction rates, flux-to-dose conversion factors, or user-defined functions; responses are tallied on geometry regions, Cartesian or cylindrical grids, or as point detectors.

MAVRIC enables users to import spent fuel isotopics from ORIGEN binary concentrations files such as those generated with ORIGAMI and to import gamma emission spectral data directly from the ORIGEN nuclear data files to streamline analysis and reduce the opportunity for human error.

Several utility modules are also provided for data introspection and conversion.

4.1 MAVRIC: MONACO WITH AUTOMATED VARIANCE REDUCTION USING IMPORTANCE CALCULATIONS

D. E. Peplow and C. Celik

4.1.1 INTRODUCTION

Monte Carlo particle transport calculations for deep penetration problems can require very long run times in order to achieve an acceptable level of statistical uncertainty in the final answers. Discrete-ordinates codes can be faster but have limitations relative to the discretization of space, energy, and direction. Monte Carlo calculations can be modified (biased) to produce results with the same variance in less time if an approximate answer or some other additional information is already known about the problem. If importances can be assigned to different particles based on how much they will contribute to the final answer, then more time can be spent on important particles, with less time devoted to unimportant particles. One of the best ways to bias a Monte Carlo code for a particular tally is to form an importance map from the adjoint flux based on that tally. Unfortunately, determining the exact adjoint flux could be just as difficult as computing the original problem itself. However, an approximate adjoint can still be very useful in biasing the Monte Carlo solution [MAVRIC-Wag97]. Discrete ordinates can be used to quickly compute that approximate adjoint. Together, Monte Carlo and discrete ordinates can be used to find solutions to thick shielding problems in reasonable times.

The MAVRIC (Monaco with Automated Variance Reduction using Importance Calculations) sequence is based on the CADIS (Consistent Adjoint Driven Importance Sampling) and FW-CADIS (Forward-Weighted CADIS) methodologies [MAVRIC-HW03, MAVRIC-Wag02, MAVRIC-WBP07, MAVRIC-WH98,

MAVRIC-WPE20]. MAVRIC automatically performs a three-dimensional, discrete-ordinates calculation using Denovo to compute the adjoint flux as a function of position and energy. This adjoint flux information is then used to construct an importance map (i.e., target weights for weight windows) and a biased source distribution that work together-particles are born with a weight matching the target weight of the cell into which they are born. The fixed-source Monte Carlo radiation transport Monaco [MAVRIC-Pep11] then uses the importance map for biasing during particle transport, and it uses the biased source distribution as its source. During transport, the particle weight is compared with the importance map after each particle interaction and whenever a particle crosses into a new importance cell in the map.

For problems that do not require variance reduction to complete in a reasonable time, execution of MAVRIC without the importance map calculation provides an easy way to run Monaco. For problems that do require variance reduction to complete in a reasonable time, MAVRIC removes the burden of setting weight windows from the user and performs it automatically with a minimal amount of additional input. Note that the MAVRIC sequence can be used with the final Monaco calculation as either a multigroup (MG) or a continuous-energy (CE) calculation.

Monaco has a wide variety of tally options: it can calculate fluxes (by group) at a point in space, over any geometrical region, or for a user-defined, three-dimensional, rectangular grid. These tallies can also integrate the fluxes with either standard response functions from the cross section library or user-defined response functions. All of these tallies are available in the MAVRIC sequence.

Although it was originally designed for CADIS, the MAVRIC sequence is also capable of creating importance maps using both forward and adjoint deterministic estimates. The FW-CADIS method [MAVRIC-WPM14] can be used for optimizing several tallies at once, a mesh tally over a large region, or a mesh tally over the entire problem. Several other methods for producing importance maps are also available in MAVRIC and are explored in Sect. 4.4.

4.1.2 CADIS METHODOLOGY

MAVRIC is an implementation of CADIS (Consistent Adjoint Driven Importance Sampling) using the Denovo SN and Monaco Monte Carlo functional modules. Source biasing and a mesh-based importance map, overlaying the physical geometry, are the basic methods of variance reduction. To make the best use of an importance map, the map must be made consistent with the source biasing. If the source biasing is inconsistent with the weight windows that will be used during the transport process, then source particles will undergo Russian roulette or splitting immediately, wasting computational time and negating the intent of the biasing.

4.1.2.1 Overview of CADIS

CADIS is well described in the literature, so only a brief overview is given here. Consider a class source-detector problem described by a unit source with emission probability distribution function $q\left(\overrightarrow{r}, E\right)$ and a detector response function $\sigma_d\left(\overrightarrow{r}, E\right)$. To determine the total detector response, R, the forward scalar flux $\phi\left(\overrightarrow{r}, E\right)$ must be known. The response is found by integrating the product of the detector response function and the flux over the detector volume V_d:

$$R = \int_{V_d} \int_E \sigma_d\left(\overrightarrow{r}, E\right) \phi\left(\overrightarrow{r}, E\right) dE\, dV. \tag{4.1.1}$$

Alternatively, if the adjoint scalar flux, $\phi^+\left(\overrightarrow{r}, E\right)$, is known from the corresponding adjoint problem with adjoint source $q^+\left(\overrightarrow{r}, E\right) = \sigma_d\left(\overrightarrow{r}, E\right)$, then the total detector response could be found by integrating the

product of the forward source and the adjoint flux over the source volume, V_s:

$$R = \int_{V_s} \int_E q\left(\vec{r}, E\right)\phi^+\left(\vec{r}, E\right) dE\, dV. \tag{4.1.2}$$

Unfortunately, the exact adjoint flux may be just as difficult to determine as the forward flux, but an approximation of the adjoint flux can still be used to form an importance map and a biased source distribution for use in the forward Monte Carlo calculation.

Wagner [MAVRIC-Wag97] showed that if an estimate of the adjoint scalar flux for the corresponding adjoint problem can be found, then an estimate of the response R can be made using Eq. (4.1.2). The adjoint source for the adjoint problem is typically separable and corresponds to the detector response and spatial area of the tally to be optimized: $q^+\left(\vec{r}, E\right) = \sigma_d\left(E\right) g\left(\vec{r}\right)$, where $\sigma_d\left(E\right)$ is a flux-to-dose conversion factor and $g\left(\vec{r}\right)$ is 1 in the tally volume and is 0 otherwise. Then, from the adjoint flux $\phi^+\left(\vec{r}, E\right)$ and response estimate R, a biased source distribution, $\widehat{q}\left(\vec{r}, E\right)$, for source sampling of the form

$$\widehat{q}\left(\vec{r}, E\right) = \frac{1}{R} q\left(\vec{r}, E\right)\phi^+\left(\vec{r}, E\right) \tag{4.1.3}$$

and weight window target values, $\overline{w}\left(\vec{r}, E\right)$, for particle transport of the form

$$\overline{w}\left(\vec{r}, E\right) = \frac{R}{\phi^+\left(\vec{r}, E\right)} \tag{4.1.4}$$

can be constructed, which minimizes the variance in the forward Monte Carlo calculation of R.

When a particle is sampled from the biased source distribution $\widehat{q}\left(\vec{r}, E\right)$, to preserve a fair game, its initial weight is set to

$$w_0\left(\vec{r}, E\right) = \frac{q\left(\vec{r}, E\right)}{\widehat{q}\left(\vec{r}, E\right)} = \frac{R}{\phi^+\left(\vec{r}, E\right)} \tag{4.1.5}$$

which exactly matches the target weight for that particle's position and energy. This is the "consistent" part of CADIS-source particles are born with a weight matching the weight window of the region/energy into which they are born. The source biasing and the weight windows work together.

CADIS has been applied to many problems-including reactor ex-core detectors, well-logging instruments, cask shielding studies, and independent spent fuel storage facility models-and has demonstrated very significant speed-ups in calculation time compared to analog simulations.

4.1.2.2 Multiple sources with CADIS

For a typical Monte Carlo calculation with multiple sources—each with a probability distribution function $q_i\left(\vec{r}, E\right)$ and a strength S_i, giving a total source strength of $S = \sum S_i$—the source is sampled in two steps. First, the specific source i is sampled with probability $p\left(i\right) = S_i/S$, and then the particle is sampled from the specific source distribution $q_i\left(\vec{r}, E\right)$.

The source sampling can be biased at both levels: from which source to sample and how to sample each source. For example, the specific source can be sampled using some arbitrary distribution, $\widehat{p}\left(i\right)$, and then the individual sources can be sampled using distributions $\widehat{q}_i\left(\vec{r}, E\right)$. Particles would then have a birth weight of

$$w_0 \equiv \left(\frac{p\left(i\right)}{\widehat{p}\left(i\right)}\right)\left(\frac{q_i\left(\vec{r}, E\right)}{\widehat{q}_i\left(\vec{r}, E\right)}\right). \tag{4.1.6}$$

For CADIS, a biased multiple source needs to be developed so that the birth weights of sampled particles still match the target weights of the importance map. For a problem with multiple sources—each with a distribution $q_i\left(\overrightarrow{r},E\right)$ and a strength S_i—the goal of the Monte Carlo calculation is to compute some response R for a response function $\sigma_d\left(\overrightarrow{r},E\right)$ at a given detector,

$$R = \int_V \int_E \sigma_d\left(\overrightarrow{r},E\right)\phi\left(\overrightarrow{r},E\right) dE\, dV. \tag{4.1.7}$$

Note that the flux $\phi\left(\overrightarrow{r},E\right)$ has contributions from each source. The response, R_i, from each specific source (S_i with $q_i\left(\overrightarrow{r},E\right)$) can be expressed using just the flux from that source, $\phi_i\left(\overrightarrow{r},E\right)$, as

$$R_i = \int_V \int_E \sigma_d\left(\overrightarrow{r},E\right)\ \phi_i\left(\overrightarrow{r},E\right) dE\, dV\,. \tag{4.1.8}$$

The total response is then found as $R = \sum_i R_i$.

For the adjoint problem, using the adjoint source of $q^+\left(\overrightarrow{r},E\right) = \sigma_d\left(\overrightarrow{r},E\right)$, the response R can also be calculated as

$$R = \int_V \int_E \left[\sum_i S_i q_i\left(\overrightarrow{r},E\right)\right] \phi^+\left(\overrightarrow{r},E\right) dE\, dV, \tag{4.1.9}$$

with the response contribution from each specific source being

$$R_i = \int_V \int_E S_i q_i\left(\overrightarrow{r},E\right)\phi^+\left(\overrightarrow{r},E\right) dE\, dV. \tag{4.1.10}$$

The target weights $\overline{w}\left(\overrightarrow{r},E\right)$ of the importance map are found using

$$\overline{w}\left(\overrightarrow{r},E\right) = \frac{R/S}{\phi^+\left(\overrightarrow{r},E\right)}. \tag{4.1.11}$$

Each biased source $\widehat{q}_i\left(\overrightarrow{r},E\right)$ pdf is found using

$$\widehat{q}_i\left(\overrightarrow{r},E\right) = \frac{S_i}{R_i} q_i\left(\overrightarrow{r},E\right)\phi^+\left(\overrightarrow{r},E\right)\ , \tag{4.1.12}$$

and the biased distribution used to select an individual source is $\widehat{p}\,(i) = R_i/\sum R_i = R_i/R$.

When using the biased distribution used to select an individual source, $\widehat{p}\,(i)$, and the biased source distribution, $\widehat{q}_i\left(\overrightarrow{r},E\right)$, the birth weight of the sampled particle will be

$$\begin{aligned} w_0 &\equiv \left(\frac{p(i)}{\widehat{p}(i)}\right)\left(\frac{q_i\left(\overrightarrow{r},E\right)}{\widehat{q}_i\left(\overrightarrow{r},E\right)}\right) \\ &= \left(\frac{\frac{S_i}{S}}{\frac{R_i}{R}}\right)\left(\frac{q_i\left(\overrightarrow{r},E\right)}{\frac{S_i}{R_i} q_i\left(\overrightarrow{r},E\right)\phi^+\left(\overrightarrow{r},E\right)}\right) \\ &= \frac{R/S}{\phi^+\left(\overrightarrow{r},E\right)}, \end{aligned} \tag{4.1.13}$$

which matches the target weight, $\overline{w}\left(\overrightarrow{r},E\right)$.

4.1.2.3 Multiple tallies with CADIS

The CADIS methodology works quite well for classic source/detector problems. The statistical uncertainty of the tally that serves as the adjoint source is greatly reduced since the Monte Carlo transport is optimized to spend more simulation time on those particles that contribute to the tally, at the expense of tracking particles in other parts of phase space. However, more recently, Monte Carlo has been applied to problems in which multiple tallies need to all be found with low statistical uncertainties. The extension of this idea is the mesh tally-where each voxel is a tally for which the user desires low statistical uncertainties. For these problems, the user must accept a total simulation time that is controlled by the tally with the slowest convergence and simulation results where the tallies have a wide range of relative uncertainties.

The obvious way around this problem is to create a separate problem for each tally and use CADIS to optimize each. Each simulation can then be run until the tally reaches the level of acceptable uncertainty. For more than a few tallies, this approach becomes complicated and time-consuming for the user. For large mesh tallies, this approach is not reasonable.

Another approach to treat several tallies, if they are in close proximity to each other, or a mesh tally covering a small portion of the physical problem, is to use the CADIS methodology with the adjoint source near the middle of the tallies to be optimized. Since particles in the forward Monte Carlo simulation are optimized to reach the location of the adjoint source, all the tallies surrounding that adjoint source should converge quickly. This approach requires the difficult question of "how close." If the tallies are too far apart, then certain energies or regions that are needed for one tally may be of low importance for getting particles to the central adjoint source. This may under-predict the flux or dose at the tally sites far from the adjoint source.

MAVRIC has the capability to have multiple adjoint sources with this problem in mind. For several tallies that are far from each other, multiple adjoint sources could be used. In the forward Monte Carlo, particles would be drawn to one of those adjoint sources. The difficulty with this approach is that typically the tally that is closest to the true physical source converges faster than the other tallies—showing that the closest adjoint source seems to attract more particles than the others. Assigning more strength to the adjoint source further from the true physical source helps to address this issue, but finding the correct strengths so that all of the tallies converge to the same relative uncertainty in one simulation is an iterative process for the user.

4.1.2.4 Forward-weighted CADIS

To converge several tallies to the same relative uncertainty in one simulation, the adjoint source corresponding to each of those tallies must be weighted inversely by the expected tally value. To calculate the dose rate at two points—say one near a reactor and one far from a reactor—in one simulation, then the total adjoint source used to develop the weight windows and biased source must have two parts. The adjoint source far from the reactor must have more strength than the adjoint source near the reactor by a factor equal to the ratio of the expected near dose rate to the expected far dose rate.

This concept can be extended to mesh tallies, as well. Instead of using a uniform adjoint source strength over the entire mesh tally volume, each voxel of the adjoint source should be weighted inversely by the expected forward tally value for that voxel. Areas of low flux or low dose rate would have more adjoint source strength than areas of high flux or high dose rate.

An estimate of the expected tally results can be found by using a quick discrete-ordinates calculation. This leads to an extension of the CADIS method: forward-weighted CADIS (FW-CADIS). First, a forward S_N calculation is performed to estimate the expected tally results. A total adjoint source is constructed so that the adjoint source corresponding to each tally is weighted inversely by those forward tally estimates. Then the standard CADIS approach is used-an importance map (target weight windows) and a biased source are made using the adjoint flux computed from the adjoint S_N calculation.

For example, if the goal is to calculate a detector response function $\sigma_d(E)$ (such as dose rate using flux-to-dose-rate conversion factors) over a volume (defined by $g\left(\overrightarrow{r}\right)$) corresponding to mesh tally, then instead of simply using $q^+\left(\overrightarrow{r},E\right) = \sigma_d(E)\ g(\overrightarrow{r})$, the adjoint source would be

$$q^+\left(\overrightarrow{r},E\right) = \frac{\sigma_d(E)\,\mathrm{g}\left(\overrightarrow{r}\right)}{\int \sigma_d(E)\,\phi\left(\overrightarrow{r},E\right)dE}\,, \tag{4.1.14}$$

where $\phi\left(\overrightarrow{r},E\right)$ is an estimate of the forward flux, and the energy integral is over the voxel at $\overrightarrow{r}$. The adjoint source is nonzero only where the mesh tally is defined ($g\left(\overrightarrow{r}\right)$), and its strength is inversely proportional to the forward estimate of dose rate.

The relative uncertainty of a tally is controlled by two components: (1) the number of tracks contributing to the tally and (2) the shape of the distribution of scores contributing to that tally. In the Monte Carlo game, the number of simulated particles, $m\left(\overrightarrow{r},E\right)$, can be related to the true physical particle density, $n\left(\overrightarrow{r},E\right)$, by the average Monte Carlo weight of scoring particles, $\overline{w}\left(\overrightarrow{r},E\right)$, by

$$n\left(\overrightarrow{r},E\right) = \overline{w}\left(\overrightarrow{r},E\right)\mathrm{m}\left(\overrightarrow{r},E\right). \tag{4.1.15}$$

In a typical Monte Carlo calculation, tallies are made by adding some score, multiplied by the current particle weight, to an accumulator. To calculate a similar quantity related to the Monte Carlo particle density would be very close to calculating any other quantity but without including the particle weight. The goal of FW-CADIS is to make the Monte Carlo particle density, $m\left(\overrightarrow{r},E\right)$, uniform over the tally areas, so an importance map must be developed that represents the importance of achieving uniform Monte Carlo particle density. By attempting to keep the Monte Carlo particle density more uniform, more uniform relative errors for the tallies should be realized.

Two options for forward weighting are possible. For tallies over some area where the entire group-wise flux is needed with low relative uncertainties, the adjoint source should be weighted inversely by the forward flux, $\phi\left(\overrightarrow{r},E\right)$. The other option, for a tally in which only an energy-integrated quantity is desired, is to weight the adjoint inversely by that energy-integrated quantity, $\int \sigma_d(E)\,\phi\left(\overrightarrow{r},E\right)$ dE. For a tally in which the total flux is desired, then the response in the adjoint source is simply $\sigma_d(E) = 1$.

To optimize the forward Monte Carlo simulation for the calculation of some quantity at multiple tally locations or across a mesh tally, the adjoint source must be weighted by the estimate of that quantity. For a tally defined by its spatial location $g\left(\overrightarrow{r}\right)$ and its optional response $\sigma_d(E)$, the standard adjoint source would be $q^+\left(\overrightarrow{r},E\right) = \sigma_d(E)\,\mathrm{g}\left(\overrightarrow{r}\right)$. The forward-weighted adjoint source, $q^+\left(\overrightarrow{r},E\right)$, depending on what quantity is to be optimized, is listed below.

For the calculation of		Adjoint source
Energy and spatially dependent flux	$\phi\left(\overrightarrow{r},E\right)$	$\frac{g\left(\overrightarrow{r}\right)}{\phi\left(\overrightarrow{r},E\right)}$
Spatially dependent total flux	$\int \phi\left(\overrightarrow{r},E\right)dE$	$\frac{g\left(\overrightarrow{r}\right)}{\int \phi\left(\overrightarrow{r},E\right)dE}$
Spatially dependent total response	$\int \sigma_d(E)\,\phi\left(\overrightarrow{r},E\right)dE$	$\frac{\sigma_d(E)\mathrm{g}\left(\overrightarrow{r}\right)}{\int \sigma_d(E)\phi\left(\overrightarrow{r},E\right)dE}$

The bottom line of FW-CADIS is that in order to calculate a quantity at multiple tally locations (or across a mesh tally) with more uniform relative uncertainties, an adjoint source must be developed for an objective

function that keeps some non-physical quantity-related to the Monte Carlo particle density and similar in form to the desired quantity-constant. FW-CADIS uses the solution of a forward discrete-ordinates calculation to properly weight the adjoint source. After that, the standard CADIS approach is used.

4.1.2.5 MAVRIC Implementation of CADIS

With MAVRIC, as with other shielding codes, the user defines the problem as a set of physical models-the material compositions, the geometry, the source, and the detectors (locations and response functions)-as well as some mathematical parameters on how to solve the problem (number of histories, etc.). For the variance reduction portion of MAVRIC, the only additional inputs required are (1) the mesh planes to use in the discrete-ordinates calculation(s) and (2) the adjoint source description—basically the location and the response of each tally to optimize in the forward Monte Carlo calculation. MAVRIC uses this information to construct a Denovo adjoint problem. (The adjoint source is weighted by a Denovo forward flux or response estimate for FW-CADIS applications.) MAVRIC then uses the CADIS methodology: it combines the adjoint flux from the Denovo calculation with the source description and creates the importance map (weight window targets) and the mesh-based biased source. Monaco is then run using the CADIS biased source distribution and the weight window targets.

Denovo

Denovo is a parallel three-dimensional SN code that is used to generate adjoint (and, for FW-CADIS, forward) scalar fluxes for the CADIS methods in MAVRIC. For use in MAVRIC/CADIS, it is highly desirable that the SN code be fast, positive, and robust. The phase-space shape of the forward and adjoint fluxes, as opposed to a highly accurate solution, is the most important quality for Monte Carlo weight-window generation. Accordingly, Denovo provides a step-characteristics spatial differencing option that produces positive scalar fluxes as long as the source (volume plus in-scatter) is positive. Denovo uses an orthogonal, nonuniform mesh that is ideal for CADIS applications because of the speed and robustness of calculations on this mesh type.

Denovo uses the highly robust GMRES (Generalized Minimum Residual) Krylov method to solve the SN equations in each group. GMRES has been shown to be more robust and efficient than traditional source (fixed-point) iteration. The in-group discrete SN equations are defined as

$$\mathbf{L}\psi = \mathrm{MS}\phi + q \tag{4.1.16}$$

where $\mathbf{L}$ is the differential transport operator, $\mathbf{M}$ is the moment-to-discrete operator, $\mathbf{S}$ is the matrix of scattering cross section moments, q is the external and in-scatter source, ϕ is the vector of angular flux moments, and ψ is the vector of angular fluxes at discrete angles. Applying the operator $\mathbf{D}$, where $\phi = \mathbf{D}\psi$, and rearranging terms, casts the in-group equations in the form of a traditional linear system, $\mathbf{A}x = b$,

$$\left(\mathbf{I} - \mathbf{D}\mathbf{L}^{-1}\mathrm{MS}\right) = \mathbf{D}\mathbf{L}^{-1}q. \tag{4.1.17}$$

The operation $\mathbf{L}^{-1}\nu$, where ν is an iteration vector, is performed using a traditional wave-front solve (transport sweep). The parallel implementation of the Denovo wave-front solver uses the well-known Koch-Baker-Alcouffe (KBA) algorithm, which is a two-dimensional block-spatial decomposition of a three-dimensional orthogonal mesh [MAVRIC-BK98]. The Trilinos package is used for the GMRES implementation [MAVRIC-WH03] Denovo stores the mesh-based scalar fluxes in a double precision binary file (*.dff) called a *Denovo flux file*. Past versions of SCALE/Denovo used the TORT [MAVRIC-RS97] *.varscl file format (DOORS package [MAVRIC-RC98]), but this was limited to single precision. Since the rest of the MAVRIC sequence has not yet been parallelized, Denovo is currently used only in serial mode within MAVRIC.

Monaco

The forward Monte Carlo transport is performed using Monaco, a fixed-source shielding code that uses the SCALE General Geometry Package (SGGP, the same as used by the criticality code KENO-VI) and the standard SCALE material information processor. Monaco can use either MG or CE cross section libraries. Monaco was originally based on the MORSE Monte Carlo code but has been extensively modified to modernize the coding, incorporate more flexibility in terms of sources/tallies, and read a user-friendly block/keyword style input.

Much of the input to MAVRIC is the same as Monaco. More details can be found in the Monaco chapter of the SCALE manual (Sect. 8.2).

Running MAVRIC

The objective of a SCALE sequence is to execute several codes, passing the output from one to the input of the next, in order to perform some analysis—tasks that users typically had to do in the past. MAVRIC does this for difficult shielding problems by running approximate discrete-ordinates calculations, constructing an importance map and biased source for one or more tallies that the user wants to optimize in the Monte Carlo calculation, and then using those in a forward Monaco Monte Carlo calculation. MAVRIC also prepares the forward and adjoint cross sections when needed. The steps of a MAVRIC sequence are listed in Table 4.1.1. The user can instruct MAVRIC to run this whole sequence of steps or just some subset of the steps to verify the intermediate steps or to reuse previously calculated quantities in a new analyses.

The MAVRIC sequence can be stopped after key points by using the "parm= *parameter* " operator on the "=mavric" command line, which is the first line of the input file. The various parameters are listed in Table Table 4.1.2. These parameters allow the user to perform checks and make changes to the importance map calculation before the actual Monte Carlo calculation in Monaco.

MAVRIC also allows the sequence to start at several different points. If an importance map and biased source have already been computed, they can then be used directly. If the adjoint scalar fluxes are known, they can quickly be used to create the importance map and biased source and then begin the forward Monte Carlo calculation. All of the different combinations of starting MAVRIC with some previously calculated quantities are listed in the following section detailing the input options.

When using MG cross section libraries that do not have flux-to-dose-rate conversion factors, use "parm=nodose" to prevent the cross section processing codes from trying to move these values into the working library.

MAVRIC creates many files that use the base problem name from the output file. For an output file called "c:path1path2*outputName*.out" or "/home/path1/path2/ *outputName*.inp", spaces in the output name will cause trouble and should not be used.

Table 4.1.1: Steps in the MAVRIC sequence

Cross section calculation	XSProc is used to calculate the forward cross sections for Monaco
Forward Denovo (optional)	
Cross section calculation	XSProc is used to calculate the forward cross sections for Denovo
Forward flux calculation	Denovo calculates the estimate of the forward flux
Adjoint Denovo (optional)	

continues on next page

Table 4.1.1 – continued from previous page

Cross section calculation	XSProc is used to calculate the adjoint cross sections for Denovo
Adjoint flux calculation	Denovo calculates the estimate of the adjoint flux
CADIS (optional)	The scalar flux file from Denovo is then used to create the biased source distribution and transport weight windows
Monte Carlo calculation	Monaco uses the biased source distribution and transport weight windows to calculate the various tallies

Table 4.1.2: Parameters for the MAVRIC command line ("parm=...")

Parameter	MAVRIC will stop after
check	input checking
forinp	Forward Denovo input construction (makes `xkba_b.inp` in the tmp area)
forward	The forward Denovo calculation
adjinp	Adjoint Denovo input construction (makes `xkba_b.inp` in the tmp area)
adjoint	The adjoint Denovo calculation
impmap	Calculation of importance map and biased source

4.1.3 MAVRIC INPUT

The input file for MAVRIC consists of three lines of text ("=mavric" command line with optional parameters, the problem title, and SCALE cross section library name) and then several blocks, with each block starting with "read xxxx" and ending with "end xxxx". There are three required blocks and nine optional blocks. Material and geometry blocks must be listed first and in the specified order. Other blocks may be listed in any order.

Blocks (must be in this order):

- Composition – (required) SCALE standard composition, list of materials used in the problem
- Celldata – SCALE resonance self-shielding
- Geometry – (required) SCALE general geometry description
- Array – optional addition to the above geometry description
- Volume – optional calculation or listing of region volumes
- Plot – create 2D slices of the SGGP geometry

Other Blocks (in any order, following the blocks listed above):

- Definitions – defines locations, response functions, and grid geometries used by other blocks
- Sources – (required) description of the particle source spatial, energy, and directional distributions
- Tallies – description of what to calculate: point detector tallies, region tallies, or mesh tallies
- Parameters – how to perform the simulation (random number seed, how many histories, etc.)
- Biasing – data for reducing the variance of the simulation
- ImportanceMap – instructions for creating an importance map based on a discrete-ordinates calculation

The material blocks (Composition and Celldata) and the physical model blocks (Geometry, Array, Volume, and Plot) follow the standard SCALE format. See the other SCALE references as noted in the following sections for details. The Biasing block and ImportanceMap block cannot both be used.

For the other six blocks, scalar variables are set by "keyword=value", fixed-length arrays are set with "keyword $value_1$ ... $value_N$", variable-length arrays are set with "keyword $value_1$ ... $value_N$ end", and some text and filenames are read in as quoted strings. Single keywords to set options are also used in some instances. The indention, comment lines, and upper/lowercase shown in this document are not required-they are used in the examples only for clarity. Except for strings in quotes (like filenames), SCALE is case insensitive.

After all input blocks are listed, a single line with "end data" should be listed. A final "end" should also be listed, to signify the end of all MAVRIC input. Nine of the blocks are the same input blocks as those used by the functional module Monaco, with a few extra keywords only for use with MAVRIC. These extra keywords are highlighted here, but without relisting all of the standard Monaco keywords for those blocks. See Example 4.1.1 for an overview of MAVRIC input file structure.

4.1.3.1 Composition block

Material information input follows the standard SCALE format for material input. Basic materials known to the SCALE library may be used as well as completely user-defined materials (using isotopes with known cross sections). Input instructions are located in the XSProc chapter (Sect. 7.1) in the SCALE manual. The Standard Composition Library chapter (Sect. 7.2) lists the different cross section libraries and the names of standard materials. An example is as follows:

```
read composition

    uo2 1 0.2 293.0 92234 0.0055 92235 3.5 92238 96.4945 end

    orconcrete 2 1.0 293.0 end

    ss304 3 1.0 293.0 end

end composition
```

Details on the cell data block are also included in the XSProc chapter (Sect. 7.1). When using different libraries for the importance map production (listed at the top of the input) and the final Monte Carlo calculation (listed in the parameters block, if different), make sure that the materials are present in both libraries.

Example 4.1.1: Overall input format for MAVRIC

```
=mavric                          % name of sequence
Some title for this problem      % title
v7-27n19g                        % cross section library name
read composition                 % SCALE material compositions
   ...                           %     [required block]
end composition                  %
read celldata                    % SCALE resonance self-shielding
   ...                           %     [optional block]
end celldata                     %
read geometry                    % SCALE SGGP geometry
   ...                           %     [required block]
end geometry                     %
read array                       % SCALE SGGP arrays
   ...                           %     [optional block]
end array                        %
read volume                      % SCALE SGGP volume calc
```

(continues on next page)

(continued from previous page)

```
   ...                          %     [optional block]
end volume                      %
read plot                       % SGGP Plots
   ...                          %     [optional block]
end plot                        %
read definitions                % Definitions
   ...                          %     [possibly required]
end definitions                 %
read sources                    % Sources definition
   ...                          %     [required block]
end sources                     %
read tallies                    % Tally specifications
   ...                          %     [optional block]
end tallies                     %
read parameters                 % Monte Carlo parameters
   ...                          %     [optional block]
end parameters                  %
read biasing                    % Biasing information
   ...                          %     [optional block]
end biasing                     %
read importanceMap              % Importance map
   ...                          %     [optional block]
end importanceMap               %
end data                        % end of all blocks
end                             % end of MAVRIC input
```

4.1.3.2 SGGP geometry blocks

MAVRIC uses the functional module Monaco for the forward Monte Carlo calculation. Monaco tracks particles through the physical geometry described by the SGGP input blocks, as well as through the mesh importance map and any mesh tallies, which are defined in the global coordinates and overlay the physical geometry. Because Monaco must track through all of these geometries at the same time, users should not use the reflective boundary capability in the SGGP geometry.

For more details on each SGGP geometry block, see the following sections of the KENO-VI chapter Sect. 8.1 of the SCALE Manual.

Geometry – *Geometry Data*

Array – *Array Data*

Volume – *Volume Data*

Plot – *Plot Data*

4.1.3.3 Other blocks shared with Monaco

The definitions, sources, tallies, and biasing blocks are all the same as Monaco. They are all fully described in the Monaco chapter (Sect. 8.2) of the SCALE Manual.

Definitions — *Definitions Block*

Sources — *Sources Block*

Tallies — *Tallies Block*

Biasing — *Biasing Block*

The parameters block includes several keywords that are not included in Monaco (see the *Parameter Block* section of the Monaco chapter (Sect. 8.2) which are used when the cross section library used in the importance

calculations differs from the library used in the final forward Monaco Monte Carlo calculation. The library listed at the beginning of the MAVRIC input file will be used for the importance calculations (forward and adjoint Denovo calculation, formation of the importance map, and biased sources). To use a different MG library in the final Monaco simulation, use the keyword "library=" with the cross section library name in quotes. A cross section library for Monaco will be made using csas-mg. If there are any extra parameters to use ("parm=" in the "=csas-mg" line of the csas-mg input), they can be passed along using the keyword "parmString=" with the extra information in quotes. For example, the following input file would use a coarse-group library for the importance calculations and a fine-group library for the final Monaco, each with CENTRM processing.

```
=mavric parm=centrm
v7-27n19g
...

read parameters

    library="v7-200n47g" parmString="centrm"

    ...

end parameters

...

end data

end
```

To use a CE cross section in the final Monaco step, use the keyword "ceLibrary=" with the cross section library name in quotes. When the "library=" or "ceLibrary=" keywords are used, they should precede the "neutron", "photon", "noNeutron", and "noPhoton" keywords. Table 4.1.3 summarizes all of the keywords in the MAVRIC parameter block.

When using two different cross section libraries, be sure that the responses and distributions are defined in ways that do not depend on the cross section library. For example, any response that is just a list of n values (corresponding to a cross section library of n groups) needs to have the group energies specifically listed so that it can be evaluated properly on the other group structure.

Table 4.1.3: Extra keywords for the parameters block

block	keyword	type	length	default	required	restrictions/comments
read parameters						
	All of the keywords listed in the Monaco manual still apply					
	Final Monaco Monte Carlo calculation is multi-group					
	library=	character		not present	no	multi-group library for final Monaco run, if different than MAVRIC
	parmString=	character		not present	no	parm for MG cross section processing using CSAS-MG
	Final Monaco Monte Carlo calculation is continuous energy					
	ceLibrary=	character		not present	no	continuous energy library name
end parameters						

4.1.3.4 Importance map block

The importance map block is the "heart and soul" of MAVRIC. This block lists the parameters for creating an importance map and biased source from one (adjoint) or two (forward, followed by adjoint) Denovo discrete-ordinates calculations. Without an importance map block, MAVRIC can be used to run Monaco and use its conventional types of variance reduction. If both the importance map and biasing blocks are specified, then only the importance map block will be used. The various ways to use the importance map block are explained in the subsections below. Keywords for this block are summarized at the end of this section, in Table 4.1.4.

Constructing a mesh for the S_N calculation

All uses of the importance map block that run the discrete-ordinates code require the use of a grid geometry that overlays the physical geometry. Grid geometries are defined in the definitions block of the MAVRIC input. The extent and level of detail needed in a grid geometry are discussed in the following paragraphs.

When using S_N methods alone for solving radiation transport in shielding problems, a good rule of thumb is to use mesh cell sizes on the order of a meanfree path of the particle. In complex shielding problems, this could lead to an extremely large number of mesh cells, especially when considering the size of the meanfree path of the lowest energy neutrons and photons in common shielding materials.

In MAVRIC, the goal is to use the S_N calculation for a quick approximate solution. Accuracy is not paramount-just getting an idea of the overall shape of the true importance map will help accelerate the convergence of the forward Monte Carlo calculation. The more accurate the importance map, the better the forward Monte Carlo acceleration will be. At some point there is a time trade-off when the computational time for calculating the importance map followed by the time to perform the Monte Carlo calculation exceeds that of a standard analog Monte Carlo calculation. Large numbers of mesh cells that result from using very small mesh sizes for S_N calculations also use a great deal of computer memory.

Because the deterministic solution(s) for CADIS and FW-CADIS can have moderate fidelity and still provide variance reduction parameters that substantially accelerate the Monte Carlo solution, mesh cell sizes in MAVRIC applications can be larger than what most S_N practioners would typically use. The use of relatively coarse mesh reduces memory requirements and the run time of the deterministic solution(s). Some general guidelines to keep in mind when creating a mesh for the importance map/biased source are as follows:

- The true source regions should be included in the mesh with mesh planes at their boundaries.
- Place point or very small sources in the center of a mesh cell, not on the mesh planes.
- Any region of the geometry where particles could eventually contribute to the tallies (the "important" areas) should be included in the mesh.
- Point adjoint sources (corresponding to point detector locations) in standard CADIS calculations do not have to be included inside the mesh. For FW-CADIS, they must be in the mesh and should be located at a mesh cell center, not on any of the mesh planes.
- Volumetric adjoint sources should be included in the mesh with mesh planes at their boundaries.
- Mesh planes should be placed at significant material boundaries.
- Neighboring cell sizes should not be drastically different.
- Smaller cell sizes should be used where the adjoint flux is changing rapidly, such as toward the surfaces of adjoint sources and shields (rather than in their interiors).

Another aspect to keep in mind is that the source in the forward Monaco Monte Carlo calculation will be a biased mesh-based source. Source particles will be selected by first sampling which mesh cell to use and then sampling a position uniformly within that mesh cell that meets the user criteria of "unit=", "region=", or "mixture=" if specified. The mesh should have enough resolution that the mesh source will be an accurate representation of the true source.

The geometry for the Denovo calculation is specified using the keyword "gridGeometryID=" and the identification number of a grid geometry that was defined in the definitions block. The material assigned to each voxel of the mesh is determined by testing the center point in the SGGP geometry (unless the macro-material option is used-see below).

Macromaterials for S_N geometries

Part of the advantage of the CADIS method is that the adjoint discrete-ordinates calculation only needs to be approximate in order to form a reasonable importance map and biased source. This usually means that the mesh used is much coarser than the mesh that would be used if the problem were to be solved only with a discrete-ordinates code. This coarse mesh may miss significant details (especially curves) in the geometry and produce a less-than-optimal importance map.

To get more accurate solutions from a coarse-mesh discrete-ordinates calculation, Denovo can represent the material in each voxel of the mesh as a volume-weighted mixture of the real materials, called *macromaterials*, in the problem. When constructing the Denovo input, the Denovo EigenValue Calculation (DEVC, see section Sect. 2.4) sequence can estimate the volume fraction occupied by using each real material in each voxel by a sampling method. The user can specify parameters for how to sample the geometry. Note that finer sampling makes more accurate estimates of the material fraction but requires more setup time to create the Denovo input. Users should understand how the macromaterials are sampled and should consider this when constructing a mesh grid. This is especially important for geometries that contain arrays. Careful consideration should be given when overlaying a mesh on a geometry that contains arrays of arrays.

Because the list of macromaterials could become large, the user can also specify a tolerance for how close two different macromaterials can be in order to be considered the same, thereby reducing the total number of macromaterials. The macromaterial tolerance, "`mmTolerance=`", is used for creating a different macromaterial from the those already created by looking at the infinity norm between two macromaterials. The number of macromaterials does not appreciably impact Denovo run time or memory requirements.

Two different sampling methods are available-point testing [MAVRIC-IPE+09] with the keyword `mmPointTest` and ray tracing [MAVRIC-Joh13] with the keyword `mmRayTest`.

Ray Tracing

This method estimates the volume of different materials in the Denovo mesh grid elements by tracing rays through the SGGP geometry and computing the average track lengths through each material. Rays are traced in all three dimensions to better estimate the volume fractions of materials within each voxel. The mmSubCell parameter controls how many rays will be traced in each voxel in each dimension. For example, if mmSubCell= n, then when tracing rays in the z dimension, each column of voxels uses a set of n × n rays starting uniformly spaced in the x and y dimensions. With rays being cast from all three orthogonal directions, a total of 3n2 rays are used to sample each voxel. One can think of subcells as an equally spaced sub-mesh with a single ray positioned at each center. The number of subcells in each direction, and hence the number of rays, can be explicitly given with mmSubCells ny nz nx nz nx ny end keyword for rays parallel to the x axis, y axis, and z axis. Fig. 4.1.1 shows different subcell configurations (in two dimensions) for a given voxel.

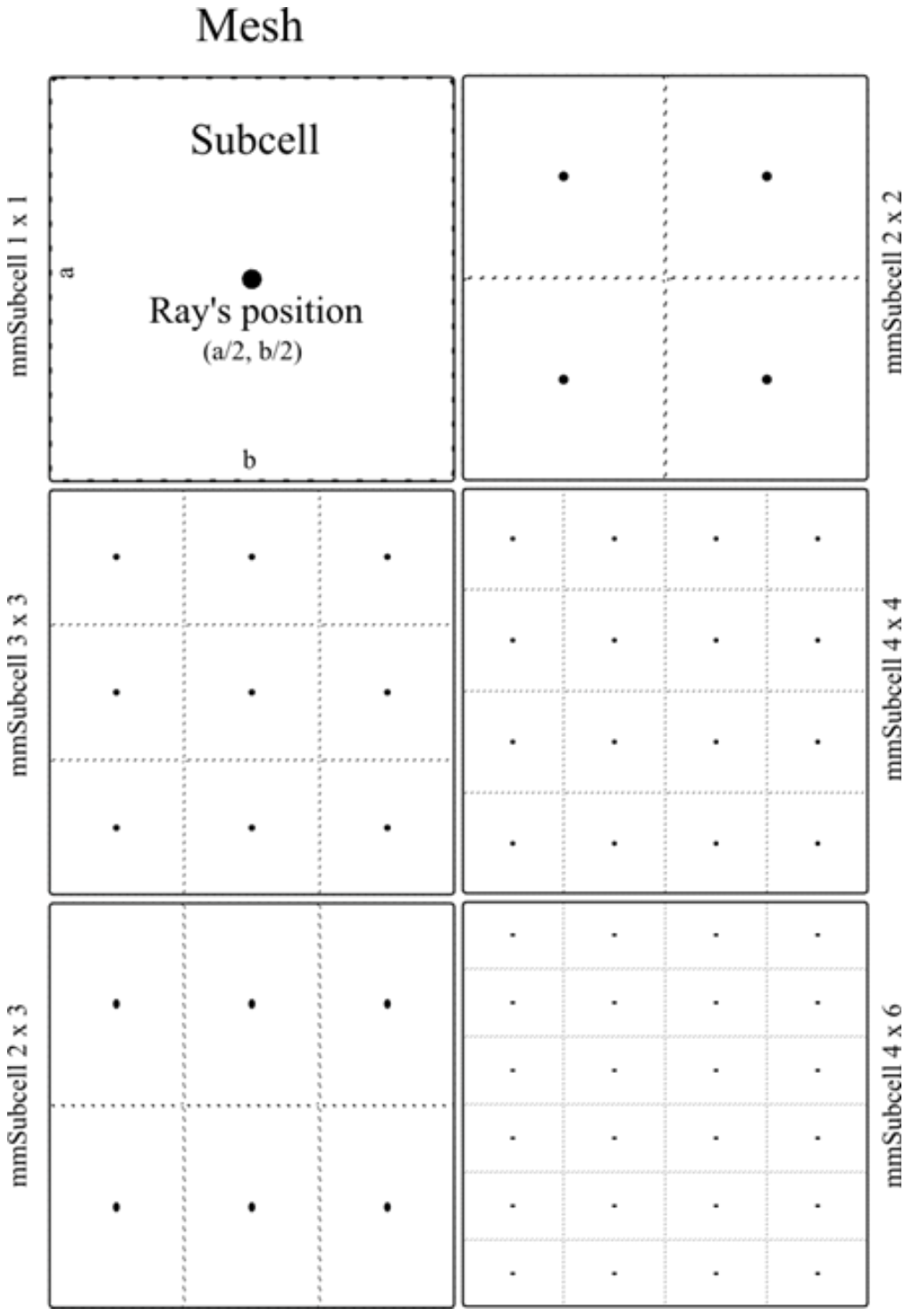


Fig. 4.1.1: Ray positions within a voxel with different mmSubCells parameters.

Ray tracing is a more robust method compared to the simple point testing method used in previous versions of SCALE/MAVRIC; however, it requires more memory than point testing. Ray tracing gives more accurate estimates of volume fractions because track lengths across a voxel give more information than a series of test points. Ray tracing is also much faster than point testing because the particle tracking routines are optimized

to quickly determine lists of materials and distance along a given ray.

Ray tracing operates on the grid geometry supplied by the user and shoots rays in all three directions, starting from the lower bounds of the mesh grid. An example of an arbitrary assembly geometry is shown in Fig. 4.1.2. A ray consists of a number of steps that each correspond to crossing a material boundary along the path of the ray. Ratios of each step's length to the voxel length in the ray's direction determine the material volume fraction of that step in that voxel, and summation of the same material volume fractions gives the material volume fraction of that material in that voxel. Ray tracing through a single voxel that contains a fuel pin is illustrated in Fig. 4.1.3.

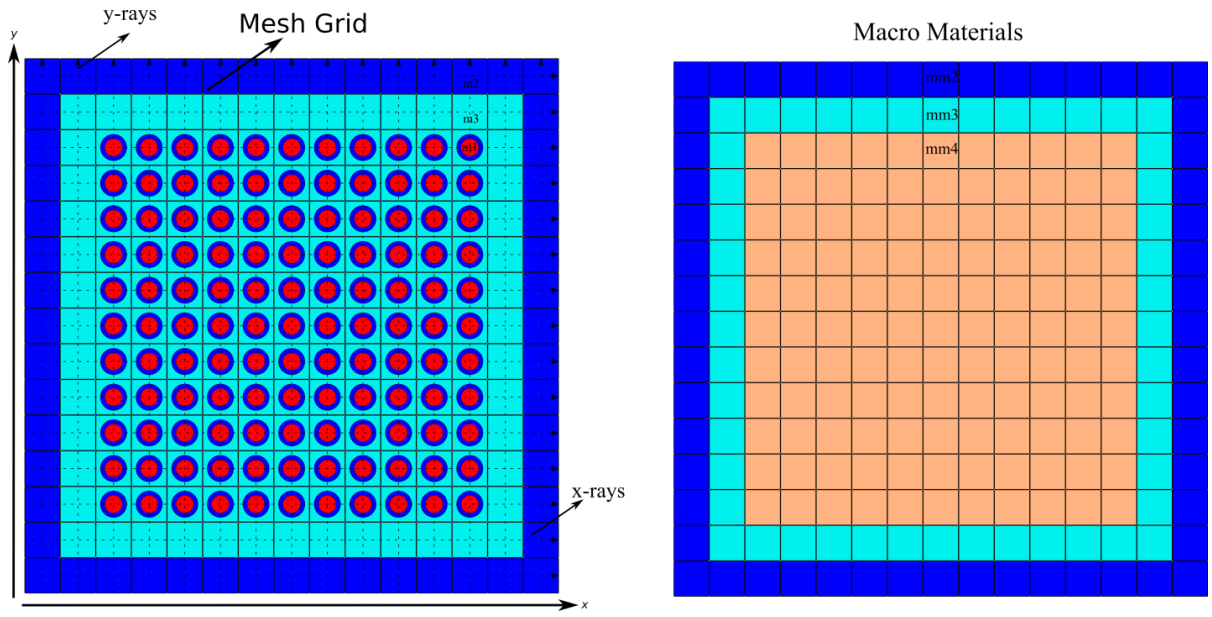


Fig. 4.1.2: Geometry model (left) and the Denovo representation (right) of an assembly using macromaterials determined by ray tracing.

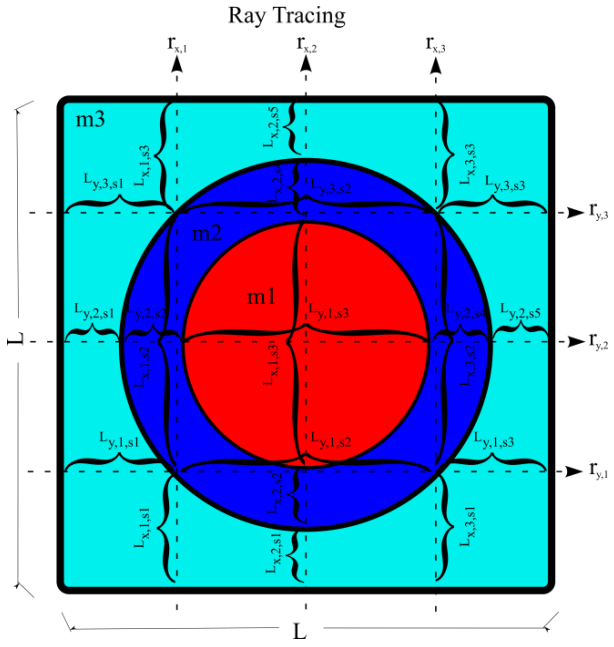


Fig. 4.1.3: Ray tracing (in two dimensions) through a voxel.

The final constructed macromaterials for this model are also shown in Fig. 4.1.2. Voxels that contain only a single material are assigned the original material number in the constructed macromaterials. For the voxels that contain a fuel pin with three different materials, the result is a new macromaterial consisting of the volume weighted fractions of each original material.

After the rays are shot in all three directions, the material volume fractions are updated, and macromaterials are created by using these material volume fractions. Material volume fraction calculations for a single voxel, as shown in Fig. 4.1.3, are given by

$$F_m = \sum_{d=x,y,z} \sum_{r=1}^{N_r} \sum_{s=1}^{N_s} \begin{cases} \frac{L_{d,r,s}}{L_d}, & m_s == m \\ 0, & \text{otherwise} \end{cases} \qquad \textit{and} \qquad V_m = \frac{F_m}{\sum_{n=1}^{N_m} F_n}, \tag{4.1.18}$$

where Fm = sampled fraction of material m in the voxel,

d = direction of the rays (x, y, z),

r = ray number,

N_r = total number of rays in the voxel for direction of d,

s = step number,

N_s = total number of steps for ray r in the voxel for direction of d,

$L_{d,r,s}$ = length of the steps s for ray r in the voxel for direction of d,

L_d = length of the voxel along direction of d,

m_s = material of step s,

m = material number,

N_m = total number of materials in the voxel, and

V_m = volume fraction of material m in the voxel.

Point Testing

The recursive bisection method is utilized in point testing and uses a series of point tests to determine the macromaterial fractions. For a given voxel, the material at the center is compared to the material at the eight corners. If they are all the same, then the entire volume is considered to be made of that material. If they are different, then the volume is divided into two in each dimension. Each subvolume is tested, and the method is then applied to the subvolumes that are not of a single material. When the ratio of the volume of the tested region to the original voxel becomes less than a user-specified tolerance (in the range of 10-1 to 10-4), then further subdivision and testing are stopped. This is illustrated in Fig. 4.1.4.

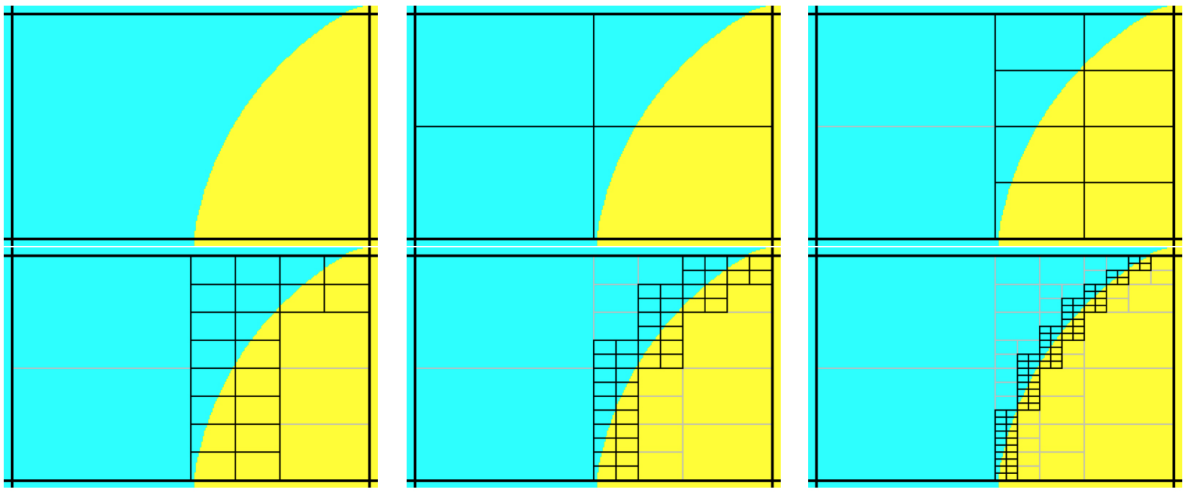


Fig. 4.1.4: Recursive bisection method.

Fig. 4 Successive steps in the recursive macromaterial method

In point testing, the keyword "mmTolerance=f" is interpreted to be where f is the smallest fraction of the voxel volume that can be achieved by bisection method and hence the limiting factor for dividing the voxel. This same tolerance f is also used to limit the number of macromaterials. Before a new macromaterial is created, if one already exists where the fraction of each actual material matches to within the given tolerance, then the existing material will be used. If using only a single point at the center of each voxel, then use "mmTolerance=1". The mmSubCell keyword is not used in point testing.

Example

Fig. 4.1.5 shows an example of a cask geometry with two types of spent fuel (yellows), steel (blue), resin (green), and other metals (gray). When the Denovo geometry is set up by testing only the center of each mesh cell, the curved surfaces are not well represented (upper right). By applying the ray-tracing method and defining a new material made of partial fractions of the original materials, an improved Denovo model can be made. In the lower left of the figure, the Denovo model was constructed using one ray (in each dimension) per voxel and a tolerance of 0.1. This gives 20 new materials that are a mixture of the original 13 actual materials and void. With mmSubCells=3 and an mmTolerance=0.01, 139 macromaterials are created.

A macromaterial table listing the fractions of each macromaterial is saved to a file called "outputName.mmt", where outputName is the name the user chose for his or her output file. This file can be used by the Mesh File Viewer to display the macromaterials as mixtures of the actual materials, as seen in the lower row of

Fig. 4.1.5. See the Mesh File Viewer help pages for more information on how to use colormap files and macromaterial tables.

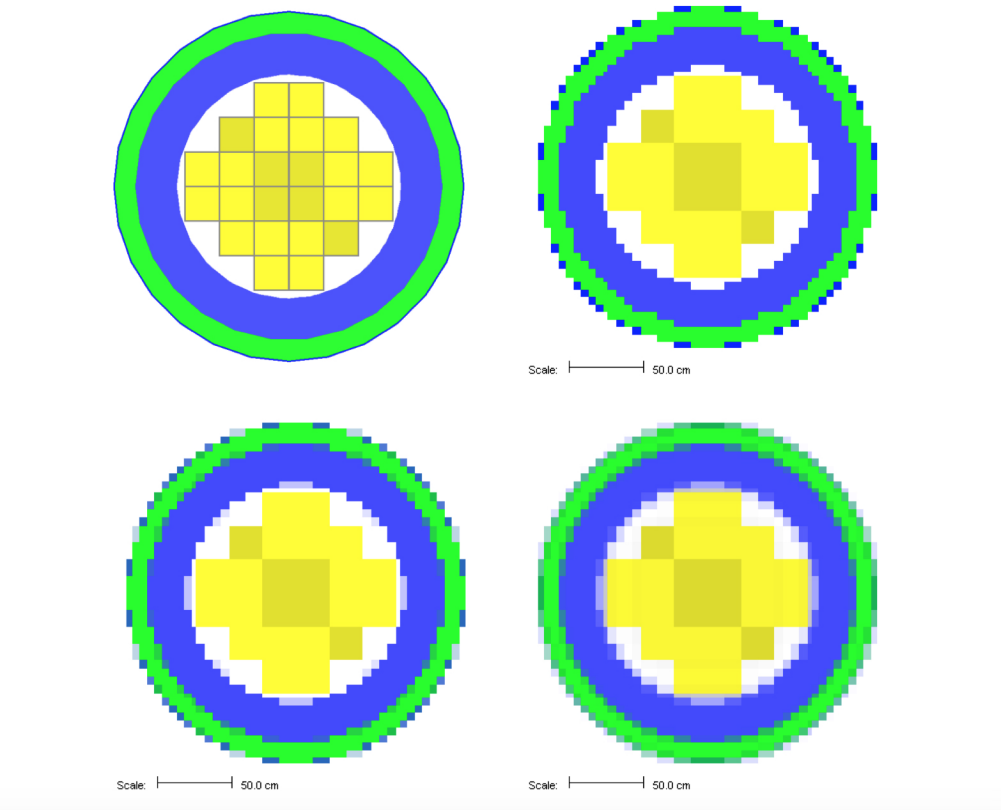


Fig. 4.1.5: Cask geometry model (upper left) and the Denovo representation using cell center testing (upper right). Representations using macromaterials determined by ray tracing are shown for mmSubCell=1/mmTolerance=0.1 (lower left) and mmSubCell=3/mmTolerance=0.01 (lower right).*

Optimizing source/detector problems

For standard source/detector problems in which one tally is to be optimized in the forward Monte Carlo calculation, an adjoint source based on that tally must be constructed. An adjoint source requires a unique and positive identification number, a physical location, and an energy spectrum. The adjoint source location can be specified either by (1) a point location ("locationID=" keyword) or (2) a volume described by a box ("boundingBox" array). A bounding box is specified by maximum and minimum extent in each dimension- x_{max} x_{min} y_{max} y_{min} z_{max} z_{min}-in global coordinates. The boundingBox should not be degenerate (should have volume>0) but can be optionally limited to areas matching a given unit number ("unit="), a given region number ("region="), or a given material mixture number ("mixture="). A mixture and a region cannot both be specified, since that would either be redundant or mutually exclusive. The energy spectrum of an adjoint source is a response function ("responseID=") listing one of the ID numbers of the responses defined in the definitions block. An optional weight can be assigned to each adjoint source using the "weight=" keyword. If not given, the default weight is 1.0.

For example, to optimize a region tally, the user would construct an adjoint source located in the same place as the tally, with an adjoint source spectrum equal to the response function that the tally is computing. Note that the grid geometry 1 and response function 3 must already be defined in the definitions block.

```
read importanceMap
   gridGeometryID=1
   adjointSource 24
       boundingBox 12.0 10.0  5.0 -5.0  10.0 -10.0
       unit=1 region=5
       responseID=3
   end adjointSource
end importanceMap
```

For optimizing a point detector for the calculation of total photon flux, the importance map block would look like the following:

```
read importanceMap
   adjointSource 21
       locationID=4
       responseID=1
   end adjointSource
   gridGeometryID=1
end importanceMap
```

where location 4 is the same location used by the point detector. To calculate total photon flux, response function 1 must be defined in the definitions block similar to this response:

```
read definitions
    response 1
         values 27r0.0 19r1. end
    end response
    ...
end definitions
```

This response is used for computing total photon flux for the 27 neutron/19 photon group coupled cross section library or like this response

```
read definitions
    response 1
         photon
         bounds 1000.0 2.0e7 end
         values  1.0   1.0   end
    end response
    ...
end definitions
```

which is independent of the cross section library.

Multiple adjoint sources

If there are several tallies in very close proximity and/or several different responses being calculated by the tallies, multiple adjoint sources can be used.

```
read importanceMap
   gridGeometryID=1
   adjointSource 1
       locationID=4  responseID=20
   end adjointSource
   adjointSource 2
       locationID=5  responseID=21
       weight=2.0
   end adjointSource
end importanceMap
```

Note that adjoint sources using point locations can be mixed with volumetric adjoint sources (using bounding boxes).

Options for Denovo S_n calculations

While the default values for various calculational parameters and settings used by Denovo for the MAVRIC sequence should cover most applications, they can be changed if desired. The two most basic parameters are the quadrature set used for the discrete ordinates and the order of the Legendre polynomials used in describing the angular scattering. The default quadrature order that MAVRIC uses is a level symmetric S_8 set, and the default scattering order is P_3 (or the maximum number of coefficients contained in the cross-section library if less than 3). S_8/P_3 is an adequate choice for many applications, but the user is free to changes these. For complex ducts or transport over large distances at small angles, S_{12} may be required. S_4/P_1 or even S_2/P_0 would be useful in doing a very cursory run to confirm that the problem was input correctly, but this would likely be inadequate for weight window generation in a problem that is complex enough to require advanced variance reduction.

These and other Denovo options are applied to both the forward and the adjoint calculations that are required from the inputs given in the importance map block.

In problems with small sources or media that are not highly scattering, discrete ordinates can suffer from "ray effects" [MAVRIC-Lat71, MAVRIC-Lat68] where artifacts of the discrete quadrature directions can be seen in the computed fluxes. Denovo has a first-collision capability to help alleviate ray effects. This method computes the uncollided flux in each mesh cell from a point source. The uncollided fluxes are then used as a distributed source in the main discrete-ordinates solution. At the end of the main calculation, the uncollided fluxes are added to the fluxes computed with the first collision source, forming the total flux. While this helps reduce ray effects in many problems, the first-collision capability can take a significant amount of time to compute on a mesh with many cells or for many point sources.

Adjoint sources that use point locations will automatically use the Denovo first-collision capability. Volumetric adjoint sources (that use a boundingBox) will be treated without the first-collision capability. The keywords "firstCollision" and "noFirstCollision" will be ignored by MAVRIC for adjoint calculations. Keywords for Denovo options in the importance map block are summarized at the end of this section, in Table 4.1.5.

Starting with an existing adjoint flux file

An importance map can be made from an existing Denovo flux file by using the keyword "adjointFluxes=" with the appropriate file name in quotes. The file must be a binary file using the *.dff file format, and the number of groups must match the number of groups in the MAVRIC cross section library (i.e., the library entered on the third line of the MAVRIC input file). Instead of performing an adjoint calculation, the fluxes read from the file will be used to create both the mesh-based importance map and the biased mesh source.

```
read importanceMap
    adjointFluxes="c:\mydocu~1\previousRun.adjoint.dff"
    gridGeometry=7
end importanceMap
```

If the "adjointFluxes=" keyword is used and any adjoint sources are defined, an error will result. If a forward flux file is supplied for forward-weighting the adjoint source (see below), then an adjoint flux file cannot be specified.

The grid geometry is not required when using a pre-existing flux file. If grid geometry is not supplied, one will be created from the mesh planes that are contained in the Denovo flux file (which were used to compute the fluxes in that file).

Forward-weighting the adjoint source

To optimize a mesh tally or multiple region tallies/point detector tallies over a large region, instead of a uniform weighting of the adjoint source, a weighting based on the inverse of the forward response can be performed. This requires an extra discrete-ordinates calculation but can help the forward Monte Carlo calculation compute the mesh tally or group of tallies with more uniform statistical uncertainties.

The same grid geometry will be used in both the forward calculation and the adjoint calculation, so the user must ensure that the mesh covers all of the forward sources and all of the adjoint sources, even if they are point sources.

To use forward-weighted CADIS, specify either of the keywords — "respWeighting" or "fluxWeighting". For either, MAVRIC will run Denovo to create an estimate of the forward flux, $\phi\left(\overrightarrow{r}, E\right)$. For response weighting ("respWeighting"), each adjoint source is inversely weighted by the integral of the product of the response function used in that adjoint source and the estimate of the forward flux. For an adjoint source described by the geometric function $g(\overrightarrow{r})$ and the response function $\sigma_d(E)$ (note that $\sigma_d(E) = 1$ for computing total fluxes), the forward-weighted adjoint source becomes

$$q_i^+\left(\overrightarrow{r}, E\right) = \frac{\sigma_d(E)\, g(\overrightarrow{r})}{\int \sigma_d(E)\ \phi\left(\overrightarrow{r}, E\right) \mathrm{dE}} \ . \tag{4.1.19}$$

Response weighting will calculate more uniform relative uncertainties of the integral quantities of the tallies in the final Monte Carlo calculation.

To optimize the calculation of the entire group-wise flux with more uniform relative uncertainties in each group, the adjoint source should be weighted inversely by the forward flux, $\phi\left(\overrightarrow{r}, E\right)$, using the "fluxWeighting" keyword. For an adjoint source described by the geometric function $g(\overrightarrow{r})$ and the response function $\sigma_d(E) = 1$, the forward-weighted adjoint source becomes

$$q_i^+\left(\overrightarrow{r}, E\right) = \frac{\sigma_d(E)\, g(\overrightarrow{r})}{\phi\left(\overrightarrow{r}, E\right)} \ . \tag{4.1.20}$$

For example, consider a problem with a single source and two detectors, one near the source that measures flux and one far from the source that measures some response. In a standard Monte Carlo calculation, it is expected that since more Monte Carlo particles cross the near detector, it will have a much lower relative uncertainty than the far detector. Standard CADIS could be used to optimize the calculation of each in separate simulations:

To optimize the flux in the near detector:

```
read importanceMap
    gridGeometryID=1
    adjointSource 1
        boundingBox x1 x2 y1 y2 z1 z2
        responseID=1
    end adjointSource
end importanceMap
```

To optimize the response in the far detector:

```
read importanceMap
    gridGeometryID=1
    adjointSource 2
        boundingBox u1 u2 v1 v2 w1 w2
        responseID=6
    end adjointSource
end importanceMap
```

where response 1 was defined as $\sigma_1(E) = 1$ and response 6 was defined as $\sigma_6(E) =$ flux-to-response conversion factors. The two options for forward weighting allow the tallies for both detectors to be calculated in the same MAVRIC simulation. Using "fluxWeighting", the importance map and biased source will be made to help distribute Monte Carlo particles evenly through each energy group and every voxel in both detectors, making the relative uncertainties close to uniform. With "respWeighting", the importance map and biased source will optimize the total integrated response of each tally.

To optimize $\phi\left(\vec{r}, E\right)$ in each detector

```
read importanceMap
    gridGeometryID=1
'   near detector
    adjointSource 1
        boundingBox x1 x2 y1 y2 z1 z2
        responseID=1
    end adjointSource
'   far detector
    adjointSource 2
        boundingBox u1 u2 v1 v2 w1 w2
        responseID=6
    end adjointSource
    fluxWeighting
end importanceMap
```

To optimize a total response $\int \sigma_d(E)\,\phi\left(\vec{r}, E\right)dE$ (either total flux or total dose)

```
read importanceMap
    gridGeometryID=1
'   near detector
    adjointSource 1
        boundingBox x1 x2 y1 y2 z1 z2
        responseID=1
    end adjointSource
'   far detector
    adjointSource 2
        boundingBox u1 u2 v1 v2 w1 w2
        responseID=6
    end adjointSource
    respWeighting
end importanceMap
```

Using flux weighting, the adjoint source will be

$$q^{+}\left(\vec{r}, E\right) = \frac{\sigma_1(E)\, g_{\text{near}}(\vec{r})}{\phi\left(\vec{r}, E\right)} + \frac{\sigma_6(E)\, g_{\text{far}}(\vec{r})}{\phi\left(\vec{r}, E\right)}, \tag{4.1.21}$$

or using response weighting, the adjoint source will be

$$q^{+}\left(\vec{r}, E\right) = \frac{\sigma_1(E)\, g_1(\vec{r})}{\int \sigma_1(E)\,\phi\left(\vec{r}, E\right) dE} + \frac{\sigma_6(E)\, g_2(\vec{r})}{\int \sigma_6(E)\,\phi\left(\vec{r}, E\right) dE}. \tag{4.1.22}$$

This implementation is slightly different from the original MAVRIC in SCALE 6. The current approach is simpler for the user and allows the importance parameters to optimize the final Monte Carlo calculation for the calculation of two different responses in two different areas.

If the number of mesh cells containing the true source is less than 10, then MAVRIC will convert these source voxels to point sources and Denovo will automatically use its first-collision capability to help reduce ray effects in the forward calculation. The user can easily override the MAVRIC defaults-to force the calculation of a first-collision source no matter how many voxels contain source; this can be done by using the keyword "firstCollision". To prevent the calculation of a first-collision source, the keyword "noFirstCollision" can be used. If the keywords "firstCollision" or "noFirstCollision" are used, then they will only apply to the forward calculation, not the subsequent adjoint calculation.

The keyword "saveExtraMaps" will save extra files that can be viewed by the Mesh File Viewer. The source used by the forward Denovo calculation is stored in "*outputName*.dofs.3dmap", where *outputName* is the name the user chose for his output file.

Forward weighting with an existing forward flux file

Similar to the capability of using pre-existing adjoint flux files, MAVRIC can use a pre-existing forward flux file to create a forward-weighted adjoint source without performing the forward Denovo calculation. The user may specify the *.dff file containing the forward fluxes using the keyword "forwardFluxes=". The filename should be enclosed in quotes, and the file must be a binary file using the Denovo flux file format. The number of groups must match the number of groups in the MAVRIC cross section library (i.e., the library entered on the third line of the MAVRIC input file).

```
read importanceMap
     forwardFluxes="c:\mydocu~1\previousRun.forward.dff"
     gridGeometry=7
     adjointSource 1
        ...
     end adjointSource
     respWeighting
end importanceMap
```

When using a pre-existing forward flux file, either "respWeighting" or "fluxWeighting" must still be specified.

Using the importance map

An importance map produced by the importance map block consists of the target weight values as a function of position and energy. The upper weight window used for splitting and the lower weight window used for Russian roulette are set by the window ratio. The window ratio is simply the ratio of the weight window's upper bound to the weight window lower bound, with the target weight being the average of the upper and lower bounds.

The keyword "windowRatio=" can be used within the importance map block to specify what window ratio will be used with the importance map that is passed to the Monaco forward Monte Carlo calculation. For a windowRatio of r, the upper weights for splitting, w_{max}, and the lower weights for Russian roulette, w_{min}, are set as

$$w_{\min} = \frac{2}{r+1}\overline{w} \tag{4.1.23}$$

and

$$w_{\max} = \frac{2r}{r+1}\overline{w} \tag{4.1.24}$$

for the target weight $\overline{w}$ in each mesh cell and for each energy of the importance map. The default value for the windowRatio is 5.0.

Other notes on importance map calculations

Since the importance map calculations all take place using mesh geometry, one of the first steps that occurs is to create a mesh representation of the true source (the forward source) on the same grid. This procedure uses the same two methods as the Monaco mesh source saver routine. Mesh cells can be subdivided and tested to see if they are within the defined source, or a set number of points can be sampled from the source. The keywords "subCells=" and "sourceTrials=" are used in the importance map block to change the default settings for constructing the mesh representation of the forward source.

If macromaterials are used ("mmTolerance<1") and the adjoint source is limited to a particular material, then the amount of adjoint source in a mesh voxel will be weighted by the material amount in that voxel.

In SCALE/MAVRIC, Denovo is called as a fixed-source S_N solver and cannot model multiplying media. Neither forward nor adjoint neutron calculations from Denovo will be accurate when neutron multiplication is a major source component. If neutron multiplication is not turned off in the parameters block of the MAVRIC input (using "fissionMult=0"), a warning will be generated to remind the user of this limitation.

By default, MAVRIC instructs Denovo not to perform outer iterations for neutron problems if the cross section library contains upscatter groups. This is because the time required calculating the fluxes using upscatter can be significantly longer than without. For problems in which thermal neutrons are an important part of the transport or tallies, the user should specify the keyword "upScatter=1" in the importance map block. This will instruct Denovo to perform the outer iterations for the upscatter groups, giving more accurate results but taking a much longer time for the discrete-ordinates calculation.

When performing a MAVRIC calculation using a coarse-group energy structure for Denovo (for example with the 27/19 library) but a fine-group energy structure (with the 200/47 library) for the final Monaco calculation, the source biasing parameters are determined on the coarse-group structure. The importance map (*.mim) file and the biased mesh source (*.msm) files all use the coarse-group structure. The source biasing information is then applied to fine-group mesh versions of the sources, resulting in the *.sampling.*.msm files. This way, the biased sources used in the final Monaco calculation retain their fine-group resolution. This can be especially important in representing the high-energy portion of the fission neutron distribution for example. When using CE-Monaco, the source sampling routines first use the *.msm files to determine the source particle's voxel and energy group. From that voxel and energy group, the user-given source distributions are used to sample the specific starting location and specific energy of the source particle. This way, the CE-Monaco calculation samples the true CE distributions.

Table 4.1.4: Keywords for the importance map block

block	keyword	type	length	default	required	restrictions/comments
importanceMap						
	Perform an adjoint S_N calculation using one (or more) adjoint source(s) and a gridGeometry					
	gridGeometryID=	integer			yes	matches one of the id numbers from gridGeometries
	adjointSource id	integer			yes	non-negative integer, unique among adjointSources
	locationID=	integer			a*	matches one of the id numbers from locations
	boundingBox	real	6		b*	parameters: x_{max} x_{min} y_{max} y_{min} z_{max} z_{min}
						*required: either a) locationID= or b) boundingBox
	responseID=	integer			c*	single id number from responses
	responseIDs	integer	any	none	d*	list of id numbers from responses
						*required: either c) responseID= or d) responseIDs
	weight=	real		1.0	no	positive real number
	unit=	integer		-1	no	limit adjoint source in boundingBox to a unit
	region=	integer		-1	no	limit adjoint source in boundingBox to a region of a unit
	mixture=	integer		-1	no	limit adjoint source in boundingBox to a mixture
	end adjointSource					
	Constructing the Denovo geometry using macro materials					
	macromaterial					
	mmSubCell=	integer		1	no	rays per dimension to throw at each voxel
	mmTolerance=	real		0.01	no	smallest volume fraction for macromaterial
	mmSubCells	integer	6		no	rays per dimension to throw (x:ny,nz; y:nx,nz; z: nx,ny)
	mmPointTest				no	use recursive bisection point testing method
	mmRayTest				no	use ray tracing method
	mmRTSpeed				no	optimize ray-tracing method for speed
	mmRTMemory				no	optimize ray-tracing method for memory conservation
	end macromaterial					
	Constructing the mesh version of the true source					
	subCells=	integer		2	no	subcells per cell (each dimension)
	sourceTrials=	integer		1000000	no	how many source particles to sample
	reduce			not present	no	store the smallest cuboid around the voxels with source
	Perform a forward S_N calculation and weight the adjoint source					
	fluxWeighting			not present	no	weight adjoint source with forward flux
	respWeighting			not present	no	weight adjoint source with integrated forward response
	saveExtraMaps			not present	no	save extra 3dmap files associated with forward calculation
	firstCollision			not present	no	forces the use a a first collision source
	noFirstCollision			not present	no	does not allow the use of a first collision source
	Use existing forward flux file for weighting the adjoint source					
	forwardFluxes=	string		not present	no	legal file name for current system, in quotes
	Use existing adjoint flux file to create importance map					
	adjointFluxes=	string		not present	no	legal file name for current system, in quotes
	While using the importance map					
	windowRatio=	real		5.0	no	real number greater than one
	mapMultiplier=	real		1.0	no	multiply targetWeights in imp. Map
end importanceMap						

Table 4.1.5: Denovo options for the importance map block

block	keyword	type	length	default	required	restrictions/comments
importanceMap						
	Optional Denovo S_N code parameters					
	discretization=	integer		4	no	0-diamond-difference, 1-diamond-difference with flux fix-up:lin0, 2-theta-weighted diamond-difference, 3-linear-discontinuous finite element, 4-step characteristics, 5-trilinear discontinuous finite elem
	quadType=	integer		0	no	0-level symmetric, 1-Gauss-Legendre product, 2-QR
	legendre=	integer		3*	no	P_L, L=highest Legendre polynomial, L=0,1,2,3,… *default is to use min(the highest available in the data,3)
	tportcorrection=	integer		1*	no	transport correction: 0-none, 1-diagonal, 2-Cesaro *For P_2 or higher, the default is 2 (Cesaro)
	upScatter=	integer		0	no	upscatter iterations: 0-none, 1-yes, 2-ignore
	xblocks=	integer		1	no	parallel stuff
	yblocks=	integer		1	no	parallel stuff
	zblocks=	integer		1	no	parallel stuff
	numSets=	integer		1	no	?
	partUpscatter=	integer		1	no	partition upscatter (0-no, 1-yes)
	quadrature=	integer		8	no	level symmetric S_N quadrature, N=2, 4, 6, 8, 10, 12, 14, 16
	polarsPerOct=	integer		0	no	Gauss-Legendre product quadrature or QR
	azimuthsPerOct=	integer		0	no	Gauss-Legendre product quadrature or QR
	maxIters=	integer		20	no	maximum number of iterations
	diagnostics=	integer		0	no	0-no diganostics, 1-all diagnostics
	output=	integer		0	no	0-no ouput, 1-all output
	krylovSpaceSize=	integer		10	no	size in memory for Krylov space
	tolerance=	double		1.E-03	no	tolerance used in convergence test
	krylovType=	integer		0	no	0-GMRES, 1-BiCGStab
	eigenSolver=	integer		0	no	0-power iteration, 1-Arnoldi, 2-shifted inverse
	multiGSolover=	integer		0	no	0-Gauss-Seidel, 1-Krylov
	withinGSolver=	integer		0	no	0-Krylov, 1-residual Krylov, 2-source iteration
	mgSettings=	integer		1	no	0-user supplied, 1-automatic
	upGroupSolver=	integer		0	no	0-same as within-group solver, 1-Krylov, 2-residual Krylov, 3-source iteration, 4-single source iteration
	acceleration=	integer		0	no	0-none, 1-two grid
	maxItersMG=	integer		20	no	maximum number of iterations
	toleranceMG=	double		0.001	no	tolerance used in convergence test
end importanceMap						

4.1.4 MAVRIC OUTPUT

4.1.4.1 Main text output file

Similar to other SCALE sequences, MAVRIC returns a text output file containing the output from the SCALE driver, the sequence itself, and all of the functional modules called. The SCALE driver output first displays the problem input file, and then the first reading of the input file by the MAVRIC sequence is shown (which includes some material processing information). If there are any errors or warnings about the input file, they will be shown next. Next in the output file are the different passes through the MAVRIC sequence—up to 10 parts. If any errors or warning messages (such as lack of memory) are generated during processing, they will be displayed here. Finally, the output files from each functional module are concatenated to the above output and shows the files returned to the user.

First, the Monaco section of output first reviews the input it received. First the geometry is reviewed, showing which materials are used in each region and the volume of that region, if input or calculated. Then a detailed list of other Monaco input is reviewed: cross section parameters, data definitions, the source description, the tallies, the Monte Carlo parameters, and the biasing parameters. For MAVRIC calculations, if an importance map is used, then its summary is also given. The "Mesh Importance Map Characterization" shows where the importance map may be changing too rapidly and may require more refinement.

For each Monaco batch, the output file lists the batch time and the starting random number for the next batch, which may be useful in rerunning only a portion of a problem. Once all of the batches are completed, a list of the various tally files that have been created is given. Finally, the tallies are summarized in a section entitled "Final Tally Results Summary." For each point detector, the total neutron and photon fluxes (uncollided and total) are given as well as the final response values for each response function. For each region tally, the total neutron and photon fluxes (both track-length and collision density estimates) are listed, followed by the final response values for each response function. Group-by-group details are saved to separate files for each tally.

4.1.4.2 Additional output files

In addition to the generous amount of data contained in the MAVRIC text output file, many other files are created containing the intermediate data used by the sequence and the final tally data. Many of the files produced can be viewed using the Mesh File Viewer or the Interactive Plotter capabilities of Fulcrum, which is distributed with SCALE. (Note that most of the images in this document were taken from the Mesh File Viewer from SCALE 6.1.) Table 4.1.6 lists the other output files, based on the name of the main output file (here called *outputName)*, that are available to the user. These files will be copied back to the directory where the input file was located. Many of the files come from Monaco and are discussed in the Monaco chapter of the SCALE manual (Sect. 8.2).

Other files that the user may be interested in are listed in Table 4.1.7. These files are kept in the temporary directory where SCALE executes and are not copied back to the directory where the input file was located, unless specifically requested using a SCALE "`shell`" command. Curious users may also be interested in viewing the various input files (i_*) that the MAVRIC sequence writes in order to run the SCALE functional modules.

Table 4.1.6: Output files that are copied back to user's area when the sequence finishes[a]

Filename	Viewer	Description
Output Summary		
outputName.out		main text output file, contains results summary
Diagnostic files		
outputName.respid.chart	P	response input and MG representation for response *id*
outputName.gridid.3dmap	V	mesh version of geometry using grid geometry *id*
outputName.cylid.3dmap	V	mesh version of geometry using cylindrical geometry *id*
outputName.distid.chart	P	distribution input and sampling test for distribution *id*
Mesh Source Saver		
filename.msm	V	mesh representation of a single source or total source
filename.id.msm	V	mesh representation of multiple sources
filename.sampling.msm	V	biased representation of a single source or total source
filename.sampling.id.msm	V	biased representation of multiple sources
Importance Map Generation		
outputName.geometry.3dmap	V	voxelized geometry (cell-center testing only)
outputName.forward.dff	V	scalar forward flux estimate, $\phi(x, y, z, E)$
outputName.adjoint.dff	V	scalar adjoint flux estimate, $\phi^{+}(x, y, z, E)$
outputName.mim	V	Monaco mesh importance map, $\overline{w}(x, y, z, E)$

continues on next page

Table 4.1.6 – continued from previous page

Filename	Viewer	Description
outputName.msm	V	Monaco mesh source, $\widehat{q}(x, y, z, E)$
outputName.mmt	V	macro-material table

Tally Files		
outputName.pdid.txt		detailed results for point detector tally *id*
outputName.pdid.chart	P	batch convergence data for point detector tally *id*
outputName.rtid.txt		detailed results for region tally *id*
outputName.rtid.chart	P	batch convergence data for region tally *id*
outputName.mtid.3dmap	V	mesh tally for meshTally *id*
outputName.mtid.respxx.3dmap	V	mesh tally of response by group for meshTally *id* response xx
outputName.mtid.flux.txt		detailed results for the group-wise flux of meshTally *id*
outputName.mtid.tfluxtxt		detailed results for total flux of meshTally *id*
outputName.mtid.respxx.txt		detailed results for response xx of meshTally *id*

[a] V – can be displayed with the Mesh File Viewer capabilities of Fulcrum. *P* – can be displayed with the 2D plotting capabilities of Fulcrum.

Table 4.1.7: Other intermediate files-available in the temporary directory (may be useful for testing and debugging)

Filename	**Description**
ft02f001	AMPX formatted cross sections for Denovo
fort.51	text file, listings of the mixing table for Monaco
fort.52	text file, review of MAVRIC sequence input variables
fort.54	energy bin boundaries for the current cross section library
xkba_b.inp	binary input file for Denovo - rename to have a *.dsi extension (Denovo simple input) to be viewed via Mesh

4.1.5 SAMPLE PROBLEMS

4.1.5.1 Graphite shielding measurements with CADIS

As shown in the Monaco sample problem for simulating the Ueki shielding experiments (Monaco chapter Graphite Shielding Measurements) (Sect. 8.2.5.4), as the amount of shielding material between a source and detector increases, the time required to reach a certain level of relative uncertainty increases quickly. This example will use the MAVRIC automated variance reduction capability to optimize the calculation of the dose rate at the detector location by specifying an importance map block with an adjoint source made from the detector response function and the detector location.

Input file

The following is a listing of the file `mavric.graphiteCADIS.inp` located in the SCALE `samples\input` directory. This calculation will use the coarse-group shielding library (27n19g) for all of the importance map calculations and the fine-group library (200n47g) for the final Monaco step. Additions, compared to the file `monaco.graphite.inp`, include a grid geometry for the Denovo computational mesh, a mesh tally to better visualize the particle flow, and the importance map block to optimize the Monte Carlo calculation of the point detector.

```
=mavric
Monaco/MAVRIC Training - Exercise 3. Graphite Shielding Measurements Revisited
v7-27n19g

'------------------------------------------------------------------------------
' Composition Block - standard SCALE input
'------------------------------------------------------------------------------
read composition
    para(h2o)    1         1.0 293.0 end
    carbon       2 den=1.7 1.0 300.0 end
end composition

'------------------------------------------------------------------------------
' Geometry Block - SCALE standard geometry package (SGGP)
'------------------------------------------------------------------------------
read geometry
    global unit 1
        cuboid  1   25.0 -25.0  25.0 -25.0   25.0 -25.0
        cone    2  10.35948  25.01   0.0 0.0   rotate  a1=-90 a2=-90 a3=0
        cuboid  3   90.0  70.0  40.0 -40.0   40.0 -40.0
        cuboid 99  120.0 -30.0  50.0 -50.0   50.0 -50.0
        media 1 1  1 -2
        media 0 1  2
        media 2 1  3
        media 0 1  99 -1  -2 -3
    boundary 99
end geometry

'------------------------------------------------------------------------------
' Definitions Block
'------------------------------------------------------------------------------
read definitions
    location 1
        position 110 0 0
    end location
    response 5
        title="ANSI standard (1977) neutron flux-to-dose-rate factors"
        specialDose=9029
    end response
    distribution 1
        title="Cf-252 neutrons, Watt spectrum a=1.025 MeV and b=2.926/MeV"
        special="wattSpectrum"
        parameters 1.025 2.926 end
    end distribution
    gridGeometry 7
        title="large meshes in paraffin, 5 cm mesh for shield thicknesses"
        xLinear 5 -25 25
        xLinear 12 30 90
        xplanes 100 110 120 -30 end
        yplanes -50 -40 40 50 end
        yLinear 7 -35 35
        zplanes -50 -40 40 50 end
        zLinear 7 -35 35
    end gridGeometry
end definitions
```

(continues on next page)

(continued from previous page)

```
'------------------------------------------------------------------------------
' Sources Block
'   Cf-252 neutrons, Watt fission spectrum model
'   with a=1.025 MeV and  b=2.926/MeV
'------------------------------------------------------------------------------
read sources
    src 1
        title="Cf-252 neutrons, Watt fission spectrum model"
        strength=4.05E+07
        cuboid  0.01 0.01  0 0  0 0
        neutrons
        eDistributionID=1
    end src
end sources

'------------------------------------------------------------------------------
' Tallies Block
'------------------------------------------------------------------------------
read tallies
    pointDetector 1
        title="center of detector"
        locationID=1
        responseID=5
    end pointDetector
    meshTally 1
        title="example mesh tally"
        gridGeometryID=7
        responseID=5
        noGroupFluxes
    end meshTally
end tallies

'------------------------------------------------------------------------------
' Parameters Block
'------------------------------------------------------------------------------
read parameters
    randomSeed=00003ecd7b4e3e8b
    library="v7-200n47g"
    perBatch=10000 batches=10
    fissionMult=0   noPhotons
end parameters

'------------------------------------------------------------------------------
' Importance Map Block
'------------------------------------------------------------------------------
read importanceMap
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
    gridGeometryID=7
            macromaterial
        mmTolerance=0.01
    end macromaterial
end importanceMap

end data
end
```

Output

MAVRIC results for the point detector response for the 20 cm case are shown below and in Fig. 4.1.6.

```
Neutron Point Detector 1. center of detector
                        average      standard     relat      FOM     stat checks
    tally/quantity       value       deviation    uncert    (/min)   1 2 3 4 5 6
    ------------------  ----------  ----------  -------  --------  -----------
    uncollided flux     1.06384E+01 1.88744E-02 0.00177
    total flux          2.36367E+02 5.47276E+00 0.02315  8.10E+02  X - X - X -
    response 5          1.28632E-02 1.74351E-04 0.01355  2.36E+03  X X X X X X
    ------------------  ----------  ----------  -------  --------  -----------
```

This problem took only ~2.5 minutes (0.2 in Denovo and 2.3 minutes in Monaco) on the same processor as the 20 minute analog case. (The figure of merit [FOM] is 15 times higher than the analog.) Note that the point detector dose rate is the same as the Monaco analog sample problem, but the relative uncertainty is smaller with less computation time. CADIS has optimized the calculation by focusing on neutrons that contribute to the dose rate at the detector location at the expense of neutrons in the paraffin block. This is demonstrated by the mesh tally of dose rates where the values for the dose rate are lower in the paraffin block and the relative uncertainties are higher. Since the calculation was optimized for the position of the detector, dose rates in other parts of the problem are underestimated and should not be believed.

The mesh tally shows that the CADIS calculation did not follow as many particles deep into the paraffin block, so the uncertainties are greater there, but that is what this problem was supposed to do-reduce the uncertainty at the point detector at the expense of the other portions of the problem.

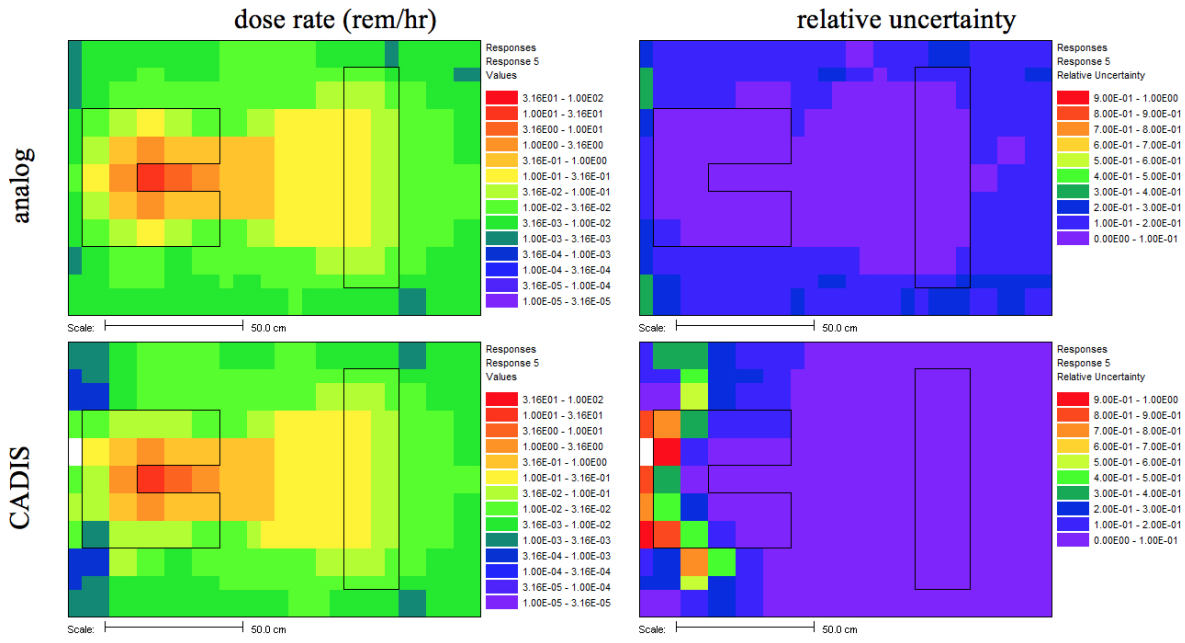


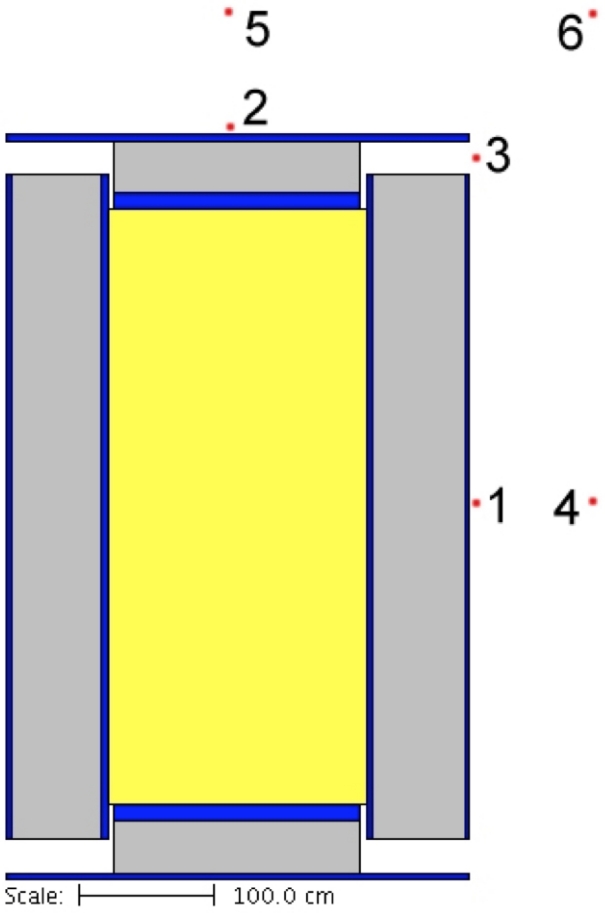
Fig. 4.1.6: Mesh tally showing neutron dose rate (rem/hr) and uncertainties for the analog case and the CADIS case.

4.1.5.2 Dose rates outside of a simple cask

This example problem is a full-size cylindrical cask model, which consists of an inner steel liner, a thick section of concrete, and an outer steel cover. This problem is intended to be used as a tool to teach users how to build MAVRIC input files. This is not a completely realistic shipping cask; it has been simplified greatly for this purpose. The goal of this example it to show how to quickly calculate neutron and photon does rates at six points outside of the cask, including in front of the vent port.

Geometry and materials

The simple model of a cask is shown in Fig. 4.1.7. Vent ports at the top and bottom of the cask are modeled as void all of the way around the cask. The interior of the cask was modeled using materials from about 20 typical pressurized water reactor (PWR) fuel assemblies (including the UO2, Zr, Fe, Ni, Cr, Sn, and other constituents), homogenized over the interior volume. The total mass of the fuel/assembly hardware in this region is 10.6 metric tonnes. Separate end regions of the assemblies are not modeled in this simple example. Also note that the fuel material is based on fresh fuel, not spent fuel with its hundreds of fission products.



Model Specifics

Materials
fuel: UO_2, Zr, others, ρ=0.9137 g/cm^3
stainless steel :SCALE ss304, ρ=7.94 g/cm^3
concrete :SCALE orconcrete, ρ=2.2994 g/cm^3

Regions (r_{min}, r_{max}, z_{min}, z_{max}, all in cm)
yellow: 0, 95, -228.6, 228.6; fuel
blue: 95, 100, -255.2 255.2; inner steel liner
blue: 0, 90, 228.6, 240.6; inner steel liner
blue: 0, 90, -240.6, -228.6; inner steel liner
gray: 100,168, -255.2, 255.2; concrete
gray: 0, 90, 240.6, 280.6; concrete
gray: 0, 90, -280.6, -240.6; concrete
blue: 168, 170, -255.2 255.2; outer steel liner
blue: 0, 170, 280.6.2, 285.6; outer steel liner
blue: 0, 170, -285.6, -280.6.2; outer steel liner

Detector locations (coordinates in cm)
1: 180, 0, 0; midplane, 10 cm from surface
2: 0, 0, 295.6; on axis, 10 cm from surface
3: 180, 0, 267.9; center of gap, 10 cm from r_{max}
4: 270, 0, 0; midplane, 100 cm from surface
5: 0, 0, 385.6; on axis, 100 cm from surface
6: 270, 0, 385.6; 100 cm from each surface

Fig. 4.1.7: Cask geometry and detector locations.

Sources and responses

Spent fuel from a typical mid-sized PWR was used to determine the source term. ORIGEN was used to deplete a full core (46.1 metric tonnes of uranium, 4.2% enriched, with O, Zr, Fe, Ni, Cr, Sn, and other constituents) to 55,000 MWdays/MTU. The contents of the modeled fuel represent typical values for PWR fuel. ORIGEN then computed the neutron and photon spectra in 27-group and 19-group energy structures for the fuel following a 10-year cooling period after the last irradiation. The total neutron source strength for the cask (1/6 of a full core, or about 20 assemblies) was 8.577×10^9 neutrons/s. The total photon source strength was 7.155×10^{16} photons/s.

Two cases will be done for this example: one for calculating the neutron dose rates from the spent fuel neutrons and the other for calculating the photon dose rates from the spent fuel photons. The source spectra and response functions are shown in Fig. 4.1.8 through Fig. 4.1.11 and listed in Table 4.1.8. Note that in this example, the neutron source shown in Fig. 4.1.8 and Table 4.1.8 is considered the final neutron source: no further neutron multiplication is considered.

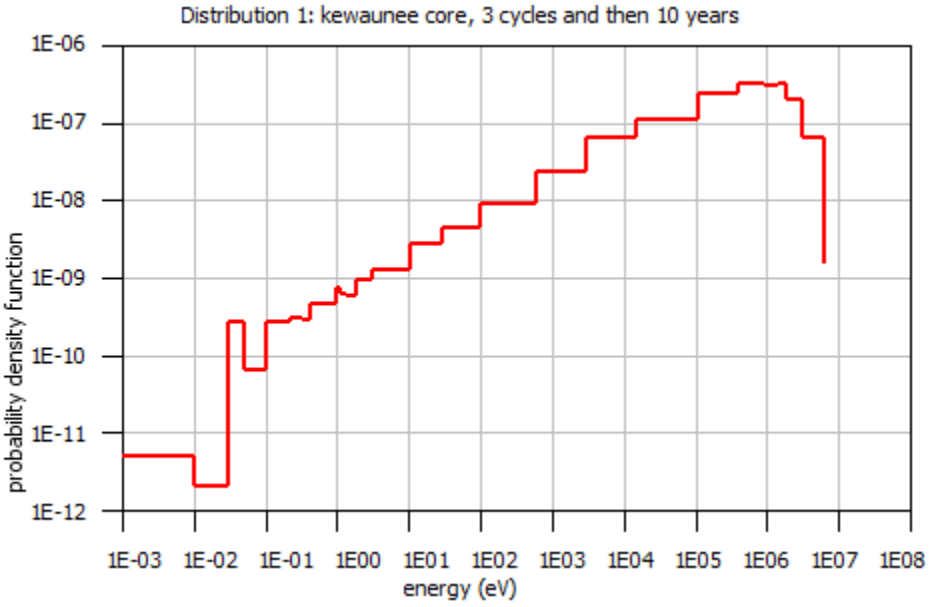


Fig. 4.1.8: Spent fuel neutron source spectrum with strength 8.577×10^9/second.

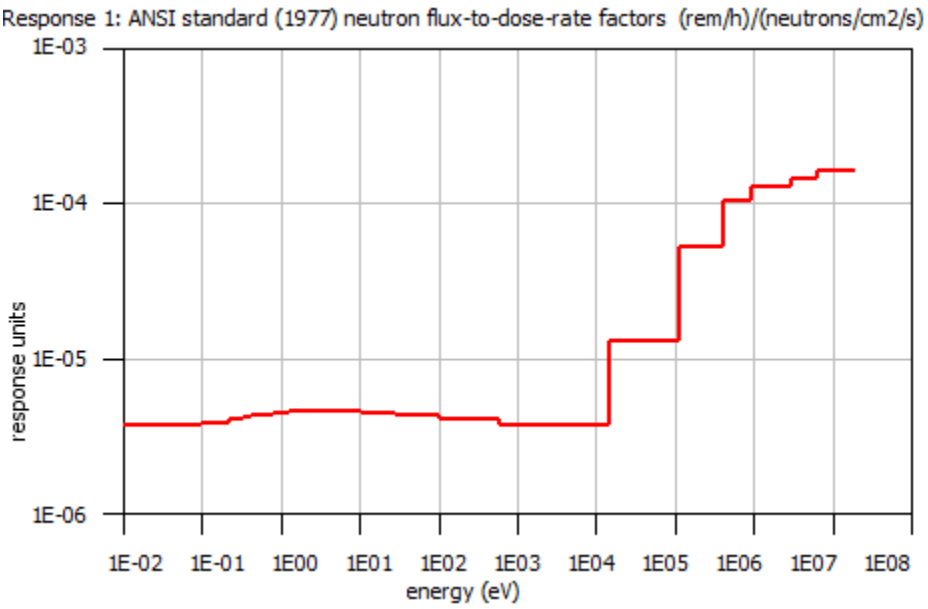


Fig. 4.1.9: ANSI-1977/ flux-to-dose-rate factors (rem/hr)/(neutrons/cm ^{2}/sec).

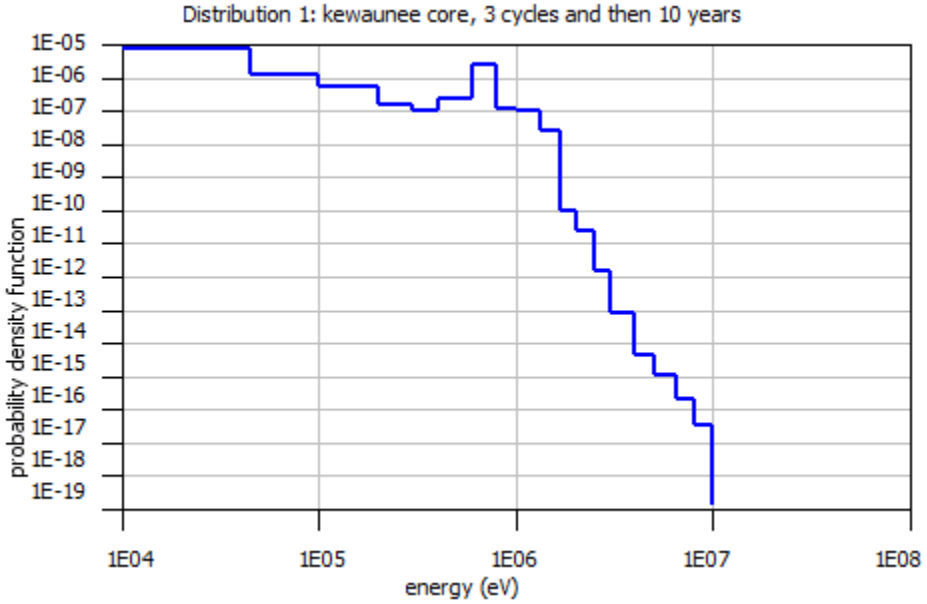


Fig. 4.1.10: Spent fuel photon source spectrum with strength 7.155×10^{16}/second

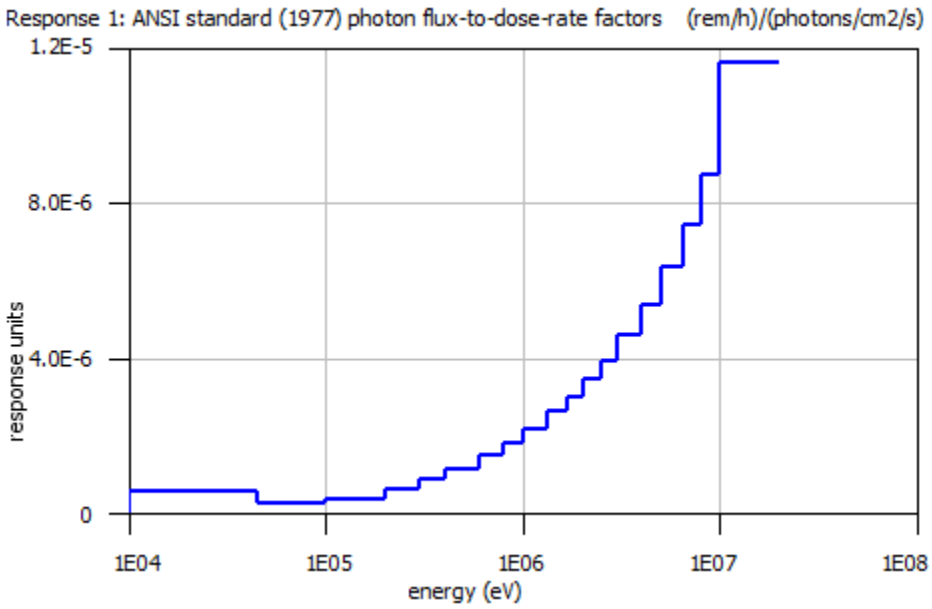


Fig. 4.1.11: ANSI-1977 photon flux-to-dose-rate factors (rem/hr)/(photons/cm ^{2}/sec)

Table 4.1.8: Source and response data using the SCALE 27-group energy structure for neutrons and the 19-group energy structure for photons

Energy (eV)	neutron source	response 9029
2.0000E+07	2.040E-02	1.615E-04
6.3763E+06	2.147E-01	1.445E-04
3.0119E+06	2.365E-01	1.270E-04
1.8268E+06	1.267E-01	1.281E-04
1.4227E+06	1.586E-01	1.298E-04
9.0718E+05	1.587E-01	1.034E-04
4.0762E+05	7.281E-02	5.266E-05
1.1109E+05	1.073E-02	1.286E-05
1.5034E+04	7.688E-04	3.736E-06
3.0354E+03	5.694E-05	3.720E-06
5.8295E+02	4.479E-06	4.009E-06
1.0130E+02	3.148E-07	4.295E-06
2.9023E+01	4.983E-08	4.473E-06
1.0677E+01	9.864E-09	4.566E-06
3.0590E+00	1.117E-09	4.560E-06
1.8554E+00	3.286E-10	4.521E-06
1.3000E+00	1.060E-10	4.487E-06
1.1253E+00	9.203E-11	4.466E-06
1.0000E+00	9.135E-11	4.434E-06
8.0000E-01	1.755E-10	4.332E-06
4.1399E-01	2.590E-11	4.203E-06
3.2500E-01	3.024E-11	4.097E-06
2.2500E-01	3.451E-11	3.840E-06
1.0000E-01	3.269E-12	3.675E-06
5.0000E-02	5.447E-12	3.675E-06
3.0000E-02	4.089E-14	3.675E-06
1.0000E-02	4.916E-14	3.675E-06
1.0000E-05		

Energy (eV)	photon source	response 9504
2.00E+07	1.320E-12	1.162E-05
1.00E+07	7.185E-11	8.745E-06
8.00E+06	3.281E-10	7.460E-06
6.50E+06	1.672E-09	6.351E-06
5.00E+06	4.167E-09	5.400E-06
4.00E+06	8.086E-08	4.602E-06
3.00E+06	7.937E-07	3.952E-06
2.50E+06	1.164E-05	3.459E-06
2.00E+06	3.331E-05	3.013E-06
1.66E+06	8.160E-03	2.620E-06
1.33E+06	3.511E-02	2.194E-06
1.00E+06	2.478E-02	1.827E-06
8.00E+05	4.827E-01	1.515E-06
6.00E+05	4.641E-02	1.160E-06
4.00E+05	9.735E-03	8.705E-07
3.00E+05	1.514E-02	6.219E-07
2.00E+05	5.182E-02	3.708E-07
1.00E+05	7.015E-02	2.688E-07
4.50E+04	2.560E-01	5.933E-07
1.00E+04		

Energies listed are the bin upper energies. Source units are particles/s normalized to a total of 1 particle/s. Response units are (rem/hr)/(particle/cm^2/s).

Analog calculation

The analog model for this problem starts with the problem title and the cross section library name, which in this example is the ENDF/B-VII.0 27 neutron group / 19 photon group library. This is in the SCALE `samples\input` directory as `mavric.caskAnalogn.inp` and `mavric.caskAnalogp.inp`.

```
=mavric
Simplified cask model
v7-27n19g
```

Then the material compositions are listed for fresh fuel, concrete, and steel.

```
read composition
    wtptFuel    1 0.913717475 18         6000  0.00939719    7014  0.00528993
                     8016  9.73397641   13000  0.00715715   14000  0.01031670
                    15000  0.02227505   22000  0.00780567   24000  0.36655141
                    25000  0.01716839   26000  0.72041451   27000  0.00523824
                    28000  0.68955526   40000 15.78990702   41000  0.05130153
                    42000  0.02844690   50118  0.25877903   92235  3.03560962
                    92238 69.24080999
                  1.0 293.0 end
    orconcrete  2 1.0 293.0 end
    ss304       3 1.0 293.0 end
end composition
```

Then the SGGP geometry is listed, with the origin of the coordinate system at the center of the cask.

```
read geometry
    global unit  1
        zcylinder  1   95.0  228.6 -228.6
        zcylinder  2  170.0  255.2 -255.2
        zcylinder  3   90.0  240.6 -240.6
        zcylinder  4   90.0  280.6 -280.6
        zcylinder  5  170.0  280.6 -280.6
        zcylinder  6  170.0  285.6 -285.6
        zcylinder  7   95.0  255.2 -255.2
        zcylinder  8  100.0  255.2 -255.2
        zcylinder  9  168.0  255.2 -255.2
        sphere    10  999.0
        media  1 1   1          vol=1.29629E+07
        media  3 1   8 -7       vol=1.56338E+06
        media  2 1   9 -8       vol=2.92216E+07
        media  3 1   2 -9       vol=1.08394E+06
        media  3 1   3 -1       vol=6.10726E+05
        media  2 1   4 -3       vol=2.03575E+06
        media  3 1   6 -5       vol=9.07920E+05
        media  0 1   5 -4 -2    vol=3.31953E+06
        media  0 1   7 -4 -1    vol=1.54598E+05
        media  0 1   10 -6      vol=4.12429E+09
    boundary 10
end geometry
```

The definitions block contains locations, response functions, grid geometries, and a distribution used by the source input block. For the neutron source/neutron dose problem, the definitions block is listed below.

```
read definitions
    location 1   position 180.0 0.0   0.0    end location
    location 2   position   0.0 0.0 295.6    end location
    location 3   position 180.0 0.0 267.9    end location
    location 4   position 270.0 0.0   0.0    end location
    location 5   position   0.0 0.0 385.6    end location
    location 6   position 270.0 0.0 385.6    end location
    response 1  specialDose=9029  end response
```

(continues on next page)

(continued from previous page)

```
    distribution 1
        title="kewaunee core, 3 cycles and then 10 years"
        neutronGroups
        truePDF  2.040E-02 2.147E-01 2.365E-01 1.267E-01 1.586E-01
                 1.587E-01 7.281E-02 1.073E-02 7.688E-04 5.694E-05
                 4.479E-06 3.148E-07 4.983E-08 9.864E-09 1.117E-09
                 3.286E-10 1.060E-10 9.203E-11 9.135E-11 1.755E-10
                 2.590E-11 3.024E-11 3.451E-11 3.269E-12 5.447E-12
                 4.089E-14 4.916E-14 end
    end distribution
end definitions
```

The neutron source from the spent fuel is then listed.

```
read sources
    src 1
        title="1/6 of kewaunee core, ~ 0.25 Ci"
        strength=8.577E+09
        neutrons
        zCylinder 95.0  228.6 -228.6
        eDistributionID=1
    end src
end sources
```

Six point detectors are used to evaluate dose rates radially, axially, and near the vent port.

```
read tallies
    pointDetector 1  locationID=1  responseID=1  end pointDetector
    pointDetector 2  locationID=2  responseID=1  end pointDetector
    pointDetector 3  locationID=3  responseID=1  end pointDetector
    pointDetector 4  locationID=4  responseID=1  end pointDetector
    pointDetector 5  locationID=5  responseID=1  end pointDetector
    pointDetector 6  locationID=6  responseID=1  end pointDetector
end tallies
```

The Monte Carlo parameters were tailored for the neutron problem to be 1-minute batches on a 2 GHz Linux computer. For the photon problem, the number per batch would be 91,000 for 1-minute batches.

```
read parameters
    randomSeed=8655745280030001
    perBatch=25400 batches=600
    fissionMult=0    noPhotons
end parameters
```

No biasing is specified, which will use the default weight window target value of 1 for every energy group in every region. To allow the neutrons to penetrate into the cask wall before being rouletted, a larger window ratio is used, making the lower weight window bound 0.01.

```
read biasing
    windowRatio=199.0
end biasing
```

The Monaco input is then ended.

```
end data
end
```

For the photon source/photon dose rate problem, the definitions block would instead contain a photon flux-to-dose-rate response function and the energy distribution for the source.

```
read definitions
    response 1
        specialDose=9504
    end response
    ...
    distribution 1
        title="kewaunee core, 3 cycles and then 10 years"
        photonGroups
        truePDF 1.320E-12 7.185E-11 3.281E-10 1.672E-09 4.167E-09
                8.086E-08 7.937E-07 1.164E-05 3.331E-05 8.160E-03
                3.511E-02 2.478E-02 4.827E-01 4.641E-02 9.736E-03
                1.514E-02 5.182E-02 7.015E-02 2.560E-01       end
    end distribution
end definitions
```

The sources block would contain the photon source information.

```
read sources
    src 1
        title="1/6 of kewaunee core, ~ 2e6 Ci"
        strength=7.155e+16
        photons
        zCylinder 95.0  228.6 -228.6
        eDistributionID=1
    end src
end sources
```

Each of the two analog problems in the `samples\input` directory will run for about 10 minutes. In this time, no meaningful results will be generated due to the difficulty of the problem. Analog results for each case running 110 hr are listed in Table 4.1.9 for the neutron source/neutron dose problem, while results for the photon problem are listed in Table 4.1.10. Note that after 110 hr, some of the relative uncertainties in the point detector tallies are still quite high, and only one of the six tallies in each problem passed all of the statistical checks. Fig. 4.1.12 is the convergence plot for the neutron dose rate at point detector 1, showing that the tally is not well converged and that some batches contain rare events that change the tally value a great deal.

Table 4.1.9: Analog Monaco results for the simplified cask model-neutron source/neutron dose rate.

detector	Dose Rate (rem/hr)	relative uncertainty
1	8.78E-04	0.1853
2	7.35E-03	0.0408
3	1.54E-02	0.0124
4	4.47E-04	0.0311
5	1.36E-02	0.0058
6	2.92E-03	0.0073

Table 4.1.10: Analog Monaco results for the simplified cask model-photon source/photon dose rate.

detector	Dose Rate (rem/hr)	relative uncertainty
1	2.66E-02	0.2713
2	3.05E-01	0.0787
3	2.97E+00	0.0199
4	2.65E-02	0.1345
5	3.26E+00	0.0133
6	3.30E-01	0.0244

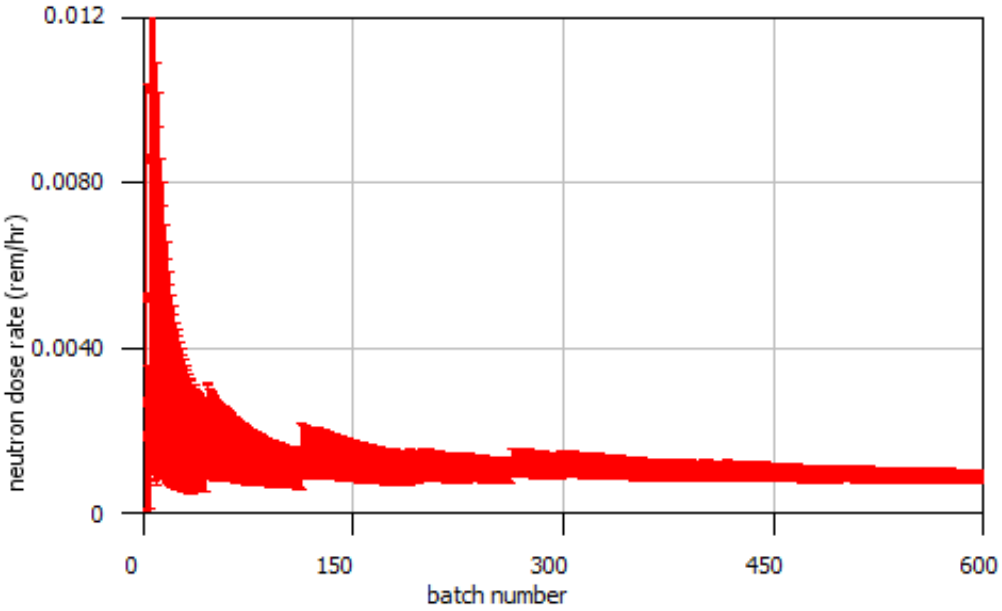


Fig. 4.1.12: Convergence plot for the neutron dose rate at point detector 1 (error bars show the 1-sigma tally uncertainties).

SAS4 calculations

Calculations for these two problems were also done using the SAS4 sequence in SCALE 5.1. SAS4 was specifically designed for cask geometries and used a one-dimensional discrete-ordinates calculation (either radially or axially) to determine weight windows. Results for the neutron problem are shown in Table 4.1.11, and results for the photon problem are shown in Table 4.1.12. Note that SAS4 using radial biasing is only expected to do well for the two radial point detector locations. Similarly, only the two axial point detectors are expected to do well when using axial biasing. SAS4 was not intended to do well for the points near the vent port, but the results using the axial biasing seem reasonable.

Table 4.1.11: SAS4 results using radial biasing (361 minutes) and axial biasing (361 minutes), for the simplified cask model-neutron source/neutron dose rate

	Radial Biasing		Axial Biasing	
detector	Dose Rate (rem/hr)	relative uncertainty	Dose Rate (rem/hr)	relative uncertainty
1	7.67E-04	0.0081	1.32E-05	0.4453
2	2.67E-02	0.8623	7.80E-03	0.0040
3	1.27E-02	0.1425	1.53E-02	0.0079
4	4.54E-04	0.0079	2.34E-04	0.6780
5	1.43E-02	0.1349	1.35E-02	0.0044
6	2.81E-03	0.1274	2.86E-03	0.0052

Table 4.1.12: SAS4 results using radial biasing (361 minutes) and axial biasing (361 minutes), for the simplified cask model-photon source/photon dose rate

	Radial Biasing		Axial Biasing	
detector	Dose Rate (rem/hr)	relative uncertainty	Dose Rate (rem/hr)	relative uncertainty
1	4.54E-02	0.0118	7.07E-04	0.5375
2	6.74E-02	0.3295	2.45E-01	0.0630
3	5.63E-01	0.3818	2.07E+00	0.2289
4	2.76E-02	0.0039	2.48E-03	0.4570
5	1.85E+00	0.1699	2.70E+00	0.0664
6	1.46E-01	0.1493	2.73E-01	0.0838

Calculations using CADIS

In the analog calculations, the dose rates at all six points could be calculated at the same time. With MAVRIC and using CADIS, the importance map will optimize the transport of particles towards only the selected detector. Hence, each detector will have a separate calculation with an importance map tailored to reduce the variance for only that detector. Calculations for close detectors could be performed at the same time. For example, detectors 1 and 4 both need to push particles out of the cask in the positive x direction, towards the z=0 plane. In this example, all six detectors will use separate importance maps.

For the importance map, in the input, the user lists what planes to use for the adjoint discrete-ordinates calculation. These planes define cells, which are treated as homogenous parallelpipeds by Denovo, made of a macro material corresponding to a mixture of materials that are in the cell in the true geometry. Users should try to bound as many materials as possible with their selection of mesh planes. More mesh planes should be used where the importance (adjoint flux) varies quickly, such as near the adjoint sources (the detector positions). It is also important to have planes on the true source bounding box.

In this example problem, different sets of mesh planes will be used for the different detector positions. For detector positions 1 and 4, the mesh planes are shown in Fig. 4.1.13 and Fig. 4.1.14. Note that there are more planes closer to the detectors. Also note that in the z dimension, it is quite easy to place mesh planes at every material boundary, but it is a bit more difficult to do so in the x and y dimensions due to the curved surfaces. Users need not worry about getting things perfect-an approximate importance map can still reduce Monte Carlo variances a great deal. The meshes used for detector positions 2/5 and positions 3/6 are also shown in Fig. 4.1.13 and Fig. 4.1.14. Mesh parameters are listed in Table 4.1.13, and the mesh planes are listed in Table 4.1.14.

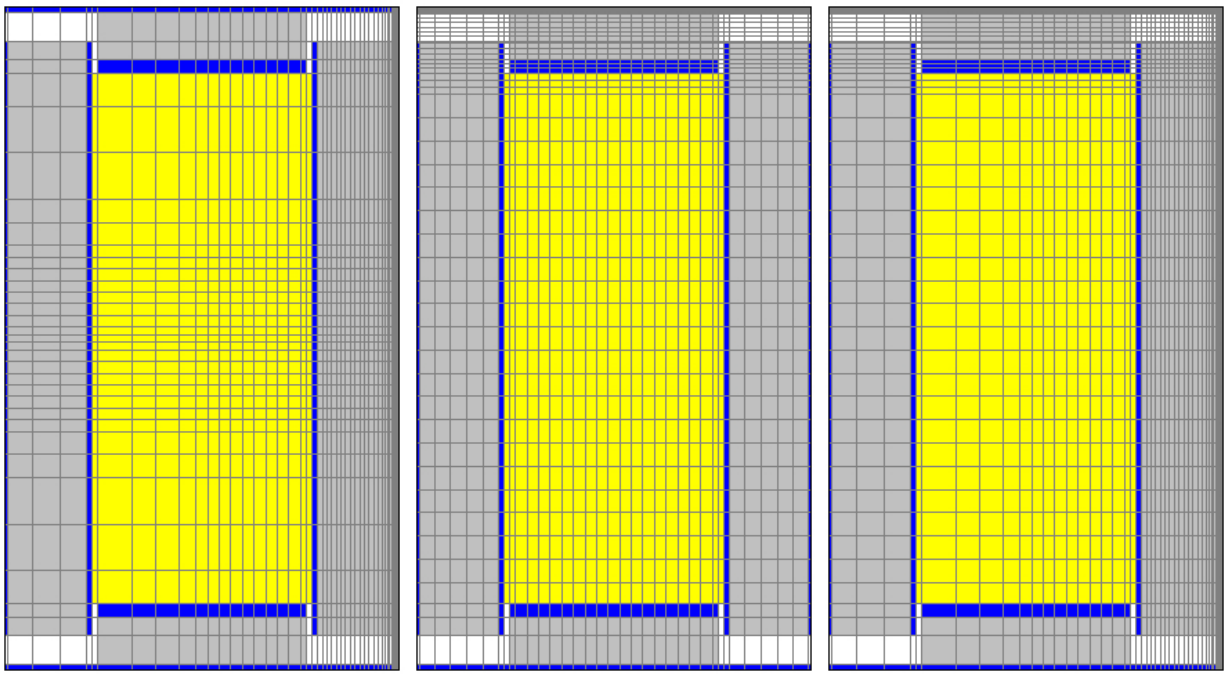


Fig. 4.1.13: Importance map mesh planes in the x and z dimensions for detector positions 1/4, 2/5, and 3/6.

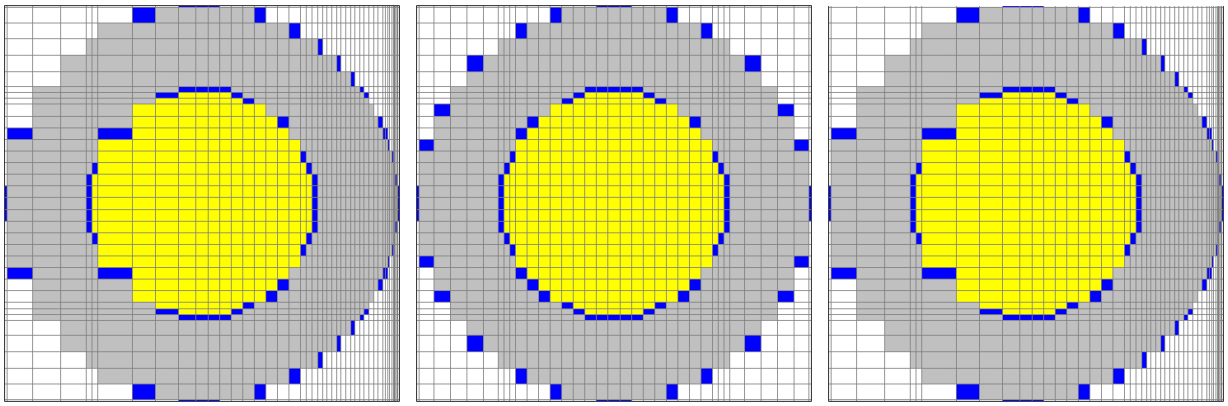


Fig. 4.1.14: Importance map mesh planes in the x and y dimensions for detector positions 1/4, 2/5, and 3/6.

Table 4.1.13: Mesh parameters

Detector position	Number	of	cells	Total cells
	x	y	z	
1/4	46	35	35	56350
2/5	35	35	49	60025
3/6	46	35	49	78890

Table 4.1.14: List of the various sets of mesh planes used for the importance calculations for six different point detectors

x 1,3,4,6		*x* 2,5		*y* 1,2,3,4,5,6		*z* 1,4		*z* 2,3,5,6	
-170	108	-170	5	-170	5	-285.6	3	-285.6	190
-168	112	-168	15	-168	15	-280.6	10	-280.6	210
-146	116	-155	25	-155	25	-255.2	20	-255.2	216.2
-122	120	-141	35	-141	35	-240.6	30	-240.6	222.4
-100	124	-127	45	-127	45	-228.6	40	-228.6	228.6
-95	128	-113	55	-113	55	-200	50	-210	232.6
-90	132	-100	65	-100	65	-160	60	-190	236.6
-60	136	-95	75	-95	75	-120	70	-170	240.6
-40	140	-90	85	-90	85	-100	80	-150	245.1
-20	144	-85	90	-85	90	-80	100	-130	249.7
-5	148	-75	95	-75	95	-70	120	-110	254.2
5	152	-65	100	-65	100	-60	160	-90	255.2
15	156	-55	113	-55	113	-50	200	-70	256.2
25	158	-45	127	-45	127	-40	228.6	-50	260.1
35	160	-35	141	-35	141	-30	240.6	-30	264
45	162	-25	155	-25	155	-20	255.2	-10	267.9
55	164	-15	168	-15	168	-10	280.6	10	271.8
65	165	-5	170	-5	170	-3	285.6	30	275.7
75	166							50	279.6
85	167							70	280.6
90	168							90	281.6
95	169							110	282.6
100	170							130	283.6
104								150	284.6
								170	285.6

MAVRIC input files

With two sources and six detectors, this example problem will require 12 separate input files. Starting with the two input files for the analog calculations, these 12 input files will share most of the same features and will differ only in blocks related to the importance map calculation: the location of the adjoint source and the planes used in the grid geometry.

To change the input for the neutron problem from an analog calculation to one using CADIS, the user first adds the mesh planes for the discrete-ordinates calculation as a grid geometry to the definitions block. This set of planes is tailored for the vent port direction toward detectors 3 and 6.

```
gridGeometry 3
        title="for importance map for detectors 3,6"
        xplanes -170 -168 -146 -122 -100
                -95 -90 -60 -40 -20 -5
                5 15 25 35 45 55 65 75 85 90 95 100
                104 108 112 116 120 124 128 132 136 140 144 148 152
                156 158 160 162
                164 165 166 167
                168 169 170  end
        yplanes -170 -168 -155 -141 -127 -113 -100
                -95 -90 -85 -75 -65 -55 -45 -35 -25 -15 -5
                5 15 25 35 45 55 65 75 85 90 95 100
                113 127 141 155 168 170  end
        zplanes -285.6 -280.6 -255.2 -240.6 -228.6 -210
                -190 -170 -150 -130 -110 -90 -70 -50 -30 -10
                10 30 50 70 90 110 130 150 170 190
                210    216.2 222.4
                228.6  232.6 236.6
                240.6  245.1 249.7 254.2
                255.2  256.2 260.1 264 267.9 271.8 275.7 279.6
                280.6  281.6 282.6 283.6 284.6 285.6  end
end gridGeometry
```

To help the mesh-based biased source represent the true source, "mixture=1" can be added to the source definition. This will ensure that particles sampled from the mesh source that are not in the fuel are rejected. Then an importance map block replaces the standard biasing block. In this case, the importance map will optimize the flow of particles to location 3 (where point detector 3 is defined).

```
read importanceMap
    adjointSource 1
        locationID=3
        responseID=1
    end adjointSource
    gridGeometryID=3
    macromaterial
        mmTolerance=0.01
    end macromaterial
end importanceMap
```

A mesh tally could be added with the following in the tallies block.

```
meshTally 1
      title="Shows how importance map changes the transport of particles"
      gridGeometryID=3
      responseID=1
end meshTally
```

The above mesh tally uses the same grid geometry as the CADIS calculations, but a different grid (or grids) could be used. The files `mavric.caskCADISn.inp` and `mavric.caskCADISp.inp` are available in the SCALE `samples\input` directory. These are for calculating the dose rates at detector position 3, but they

can be modified for the other five positions (by changing the geometry grid planes and the adjoint source location).

Neutron source/neutron response results

The above MAVRIC input file first performed the discrete-ordinates calculation to determine the adjoint flux from detector position 3. The adjoint Denovo flux file (*.adjoint.dff) produced can be viewed using the Mesh File Viewer and is shown in Fig. 4.1.15 for several of the neutron energy groups.

MAVRIC then combined a mesh representation of the true source (space and energy) with the adjoint fluxes to create the importance map and mesh-based biased source. These are shown in Fig. 4.1.16 for the fifth neutron group, covering the energy range of 0.9 to 1.4 MeV. Notice how the most important region (lowest target weights) is right around the vent port near detector position 3. This is something we know qualitatively, but quantitative values for exactly how the importance changes with space and energy are difficult to guess. Also notice the "consistent" part of CADIS—the source particles—are born with a weight that matches the target weight for the position they are born into. The biased source sampling distribution is depicted in Fig. 4.1.17, showing how the source particles nearest to detector 3 will be sampled more often.

The biased source distribution and the importance map are then used by Monaco to compute the dose equivalent rate response at detector 3.

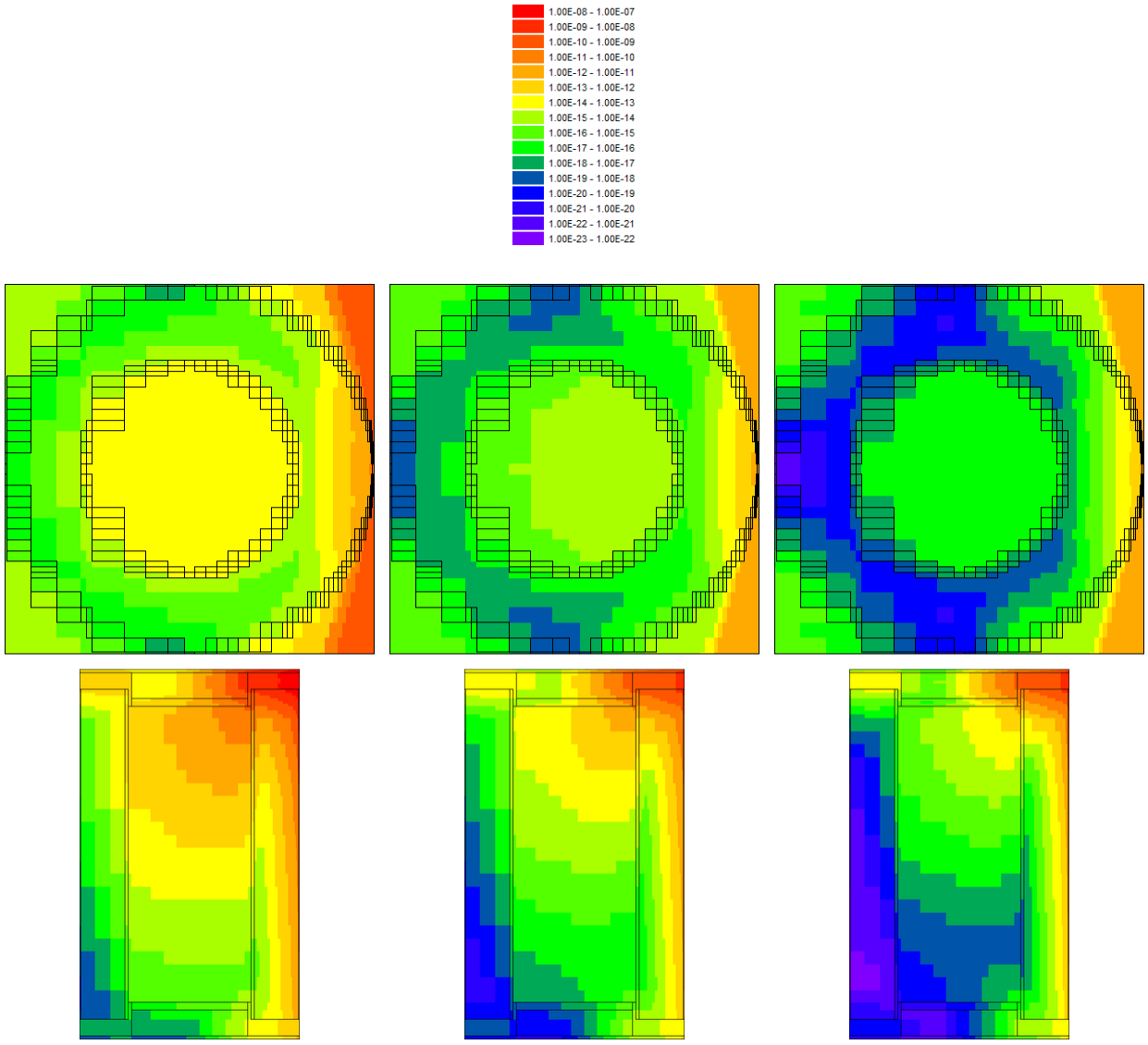


Fig. 4.1.15: Adjoint neutron fluxes (cm^2/s) for groups 5 (0.9-1.4 MeV), 10 (0.58-3.0 keV), and 19 (0.8-1 eV) calculated by Denovo.

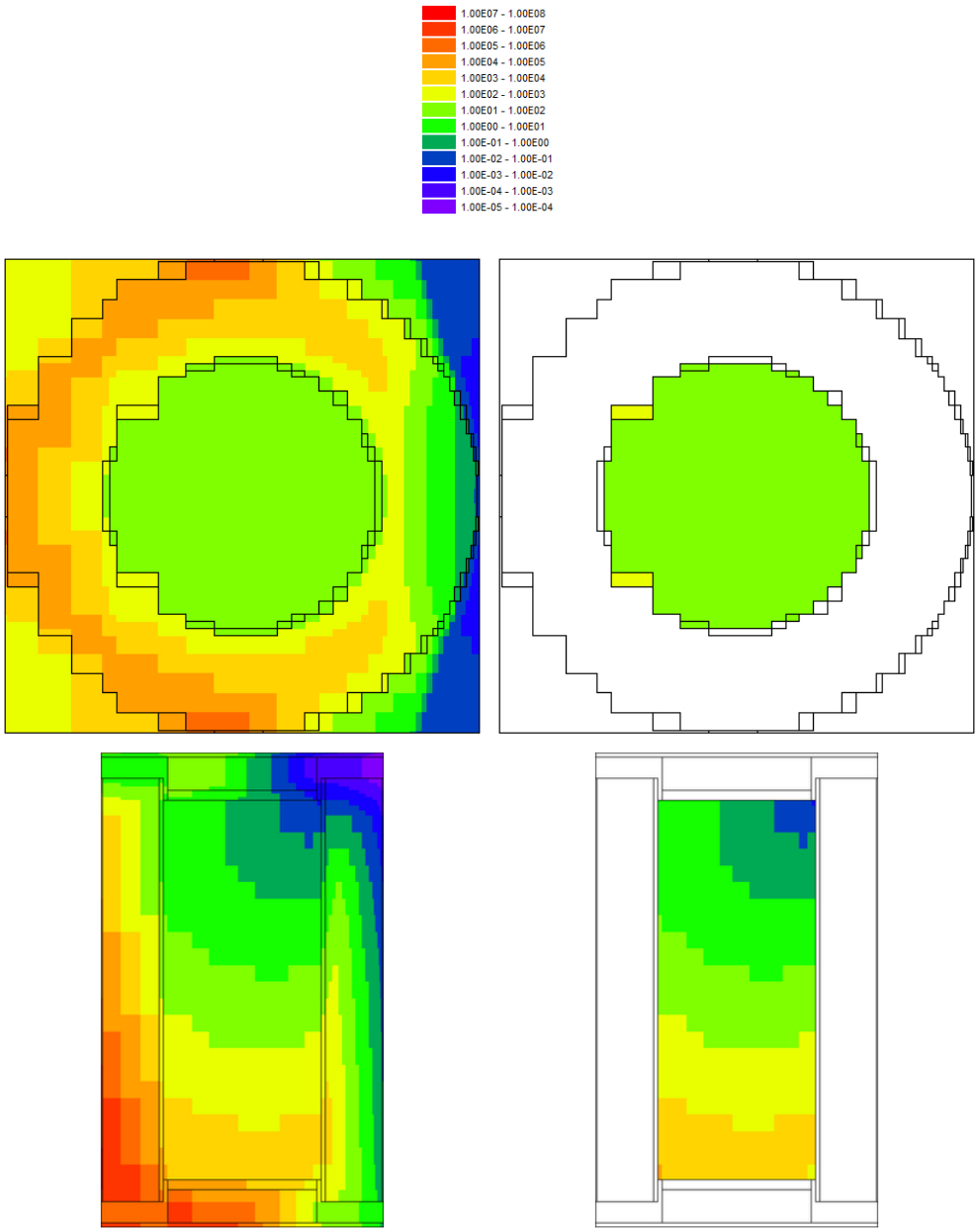


Fig. 4.1.16: Neutron target weights from the importance map and source weights (at birth) for neutron group 5 (0.9 to 1.4 MeV).

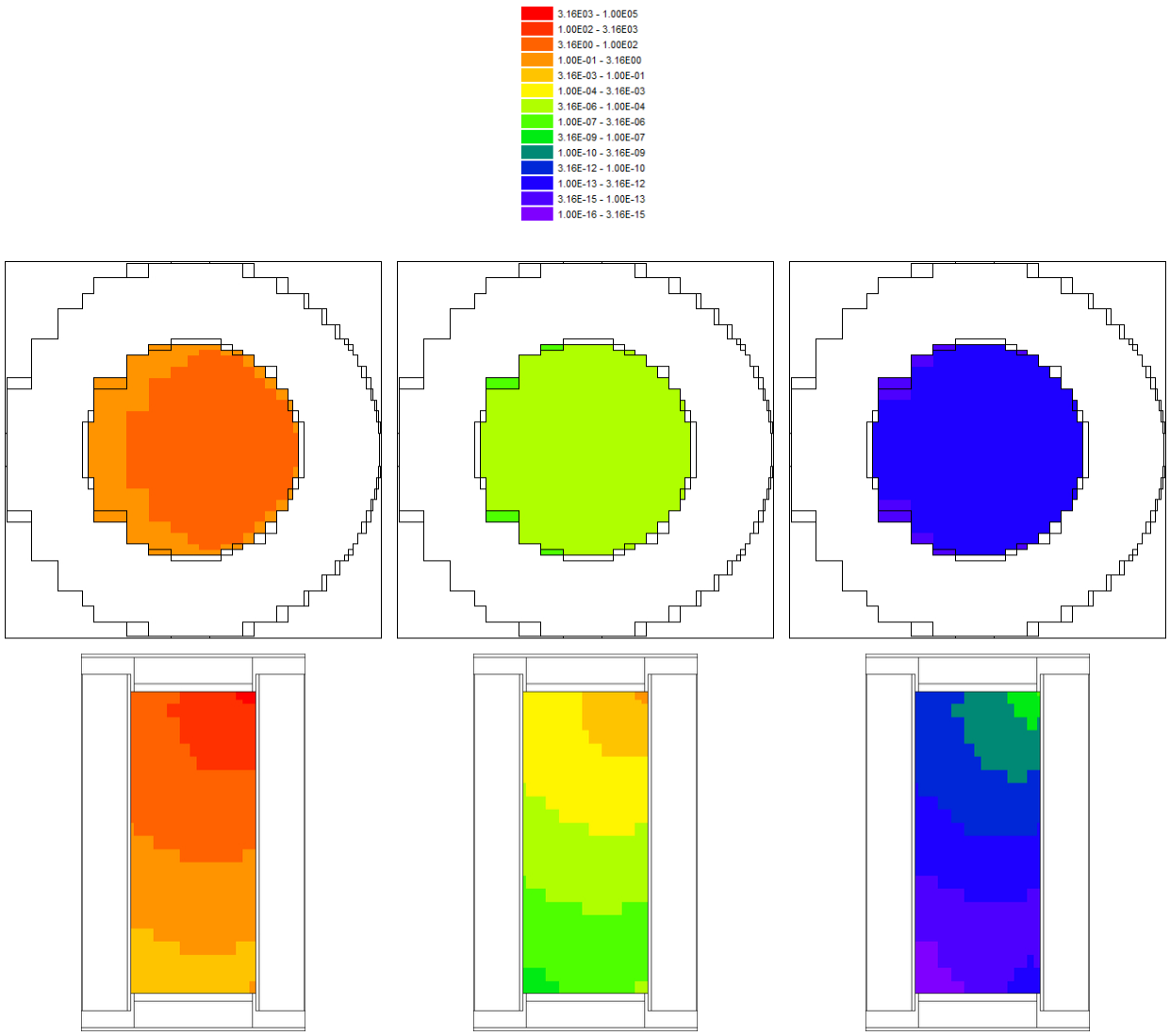


Fig. 4.1.17: Biased source sampling probability (neutrons/cm3) for neutron groups 5 (0.9-1.4 MeV), 10 (0.58-3.0 keV), and 19 (0.8-1 eV).

The results for all six neutron cases, each using their own importance map and biased source, are shown in Table 4.1.15.

This example shows that MAVRIC using CADIS obtains the correct answer much faster than the analog calculations. This is shown with a comparison to the results from the analog Monaco and SAS4 calculations, all of which are listed in Table 4.1.16.

To account for the time it takes to achieve a given level of uncertainties, the calculation figure-of-merit—FOM=1/time/(relative uncertainty)^2—can be calculated for each of the codes. The ratios of each code FOM to the FOM of analog Monaco (speedup) are listed in Table 4.1.17 to show how much faster MAVRIC and SAS4 are compared to analog Monaco. The FOMs for MAVRIC include the Denovo calculation times. The FOMs for analog Monaco and SAS4 were modified to account for calculating all six detectors at once.

Table 4.1.15: Final MAVRIC results (rem/hr) for each point detector in the neutron source/neutron dose rate problem

	Time (minutes)		Dose Rate	relative
detector	Denovo	Monaco	(rem/hr)	uncertainty
1	6.9	131	7.65E-04	0.0081
2	7.8	64	7.83E-03	0.0030
3	9.8	65	1.54E-02	0.0029
4	6.4	120	4.57E-04	0.0029
5	7.3	64	1.36E-02	0.0028
6	9.5	65	2.93E-03	0.0023

Table 4.1.16: Comparison of neutron dose rates (rem/hr) to other SCALE codes

	Analog Monaco	SAS4 radial	SAS4 axial	MAVRIC
detector	6595 min	360 min	361 min	556 min
1	8.78E-04 ± 19%	7.67E-04 ± 0.8%	1.32E-05 ± 45%	7.65E-04 ± 0.8%
2	7.35E-03 ± 4.1%	2.67E-02 ± 86%	7.80E-03 ± 0.4%	7.83E-03 ± 0.3%
3	1.54E-02 ± 1.2%	1.27E-02 ± 14%	1.53E-02 ± 0.8%	1.54E-02 ± 0.3%
4	4.47E-04 ± 3.1%	4.54E-04 ± 0.8%	2.34E-04 ± 68%	4.57E-04 ± 0.3%
5	1.36E-02 ± 0.6%	1.43E-02 ± 13%	1.35E-02 ± 0.4%	1.36E-02 ± 0.3%
6	2.92E-03 ± 0.7%	2.81E-03 ± 12.7%	2.86E-03 ± 0.5%	2.93E-03 ± 0.2%

Table 4.1.17: Ratio of the figure-of-merit (speed-up) of MAVRIC and SAS4 compared to analog Monaco

detector	Monaco	SAS4 radial	SAS4 axial	MAVRIC
1	1	9498	3.2	4178
2	1	0.0	1874	2858
3	1	0.1	45	270
4	1	287	0.0	1002
5	1	0.0	32	65
6	1	0.1	35.8	146

Photon source/photon response results

The results for the photon source/photon response are similar to the results of the neutron source/neutron response. For the MAVRIC calculation using the photon importance map made from an adjoint source located at detector position 3, Fig. 4.1.18 details the adjoint photon flux. Fig. 4.1.19 compares the target weights from the importance map and the source birth weights. Fig. 4.1.20 shows the distribution of the sampled source positions from the biased source.

Table 4.1.18 shows the results from all six photon MAVRIC runs, each using its own importance map and biased source.

The MAVRIC results of the photon problem compare well against SAS4 and analog Monaco, as shown in Table 4.1.19 and Table 4.1.20.

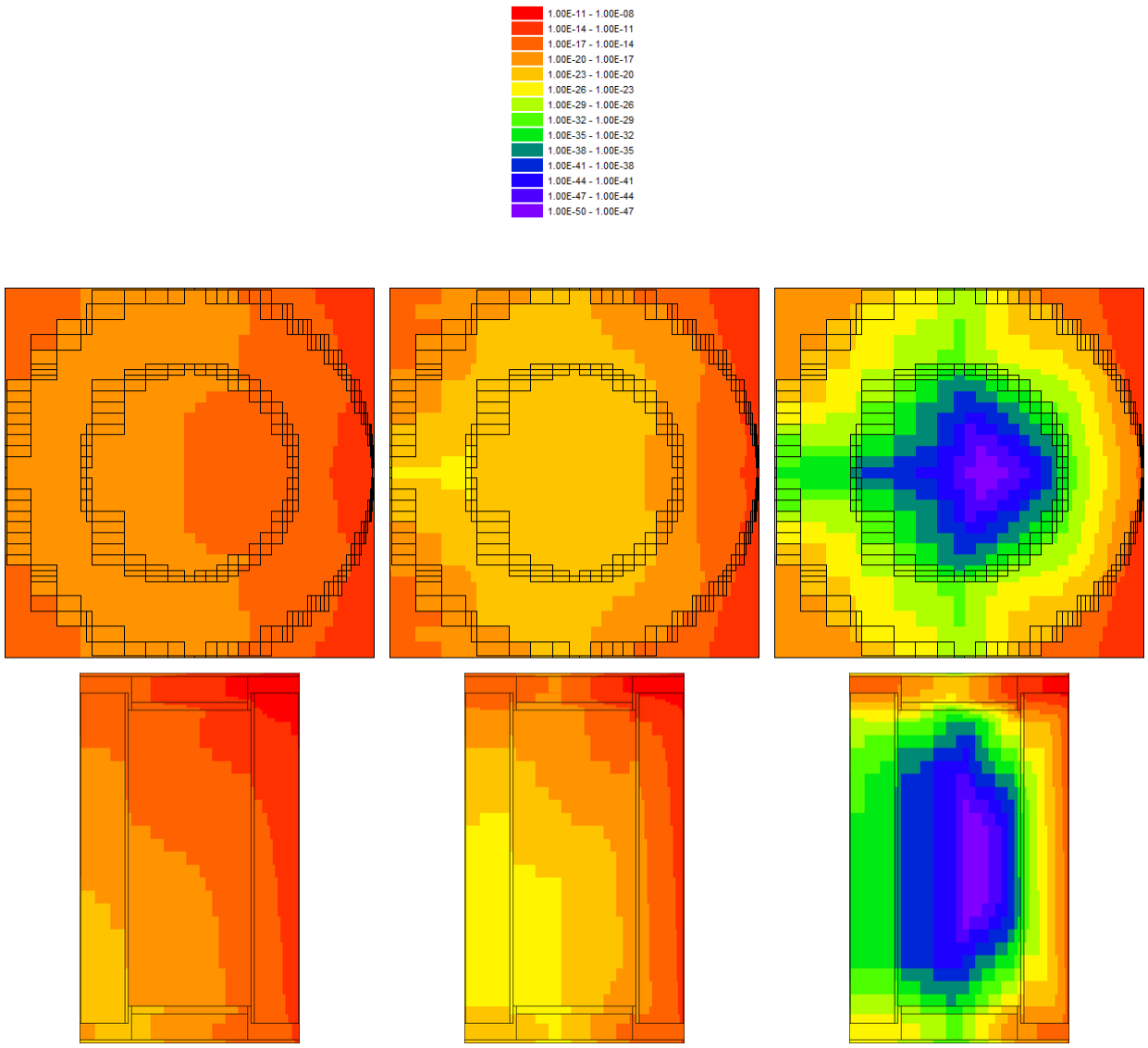


Fig. 4.1.18: Adjoint photon fluxes (/cm: sup:*2*/s) for groups 2 (8-10 MeV), 2 (0.8-1.0 MeV), and 18 (45-00 keV) calculated by Denovo.

Table 4.1.18: Final MAVRIC results (rem/hr) for each point detector in the photon source/photon dose rate problem

detector	Monaco	SAS4 radial	SAS4 axial	MAVRIC
1	1	9498	3.2	4178
2	1	0.0	1874	2858
3	1	0.1	45	270
4	1	287	0.0	1002
5	1	0.0	32	65
6	1	0.1	35.8	146

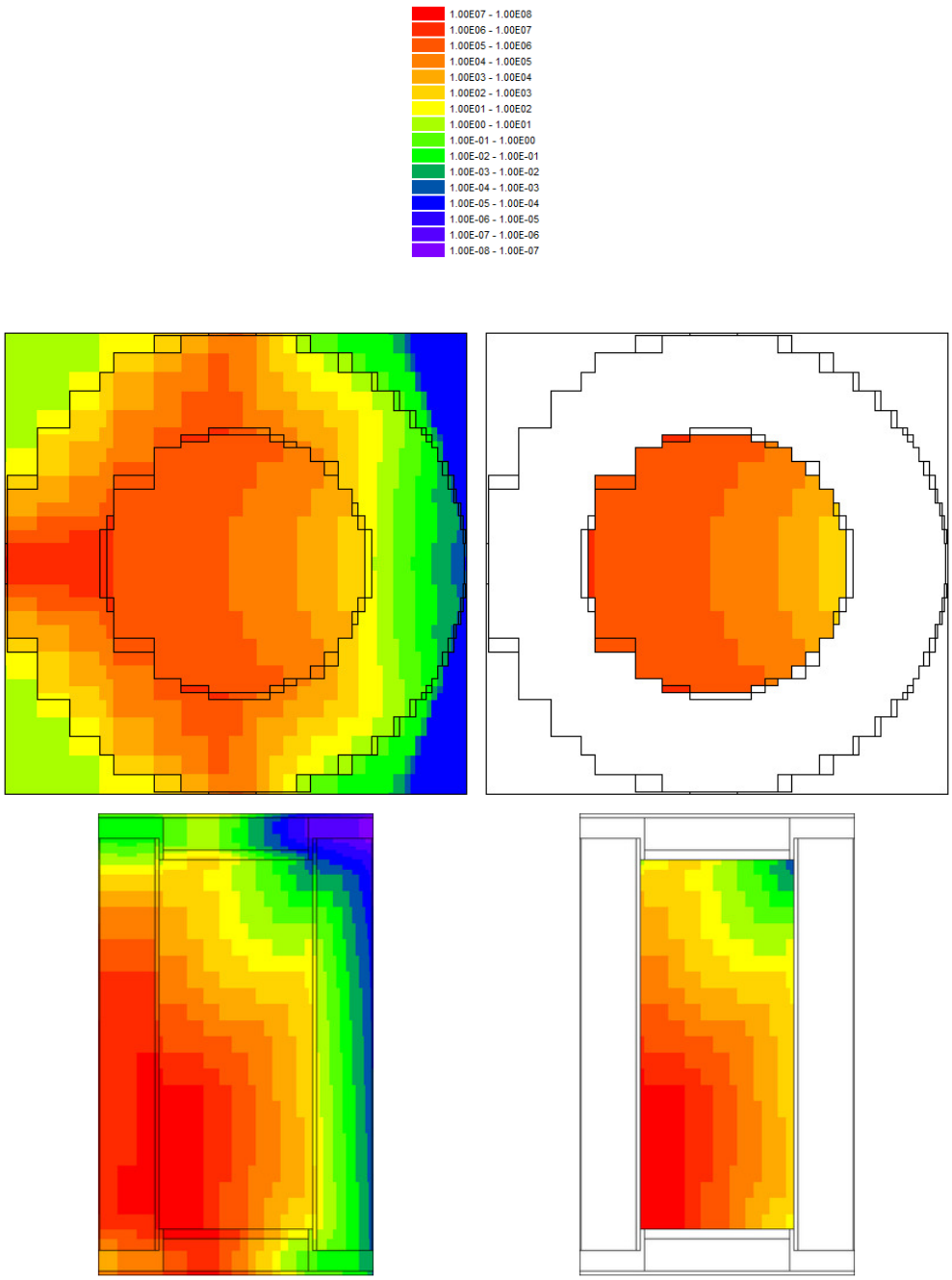


Fig. 4.1.19: Photon target weights from the importance map and source weights (at birth) for photon group 12 (0.8-1.0 MeV).

Table 4.1.19: Comparison of the photon dose rates (rem/hr) to other SCALE codes

detector	Analog Monaco 6678 min	SAS4 radial 361 min	SAS4 axial 361 min	MAVRIC 520 min
1	2.66E-02 ± 27%	4.54E-02 ± 1.2%	7.07E-04 ± 54%	5.02E-02 ± 0.2%
2	3.05E-01 ± 7.9%	6.74E-02 ± 33%	2.45E-01 ± 6.3%	3.03E-01 ± 1.8%
3	2.97E+00 ± 2.0%	5.63E-01 ± 38%	2.07E+00 ± 22.9%	2.89E+00 ± 1.0%
4	2.65E-02 ± 13.4%	2.76E-02 ± 0.4%	2.48E-03 ± 46%	3.18E-02 ± 0.2%
5	3.26E+00 ± 1.3%	1.85E+00 ± 17%	2.70E+00 ± 6.6%	3.07E+00 ± 1.6%
6	3.30E-01 ± 2.4%	1.46E-01 ± 14.9%	2.73E-01 ± 8.4%	3.14E-01 ± 1.4%

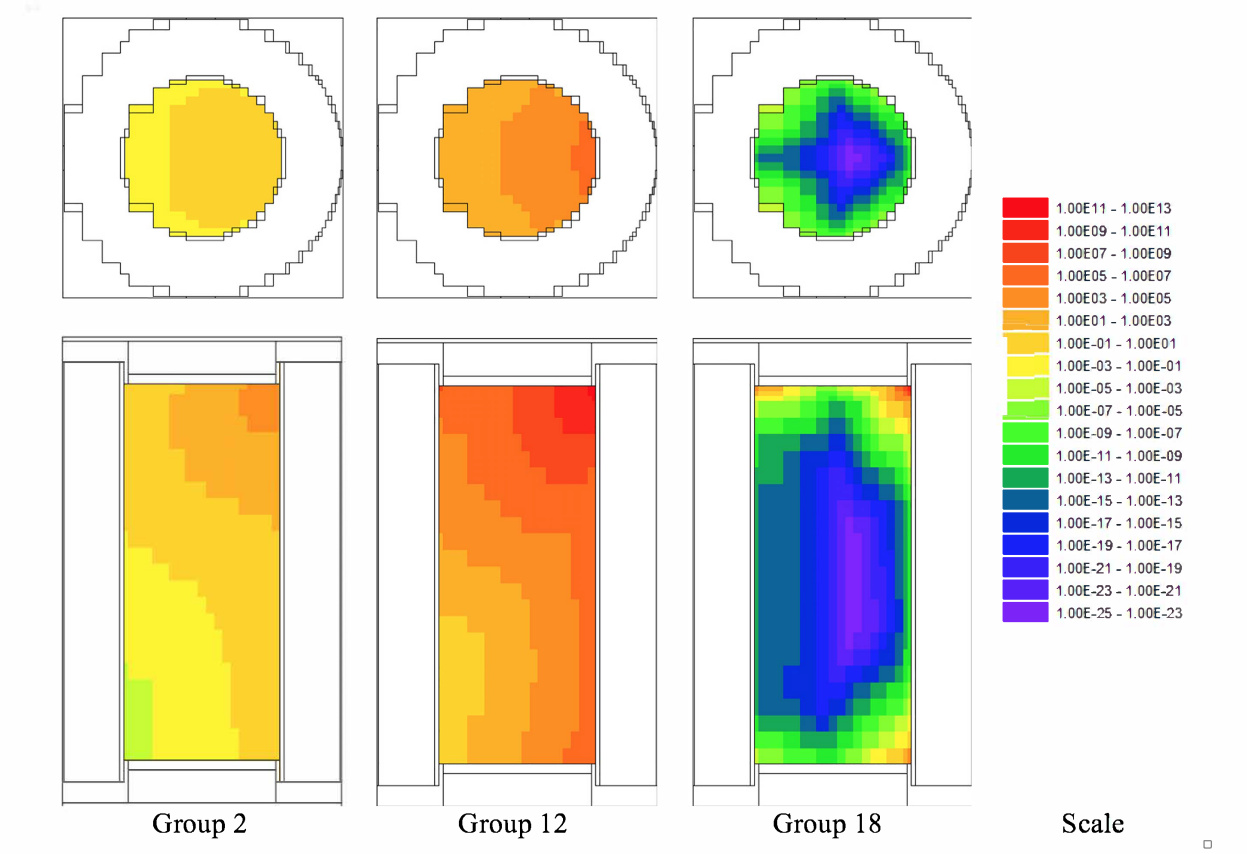


Fig. 4.1.20: Biased source sampling probability (photons/cm3) for groups 2 (8-10 MeV), 12 (0.8-1.0 MeV), and 18 (45-100 keV).

Table 4.1.20: Ratio of the FOM (speed-up) of MAVRIC and SAS4 compared to analog Monaco

detector	Monaco	SAS4 radial	SAS4 axial	MAVRIC
1	1	9759	4.7	114619
2	1	1.1	29	325
3	1	0.1	0	60
4	1	21570	1.6	26329
5	1	0.1	1	13
6	1	0.5	1.6	53

4.1.5.3 Gamma-ray litho-density logging tool using FW-CADIS

Consider a simple model of a gamma-ray litho-density tool [MAVRIC-GV91, MAVRIC-WH98] used in well-logging studies, shown in Fig. 4.1.21. This model uses a 10 cm diameter tool made of iron (with a tungsten density) in a 20 cm borehole filled with water. The near detector is a 2 cm diameter cylinder, 2 cm in length, located 20 cm from the source. The far detector is a 4 cm by 4 cm cylinder located 40 cm from the source. Each detector is made of NaI and collimated to look out into the formation. The source is contained in an angled collimator, aiming upward and into the formation. The collimators are filled with oxygen. The extent of the modeled formation is $100 \times 100 \times 140$ cm. The source is an isotropic ^{137}Cs source emitting 10^{11} photons/s (661.7 keV). For these calculations, no response function is used-the goal of this example is to calculate the total photon flux within each detector volume.

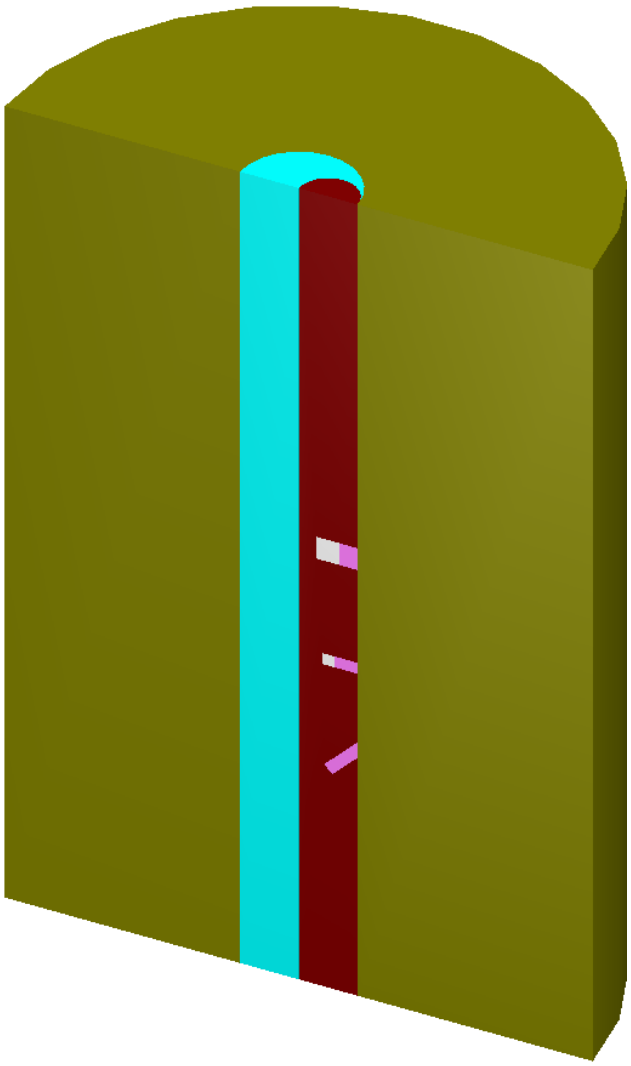


Fig. 4.1.21: Cutaway view of the litho-density tool in a rock formation.

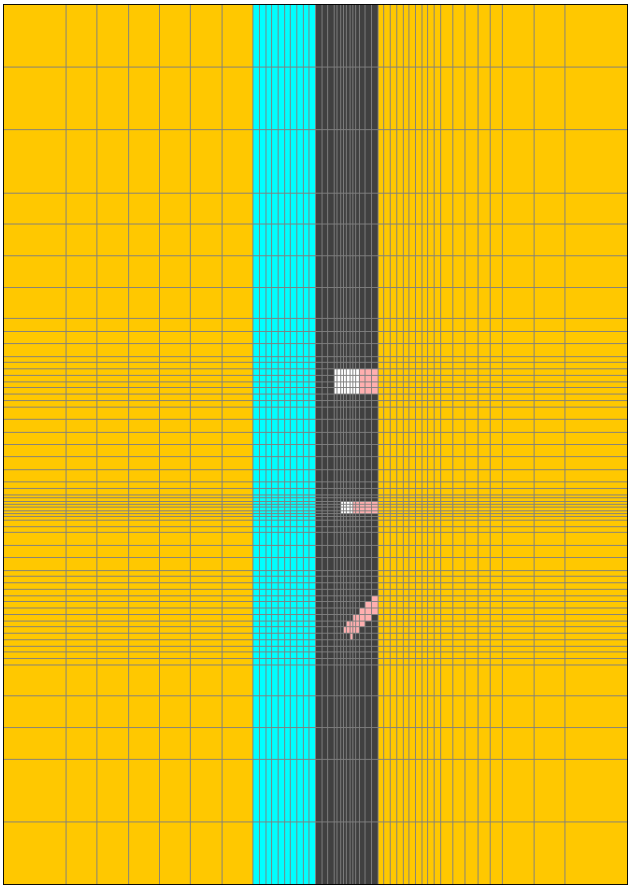


Fig. 4.1.22: Denovo mesh, y=0 plane.

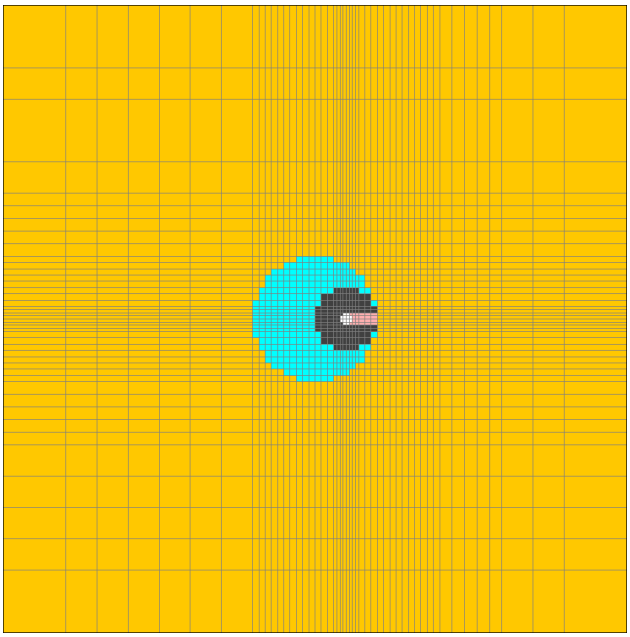


Fig. 4.1.23: Denovo mesh, z=20 plane (through near detector).

Input file

The following input file represents the simple model of the litho-density tool. It lists of the file `mavric.lithoFW.inp` located in the SCALE `samples\input` directory. The 27 neutron group/19 photon group library based on ENDF/B-VII.0 data was used for the discrete-ordinates calculations, whereas the final Monte Carlo calculation used the 200 neutron group / 47 photon group library.

The input file starts with the problem title, the library for the importance calculations, and the materials.

```
=mavric
Photon lithodensity logging tool - using FW-CADIS
v7-27n19g

read composition
    o           1 den=1.0               1.0 293.0 end
    wtptNaI     2 3.67  2 11000 15.337
                          53129 84.663  1.0 293.0 end
    iron        3 den=19.3              1.0 293.0 end
    h2o         4                       1.0 293.0 end
    wtptCaCO3   5 2.36  4  1000  1.0
                           6000 11.0
                           8016 51.4
                          20000 36.6    1.0 293.0 end
end composition
```

The geometry is relatively simple. Volumes are only needed for the regions where tallies will be made.

```
read geometry
    global unit  1
        zcylinder  2 1.0 10.0  0.0  origin x=5.0 rotate a1=90 a2=45
        zcylinder  3 1.0 21.0 19.0  origin x=5.0
        xcylinder  4 1.0 10.0  5.0  origin z=20.0
        zcylinder  5 2.0 42.0 38.0  origin x=5.0
        xcylinder  6 2.0 10.0  5.0  origin z=40.0

        zcylinder  7  5.0 100 -40 origin x=5.0
        zcylinder 11 10.0 100 -40
        cuboid    12 50 -50  50 -50   100 -40

        media   1 1   2 7
        media   2 1   3          vol=6.2831853072
        media   1 1   4 -3 7
        media   2 1   5          vol=50.265482456
        media   1 1   6 -5 7
        media   3 1   7 -2 -3 -4 -5 -6
        media   4 1   11 -7
        media   5 1   12 -11
    boundary 12
end geometry
```

The definitions block lists the response (total photon flux) so that it can be understood by both libraries used in the problem. The mesh grid for the importance calculations used $49 \times 43 \times 59 = 124{,}313$ mesh cells, with particular emphasis on geometric representation of the collimators to ensure accurate importance maps. This mesh grid is shown in Fig. 4.1.22 and Fig. 4.1.23. The source is represented by a distribution in which the most probable bin is located right near the 661.7 keV line, which can be translated into one photon group in either library (from 600 to 800 keV in the 27/19 library or from 600 to 700 keV in the 200/47 library).

```
read definitions
    response 1
        title="for computing total photon flux"
        photonBounds 1.0e4 2.0e7 end
```

(continues on next page)

(continued from previous page)

```
        values         1.0   1.0 end
    end response
    gridgeometry 1
        title="entire formation"
        xplanes -50 -40 -35 -30 -25 -20 -15
                -10 -9 -8 -7 -6 -5 -4 -3 -2 -1
                0 1 2 3 3.5 4 4.5 5 5.5 6 6.5 7 8 9
                10 11 12 13 14 15 16 17 18 19 20
                22 24 26 28 30 35 40 50 end
        yplanes -50 -40 -35 -30 -25 -20 -18 -16 -14 -12
                -10 -9 -8 -7 -6 -5 -4 -3 -2 -1.5 -1 -0.5
                0 0.5 1 1.5 2 3 4 5 6 7 8 9
                10 12 14 16 18 20 25 35 40 50 end
        zplanes -40 -30 -20 -15 -10 -5
                -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10
                12   14 16 17 18 18.5
                19   19.5 20 20.5
                21   21.5 22 23 24 26 28 30 32 34 36 37
                38   39 40 41
                42   43 44 46 48 50 55 60 65 70 80 90 100 end
    end gridgeometry
    gridgeometry 2
        title="y=0 plane in detail"
        xLinear 100  -50.0  50.0
        yLinear   1   -1.0   1.0
        zLinear 140  -40.0 100.0
    end gridgeometry
    distribution 1
             title="cesium-137 gammas/decay"
        discrete 31817  32194  36304   36378   37255   283500    661657 end
        truepdf  0.0199 0.0364 0.00348 0.00672 0.00213 0.0000058 0.851  end
    end distribution
end definitions
```

The source is a simple point source. For more realistic calculations, the source strength could be modified so that the total energy emitted is preserved, $S' = S\left(\frac{661.7}{0.5}(700+600)\right)$, which uses the ratio of the line energy to the energy at the center of the group in the 200/47 library.

```
read sources
    src 1
        title="Cs-137 Source: 661.7 keV"
        strength=1.0e11
        photons
        sphere  0.0 origin  x=5.0
        eDistributionID=1
    end src
end sources
```

Each detector is represented by a region tally. A mesh tally is made for one slice in y for visualization.

```
read tallies
    regionTally 1
        photon  unit=1 region=2
    end regionTally
    regionTally 2
        photon  unit=1 region=4
    end regionTally
    meshTally 1
        title="mesh tally in just the y=0 plane"
        photon
        gridGeometryID=2
        noGroupFluxes
    end meshTally
```

(continues on next page)

(continued from previous page)

```
end tallies
read parameters
    library="v7-200n47g"
    randomSeed=00003ecd7b4e3e8b
    perBatch=466000 batches=24
    noNeutrons
end parameters
```

The importance map defines two adjoint sources that correspond to the two tallies. Forward weighting, based on the response integrated over energy ("respWeighting"), is used. Because the true source is a point source, the subcell method of making a mesh source will fail, so the number of source trials is input. This number is small since the source is a monoenergetic point source. The Denovo calculations used the default S_8 quadrature and the P_3 Legendre order.

```
read importanceMap
    gridGeometryID=1

'   near detector
    adjointSource 1
        boundingBox 6 4  1 -1  21 19
        unit=1 region=2
        responseID=1
    end adjointSource

'   far detector
    adjointSource 2
        boundingBox 7 3  2 -2  42 38
        unit=1 region=4
        responseID=1
    end adjointSource
    respWeighting
    sourceTrials=100
end importanceMap

end data
end
```

Output

The results for the two region tallies, the first for the near detector and the second for the far detector after 60 minutes of computations (3 forward Denovo, 4 adjoint Denovo and 52 Monaco), were as follows.

```
Photon Region Tally 1.
                        average      standard     relat      FOM    stat checks
    tally/quantity       value       deviation    uncert   (/min)   1 2 3 4 5 6
    ------------------  -----------  -----------  -------  --------  -----------
    total flux (tl)     1.47918E+03  1.57461E+01  0.01065  1.70E+02  X - X X X X
    total flux (cd)     1.47936E+03  1.56963E+01  0.01061  1.71E+02  X - X - X X
    ------------------  -----------  -----------  -------  --------  -----------

 Photon Region Tally 2.
                        average      standard     relat      FOM    stat checks
    tally/quantity       value       deviation    uncert   (/min)   1 2 3 4 5 6
    ------------------  -----------  -----------  -------  --------  -----------
    total flux (tl)     4.57691E+01  2.81778E-01  0.00616  5.07E+02  X X X X X X
    total flux (cd)     4.57825E+01  2.80714E-01  0.00613  5.11E+02  X X X X X X
    ------------------  -----------  -----------  -------  --------  -----------
```

Note that both detectors have similarly low relative uncertainties (about 1%) even though the tally values differ by a factor of 30. These results should be compared to analog results (no biasing at all) and optimizations of each detector in separate input files, as shown in Table 4.1.21. The CADIS calculations for each detector

(near or far) do exactly what they were supposed to do—optimize the Monte Carlo calculation for one tally or the other. The FOMs for the FW-CADIS calculation were about half of the FOMs for the single-detector CADIS calculations. Hence, for this two-detector problem, two CADIS calculations are just as efficient as one FW-CADIS calculation. For modern well-logging tools with up to a dozen detectors, a single FW-CADIS would be much more efficient and manageable. Note that the near detector still needs more time to pass the second (uncertainty fit) and the fourth (VOV fit) statistical checks. Neither of the single-detector CADIS calculations passed the fourth statistical check within an hour.

Fig. 4.1.24 shows a mesh tally of total photon flux in the y=0 plane for all four of the simulations: analog, CADIS for the near detector, CADIS for the far detector, and the FW-CADIS calculation for both.

Table 4.1.21: Comparison of different CADIS techniques for the litho-density problem

	Time (min)			Near Detector			Far Detector		
	For	Adj	MC	flux (/cm^2/s)	relative uncert.	FOM (/min)	flux (/cm^2/s)	relative uncert.	FOM (/min)
Analog			60.1	1.19E+03	0.3765	0.12	1.17E+02	0.7098	0.03
CADIS - Near		4.8	57.2	1.49E+03	0.0064	390	0.00E+00	0.0000	----
CADIS - Far		4.7	57.0	8.28E+02	0.4162	0.09	4.56E+01	0.0048	703
FW-CADIS	2.8	4.4	52.1	1.48E+03	0.0107	149	4.58E+01	0.0062	444

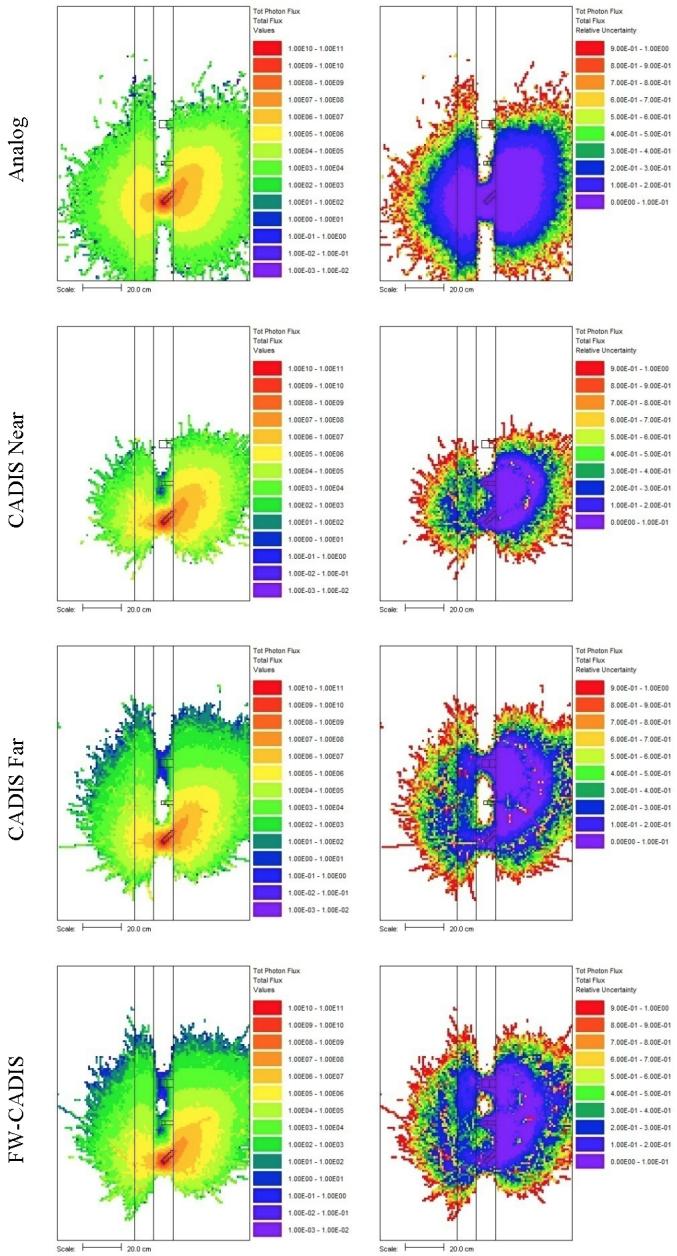


Fig. 4.1.24: Mesh tallies showing total photon flux in cm^2/s (left column) and its relative error (right column) in the y=0 plane.

4.1.5.4 AOS-100 using FW-CADIS and continuous-energy transport

The advanced variance reduction in MAVRIC can also be used when the final Monaco Monte Carlo transport calculation is performed using a CE cross section library. Simulations involving discrete-energy photon sources are best handled with CE. Consider the AOS-100, one of several radioactive material transport packaging systems developed by Alpha Omega Services, Inc., and a ^{60}Co source. (International Isotopes Inc. of Idaho Falls, Idaho, distributes the AOS Radioactive Material Transport Packaging Systems.)

A simple model of the AOS-100 package, which is constructed primarily of steel and tungsten, is shown in Fig. 4.1.25. The diameter is 71.12 cm, and the height is 91.44 cm. The innermost cylinder (16.51 cm diameter and 50.8 cm height) typically contains the material to be transported, but in this study, this region is simply modeled as an air region containing a uniform source of 1 Ci of ^{60}Co. This is conservative-it assumes the radioactive material containing the cobalt provides no self-shielding.

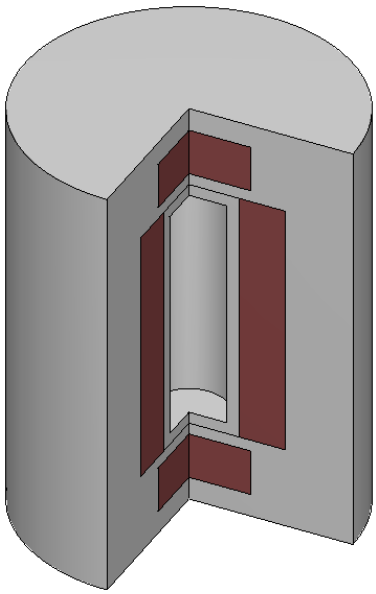


Fig. 4.1.25: Simple AOS-100 cask geometry showing tungsten (brown) and steel (gray).

The objective in this study is to compute the dose rates around the cask. This can be done using a mesh tally. Note that the dose rates inside the package are not of concern-only the dose rates outside the package are.

The results of a 15-hour analog calculation (using only implicit capture with a lower weight limit of 10^{-7}) are shown in Fig. 4.1.26. This calculation does not show many photons that have escaped the package into the air, which is similar to reality for this heavily shielded cask. To compute dose rates outside the package, variance reduction is needed.

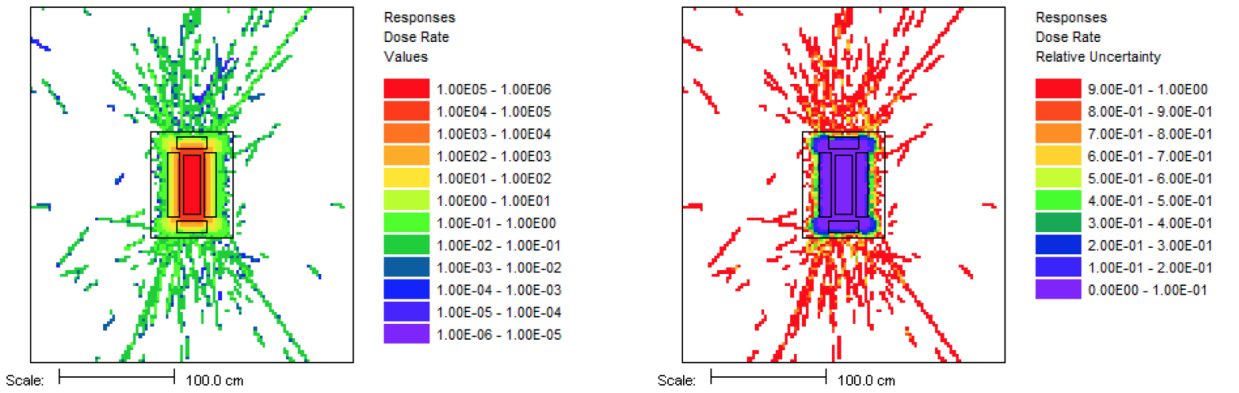


Fig. 4.1.26: Dose rates (mrem/hr/Ci) and relative uncertainties along the $y = 0$ midplane of the cask as calculated by an analog calculation.

Input file

The sample problem file `mavric.aos100.inp` is located in the `samples\input` directory. For this demonstration, a very simplified geometry is used, with a few tungsten regions inside a steel cylinder. The library given at the beginning of the input is the MG library for the importance calculations. The final Monaco Monte Carlo calculation will use the CE library, which is specified in the parameters block. About 1 meter of air is modeled around the cask.

```
=mavric
AOS-100: Demonstrate use of MG importance maps with final MC using CE transport
v7-200n47g
read composition
    ss304           1   end
    wtptTungsten    2  17.8 4  74182 26 74183 14 74184 31 74186 29 end
    dry-air         3   end
end composition

read geometry
    global unit 1
        cylinder 1  8.255   25.40 -25.40
        cylinder 2 10.795   27.94 -27.94
        cylinder 3 20.955   27.94 -27.94
        cylinder 4 13.335   40.64  30.48
        cylinder 5 13.335  -30.48 -40.64
        cylinder 6 35.56    45.72 -45.72

        cuboid  99  139.7 -139.7   139.7 -139.7  152.4 -152.4

        media 3 1  1
        media 1 1  2 -1
        media 2 1  3 -2
        media 2 1  4
        media 2 1  5
        media 1 1  6 -3 -4 -5

        media 3 1  99 -6
    boundary 99
end geometry
```

In the definitions block, the photon dose response function, the cobalt-60 line spectrum, two grid geometries, and an energy boundaries structure are all defined. The first grid geometry is used for the deterministic

calculation, the importance map, and the biased source. The second grid geometry is for the high-resolution (1 in.) mesh tally. The energyBounds defined here has a base structure of 30 bins that are 50 keV wide, with three bins that are 2 keV wide at the dominant cobalt line energies and the 511-keV annihilation gamma energy.

```
read definitions
    response 5
        title="ANSI standard (1977) neutron flux-to-dose-rate factors"
        doseData=9504
    end response

    distribution 1
        title="cobalt-60 gammas/decay"
        discrete   347140   826100 1173228 1332492  2158570    2505692 end
        truepdf  0.000075 0.000076 0.9985 0.999826 0.000012 0.00000002 end
    end distribution


    gridGeometry 7
        title="mesh for discrete ordinates/importance map/biased source"
        xLinear 28   -35.56  35.56
        yLinear 28   -35.56  35.56
        zLinear 36   -45.72  45.72

        xLinear 22  -139.7  139.7
        yLinear 22  -139.7  139.7
        zLinear 24  -152.4  152.4
    end gridGeometry

    gridGeometry 8
        title="mesh for mesh tally - 1 inch voxels"
        xLinear  110  -139.7   139.7
        yLinear  110  -139.7   139.7
        zLinear  120  -152.4   152.4
    end gridGeometry

    energyBounds 1
        linear 30 0.00e6 1.50e6
        bounds  0.510e+6 0.512e+6   1.172e6 1.174e6   1.331e6 1.333e6 end
    end energyBounds
end definitions
```

The source is modeled as a uniform volumetric source in the center region of the cask. Because the distribution of cobalt gamma rays was entered as gammas per decay, the keyword “useNormConst” will set the source strength to be the total of the energy distribution-about 2 photons/decay. The “multiplier” keyword is used to multiply that strength by 37e9 decays/sec to get 1 Ci.

```
read sources
    src 1
        title="one Ci of cobalt-60"
        useNormConst
        multiplier=37e9
        cylinder  8.255   25.40 -25.40
        photons
        eDistributionID=1
    end src
end sources
```

A single mesh tally is defined and is limited to the air region outside of the cask. A multiplier of 1,000 is used to convert the response values from rem/hr to mrem/hr.

```
read tallies
    meshTally 1
        photon
        gridGeometryID=8
        responseID=5
        unit=1 region=7
        energyBoundsID=1
    end meshTally

    multiplier=1000.0
end tallies
```

In this problem, the importance calculations will use the 200/47 MG library, which will transport all particles contained in the library by default. The keyword "noNeutrons" is used to turn off neutron transport during the Denovo calculations, thereby saving time. The final Monaco calculation will use the CE library ("ceLibrary="), which only transports the particles specifically requested by the user (to avoid loading large amounts of cross section from disk to memory). Thus, the keyword "photons" is included to specifically tell the CE Monaco to transport photons.

```
read parameters
    randomSeed=00003ecd7b4e3e8b
    ceLibrary="ce_v7np_endf.xml"
    perBatch=2000000 batches=233
    photons   noNeutrons
end parameters
```

The importance map uses FW-CADIS to construct a map and biased source that will optimize the photon dose rate in the air outside the cask. Since photon scatter is typically forward peaked, an S_{12} quadrature and P_5 Legendre scattering expansion are used.

```
read importanceMap
    adjointSource 1
        boundingBox 139.7 -139.7   139.7 -139.7  152.4 -152.4
        unit=1 region=7
        responseID=5
    end adjointSource
    respWeighting

    gridGeometryID=7
    macromaterial
        mmTolerance=0.01
    end macromaterial

    quadrature=12
    legendre=5
end importanceMap
```

Output file

Results for the mesh tally after 958 minutes (27 forward Denovo, 31 adjoint Denovo, and 900 CE-Monaco) are shown in Fig. 4.1.27. Since dose rates inside the package are not of concern, that region was excluded from the mesh tally. Due to the optimization that focused the Monte Carlo calculation on dose rates outside the cask, values of the dose rate inside the cask are underestimated and should not be used. Also note that voxels along the outer cylindrical edge of the package could show low dose rates, since the voxel value is an average and only part of the voxel is actually outside the package. The resolution of the mesh tally is 2.54 cm.

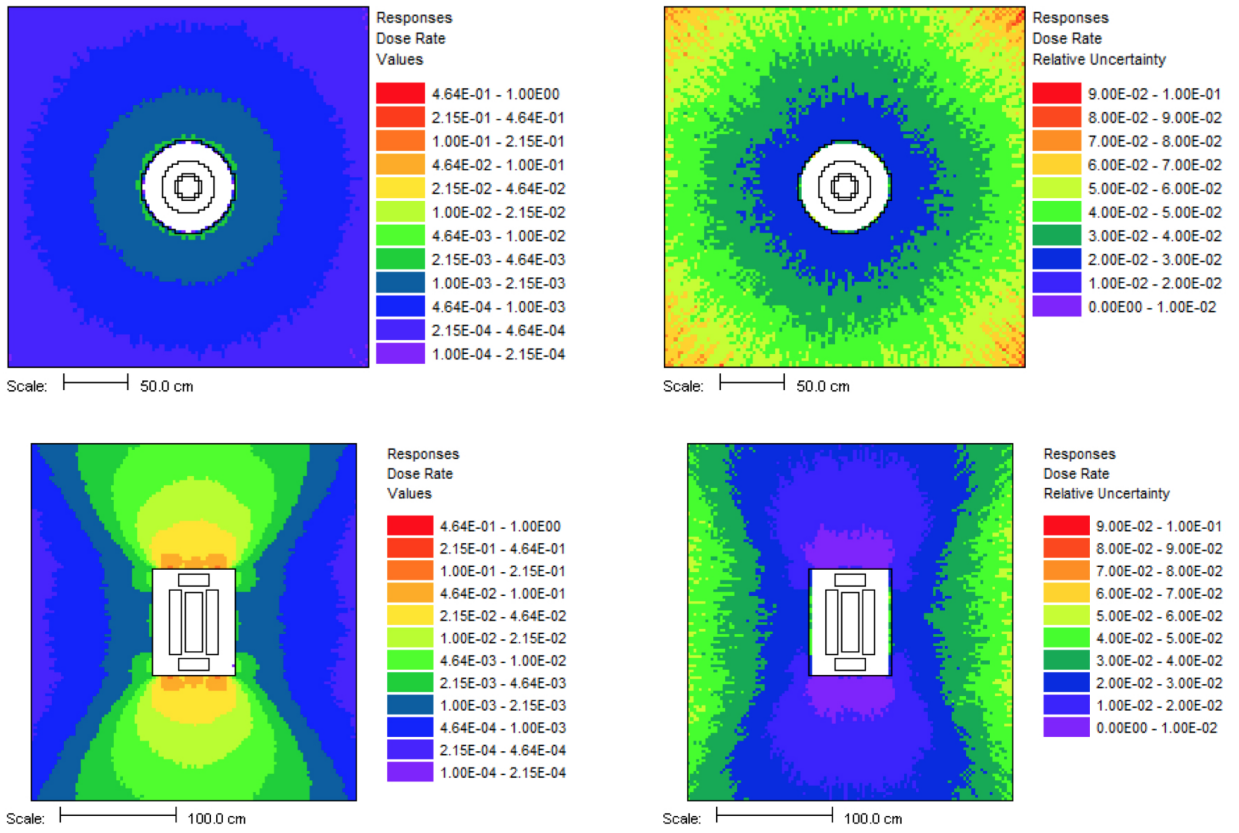


Fig. 4.1.27: Dose rates (mrem/hr/Ci) and relative uncertainties from the CE FW-CADIS calculation showing the midplane views of the cask ($z = 0$ above and $y = 0$ below).

The flux using the user-defined energy bin boundaries for a point 10 cm above the top of the package (along the axis) is shown in Fig. 4.1.28. Note that since the importance map was made to optimize the photon dose rate, some of the low-energy bins that do not contribute much to dose may have larger uncertainties.

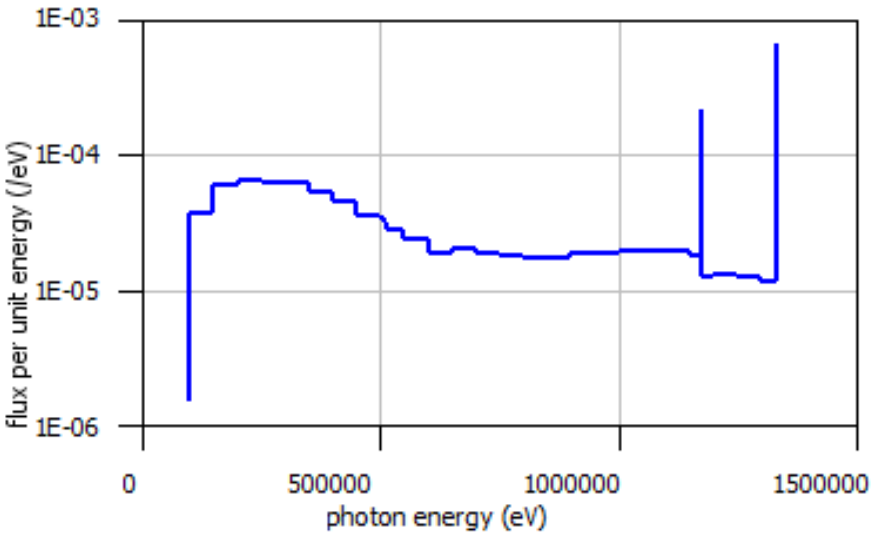


Fig. 4.1.28: Flux 10 cm above the cask.

Fig. 4.1.29 and Fig. 4.1.30 show the ratios of the dose rates computed using a MG calculation to the CE calculation. Dose rates inside the cask should not be compared because the importance map is focused on dose outside the cask, so low-energy photons are not simulated inside. The 47-group MG calculation is fairly close to the CE-calculation in dose rate (10% high axially, 20% high radially), but the 19-group MG dose rates are much higher than the CE. Neither of the MG calculations shows the 1.17 and 1.33 MeV lines in the energy spectra.

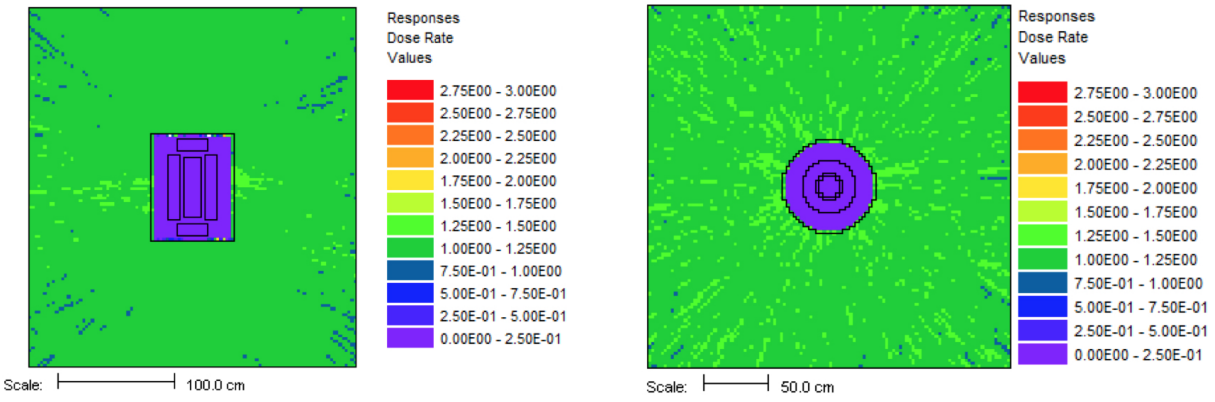


Fig. 4.1.29: Ratio of the 47-group MG computed dose rates to the CE dose rates ($y = 0$ left and $z = 0$ right).

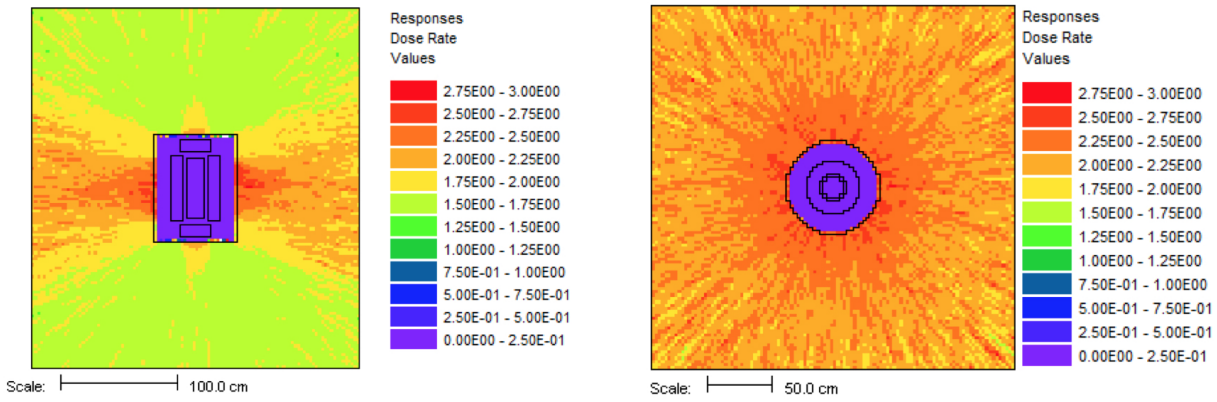


Fig. 4.1.30: Ratio of the 19-group MG computed dose rates to the CE dose rates ($y = 0$ left and $z = 0$ right).

4.1.5.5 Independent spent fuel storage installation

A good example of a problem in which the dose rate needs to be known well everywhere—in low dose areas and high dose areas—is an independent spent fuel storage installation. The dose rates around these arrays of spent fuel casks need to be evaluated over a large area of ground to determine the boundary of the controlled area. This sample problem uses a simplified model of a cask array, shown in Fig. 4.1.31 and Fig. 4.1.32, to demonstrate the FW-CADIS method for obtaining the dose rates with reasonably low relative uncertainties over a mesh tally which covers a very large area.

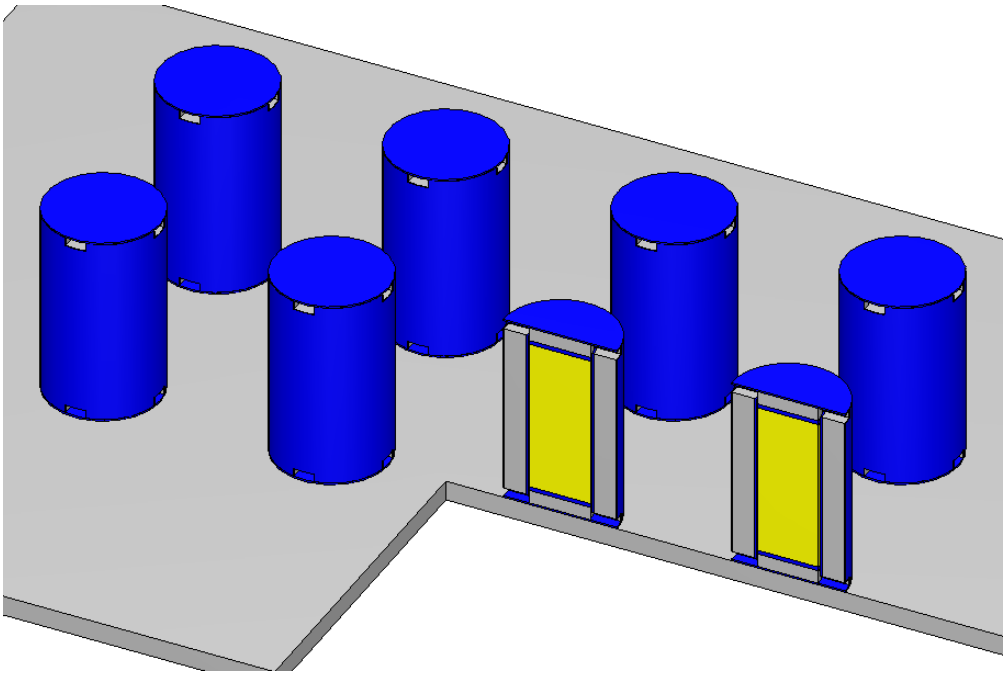


Fig. 4.1.31: Array of eight casks.

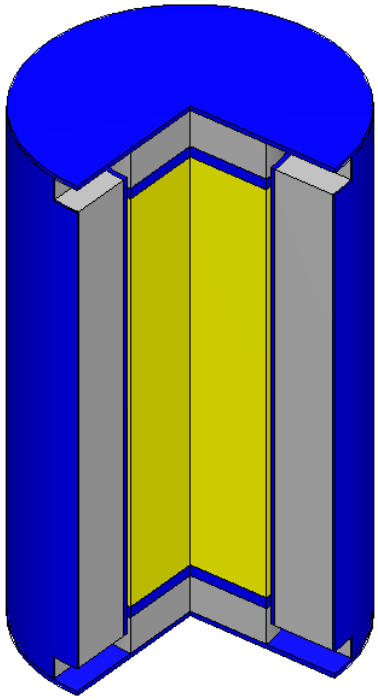


Fig. 4.1.32: Cask detail.

Source term

The calculation of the spent fuel source term using data from the Kewaunee nuclear power plant is described in the cask example problem and is modeled in the first part of `mavric.isfsi.inp` with ORIGEN. The result of this calculation is a binary concentration (ft71f001) file. This file also contains a list of the group-wise neutron source term ("total neutron spectra, neutrons/sec/basis") and photon source term ("gamma spectra, photons/sec/basis") for each time step. In this example problem, the source term for the MAVRIC calculation will be read directly from the binary concentration file. The file contains the concentration and source-term data at each of 64 time steps. For this analysis, the photon source at the last time step will be used.

Input file

The following is a listing of the second part (the MAVRIC input) of the file `mavric.isfsi.inp` located in the SCALE `samples\input` directory. Note that soil and air are included in the model to properly account for dose near the ground. The mesh tally does not extend into the air above 2 meters since this problem is only concerned with the area near the ground where people could be. For MAVRIC to read the ORIGEN binary concentration file, it must be named as a Fortran unit file with a number matching the distribution input. In this example, unit 71 is used in both the ORIGEN input and MAVRIC input.

```
=mavric
ISFSI - forward-weighted adjoint using photon dose response
v7-27n19g
```

The materials include the fresh fuel (with a density representing homogenization), concrete, steel, air, and ordinary soil.

```
read composition
    wtptFuel    1 0.913717475 18         6000  0.00939719    7014  0.00528993
                     8016  9.73397641   13000  0.00715715   14000  0.01031670
                    15000  0.02227505   22000  0.00780567   24000  0.36655141
                    25000  0.01716839   26000  0.72041451   27000  0.00523824
                    28000  0.68955526   40000 15.78990702   41000  0.05130153
                    42000  0.02844690   50118  0.25877903   92235  3.03560962
                    92238 69.24080999
                  1.0 293.0 end
    orconcrete  2 1.0 293.0 end
    ss304       3 1.0 293.0 end
    dry-air     4 1.0 293.0 end
    wtptSoil    5 1.685 12  1000 3.498   6000  2.138   8016 60.826  11000 0.579
                           13000 7.489  14000 18.841  17000  0.026  19000 0.582
                           20000 0.896  22000  0.488  25000  0.097  26000 4.540
                  1.0 293.0 end
end composition
```

The geometry models one cask and then repeats the cask model eight times using an array.

```
read geometry
  unit  1
    zcylinder  1   90.0  228.6 -228.6
    zcylinder  2   90.0  240.6 -240.6
    zcylinder  3   90.0  280.6 -280.6
    zcylinder  4   95.0  280.6 -280.6
    zcylinder  5  100.0  280.6 -280.6
    zcylinder  6  168.0  280.6 -280.6
    zcylinder  7  170.0  280.6 -280.6
    zcylinder  8  170.0  285.6 -285.6
    cuboid    11  170  90 30 -30   280.6 255.6
    cuboid    12  30 -30 170 90    280.6 255.6
    cuboid    13  -90 -170 30 -30  280.6 255.6
    cuboid    14  30 -30 -90 -170  280.6 255.6
    cuboid    21  170  90 30 -30   -255.6 -280.6
    cuboid    22  30 -30 170 90    -255.6 -280.6
    cuboid    23  -90 -170 30 -30  -255.6 -280.6
    cuboid    24  30 -30 -90 -170  -255.6 -280.6
    cuboid    99  340.0 -340.0   340.0 -340.0   285.6 -285.6

    media  1    1   1
    media  3    1   2 -1
    media  2    1   3 -2
    media  4    1   4 -3
    media  3    1   5 -4  -11 -12 -13 -14  -21 -22 -23 -24
    media  2    1   6 -5  -11 -12 -13 -14  -21 -22 -23 -24
    media  3    1   7 -6  -11 -12 -13 -14  -21 -22 -23 -24
    media  3    1   8 -7
    media  4    1   11 -4 7
    media  4    1   12 -4 7
    media  4    1   13 -4 7
    media  4    1   14 -4 7
    media  4    1   21 -4 7
    media  4    1   22 -4 7
    media  4    1   23 -4 7
    media  4    1   24 -4 7
    media  4    1   99 -8
  boundary 99
  global unit 2
    cuboid 1   1360 -1360      680 -680       571.2    0.0
    cuboid 2   2040 -2040     1360 -1360        0.0  -60.0
    cuboid 3  12000 -12000   12000 -12000       0.0 -100.0
    cuboid 4  12000 -12000   12000 -12000   10000.0 -100.0
    array 10 1 place 1 1 1  -1020 -340 285.6
    media  2  1   2
    media  5  1   3 -2
```

(continues on next page)

(continued from previous page)

```
    media  4  1   4 -3 -1
  boundary 4
end geometry

read array
    ara=10 nux=4 nuy=2 nuz=1  fill 1 1 1 1   1 1 1 1   end fill
end array
```

The definitions block contains the photon dose response, a mesh geometry for the discrete-ordinates calculations, a uniform mesh for the mesh tally, and the photon energy distribution read from the binary concentration file (located on unit 71, the last time step case 64).

```
read definitions
    response 6
        specialDose=9504
    end response
    gridGeometry 3
        title="exercise geometry 68x44x24 = 71808"
        xplanes -12000 -10000 -8000 -6000 -4000 -3000
                -2040 -1870 -1700 -1530
                -1360 -1275 -1190 -1150 -1110 -1070 -1020 -970 -930 -890 -850 -765
                -680 -595 -510 -470 -430 -390 -340 -290 -250 -210 -170 -85
                0 85 170 210 250 290 340 390 430 470 510 595 680
                765 850 890 930 970 1020 1070 1110 1150 1190 1275 1360
                1530 1700 1870 2040
                3000 4000 6000 8000 10000 12000 end
        yplanes -12000 -10000 -8000 -6000 -4000 -2000
                -1360 -1190 -1020 -850
                -680 -595 -510 -470 -430 -390 -340 -290 -250 -210 -170 -85
                0 85 170 210 250 290 340 390 430 470 510 595 680
                850 1020 1190 1360
                2000 4000 6000 8000 10000 12000 end
        zplanes -100 -50 -30 -15 0 25 50 75 100 200 300 400 500 570 600 800
                900 1000 1500 2000 3000 4000 6000 8000 10000 end
    end gridGeometry
    gridGeometry 4
        title="large voxels for mesh tally"
        xLinear 24 -12000.0 12000.0
        yLinear 24 -12000.0 12000.0
        zLinear  1      0.0   200.0
    end gridGeometry

    distribution 1
'       use result from kewaunee.origen.inp, case 64, master photon library
        special="origensBinaryConcentrationFile"
        filename="ft71f001"
        parameters 71 64 4 end
    end distribution
end definitions
```

The source can be defined in two ways. Since there are eight casks, eight sources could be defined, each over a cylinder with a strength of 7.155×10^{16} photon/s. In this example of eight identical sources, one source region can be defined and then limited to only exist in the spent fuel material (the first material) but with a strength for all the casks of 5.724×10^{17} photon/s.

```
read sources
    src 1
        title="each cask holds 1/6 of kewaunee core"
        strength=5.724e17
        cuboid -1115 1115  435 -435  514.2  57
        mixture=1
        photons
```

(continues on next page)

(continued from previous page)

```
        eDistributionID=1
    end src
end sources
```

The mesh tally covers only the first 2 meters above the ground since the dose rate above that would not impact a person.

```
read tallies
    meshTally 1
        title="120m from cask array center"
        photon
        gridGeometryID=4
        responseID=6
    end meshTally
end tallies

'------------------------------------------------------------------------------
' Parameters Block
'------------------------------------------------------------------------------
read parameters
    randomSeed=8655740521000041
    perBatch=940000  batches=60
end parameters
```

The adjoint source is placed everywhere that the dose rate is desired—near the ground but not very close to or in between the casks (where people will not be). The macro material option is used here to ensure that the discrete-ordinates calculation is more representative of the problem.

```
read importanceMap
    gridGeometryID=3
    adjointSource 1
        boundingBox 12000 -12000   12000 -12000  600.0 0.0
        responseID=6
        unit=2 region=3
    end adjointSource
    respWeighting
    macromaterial
        mmTolerance=0.001
    end macromaterial
    saveExtraMaps
end importanceMap

end data
end
```

Output

The MAVRIC calculational times were 3 minutes for the forward Denovo, 3.5 minutes for the adjoint Denovo, and 60 minutes for the Monaco forward Monte Carlo. Results for the mesh tally of dose rates are shown in Fig. 4.1.33. For a longer Monaco calculation (64 hr), the mesh tally results are shown in Fig. 4.1.34. Note that the scale for the dose rate has been adjusted to only show the values outside of the cask array.

Using the Mesh File Viewer, the distribution of relative uncertainty in a mesh tally can be plotted. For this 1 hr calculation, 50% of the voxels in the mesh tally had less than 20% relative uncertainty. Fig. 4.1.35 shows how the distribution of relative errors changes with longer run times. For the 64 hr run, 90% of the voxels had less than 10% relative uncertainty.

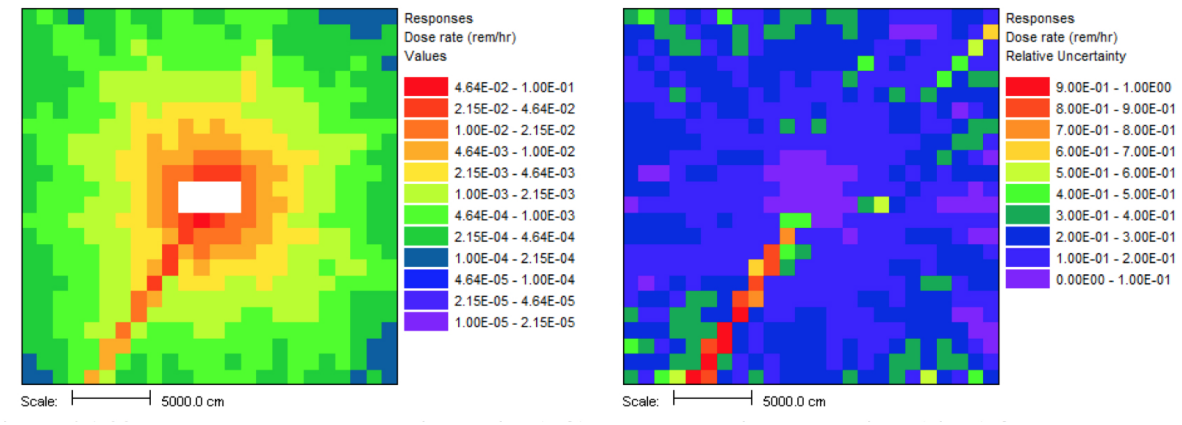


Fig. 4.1.33: Photon dose rate values in rem/hr (left) and the relative uncertainty (right) for the area around the cask array (1 hr Monaco).

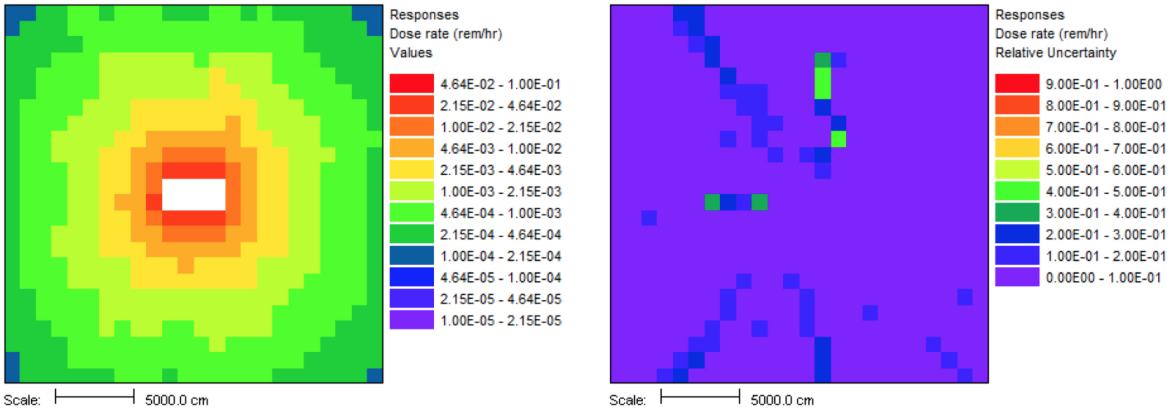


Fig. 4.1.34: Photon dose rate values in rem/hr (left) and the relative uncertainty (right) for the area around the cask array (64 hr Monaco). For the uncertainty plot, purple represents <=5%.

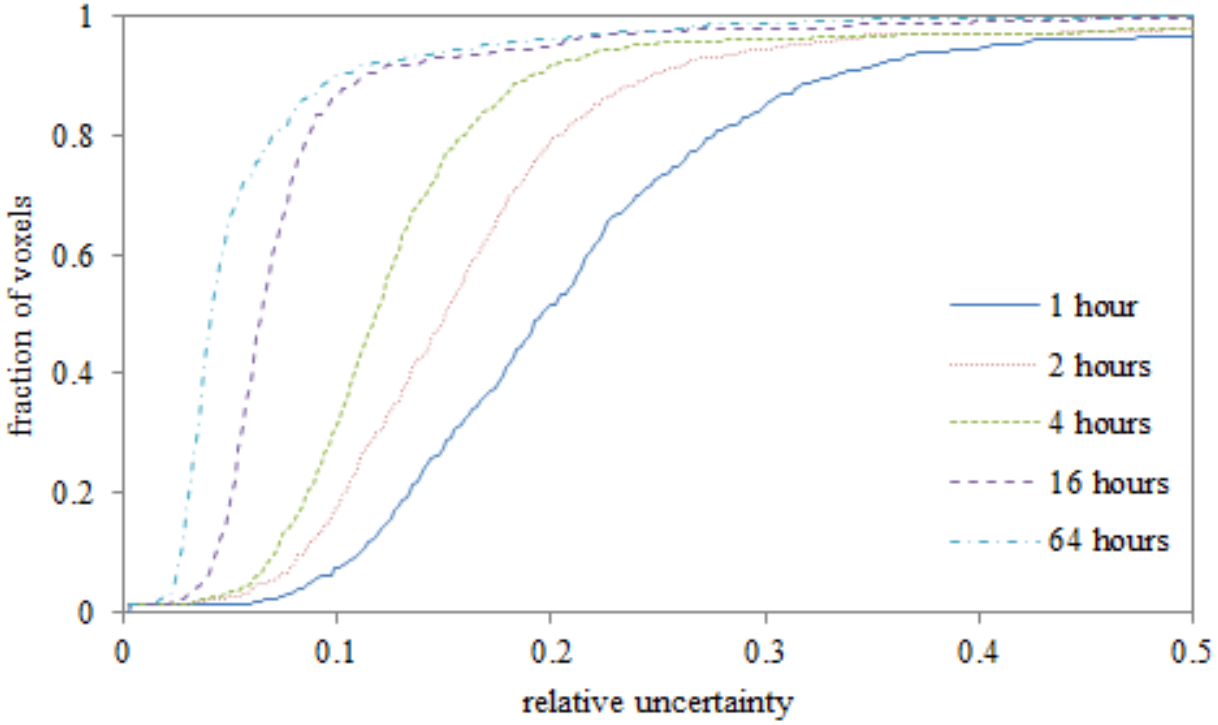


Fig. 4.1.35: Distribution of relative uncertainties for different run times of `mavric.isfsi.inp` showing the fraction of the mesh tally voxels that had less than a given amount of relative uncertainty.

4.1.5.6 TN24-P spent fuel cask

For an example that uses multiple sources, user-defined distributions in those sources, macromaterials for improved S_N calculations, and the automated variance reduction capabilities in MAVRIC, consider the model for the TN-24P cask, as used in previous SCALE shielding reports [MAVRIC-WDP+09] and shown in Fig. 4.1.36. This model contains two types of PWR spent fuel assemblies (Types V and W, both Westinghouse 15×15 assemblies of different starting enrichments and burnups), each with specified neutron and photon sources, in an aluminum/boron fuel basket. The cask is made of forged steel for photon shielding with a resin layer for neutron shielding. The model also includes three activated hardware regions (bottom nozzle, top nozzle, and top plenum), which consist of specified amounts of ^{60}Co (a photon source). The task for this example is to calculate the total dose rate within 2 meters of the cask surface.

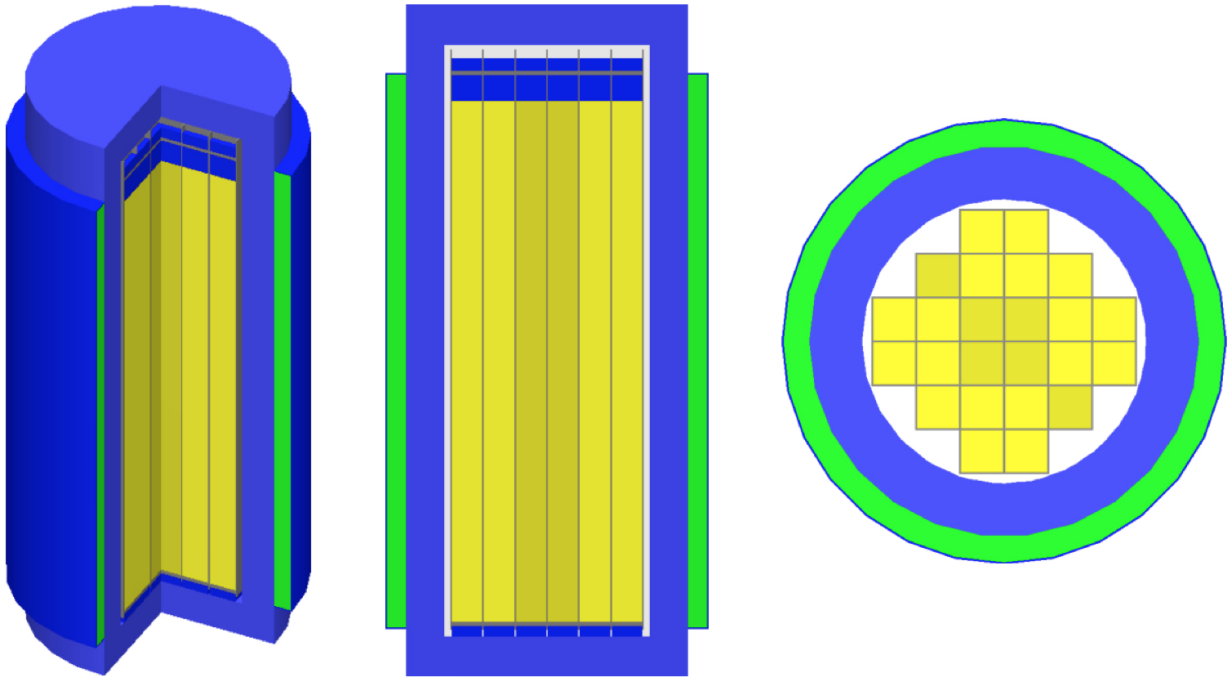


Fig. 4.1.36: MAVRIC model of the TN24-P cask. Materials: spent fuel (light and dark yellow), steels (blues), resin (green), and other metals (gray).

For MAVRIC, this means that the calculation of a dose rate mesh tally is calculated using FW-CADIS to ensure that each voxel has low relative uncertainty, independent of the dose rate. Without MAVRIC, the calculation of dose rate everywhere in three dimensions would be too challenging. Most likely, the dose rate would be evaluated with reasonable uncertainty at only a few locations. In fact, with analog calculations, this example would be a very difficult problem since most source particles never leave the cask, just as in the real-life situation. This type of problem really benefits from the CADIS-biased source distribution, in which source particles deep inside the cask are sampled very rarely since they do not contribute significantly to the response.

Input file

The following is a partial list of the file `mavric.tn24p.inp` located in the SCALE `samples\input` directory. This calculation will use the coarse-group shielding library for all of the importance map calculations but then will use the fine-group library for the final Monaco step. The full geometry and source distributions are not printed here due to their length.

```
=mavric
Monaco/MAVRIC Training - Exercise 3. Graphite Shielding Measurements Revisited
v7-27n19g
read composition
    ...
end composition

read geometry
    ...
end geometry

read array
    ...
end array
```

The definitions block includes three responses (neutron, photon, total), two grid geometries (one for the importance map calculations and one for a mesh tally), one cylindrical mesh for a mesh tally, five distributions for the energy spectra of the sources, and two distributions for the spatial distributions of the fuel assembly sources.

```
read definitions
    response 1  specialDose=9029  end response
    response 2  specialDose=9504  end response
    response 3  specialDose=9729  end response

    gridGeometry 1
        title="for discrete ordinates calculations  48 x 48 x 61"
        ...
    end gridGeometry
    gridGeometry 2
        title="for mesh tallies"
        xLinear 66 -330 330
        yLinear 66 -330 330
        zLinear 90 -435 465
    end gridGeometry
    cylGeometry 3
        radiusLinear 66 0 330
        degreeLinear 72 0 360
        zLinear 90 -435 465
    end cylGeometry

    distribution 1
        title="neutron energy distribution for W assembly"
        abscissa ... end
        truePDF ... end
    end distribution
    distribution 2
        title="photon energy distribution for W assembly"
        abscissa ... end
        truePDF ... end
    end distribution


    distribution 3
        title="neutron energy distribution for V assembly"
        abscissa ... end
        truePDF ... end
    end distribution
    distribution 4
        title="photon energy distribution for V assembly"
        abscissa ... end
        truePDF ... end
    end distribution
    distribution 5
        title="cobalt-60 gammas/decay"
        special="origensDiscreteGammas"
        parameters 27 60 end
    end distribution

    distribution 101  special="pwrNeutronAxialProfile"  end distribution
    distribution 102  special="pwrGammaAxialProfile"   end distribution

    runSampleTests   makeCharts
end definitions
```

Seven sources are defined—a neutron and photon source for each type of fuel assembly, and three activated hardware regions.

```
read sources
    src 1
        title="W assembly, neutron source"
        neutrons
        strength=1.6472e9
        zCylinder 71.0  182.9 -182.9
        zDistributionID=101  zScaleDist
        mixture=2
        eDistributionID=1
    end src
    src 2
        title="W assembly, photon source"
        photons
        strength=5.3638e16
        zCylinder 71.0  182.9 -182.9
        zDistributionID=102  zScaleDist
        mixture=2
        eDistributionID=2
    end src

    src 3
        title="V assembly, neutron source"
        neutrons
        strength=7.1927e8
        zCylinder 63.0  182.9 -182.9
        zDistributionID=101  zScaleDist
        mixture=12
        eDistributionID=3
    end src
    src 4
        title="V assembly, photon source"
        photons
        strength=1.8234e16
        zCylinder 63.0  182.9 -182.9
        zDistributionID=102  zScaleDist
        mixture=12
        eDistributionID=4
    end src
     src 6
        title="bottom nozzel Co-60 source"
        photons
        strength=1.7960e14
        zCylinder 71.0 -186 -193
        mixture=9
        eDistributionID=5
    end src

    src 7
        title="top nozzle Co-60 source"
        photons
        strength=1.0103e14
        zCylinder 71.0 212.8 203.9
        mixture=10
        eDistributionID=5
    end src

    src 8
        title="plenum Co-60 source"
        photons
        strength=6.3137e13
        zCylinder 71.0 201.0 182.9
        mixture=11
        eDistributionID=5
    end src
end sources
```

Four mesh tallies are used to collect the neutron dose rate, the photon dose rate, and the total dose rate.

```
read tallies
 meshTally 1
     neutron
     gridGeometryID=2
     responseID=1
     noGroupFluxes
 end meshTally
 meshTally 2
     photon
     gridGeometryID=2
     responseID=2
     noGroupFluxes
 end meshTally
 meshTally 11
     neutron
     cylGeometryID=3
     responseID=1
     noGroupFluxes
 end meshTally
 meshTally 12
     photon
     cylGeometryID=3
     responseID=2
     noGroupFluxes
 end meshTally
end tallies
```

The goal of this example is to calculate the total dose outside the cask. Therefore, the adjoint source uses the total dose rate response function for its energy component, whereas for the spatial component, it uses a large block around the cask. Note the "mixture=" keyword, which restricts the adjoint source to only exist where the material is air (13). There is no need to put adjoint source deep in the cask since the dose rates inside the cask are not of interest. Response weighting is used to put more adjoint source in the low dose areas outside the cask. Note that since this area is air, not many interactions/scatter take place, so a great balance in relative uncertainties near and far from the cask should not be expected. Macromaterials are used to improve the results from the discrete-ordinates calculations, which should improve the FOM of the final Monaco calculation. The geometry images in Sect. 4.1.3.4.2 are taken from this problem.

```
read importanceMap
    gridGeometryID=1
    adjointSource 1
        boundingBox  330 -330   330 -330  465 -435
        responseID=3
        mixture=13
    end adjointSource
    respWeighting
    reduce
    subCells=3
    macromaterial
        mmTolerance=0.001
    end macromaterial
end importanceMap

end data
end
```

Output

The distributions used by the source descriptions of the TN24-P model are shown in Fig. 4.1.38, and the responses used in this problem are shown in Fig. 4.1.37. Fig. 4.1.39 and Fig. 4.1.40 show the total (neutron + photon) dose rate outside the TN24-P cask, up to 2 meters from each surface using rectilinear and cylindrical mesh tallies with 10 cm voxels. Uncertainties in the computed dose rates were 3-4% after this 16 hr calculation. The scale of the figure was adjusted to only show the dose rate outside the cask; dose rates inside the cask went as high as 6×10^5 rem/hr. Likewise, the scale of the uncertainties ranged from 0 to 10% to highlight the uncertainties outside the cask. Relative uncertainties inside the cask were much higher (white areas) since those areas were deemed unimportant by the chosen adjoint source. Obtaining dose rates on a 10 cm mesh would not be achievable without the variance reduction capabilities of MAVRIC.

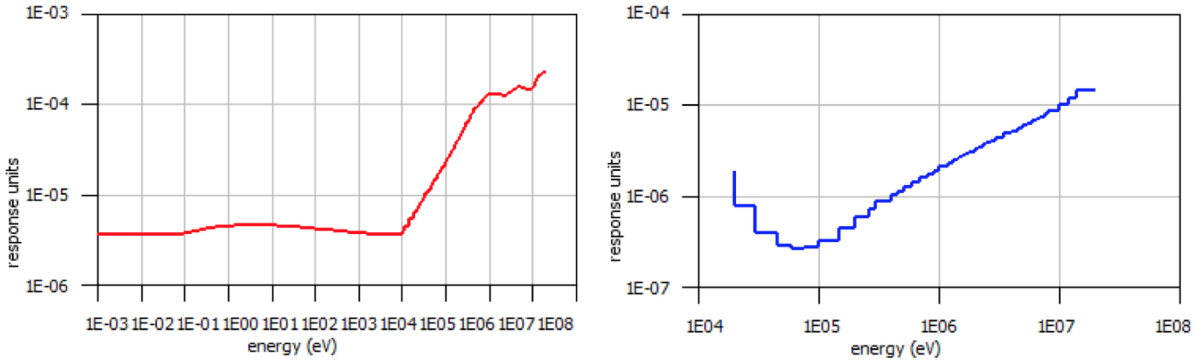
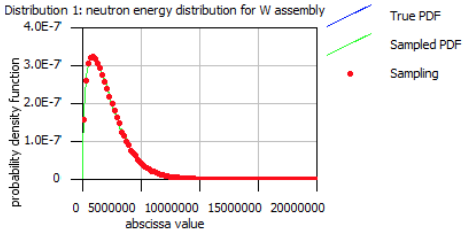
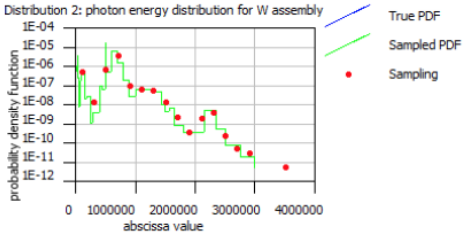


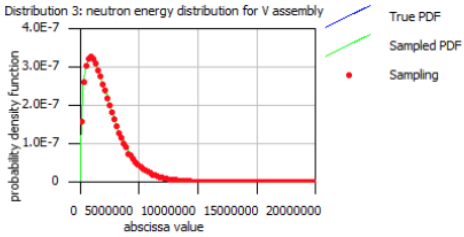
Fig. 4.1.37: Neutron (left) and photon (right) dose rate response functions.



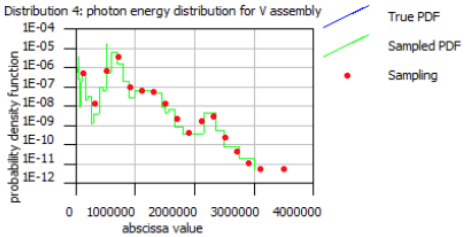
W assembly, neutron source



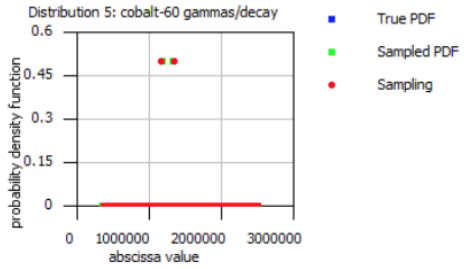
W assembly, photon source



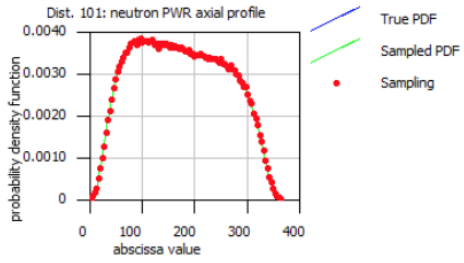
V assembly, neutron source



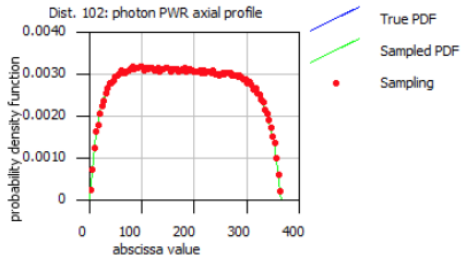
V assembly, photon source



Cobalt-60 source



Neutron Axial Profile



Gamma Axial Profile

Fig. 4.1.38: Distributions in the TN24-P model.

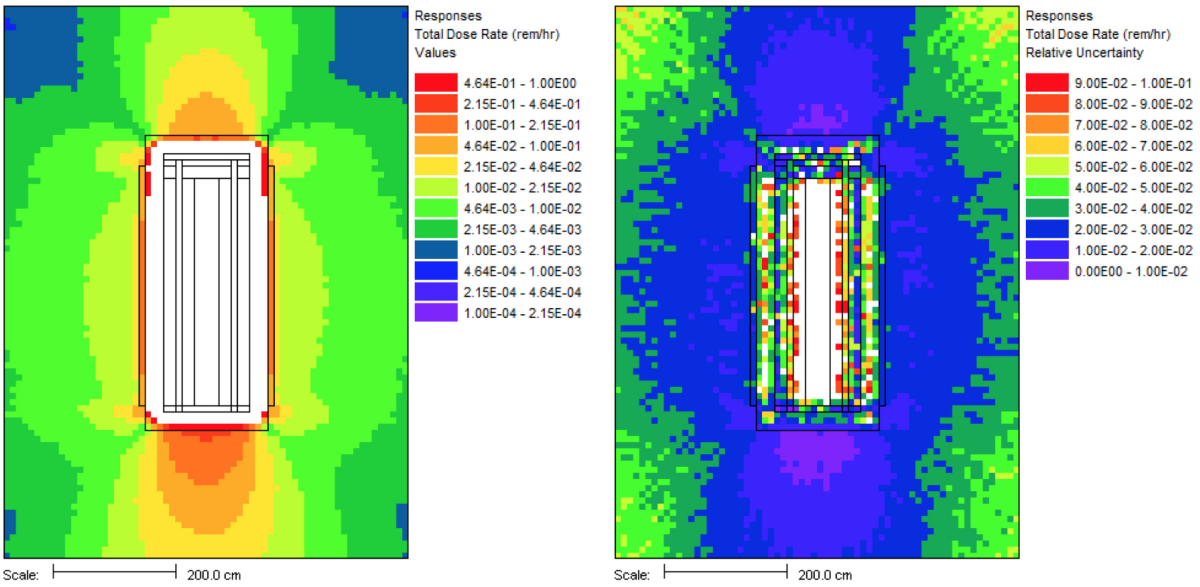


Fig. 4.1.39: Rectilinear mesh tally of total dose rate (rem/hr) and its relative uncertainty along the x=0 plane.

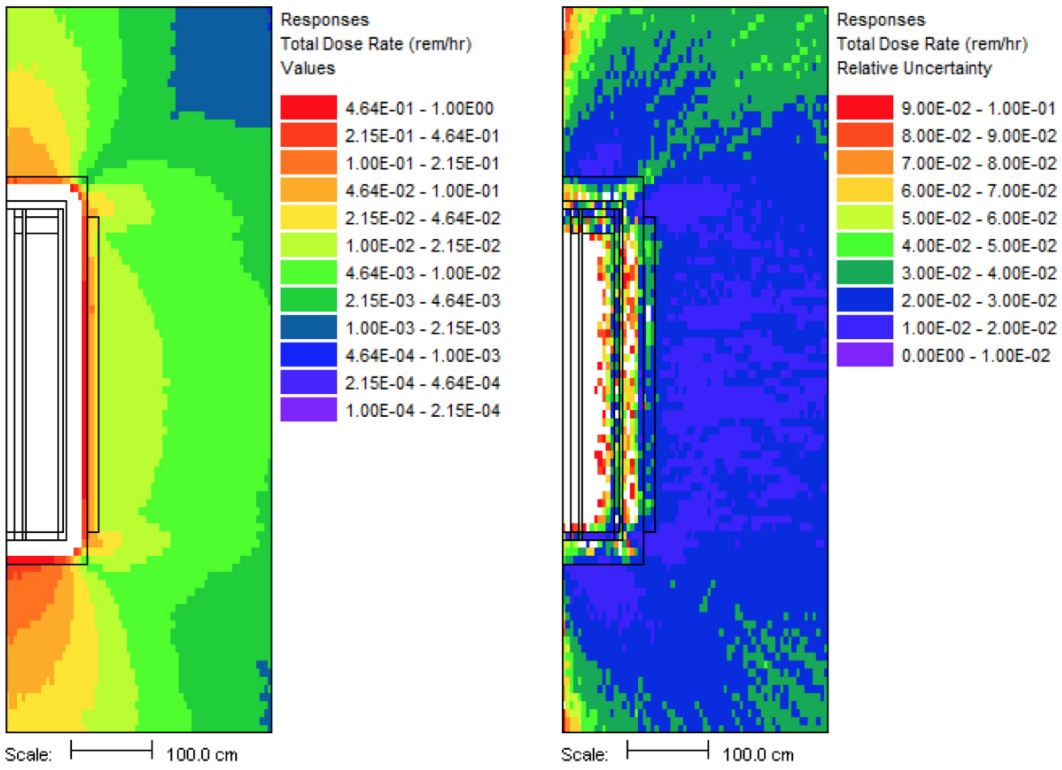


Fig. 4.1.40: Cylindrical mesh tally of total dose rate (rem/hr) and its relative uncertainty along the $\theta = 0°$ plane.

4.2 MAVRIC CAAS CAPABILITY

4.2.1 INTRODUCTION

Modeling criticality accident alarm systems (CAAS) presents challenges since the analysis consists of both a criticality problem and a deep-penetration shielding problem [CAASPLMP10, CAASPPJ09]. Modern codes are typically optimized to handle one of those types of problems but usually not both. The two problems also differ in scale-the criticality problem depends on materials relatively close to the fissionable materials while the shielding problem can cover a much larger range. SCALE now contains fully three-dimensional tools to perform both parts of a CAAS analysis.

CAAS analysis can be performed with SCALE using the KENO-VI criticality code and the MAVRIC shielding sequence. First, the fission distribution (in space and energy) is determined by KENO-VI. This distribution is saved to a file using a user-specified three-dimensional mesh grid and an energy structure from the cross section library (or a user-defined energy structure). MAVRIC then uses the fission distribution as the source for a shielding calculation. MAVRIC is designed to implement advanced variance reduction methods to calculate dose rates or detector responses for difficult shielding problems.

For different types of shielding and different combinations of sources and detector locations, different strategies can be used within SCALE [CAASPW12, CAASWODonnellN12]. Due to the way that cross sections for neutron reactions that create photons are stored in ENDF, some of the parameters used in the CAAS capability have changed since SCALE 6.1. Please be sure to understand and follow the new guidance for the correct accounting of secondary photons from neutron reactions [CAASMP13a, CAASMP13b, CAASMP13c].

4.2.2 METHODS

The CAAS capability in SCALE is a two-step approach using KENO-VI and MAVRIC. The first step is the determination of the source distribution, done with the CSAS6 sequence which uses the KENO-VI functional module. Along with calculating the system k_{eff}, KENO-VI has been modified to now accumulate the fission distribution over the non-skip generations. This information is collected on a three-dimensional Cartesian mesh that overlays the physical geometry model and is saved as a Monaco mesh tally file. A utility program is used to convert the mesh tally into a Monaco mesh source.

The mesh source is then used in the second step as the source term in MAVRIC. The absolute source strength is set by the user based on the total number of fissions (based on the total power released) during the criticality excursion. Further neutron multiplication should be prevented in the MAVRIC transport calculation. Because the fission neutrons have already been accounted for in the KENO-VI calculation, failure to suppress neutron multiplication in the MARVIC sequence would lead to incorrect flux estimates. In addition, if further fissions were allowed, Monaco would add neutrons to its particle bank faster than they could be removed (since the system is at or above critical) and the simulation may never finish.

For the transport part of a CAAS analysis, MAVRIC can be optimized to calculate one specific detector response at one location using CADIS or can be optimized to calculate multiple responses/locations with roughly the same relative uncertainty using FW-CADIS. For calculating mesh tallies of fluxes or dose rates, MAVRIC also uses FW-CADIS to help balance the Monaco Monte Carlo calculation such that low flux voxels are computed with about the same relative uncertainty as high flux voxels.

With this two-step approach, users will have a great deal of flexibility in modeling CAAS problems. The CSAS6 step and the MAVRIC step could both use the same geometry and materials definitions or could have different levels of detail included in each. The fission source distribution from one CSAS6 calculation could be used in a number of different MAVRIC building/detector models, with each MAVRIC calculation being optimized for a given type of detector.

4.2.3 USER INPUT

The user can create either one input file containing both the CSAS6 and MAVRIC calculations or can create two input files-one for each sequence. The materials and geometry for these two models could be the same but do not have to be. For example, the CSAS6 sequence might only contain the materials and geometry important to the criticality calculation. Note, however, that the critical source geometry and materials should be modeled identically in both problems. This allows users greater flexibility in modeling their specific problems.

4.2.3.1 KENO-VI input

For the criticality problem, the only extra input a user needs to supply is the keyword *cds=yes* in the parameter block and a spatial mesh around all of the fissionable materials of the problem in its own gridGeometry block. Standard input for KENO-VI is described in the KENO-VI chapter and CSAS6 chapter. The mesh used for the fission source distribution is input using the *read gridGeometry *id** block, where *id* is an identification number for that grid. Note that only one grid can be specified, but that may change in the future. The cells of the mesh are specified in each dimension separately by either (1) listing all of the planes bounding the cells (keyword "xplanes" followed by an "end"), (2) using keyword *xLinear *n* *a* *b** to specify *n* cells between *a* and *b*, or by (3) specifying the minimum plane, the maximum plane, and how many cells to make in that dimension (*xmin=*, *xmax=*, *numXCells=*). The keywords *xplanes* and *xLinear* can be used together and multiple times. Similar keywords are used for the *y*- and *z*-dimensions. An example CSAS6 input file that collects the fission distribution information would be as follows:

```
=csas6
CAAS Example
v7-238n
read composition
    ...
end composition
read parameters
    ...
    cds=yes
end parameters
read geometry
    ...
end geometry
read gridGeometry 1
    title="Mesh for Collecting Fission Source"
    xLinear 13 0.0 78.0
    yplanes 0 8 16 24 32 34 36 38 40 48 56 64 72 end
    zLinear 10 -2.54 77.46
end gridGeometry
end data
end
```

The fission source distribution collected by KENO-VI is saved to a Monaco mesh tally file and copied back to the home area with the name **problemName*.fissionSource.3dmap*. This file can be viewed with the Mesh File Viewer capability of Fulcrum that comes with SCALE. Note that the finer the mesh spacing is the more generations/histories will have to be simulated by the criticality calculation in order to reduce the stochastic uncertainty in each mesh voxel of the distribution. Regardless of the mesh size, creation of a fission mesh source file will take more iterations than the number required to find k_{eff}. KENO-VI also saves the value of the system the average number of neutrons per fission, in a file called **problemName*.kenoNuBar.txt*. This value is needed to properly determine the source strength.

4.2.3.2 Mesh tally to mesh source conversion

A utility program is used to convert the Monaco mesh tally file into a Monaco mesh source file. It can be part of the CSAS6 input file. The user then needs to copy the resulting *.msm file back to his home area.

```
=csas6
...
end

=mt2msm
'fissionSource.3dmap'    ! existing Keno fission source mesh tally
1                         ! which family (for Keno files, there is only 1)
-1                        ! use the whole family (keep all energy groups)
1                         ! particle type for *.msm file (1-neutron, 2-photon)
'fissionSource.msm'      ! name of newly created mesh source map file
end

=shell
  copy fissionSource.msm "C:\mydocu~1\caasExample"
end
```

Details on the conversion utility program are contained in Sect. 4.3 of the MAVRIC manual.

In SCALE 6.1, the fission source distribution mesh tally produced by KENO contained data representing the number of fissions in each mesh cell in each energy group. In SCALE 6.2, the data stored was changed to be the fissions per unit volume - the fission density. This is more consistent with other mesh tallies from Monaco which store flux or dose rates that represent averages over the mesh cells. This change also allows the Mesh File Viewer to display the KENO fission source distribution better. The mt2msm utility program also changed from SCALE 6.1 to SCALE 6.2 to account for the change in what is stored in the Keno mesh tally file. Therefore, **KENO-produced fission source mesh tallies and the mt2msm utility should not be mixed-and-matched across versions of SCALE**. Doing so would result in the final Monaco mesh source file being improperly normalized, which would not represent the KENO fission source distribution and would give incorrect results in subsequent MAVRIC calculations. Because there is not a specific 'version flag' in a mesh tally file or mesh source map file, the user must ensure that they have used the same version of SCALE for both the CSAS6 and MAVRIC sequences any time the CAAS capability is used.

4.2.3.3 MAVRIC input

The input for the MAVRIC portion of the CAAS problem should include the materials and geometry of the criticality model, use the fission distribution as a source, set the source strength, and set any optional modifiers to the source to change its location or add fission photons. The cross section library used by the MAVRIC calculation does not need to have the same group structure as the fission distribution. MAVRIC will automatically convert the fission source group structure to match the group structure of its cross section library.

The shielding calculation needs to specify that the source is the fission distribution file, which is typically *fissionSource.msm*. The total source strength can be specified by either the number of fissions in the criticality accident (fission rate or total number) or by the number of released neutrons (the fission rate multiplied by $\bar{\nu}$ per fission). The value of $\bar{\nu}$ will be read from the file *kenoNuBar.txt* in the SCALE temporary directory if it is not given in the source input with the keyword *nu-bar=*. The mesh source can also be placed at different coordinates in the geometry using the *origin x= *x* y= *y* z=*z** keywords, if a different reference frame was used with the criticality geometry model that created the mesh source. Rotations of mesh sources are not available at this time. It is also recommended to use filters in the source block to define the source, such as the *mixture=* filter to only allow source sampling from a specific mixture since the mesh source can be transformed from its original origin or meshes can cover non-fissionable materials.

For example, using a KENO-VI fission distribution, placing it somewhere in the MAVRIC model and setting the source strength (in neutrons/s) to correspond to 10^{17} fission/s would look like

```
=shell
    copy "C:\mydocu~1\caasExample\kenoInput.kenoNuBar.txt"  kenoNuBar.txt
end

=mavric
...
read sources
    src 1
        meshSourceFile "C:\mydocu~1\caasExample\fissionSource.msm"
            origin x=600 y=650 z=400
        fissions=1.0e17
    end src
end sources
...
end data
end
```

The source strength in neutrons/s will be calculated by MAVRIC to be the fission rate multiplied by the value of read from the "kenoNuBar.txt" file. The neutron strength could have alternatively been specified using the standard source strength keyword "strength=2.5e17" (for an example with the average number of neutrons per fission of 2.5).

The Monte Carlo functional module used by MAVRIC, Monaco, is a fixed-source code. Unless told otherwise, neutrons will multiply in fissionable materials. Since all of the neutrons were part of the source, neutron multiplication should not be allowed and MAVRIC should be run with the keyword "fissionMult=0" in the parameters block. For systems at or near critical without the "fissionMult=0" keyword, Monaco simulations may not end since neutrons will be added to the particle bank at the same rate they leave the system or get killed.

The shielding calculation can be run using standard variance reduction methods (such as path length stretching, user-defined weight windows based on geometry regions, and user-defined source biasing) or using the automated tools which employ approximate discrete-ordinates calculations to determine the space/energy weight windows as well as a biased source distribution in space and energy. The automated tools can be used to optimize the shielding calculation to determine one specific tally using CADIS or several separate tallies or a mesh tally over a large volume of the problem space using FW-CADIS. When using these advanced variance reduction methods, remember to include planes in the discrete-ordinates mesh definition that correspond to the planes in the fission distribution that the source is based on. If they are not included, MAVRIC will resample the fission source on the discrete-ordinates mesh it is using for the importance map, possibly smearing or reducing the original resolution of the fission distribution.

4.2.4 EXAMPLE PROBLEM

Consider the Jezebel critical plutonium sphere experiment, shown in Fig. 4.2.1, taking place inside a simple fictitious building, shown in Fig. 4.2.2. The building has two rooms: an experiment room and a control room. In the control room there is a criticality alarm detector, and it is positioned furthest from the entry to the experiment room. For this example, assume that a criticality excursion results in a total of 10^{18} fissions. This example will calculate the neutron and photon doses seen by a detector in the control room, as well as calculate a dose map for the entire building.

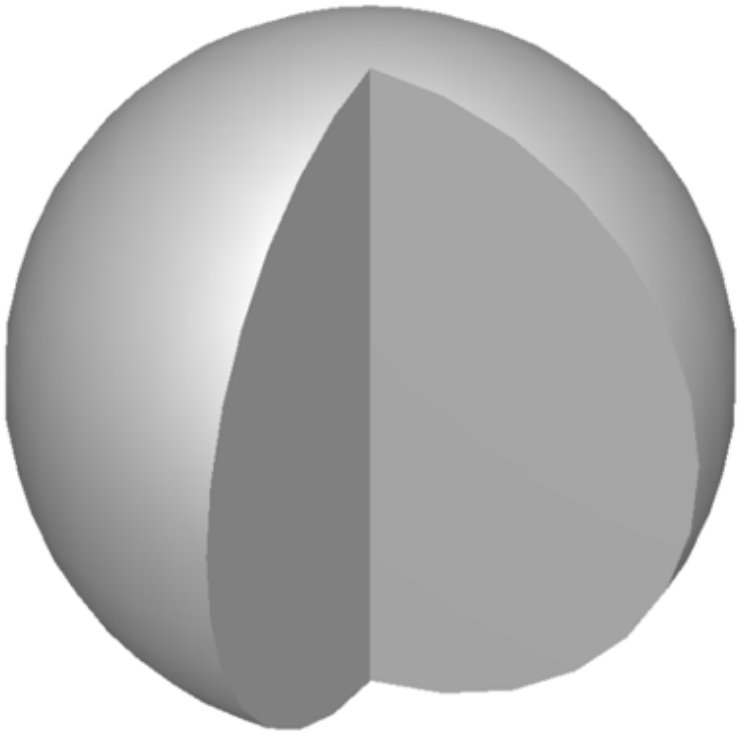


Fig. 4.2.1: Cutaway view of Jezebel.

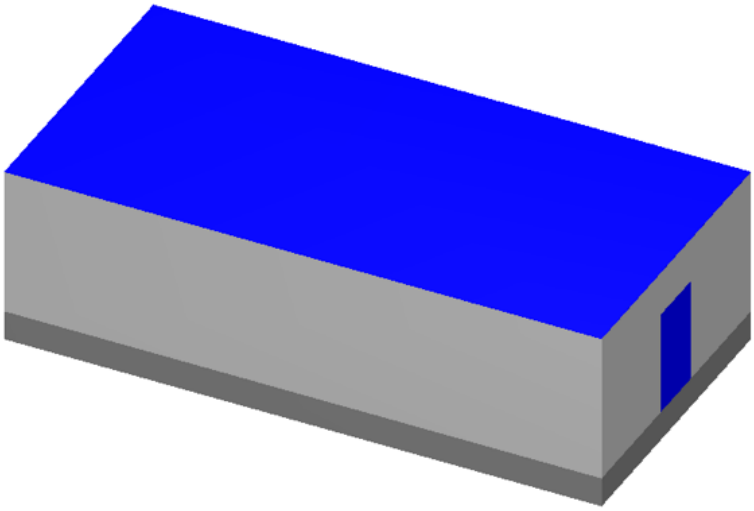


Fig. 4.2.2: Simple two-room building.

4.2.4.1 KENO-VI criticality and fission source distribution

For the criticality calculation, consider just a bare sphere of plutonium, with a radius of 6.38493 cm. Atom densities (atoms/b·Pu 0.037047; ^{240}Pu 0.0017512; ^{241}Pu 0.00011674; and Cu 0.0013752. This can be easily modeled as a sphere at the origin. For collecting the fission distribution, a uniform mesh grid can be constructed around the sphere, extending 7 cm in each direction, with a $1 \times 1 \times 1$ cm voxel size. The first portion of the input files `mavric.caasA.inp` and `mavric.caasB.inp` looks like the following:

```
=csas6
Dose Rates from a Jezebel Accident in a Block Building
v7-238n
'------------------------------------------------------------------------------
' Composition Block
'------------------------------------------------------------------------------
read composition
    Pu-239 1 0 0.037047   end
    Pu-240 1 0 0.0017512  end
    Pu-241 1 0 0.00011674 end
    Cu     1 0 0.0013752  end
end composition

'------------------------------------------------------------------------------
' Parameters Block
'------------------------------------------------------------------------------
read parameters
    gen=250 npg=200000 nsk=50 htm=no
    cds=yes
end parameters
'------------------------------------------------------------------------------
' Geometry Block - SCALE standard geometry package (SGGP)
'------------------------------------------------------------------------------
read geometry
    global unit 2
        sphere 1  6.38493
        media 1 1 1 vol=1090.3277
    boundary 1
end geometry

'------------------------------------------------------------------------------
' Grid Block
'------------------------------------------------------------------------------
read gridGeometry 1
    title="Mesh for Collecting Fission Distribution"
    xLinear 14 -7.0 7.0
    yLinear 14 -7.0 7.0
    zLinear 14 -7.0 7.0
end gridGeometry

end data
end

=mt2msm
'fissionSource.3dmap'
1
-1
1
mavric.caas[A/B].fissionSource.msm'
end
```

The results of this 26 minute calculation are shown in Table 4.2.1, and details about the calculated fission distribution are shown in Fig. 4.2.3 and Fig. 4.2.4.

Table 4.2.1: Results of the CSAS6 calculation

Quantity	Value	Uncertainty	
k_{eff}	best estimate system k-eff	1.00024	0.00014
$\bar{\nu}$	system nu bar	3.15671	4.77938E-05

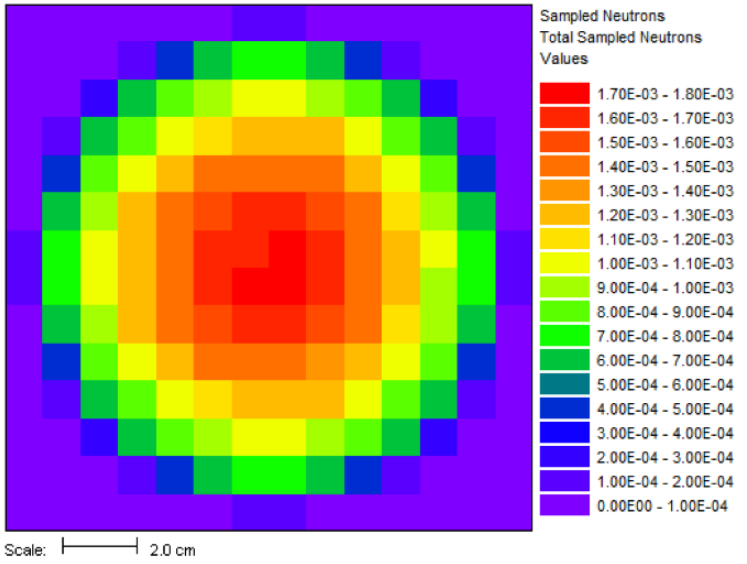


Fig. 4.2.3: Fission source spatial distribution for the center horizontal slice.

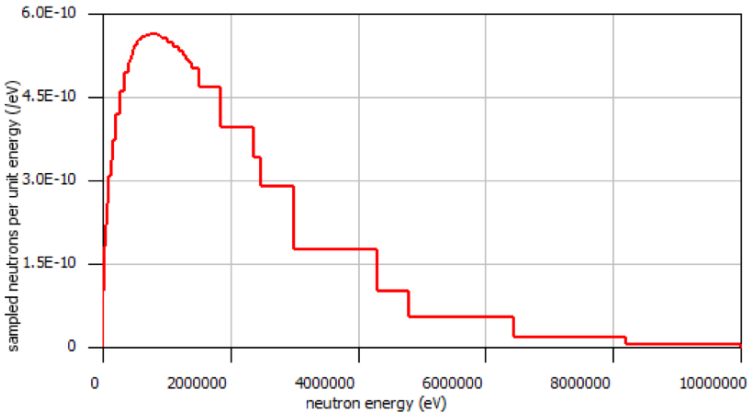


Fig. 4.2.4: Fission source energy distribution for the center voxel.

4.2.4.2 MAVRIC transport calculations

Two MAVRIC calculations will be done-one that calculates the doses seen at the detector and one that computes mesh tallies of doses over the entire building. They will share the same materials, geometry, and source but will have different tally and variance reduction options.

The two-room building will be a simple model using concrete-block walls, a concrete floor, and a steel roof, with dimensions shown in Fig. 4.2.5. The building exterior dimensions are 1200 cm long, 600 cm wide, and 300 cm high above the ground. The exterior and interior walls are all made of a double layer of typical concrete blocks (total of 40 cm thick.) Concrete blocks are typically 39 × 19 × 19 cm and weigh ~13.5 kg, since they have a volume fraction of 33.2%. The floor is made of poured concrete, extending 60 cm into the ground. The roof and the exterior door (120 cm wide and 210 cm tall) are made of 1/8 in. (0.3175 cm) thick steel. The experiment room on the left connects to the control room on the right through a maze that prevents radiation streaming. Assume that the critical experiment was in the center of the experiment room, 100 cm above the floor. Assume the detector in the control room is a 30 cm diameter sphere located at position (1145, 55, 285).

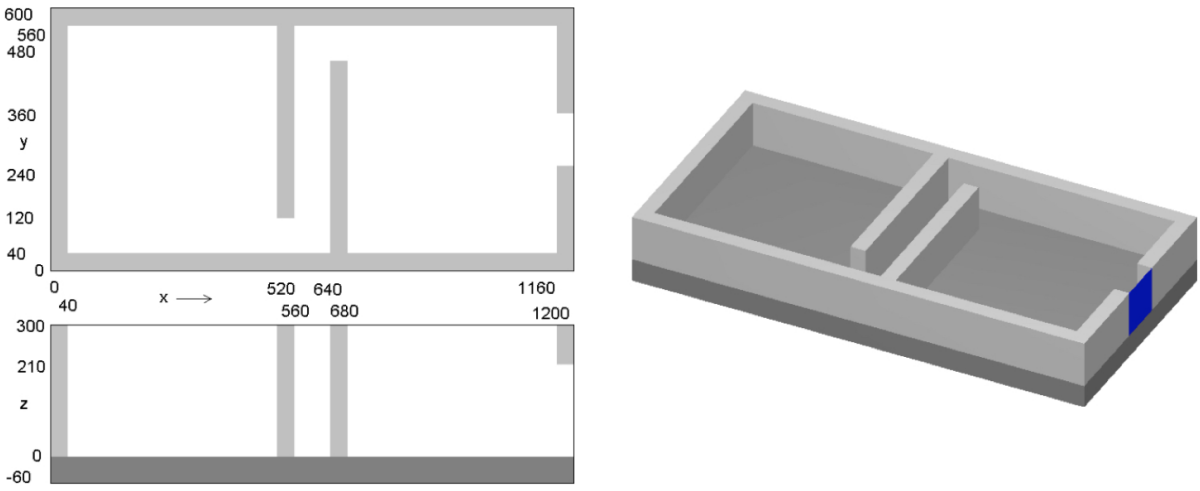


Fig. 4.2.5: Coordinates of the floor, walls, ceiling, and door of the simple block building model (in cm).

If the MAVRIC transport calculation is not in the same file as the CSAS6 calculation, the MAVRIC input would start by moving the KENO-VI results into the SCALE temporary area:

```
=shell
    copy %RTNDIR%\caas.kenovi.fissionSource.msm  fissionSource.msm
    copy %RTNDIR%\caas.kenovi.kenoNuBar.txt      kenoNuBar.txt
end
```

The materials and geometry blocks of the two MAVRIC input files for each of the two calculations, `smplprbs/caas.mavricA.inp` and `smplprbs/caas.mavricB.inp`, look like the following:

```
'------------------------------------------------------------------------------
' Composition Block
'------------------------------------------------------------------------------
read composition
    pu-239 1 0 0.037047   end
```

(continues on next page)

(continued from previous page)

```
    pu-240 1 0 0.0017512  end
    pu-241 1 0 0.00011674 end
    cu     1 0 0.0013752  end
    orconcrete 2 1.0     293.0 end
    orconcrete 3 0.33198 293.0 end
    ss304      4 1.0     293.0 end
end composition

'------------------------------------------------------------------------------
' Geometry Block - SCALE standard geometry package (SGGP)
'------------------------------------------------------------------------------
read geometry
    global unit 1
        com="jezebel"
        sphere   1  6.38493  origin x=280  y=300  z=100

        com="exterior of the building, roof, floor"
        cuboid   10 1200 0    600 0     300.3175 -60.0
        cuboid   11 1200 0    600 0     300.3175 300.0
        cuboid   12 1200 0    600 0     0.0        -60

        com="air space in building - two rooms and maze"
        cuboid   20 1160 40   560 40    300 0

        com="interior walls to form maze to prevent streaming"
        cuboid   21  560 520  560 120   300 0
        cuboid   22  680 640  480  40   300 0

        com="exterior door"
        cuboid   30 1200 1160       360 240   210 0
        cuboid   31 1200 1199.6825  360 240   210 0

        com="detector sphere"
        sphere  40   15.0  origin x=1145 y=55 z=285

        com="jezebel"
        media  1 1   1   vol=1090.3277

        com="walls, roof, floor"
        media  3 1   10 -20 -11 -12  -30
        media  4 1   11
        media  2 1   12

        com="air space (void) and maze walls"
        media  0 1   20 -21 -22   -40  -11 -12  -1
        media  3 1   21                -11 -12
        media  3 1   22                -11 -12

        com="exterior door"
        media  0 1   30 -31
        media  4 1   31

        com="detector"
        media  0 1   40   vol=14137.167

    boundary 10
end geometry
```

The response functions used to compute the doses will be the standard flux-to-dose rate conversion factors for neutrons and photons. These are defined in the definitions block. Note that these responses have units of (rem/hr)/(/cm ^{2}/s).

```
'------------------------------------------------------------------------------
' Definitions Block
```

(continues on next page)

(continued from previous page)

```
'--------------------------------------------------------------------------------
read definitions
    response 5
        title="ANSI (1977) neutron flux-to-dose-rate"
        specialDose=9029
    end response
    response 6
        title="ANSI (1977) photon flux-to-dose-rate"
        specialDose=9504
    end response
end definitions
```

The source used by each MAVRIC simulation will be based on the fission distribution mesh source determined by KENO-VI. The strength of the source can be specified by the total number of fissions that occurred in the criticality event. Fission photons will be added for ^{239}Pu. MAVRIC will determine the total source strength, including the fission photons, from the value of saved by KENO-VI and the multiplicity data from the fission photon data file.

```
'--------------------------------------------------------------------------------
' Sources Block
'--------------------------------------------------------------------------------
read sources
    src 1
        meshSourceFile="fissionSource.msm"
        origin x=280  y=300  z=100
        fissions=1.0e18
        mixture=1
    end src
end sources
```

Note that further multiplication needs to be turned off in MAVRIC using the *fissionMult=0* keyword in the parameter block as shown below.

For the responses from the tallies, MAVRIC usually calculates dose rates (rem/hr) using a source strength in particles/s. For this example problem, instead of a source rate, we used a total number of particles (by specifying the number of fissions). Hence, the computed fluxes will have units of particles/cm^2 and the computed responses using the standard dose responses from the cross section libraries will have units of rem s/hr. To get a dose in rem, the responses need to be multiplied by (3600 s/hr)$^{-1}$. This can be done using the MAVRIC tally multiplier keyword.

Each MAVRIC simulation will need a discrete-ordinates mesh. The planes in each dimension where there are material changes are listed in Table 4.2.2. In addition to these planes, the discrete-ordinates mesh should also subdivide the thick shields in the direction of particle travel. For example, the walls of the maze should be divided to better model the radiation attenuation through the walls in the Denovo calculation. The interior walls of the building will reflect particles, so the first few centimeters are the most important to capture in the importance map. Mesh planes should also be added that correspond to the mesh source after it is placed into the geometry model.

Table 4.2.2: Main planes in the building

x	y	z
0	0	-60
40	40	0
520	120	210

continues on next page

Table 4.2.2 – continued from previous page

x	y	z
560	240	300
640	360	300.318
680	480	
1160	560	
1199.68	600	
1200		

Detector doses using CADIS

The grid geometry for this calculation should also include planes that bound the adjoint source, which is the detector area (these values are shown in brackets [] below). The definitions block in `smplprbs/caas.mavricA.inp` also includes the location of the center of the detector, which is used in the adjoint source description.

```
location 1
     position 1145 55 285
 end location

 gridGeometry 1
     title="mesh for discrete ordinates 57 x 47 x 31 = 83049"
     xplanes 0     10 20 30 35
             40    120 160 240
                   270 272 274 276 278 280 282 284 286 288 290
                   360 440
             520   525 530 550 555
             560   600
             640   645 650 670 675
             680   760 840 920 1000 1080
                   [1130 1140 1150]
             1160  1165 1170 1180 1190
             1199.6825
             1200  end
     yplanes 0     10 20 30 35
             40    [50 60] 70
             120   125 130 140 200
             240   280 290 292 294 296 298 300 302 304 306 308 310
                   320
             360   440 460 470 475
             480
             560   565 570 580 590
             600   end
     zplanes -60   -30 -20 -10 -5
             0     45
                   90 92 94 96 98 100 102 104 106 108 110
                   140 175
             210   255  [280 290]
             300
             300.3175   end
 end gridGeometry
```

The tallies are region tallies over the detector region (the 10th media card in unit 1) using the appropriate response function for the particle type of the tally. The volume of the detector sphere needs to be listed in the geometry block so that the fluxes and tallies will be correctly computed. The importance map uses standard CADIS to bias the particles towards the detectors, optimizing the calculation of the total dose (by listing both response functions together, the total response will be used in the adjoint source).

```
'--------------------------------------------------------------------------------
' Tallies Block
'--------------------------------------------------------------------------------
read tallies
    regionTally 1
        title="Doses seen by the detector"
        neutron
        unit=1 region=10
        responseID=5
        multiplier=2.777778e-4
    end regionTally
    regionTally 2
        title="Doses seen by the detector"
        photon
        unit=1 region=10
        responseID=6
        multiplier=2.777778e-4
    end regionTally
end tallies

'--------------------------------------------------------------------------------
' Parameters Block - 3 min batch
'--------------------------------------------------------------------------------
read parameters
    randomSeed=3263827
    perBatch=654000  batches=40
    fissionMult=0
end parameters

'--------------------------------------------------------------------------------
' Importance Map Block
'--------------------------------------------------------------------------------
read importanceMap
    gridGeometryID=1
    adjointSource 1
        locationID=1
        responseIDs 5 6 end
    end adjointSource
end importanceMap
```

The results of this example problem are shown in Table 4.2.3. Calculation times were 12 minutes for Denovo and 135 minutes for Monaco. Note that the uncertainty for the photon dose is much higher than the neutron dose uncertainty. This is because the simulation was optimized for the calculation of total dose, and the photon component of the total dose is less than 2%. Had a separate calculation been done that used an adjoint source of just the photon response, the photon dose rate uncertainty would have been much smaller but at the expense of the neutron dose rate uncertainty. A single calculation could have also been performed using two adjoint sources, one using the neutron dose response and one using the photon dose response, and forward weighting to help calculate each component of dose with more uniform relative uncertainties.

Table 4.2.3: Doses seen by the detector

	Value	**Rel.**
Dose	**(rem)**	**Unc.**
neutron	1539	0.78%
photon	30.0	8.00%

Dose map using FW-CADIS

The grid geometry for this calculation does not need extra planes around the detector. The grid geometry in `smplprbs/caas.mavricB.inp` looks like the following:

```
gridGeometry 1
        title="mesh for discrete ordinates 46x36x23 = 38088"
        xplanes 0     10 20 30 35
                40    120 160 240
                      270 272 274 276 278 280 282 284 286 288 290
                      360 440
                520   525 530 550 555
                560   600
                640   645 650 670 675
                680   760 840 920 1000 1080
                1160  1165 1170 1180 1190
                1199.6825
                1200  end
        yplanes 0     10 20 30 35
                40
                120   125 130 140 200
                240   280 290 292 294 296 298 300 302 304 306 308 310
                      320
                360   440 460 470 475
                480
                560   565 570 580 590
                600   end
        zplanes -60   -30 -20 -10 -5
                0     45
                      90 92 94 96 98 100 102 104 106 108 110
                      140 175
                210   255
                300   300.3175   end
end gridGeometry
```

A second grid geometry also needs to be added to the definitions block for the mesh tally to use.

```
gridGeometry 2
    title="mesh for uniform mesh tally - 40x40x30 cm voxels"
    xLinear 30 0.0 1200.0
    yLinear 15 0.0  600.0
    zLinear 10 0.0  300.0
end gridGeometry
```

The mesh tallies for each particle type are listed, along with the appropriate response function. The importance map uses FW-CADIS to better spread the particles out over the entire geometry, optimized for the calculation of total dose in the void regions.

```
'------------------------------------------------------------------------------
' Tallies Block
'------------------------------------------------------------------------------
read tallies
    meshTally 1
        title="Neutron doses mapped over the entire building"
        neutron
        gridGeometryID=2
        responseID=5
        noGroupFluxes
        multiplier=2.777778e-4
    end meshTally
    meshTally 2
        title="Photon doses mapped over the entire building"
        photon
```

(continues on next page)

(continued from previous page)

```
        gridGeometryID=2
        responseID=6
        noGroupFluxes
        multiplier=2.777778e-4
    end meshTally
end tallies

'-------------------------------------------------------------------------------
' Parameters Block - 3 min batch
'-------------------------------------------------------------------------------
read parameters
    randomSeed=3263827
    perBatch=669000  batches=40
    fissionMult=0
end parameters

'-------------------------------------------------------------------------------
' Importance Map Block
'-------------------------------------------------------------------------------
read importanceMap
    gridGeometryID=1
    adjointSource 1
        boundingBox 1200 0  600 0  300.3175 -60.0
        responseIDs 5 6 end
        mixture=0
    end adjointSource
    respWeighting
end importanceMap
```

This calculation used 5 minutes for the forward Denovo S_N calculation, 6 minutes for the adjoint Denovo, and 120 minutes for the Monaco Monte Carlo. The resulting mesh tally is shown in Fig. 4.2.6. The uncertainties for each voxel are shown in Fig. 4.2.7. Note that the dose in the voxel containing the detector (not shown in the figures) is 1.552×10^3 rem with a relative uncertainty of 3.8%, closely matching the value calculated with the first MAVRIC simulation.

Similar to the detector doses above, a single calculation could have also been performed for the dose maps using two adjoint sources, one using the neutron dose response and one using the photon dose response, and forward weighting to help calculate each component of dose with more uniform relative uncertainties.

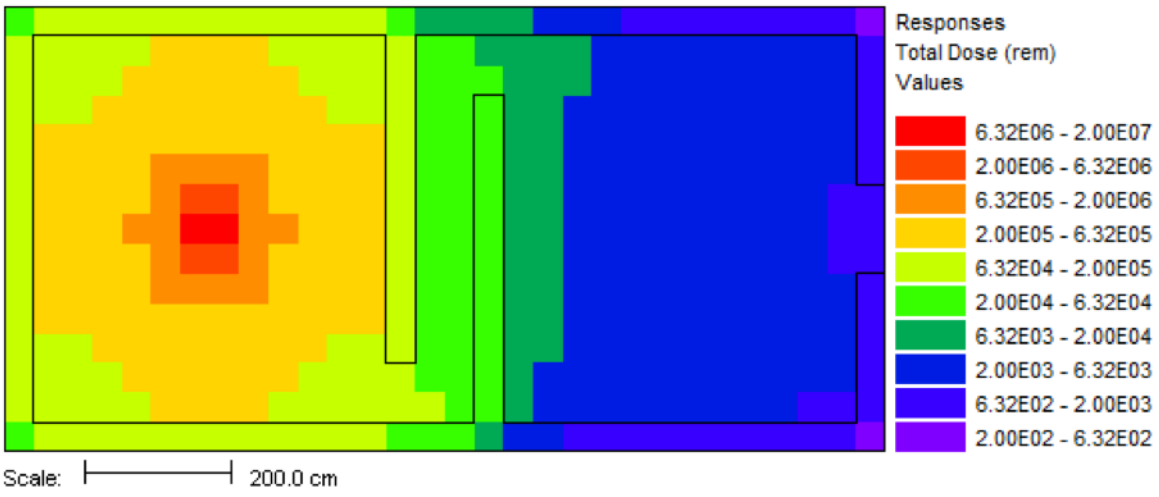


Fig. 4.2.6: Dose (rem) results for the z=100 cm plane (containing the source).

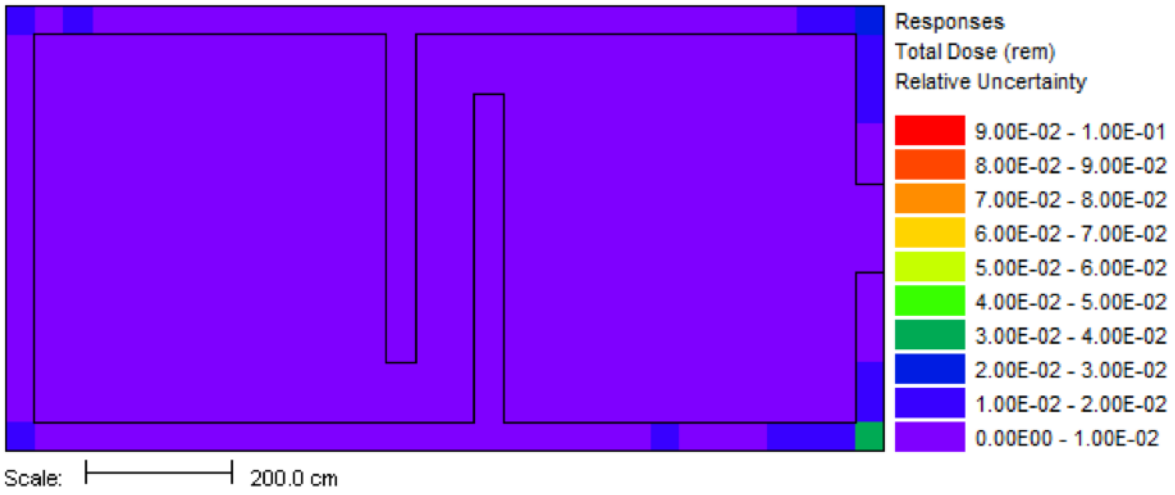


Fig. 4.2.7: Relative uncertainties in the dose, most less than 1%, for the z=100 cm plane.

4.2.5 SUMMARY

SCALE now has the capability to do detailed simulations of criticality accident alarm systems. The advanced variance reduction capabilities of the MAVRIC radiation transport sequence allow for the full three-dimensional analysis of CAAS problems in reasonable amounts of computer time. This enables the use of more realistic source definitions, such as a detailed spatial/energy dependent fission source distribution determined by the KENO-VI criticality code, and the critical assembly itself can be included in the transport model.

4.3 MAVRIC UTILITIES

4.3.1 INTRODUCTION

Several utilities are provided to aid users in dealing with some of the output files produced by Monaco and MAVRIC. These utilities were developed at ORNL for specific projects and have been added to SCALE so that all users can benefit. More utilities have been added that deal with Denovo, including the older *.varscl files for flux output used in SCALE 6.1 and the current binary *.dff file for flux output used in SCALE 6.2. These tools do not have the modern block/keyword input structure but instead have a fixed format, which is fairly simple since each utility is made for a very specific function.

Each is described in the following sections. Five sample problems, `mavricUtilities1.inp`, `mavricUtilities2.inp`, `mavricUtilities3.inp`, `mavricUtilities4.inp` and `mavricUtilities5.inp` demonstrate the use of some of these. For all of these utilities, filenames should be enclosed in quotes.

4.3.2 UTILITIES WORKING WITH MONACO MESH TALLY (3DMAP) FILES

mt2ascii	Convert a mesh tally into an ASCII text file.
mt2msl	Convert a mesh tally into a mesh source lite.
mt2msm mt2silo	Convert a mesh tally into a mesh source. Convert a mesh tally file into a Silo file for VisIt.
mt2vtk	Convert one dataset of one family in a mesh tally to VTK format.

continues on next page

Table 4.3.1 – continued from previous page

mtAdder	Add several Monaco mesh tally files together into one mesh tally.
mtAverager	Average several Monaco mesh tally files into one mesh tally.
mtBinOp	Binary operation of mesh tally files: sum, difference, product, and ratio.
mtDisp	Display the basics of a mesh tally file.
mtExpand	Expand a space-only mesh from a mesh tally with an energy function.
mtFilter	Perform various filters on a mesh tally file.
mtInv	Invert all of the values in a mesh tally.
mtMask	Keep only or remove specified voxels of a mesh tally based on geometry.
mtMinMax	Find the location/value of the min or max of each real mesh in a mesh tally.
mtMultiply	Multiply a mesh tally by a constant factor.
mtPull	Pull values from certain voxels out of a mesh tally file.
mtRefine	Subdivide the mesh into smaller meshes for better visualization.
mtResp	Apply a response function to one family of a mesh tally file.
mtSplit	Split off part of a mesh tally file into a separate mesh tally file.

mt2ascii - Convert a mesh tally into an ASCII text file.

Intended use: Since mesh tally files are in binary, the viewer can be used to list mesh values. To get the values from the entire file, this utility can be used to create an ASCII text version.

Input: The mesh tally file name and the filename for the resulting ASCII file

Output: An ASCII formatted file

Example:

```
=mt2ascii
"/optional/path/meshTallyFilename.3dmap"  ! the mesh tally
"/optional/path/outputFilename.txt"       ! output file name
end
```

mt2msl - Convert a mesh tally into a mesh source lite.

Intended use: Convert a fissionSource.3dmap mesh tally computed by KENO into a meshSoureLite (*.msl) file that can be used by a subsequent KENO run using starting source type nst=9.

Input: Name of mesh tally (*.3dmap) file

Output: A mesh source lite (*.msl) file

Example:

```
=mt2msl
"input.3dmap"    ! mesh file (*.3dmap) name
1                ! family
14               ! group, or 0 for family total
"result.msl"     ! mesh source lite (*.msl) file name
end
```

mt2msm - Convert a mesh tally into a mesh source.

Intended use: Turn a tally of fission rate data into a mesh source file. Mesh tallies are stored in a generic *.3dmap format, which consist of several families, each with one or more datasets. A typical mesh tally (without the “noGroupFluxes” keyword) contains three families: the neutron fluxes with each energy group

as a dataset, the photon fluxes with each energy group as a data set, and the responses with each response as a dataset. This program uses the spatial information of the mesh tally and combines a user-given energy distribution for all voxels. A second way to use this program is to use a whole family (all the energy groups) without a user-given energy distribution.

Input:

Line 1: filename of mesh tally
Line 2: which family of tally
Line 3: which dataset of that family (or 0 for sum of family)
Line 4: source type: 1-neutron, 2-photon
Line 5: number of bins for mesh source
Lines : energy (eV) and pdf values
Lines : energy (eV) and pdf values

... ...

Lines : energy (eV) and pdf values
Line : energy (eV)
Line last: desired output name

Input:

Line 1: filename of mesh tally
Line 2: which family of tally
Line 3: -1 (meaning use the whole family)
Line 4: source type: 1-neutron, 2-photon
Line 5: desired output name

Output: The resulting mesh source stored with the desired filename

---

Note: Statistics of mesh tally are discarded.

Example:

```
=shell
   cp ${RTNDIR}/barrel1.mt1.3dmap .
end

=mt2msm
"barrel1.mt1.3dmap"    ! mesh tally
```

(continues on next page)

(continued from previous page)

```
3                     ! mesh tally family (1-n, 2-p, 3-responses)
1                     ! real mesh in that family (0 means total of family)
1                     ! mesh source particle type  1-neutron, 2-photon
143                   ! number of bins in binned histogram distribution
1.9640E+07 1.29403E-08 ! E_1  pdf_1
1.7332E+07 4.60970E-07 ! E_2  pdf_2
1.6905E+07 2.56619E-06 ! E_3  pdf_3
   ...        ...
1.2341E+03 5.28408E-06 ! E_142  pdf_142
9.6112E+02 1.77756E-06 ! E_143  pdf_143
7.4852E+02             ! E_144
"barrel.fission.msm"   ! output filename
end

=shell
   cp barrel.fission.msm ${RTNDIR}
end
```

Example:

```
=mt2msm
"fissionSource.3dmap"
1                      ! neutron flux (for KENO 3dmap files there is only one family)
-1                     ! use the whole family (keep all the energy groups)
1                      ! particle type (neutron)
"caas.kenovi.fissionSource.msm"
end
```

In SCALE 6.1, the fission source distribution mesh tally produced by KENO contained data representing the number of fissions in each mesh cell in each energy group.

In SCALE 6.2, the data stored was changed to be the fissions per unit volume–the fission density. This is more consistent with other mesh tallies from Monaco which store flux or dose rates that represent averages over the mesh cells. This change also allows the MeshFileViewer to display the KENO fission source distribution better. The mt2msm utility program also changed from SCALE 6.1 to SCALE 6.2 to account for the change in what is stored in the KENO mesh tally file. Therefore, **KENO-produced fission source mesh tallies and the mt2msm utility should not be mixed-and-matched across versions of SCALE.** Doing so would result in the final Monaco mesh source file being improperly normalized, which would not properly represent the KENO fission source distribution and would give incorrect results in subsequent MAVRIC calculations. Because there is not a specific 'version flag' in a mesh tally file or mesh source map file, the user must ensure that they have used the same version of SCALE for both the CSAS6 and MAVRIC sequences any time the CAAS capability is used.

mt2silo - Convert a mesh tally file into a Silo file for VisIt.

Input: Name of mesh file (*.3dmap), name of a Silo file, and a format

Output: A new Silo file

Note: For format, use either 2 (PDB) or 7 (HDF5).

Example:

```
=mt2silo
"perfect.3dmap"        ! the existing mesh tally
"perfect.silo"         ! the new silo file
7                      ! format - HDF5
end
```

mt2vtk - Convert one dataset of one family in a mesh tally to VTK format.

Intended use: This is a way to transfer Monaco mesh tally data into a common format that can be used by many data visualization packages, including VisIt. Mesh tallies are stored in a generic *.3dmap format, which consist of several families, each with one or more datasets. A typical mesh tally contains three families: the neutron fluxes with each energy group as a dataset, the photon fluxes with each energy group as a data set, and the responses with each response as a dataset. This program selects one dataset of one family and saves the data (and optionally the absolute uncertainties) in an ASCII file using a VTK file format.

Input: The mesh file name, which family, which dataset of that family, whether or not to include absolute uncertainties and the filename for the resulting VTK file

Output: An ASCII VTK-formatted file

Example:

```
=mt2vtk
"/optional/path/meshTallyFilename.3dmap"  ! the mesh tally
1                                         ! neutron flux family
5                                         ! energy group 5
true                                      ! include uncertainties
"/optional/path/outputFilename.vtk"       ! output file name
end
```

Example:

```
=mt2vtk
"/optional/path/meshTallyFilename.3dmap"  ! the mesh tally
3                                         ! the response family
1                                         ! first response
false                                     ! do not include uncertainties
"/optional/path/outputFilename.vtk"       ! output file name
end
```

mtAdder - Add several Monaco mesh tally files together into one mesh tally.

Intended use: Add mesh tally results from different sources into one tally. The resulting mesh tally is the sum of all the components in the several mesh tallies-fluxes are added and responses are added. For example, two runs of MAVRIC from two different sources can be made. The mesh tally results can then be added together, getting the total fluxes and total responses from each.

Input: The number of files, followed by the list of mesh tally filenames to add, then the name of the total mesh tally

```
  =mtAdder
  n
  "filename_1"
  "filename_2"
  ...
  "filename_n"
  "resultFilename"
end
```

Output: A new mesh tally file

Note: All of the mesh tally files must be the same size and shape (number of families, x cells, y cells, z cells, and energy groups in each family) and have the same number of responses. Responses (if any) must be consistent to calculate meaningful results.

Example:

```
=mtAdder
3
"meshFilename_1.3dmap"
"meshFilename_2.3dmap"
"meshFilename_3.3dmap"
"meshFilenameTotal.3dmap"
end
```

mtAverager - Average several Monaco mesh tally files into one mesh tally.

Intended use: Combine (average) separate runs of the same problem with different random number seeds into one tally. For example, if a user does 10 separate runs of the same problem (poor man's parallel) and wants to combine the results as if they were from one run, an average is needed. The average and uncertainties are weighted by the number of histories in each run, to maintain proper statistics.

Input: The number of files, each filename and how many histories, then the name of the total mesh tally

```
=mtAverager
n
"filename_1"   histories_1
"filename_2"   histories_2
...           ...
"filename_n"   histories_n
"resultFilename"
end
```

Output: A new mesh tally file

Note: All of the mesh tally files must be the same size and shape (number of families, x cells, y cells, z cells, and energy groups in each family) and have the same number of responses. Responses (if any) must be consistent to calculate meaningful results.

Example:

```
=mtAverager
3
"meshFilename_1.3dmap"   800000
"meshFilename_2.3dmap"   900000
"meshFilename_3.3dmap"   800000
"/home/area/meshFilename.ave.3dmap"
end
```

mtBinOp - Binary operation of mesh tally files: sum, difference, product, and ratio.

Intended use: Apply simple math to the results stored in mesh tally files

Input: The first mesh tally, the operator: add (or sum, +), subtract (or difference, -), multiply (or product, x, *) and divide (or ratio, ÷," /"), the second mesh tally name and then name of the resulting mesh tally file.

Output: A new mesh tally file

Note: Uncertainties are propagated assuming the two mesh tallies are uncorrelated, which may not always be a good assumption. Mesh tallies must have the same grid structure and number of families and groups. Dataset names in the results are inherited from the first mesh tally and may not make sense after the operation. When using the / (slash) for division, enclose it in quotes ("/").

```
=mtBinOp
"neutron.3dmap"     ! first operand
divide              ! operation
"total.3dmap"       ! second operand
"ratio.3dmap"       ! output file name
end
```

mtDisp - Display the basics of a mesh tally file.

Input: A mesh tally (*.3dmap) file

Output: Some of the basic details of mesh file

Example:

```
=mtDisp
"simulation.mt2.dff"    ! existing mesh file
end
```

mtExpand - Expand a space-only mesh from a mesh tally with an energy function

Input: A mesh tally (*.3dmap) file and some parameters

Output: A mesh file similar to a mesh source but with uncertainty

Example:

```
=mtExpand
'activate.mt1.3dmap'
2 1  ! response family, first response - cobalt activate rate
true ! multiply by voxel volumes
2    ! make photon source
19   ! groups
2.00E+07 0
1.00E+07 0
8.00E+06 0
6.50E+06 0
5.00E+06 0
4.00E+06 0
3.00E+06 0
2.50E+06 0
2.00E+06 0
1.66E+06 0.5
1.33E+06 1.5
1.00E+06 0
8.00E+05 0
6.00E+05 0
4.00E+05 0
3.00E+05 0
2.00E+05 0
1.00E+05 0
4.50E+04 0
1.00E+04
'photonSource.3dmap'
end
```

mtFilter - Perform various filters on a mesh tally file.

Input: A *.3dmap mesh tally file and a group-wise response function

Output: A *.3dmap mesh tally file

Note: Three basics types of filters: (0) flattening filter, (1) high-pass filter, (2) low-pass filter. For types 1 and 2, the values plus a given number of standard deviations will be compared to the criteria. The input

list depends on filter type. Types 1 and 2 require a value and a number of standard deviations (n_sigma). A flattening filter turns any positive value into the value of "1.0".

Filtering performed based on following comparisons;

value + n_sigmas*abs_unc > minValue (high-pass)

or

value + n_sigmas*abs_unc < maxValue (low-pass)

The number of sigmas can be positive or negative.

Examples:

```
=mtFilter
"doseRates.3dmap"     ! existing mesh tally file
1                     ! high-pass filter:
0.150                 !    keep dose rates above 0.150
-3.0                  ! add -3.0 standard deviations to values before comparing
"above.3dmap"         ! new mesh tally file
end

=mtFilter
"above.3dmap"         ! existing mesh tally file
0                     ! flattening filter
"boolean.3dmap"       ! new mesh tally file
end
```

mtInv - Invert all of the values in a mesh tally.

Intended use: Invert non-zero values in a mesh tally to be used in further processing.

Input: The original mesh tally, the name of the resulting mesh tally file

Output: A new mesh tally file

Note: Uncertainties are propagated (the relative uncertainty of the reciprocal of a value is the same as the relative uncertainty of the value).

Example:

```
=mtInv
'someTally.3dmap'     ! existing mesh tally file
'inverted.3dmap'      ! new mesh tally file
end
```

mtMask - Keep only or remove specified voxels of a mesh tally based on geometry.

Intended use: Only keep or remove certain portions of a mesh tally based on the unit, media, or mixture at the center of the voxel.

Input: A mesh tally file, an action (keeponly or remove), an operation (intersection or union) of the unit=u, media=r and mixture=m, a replacement value for voxels not kept and the file name of the resulting mesh tally file. User can specify things such as 1) keep only the voxels that have unit=2 and mixture=5, 2) keep only the voxels that have media=3 or mixture=4, 3) remove voxels that have unit=2 and mixture=5, 4) remove voxels that have media=3 or mixture=4. To not include the unit, media, or mixture in the specification, use a value of -1.

Output: A new mesh tally file.

Note: When processing a file before finding the maximum, make the replacement value something very low. If `mtMask` is being used before finding the minimum, then set the replacement value high. Media is the SGGP media number within the unit.

```
=mtMask
"theTally.3dmap"       ! existing mesh tally file
keeponly               ! use 'keeponly' or 'remove'
intersection           ! use 'intersection' or 'union'
2 -1 5                 ! unit=2 AND mixture=5
0.0                    ! replacement value for voxels not kept
'new.3dmap'           ! new mesh tally file
end

=mtMask
"theTally.3dmap"       ! existing mesh tally file
remove                 ! use 'keeponly' or 'remove'
union                  ! use 'intersection' or 'union'
-1 3 4                 ! media=3 OR mixture=4
0.0                    ! replacement value for voxels removed
'new.3dmap'           ! new mesh tally file
end
```

mtMinMax - Find the location/value of the min or max of each real mesh in a mesh tally.

Intended use: Determine the minimum or maximum values in a mesh tally.

Input: The mesh tally, what to find (minimums or maximums), how many mins/maxs for each real mesh in the mesh tally, and the name of the text output file to store the results

Output: A text output containing the values and locations of the minimums or maximums of each real mesh in a tally file

---

Note: The same information is also in the main SCALE output file.

Example:

```
=mtMinMax
'bigOleMeshTally.3dmap'   ! existing mesh tally file
maximum                    ! find either minimums or maximums
5                          ! list top 5 maximum values in each real mesh
'theList.txt'             ! file name to store all of the results
end
```

mtMultiply - Multiply a mesh tally by a constant factor.

Intended use: Multiply every group of every family in a mesh tally for either a change in source strength or a change in units.

Input: The original mesh tally, the multiplier, and the name of the resulting mesh tally file

Output: A new mesh tally file

Example:

```
=mtMultiply
"simulation.mt1.3dmap"       ! the mesh tally
25.0                         ! source strength increase of 25
"simulation.bigger.3dmap"    ! output file name
end
```

mtPull - Pull values from certain voxels out of a mesh tally file.

Intended use: Get energy-dependent fluxes for certain locations from a mesh file.

Input: A mesh file (*.3dmap) file and a list of positions and/or voxels

Output: Listing of energy-dependent fluxes from each desired location to an ASCII text file

---

Note: Can pull fluxes either by a physical coordinate position or by voxel indices. Positions should be entered as a set of x, y, z for a Cartesian coordinate system and r, θ, z for a cylindrical coordinate system.

Example:

```
=mtPull
"duh.mt2.3dmap"    ! existing mesh file
n                  ! number of x,y,z points to pull
x_1 y_1 z_1        ! coordinates of point 1
x_1 y_2 z_2        ! coordinates of point 2
...
x_n y_n z_n        ! coordinates of point n
m                  ! number of i,j,k voxels to pull
i_1 j_1 k_1        ! indices of voxel 1
i_2 j_2 k_2        ! indices of voxel 2
...
i_m j_m k_m        ! indices of voxel m
"outputName.txt"   ! name of output text file
=end
```

mtRefine - Subdivide the mesh into smaller meshes.

Input: A *.3dmap mesh tally file with geometry mesh size (I,J,K) and three integers describing how many subdivisions of each voxel to create in each dimension

Output: A *.3dmap mesh tally file with geometry mesh size (I*nx,J*ny,K*nz)

Example:

```
=mtRefine
"fluxes.3dmap"        ! existing *.3dmap mesh tally file (I,J,K)
nx ny nz              ! how to subdivide each
"refined.3dmap"       ! new (largerer) *.3dmap mesh tally file (I*nx,J*ny,K*nz)
end
```

mtResp - Apply a response function to one family of a mesh tally file.

Intended use: Compute group-wise dose or reaction rates by combining a response function with the scalar fluxes.

Input: A *.3dmap mesh tally file and a group-wise response function

Output: A *.3dmap mesh tally file containing one family

Example:

```
=mtResp
"fluxes.3dmap"          ! existing *.3dmap mesh tally file
200                     ! number of bins in response
1                       ! which family
2.2675480E-04           ! response group 1
2.2283355E-04           ! response group 2
2.1878259E-04           ! response group 3
...
3.6748440E-06           ! response group nbins-2
3.6748443E-06           ! response group nbins-1
3.6748436E-06           ! response group nbins
"doseByGroup.3dmap"     ! new (smaller) *.3dmap mesh tally file
end
```

mtSplit - Split off part of a mesh tally file into a separate mesh tally file.

Intended use: Some mesh tallies may become so large that the MeshFileViewer cannot load the entire file to view. This utility allows users to split off one family or just one group of one family into a separate mesh tally file.

Input: The original mesh tally, which family (neutron, photon, or responses), and which dataset (usually a group). Instead of a dataset, users may specify 0 to get the total of a family or -1 to get all datasets for that family. The name of the resulting mesh tally also needs to be given.

Output: A new, smaller, mesh tally file

Example:

```
=mtSplit
"mavricUtilities3.mt1.3dmap"        ! the mesh tally
1                                   ! the family of neutron fluxes
5                                   ! fifth neutron flux group
"mavricUtilities3.nfluxg5.3dmap"    ! output file name
end
```

4.3.3 UTILITIES FOR WORKING WITH DENOVO BINARY FLUX (*.DFF) FILES

These utilities include the following:

dff2dso	Convert a Denovo flux file into a Denovo spatial output file.
dff2mai	Convert a Denovo flux file into a mesh angular information file.
dff2mim	Invert a Denovo flux file and store as a mesh importance map.
dff2msl	Convert a Denovo flux file into a mesh source lite.
dffBinOp	Binary operation of Denovo flux files: sum, difference, product, and ratio.
dffDisp	Display the basics of a Denovo flux file.
dffExpand	Expand a space-only Denovo flux file by an energy function.
dffFilter	Perform various filters on a Denovo flux file.
dffFix	Fix the zero and negative values in a Denovo flux file.
dffInt	Integrate a single particle type from a Denovo flux file.
dffInv	Invert the values in a Denovo flux file.
dffMult	Multiply a Denovo flux file by a constant factor.
dffPull	Pull fluxes from certain voxels out of a Denovo flux file.
dffResp	Apply a response function to scalar fluxes in a Denovo flux file.
dffSplit	Split off a single particle type from a Denovo flux file.

dff2dso - Convert a Denovo flux file into a Denovo spatial output file.

Input: A binary (stream) Denovo flux file and which particle types to convert

Output: A binary (stream) Denovo Spatial Output file

Note: For particle type, use 1 for neutron, 2 for photon, and 0 for all types.

Example:

```
=dff2dso
"neatStuff.dff"       ! existing Denovo flux file
1                     ! keep only neutron information
"neatStuff.dso"       ! new Denovo spatial output file
end
```

dff2mai - Convert a Denovo flux file into a mesh angular information file.

Intended use: Take the optional net current information from a Denovo flux file and create the adjoint current unit vectors and lambda parameters required for directional CADIS. This is stored in a mesh angular information (*.mai) file.

Input: A binary (stream) denovoFluxFile

Output: A binary (stream) meshAngularInfoFile, a mesh angular information file

Example:

```
=dff2mai
"mavricUtilities3.adjoint.dff"          ! new denovoFluxFile
"mavricUtilities3.mai"                  ! mesh angular info file
end
```

dff2mim - Invert a Denovo flux file and store as a mesh importance map.

Intended use: Make weight targets without a consistent biased mesh source.

Input: A Denovo flux (*.dff) file, a scalar constant, and the name of Monaco mesh importance map (*.mim) file.

Output: A Monaco mesh importance map (*.mim) file.

Example:

```
=dff2mim
"adjoint.dff"    ! existing adjoint denovoFluxFile
3.0e-10          ! constant  targetWeight = constant/adjFlux
"test.mim"       ! new Monaco mesh importance map
end
```

dff2msl - Convert a Denovo flux file into a mesh source lite.

Intended use: Take Denovo fission source information stored in a *.dff file and convert it to a mesh source lite file (*.msl) to be used as a KENO starting source, nst=9.

Input: A Denovo flux (*.dff) file

Output: A mesh source lite (*.msl) file

Example:

```
=dff2msl
"wishfulThinking.dff"    ! existing Denovo flux file
"startingSource.msl"     ! mesh source lite file
end
```

dffBinOp - Binary operation of Denovo flux files: sum, difference, product and ratio.

Intended use: Apply simple math to the results stored in Denovo flux files.

Input: The first flux file, the operator: add (or sum, +), subtract (or difference, -), multiply (or product, x, *), or divide (or ratio, ÷, "/"), the second flux file name, and the name of the resulting flux file

Output: A Denovo flux file

Note: Flux files need to have the same grid structure and number of groups. When using the / (slash) for division, enclose it in quotes ("/").

Example:

```
=dffBinOp
"neutron.dff"     ! first operand
divide            ! operation
"total.dff"       ! second operand
"ratio.dff"       ! output file name
end
```

dffDisp - Display the basics of a Denovo flux file.

Input: A Denovo flux (*.dff) file

Output: Some of the basic details of the Denovo flux file

Example:

```
=dffDisp
"fluxes.dff"          ! existing Denovo flux file
end
```

dffExpand - Expand a space-only Denovo flux file by an energy function.

Input: A Denovo flux (*.dff) file (with a single group - a space-only function), one or more particle types, and an energy function for each

Output: A full space/energy Denovo flux file

Example:

```
=dffExpand
"spatialFluxes.dff"       ! existing Denovo flux file (single group)
2                         ! number of particles
1                         ! particle type (1-neutron, 2-photon)
27                        ! number of bins in binned histogram distribution
2.00000E+07 3.0658021E-09 ! E_1  amount_1
6.37630E+06 6.9767163E-09 ! E_2  amount_2
3.01190E+06 1.1495182E-08 ! E_3  amount_3
   ...         ...
3.00000E-02 1.7127996E-04 ! E_26  amount_26
1.00000E-02 3.0910611E-04 ! E_27  amount_27
```

(continues on next page)

(continued from previous page)

```
1.00000E-05               ! E_28
2                         ! particle type (1-neutron, 2-photon)
19                        ! number of bins in binned histogram distribution
2.00E+07 0.0              ! E_1  amount_1
1.00E+07 0.0              ! E_2  amount_2
8.00E+06 0.0              ! E_3  amount_3
   ...       ...
1.00E+05 0.0              ! E_17  amount_17
4.50E+04 0.0              ! E_18  amount_18
1.00E+04                  ! E_19
"expanded.dff"            ! new Denovo flux file
end
```

dffFilter - Perform various filters on a Denovo flux file.

Intended use: Keep fluxes in a *.dff file where the flux or response meets a specified criterion.

Input: A Denovo flux file name, filter type, filter options, the output file name

Output: A Denovo flux file

Note: There are three basics types of filters: (0) flattening filter, (1) high-pass filter, (2) low-pass filter. For types 1 and 2, the criteria could be a computed response. The input list changes depending on the filter type and whether a response function is included. For no response function, use 0 for the number of groups. A flattening filter turns any positive value into a value of "1.0".

Examples:

```
=dffFilter
"some.dff"                ! input Denovo flux filename
0                         ! filter type
"flattened.dff"           ! output Denovo flux filename
end

=dffFilter
"some.dff"                ! input Denovo flux filename
2                         ! filter type
10.0                      ! maximum value
0                         ! number of groups for response function
"simpleFiltered.dff"      ! output Denovo flux filename
end

=dffFilter
"some.dff"                ! input Denovo flux filename
1                         ! filter type
10.0                      ! minimum value
19                        ! number of groups for response function
1.1620022E-05             !   should match total groups in file
8.7445696E-06
7.4596655E-06
6.3505804E-06
5.3994922E-06
4.6016462E-06
3.9522688E-06
3.4588520E-06
3.0130868E-06
2.6200121E-06
2.1944491E-06
1.8269592E-06
1.5149031E-06
```

(continues on next page)

(continued from previous page)

```
1.1595382E-06
8.7044964E-07
6.2187445E-07
3.7080767E-07
2.6877788E-07
5.9327226E-07
"respFiltered.dff"       ! output Denovo flux filename
end
```

dffFix - Fix the zero and negative values in a Denovo flux file.

Intended use: Replace zero or negative values with nearest good neighboring value. Checks previous group, previous x voxel, previous y voxel, then previous z voxel.

Input: A Denovo flux file

Output: A new Denovo flux file

Example:

```
=dffFix
"original.dff"    ! existing Denovo flux file
"repaired.dff"    ! new Denovo flux file
end
```

dffInt - Integrate a single particle type from a Denovo flux file.

Input: A Denovo flux file, which particle type to integrate (1-neutron, 2-photon), and the filename of the resulting integrated file

Output: A single-group Denovo flux file

Example:

```
=dffInt
"coupled.dff"       ! existing Denovo flux file
2                   ! particle type
"photonTotal.dff"   ! new Denovo flux file (single group)
end
```

dffInv - Invert the values in a Denovo flux file.

Input: A Denovo flux file

Output: A Denovo flux file

Note: Only non-zero values are inverted.

Example:

```
=dffInv
"fluxes.dff"        ! existing Denovo flux file
"inverted.dff"      ! new Denovo flux file
end
```

dffMult - Multiply a Denovo flux file by a constant factor.

Intended use: source strength change, change in units, etc.

Input: A Denovo flux file and a constant factor

Output: A Denovo flux file

Example:

```
=dffMult
"fluxes.dff"        ! existing Denovo flux file
10000.0             ! change units from (/cm^2/s) to (/m^2/s)
"multiplied.dff"    ! new Denovo flux file
end
```

dffPull - Pull fluxes from certain voxels out of a Denovo flux file.

Intended use: Get energy-dependent fluxes for certain locations from a flux file.

Input: A Denovo flux file and a list of positions and/or voxels

Output: Listing of energy-dependent fluxes from each desired location to an ASCII text file

---

Note: Can pull fluxes either by a physical coordinate position or by voxel indices.

Example:

```
=dffPull
"fluxes.dff"        ! file with the scalar fluxes you want
n                   ! number of x,y,z points to pull
x_1 y_1 z_1         ! coordinates of point 1
x_1 y_2 z_2         ! coordinates of point 2
...
x_n y_n z_n         ! coordinates of point n
m                   ! number of i,j,k voxels to pull
i_1 j_1 k_1         ! indices of voxel 1
i_2 j_2 k_2         ! indices of voxel 2
...
i_m j_m k_m         ! indices of voxel m
"outputName.txt"    ! name of output text file
=end
```

dffResp - Apply a response function to scalar fluxes in a Denovo flux file.

Intended use: Compute group-wise dose or reaction rates by combining a response function with the scalar fluxes. This can be done for every particle type in the flux file or a single specific particle type.

Input: A Denovo flux file, particle indicator and a group-wise response function

Output: A Denovo flux file

Note: 0-all particles, 1-neutron, 2-photon.

Example:

```
=dffResp
"fluxes.dff"          ! existing coupled Denovo flux file
1                     ! keep only neutron information
200                   ! number of bins in response
2.2675480E-04         ! response group 1
2.2283355E-04         ! response group 2
2.1878259E-04         ! response group 3
...
3.6748440E-06         ! response group nbins-2
3.6748443E-06         ! response group nbins-1
3.6748436E-06         ! response group nbins
"doses.dff"           ! new (smaller) Denovo flux fle
end
```

or

```
=dffResp
"fluxes.dff"          ! existing Denovo flux file
0                     ! keep all particles information
46                    ! number of bins in response
1.6151395E-04         ! response group 1, first neutron
1.4451494E-04         ! response group 2
1.2703618E-04         ! response group 3
...
3.6748447E-06         ! response group 27, last neutron
1.1620022E-05         ! response group 28, first photon
8.7445696E-06         ! response group 29
7.4596655E-06         ! response group 30
...
5.9327226E-07         ! response group 46, last photon
"doses.dff"           ! new Denovo flux file
end
```

dffSplit - Split off a single particle type from a Denovo flux file.

Intended use: Make a flux file containing a single particle type from another Denovo flux file.

Input: A Denovo flux file and a particle type

Output: A (smaller) Denovo flux file

Note: 1-neutron, 2-photon.

Example:

```
=dffSplit
"coupled.dff"     ! existing Denovo flux file
2                 ! particle type
"photons.dff"     ! new (smaller) Denovo flux file
end
```

4.3.4 UTILITIES FOR WORKING WITH DENOVO *.VARSCL (A TORT FORMAT) FILES

These utilities include the following:

vs2dff	Convert a varscl file into a Denovo flux file.
vsAdder	Add two TORT *.varscl files together into one *.varscl file.
vsBinOp	Binary operation of TORT *.varscl files: sum, difference, product and ratio.

continues on next page

Table 4.3.3 – continued from previous page

vsDisp	Display the basic contents of a TORT *.varscl file.
vsFilter	Perform various filters on a TORT *.varscl file.
vsInt	Integrate a single particle type from a TORT *.varscl file.
vsInv	Invert the values in a TORT *.varscl file.
vsMult	Multiply a TORT *.varscl file by a constant factor.
vsPull	Pull fluxes from certain voxels out of a TORT *.varscl file.
vsReGrp	Regroup a TORT *.varscl file.
vsResp	Apply a response function to scalar fluxes in a TORT *.varscl file.
vsSplit	Split off part of a TORT *.varscl file into a separate *.varscl file.

These utilities work with the *.varscl files produced with SCALE 6 and SCALE 6.1. The *.varscl format (a TORT format) is a single precision, binary format that has been replaced with the double precision, binary *.dff file (Denovo flux file) in SCALE 6.2. SCALE 6 and SCALE 6.1 users can request the executable binaries for these utilities by sending an email to scaleHelp@ornl.gov.

vs2dff - Convert a varscl file into a Denovo flux file.

Intended use: Convert a varscl file (used in previous versions of MAVRIC) into a Denovo flux file (introduced in SCALE 6.2).

Input: The *varscl file name, whether or not it is an adjoint flux, and the filename for the resulting denovoFluxfile

Output: A binary (stream) denovoFluxFile

Example:

```
=vs2dff
"mavricUtilities3.adjoint.varscl"        ! the TORT varscl file
true                                     ! it is an adjoint flux
"mavricUtilities3.adjoint.dff"           ! new denovoFluxFile
end
```

vsAdder - Add two TORT *.varscl files together into one *.varscl file.

Intended use: Beta versions of MAVRIC used TORT and GRTUNC-3D and could add the *.varscl files from each together before using them to create importance maps. MAVRIC now uses Denovo and no longer needs to add separate GRTUNC/TORT files. This utility is designed for people wishing to use the older files with the latest MAVRIC.

Input: Two *.varscl file names, typically one from GRTUNC-3D and the other from TORT, the filename of the added file, whether you want lots of output displayed ("true" or "false") and whether or not there is a minimum value of flux to use. If so, it is then listed.

Output: A single *.varscl with the specified name

Note: Addition is commutative, but not all varscl files are created equal. Do not mix up the GRTUNCL and the TORT files. GRTUNCL3D does not fill in the header info quite right, so the added varscl file takes header info only from the TORT varscl file.

Example:

```
=vsAdder
"/some/path/problem.gtunc.adjoint.varscl"
"/some/path/problem.tort.adjoint.varscl"
"total.varscl"
f
t
1.0e-25
end

=shell
  cp total.varscl ${RTNDIR}/total.varscl
end
```

vsBinOp - Binary operation of TORT *.varscl files: sum, difference, product and ratio.

Intended use: Apply simple math to the results stored in TORT *.varscl files.

Input: The first flux file, the operator: add (or sum, +), subtract (or difference, -), multiply (or product, x, *), and divide (or ratio, ÷, "/"), the second flux file name, and the name of the resulting flux file

Output: A TORT *.varscl file

Note: Flux files must have the same grid structure and number of groups. When using the / (slash) for division, enclose it in quotes ("/").

Example:

```
=vsBinOp
"neutron.varscl"     ! first operand
false                ! are these adjoint files?
divide               ! operation
"total.varscl"       ! second operand
"ratio.varscl"       ! output file name
end
```

vsDisp - Display the basic contents of a TORT *.varscl file.

Input: A TORT *.varscl file name and adjoint flag

Output: Text display

Examples:

```
=vsDisp
"some.varscl"            ! input TORT *.varscl filename
false                    ! is this an adjoint varscl?
end

=vsDisp
"some.varscl"            ! input TORT *.varscl filename
adjoint                  ! is this an adjoint varscl?
end
```

vsFilter - Perform various filters on a TORT *.varscl file.

Intended use: Keep fluxes in a *.varscl file where the flux or response meets a specified criterion.

Input: A TORT *.varscl file name, filter type, filter options, the output file name

Output: A TORT *.varscl file

Note: There are three basics types of filters: (0) flattening filter, (1) high-pass filter, (2) low-pass filter. For types 1 and 2, the criteria could be a computed response. The input list changes depending on the filter type and whether a response function is included. For no response function, use 0 for the number of groups.

Examples:

```
=vsFilter
"some.varscl"            ! input TORT *.varscl filename
false                    ! is this an adjoint varscl?
0                        ! filter type
"flattened.varscl"       ! output TORT *.varscl filename
end

=vsFilter
"some.varscl"            ! input TORT *.varscl filename
false                    ! is this an adjoint varscl?
2                        ! filter type
10.0                     ! maximum value
0                        ! number of groups for response function
"simpleFiltered.varscl"  ! output TORT *.varscl filename
end

=vsFilter
"some.varscl"            ! input TORT *.varscl filename
false                    ! is this an adjoint varscl?
1                        ! filter type
10.0                     ! minimum value
19                       ! number of groups for response function
1.1620022E-05            !   should match total groups in file
8.7445696E-06
7.4596655E-06
6.3505804E-06
5.3994922E-06
4.6016462E-06
3.9522688E-06
3.4588520E-06
3.0130868E-06
2.6200121E-06
2.1944491E-06
1.8269592E-06
1.5149031E-06
1.1595382E-06
8.7044964E-07
6.2187445E-07
3.7080767E-07
2.6877788E-07
5.9327226E-07
"respFiltered.varscl"    ! output TORT *.varscl filename
```

vsInt - Integrate a single particle type from a TORT *.varscl file.

Input: A TORT *.varscl file

Output: A single-group TORT *.varscl file

Example:

```
=vsInt
"coupled.varscl"       ! existing TORT *.varscl file
false                  ! is this an adjoint file?
2                      ! particle type (0-all, 1-neutron, 2-photon)
"photonTotal.varscl"   ! new TORT *.varscl file (single group)
end
```

vsInv - Invert the values in a TORT *.varscl file.

Input: A TORT *.varscl file

Output: A TORT *.varscl file

Note: Only non-zero values are inverted.

Example:

```
=vsInv
"fluxes.varscl"        ! existing TORT *.varscl file
false                  ! is this an adjoint file?
"inverted.varscl"      ! new TORT *.varscl file
end
```

vsMult - Multiply a TORT *.varscl file by a constant factor.

Intended use: source strength change, change in units, etc.

Input: A TORT *.varscl file and a constant factor

Output: A TORT *.varscl file

Example:

```
=vsMult
"fluxes.varscl"        ! existing TORT *.varscl file
false                  ! is this an adjoint file?
10000.0                ! change units from (/cm^2/s) to (/m^2/s)
"multiplied.varscl"    ! new TORT *.varscl file
end
```

vsPull - Pull fluxes from certain voxels out of a TORT *.varscl file.

Intended use: Get energy-dependent fluxes for certain locations from a flux file.

Input: A TORT *.varscl file and a list of positions and/or voxels

Output: Listing of energy-dependent fluxes from each desired location to an ASCII text file

---

Note: Can pull fluxes either by a physical coordinate position or by voxel indices.

Example:

```
=vsPull
"fluxes.varscl"    ! file with the scalar fluxes you want
false              ! is this an adjoint file?
n                  ! number of x,y,z points to pull
x_1 y_1 z_1        ! coordinates of point 1
x_1 y_2 z_2        ! coordinates of point 2
...
x_n y_n z_n        ! coordinates of point n
m                  ! number of i,j,k voxels to pull
i_1 j_1 k_1        ! indices of voxel 1
i_2 j_2 k_2        ! indices of voxel 2
...
i_m j_m k_m        ! indices of voxel m
"outputName.txt"   ! name of output text file
end
```

vsReGrp - Regroup a TORT *.varscl file.

Input: A TORT *.varscl file and adjoint flag, then a list of how the new groups should be formed from the old groups

Output: A smaller TORT *.varscl file

Example:

```
=vsReGrp
"coupled.varscl"       ! existing TORT *.varscl file
false                  ! is this an adjoint file?
27                     ! number of neutron groups in file
1                      ! new group assignment for each
1                      !     existing neutron group
1                      !   must start with one
2                      !   each entry is same as last or
...                    !     increases by 1
8                      ! new group assignment for neutron group 27
19                     ! number of photon groups in file
1                      ! new group assignment for each
1                      !     existing photon group
1                      !   must start with one
2                      !   each entry is same as last or
...                    !     increases by 1
4                      ! new group assignment for photon group 19
"smaller.varscl"       ! new TORT *.varscl file name
end
```

vsResp - Apply a response function to scalar fluxes in a TORT *.varscl file.

Intended use: Compute group-wise dose or reaction rates by combining a response function with the scalar fluxes.

Input: A TORT *.varscl file and a group-wise response function

Output: A TORT *.varscl file

Example:

```
=vsResp
"fluxes.varscl"       ! existing coupled TORT *.varscl file
false                 ! is this an adjoint file?
200                   ! number of bins in response
2.2675480E-04         ! response group 1
2.2283355E-04         ! response group 2
2.1878259E-04         ! response group 3
...
3.6748440E-06         ! response group 198
3.6748443E-06         ! response group 199
3.6748436E-06         ! response group 200
"doses.varscl"        ! new (smaller) Denovo flux fle
end
```

vsSplit - Split off part of a TORT *.varscl file into a separate *.varscl file.

Intended use: Make a flux file containing a single particle type from another TORT *.varscl file.

Input: A TORT *.varscl file and a particle type

Output: A (smaller) TORT *.varscl file

Example:

```
=vsSplit
"coupled.varscl"     ! existing TORT *.varscl file
false                ! is this an adjoint file?
2                    ! particle type (1-neutron, 2-photon)
"photons.varscl"     ! new (smaller) TORT *.varscl file
end
```

4.3.5 MISCELLANEOUS UTILITIES

These utilities include the following:

dsi2asc	Convert a Denovo simple input (*.dsi) from binary to ASCII.
dsiDisp	Display the basics of a Denovo simple input file.
dso2msl	Use a Denovo spatial output to create a mesh source lite.
dsoDisp	Display the basics of a Denovo spatial output file.
mim2wwinp	Convert a mesh importance map into an MCNP weight window file.
mimDisp	Display the basics of a mesh importance map (*.mim) file.
mimNorm	Normalize a mesh importance map to a given location/energy.
msmDisp	Display the basics of a mesh source map (*.msm) file.

dsi2asc - Convert a Denovo simple input (*.dsi) from binary to ASCII.

Intended use: Check a Denovo input file for correctness.

Input: Names of original binary Denovo simple input (*.dsi) file and the desired ASCII text file

Output: Human-readable form of the Denovo input file

Example:

```
=dsi2asc
"input.dsi"    ! Denovo simple input file (binary)
"ascii.txt"    ! new ascii text file
end
```

dsiDisp - Display the basics of a Denovo simple input file.

Input: A Denovo simple input (*.dsi) file

Output: Some of the basic details of the Denovo simple input file

Example:

```
=dsiDisp
"godiva.dsi"          ! existing Denovo simple input file
end
```

dso2msl - Use a Denovo spatial output to create a mesh source lite.

Input: A *.dso file is made of three-dimensional data sets called fields. Which field to convert? 1-n: convert that field 0: convert sum of all fields

Output: A mesh source lite (*.msl) file for KENO-VI.

Example:

```
=dso2msl
"fisSource.dso"  ! Denovo spatial output file with many fields
1                ! which field to use
"test.msl"       ! new Monaco mesh source lite
end
```

dsoDisp - Display the basics of a Denovo spatial output file.

Input: A Denovo spatial output (*.dsi) file

Output: Some of the basic details of the Denovo spatial output file

Example:

```
=dsoDisp
"godiva.dso"          ! existing Denovo spatial output file
end
```

mim2wwinp - Convert a mesh importance map into an MCNP weight window file.

Intended Use: To create an MCNP weight window file from a Monaco mesh importance map file outside of a MAVRIC calculation. Monaco mesh importance map files store target weights, but MCNP wwinp files store lower weight bounds. To convert, the user needs to supply the windowRatio, r (the ratio of the upper weight bound for splitting to the lower weight bound for roulette). Target weights, t, are the average of the upper, u, and lower, l, weight window bounds, so l=2t/(r+1). For example, for a Monaco target weight of 1.0 and a windowRatio of 10.0, the MCNP lower weight bound will be l=2(1.0)/(10.0+1)=0.1818. To reduce the size of the map, the user can specify which neutron and photon groups to store in the new file. If the last group is less than the first group, no groups of that particle will be stored.

Input:

Line 1: filename of the Monaco mesh importance map file
Line 2: windowRatio (>1.0)
Line 3: first_neutron_group last_neutron_group
Line 4: first_photon_group last_photon_group
Line 5: filename of the MCNP weight window input file

Output: The resulting weight window input file stored with the desired filename

---

Note: Geometry information in the Monaco mesh importance map file is lost since the MCNP wwinp format does not support it.

Example:

```
=mim2wwinp
"/scale/smplprbs/mavric.graphiteCADIS.mim"  ! importance map
19.0                                        ! window ratio
5 22                                        ! save n groups 5-22
19 1                                        ! save no p groups
"/scale/test9/testmimww.wwinp"              ! new file
end
```

mimDisp - Display the basics of a mesh importance map file.

Input: A mesh importance map (*.mim) file

Output: Some of the basic details of mesh importance map file

Example:

```
=mimDisp
"the.mim"    ! existing mesh importance map file
end
```

mimNorm - Normalize a mesh importance map to a given location/energy.

Input: A mesh importance map (*.mim) file, a location (x, y, z), a particle type and energy, and a filename for the normalized map file. Use 1 for neutron and 2 for photon. Energy should be in eV. The new importance map file will be normalized such that the given location/energy has a target weight of 1. If a particle type or energy is 0, then the energy group with the minimum non-zero target value at the given location will be the group that is set to 1.0 in the new file. (This option is similar to the MCNP weight window generator.)

Output: A mesh importance map file

Example:

```
=mimNorm
"the.mim"       ! existing mesh importance map file
27.5 -16.5 32.0 ! location
1 1.0e6         ! neutron, 1 MeV
"normed.mim"    ! new file that is normalized
end
```

msmDisp - Display the basics of a mesh source map file.

Input: A mesh source map (*.msm) file

Output: Some of the basic details of mesh source map file

Example:

```
=msmDisp
"the.msm"    ! existing mesh source map file
```

4.4 MAVRIC ADVANCED FEATURES

This appendix contains information on several advanced features that are still under development or are non-standard use of the MAVRIC sequence.

4.4.1 ALTERNATE NORMALIZATION OF THE IMPORTANCE MAP AND BIASED SOURCE

The importance map and biased source implemented in MAVRIC are only functions of space and energy. The importance for a specific location and energy represents the average over all directions. For applications involving a collimated beam source, a space/energy importance map may not be representative of the true importance of the particles as they stream away from the source.

As an example, consider a 14.1 MeV active interrogation beam source 1 meter from a small spherical boat containing illicit nuclear material. The objective is to compute the fission rate in the nuclear material. To create the biasing parameters, an adjoint source is located within the nuclear material and the resulting importance map is shown in Fig. 4.4.1. Note that in both the air and water, the importances change with

distance from the ship, but for the beam source, the importance (to causing a fission in the nuclear material) anywhere along the beam should be the same, since there is little chance a 14.1 MeV neutron will interact with the air before striking the ship.

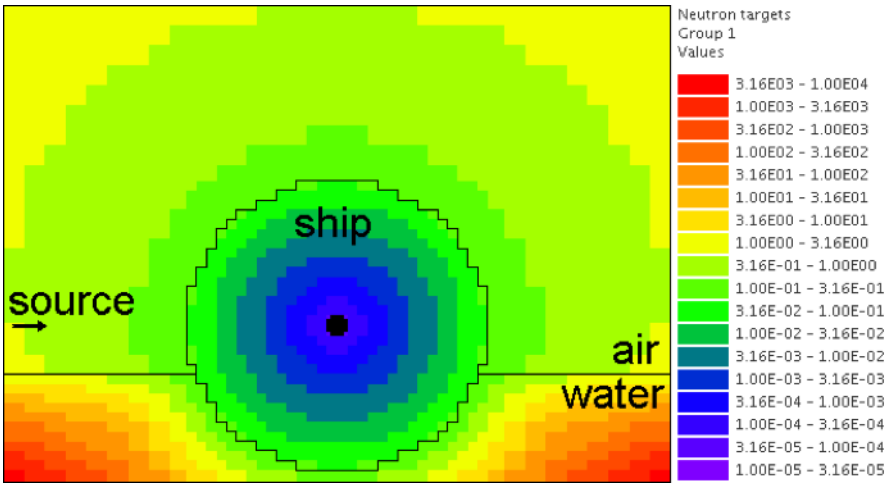


Fig. 4.4.1: Importance map computed using standard CADIS.

The CADIS algorithm has done exactly what it was supposed to: it made a space/energy importance map and normalized it such that the target weight where the 14.1 MeV source particles are born is 1. The problem with this is that the source particles will stream towards the ship and strike the hull where the target weight is 0.092. Since source particles have little chance of interacting in the air, the weight windows are not used to split the particle as they travel towards the ship. When source particles cross into the ship, they are split by a factor of 11 to match the target weight. For this example, splitting each particle by a factor of 11 once they strike the ship is not so bad, but for longer distances, this will result in much larger splits. For a polyenergetic source, this could lead to undersampling of the source and could result in higher variances.

To remedy this problem when using beam sources, the normalization of the importance map and biased source should not be done at the source location but instead at the point where the source particles first interact with the ship. The keyword “shiftNormPos ΔDelta`y ΔDelta`x, ΔDelta`z when the biased source and importance map are developed. For the Monaco Monte Carlo calculation, the source is returned to its normal position. The source input for the above problem would then be

```
read sources
    src 1
        title="14.1 DT neutrons - collimated"
        strength=1e30
'       (true source position)
        sphere 0 origin x=-195 y=0 z=0
'       (a mono-energetic 14.1 MeV distribution)
        eDistributionID=1
        direction 1.0 0.0 0.0
'       (a 2° beam )
        dDistributionID=2
'       (just inside the hull)
        shiftNormPos 107.7 0.0 0.0
    end src
end sources
```

where the shift moves the source position from x = -195 to x = -87.3, just inside the hull. The resulting target weights are shown in Fig. 4.4.2 The source particles are born with weight 1 in a location with a target weight

10.9. The particle weight is not checked until the particle crosses into the hull, where the target weight is 1.0.

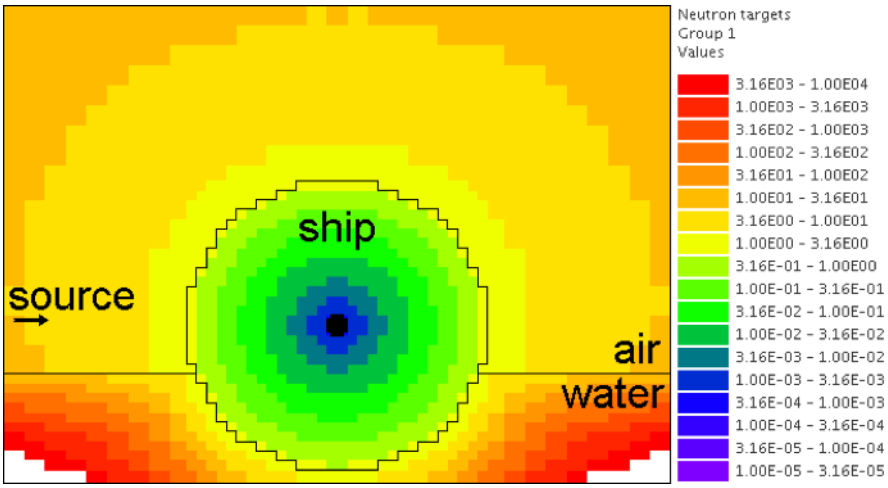


Fig. 4.4.2: Targets weights using the "shiftNormPos" keyword.

Other options to manipulate the importance map for special situations include the "mapMultiplier=*f*" keyword (in the importanceMap block or the biasing block), which will multiply every target weight by the factor *f*, and the keyword "noCheckAtBirth" in the parameters block will prevent the weight windows from being applied to source particles when they are started. When used in the MAVRIC sequence, the "shiftNormPos" capability automatically adds "noCheckAtBirth" to the Monaco input that is created.

4.4.2 IMPORTANCE MAPS WITH DIRECTIONAL INFORMATION

In MAVRIC, the CADIS method is implemented in space and energy, but in general, it could also include particle direction as well. This formulation would be the following:

True source:

$$q\left(\vec{r}, E, \widehat{\Omega}\right) \tag{4.4.1}$$

Desired response:

$$\sigma\left(\vec{r}, E, \widehat{\Omega}\right) \tag{4.4.2}$$

Adjoint flux using $q^{+}\left(\vec{r}, E, \widehat{\Omega}\right) = \sigma\left(\vec{r}, E, \widehat{\Omega}\right)$:

$$\psi^{+}\left(\vec{r}, E, \widehat{\Omega}\right) \tag{4.4.3}$$

Estimate of detector response

$$R = \iiint q(\vec{r}, E, \widehat{\Omega})\psi^{+}(\vec{r}, E, \widehat{\Omega}) d\Omega dE dV \tag{4.4.4}$$

Biased source:

$$\widehat{q}\left(\vec{r}, E, \widehat{\Omega}\right) = \frac{1}{R} q\left(\vec{r}, E, \widehat{\Omega}\right) \psi^{+}\left(\vec{r}, E, \widehat{\Omega}\right) \tag{4.4.5}$$

Target weight windows:

$$\overline{w}\left(\vec{r}, E, \widehat{\Omega}\right) = \frac{R}{\psi^{+}\left(\vec{r}, E, \widehat{\Omega}\right)} \tag{4.4.6}$$

For a system using a deterministic method to compute the adjoint fluxes, this completely general, space/energy/angle, approach presents many difficulties in implementation, namely,

a. dealing with the amount of memory required for a $\left(\vec{r}, E, \widehat{\Omega}\right)$ importance map in memory,

b. interpolating the importance for particle directions in between quadrature angles, and

c. expressing the biased source in a form suitable for a general MC code since the above biased source is, in general, not separable.

4.4.2.1 Approaches incorporating directional information

Completely general space/energy/angle CADIS is most likely too difficult to implement and may not be necessary for most applications. In most real problems that involve directionally dependent source distributions, the directional dependence is azimuthally symmetric about some reference direction, $\widehat{d}$. The angular distribution, $q_i\left(\widehat{\Omega}\right)$, can be expressed as the product of the uniform azimuthal distribution and a polar distribution about reference direction $\widehat{d_i}$ giving $\frac{1}{2\pi} q_i\left(\widehat{\Omega} \bullet \widehat{d_i}\right)$. The geometric size of these sources tends to be small, allowing each source distribution to be expressed as the product of two separable distributions: $q_i\left(\vec{r}, E, \widehat{\Omega}\right) \cong q_i\left(\vec{r}, E\right)\ q_i\left(\widehat{\Omega}\right)$.

What is needed is a CADIS method that (1) can account for the importance of a particle traveling in a certain direction; (2) can be cast as a simple modification of the space/energy CADIS method using $\overline{w}\left(\vec{r}, E\right)$ and $\widehat{q}\left(\vec{r}, E\right)$; and (3) is simpler than the full space/angle/energy approach. This can be done starting with the approximation that the angular component of the adjoint flux $\psi^{+}\left(\vec{r}, E, \widehat{\Omega}\right)$ is separable and symmetric about the average adjoint current direction $\widehat{n}\left(\vec{r}, E\right)$, such that

$$\psi^{+}\left(\vec{r}, E, \widehat{\Omega}\right) \cong \phi^{+}\left(\vec{r}, E\right)\ \frac{1}{2\pi} f\left(\widehat{\Omega} \bullet \widehat{n}\right)\ . \tag{4.4.7}$$

This is similar to the AVATAR approach [MAV-AppC-VRUS97] but with explicitly including the azimuthal distribution so that the standard definition $\int \phi^{+}\left(\vec{r}, E\right)\ \frac{1}{2\pi} f\left(\widehat{\Omega} \bullet \widehat{n}\right)\ d\widehat{\Omega} = \phi^{+}\left(\vec{r}, E\right)$ applies. The probability distribution function $f(\mu)$ describing the shape of the azimuthally symmetric current at $\left(\vec{r}, E\right)$ has the form of

$$f(\mu) = \frac{\lambda e^{\lambda\mu}}{2\sinh(\lambda)}, \tag{4.4.8}$$

with the single parameter $\lambda\left(\vec{r}, E\right)$ determined from $\overline{\mu}\left(\vec{r}, E\right)$, the average cosine of scatter.

From this, we can propose that weight window targets be developed that are inversely proportional to the approximation of the adjoint angular flux:

$$\overline{w}\left(\vec{r}, E, \widehat{\Omega}\right) = \frac{2\pi\ k}{\phi^{+}\left(\vec{r}, E\right)\ \ f\left(\widehat{\Omega} \bullet \widehat{n}\right)}\ , \tag{4.4.9}$$

where k is the constant of proportionality that will be adjusted to make the importance map consistent with the biased source(s). Two methods will be examined here, one without and one with biasing of the source directional dependence.

For both of the methods, the S_N code Denovo was modified to report not only the adjoint scalar fluxes, $\phi^+\left(\vec{r},E\right)$, but also the adjoint net currents in x, y, and z directions: $J_x\left(\vec{r},E\right)$, $J_y\left(\vec{r},E\right)$, and $J_z\left(\vec{r},E\right)$. These currents are used to find $\widehat{n}\left(\vec{r},E\right)$ and $\lambda\left(\vec{r},E\right)$. The following methods have been developed so that the standard CADIS routines can be used to compute space/energy quantities of the response per unit source R, the weight window target values $\overline{w}\left(\vec{r},E\right)$, and biased source $\widehat{q}\left(\vec{r},E\right)$ with just the adjoint scalar fluxes. These quantities are then modified by the directional information.

4.4.2.2 Directionally dependent weight windows without directional source biasing

It is proposed that the biased source $\widehat{q}\left(\vec{r},E,\widehat{\Omega}\right)$ should be proportional to both the true source distribution and the space/energy component of the adjoint flux:

$$\widehat{q}\left(\vec{r},E,\widehat{\Omega}\right)=\frac{1}{R}\left[q\left(\vec{r},E\right)\ \frac{1}{2\pi}q\left(\widehat{\Omega}\bullet\widehat{d}\right)\right]\ \phi^+\left(\vec{r},E\right)\ , \tag{4.4.10}$$

where the constant of proportionality, R, is determined by forcing $\widehat{q}\left(\vec{r},E,\widehat{\Omega}\right)$ to be a pdf. Since the angular component of the adjoint flux is not included, the directional distribution of the biased source will be exactly the same as the true source. Note that this approach would be exact for cases where no directional biasing could be applied — beam sources.

For multiple sources (each with a probability distribution function $q_i\left(\vec{r},E\right)$ and a strength S_i, giving a total source strength of $S=\sum S_i$), the user is required to provide one point in phase space $\left(\vec{r}_i,E_i,\widehat{\Omega}_i\right)$ for each source i that is representative of that entire source where the biased source will match the target weight windows. For each source, a vector $\widehat{n}_i=\widehat{n}\left(\vec{r}_i,E_i\right)$ is computed using that point. For the general case of multiple sources, the biased source sampling distribution, the biased source distributions, and the weight windows are computed using

$R_i=\iint q_i\left(\vec{r},E\right)\ \phi^+\left(\vec{r},E\right)dE\ dr$ (estimated response from source i)

$\widehat{p}\left(i\right)=\frac{S_iR_i\ f\left(\widehat{\Omega}_i\bullet\widehat{n}_i\right)}{\sum S_iR_i\ f\left(\widehat{\Omega}_i\bullet\widehat{n}_i\right)}$ (biased sampling of source i)

$$\widehat{q}_i\left(\vec{r},E,\widehat{\Omega}\right)\ =\ \tfrac{1}{R_i}q_i\left(\vec{r},E\right)\ \phi^+\left(\vec{r},E\right)\ \tfrac{1}{2\pi}q_i\left(\widehat{\Omega}\bullet\widehat{d}_i\right)\ =\ \widehat{q}_i\left(\vec{r},E\right)\ \tfrac{1}{2\pi}q_i\left(\widehat{\Omega}\bullet\widehat{d}_i\right)$$

$$\overline{w}\left(\vec{r},E,\widehat{\Omega}\right)\ \ =\ \ \frac{\sum S_iR_i\ f\left(\widehat{\Omega}_i\bullet\widehat{n}_i\right)}{S\phi^+\left(\vec{r},E\right)}\frac{1}{f\left(\widehat{\Omega}\bullet\widehat{n}\right)}\ \ =\ \ \frac{\sum S_iR_i\ f\left(\widehat{\Omega}_i\bullet\widehat{n}_i\right)}{\sum S_iR_i}\overline{w}\left(\vec{r},E\right)\frac{1}{f\left(\widehat{\Omega}\bullet\widehat{n}\right)}$$

4.4.2.3 Directionally dependent weight windows with directional source biasing

Here it is proposed that the biased source be proportional to both the true source distribution and the approximation of the adjoint angular flux. With a small geometric source, it is also assumed that there is one vector, $\widehat{n}_0=\widehat{n}\left(\vec{r}_0,E_0\right)$, evaluated at a specific location and energy, which represents the adjoint current direction over that source. The biased source then looks like

$$\begin{aligned}\widehat{q}\left(\vec{r},E,\widehat{\Omega}\right)&=\frac{1}{\mathrm{Rc}}q\left(\vec{r},E,\widehat{\Omega}\right)\ \psi^+\left(\vec{r},E,\widehat{\Omega}\right)\\&=\frac{1}{\mathrm{Rc}}\left[q\left(\vec{r},E\right)\ \frac{1}{2\pi}q\left(\widehat{\Omega}\bullet\widehat{d}_i\right)\right]\left[\phi^+\left(\vec{r},E\right)\ \frac{1}{2\pi}\ f\left(\widehat{\Omega}\bullet\widehat{n}_0\right)\right]\ ,\end{aligned} \tag{4.4.11}$$

where the constant Rc is used to make $\widehat{q}$ a pdf. Note that if either the original source directional distribution $q\left(\widehat{\Omega}\right)$ or the adjoint angular flux distribution at the source is isotropic, then $c=\frac{1}{4\pi}$.

For the general case of multiple sources, the biased source sampling distribution, the biased source distributions and the weight windows are

$$R_i = \iint q_i\left(\vec{r}, E\right)\ \phi^+\left(\vec{r}, E\right)\mathrm{dE\ dr} \tag{4.4.12}$$

$$c_i = \int \frac{1}{2\pi} q_i\left(\widehat{\Omega} \bullet \widehat{d_i}\right)\ \frac{1}{2\pi} f\left(\widehat{\Omega} \bullet \widehat{n_i}\right) d\widehat{\Omega} \tag{4.4.13}$$

$$\widehat{p}\,(i) = \frac{S_i R_i c_i}{\sum S_i R_i c_i} \tag{4.4.14}$$

$$\begin{aligned} \widehat{q_i}\left(\vec{r}, E, \widehat{\Omega}\right) &= \left[\frac{1}{R_i}\ q_i\left(\vec{r}, E\right)\ \phi^+\left(\vec{r}, E\right)\right]\left[\frac{1}{c_i}\ q_i\left(\widehat{\Omega}\right)\ f\left(\widehat{\Omega}\right)\right] \\ &= \widehat{q_i}\left(\vec{r}, E\right)\ \frac{1}{c_i}\ \frac{1}{2\pi} q_i\left(\widehat{\Omega} \bullet \widehat{d_i}\right)\ \frac{1}{2\pi} f\left(\widehat{\Omega} \bullet \widehat{n_i}\right) \end{aligned} \tag{4.4.15}$$

$$\begin{aligned} \overline{w}\left(\vec{r}, E, \widehat{\Omega}\right) &= \frac{\sum S_i R_i c_i}{S\,\phi^+\left(\vec{r}, E\right)}\ \frac{2\pi}{f\left(\widehat{\Omega} \bullet \widehat{n}\right)} \\ &= \frac{\sum S_i R_i c_i}{\sum S_i R_i}\ \overline{w}\left(\vec{r}, E\right)\ \frac{2\pi}{f\left(\widehat{\Omega} \bullet \widehat{n}\right)}\,. \end{aligned} \tag{4.4.16}$$

More details on the development of these methods and their application for several problems have been presented [MAV-AppC-PME12, MAV-AppC-PMEW10].

4.4.2.4 Using space/energy/angle CADIS in MAVRIC

The two angular CADIS methods that use the AVATAR-type approximation of adjoint flux are specified in MAVRIC with the "angularBiasing=" keyword in the importanceMap block. Values for this keyword are 1 or 2.

Space/Energy/Angle CADIS without directional biasing (for beam sources) — This method uses one specific location, $\vec{r}_0$, energy, E_0, and direction, $\widehat{\Omega}_0$, which is the reference direction of the source $\widehat{d}$, where the weight of the biased source matches the weight window.

Space/Energy/Angle CADIS with directional biasing (for general sources) — This method uses one specific energy, E_0, to determine the adjoint current vector $\widehat{n}_0$ and the λ_0 parameter for the biased angular distribution for each source.

With each method, the user must specify at what energy the importance map and the biased sources should be made consistent. The particle type must also be specified. This is done with the keywords "angBiasParType=" (1 for neutron or 2 for photon) and "angBiasEnergy=" (with a value in eV), also in the importanceMap block.

Note that all sources should have a direction $\widehat{d}$ set, using "direction *u v w*" within each source, even if the angular distribution for a given source is isotropic. The direction is used for source biasing and for aligning the weight windows and biased sources. Also note that for either angular biasing method, the Denovo S_N calculation must use a Legendre order greater than 0.

With angular biasing, a mesh angular information (*.mai) file is produced which can be visualized with the MeshFileViewer. This file contains the space/energy-dependent $\lambda\left(\vec{r}, E\right)$ values and components of the average adjoint current direction $\widehat{n}\left(\vec{r}, E\right) = \left\langle n_x, n_y, n_z\right\rangle$. An existing mesh angular information (*.mai) file can be used in a separate MAVRIC problem by using the "meshAngInfoFile=" keyword in the biasing block.

4.4.2.5 Example problem

Consider the Ueki shielding problem used as sample problems in the Monaco and MAVRIC manuals. The goal is to calculate the neutron dose on one side of a shield from a partially collimated ^{252}Cf source on the other side of the shield. Both of the angular approaches discussed above can be compared to analog and standard space/energy CADIS calculations. For the analog calculations, no importanceMap block is used. For the other cases, the importance map blocks are shown below.

CADIS

```
read importanceMap
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
    gridGeometryID=7
    macromaterial
      mmTolerance=0.01
    end macromaterial



end importanceMap
```

Angular CADIS 1 - without a biased source angular dist.

```
read importanceMap
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
    gridGeometryID=7
    macromaterial
      mmTolerance=0.01
    end macromaterial

    angularbiasing=1
    angBiasParType=1
    angBiasEnergy=2.0e6
end importanceMap
```

Angular CADIS 2 - with a biased source angular dist.

```
read importanceMap
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
    gridGeometryID=7
    macromaterial
      mmTolerance=0.01
    end macromaterial

    angularbiasing=2
    angBiasParType=1
    angBiasEnergy=2.0e6
end importanceMap
```

Note that the energy at which to tie the importance map to the biased source, 2 MeV, is about the average energy of the source energy distribution. The figure-of-merit (FOM) of the calculation could change as this parameter is varied.

Results, shown in Table 4.4.1, demonstrate that the two directional approaches improved the FOM for this problem by more than a factor of 2. A larger boost is seen in method 2 where biasing is applied to the source directional distribution. In this case, the biased source distribution was an exponential distribution with a power of 2.45, significantly sampling more source neutrons in the direction of the shield than into the paraffin collimator block.

Note that the space/energy/angle CADIS methods require more memory to hold the importance information. Improvements to the CADIS calculation can also be made with a judicious choice of standard source direction biasing, without an increase in memory requirements. A simple distribution (where $\mu = 0.924$ represents the edge of the cone cutout and $\mu = 0.974$ is the center half of the shield dimension) added that to the source

```
distribution 2
    abscissa -1.0     0.924    .974   1.0 end
    truepdf      0.962     0.025   .013   end
    biasedpdf    0.500     0.250  0.250   end
end distribution

src 1
    neutrons  strength=4.05E+07
    cuboid  0.01 0.01  0 0  0 0
    eDistributionID=1
    direction 1.0 0.0 0.0
    dDistributionID=2
end src
```

results in an FOM improvement of nearly 3 over the standard CADIS without the overhead of the angular CADIS methods.

Table 4.4.1: Results of the Ueki Shielding Problem (35 cm graphite shield)

Calculation Method	Adj. S_N (min)	MC (min)	dose rate (rem/hr)	relative uncert	MC FOM (/min)
Analog	0.0	152.7	3.998E-03	0.0101	64.7
CADIS	0.2	9.9	3.998E-03	0.0081	1550
Directional CADIS 1, no src bias	0.2	10.3	4.035E-03	0.0054	3390
Directional CADIS 2, with src bias	0.2	10.0	4.012E-03	0.0049	4190
CADIS, standard src dir. bias	0.2	10.0	3.998E-03	0.0047	4550

A series of problems was used to compare the space/energy/angle CADIS to the standard space/energy CADIS. Most of the problems saw an improvement of a factor of about 2 or 3. Some problems did not improve at all, and some photon problems actually performed worse. In that case (a photon litho-density gauge), it could be that the angular approximation for importance, an exponential function in μ, cannot adequately describe the true importance.

Since the space/energy/angle CADIS methods are more difficult to explain (more theory, adjustable parameters set by the user), require more memory than standard CADIS, and may not offer any improvement over standard space/energy CADIS, they have not been made part of the main MAVRIC manual and have been left as an advanced/developing feature. These methods were not removed from the MAVRIC code since they may be helpful to future problems.

4.4.3 UNIVERSITY OF MICHIGAN METHODS FOR GLOBAL VARIANCE REDUCTION

The use of hybrid deterministic/Monte Carlo methods, particularly for global variance reduction, has been an active area of research by the transport team at the University of Michigan for a long time. One of the first approaches studied was a way to develop Monte Carlo weight window target values that were proportional to deterministically estimated values of the forward flux [MAV-AppC-CL01]. For global problems using isotropic weight windows, this reference argues that in order to get uniform relative uncertainties in the Monte Carlo calculation, the weight windows should be set such that the number density of Monte Carlo particles, $m\left(\overrightarrow{r}\right)$, is constant. The physical particle density, $n\left(\overrightarrow{r}\right)$, is related by the average weight, $\overline{w}\left(\overrightarrow{r}\right)$, to the Monte Carlo particle density by

$$n\left(\overrightarrow{r}\right) = \overline{w}\left(\overrightarrow{r}\right)\, m\left(\overrightarrow{r}\right) \ . \tag{4.4.17}$$

To make $m\left(\overrightarrow{r}\right)$ constant over the geometry, the weight window targets $\overline{w}\left(\overrightarrow{r}\right)$ need to be proportional to the physical particle density. Cooper and Larsen calculate the weight window targets from an estimate of the forward scalar flux $\phi\left(\overrightarrow{r}\right)$ to be

$$\overline{w}\left(\overrightarrow{r}\right) = \frac{\phi\left(\overrightarrow{r}\right)}{\max\left(\phi\left(\overrightarrow{r}\right)\right)} \ . \tag{4.4.18}$$

Two approaches have since been developed [MAV-AppC-BL09], based on what global information the user desires from the simulation: global flux weight windows, for obtaining every energy group at every location, and global response weight windows, for obtaining an energy-integrated response at every spatial location. Both of these methods are designed for calculating the "global solution" — everywhere in the geometry of the problem — with nearly uniform statistics.

Note that none of the University of Michigan methods discussed here included the development of biased sources. These methods have all been extended to include a consistent biased source by ORNL during their implementation in the MAVRIC sequence of SCALE. The methods have also been extended by ORNL for multiple sources.

4.4.3.1 Weight windows using only forward estimates of flux

Global flux weight windows

This method keeps the Monte Carlo particle distribution uniform in space and energy. Note that this is a space/energy version of the original space-only Cooper's Method. The target weight windows, $\overline{w}\left(\overrightarrow{r}, E\right)$, should be proportional to the estimate of the forward scalar flux, $\phi\left(\overrightarrow{r}, E\right)$.

$$\overline{w}\left(\overrightarrow{r}, E\right) = c\, \phi\left(\overrightarrow{r}, E\right) \ . \tag{4.4.19}$$

A biased source distribution, $\widehat{q}\left(\overrightarrow{r}, E\right)$, that is consistent with the target weight windows can be found from the true source distribution, $q\left(\overrightarrow{r}, E\right)$, and the forward flux to be

$$\widehat{q}\left(\overrightarrow{r}, E\right) = \frac{1}{c}\, \frac{q\left(\overrightarrow{r}, E\right)}{\phi\left(\overrightarrow{r}, E\right)} \ , \tag{4.4.20}$$

where the constant c can be determined so that the biased source distribution is a probability distribution function,

$$c = \int_V \int_E \frac{q\left(\overrightarrow{r}, E\right)}{\phi\left(\overrightarrow{r}, E\right)} \, \mathrm{dE}\, d\overrightarrow{r} \ . \tag{4.4.21}$$

Thus, using the estimate of forward flux, we can first compute c and then form a consistent set of weight window target values and a biased source distribution.

Global response weight windows

For an energy-integrated response (such as dose) desired at all spatial locations, this method keeps the Monte Carlo particle distribution proportional to the integral of the product of the response function, $\sigma_d\left(\overrightarrow{r},E\right)$, and the estimate of the forward flux, $\phi\left(\overrightarrow{r},E\right)$. The energy-integrated response, $D\left(\overrightarrow{r}\right)$, is found from the estimate of the forward flux as

$$D\left(\overrightarrow{r}\right)=\int_E \sigma_d\left(\overrightarrow{r},E\right)\ \phi\left(\overrightarrow{r},E\right)DE. \tag{4.4.22}$$

The target weight windows, $\overline{w}\left(\overrightarrow{r},E\right)$, should be

$$\overline{w}\left(\overrightarrow{r},E\right)=c\ \frac{D\left(\overrightarrow{r}\right)}{\sigma_d\left(\overrightarrow{r},E\right)}\ . \tag{4.4.23}$$

A biased source distribution, $\widehat{q}\left(\overrightarrow{r},E\right)$, that is consistent with the target weight windows can be found from the true source distribution, $q\left(\overrightarrow{r},E\right)$, and the forward dose estimate to be

$$\widehat{q}\left(\overrightarrow{r},E\right)=\ \frac{1}{c}\ \frac{q\left(\overrightarrow{r},E\right)\ \sigma_d\left(\overrightarrow{r},E\right)}{D\left(\overrightarrow{r}\right)} \tag{4.4.24}$$

using a proportionality constant of

$$c=\int_V \frac{1}{D\left(\overrightarrow{r}\right)}\int_E q\left(\overrightarrow{r},E\right)\ \sigma_d\left(\overrightarrow{r},E\right)\ \mathrm{dE}\ d\overrightarrow{r}\ . \tag{4.4.25}$$

Implementation in MAVRIC

The global flux weight windows (GFWW) approach and the global response weight windows (GRWW) are both triggered by specifying an importance map block without any adjoint sources. For the GRWW approach, the response of interest is listed in the importance map block. If none is listed, GFWW is used.

For problems with multiple sources (each with probability distribution function $q_i\left(\overrightarrow{r},E\right)$ and strength S_i, giving a total source strength of $S=\sum S_i$), the GWW methods require a biased mesh source for each. This was implemented in MAVRIC in a manner similar to the multiple source CADIS routines. Each biased source, $\widehat{q_i}\left(\overrightarrow{r},E\right)$, is developed as

$$\widehat{q_i}\left(\vec{r},E\right)=\begin{cases}\frac{1}{c_i}\frac{q_i\left(\vec{r},E\right)}{\phi\left(\vec{r},E\right)} & \text{global flux weight windows}\\ \frac{1}{c_i}\frac{q_i\left(\vec{r},E\right)\sigma_d\left(\vec{r},E\right)}{D\left(\vec{r}\right)} & \text{global response weight windows}\end{cases} \tag{4.4.26}$$

where c_i is a normalization constant. The weight windows are then set to

$$\overline{w}\left(\overrightarrow{r},E\right)=\begin{cases}\frac{\sum c_i}{\sum S_i}\ \phi\left(\overrightarrow{r},E\right) & \text{global flux weight windows}\\ \frac{\sum c_i}{\sum S_i}\ \frac{D\left(\overrightarrow{r}\right)}{\sigma_d\left(\overrightarrow{r},E\right)} & \text{global response weight windows}\end{cases}\ . \tag{4.4.27}$$

In the final Monte Carlo, the specific source i is sampled with probability $p(i) = S_i/S$, and then the particle is sampled from the biased mesh source $\widehat{q_i}\left(\overrightarrow{r}, E\right)$. Unlike the CADIS method for multiple sources, there is no way to develop a biased probability distribution for which source to sample without knowing the contribution to the global estimate from each source separately.

For multiple source problems where the expected contribution from each source is very different from the true strengths of those sources, it may be more efficient to run a series of problems with one source each for different amounts of time. The resulting mesh tallies can then be added together using the mesh tally adder (part of the MAVRIC utilities).

4.4.3.2 Methods using forward and adjoint estimates

Becker [MAV-AppC-Bec09] proposed three methods for developing weight windows based on both forward and adjoint deterministic solutions. These three methods correspond to the portion of the phase space over which uniform relative uncertainties are desired: a small "detector" region, a region comprising a significant portion of the entire problem, and the global problem. In this discussion, only a brief outline of each method, focusing on its implementation into MAVRIC, will be given.

Source/detector problems

For a small detector of volume $V_D$where we want to optimize the MC calculation of the detector response

$$R = \int_{V_D} \int_0^\infty \sigma\left(\overrightarrow{r}, E\right)\ \phi\left(\overrightarrow{r}, E\right)\text{dE dV} \tag{4.4.28}$$

or optimize for the energy dependent flux at the detector, the following is used:

forward flux estimate

$$\phi\left(\overrightarrow{r}, E\right) \tag{4.4.29}$$

adjoint source for flux

$$q^+\left(\overrightarrow{r}, E\right) = \frac{1}{\phi\left(\overrightarrow{r}, E\right)} \tag{4.4.30}$$

or for response

$$q^+\left(\overrightarrow{r}, E\right) = \sigma\left(\overrightarrow{r}, E\right) \tag{4.4.31}$$

adjoint flux

$$\phi^+\left(\overrightarrow{r}, E\right) \tag{4.4.32}$$

contributon flux

$$\phi^c\left(\overrightarrow{r}, E\right) = \phi\left(\overrightarrow{r}, E\right)\ \phi^+\left(\overrightarrow{r}, E\right) \tag{4.4.33}$$

normalization constant

$$C_{\text{norm}} = \frac{V_D}{\int_{V_D} \int_0^\infty \phi^c\left(\overrightarrow{r}, E\right) dE\ dV} \tag{4.4.34}$$

space-only contributon flux

$$\widetilde{\phi}^c\left(\overrightarrow{r}\right) = C_{\text{norm}} \int_0^\infty \phi^c\left(\overrightarrow{r}, E\right) dE \tag{4.4.35}$$

spatial parameter

$$\alpha\left(\overrightarrow{r}\right) = \left[1 + exp\left(\frac{\widetilde{\phi}^c_{\max \in V_D}}{\widetilde{\phi}^c\left(\overrightarrow{r}\right)} - \frac{\widetilde{\phi}^c\left(\overrightarrow{r}\right)}{\widetilde{\phi}^c_{\max \in V_D}}\right)\right]^{-1} \tag{4.4.36}$$

spatial parameter

$$B\left(\overrightarrow{r}\right) = \alpha\left(\overrightarrow{r}\right)\widetilde{\phi}^c\left(\overrightarrow{r}\right) + 1 - \alpha\left(\overrightarrow{r}\right) \tag{4.4.37}$$

weight windows

$$\overline{w}\left(\overrightarrow{r}, E\right) = \frac{B\left(\overrightarrow{r}\right)}{\phi^+\left(\overrightarrow{r}, E\right)} \tag{4.4.38}$$

Source-region problems

For a detector of volume V_D and surface area A_D (smaller than the entire problem) where we want to optimize the MC calculation of the detector response

$$R\left(\overrightarrow{r}\right) = \int_0^\infty \sigma\left(\overrightarrow{r}, E\right)\ \phi\left(\overrightarrow{r}, E\right) dE \qquad \overrightarrow{r} \in V_D \tag{4.4.39}$$

or optimize for the energy dependent flux in the region, the following is used:

forward flux estimate

$$\phi\left(\overrightarrow{r}, E\right) \tag{4.4.40}$$

adjoint source for flux

$$q^+\left(\overrightarrow{r}, E\right) = \frac{1}{\phi\left(\overrightarrow{r}, E\right)} \tag{4.4.41}$$

adjoint source for response

$$q^+\left(\overrightarrow{r}, E\right) = \frac{\sigma\left(\overrightarrow{r}, E\right)}{\int_0^\infty \sigma\left(\overrightarrow{r}, E\right)\ \phi\left(\overrightarrow{r}, E\right) dE} \tag{4.4.42}$$

adjoint flux estimate

$$\phi^+\left(\overrightarrow{r}, E\right) \tag{4.4.43}$$

contributon flux

$$\phi^c\left(\overrightarrow{r},E\right)=\phi\left(\overrightarrow{r},E\right)\ \phi^+\left(\overrightarrow{r},E\right) \tag{4.4.44}$$

normalization constant

$$C_{\text{norm}}=\frac{A_D}{\int_{A_D}\int_0^\infty \phi^c\left(\overrightarrow{r},E\right)dE\ dA} \tag{4.4.45}$$

space-only contributon flux

$$\widetilde{\phi}^c\left(\overrightarrow{r}\right)=\ C_{\text{norm}}\int_0^\infty \phi^c\left(\overrightarrow{r},E\right)dE \tag{4.4.46}$$

spatial parameter

$$\alpha\left(\overrightarrow{r}\right)=\left[1+exp\left(\frac{\widetilde{\phi}^c_{\max\in V_D}}{\widetilde{\phi}^c\left(\overrightarrow{r}\right)}-\frac{\widetilde{\phi}^c\left(\overrightarrow{r}\right)}{\widetilde{\phi}^c_{\max\in V_D}}\right)\right]^{-1} \tag{4.4.47}$$

spatial parameter

$$B\left(\vec{r}\right)=\begin{cases}\tilde{\phi}^c\left(\vec{r}\right) & \vec{r}\in V_D\\ \alpha\left(\vec{r}\right)\tilde{\phi}^c\left(\vec{r}\right)+1-\alpha\left(\vec{r}\right) & \text{otherwise}\end{cases} \tag{4.4.48}$$

weight windows

$$\overline{w}\left(\overrightarrow{r},E\right)=\frac{B\left(\overrightarrow{r}\right)}{\phi^+\left(\overrightarrow{r},E\right)} \tag{4.4.49}$$

Note that A_D does not include surfaces of V_D which are on the boundary of the problem.

Global response problem

For optimizing the Monte Carlo calculation of a detector response everywhere in phase space

$$R(\overrightarrow{r})=\int_0^\infty \sigma\left(\overrightarrow{r},E\right)\ \phi\left(\overrightarrow{r},E\right)\text{dE} \tag{4.4.50}$$

or optimizing for the energy-dependent flux everywhere, the following is used:

forward flux estimate

$$\phi\left(\overrightarrow{r},E\right) \tag{4.4.51}$$

adjoint source for flux

$$q^+\left(\overrightarrow{r},E\right)=\frac{1}{\phi\left(\overrightarrow{r},E\right)} \tag{4.4.52}$$

adjoint source for response

$$q^{+}\left(\overrightarrow{r},E\right)=\frac{\sigma\left(\overrightarrow{r},E\right)}{\int_{0}^{\infty}\sigma\left(\overrightarrow{r},E\right)\ \phi\left(\overrightarrow{r},E\right)dE} \tag{4.4.53}$$

adjoint flux estimate

$$\phi^{+}\left(\overrightarrow{r},E\right) \tag{4.4.54}$$

contributon flux

$$\phi^{c}\left(\overrightarrow{r},E\right)=\phi\left(\overrightarrow{r},E\right)\ \phi^{+}\left(\overrightarrow{r},E\right) \tag{4.4.55}$$

space-only contributon flux

$$\phi^{c}\left(\overrightarrow{r}\right)=\int_{0}^{\infty}\phi^{c}\left(\overrightarrow{r},E\right)dE \tag{4.4.56}$$

spatial parameter

$$B\left(\overrightarrow{r}\right)=\phi^{c}\left(\overrightarrow{r}\right) \tag{4.4.57}$$

weight windows

$$\overline{w}\left(\overrightarrow{r},E\right)=\frac{B\left(\overrightarrow{r}\right)}{\phi^{+}\left(\overrightarrow{r},E\right)} \tag{4.4.58}$$

4.4.3.3 Implementation in MAVRIC

Like CADIS and FW-CADIS, the Denovo S_N code is used to calculate the forward flux estimate, $\phi\left(\overrightarrow{r},E\right)$, and the estimate of the adjoint flux, $\phi^{+}\left(\overrightarrow{r},E\right)$, for all of the Michigan weight window methods.

None of the above discussions of the University of Michigan methods include information on how the weight window target values were adjusted to match the source sampling. When implemented into MAVRIC, each of the above problem types will compute a biased source, $\widehat{q}\left(\overrightarrow{r},E\right)$, along with the target weight, $\overline{w}\left(\overrightarrow{r},E\right)$, that are produced. For a problem with a single source of strength S and distribution $q\left(\overrightarrow{r},E\right)$, the biased source distribution $\widehat{q}\left(\overrightarrow{r},E\right)$ is found by using

$$\widehat{q}\left(\overrightarrow{r},E\right)=\frac{q\left(\overrightarrow{r},E\right)}{\overline{w}\left(\overrightarrow{r},E\right)}\ . \tag{4.4.59}$$

The weight windows are multiplied by a factor of R/S, where R is defined as

$$R=\iint\widehat{q}\left(\overrightarrow{r},E\right)\phi^{+}\left(\overrightarrow{r},E\right)\ dE\ dV. \tag{4.4.60}$$

Sampled source particles will then be born with a weight that matches the weight window of the phase space where they are born.

For multiple sources, each with strength S_i and distribution $q_i\left(\overrightarrow{r}, E\right)$, each biased source distribution $\widehat{q_i}\left(\overrightarrow{r}, E\right)$ is found by using

$$\widehat{q_i}\left(\overrightarrow{r}, E\right) = \frac{q_i\left(\overrightarrow{r}, E\right)}{\overline{w}\left(\overrightarrow{r}, E\right)} \tag{4.4.61}$$

and the response from each source being

$$R_i = \iint \widehat{q_i}\left(\overrightarrow{r}, E\right) dE\, dV\ . \tag{4.4.62}$$

The individual sources are sampled with a biased probability of $\widehat{p}(i) = R_i / \sum R_i$. The weight windows are then multiplied by a factor of

$$\frac{\sum_i R_i}{\sum_i S_i} \tag{4.4.63}$$

to match the source birth weights.

To use one of the Becker methods in MAVRIC, the keyword "beckerMethod=" is used with values of 1, 2, or 3 for the source/detector, source/region, or global method. Adjoint sources are described just like standard MAVRIC CADIS and FW-CADIS problems. To switch between optimizing flux in every group or optimizing a response, the keywords "fluxWeighting" and "respWeighting" are used. Just like FW-CADIS, the response listed in each adjoint source is the response that is optimized. Note that even when starting a calculation with known forward and adjoint flux files, the adjoint source(s) still need to be listed since they are used in the final normalization of the weight windows.

Example problems

The first Becker method (source/detector) is demonstrated using the Ueki shielding problem (used as sample problems in the Monaco and MAVRIC manuals and above in the space/energy/angle CADIS example). The goal is to calculate the neutron dose on one side of a shield from a partially collimated ^{252}Cf source on the other side of the shield. For the analog calculations, no importanceMap block is used. For the other cases, the importance map blocks are shown below.

CADIS

```
read importanceMap
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
    gridGeometryID=7
    macromaterial
      mmTolerance=0.01
    end macromaterial

end importanceMap
```

Becker 1 - flux optimization

```
read importanceMap
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
    gridGeometryID=7
    macromaterial
      mmTolerance=0.01
    end macromaterial

    beckerMethod=1
    fluxWeighting
end importanceMap
```

Becker 2 - response optimization

```
read importanceMap
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
    gridGeometryID=7
    macromaterial
      mmTolerance=0.01
    end macromaterial

    beckerMethod=1
    respWeighting
end importanceMap
```

The results are shown in Table 4.4.2. The Becker response-optimized source/detector method, which requires two deterministic estimates, does slightly better for this problem than standard CADIS, which only requires one deterministic estimate.

Table 4.4.2: Results of the Ueki Shielding Problem (35 cm graphite shield)

Calculation method	For. S_N	Adj. S_N	MC	dose rate (rem/hr)	relative uncert	MC FOM (/min)
Analog	0.0	0.0	152.7	3.998E-03	0.0101	6.47
CADIS	0.0	0.2	9.9	3.988E-03	0.0081	1550
Becker, source/detector flux	0.2	0.2	9.9	4.027E-03	0.099	1040
Becker, source/detector resp	0.2	0.2	10.1	4.025E-03	0.0075	1760

A global problem

For an example of a global problem, consider a two-room block building with a criticality accident in one room. The objective is to find the photon dose everywhere in order to see the locations where criticality alarms would trigger. The building is 12 meters long, 6 meters wide, and 3 meters tall. A comparison will be made between an analog calculation, an FW-CADIS calculation (using response weighting), GRWW, Becker's source/region method (response optimization), and Becker's global method (response optimization). The MAVRIC importance map block of each calculation is as follows.

FW-CADIS

```
read importanceMap
    gridGeometryID=1
    adjointSource 1
        boundingBox
            1200 0
            600 0
            300 -60.0
        responseID=6
    end adjointSource

    respWeighting
end importanceMap
```

GRWW

```
read importanceMap
    gridGeometryID=1
    responseID=6
end importanceMap
```

Becker source/region

```
read importanceMap
    gridGeometryID=1
    adjointSource 1
        boundingBox
            1190 10
            590 10
            290 -560.0
        responseID=6
    end adjointSource
    beckerMethod=2
    respWeighting
end importanceMap
```

Becker - global

```
read importanceMap
    gridGeometryID=1
    adjointSource 1
        boundingBox
            1200 0
            600 0
            300 -60.0
        responseID=6
    end adjointSource
    beckerMethod=3
    respWeighting
end importanceMap
```

Note that Becker's source/region method is designed for regions smaller than the entire problem, so this is not a fair comparison, just a demonstration on how to use it in MAVRIC. The bounding box of the adjoint source in this case is set slightly smaller than the extent of the entire problem.

Results for the photon dose and its relative uncertainty using the five different methods of calculation are shown in Fig. 4.4.3. Information on the distribution of relative uncertainties is shown in Fig. 4.4.4 and listed in Table 4.4.3.

A more detailed comparison of the different hybrid methods for representative shielding problems can be found in [MAV-AppC-Pep13].

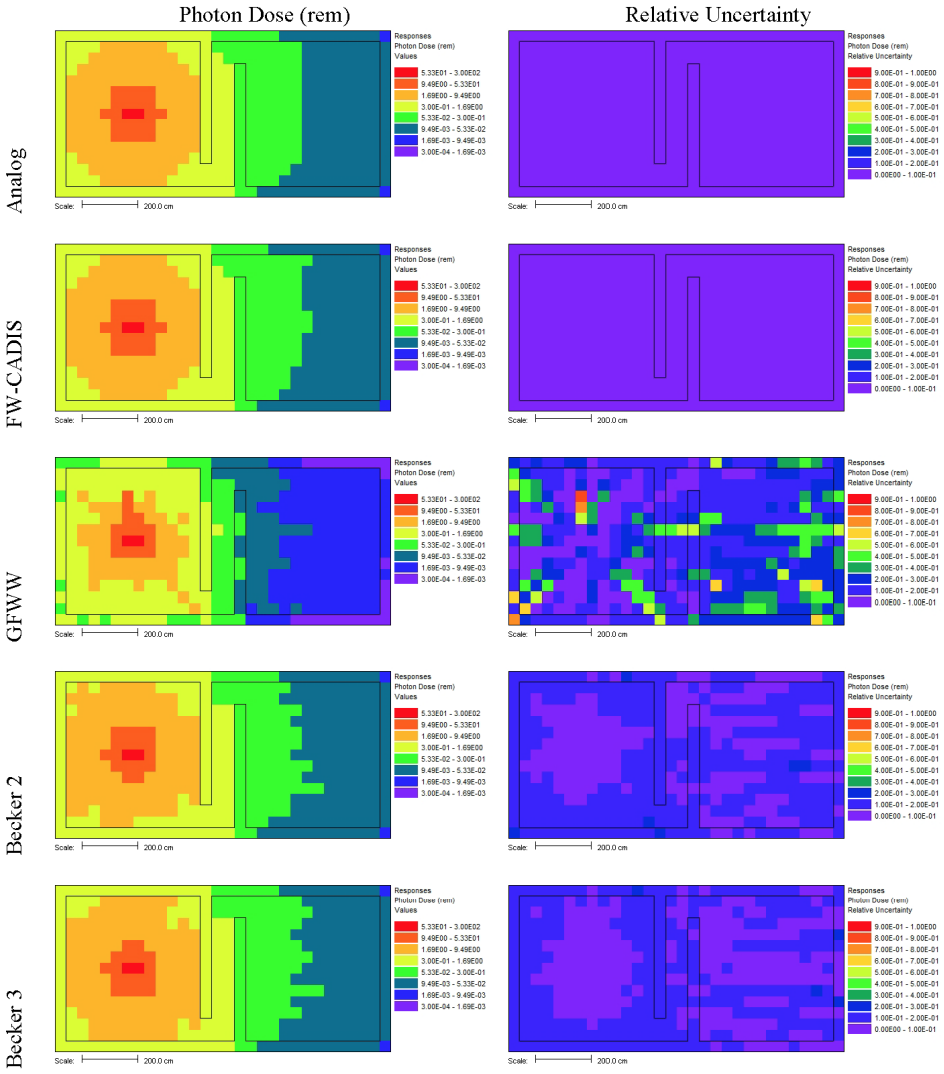


Fig. 4.4.3: Mesh tally results for the photon dose over the entire building use five different methods.

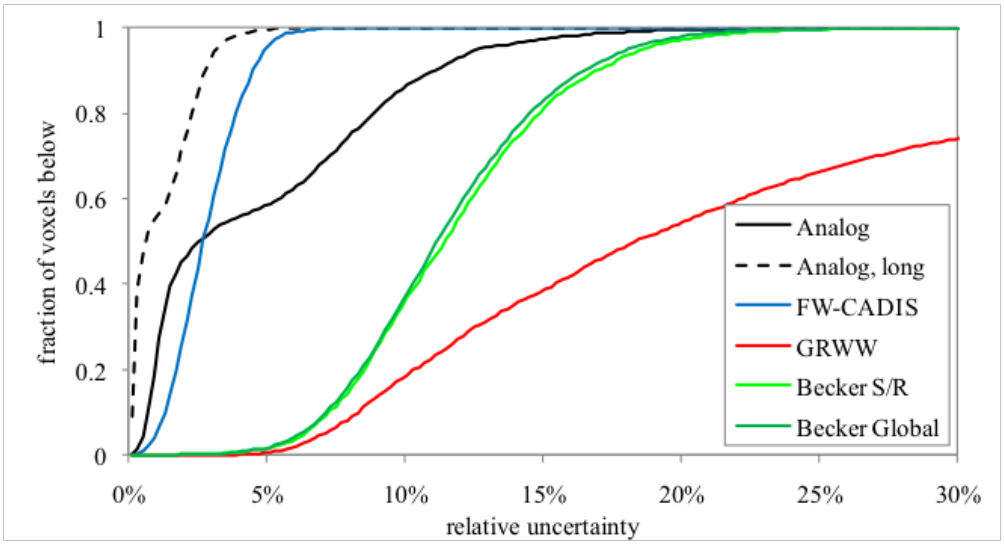


Fig. 4.4.4: The fraction of mesh tally voxels with less than a given amount of relative uncertainty for the five different methods.

Table 4.4.3: Comparison of the five different methods

Calculation method	Forward S_N (min)	Adjoint S_N (min)	Monte Carlo (min)	fraction of voxels with relative uncertainties less than 5%	fraction of voxels with relative uncertainties less than 10%	fraction of voxels with relative uncertainties less than 15%
Analog			20.1	0.583	0.857	0.973
Analog, long			308.1	0.996	1.000	1.000
FW-CADIS, response weighting	3.5	3.4	20.3	0.946	1.000	1.000
GRWW	4.0		20.2	0.005	0.180	0.387
Becker, source/region	4.0	3.6	19.0	0.016	0.350	0.800
Becker, global response	4.2	4.4	20.4	0.016	0.358	0.832

4.4.3.4 Other special options for the importance map block input

In MAVRIC, the above methods have been extended to use biased sources. To turn off the use of a biased source, use the keyword "selfNormalize". For forward flux-based methods, this will scale the target weights in the importance map so that the largest value is 1, which typically is at the source location. To use the forward-based Michigan methods (Cooper, GFWW, GRWW) or the van Wijk [MAV-AppC-VWVdEH11] methods as they were presented (without the MAVRIC biased source), use this keyword.

Use the keyword "spaceOnly=n" to create an importance map that contains one energy group over the entire range of energy. Use a value of 1 for neutron problems or a value of 2 for photon problems. This option is not designed for coupled problems. Both of the methods that van Wijk presents use importance maps that are space only.

Use the keyword "forwardParam=p" to allow the target weights to be proportional to the forward estimate (using Denovo) of flux raised to any power p.

$$\overline{w}\left(\overrightarrow{r}\right) = \frac{\phi^p\left(\overrightarrow{r}\right)}{\max\left(\phi^p\left(\overrightarrow{r}\right)\right)} \; . \tag{4.4.64}$$

For Cooper's method or van Wijk's first method, $p = 1$, which is the default if "forwardParam" is not specified. For van Wijk's second method, based on the relative error estimate $\mathrm{Re}\left(\overrightarrow{r}\right) \propto \frac{1}{\sqrt{\phi\left(\overrightarrow{r}\right)}}$, set this parameter to $p = 0.5$. Note that with a power less than 1, the weight window targets will span a smaller range than the expected Monte Carlo flux values. Particles may be rouletted before they reach the lowest flux areas of the problem.

When applying FW-CADIS for a semi-global problem, the adjoint source should cover the area where the lowest uncertainties are desired. Sometimes the area where this is desired is not a fixed geometric area but instead an area that has a final answer of a certain range. For example, when finding dose rate maps, the user may only be interested in getting low uncertainties for areas below the dose limit — areas above the dose limit may not matter as much since people will not be allowed in those areas. Another example may be that the dose map should be optimized in those areas above the background dose rate, since dose rates below that are of no concern. To allow the adjoint source area to be tailored in this way, the keywords "minForwardValue=" and/or "maxForwardValue="can be used. Default values are 0 and 10^{100}, respectively. For flux weighting, only the voxels within the adjoint source bounding box and containing a forward flux estimate between the min and max are included as adjoint sources.

A comparison [MAV-AppC-Pep13] of FW-CADIS to the University of Michigan methods and the Van Wijk methods showed that FW-CADIS performs better and is more straightforward to use than the other methods. These other methods were added to SCALE so that a fair comparison could be made and have been left in the code base for academic use in case some types of problems could benefit from them.

4.4.4 USING MAVRIC TO RUN FIXED-SOURCE DENOVO CALCULATIONS

The MAVRIC sequence of SCALE can be used to run the new discrete-ordinates code Denovo [MAV-AppC-ESSC10] The sequence is designed to use Denovo to compute an importance map for the Monte Carlo code Monaco but can be stopped without starting the final Monaco calculation. The full version of Denovo has a python-based user interface where the user must construct the geometry, source, and cross sections on their own. MAVRIC provides an easy interface to run the serial-only version of Denovo within SCALE (called 'xkba'). Denovo inputs made using MAVRIC can also be imported into the python interface for the HPC version of Denovo.

To create Denovo inputs or run Denovo using MAVRIC, input the composition, geometry (including arrays if needed), definitions (grid geometry and response information), and sources block as if an actual full MAVRIC case were being run. The sequence should then be started with a `parm=xxx` flag to indicate to MAVRIC to stop after making Denovo inputs or running Denovo.

parm=	MAVRIC stops after
Forward Calculations	
forinp	Making cross sections and Denovo input
forward	Running Denovo

Adjoint Calculations	
adjinp	Making cross sections and Denovo input
adjoint	Running Denovo

For forward Denovo simulations, the importance map block should contain which grid geometry to use and any of the optional Denovo parameters:

For forward Denovo simulations, the importance map block should contain which grid geometry to use and any of the optional Denovo parameters:

```
read importanceMap
    gridGeometryID=7
end importanceMap
```

For adjoint Denovo calculations, the importance map block should contain which grid geometry to use, one or more adjoint sources, and any of the optional Denovo parameters:

```
read importanceMap
    gridGeometryID=7
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
end importanceMap
```

For either forward or adjoint, the following files will be produced and should be copied out of the SCALE temporary directory back to your home area:

ft02f001	the AMPX mixed working library cross section file
xkba_b.inp	binary input file for Denovo
*.dff	Denovo flux file (binary) - scalar fluxes

The scalar flux file is automatically copied back to the home area when SCALE completes. The others can be copied back if desired using a shell command at the end of the MAVRIC input file:

```
=shell
    cp ft02f001   ${OUTDIR}/mixed.ft02f001
    cp xkba_b.inp ${OUTDIR}/denovo.dsi
end
```

The xkba_b.inp file and the *.dff file can both be viewed with the Java utility, Mesh File Viewer, that is shipped with SCALE. The xkba_b.inp file shows the Denovo source distribution (space and energy), and the *.dff shows the final computed scalar fluxes. The binary Denovo input file should be renamed with an extension of *.dsi (Denovo simple input) so that the viewer can properly interpret the data.

4.4.4.1 Optional Denovo parameters

The default values for various calculational parameters and settings used by Denovo for the MAVRIC sequence should cover most applications. These settings can be changed by the interested user in the importance map block. The two most basic parameters are the quadrature set used for the discrete ordinates and the order of the Legendre polynomials used in describing the angular scattering. The default quadrature order that MAVRIC uses is S_8, and for the order of Legendre polynomials, the default is P_3 (or the maximum number of coefficients contained in the cross section library, if less than 3). S_8/P_3 should be a good choice for many applications, but the user is free to changes these. For complex ducts or transport over large distance at small angles, S_{12} may be required. S_4/P_1 would be useful in doing a very cursory run just to check if the problem was input correctly. Other parameters that can be set by the user in the importance map block for Denovo calculations are listed in Table 4.4.4.

Table 4.4.4: Optional Denovo parameters for the MAVRIC Importance Map Block

block keyword	type	length	default	required	restrictions/comments
importanceMap					
Optional Denovo S_N code parameters					
discretization=	integer		4	no	0-diamond-difference, 1-diamond-difference with flux fix-up:lin0, 2-theta-weighted diamond-difference, 3-linear-discontinuous finite element, 4-step characteristics, 5-trilinear discontinuous finite elem
quadType=	integer		0	no	0-level symmetric, 1-Gauss-Legendre product, 2-QR
legendre=	integer		3*	no	P_L, L=highest Legendre polynomial, L=0,1,2,3,… *default is to use min(the highest available in the data,3)
tportcorrection=	integer		1*	no	transport correction: 0-none, 1-diagonal, 2-Cesaro *For P_2 or higher, the default is 2 (Cesaro)
upScatter=	integer		0	no	upscatter iterations: 0-none, 1-yes, 2-ignore
xblocks=	integer		1	no	parallel stuff
yblocks=	integer		1	no	parallel stuff
zblocks=	integer		1	no	parallel stuff
numSets=	integer		1	no	?
partUpscatter=	integer		1	no	partition upscatter (0-no, 1-yes)
quadrature=	integer		8	no	level symmetric S_N quadrature, N=2, 4, 6, 8, 10, 12, 14, 16
polarsPerOct=	integer		0	no	Gauss-Legendre product quadrature or QR
azimuthsPerOct=	integer		0	no	Gauss-Legendre product quadrature or QR
maxIters=	integer		20	no	maximum number of iterations
diagnostics=	integer		0	no	0-no diganostics, 1-all diagnostics
output=	integer		0	no	0-no ouput, 1-all output
krylovSpaceSize=	integer		10	no	size in memory for Krylov space
tolerance=	double		1.E-03	no	tolerance used in convergence test
krylovType=	integer		0	no	0-GMRES, 1-BiCGStab
eigenSolver=	integer		0	no	0-power iteration, 1-Arnoldi, 2-shifted inverse
multiGSolover=	integer		0	no	0-Gauss-Seidel, 1-Krylov
withinGSolver=	integer		0	no	0-Krylov, 1-residual Krylov, 2-source iteration
mgSettings=	integer		1	no	0-user supplied, 1-automatic
upGroupSolver=	integer		0	no	0-same as within-group solver, 1-Krylov, 2-residual Krylov, 3-source iteration, 4-single source iteration
acceleration=	integer		0	no	0-none, 1-two grid
maxItersMG=	integer		20	no	maximum number of iterations
toleranceMG=	double		0.001	no	tolerance used in convergence test
end importanceMap					

In problems with small sources or media that are not highly scattering, discrete ordinates can suffer from "ray effects"-where artifacts of the quadrature directions can be seen in the computed fluxes. To help alleviate the ray effects problem, Denovo has a first-collision capability. This computes the amount of uncollided flux in each mesh cell from a point source. These uncollided fluxes are then used as a distributed source in the

main discrete-ordinates solution. At the end of the main calculation, the uncollided fluxes are added to the fluxes computed with the first-collision source, forming the total flux. While this helps reduce ray effects in many problems, the first-collision capability can take a long time to compute on a mesh with many cells or for many point sources.

The macromaterials option ("mmTolerance=") can be used to better represent the geometry in Denovo. Refer to the main MAVRIC manual for details on macromaterials.

4.4.4.2 Forward source preparations

The user-entered source description is converted to a mesh-based source for Denovo. To create the mesh source, MAVRIC determines if the defined source exists within each cell. This is done by dividing each mesh cell into $n \times n \times n$ subcells (from the keyword "subCells=n" in the importance map block with a default of n=2) and testing each subcell center. For every subcell center that is a valid source position (within the spatial solid and meets optional unit, region, or mixture requirements), an amount of source proportional to the subcell volume is assigned to the mesh cell. The keyword "subCells=" can be used to better refine how much source is computed for the mesh cells at the boundary of a curved source region. Of course, more subcell testing takes more time.

The above process may miss small sources or degenerate sources (surfaces, lines, points) that do not lie on the tested subcell centers. If none of the mesh cells contain any source after the subcell method, then random sampling of the source is used. A number of source positions are sampled from the source (set by the "sourceTrials=" keyword, default of 1000000) and then placed into the proper mesh cell. If this method is used, the resulting Denovo input file should be visualized to ensure that the statistical nature of the source trials method does not unduly influence the overall mesh source.

For forward calculations, if the number of mesh cells containing the true source is less than 10, then MAVRIC will convert these source voxels to point sources, to allow Denovo to use its first-collision capability to help reduce ray effects. The user can easily override the MAVRIC defaults-to force the calculation of a first-collision source no matter how many voxels contain source-by using the keyword "firstCollision." To prevent the calculation of a first-collision source, the keyword "noFirstCollision" can be used.

For coupled problems, there are two ways to make Denovo compute photon fluxes from a neutron source: 1) include a tally using a photon response or 2) manually specify the "startGroup=" and the "endGroup=" to cover the particles/energy groups that are desired in the final Denovo output.

4.4.4.3 Adjoint source preparation

For adjoint calculations, adjoint sources that use point locations will use the Denovo first-collision capability. Volumetric adjoint sources (that use a boundingBox) will be treated without the first-collision capability. The keywords "firstCollision" and "noFirstCollision" will be ignored by MAVRIC for adjoint calculations.

Note that for an adjoint Denovo calculation, the MAVRIC input must still list a forward source. Otherwise, the sequence will report an error and stop. The forward source is not used for an adjoint Denovo calculation, but it must be present to make a legal MAVRIC input. For a coupled problem using an adjoint photon source, using a neutron forward source will make Denovo compute both photon and neutron adjoint fluxes. If the forward source(s) and adjoint source(s) are all photon, then only photon adjoint fluxes will be computed. The keywords "startGroup=" and "endGroup=" can also be used to manually set the particles/energy groups that are desired in the final Denovo output.

4.4.4.4 Other notes

Denovo (in SCALE 6) is a fixed-source S_N solver and cannot model multiplying media. Neither forward nor adjoint neutron calculations from Denovo will be accurate when neutron multiplication is a major source component. If neutron multiplication is not turned off in the parameters block of the MAVRIC input (using keyword "noFissions"), a warning will be generated to remind the user of this limitation.

By default, MAVRIC instructs Denovo not to perform outer iterations for neutron problems if the cross-section library contains upscatter groups. This is because the time required to calculate the fluxes using upscatter can be significantly longer than without. For problems where thermal neutrons are an important part of the transport or tallies, the user should specify the keyword "upScatter=1" in the importance map block. This will instruct Denovo to perform the outer iterations for the upscatter groups, giving more accurate results but taking a longer time.

4.4.4.5 MAVRIC utilities for Denovo

Denovo simply calculates scalar fluxes for every mesh cell and energy group — it does not compute responses based on those fluxes. Several utilities have been added to the collection of MAVRIC utilities in order to allow the user to further process Denovo scalar flux results. The full details and input descriptions for these utilities can be found in the MAVRIC Utilities description in Sect. 4.3.

dff2dso	Convert a Denovo flux file into a Denovo spatial output file.
dff2mai	Convert a Denovo flux file into a mesh angular information file.
dff2mim	Invert a Denovo flux file and store as a mesh importance map.
dff2msl	Convert a Denovo flux file into a mesh source lite.
dffBinOp	Binary operation of Denovo flux files: sum, difference, product, and ratio.
dffDisp	Display the basics of a Denovo flux file.
dffFilter	Perform various filters on a Denovo flux file.
dffFix	Fix the zero and negative values in a Denovo flux file.
dffInt	Integrate a single particle type from a Denovo flux file.
dffInv	Invert the values in a Denovo flux file.
dffMult	Multiply a Denovo flux file by a constant factor.
dffPull	Pull fluxes from certain voxels out of a Denovo flux file.
dffResp	Apply a response function to scalar fluxes in a Denovo flux file.
dffSplit	Split off a single particle type from a Denovo flux file.

There are also two utility programs that look at the Denovo simple input file (binary) that MAVRIC creates. In the SCALE temporary directory, the file is called xkba_b.inp. In order to display in the MeshFileViewer, Denovo simple input files need to be renamed with a *.dsi extension. The utilities are as follows.

dsi2asc	Convert a Denovo simple input (*.dsi) from binary to ASCII (so the user can see every detail).
dsiDisp	Display the basics of a Denovo simple input file.

4.4.4.6 Example problem

Forward

As an example, consider a simulation based on the Ueki shielding experiments (see Monaco manual). A ^{252}Cf neutron source was placed in the center of a 50 cm cube of paraffin which had a 45° cone cut-out. The goal is to calculate the neutron dose at a detector 110 cm from the source.

The input file (`denovo1.inp`) needs the compositions of the paraffin and graphite

```
read composition
    para(h2o)    3         1.0 293.0 end
    carbon       4 den=1.7 1.0 293.0 end
end composition
```

then the geometry

```
read geometry
    global unit 1
        cuboid  1   25.0 -25.0  25.0 -25.0   25.0 -25.0
        cone    2  10.35948  25.01   0.0 0.0   rotate  a1=-90 a2=-90 a3=0
        cuboid  3   90.0  70.0  40.0 -40.0   40.0 -40.0
        cuboid 99  120.0 -30.0  50.0 -50.0   50.0 -50.0
        media 3 1  1 -2
        media 0 1  2
        media 4 1  3
        media 0 1  99 -1  -2 -3
    boundary 99
end geometry
```

then the position of the detector, the response of the detector and the mesh to use for Denovo

```
read definitions
    location 1
        position 110 0 0
    end location
    response 5
        title="ANSI standard (1977) neutron flux-to-dose-rate factors"
        specialDose=9029
    end response
    distribution 1
        title="Cf-252 neutrons, Watt spectrum a=1.025 MeV and b=2.926/MeV"
        special="wattSpectrum"
        parameters 1.025 2.926 end
    end distribution
    gridGeometry 7
        xLinear 4 -25 -5
        xLinear 15 -5 25
        xLinear  9 25 70
        xLinear  8 70 90
        xplanes -30 95 100 105 109 111 115 120  end

        yplanes -50 -40 40 50 end
        yLinear  5 -40 -15
        yLinear 15 -15 15
        yLinear  5  15 40

        zplanes -50 -40 40 50 end
        zLinear  5 -40 -15
        zLinear 15 -15 15
        zLinear  5  15 40
    end gridGeometry
end definitions
```

the description of the true source

```
read sources
    src 1
        neutrons  strength=4.05E+07
        cuboid  0.01 0.01  0 0  0 0
        eDistributionID=1
    end src
end sources
```

and finally the importance map block to trigger the Denovo calculation

```
read importanceMap
    gridGeometryID=7
    mmTolerance=0.001
end importanceMap
```

An optional shell command can be used to retrieve the cross section file and the Denovo input file.

```
=shell
  cp ft02f001       ${OUTDIR}/denovo1.ft02f001
  cp xkba_b.inp     ${OUTDIR}/denovo1.xkba.dsi
end
```

The `denovo1.xkba.dsi` file (the Denovo simple input) contains both the source and mesh geometry that MAVRIC prepared for Denovo, as shown in Fig. 4.4.5.

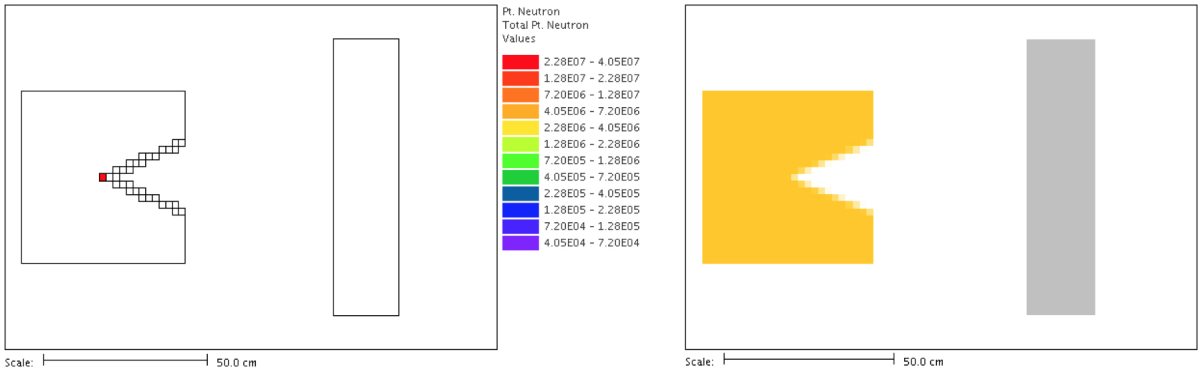


Fig. 4.4.5: Forward Denovo source (left) and mesh geometry (right).

The result of the Denovo-only MAVRIC calculation is a file, `denovo1.forward.dff`, containing the scalar fluxes for each energy group. The Mesh File Viewer can be used to display each group or the total scalar flux, which is shown in Fig. 4.4.6.

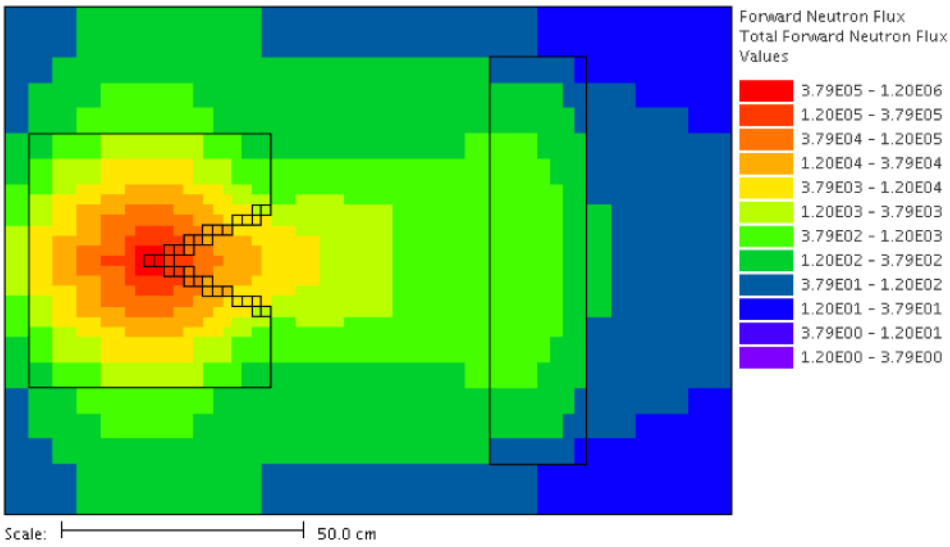


Fig. 4.4.6: Total forward fluxes.

The MAVRIC utilities can be used to further process the Denovo flux into a detector response. The fluxes at the location of the detector need to be multiplied by the photon flux-to-dose conversion factors (the response function) and then summed. This is can be done in the same input file using

```
=dffResp
"denovo1.forward.dff"
1
27
1.6151395E-04
1.4451494E-04
1.2703618E-04
1.2810882E-04
1.2983654E-04
1.0343020E-04
5.2655141E-05
1.2860853E-05
3.7358122E-06
3.7197628E-06
4.0085556E-06
4.2945048E-06
4.4731187E-06
4.5656334E-06
4.5597271E-06
4.5209654E-06
4.4872718E-06
4.4659614E-06
4.4342228E-06
4.3315831E-06
4.2027596E-06
4.0974155E-06
3.8398102E-06
3.6748431E-06
3.6748440E-06
3.6748431E-06
3.6748447E-06
"denovo1.doses.dff"
end

=dffInt
```

(continues on next page)

(continued from previous page)

```
" denovo1.doses.dff"
1
" denovo1.total.dff"
end

=dffPull
" denovo1.total.dff"
1
110.0 0.0 0.0
0
"denovo1.detectorDose.txt"
end
```

and results in a neutron dose rate of 5.426×10^{-3} rem/hr, calculated in about 2 minutes. Other combinations of the MAVRIC utilities can be used to simply "pull" out the fluxes from the detector location and then use a spreadsheet to compute the dose rate. With upscatter on, the result is 5.424×10^{-3} rem/hr, showing that upscatter does not contribute to dose rate at the detector. Note that with upscatter on, the Denovo calculation required 81 minutes.

Adjoint

For the same calculation of neutron dose as above, a Denovo adjoint calculation can be performed. The input file (`denovo2.inp`) has the same composition, geometry, definitions, and source blocks as the forward example. The adjoint input importance map block contains a description of the adjoint source:

```
read importanceMap
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
    gridGeometryID=7
    mmTolerance=0.001
end importanceMap
```

The result of this Denovo-only MAVRIC calculation is a file, `denovo2.adjoint.dff`, containing the scalar adjoint fluxes for each energy group, as shown in Fig. 4.4.7.

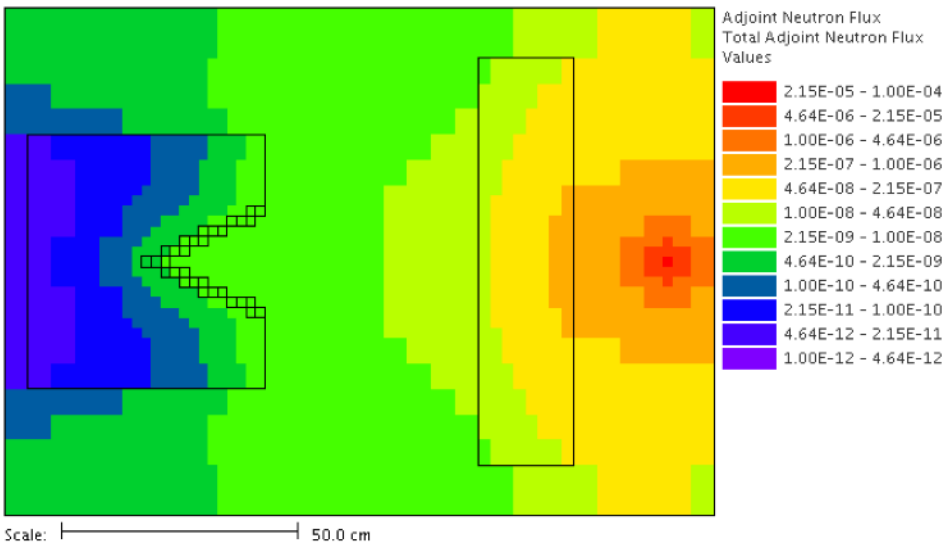


Fig. 4.4.7: Total adjoint fluxes.

As with the forward fluxes, the MAVRIC utilities can be used to further process the Denovo adjoint fluxes into a detector response. The adjoint fluxes at the location of the forward source need to be multiplied by the source distribution and strength and then summed. This is can be done using

```
=dffResp
"denovo2.adjoint.dff"
1
27
3.365455E-02
2.398546E-01
2.398289E-01
1.085244E-01
1.527957E-01
1.463752E-01
6.669337E-02
1.164470E-02
5.713197E-04
5.240384E-05
4.470401E-06
2.956226E-07
4.160008E-08
1.011605E-08
9.667897E-10
3.579175E-10
9.880883E-11
6.634570E-11
9.742191E-11
1.538334E-10
2.777230E-11
2.690255E-11
2.572180E-11
7.001169E-12
2.049478E-12
1.437105E-12
3.424200E-13
"denovo2.doses.dff"
end

=dffMult
" denovo2.doses.dff"
4.05E+07
" denovo2.doses.dff"
end

=dffInt
" denovo2.doses.dff"
1
" denovo2.total.dff"
end


=dffPull
" denovo2.total.dff"
1
0.0 0.0 0.0
0
"denovo2.detectorDose.txt"
end
```

and results in a neutron dose rate of $5.463 \cdot 10^{-3}$ rem/hr.

The results of the example problems above used a fairly coarse mesh ($44 \times 27 \times 27$) and the default values of S_8 and P_3. Run times were 2 minutes for the forward case and 2.5 minutes for the adjoint. With a finer mesh, larger quadrature order and larger numbers of Legendre moments for the scattering representation, the values calculated using Denovo should converge towards the Monte Carlo solution of $1.494 \cdot 10^{-2}$ rem/hr $\pm$ 1.2%.

5. DEPLETION, ACTIVATION, AND SPENT FUEL SOURCE TERMS

Introduction by W. A. Wieselquist

SCALE's general depletion, activation, and spent fuel source terms analysis capabilities are enabled through a family of modules related to the main ORIGEN depletion/irradiation/decay solver. The nuclide tracking in ORIGEN is based on the principle of explicitly modeling all available nuclides and transitions in the current fundamental nuclear data for decay and neutron-induced transmutation and relies on fundamental cross section and decay data in ENDF/B VII. Cross section data for materials and reaction processes not available in ENDF/B-VII are obtained from the JEFF-3.0/A special purpose European activation library containing 774 materials and 23 reaction channels with 12,617 neutron-induced reactions below 20 MeV. Resonance cross section corrections in the resolved and unresolved range are performed using a continuous-energy treatment by data modules in SCALE. All nuclear decay data, fission product yields, and gamma-ray emission data are developed from ENDF/B-VII.1 evaluations. Decay data include all ground and metastable state nuclides with half-lives greater than 1 millisecond. Using these data sources, ORIGEN currently tracks 174 actinides, 1149 fission products, and 974 activation products.

The purpose of this chapter is to describe the stand-alone capabilities and underlying methodology of ORIGEN-as opposed to the integrated depletion capability it provides in all coupled neutron transport/depletion sequences in SCALE, as described in other chapters. Through the stand-alone capabilities, there is generality to handle arbitrary systems (e.g., fast reactor fuel depletion or structural activation) by providing arbitrary flux spectra and arbitrary one-group cross sections to the module COUPLE, which in turn creates ORIGEN library (.f33) files containing the problem-dependent, one-group reaction coefficients required to solve the actual equations governing depletion/decay. These libraries are required input for the ORIGEN module, along with the initial isotopics and irradiation history, in terms of either a time-dependent power or flux level. Two high-performance equation solvers are available: the hybrid linear chains and matrix exponential method and a new Chebyshev Rational Approximation Method (CRAM). Typical execution times are on the order of a few seconds for a multi-step solution, with each individual solution (step) taking approximately 10 milliseconds. ORIGEN also includes capabilities for continuous feed and removal by element. Output capabilities include isotopics (moles or grams), source spectra (alpha, beta, gamma, and neutron), activity (becquerels or curies), decay heat (total watts or gamma only), and radiological hazard factors (maximum permissible concentrations in air or water). These results can be displayed in the output file (.out extension) and/or archived in an ORIGEN binary results file (.f71 extension).

The use of current, fundamental data resources is a key feature of ORIGEN, including nuclear decay data, multigroup neutron reaction cross sections, neutron-induced fission product yields, and decay emission data for photons, neutrons, alpha particles, and beta particles. The nuclear decay data are based primarily on ENDF/B-VII.1 evaluations. The multigroup nuclear reaction cross section libraries now include evaluations from the JEFF-3.0/A neutron activation file containing data for 774 target nuclides, more than 12,000 neutron-induced reactions, and more than 20 different reaction types below 20 MeV, provided in various energy group structures. Energy-dependent ENDF/B-VII.0-based fission product yields are available for 30 fissionable actinides. Gamma-ray and x-ray emission data libraries are based on ENDF/B-VII.1. The photon libraries contain discrete photon line energy and intensity data for decay gamma-ray and x-rays emission for 1,132 radionuclides, prompt and delayed continuum spectra for spontaneous fission, (α, n) reactions in oxide fuel, and bremsstrahlung from decay beta (electron and positron) particles slowing down in either a UO_2 fuel or water matrix. Methods and data libraries used to calculate the neutron yields and energy spectra for spontaneous fission, (α, n) reactions, and delayed neutron emission are adopted from the SOURCES4C code. Capabilities to calculate the beta and alpha particle emission source and spectra have also been added.

The ORIGEN reactor libraries distributed with SCALE include a set of pre-calculated ORIGEN libraries (with TRITON) for a variety of fuel assembly designs:

- BWR 7 × 7, 8 × 8-1, 8 × 8-2, 9 × 9-2, 9 × 9-9, 10 × 10-9, 10 × 10-8, SVEA-64, SVEA-96, and SVEA-100;
- PWR 14 × 14, 15 ×times` 16, 17 × 17, 18 × 18;
- CANDU reactor (19-, 28-, and 37-element bundle designs);
- Magnox graphite reactor;
- Advanced Gas-Cooled Reactor (AGR);
- VVER 440 and VVER 1000;
- RBMK;
- IRT;
- MOX BWR 7 × 7, 8 × 8-1, 8 × 8-2, 9 × 9-2, 9 × 9-9, 10 × 10-9, 10 × 10-8, SVEA-64, SVEA-96, and SVEA-100;
- MOX PWR 14 × 14, 15 × 15, 16 × 16, 17 × 17, 18 × 18.

These libraries may be used to rapidly assess spent fuel isotopics and source terms in these systems for arbitrary burnups and decay times. For UO_2-based assembly isotopics, the new ORIGAMI sequence provides a very convenient, easy-to-use interface. The most general capability, and requiring more user input, is available using the ARP interpolator module in conjunction with the ORIGEN solver module.

Finally, with regards to user interfaces, ORIGEN has a new keyword-based input in SCALE 6.2 but also maintains the ability to read SCALE 6.1 input. Both ORIGEN and ORIGAMI are tightly integrated with the SCALE graphical user interface, Fulcrum, which includes syntax highlighting, input checking with immediate feedback, and (.f71) output viewing. Additionally, Fulcrum provides an ORIGAMI Automator project interface to characterize the fuel inventory for an entire reactor site and generate data needed for severe accident analysis. ORIGAMI Automator is not documented in this chapter, but a primer is available with step by step instructions on its use.

5.1 ORIGEN: NEUTRON ACTIVATION, TRANSMUTATION, FISSION PRODUCT GENERATION, & RADIATION SOURCE TERM CALCULATION

ORIGEN (Oak Ridge Isotope Generation code) calculates time-dependent concentrations, activities, and radiation source terms for a large number of isotopes simultaneously generated or depleted by neutron transmutation, fission, and radioactive decay. ORIGEN is used internally within SCALE's TRITON and Polaris sequences to perform depletion and decay. As a stand-alone SCALE module, ORIGEN provides additional unique capabilities to (1) simulate continuous nuclide feed and chemical removal, which can be used to model reprocessing or liquid fuel systems, and (2) generate alpha, beta, neutron and gamma decay emission spectra. A standard decay library is provided to perform decay calculations. For neutron activation and fuel depletion problems, neutron spectrum-dependent ORIGEN libraries are required and may be created from (1) user-defined spectrum and self-shielded cross sections using the COUPLE module or (2) interpolation of existing ORIGEN reactor libraries (precalculated by TRITON) using the Automated Rapid Processing (ARP) module. Post-processing using the OPUS module allows calculated isotopics and spectra to be sorted, ranked, and converted to other units.

5.1.1 ACKNOWLEDGEMENTS

Development and maintenance of ORIGEN and related codes and methods have been sponsored by many organizations including the US Nuclear Regulatory Commission (NRC), the US Department of Energy (DOE), and nuclear power and research institutions.

5.1.2 VERSION INFORMATION

The ORIGEN (Oak Ridge Isotope Generation) code [ORIGEN-Bel73] was developed at Oak Ridge National Laboratory (ORNL) to calculate nuclide compositions and radioactivity of fission products, activation products, and products of heavy metal transmutation. Since 1991, ORIGEN has been developed as the depletion/decay module in SCALE with support from the NRC. ORIGEN in SCALE is the only version supported at ORNL, and it supersedes all earlier versions. The following is a brief description of the major enhancements in each version. Data are described in the ORIGEN Data Resources chapter.

5.1.2.1 Version 6.3 (2021)

Code Responsible(s): W. A. Wieselquist

Contributors: W. Wieselquist, S. Hart, K. Bledsoe, S. Skutnik, K. Bekar

Continued modernization focused on incorporating the capabilities of COUPLE into the ORIGEN sequence, improved data formats, and interoperation with ENDF-formatted data. The main new feature is a nuclear data sensitivity capability.

5.1.2.2 Version 6.2 (2016)

Code Responsible(s): W. A. Wieselquist

Contributors: W. Wieselquist, S. Hart, A. Isotalo[1], F. Havlůj[2], S. Skutnik, R. Lefebvre, I. Gauld, D. Wiarda, J. Lefebvre, G. Hu[4], N. Sly[3], and D. Lago[5]

A major modernization effort for ORIGEN was initiated by I. Gauld in 2011 and has resulted in approximately 5 person-years of effort refactoring the ORIGEN and related codes to be more efficient and easily testable. The major enhancements and responsible parties are listed below.

- Extensive refactor and modernization of Fortran 77 to Fortran 90+ performed by F. Havlůj, including substantial extension of the output capability
- Implementation of an alpha and beta spectrum calculation by F. Havlůj and I. Gauld
- Introduction of C++ core data structures with Fortran bindings, implemented by S. Skutnik using R. Lefebvre's C++/C/Fortran binding generator created for this purpose
- Testing suite developed by S. Skutnik, W. Wieselquist, D. Lago, and N. Sly
- Standardization of codebase while developing application programming interface (API) for high-performance depletion in the Consortium for Advanced Simulation of Light Water Reactors (CASL) and Nuclear Energy Advanced Modeling and Simulation (NEAMS) projects performed by W. Wieselquist
- Unification of readers/writers for ORIGEN data files developed by W. Wieselquist

[1] Aalto University, Finland
[2] ÚJV Řež, a. s., Czech Republic
[4] University of Illinois, Urbana-Champaign
[3] University of Tennessee, Knoxville
[5] Georgia Tech

- Improvement of binary formats for the ORIGEN library (f33) and ORIGEN concentration file (f71) by J. Lefebvre, R. Lefebvre, and W. Wieselquist
- Implementation of Chebyshev Rational Approximation Method (CRAM) solver by A. Isotalo
- Development of new input format (ORIGEN sequence only) by S. Hart and W. Wieselquist using the SCALE Object Notation (SON) syntax developed by R. Lefebvre
- Improvement of cubic spline interpolation scheme for ARP by S. Skutnik and W. Wieselquist with monotonicity fix-up determined by G. Hu
- Major revision of manuals by W. Wieselquist, combining ORIGEN, ARP, COUPLE, and OPUS into a single manual

Additional guidance provided by D. Wiarda and I. Gauld with testing by J. W. Hu.

5.1.2.3 Version 6.1 (2011)

The following section acknowledgements appeared in the SCALE 6.1 manual.

ORIGEN

Code Responsible(s): I. C. Gauld

The ORIGEN code was first developed by M. J. Bell with contributions from J. P. Nichols and other members of the Chemical Technology Division at ORNL. Development of the ORIGEN code as a depletion module of the SCALE code system was performed by O. W. Hermann with contributions from R. M. Westfall, supported by the NRC.

COUPLE

Code Responsible(s): D. Wiarda and I. C. Gauld

The COUPLE code was originally developed by O. W. Hermann with guidance from staff members including L. M. Petrie, N. M. Greene, W. E. Ford III, and R. M. Westfall, who contributed greatly to formulation of the methods, design of the data library interface with other modules, and testing. Many valuable suggestions concerning code applications were received from J. C. Ryman, J. R. Knight, and E. J. Allen.

ARP

Code Responsible(s): I. C. Gauld, S. M. Bowman, and J. E. Horwedel

The authors thank S. B. Ludwig for his support in earlier stages of this work. The authors are grateful for the technical advice received from B. L. Broadhead, M. D. DeHart, N. M. Greene, O. W. Hermann, C. V. Parks, L. M. Petrie, and J. C. Ryman. The authors thank Germina Ilas and Georgeta Radulescu for reviewing the manual and Willena Carter for preparation of the manuscript.

OPUS

Code Responsible(s): I. C. Gauld and J. E. Horwedel

The work of O. W. Hermann in developing the PLORIGEN program, from which OPUS was later developed, and the work of D. L. Barnett in developing the original version of PlotOPUS, are acknowledged. Appreciation is extended to J. C. Ryman for his review and testing of the program. Finally the authors thank S. J. Poarch for formatting the manuscript.

5.1.3 METHOD OF SOLUTION

ORIGEN solves the system of ordinary differential equations (ODEs) that describe nuclide generation, depletion, and decay,

$$\frac{dN_i}{\mathrm{dt}} = \sum_{j \neq i} (l_{\mathrm{ij}} \lambda_j + f_{\mathrm{ij}} \sigma_j \Phi) N_j(t) - (\lambda_i + \sigma_i \Phi) N_i(t) + S_i(t) \tag{5.1.1}$$

where

- N_i = amount of nuclide i (atoms),
- λ_i = decay constant of nuclide i (1/s),
- l_{ij} = fractional yield of nuclide i from decay of nuclide j,
- σ_i = spectrum-averaged removal cross section for nuclide i (barn),
- f_{ij} = fractional yield of nuclide i from neutron-induced removal of nuclide j,
- Φ = angle- and energy-integrated time-dependent neutron flux (neutrons/cm^2-s), and
- S_i = time-dependent source/feed term (atoms/s).

Note that Eq. (5.1.1) has no spatial dependence and can be interpreted as either a solution at a point in space or the spatial average over some volume. The latter interpretation is preferred here, such that Φ is the spatially averaged neutron flux magnitude, and all energy-dependence is embedded in the one-group flux-weighted average cross sections σ_i and reaction yields f_{ij}. Eq. (5.1.1) is conveniently written in matrix form as

$$\frac{d\vec{N}}{dt} = \mathbf{A}\vec{N}(t) + \vec{S}(t) \tag{5.1.2}$$

with a $\mathbf{A}$ commonly referred to as the "transition matrix." The representation of the transition matrix as $\mathbf{A} = \mathbf{A}_\sigma \Phi + \mathbf{A}_\lambda$, where $\mathbf{A}_\sigma$ is the part of the transition matrix containing reaction terms and $\mathbf{A}_\lambda$ is the part containing decay terms, is convenient, as the numerical solution of this system of ODEs holds the reaction, flux, and feed terms constant over step n,

$$\frac{d\vec{N}}{\mathrm{dt}} = (\mathbf{A}_{\sigma,n} \Phi_n + \mathbf{A}_\lambda) \vec{N}(t) + \vec{S}_n \tag{5.1.3}$$

over time step $t_{n-1} \leq t \leq t_n$.

Adding a continuous removal process described with rate constant $\lambda_{i,rem}$ simply modifies the decay constant, $\lambda_i \rightarrow \lambda_i + \lambda_{i,rem}$, whereas a continuous feed process defines a nonzero component of the $\vec{S}$ vector.

ORIGEN can also compute the alpha, beta, neutron, and gamma emission spectra during decay. For the "stand-alone" ORIGEN calculations described here, the transition matrix is loaded from an ORIGEN binary library file (f33), which uses sparse-matrix storage to store one or more transition matrices. The f33s may be created using COUPLE, saved from TRITON depletion calculations, or interpolated using ARP from a set of precompiled f33s distributed with SCALE.

Results from ORIGEN calculations may be stored on a binary concentration file (f71), which facilitates transfer of isotopics to other codes in SCALE. The f71 file can also store calculated emission spectra. Within ORIGEN, the f71 can be used to restart calculations from an existing set of compositions.

5.1.3.1 Methodology

This section describes the methodology used in performing the following main functions:

- generation of problem-dependent transition matrices,
- solution of the system of depletion/decay equations,
- conversion from power to flux (important for reactor applications),
- calculation of emission spectra, and
- interpolation of pregenerated sets of transition matrices.

Generation of Problem-dependent Transition Matrix

In the transition matrix **A** from Eq. (5.1.2), each matrix element a_{ij} is the first-order rate constant for the formation of nuclide i from nuclide j given below.

$$a_{ij} = \begin{cases} l_{ij}\lambda_j + f_{ij}\sigma_j\Phi & i \neq j \\ \lambda_i - \sigma_i\phi & \text{otherwise} \end{cases} \tag{5.1.4}$$

The transition matrix coefficients for decay and reaction transitions are stored separately and reaction transitions are always stored with $\Phi = 1$ and later during solution of the system, depending on the step-average flux level the actual transition matrix $\mathbf{A}_n = \mathbf{A}_{\sigma,n}\Phi_n + \mathbf{A}_\lambda$ is reconstructed using step-average flux, Φ_n.

The decay coefficients $l_{\mathrm{ij}}\lambda_j$ and λ_i are generated directly from ORIGEN decay resource data. The reaction coefficients $f_{\mathrm{ij}}\sigma_j$ and σ_i are generated using the following two-stage procedure.

1. Calculate all removal cross sections σ_i and yields f_{ij}, including isomeric branching ratios and fission yields, by folding provided flux spectrum ϕ^g with multigroup cross sections from the ORIGEN reaction resource and energy-dependent fission yield data from the ORIGEN yield resource.
2. Overwrite specific removal cross sections and yields based on a provided multigroup cross section library (see the SCALE Nuclear Data Libraries chapter) and/or user-provided one-group cross sections and yields.

The second stage is optional, but it is important for cases which there is significant self-shielding because ORIGEN's reaction resource assumes infinite dilution for its multigroup data. The decay, reaction, and yeld resources mentioned here are described in the ORIGEN Data Resources chapter. The collapse to a one-group cross section in either stage is given by

$$\sigma_{\mathrm{ri}} = \frac{\sum_g \sigma_{\mathrm{ri}}^g \phi^g}{\sum_g \phi^g} \tag{5.1.5}$$

for reaction type r, nuclide i, and provided multigroup flux ϕ^g. Different reaction types are recognized by their ENDF MT numbers [SCALE Cross Section Libraries chapter] on the appropriate data resource For example, MT=16 is $(n, 2n)$, and MT=107 is (n, α). The removal cross section σ_i is simply calculated as the sum over all relevant reactions for a particular nuclide, $\sigma_i = \sum_r \sigma_{\mathrm{ri}}$. This type of reaction-dependent multigroup data may be contained in either the data sources available in stage 1 or 2 above. However, only two types of data are expected to be available in stage 1 reaction resource data: (1) isomeric branching and (2) fission yields.

The energy-dependent isomeric branching that describes the yield of each excited level (metastable state) of a daughter nucleus is calculated in a similar way,

$$f_{\text{rim}} = \frac{\sum_g f_{\text{rim}}^g \sigma_{\text{ri}}^g \phi^g}{\sum_g \sigma_{\text{ri}}^g \phi^g} \tag{5.1.6}$$

where m indicates the possible metastable states and the fractions always satisfy $\sum_m f_{\text{rim}} = 1$.

Fission product yields are typically tabulated at discrete neutron energies such as thermal (0.0253 eV), fission (500 keV), and high energy (14 MeV). The yield for each fissionable nuclide is calculated in stage 1 by linearly interpolating the tabulated data using the computed average energy of fission,

$$\overline{E}_{\text{fi}} = \frac{\sum_g \overline{E}^g \sigma_{\text{fi}}^g \phi^g}{\sum_g \sigma_{\text{fi}}^g \phi^g} \tag{5.1.7}$$

where σ_{fi}^g is the multigroup fission cross section, and $\overline{E}^g$ is the average energy in the group (simple midpoint energy used). In addition to generating transition data for daughter/residual nuclides, the coefficients for byproducts such as He-4/α byproducts from (n, α) reactions are also retained in the transition matrix and associated to an appropriate nuclide in the system: hydrogen, deuterium, tritium, ^{3}He, or ^{4}He.

Solution of the Depletion/Decay Equations

ORIGEN includes two solver kernels that can solve the depletion/decay equations of Eq. (5.1.3):

1. a hybrid matrix exponential/linear chains method (MATREX) and
2. a Chebyshev Rational Approximation Method (CRAM).

They are described in the following sections.

MATREX

Referring to the system of ODEs shown in Eq. (5.1.2) and setting the external feed/source $S\ (t) = 0$, there is a formal solution by matrix exponential (an analog to the solution of a single ODE of this type by exponential),

$$\overrightarrow{N}(t) = \exp(\mathbf{A}t)\,\overrightarrow{N}(0) \tag{5.1.8}$$

where $\overrightarrow{N}(0)$ is a vector of initial nuclide concentrations, by defining the series expansion of $exp(\mathbf{A}t)$ to be

$$\exp(\mathbf{A}t) = \mathbf{I} + \mathbf{A}t + \frac{(\mathbf{A}t)^2}{2} + \ldots = \sum_{k=0}^{\infty} \frac{(\mathbf{A}t)^k}{k!} \tag{5.1.9}$$

with $\mathbf{I}$ the identity matrix. Eq. (5.1.8) and Eq. (5.1.9) describe the matrix exponential method, which yields a complete solution to the problem. However, in certain instances related to limitation in computer precision, difficulties occur in generating accurate values of the matrix exponential function. Under these circumstances, alternative procedures using either the generalized Bateman equations [ORIGEN-Bat10] or Gauss-Seidel iterative techniques are applied. These alternative procedures will be discussed in further sections.

A straightforward solution of Eq. (5.1.8) and Eq. (5.1.9) would require storage of the complete transition matrix. To avoid excessive memory requirements, a recursion relation has been developed. Substituting Eq. (5.1.9) into Eq. (5.1.8),

$$\overrightarrow{N}(t) = \left[\mathbf{I} + \mathbf{A}t + \frac{(\mathbf{A}t)^2}{2} + \ldots\right]\overrightarrow{N}(0) \tag{5.1.10}$$

one may recognize a recursion relation for a particular nuclide, $N_i(t)$.

$$N_i(t) = N_i(0) + t\sum_j a_{ij}N_j(0) + \frac{t}{2}\sum_k\left[a_{ik}t\sum_j a_{kj}N_j(0)\right] + \frac{t}{3}\sum_m\left\{a_{im}\frac{t}{2}\sum_k\left[a_{mk}t\sum_j a_{kj}N_j(0)\right]\right\} + \ldots \tag{5.1.11}$$

where the range of indices, j, k, m, is 1 to M for matrix $\mathbf{A}$ of size $M \times M$. The result is a series of terms that arise from the successive post-multiplication of the transition matrix by the vector of nuclide concentration increments produced from the computation of the previous terms. Within the accuracy of the series expansion approximation, physical values of the nuclide concentrations are obtained by summing a converged series of these vector terms. By defining the terms $C_i^n(t)$ as

$$\begin{aligned} C_i^0 &= N_i(0) \\ C_i^{n+1} &= \frac{t}{n+1}\sum_j a_{\mathrm{ij}}C_j^n \end{aligned} \tag{5.1.12}$$

the solution for $N_i(t)$ is given as

$$N_i(t) = \sum_{n=0}^{\infty} C_i^n \tag{5.1.13}$$

The use of Eq. (5.1.12) and Eq. (5.1.13) requires storage of only two vectors–$\overrightarrow{C}^n$ and $\overrightarrow{C}^{n+1}$ —-in addition to the current value of the solution. However, the series summation solution in Eq. (5.1.13) is not valid until a finite limit is identified which can achieve a reasonable accuracy, i.e.,

$$N_i(t) = \sum_{n=0}^{n_{\mathrm{term}}} C_i^n + \epsilon_{\mathrm{trunc}} \tag{5.1.14}$$

where n_{term} is the number of terms and $\epsilon_{\mathrm{trunc}}$ is the truncation error. The key is to split the nuclides into two sets: those that are long-lived and permit a rapid, accurate solution via Eq. (5.1.14), and those that are short-lived and require an alternate solution.

Solution for Long-Lived Nuclides

This section describes the various tests used to ensure that the summations indicated in Eq. (5.1.14) do not lose accuracy due to large changes in magnitudes or small differences between positive and negative rate constants. Nuclides with large rate constants (short-lived) are removed from the transition matrix and treated separately. For example, in the decay chain A → B → C, if the decay constant for B is large, a new rate constant is inserted in the matrix for A → C. This technique was originally employed by Ball and Adams [ORIGEN-BA67]. The key to determining which transitions should be removed involves calculation of the matrix norm. The norm of matrix $\mathbf{A}$ is defined by Lapidus and Luus [ORIGEN-LL67] as being the smaller of the maximum-row absolute sum and the maximum-column absolute sum,

$$[\mathbf{A}] = \min\left\{\max_j \sum_i \left|a_{\mathrm{ij}}\right|, \max_i \sum_j \left|a_{\mathrm{ij}}\right|\right\} \tag{5.1.15}$$

To maintain precision in performing the summations of Eq. (5.1.14), the matrix norm is used to balance the user-specified time step, t, with the precision associated with the word len>h employed in the machine calculation. The constraint on the matrix norm has been chosen as

$$[\mathbf{A}]\, t \leq \; -2 \; \ln(0.001) = 13.8155 \tag{5.1.16}$$

The remainder of this section shows that this constraint serves two purposes.

- It allows reasonable accuracy for a reasonable number (20–60) of matrix exponential terms.
- It defines what "short-lived" means over a particular time step, dictating which concentrations must be solved by alternate means.

A relationship between m digits of machine precision and p significant digits required in all results can be stated by the following inequality:

$$(\text{Largest term in series}) \times 10^{-m} \leq (\text{Series result}) \times 10^{-p} \tag{5.1.17}$$

In this particular series, the relationship may be represented as

$$\max_n \frac{|[\mathbf{A}]\, t|^n}{n!} 10^{-m} \leq \; e^{-[\mathbf{A}]t} 10^{-p} \text{`}, \tag{5.1.18}$$

or alternatively,

$$\max_n \frac{|[\mathbf{A}]\, t|^n}{n!} e^{[\mathbf{A}]t} \leq \; 10^{m-p}. \tag{5.1.19}$$

Lapidus and Luus have shown that the maximum term in the summation for any element in the matrix exponential function cannot exceed $\frac{([\mathbf{A}]t)^n}{n!}$, where n is the largest integer not larger than $[\mathbf{A}]\, t$. For the constraint in Eq. (5.1.16), this yields n=13 and yields limit $\frac{([\mathbf{A}]t)^n}{n!} \approx 10^5$. With $e^{[\mathbf{A}]t} \approx 10^6$ and standard double precision with *m=16*, Eq. (5.1.19) evaluates to $10^{11} \leq 10^{16-j}$, which indicates that five significant figures will be maintained in values as small as 10^{-6}. The number of terms required to converge the matrix exponential series can be investigated by a plot of the $\frac{|[\mathbf{A}]t|^n}{n!} e^{[\mathbf{A}]t}$ as a function of term index n in Eq. (5.1.19), as shown in Fig. 5.1.1

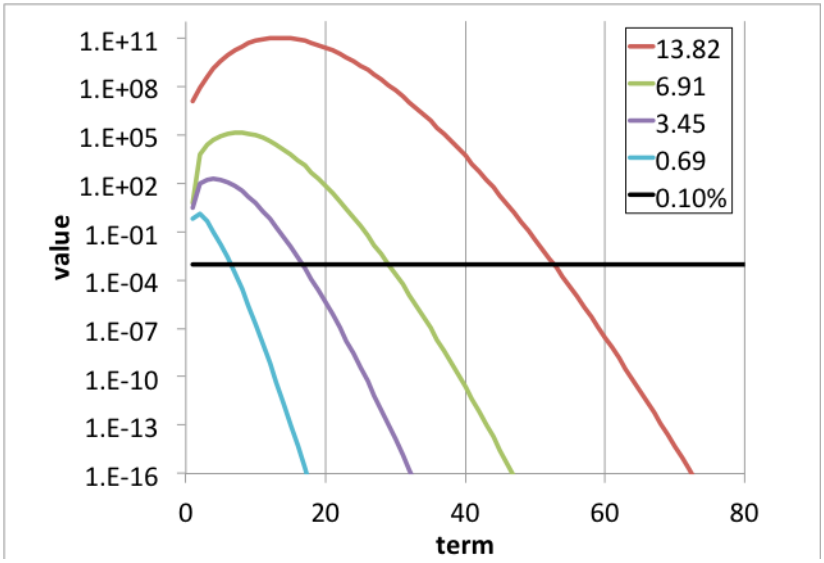


Fig. 5.1.1: Values of terms in series for various values of the matrix norm.

The intersection between the black line in Fig. 5.1.1 and the various curves indicates the number of terms needed to achieve $\epsilon_{\text{trunc}} \leq 0.1\%$. For example, with $[\mathbf{A}]\,t = 13.8155$, $n_{\text{term}} = 54$ is required, and with $[\mathbf{A}]\,t = 13.8155/2$, approximately $n_{\text{term}} = 29$ is required. This behavior has been used to develop the heuristic

$$n_{\text{term}} = 7\ [\mathbf{A}]t/2 + 6. \tag{5.1.20}$$

Thus it has been shown that the limit imposed in Eq. (5.1.16) leads to a maximum of $n_{\text{term}} = 54$ terms with $\epsilon_{\text{trunc}} \leq 0.1\%$.

It remains to be shown that any arbitrary system can be modified so that it does not violate Eq. (5.1.16). Because the time step t is provided and fixed, $[\mathbf{A}]\,t \leq -2\ \ln(0.001)$ cannot be satisfied unless the system is modified. The physical nature of the system leads to $\max_j \sum_i \left|a_{\text{ij}}\right| \leq \max_j 2|a_{\text{jj}}|$ based on production rates equal to loss rates when both parent and daughter nuclide are included in the system. The maximum column sum in Eq. (5.1.15) can then be bounded by twice the maximum diagonal term, $\max_j 2|a_{\text{jj}}|$. Using this upper limit as the matrix norm and substituting into Eq. (5.1.16) yields

$$[\mathbf{A}]\,t \leq 2 \max_j \left|a_{\text{jj}}\right| \leq -2 \ln(0.001) \tag{5.1.21}$$

Rearranging Eq. (5.1.21) leads to the condition

$$e^{-\|a_{jj}\|t} < 0.001 \tag{5.1.22}$$

which is used to mark nuclide j as a short-lived nuclide for this time step, to be solved with linear chains instead of the series-based matrix exponential. An alternative interpretation of the short-lived condition can be made by rewriting Eq. (5.1.22) in terms of an effective half-life, $t_{1/2} = \frac{\ln(2)}{|a_{\text{jj}}|}$, which results in $t_{1/2} < \frac{-ln(2)}{\ln(0.001)} t \approx 0.1t$. In other words, when a nuclide's effective half-life (including destruction by both decay and reaction mechanisms) is less than 10% of the time step, it can be considered short-lived.

Finally, as a note for applications where the nuclides of interest are in long transmutation chains, it has been found that the above algorithm may not yield accurate concentrations for those nuclides near the end of the chain that are significantly affected by those near the beginning of the chain. In these applications, specifying the minimum n_{term} as

$$n_{\text{term}} \geq |\Delta Z| + |\Delta A| + 5 \tag{5.1.23}$$

where ΔZ is the atomic number difference and ΔA is the mass number difference, has been found to ameliorate the issue.

Solution for Short-Lived Nuclides

The condition in Eq. (5.1.22) forms the basis for declaring a nuclide short-lived, and its solution is found via solution of the nuclide chain equations. In conjunction with maintaining the transition matrix norm below the prescribed level, a queue is formed of the short-lived precursors of each long-lived isotope. These queues extend back up the several chains to the last preceding long-lived precursor. According to Eq. (5.1.22), the queues will include all nuclides whose effective half-lives are less than 10% of the time interval. A generalized form of the Bateman equations developed by Vondy [ORIGEN-Von62] is used to solve for the concentrations of the short-lived nuclides at the end of the time step. For an arbitrary forward-branching chain, Vondy's form of the Bateman solution is given by,

$$N_i(t) = N_i(0)\, e^{-d_i t} + \sum_{k=1}^{i-1} N_k(0) \left[\sum_{j=k}^{i-1} \frac{e^{-d_j t} - e^{-d_i t}}{d_i - d_j} a_{j+1,j} \prod_{\substack{n=k \\ n \neq j}}^{i-1} \frac{a_{n+1,n}}{d_n - d_j} \right] \tag{5.1.24}$$

where $N_1\,(0)$ is the initial concentration of the first precursor, $N_2\,(0)$ is that of the second precursor, etc.

As in Eq. (5.1.4), a_{ij} is the first-order rate constant, and $d_i = -a_{\mathrm{ii}}$ which is the magnitude of the diagonal element. Bell recast Vondy's form of the solution through multiplication and division by $\prod_{n=k}^{i-1} d_n$ and rearranged to obtain

$$N_i\,(t) = N_i\,(0)\,e^{-d_i t} + \sum_{k=1}^{i-1} N_k(0) \prod_{n=k}^{i-1} \frac{a_{n+1,n}}{d_n} \left[\sum_{j=k}^{i-1} d_j \frac{e^{-d_j t} - e^{-d_i t}}{d_i - d_j} \prod_{\substack{n=k \\ n \neq j}}^{i-1} \frac{d_n}{d_n - d_j} \right] \tag{5.1.25}$$

The first product over isotopes n is the fraction of atoms that remains after the *kth* particular sequence of decays and captures. If this product becomes less than 10^{-6}, the contribution of this sequence to the concentration of nuclide i is neglected. Indeterminate forms that arise when *di=dj* or *dn=dj* are evaluated using L'Hôpital's rule. These forms occur when two isotopes in a chain have the same diagonal element.

Eq. (5.1.25) is applied to calculate all contributions to the "queue end-of-interval concentrations" of each short-lived nuclide from the initial concentrations of all others in the queue described above. It is also applied to calculate contributions from the initial concentrations of all short-lived nuclides in the queue to the long-lived nuclide that follows the queue, in addition to the total contribution to its daughter products. These values are appropriately applied either before or after the matrix expansion calculation is performed to correctly compute concentrations of long-lived nuclides and the long-lived or short-lived daughters. Eq. (5.1.25) is also used to adjust to certain elements of the final transition matrix, which now excludes the short-lived nuclides. The value of the element must be determined for the new transition between the long-lived precursor and the long-lived daughter of a short-lived queue. The element is adjusted so that the end-of-interval concentration of the long-lived daughter calculated from the single link between the two long-lived nuclides (using the new element) is the same as what would be determined from the chain including all short-lived nuclides. The method assumes zero concentrations for precursors to the long-lived precursor. The computed values asymptotically approach the correct value with successive steps through time. For this reason, at least five to ten time intervals during the decay of discharged fuel is reasonable, because long-lived nuclides have built up by that time.

If a short-lived nuclide has a long-lived precursor, an additional solution is required. First, the amount of short-lived nuclide i due to the decay of the initial concentration of long-lived precursor j is calculated as

$$N_{j\to i}\,(t) = N_j\,(0)\,a_{ij} \frac{e^{-d_j t}}{d_i - d_j} \tag{5.1.26}$$

from Eq. (5.1.24), assuming $e^{-d_i t} \ll e^{-d_j t}$. However, the total amount of nuclide i produced depends on the contribution from the precursors of precursor j, in addition to that given by Eq. (5.1.26). The quantity of nuclide j not accounted for in Eq. (5.1.26) is denoted by $N'_j\,(t)$, the end-of-interval concentration, minus the amount that would have remained had there been no precursors to nuclide j:

$$N_j^{'(t)} = N_j\,(t) - N_j(0) e^{-d_j t} \tag{5.1.27}$$

Then the short-lived daughter and subsequent short-lived progeny are assumed to be in secular equilibrium with their parents, which implies that the time derivative is zero,

$$\frac{dN_i}{\mathrm{dt}} = \sum_j a_{\mathrm{ij}} N_j(t) = 0. \tag{5.1.28}$$

The queue end-of-interval concentrations of all the short-lived nuclides following the long-lived precursor are augmented by amounts calculated with Eq. (5.1.25). The concentration of the long-lived precursor used in Eq. (5.1.27) is that given by Eq. (5.1.26). The set of linear algebraic equations given by Eq. (5.1.28) is solved by the Gauss-Seidel iterative technique. This algorithm involves an inversion of the diagonal terms and an iterated improvement of an estimate for $N_i(t)$ through the expression

$$N_i^{k+1} = -\frac{1}{a_{\text{ii}}} \sum_j a_{\text{ij}} N_j^k \tag{5.1.29}$$

Since short-lived isotopes are usually not their own precursors, this iteration often reduces to a direct solution.

Solution of the Nonhomogeneous Equation

The previous sections have presented the solution of the homogeneous equation in Eq. (5.1.8), applicable to fuel burnup, activation, and decay calculations. However, the solution of a nonhomogeneous equation is required to simulate reprocessing or other systems that require an external feed term, $S\,(t) \neq 0$. The nonhomogeneous equation is given in matrix form (assumed constant over a step n) as

$$\frac{d\overrightarrow{N}}{\text{dt}} = \mathbf{A}\overrightarrow{N}\,(t) + \overrightarrow{S} \tag{5.1.30}$$

for a fixed feed or removal rate, $\overrightarrow{S}$. A particular solution of Eq. (5.1.30) will be determined and added to the solution of the homogeneous equation given by Eq. (5.1.10). As before, the matrix exponential method is used for the long-lived nuclides, and solution by linear chains is used for the short-lived nuclides. Assume $\overrightarrow{C}$ an arbitrary vector with which to test a particular solution of the form

$$\overrightarrow{N}\,(t) = \sum_{k=0}^{\infty} \frac{(\mathbf{A}t)^k}{(k+1)!} \overrightarrow{C}\,t \tag{5.1.31}$$

Substituting Eq. (5.1.31) into Eq. (5.1.30) yields

$$\sum_{k=0}^{\infty} \frac{A^k t^k}{k!} \overrightarrow{C} = \sum_{k=0}^{\infty} \frac{A^{k+1} t^{k+1}}{(k+1)!} \overrightarrow{C} + \overrightarrow{S} \tag{5.1.32}$$

in which the *k=0* term may be extracted from the LHS,

$$\overrightarrow{C} + \sum_{k=1}^{\infty} \frac{A^k t^k}{k!} \overrightarrow{C} = \sum_{k=0}^{\infty} \frac{A^{k+1} t^{k+1}}{(k+1)!} \overrightarrow{C} + \overrightarrow{S} \tag{5.1.33}$$

which allows the summations on the left and right to be easily shown equal. This proves the particular solution is indeed valid if the arbitrary vector is in fact the feed term $\overrightarrow{C} = \overrightarrow{S}$. The solution to the nonhomogeneous problem is therefore (as a series),

$$\overrightarrow{N}\,(t) = \sum_{k=0}^{\infty} \frac{(\mathbf{A}t)^k}{k!} \overrightarrow{N}\,(0) + \sum_{k=0}^{\infty} \frac{(\mathbf{A}t)^k}{(k+1)!} \overrightarrow{S}\,t \tag{5.1.34}$$

For the second term in Eq. (5.1.34), a new recursion relation is developed for the particular solution in the same manner as was done for the homogeneous solution,

$$N_i^P\,(t) = \sum_{n=1}^{\infty} D_i^n \tag{5.1.35}$$

where

$$D_i^1 = S_i t;\ D_i^{n+1} = \frac{1}{n+1} \sum_j a_{ij} D_j^n \tag{5.1.36}$$

For the short-lived nuclides, the secular equilibrium equations are modified to become

$$\frac{dN_i}{\mathrm{dt}} = \sum_j a_{\mathrm{ij}} N_j(t) + S_i = 0. \tag{5.1.37}$$

The Gauss-Seidel iterative method is applied to determine the solution. The complete solution to the nonhomogeneous equation in Eq. (5.1.31) is given by the sum of the homogeneous solutions described in previous sections and the particular solutions described here.

CRAM

The solver kernel based on the Chebyshev Rational Approximation Method (CRAM) is described in detail in references [ORIGEN-IA11, ORIGEN-Pus11, ORIGEN-Pus13, ORIGEN-PL10]. Compared to the MATREX solver, CRAM generally has similar runtimes but is more accurate and robust on a larger range of problems. CRAM relies on the lower upper (LU) decomposition, so the SuperLU library has been used. The accuracy of CRAM is related to the order, with an order 16 solution having a truncation error less than 0.01% for all nuclides in most problems.

Unlike many methods for solving this type of system of ODEs, the len>h of a step does not significantly affect the accuracy of CRAM. However, any significant errors from CRAM will shrink rapidly over multiple steps as long as there are no large changes in reaction rates. The CRAM solver has an efficient internal substepping algorithm that can perform multiple same-sized substeps (with the same transition matrix) very efficiently by reusing the LU decomposition. When using internal substepping, 2–4 substeps are typical, with a large gain in accuracy for marginal increase in runtime.

Power Calculation

The following is formula is used to calculate power during irradiation ($\Phi > 0$),

$$P(t) = \sum_i \left(\kappa_{fi}\sigma_{fi} + \kappa_{ci}\sigma_{ci}\right)\phi N_i(t) \tag{5.1.38}$$

where κ_{fi} and κ_{ci} are nuclide-dependent energy released per fission and "capture," with *capture* defined as removal minus fission: $\sigma_{\mathrm{ci}} = \sigma_i - \sigma_{\mathrm{fi}}$. The σ_{fi} and σ_{ci} terms are extracted from the transition matrix itself, whereas the κ_{fi} and κ_{ci} are available from a separate ORIGEN energy resource (see ORIGEN Data Resources chapter). If the flux ϕ is specified, then the power can be calculated at any time according to Eq. (5.1.38). However in reactor fuel systems, it is convenient to be able to specify the power produced by the system and internally to the depletion code, to convert the power to an equivalent flux. Solving Eq. (5.1.38) for the flux, however,

$$\Phi(t) = \frac{P}{\sum_i \left(\kappa_{\mathrm{fi}}\sigma_{\mathrm{fi}} + \kappa_{\mathrm{ci}}\sigma_{\mathrm{ci}}\right) N_i(t)} \tag{5.1.39}$$

it is apparent that a fixed power over a time step n does not lead to a fixed flux, due to changing isotopics that produce different amounts of power per fission and capture. ORIGEN performs a flux-correction calculation to obtain an estimate of the average flux over the step. The beginning-of-step flux is first calculated for the initial compositions: Eq. (5.1.39) is evaluated as $\Phi(t_{n-1})$, and then Eq. (5.1.38) is solved with that flux. The

flux is then recalculated at the end of step $\Phi(t_n)$ using the estimated end-of-step isotopics, and the step-average flux Φ_n is estimated as the simple average of the beginning and end-of-step fluxes, i.e. Eq. (5.1.40),

$$\Phi_n = 0.5[\Phi(t_n) + \Phi^{\text{pred}}(t_{n+1})] \tag{5.1.40}$$

noting that the "predicted" flux at end-of-step $\Phi^{\text{pred}}(t_{n+1})$ is based on "predicted" end-of-step isotopics, based on a beginning-of-step flux level.

Decay Emission Sources Calculation

ORIGEN can calculate the emission sources (and spectra) during decay for alpha, beta, neutron, and photon particles according to

$$R_x^g(t) = \sum_i \lambda_i N_i(t) \int_{E^g}^{E^{g-1}} w_{i,x}(E)\,\mathrm{dE} \tag{5.1.41}$$

where $w_{i,x}(E)$ is the number of particles of type x emitted per disintegration of nuclide i at energy E, using provided energy bins defined by energy bounds E^g to E^{g-1}, where g is an energy index. The fundamental data resources for performing emission source calculations are described in the ORIGEN Data Resources chapter.

Neutron Sources

Computed neutron sources include neutrons spontaneous fission, (α, n) reactions, and delayed (β^-, n) neutron emission,

$$w_{i,n}(E) = w_{i,SFn}(E) + w_{i,(\alpha,n)}(E) + w_{i,Dn}(E) \tag{5.1.42}$$

with components that will be described below. The method of computing the spontaneous fission and delayed neutron source is independent of the medium containing the fuel. However, (α, n) production varies significantly with the composition of the medium. The homogeneous medium (α, n) calculation methodology has been adopted from the Los Alamos code SOURCES 4B [ORIGEN-Sho00, ORIGEN-WPC+99, ORIGEN-WPS+83].

The total yield of spontaneous fission neutrons from decay of nuclide i is

$$Y_{i,SFn} = \frac{\lambda_{i,SFn}}{\lambda_i}, \tag{5.1.43}$$

where $\frac{\lambda_{i,SFn}}{\lambda_i}$ is the fraction of decays which undergo spontaneous fission. The distribution of spontaneous fission neutrons, $w_{i,SFn}(E)$ is given by a Watt fission spectrum,

$$w_{i,SFn}(E) = Y_{i,SFn}\, C_i\, e^{-E/A_i} \sinh \sqrt{B_i E} \tag{5.1.44}$$

where Ai, Bi, and Ci are model parameters.

The (α, n) neutron source is strongly dependent on the low-Z content of the medium containing the alpha-emitting nuclides and requires modeling the slowing down of the alpha particles and the probability of neutron production as the α particle slows down. The calculation assumes (1) a homogeneous mixture in which the alpha-emitting nuclides are uniformly intermixed with the target nuclides and (2) that the dimensions of the

target are much larger than the range of the alpha particles. Thus, all alpha particles are stopped within the mixture. The yield of a particular α is given by

$$Y_{i,\alpha}^{\ell} = f_{i,\,\alpha}^{\ell} \frac{\lambda_{i,\alpha}}{\lambda_i} \tag{5.1.45}$$

where $\frac{\lambda_{i,\alpha}}{\lambda_i}$ is the relative probability of α–decay, and $f_{i,\,\alpha}^{\ell}$ is the fraction of those α–decays producing an α particle with initial energy $E_{i,\alpha}^{\ell}$, and is considered fundamental data. The *total* neutron yield from an alpha particle ℓ emitted by nuclide i and interacting with target k is given by the following,

$$Y_{i,k,(\alpha,n)}^{\ell} = Y_{i,\alpha}^{\ell} \frac{N_k}{N} \int_0^{E_{i,\alpha}^{\ell}} \frac{\sigma_{k,(\alpha,n)}(E_\alpha)}{S(E_\alpha)} dE_\alpha \tag{5.1.46}$$

where $S\,(E_\alpha)$ is the total stopping power of the medium, $\sigma_{k,(\alpha,n)}(E_\alpha)$ is the (α, n) reaction cross section for target nuclide k, and $\frac{N_k}{N}$ is the fraction of atoms in the medium composed of nuclide k. This expression is used to calculate the neutron yield for each target nuclide and from each discrete-energy alpha particle emitted by all alpha-emitting nuclides in the material. The stopping power for compounds, rather than pure elements, is approximated using the Bragg-Kleeman additivity rule. The energy-dependent elemental stopping cross sections are determined as parametric fits to evaluated data. Eq. (5.1.46) is solved for the total neutron yields from the alpha particle, as it slows down in the medium by subdividing the maximum energy $E_{i,\alpha}^{\ell}$ into a number of discrete energy bins and evaluating stopping power and (α, n) reaction cross section at the midpoint energy of the bin. The distribution of (α, n) neutrons as required by Eq. (5.1.41) is

$$w_{i,(\alpha,n)}\,(E) = \sum_k \sum_{\ell \in i} Y_{i,k}^{\ell}\,(\alpha, n)\, X_{i,k}^{\ell}\,(\alpha, n)\,(E) \tag{5.1.47}$$

with the distribution of (α, n) neutrons in energy, $X_{i,k,(\alpha,n)}^{\ell}\,(E)$, calculated using nuclear reaction kinematics, assuming that the (α, n) reaction emits neutrons with an isotropic angular distribution in the center-of-mass system. The maximum and minimum permissible energies of the emitted neutron are determined by applying mass, momentum, and energy balance for each product's nuclide energy level. The product nuclide levels, the product level branching data, the (α, n) reaction Q values, the excitation energy of each product nuclide level, and the branching fraction of (α, n) reactions result in the production of product levels. A more detailed discussion of the theory and derivation of the kinematic equations can be found in [ORIGEN-WPC+99].

Delayed neutrons are emitted by decay of short-lived fission products. The observed delay is due to the decay of the precursor nuclide. The total yield of delayed neutrons from decay of nuclide i is

$$Y_{i,Dn} = \frac{\lambda_{i,Dn}}{\lambda_i} \tag{5.1.48}$$

where $\frac{\lambda_{i,Dn}}{\lambda_i}$ is the fraction of decays which emit delayed neutrons. The delayed neutrons emitted per decay of nuclide i at energy E is given by

$$w_{i,Dn}\,(E) = Y_{i,Dn} X_{i,Dn}\,(E) \tag{5.1.49}$$

where the spectrum $X_{i,Dn}\,(E)$ is fundamental library data. Delayed neutrons are not important in typical spent fuel applications due to the very short half-lives of the parent nuclides, dropping off significantly after ~10 seconds, but they may be of value in specialized applications where calculating time-dependent delayed neutron source spectra is important.

Alpha Sources

An α slowing down calculation is performed as part of the (α, n) neutron calculation. However, the alpha source (i.e. without considering slowing down in the media) is also available, simply as the sum of delta functions at the discrete initial alpha particle energies $w_{i,\alpha}(E) = \sum_{\ell \in i} Y^{\ell}_{i,\alpha} \delta\left(E - E^{\ell}_{i,\alpha}\right)$ with yields $Y^{\ell}_{i,\alpha}$, as required by Eq. (5.1.41).

Beta Sources

The beta source (i.e. without considering slowing down in the media) is available as the sum of the continuous emission spectra for each β^- decay in nuclide i. The total yield of beta particles from decay of nuclide i is

$$Y_{i,\beta} = \frac{\lambda_{i,\beta}}{\lambda_i} \tag{5.1.50}$$

where $\frac{\lambda_{i,\beta}}{\lambda_i}$ is the fraction of decays which emit betas. The betas emitted by nuclide i at energy E is given by

$$w_{i,\beta}(E) = Y_{i,\beta} X_{i,\beta}(E) \tag{5.1.51}$$

where the spectrum $X_{i,\beta}(E)$ is fundamental data, independent of the media. The spectrum includes betas from allowed transitions and first, second, and third forbidden transitions.

Photon Sources

The total yield of photons from decay of nuclide i is

$$Y_{i,\gamma} = \frac{\lambda_{i,\gamma}}{\lambda_i} \tag{5.1.52}$$

where $\frac{\lambda_{i,\gamma}}{\lambda_i}$ is the fraction of decays which emit photons. The photons emitted by nuclide i at energy E is given by

$$w_{i,\gamma}(E) = Y_{i,\gamma} X_{i,\gamma}(E) \tag{5.1.53}$$

where the spectrum $X_{i,\gamma}(E)$ is fundamental data and includes both line data from x-rays, gamma-rays and continuum data from Bremsstrahlung, spontaneous fission gamma rays, and gamma rays accompanying (α, n) reactions. The Bremsstrahlung component of the photon source has been tabulated for various media and no on-the-fly slowing down calculation is performed.

Library Interpolation

Accurate solution of fuel depletion with Eq. (5.1.1) requires coupling to self-shielding and neutron transport to accurately capture the time-dependent change in space and energy flux distribution and 1-group cross sections with isotopic change. This is in generally a fairly computationally intensive problem compared to stand-alone depletion. In typical assembly design and analysis, the same basic assembly problem must be solved repeatedly with variations in power history, different periods of decay/burnup, different moderator density, etc. A question naturally arises: could the isotopics from numerous well-constructed cases be saved and interpolated to the actual system? Interpolating the isotopics themselves is fraught with difficulty. For example, consider two cases with the same burnup but different periods of decay between cycles. A better approach—the ORIGEN Automated Rapid Processing (ARP)—was developed with the key realization that one can reconstruct very accurate isotopics from stand-alone depletion calculations by interpolating *transition matrices* rather than *isotopics*.

The accuracy of the interpolation methodology compared to the coupled transport/depletion solution (e.g., with TRITON) depends on the suitability of the interpolation parameters and the deviation of the desired system from the systems used to create the library. For example, for thermal systems with uranium-based fuels, it was found that enrichment, water density, and burnup were the dominant independent variables and thus were best suited for interpolation. An example of the variation of removal cross sections for key actinides is shown in Fig. 5.1.2 for a Westinghouse 17 × 17 pressurized water reactor (PWR) assembly type with 5% initial enrichment in ^{235}U. Each cross section has been divided by its initial value at zero burnup to show the variation more clearly. ^{240}Pu has been observed to have the most variation with spectral changes, with ~60% reduction in cross section from beginning to end of life. The variations in ^{240}Pu with respect to enrichment and moderator density are shown in Fig. 5.1.3, Fig. 5.1.4, Fig. 5.1.5, and Fig. 5.1.6.

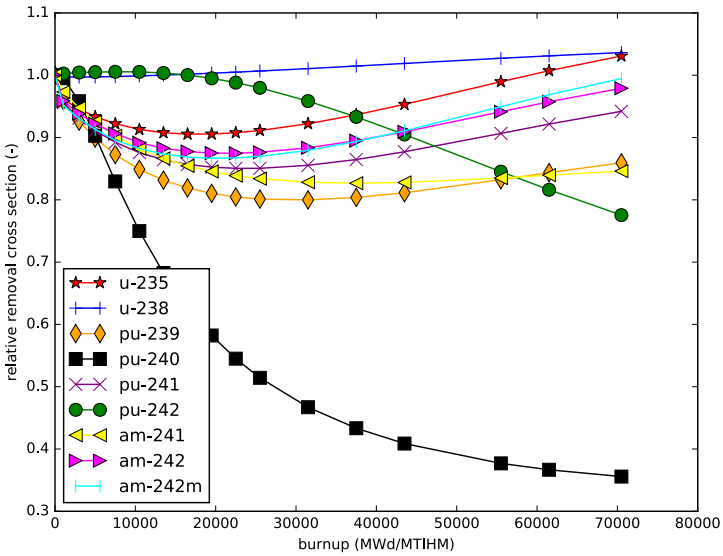


Fig. 5.1.2: Relative removal cross section as a function of burnup for key actinides (Westinghouse 17 × 17 assembly with 5\% enrichment).

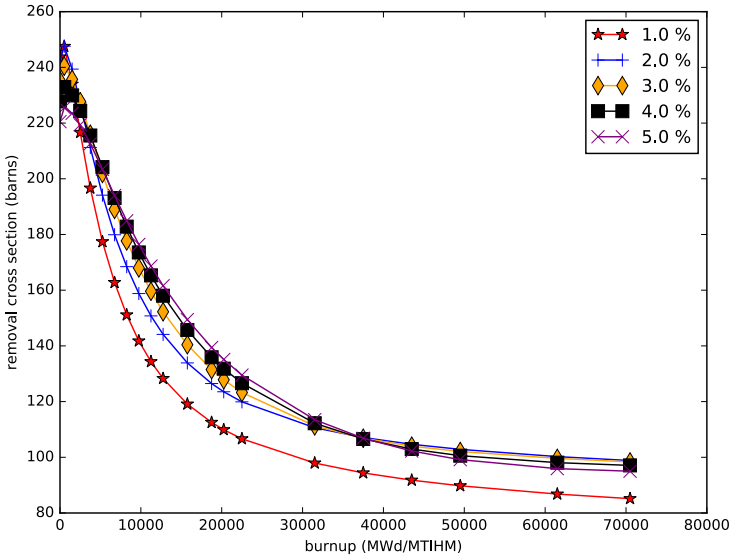


Fig. 5.1.3: ^{240}Pu-240 removal cross section as a function of burnup for various enrichments (GE 10 × 10 assembly).

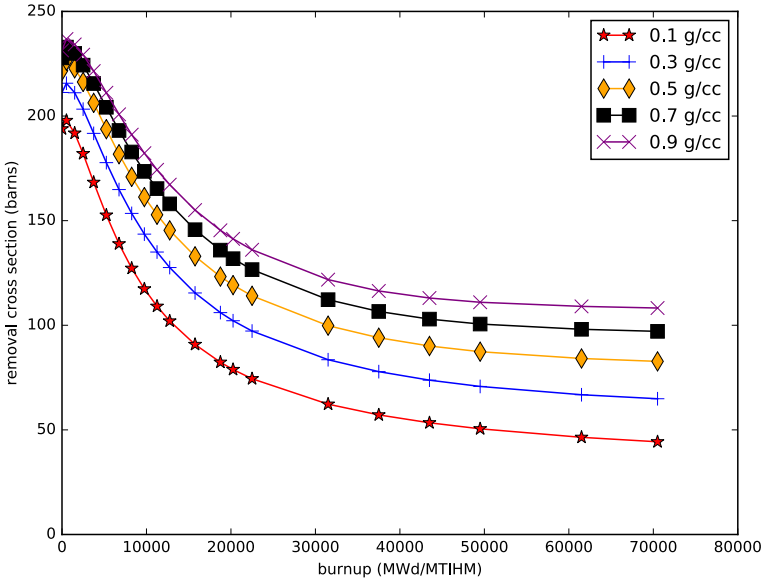


Fig. 5.1.4: ^{240}Pu removal cross section as a function of burnup for various moderator densities (GE 10 × 10 assembly).

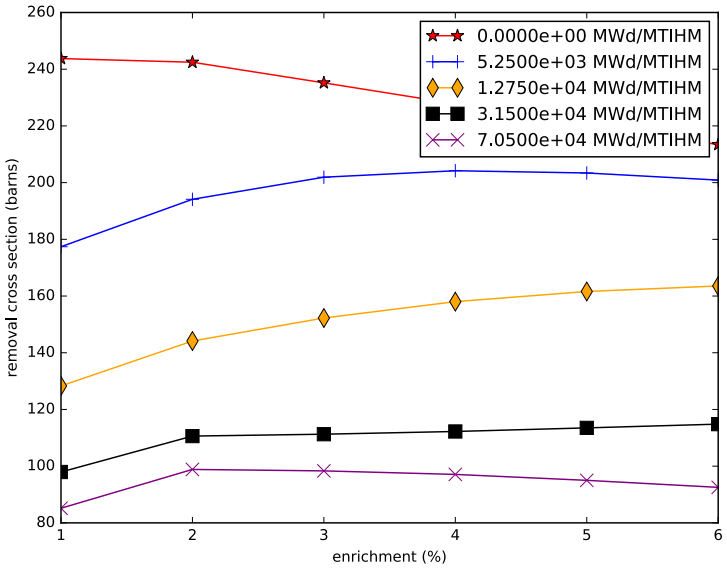


Fig. 5.1.5: ^{240}Pu removal cross section as a function of initial enrichment for various burnups (GE 10 × 10 assembly).

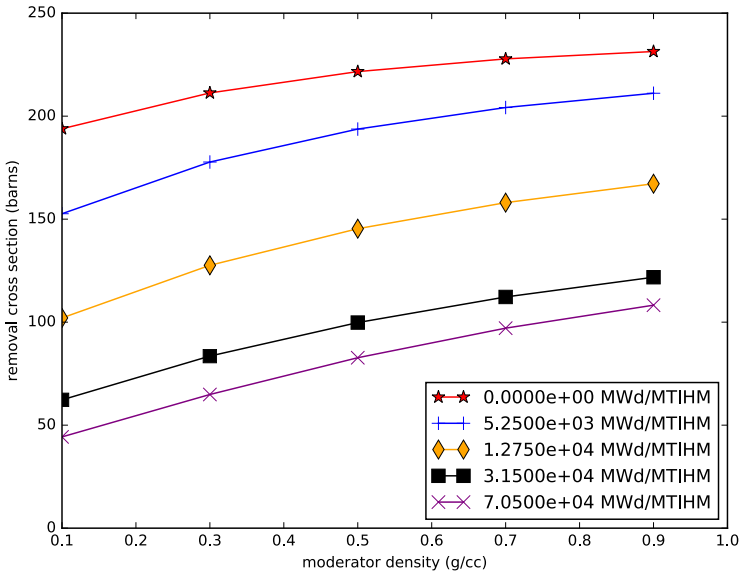


Fig. 5.1.6: ^{240}Pu removal cross section as a function of moderator density for various burnups (GE 10 × 10 assembly).

Currently there are two interpolation methods: a Lagrangian based on low-order polynomials and a cubic spline with an optional monotonicity fix-up.

Lagrangian Interpolation

Lagrangian interpolation [ORIGEN-REFed68] seeks the unique *n-1* order polynomial that will pass through *n*-points of the function and then interpolating to the desired point by evaluating the polynomial,

$$y(x) = \prod_{i=1}^{n} (x - x_i) \sum_{k=1}^{n} \frac{y_k}{(x - x_k) \prod_{\substack{i=1 \\ i \neq k}}^{n} (x_k - x_i)} \tag{5.1.54}$$

where x_i and y_i are the known x- and y-values in the neighborhood of the desired x-value x, with n the number of data points/order of Lagrangian interpolation. Note that Lagrangian interpolation is by definition *local*, involving only points in the neighborhood of the desired value. Global alternatives such as Hermite cubic splines use the entire data set to construct the interpolants. Common interpolation methods based on polynomials can have difficulty with data that vary quickly and have uneven x-spacing, as is expected with transition data. Polynomials tend to produce unphysical oscillations in these cases. In cases with very small y-values (~10^{-10}), oscillations of the interpolant can produce negative interpolated values.

Cubic Spline Interpolation

Cubic spline interpolation has been observed to produce fewer, lower frequency oscillations. Oscillations can be effectively eliminated by enforcing monotonicity on the interpolation: that is, additional max maxima or minima are not introduced by the interpolant between known values of the function. Monotonic cubic splines [ORIGEN-WA02] have shown particularly stable behavior and have been implemented as an interpolation option.

5.1.4 ORIGEN FAMILY OF MODULES

This section describes how to perform the calculations and evaluations described in Sect. 5.1.3 using the ORIGEN family of modules in SCALE. These modules are summarized briefly below.

1. The ARP module creates a new ORIGEN library by interpolating a set of existing libraries. ORIGEN libraries organized for this purpose are called ORIGEN Reactor libraries. Those distributed with SCALE are described in Sect. 5.3.
2. The ORIGEN module is used to solve depletion, decay, activation, and feed problems described by Eq. (5.1.1), as well as the decay emission calculations described by Eq. (5.1.41). For spent fuel calculations using the ARP interpolation methodology, it may be more convenient to use ORIGAMI, as described in the ORIGAMI chapter.
3. The OPUS module is used to perform post processing and analysis on ORIGEN results contained in ORIGEN concentrations files, including sorting, ranking, and unit conversion.

Two types of files are an integral part of the ORIGEN family of modules: the library file and the concentrations files.

- The library file is a binary file, usually either with the complete filename "ft33f001" or with extension ".f33," and it contains a collection of transition matrices **A**, usually corresponding to different burnups. It is typically called an "ORIGEN library," "ft33," or "f33" file.
- The concentrations file is also a binary file, usually either with the complete filename "ft71f001" or with extension ".f71." The f71 is a solution archive containing isotopics vectors $\vec{N}$ corresponding to different materials or different points in time.

Note: A fundamental cability to create ORIGEN libraries within the ORIGEN sequence has been added to SCALE 6.3, effectively deprecating the stand-alone COUPLE module.

5.1.5 ORIGEN MODULE

The ORIGEN module drives depletion, decay, and activation calculations as described in Sect. 5.1.3.1, including the conversion of generated powers to fluxes described in Sect. 5.1.3.1.8, as well as alpha, beta, gamma, and neutron source calculations described in Sect. 5.1.3.1.9.

5.1.5.1 Key Features

This section briefly highlights some key features in ORIGEN and describes how they are used.

Nuclide Specification and ORIGEN Sub-libraries

The nuclide identifiers in ORIGEN are more flexible than those in other modules of SCALE and even within the ORIGEN family. Table 5.1.1 shows the possible ways to specify nuclides (and elements).

Table 5.1.1: Nuclide / element specification in ORIGEN

Identifier Form	**Comments**	**Examples**	
		nuclide	*input id*
IZZZAAA I - *isomeric state* ZZZ - *atomic nummber* AAA - *mass number*	Standard numeric identifier with one optional digit of isomeric state, three digits of atomic number, three digit of mass number; elements have mass number of 000.	^{235}U	92235
		^{235m}U	1092235
		^{135}Xe	54135
		^{1}H	1001
		^{10}B	5010
		Fe	54000
EAm E - *element symbol* A - *mass number* m - *metastable indicator*	Standard symbolic identifier with element symbol followed by mass number, followed by optional metastable indicator; can include a dash between E and A (E-Am); case insensitive.	^{235}U	u235
		^{235m}U	u235m
		^{135}Xe	xe135
		^{1}H	h1
		^{10}B	b10
		Fe	fe

One important aspect ORIGEN users must be aware of is that the ORIGEN library (f33) being used dictates the set of nuclides available in a calculation and that there may be more than one *version* of a nuclide in a library. The duplicates arise in large part from the need to analyze fission products separately. For example, a gadolinia-doped uranium oxide fuel with burnup will have some ^{155}Gd from the initial gadolinia loading and some ^{155}Gd generated as a fission product. Although these fuels physically behave the same way, it is sometimes important to be able to analyze them separately. These groups, versions, or categories are referred to as sublibraries because in an ORIGEN library, they appear almost like three separate, smaller ORIGEN libraries. The three libraries are for

1. naturally occurring, light nuclides, sometimes called “light elements” or “activation products,”
2. actinides and their reaction and decay products, and
3. fission products.

Called “sublibs” for short, they are identified by a number or 2-character specifier:

1. light nuclides with “LT” or 1,
2. actinides with “AC” or 2, and
3. fission products with “FP” or 3.

The production of fission products from actinides (2/AC3/FP) is the only type of transition in a typical ORIGEN library that spans sublibs. The sublib is optional in a nuclide specification and is indicated in parentheses after the identifier-IZZZAAA(S), EAm(S). If the sublib for a nuclide/element is not provided, it is guessed in the following manner:

1. If the nuclide is in fact an element, then it is placed in sublib=1/LT.
2. If the atomic number Ztextlt 26, an *attempt* is made to place it in sublib=1/LT.
3. Otherwise (Ztextgeq 26 or attempt fails), sublibs are searched in reverse order, from 3/FP, 2/AC, and then 1/LT.

The third rule, which is to search sublibs in reverse order, correctly handles spent reactor fuel, a common and important scenario. The other two conditions can be interpreted as exceptions. The first exception correctly handles activation scenarios where it is most convenient to specify the initial elemental constituents. The second exception handles light nuclides that could not be real fission products, as fission products have Ztextgeq 26 by definition. The byproducts ^{1}H, ^{2}H, ^{3}H, ^{3}He, and ^{4}He actually exist in all sublibs, but FP and AC byproducts have a reduced set of transitions compared to the LT version, which has full decay and activation chains. Thus when a user specifies one of the byproduct nuclides as input, it is best to associate it to the LT version.

Physical Units in Calculations

A variety of units can be used in the input and specified for the output of an ORIGEN calculation. The input allows for initial concentrations in

1. grams,
2. moles (or gram-atoms),
3. number density in atoms/barn-cm, and
4. curies.

Time may be expressed in seconds, minutes, hours, days, years, or a user-defined unit. Irradiation may be expressed in terms of neutron flux (n/cm^2-s) or power (W). The allowed units for output include those for input, as well as the following decay quantities:

1. total decay heat power (W),
2. gamma decay heat power (W),
3. airborne toxicity (m^3) required to dilute activities to the Radiation Concentration Guide (RCG) limit for air,
4. ingestion toxicity (m^3) required to dilute activities to the RCG limit for water, and
5. alpha, beta, neutron, photon sources (particles/s or MeV/s).

Table 5.1.2 summarizes the available units in ORIGEN. During irradiation cases, the following can also be returned:

1. absorption rates (absorptions/s),
2. fission rates (fissions/s), and
3. infinite neutron multiplication constant, k_∞.

Table 5.1.2: Availalble physical units in ORIGEN

Unit name	Description	Input	Output	
			(irrad.)	(decay)
GRAMS	Mass in grams	*	*	*
MOLES or GRAM-ATOMS	Number in moles (or legcy equivalent of gram-atoms)	*	*	*
ATOMS-PER-BARN-CM	Density in atoms/barn-cm (10^{-24} cm/barn $\times$ density in atoms/ cm^3); requires volume input	*	*	*
CURIES	Activity in curies	*	*	*
BECQUERELS	Activity in becquerels	*	*	*
ATOMS_PPM	Atom fractions x 10^6		*	*
WEIGHT_PPM	Weight fractions x 10^6		*	*
WATTS	Total decay heat in watts			*
G-WATTS	Total decay heat from photons in watts			*
M3_AIR	Radiotoxicity m^3 for for inhalation			*
M3_WATER	Radiotoxicity in m^3 for ingestion			*

Saving Results

ORIGEN can save any results (isotopics and source spectra) in a special ORIGEN binary concentrations file (f71). The file is a simple sequence of solutions, and new results are simply appended on to the end of an existing file. Note that no matter how initial isotopics are entered or what units are asked for in the output file, the ORIGEN f71 contains **moles** (gram-atoms) of each isotope and an optional **volume** to permit unit conversions to number density (atoms/barn-cm). Isotopics for an ORIGEN calculation can be initialized from any position on this file in an ORIGEN calculation. The f71 can also be read by OPUS to perform various post-processing tasks.

5.1.5.2 Input Description

ORIGEN uses the SCALE Object Notation (SON) language for its input, although it can also read FIDO-based input for backwards compatibility with SCALE 6.1 [ORIGEN-ORN11]. The basic structure of an ORIGEN input is shown in Example 5.1.1.

Example 5.1.1: ORIGEN input file overview

```
'SCALE comment
=origen

% ORIGEN comment %

bounds{ ... }
solver{ ... }
options{ ... }

case(A){
    time=[31 365] % days
    ...
}

case(B){
   ...
}
```

(continues on next page)

(continued from previous page)

```
...
% more cases?

end
```

The ORIGEN input is hierarchical, containing four levels, where level 0 is the "root" level, allowed between "=origen" and "end." The complete set of keywords is shown in Table 5.1.3, with arrays denoted with "=[]" and blocks with "{}". Referring to the overview in Example 5.1.1, at the root level, there is a "solver" block for changing solver options, a "bounds" block for entering the energy boundaries for various particle emissions, and an "options" block for altering the miscellaneous global options. These blocks may only appear once. The remainder of the input is a sequence of "case" blocks (in the above examples there are two cases with identifiers "A" and "B"), which each case is executed in order, with each case possibly depending on one or more of the previous cases.

Table 5.1.3: List of all available ORIGEN input commands

Level 0	**Level 1**	**Level 2**	**Level 3**
case{}	title		
	time{}	start t=[] units custom_name custom_length	
	lib{}	file pos	
	flux=[]		
	power=[]		
	print{}	cutoff_step absfrac_step absfrac_sublib rel_cutoff cutoffs fisrate kinf	
		nuc{} ele{}	sublibs=[] total units=[]
		neutron{}	summary spectra detailed
		gamma{}	summary spectra principal_step unbinned_warning principal_cutoff
		alpha{}	summary spectra

continues on next page

Table 5.1.3 – continued from previous page

Level 0	Level 1	Level 2	Level 3
		beta{}	summary spectra principal_step principal_cutoff
	mat{}	iso=[] feed=[] units previous volume blend=[]	
		load{}	file pos
	save{}	steps=[] file time_offset time_units	
	neutron{}	alphan_medium alphan_bins alphan_cutoff alphan_step	
	gamma{}	sublib adjust_for_missing conserve_line_energy split_near_boundary continuum immediate brem_medium spont	
	alpha{}		
	beta{}	sublib	
build_lib{}	nuclide{}	type=NAMED_SET	list
	decay{}	coeff_update[] allow_zero	
		type=ENDF_DECAY	resource
		type=ORIGEN_LIBRARY	file
	neutron{}	coeff_update[] allow_zero reaction_resource fp_yield_resource	
		type= ENDF_ENERGY_ DEPENDENT	spectrum{} xs_update{}

continues on next page

Table 5.1.3 – continued from previous page

Level 0	Level 1	Level 2	Level 3
bounds{}	alpha=[] beta=[] gamma=[] neutron=[]		
solver{}	type		
	opt{}	terms maxp abstol reltol calc_type order substeps	
options{}	print_xs digits fixed_fission_energy		

The percent sign (%) is the comment character *inside the ORIGEN sequence,* between "=origen" and "end." The % is a very flexible comment that may be placed almost anywhere in the input and continues until the end of the line. Outside the ORIGEN sequence, the SCALE comment character of a single quote ‘ at the beginning of a line must be used. Arrays in SON begin with "[" and end with "]" and support the following special shortcuts inherited from FIDO. Note that the interpolation shortcuts (I and L) *insert* values between two specified values so that there will be N+2 values in the final expanded array section.

Table 5.1.4: Array entry shortcuts

Shortcut	**Format**	**Examples** *shortcutexpansion*
Repeat (**R**)	*N***R***X*	3r1e141e14 1e14 1e14 6r3 3 3 3 3 3 3
Linear interpolation (**I**)	*N***I** *X Y*	3i 5 1 5 4 3 2 1 9i 0.0 1.0 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
Log interpolation (**L**)	*N***L** *X Y*	3l 1 5 5l 1e-9 1e-3 1e-9 1e-8 1e-7 1e-6 1e-5 1e-4 1e-3

As an alternative to manually creating an ORIGEN input file via a text editor, the user may use the SCALE graphical user interface (GUI) Fulcrum to create ORIGEN input files. Advantages to using Fulcrum include syntax highlighting, autocomplete, immediate feedback when input is incorrect, and one-click running of calculations.

Calculation Case (case)

A single ORIGEN sequence may contain an unlimited number of case blocks. Each case block is processed in order and can represent either an independent calculation or continuation of a previous case. The complete contents of a single case block are shown in Table 5.1.1.

Example 5.1.2: ORIGEN "case" overview

```
case(ID){
   title="my title"

    lib{ ... }
    mat{ ... }
    time{ ... }
    flux{ ... } % or power{ ... }

    print{ ... }
    save{ ... }

    alpha{ ... }
    beta{ ... }
    gamma{ ... }
    neutron{ ... }
}
```

The most important three components are the lib, mat, and time/power/flux inputs:

1. an ORIGEN library and the transition matrix data set on it to use (lib),
2. initial amounts of nuclides (mat), and
3. a power or flux history (time/power or time/flux).

The case identifier and case title string (shown as ID and title="my title" in Example 5.1.2) are echoed in the output file and can be a convenient way to differential cases. Both are optional, with the ID defaulting to the case index, with "1" for the first case, "2" for the second, etc. The "print" and "save" blocks represent two ways to analyze the output from a calculation. The "print" block prints tables directly to the output file, and the "save" block saves the solution in a special ORIGEN binary concentration file (f71), e.g., for later post-processing. Finally, the "alpha," "beta," "gamma," and "neutron" blocks control the emission source calculations for alpha, beta, gamma, and neutron particles, respectively. The remaining subsections will describe the input for each of these blocks.

Transition Matrix Specification (lib)

The transition matrix to use in a case is controlled by the "lib" shown in Example 5.1.3.

Example 5.1.3: ORIGEN "lib" overview

```
lib{
    file="origen.f33"  % ORIGEN library filename
    pos=1              % data set position on library
}
```

A "lib" **must** be present in the first case with a defined ORIGEN library file. The default position is "pos=1". The "lib" may be omitted in subsequent cases, and if so, the previous case's "lib" is used. The position refers to the set of transition coefficients (transition matrix **A**) to load. To load another position on the same library file, the "lib" block with "pos=X" can be used to load position X. When ARP generates an ORIGEN

library, it will contain a set of transition coefficients for each requested burnup. When COUPLE generates an ORIGEN library, it will contain a single position. **For decay calculations, file="end7dec" can be used to load a decay-only library.**

Material Specification (mat)

The initial isotopics for a case a controlled by the "mat" shown in Example 5.1.4. Note that the material specification has a few different variants, with only one allowed to specify the material in a given case.

Example 5.1.4: ORIGEN "mat" block overview

```
% from iso
mat{
    iso=[ u235=1.0 u238=9.0 ] %id(sublib)=amount
    units=GRAMS               %units in iso array
}

% from iso with number density input
mat{
    iso=[ u235=1e-2 u238=1e-1 ] %id(sublib)=amount
    units=ATOMS-PER-BARN-CM     %units in iso array
    volume=200                  %cm^3
}

% from position on f71 file
mat{
    load{ file="origen.f71" pos=11 }
}

% from previous case (previous=LAST is default)
mat{
    previous=4 %step index from previous case
}
```

In the first variant in Example 5.1.4, the isotopic distribution "iso" is used with "units." The "iso" array contains a sequence of "id=amount" pairs, where "id" is a nuclide identifier in the format described in Sect. 5.1.5.1.1, and the units of the amount are given by the "units" keyword, one of the unit names listed in the third column of Table 5.1.2. Default units are MOLES.

In the second variant, the number density (ATOMS-PER-BARN-CM) is specified which requires an additional specification of the "volume" in cm^3. Internally, the number density will be converted to MOLES using that volume. For any type of units specified internally for calculations, isotopics are always converted to MOLES and then reconverted to the output units required.

In the third variant, the isotopics are loaded from a specific position on the f71 file. Note that the position index starts at one (not zero) and because the f71 is always appended to, it may contain multiple materials, cases, timelines, etc. In the fourth and final variant, the isotopics are loaded from *end* of step 4 from the previous case ("previous=4"). The index zero (e.g., "previous=0") corresponds to the initial isotopics of the previous case. The keyword "LAST" may be used to load the isotopics from the end of the last step, "previous=LAST". This is the default behavior, used when a "mat" block is not present.

There are two additional special material specifications shown in Example 5.1.5: (1) with a feed rate term, $\vec{S}(t)$ in Eq. (5.1.4), or (2) the blend array. The feed specified is in the units specified *per second* and constant for the entire case. It is possible to perform a calculation with feed but with zero initial isotopics by specifying "iso=0". Feed can be negative, however, the calculation becomes undefined and will abort when the number of atoms of any nuclide becomes negative.

The blend array allows a fraction of each result from the previous cases to be loaded. The identifier is the case name, or the *index* of the case if a case name is not provided and the fraction is the atom fraction. The step index for the isotopics can be specified in parentheses. For example, B(2)=0.9 indicates that 90% of the case(B) isotopics should be taken at the end of step 2. The default step index is the final step for the case. **Only one blend is allowed in an ORIGEN input (between "=origen" and "end").** Multiple blends currently requires saving isotopics to an f71 file and reloading in a subsequent calculation.

Example 5.1.5: ORIGEN "feed" and "blend" arrays

```
% with feed array
mat{
    % units for iso and feed
    units=GRAMS

    % material is natural sodium
    iso=[na=1.0e6]
    % with feed array, set initial isotopics of zero
    %iso=0

    % continuous feed of U-235 at 1 kg/day
    % converted to grams/second
    feed=[u235=0.01157]
}

% with blend array (only one allowed in an input)
case(A){ ... }
case(B){ ... }
case{
    ...
    mat{
        % case ID(step index)=fraction of atoms
        blend[ A=0.1 B(2)=0.9 ]
    }
}
```

Operating History (power, flux, time)

The operating history is specified using "time," "power," and "flux," with examples shown in Example 5.1.6. For decay cases, only the "time" array in units of days is required. For irradiation cases, either "power" or "flux" may be provided. When flux is used, it is the step-wise flux Φ_n $\left(\frac{n}{cm^2 s}\right)$ appearing directly in the depletion equations of Eq. (5.1.4). When power is used, it is the total step-average power– P_n (MW) –converted to step-wise average flux using Eq. (5.1.40). With irradiation cases using flux or power, the same number of entries must be specified on the time and flux/power array. The start time corresponding to the initial conditions is not included in the array of time values. Additionally, the time specification allows time units (including a custom unit) and a start time in which the block form of "time" must be used "time{...}."

Example 5.1.6: ORIGEN operating history blocks ("time," "flux," and "power").

```
% simple decay case (two steps 0 unicode::U+2192 1 and 31 unicode::U+2192 65 days)
time = [ 31 365 ]

% flux irradiation (decay if flux=0)
time = [ 31   365  396 ]
flux = [ 2e14 1e14 0 ]

% power irradiation (decay if power=0)
time = [ 31 365 396 ]
power= [ 50 45  0 ] %50 MW, 45 MW, then decay
```

(continues on next page)

(continued from previous page)

```
% changing units using time block
time{
    t = [ 5 15 300 ]
    units = HOURS
    % available units:
    % SECONDS, MINUTES, HOURS, DAYS, YEARS, CUSTOM
}

% custom units
time{
    t = [ 1 2 3 ]
    units = CUSTOM
    custom_name = "MONTH"
    custom_length = 2678400 %seconds per "MONTH"
}

% 10-step detailed power history
time=[   5  10 20 100 300 400 405 500 800 1000]
power=[ 20  41 43  42  37  33  16 14.5 28.5 26]
```

To illustrate some aspects of specifying a power history, refer to Fig. 5.1.7, where the black line ("actual power") shows a piecewise linear power history that is translated to a possible step-wise constant power history shown by the red line ("step-wise constant power"), with input shown in Example 5.1.6 labeled "10-step detailed power history". The secondary (right) y-axis shows the step-wise flux, calculated from the step-wise power via the predictor-corrector approach of Eq. (5.1.40). The dependence of the power-to-flux conversion on the actual material composition is shown in the comparison of flux results for an initial composition with 6% fissile ^{235}U (blue dotted line) versus 2% fissile ^{235}U (purple dashed line). The flux at the beginning of the irradiation is a factor of 3 higher with the 2% fissile case, due to approximately a factor of 3 lower fissile content. With time, fissile plutonium build-up closes the gap to a factor of 1.5.

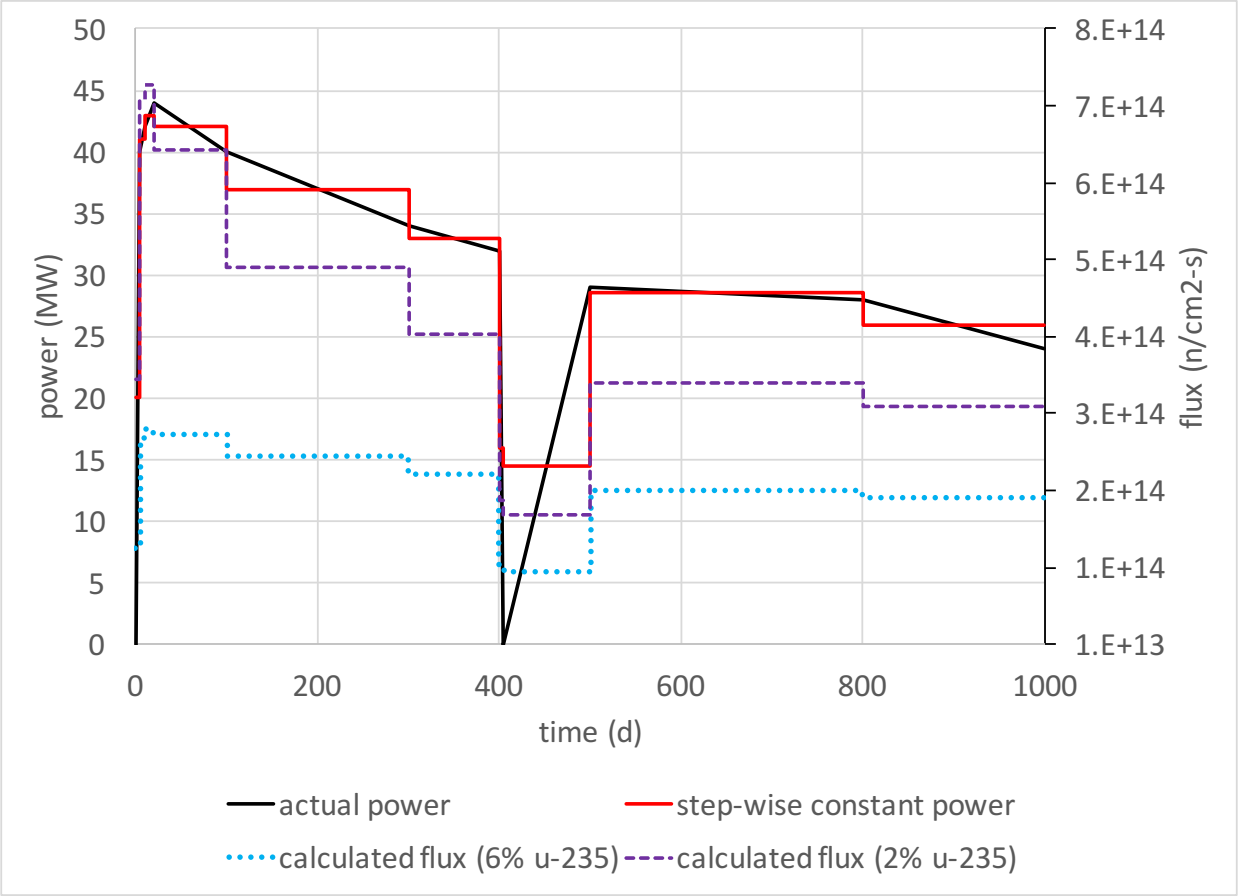


Fig. 5.1.7: Example of ORIGEN operating history and power-to-flux conversion.

Example 5.1.7: ORIGEN "start" time usage.

```
% first case
case{
    mat{ ... }
    time=[ 1 10 25 50]
    flux=[ 4r1e14 ]
}
% continuation case (time zero is 50 days)
case{
    time{
        units=YEARS
        %without start, times must continue > 50 days
        %t=[50/365.+0.1 50/365.+0.3 ... ]
        %with start=0, times given assume start at zero
        start=0
        t=[0.1 0.3 0.9 2.7]
    }
}
```

By default, subsequent cases that continue operations on a material, continue the timeline of that material. Using "start=0" is convenient when switching time unitsfrom irradiation in days to decay time in years, for example. Otherwise, the final time must be converted to years.

Printing Options (print)

The "print" command is one of the most complex inputs, with options to set printing cutoffs and control the concentrations returned, broken down by nuclides and elements and emission sources for gamma, neutron, alpha, and beta particles. Additionally, there are options to print fission rates, absorption rates, and the ratio of fission rate to absorption rate. Each case is allowed a print block.

Inventories by nuclide and element

The options for printing nuclides and elements are shown in Example 5.1.8. The print block allows the "nuc" block and "ele" block for printing nuclide and element results, respectively.

Example 5.1.8: ORIGEN nuclide and element "print" blocks

```
% print each nuclide (total across all sublibs) in grams
print{
    nuc{ total=yes units=GRAMS }
}

% print each element (total across all sublibs)
% in moles, grams, and curies with cutoffs of 1%
print{
    ele{ total=yes units=[MOLES GRAMS CURIES] }
    cutoffs[ ALL=1.0 ]
}

% print decay heat and mass (by element)
% of fission products and actinides only
print{
    ele{ sublibs=[AC FP] units=[GRAMS WATTS] }
}

% change cutoff to absolute curies by element,
% in step of interest (7), but print GRAMS
print{
    cutoff_step = 7 % default -1 for average
    rel_cutoff = no % default is yes for cutoff in percent
    % only print above 1e-3 curies
    cutoffs[ CURIES=1e-3 ]   % default is 1e-6 percent
    nuc{
     total=yes
     units=GRAMS
    }
}
```

Inside the "nuc" or "ele" blocks, there are three possible entries:

- a "sublibs" array (a list from LT, AC, FP, ALL),
- a "total" (yes or no), and
- a "units" array (see column 1 of Table 5.1.2 for possible units).

The "total" is whether to sum over all "sublibs," i.e., if Gd-155 occurs in both LT and FP sublibs, then the total will be the sum of the two. It is possible to have "sublibs=[LT AC FP] total=yes," which results in four output tables, one for each of the sublibs and one for the total. The specification of "sublibs=ALL" is the same as "sublibs=[LT AC FP]."

Three parameters set the cutoff for printing a nuclide or element:

- "cutoff_step" sets the index on which to base the cutoff (default -1 means use an average over all steps),

- "rel_cutoff" determines whether to treat the cutoff as a percent of the total (default/yes) or an absolute amount (no), and
- the "cutoffs" array allows one to specify the cutoff for each unit in Table 5.1.2 as a sequence of "unit=cutoff" pairs.

Radiological Emissions (alpha, beta, gamma, neutron)

The emission printing options are controlled by the "alpha," "beta," "gamma," and "neutron" emission blocks inside a "print" block (examples shown in Example 5.1.9).

Example 5.1.9: ORIGEN emission "print" blocks.

```
print{
    % default neutron options
    neutron{
     summary=yes
     spectra=no
     detailed=no
    }

    % default gamma options
    gamma{
     summary=yes
     spectra=no
     principal_step=NONE  %step index to calculate
                          %(NONE to suppress)
     principal_cutoff=2   %principal emitter cutoff
                          %in percent
     unbinned_warning=no  %print warning
                          %when line not binned
    }

    % default alpha options
    alpha{
     summary=yes
     spectra=no
    }

    % default beta options
    beta{
     summary=yes
     spectra=no
     principal_step=NONE  %step index to calculate
                          %principal (NONE to suppress)
     principal_cutoff=2   %principal emitter cutoff
                          %in percent
    }
} %end print
```

Neutron print options

The "neutron" print options are

- "summary" (yes/no) controls the printing of a source strength summary,
- "spectra" (yes/no) controls the printing of the spectra (energy group-wise), and
- "detailed" (yes/no) controls the printing of extra details about the neutron calculation.

The "gamma" print options are

- "summary" (yes/no) controls the printing of a source strength summary and
- "spectra" (yes/no) controls the printing of the spectra (energy group-wise).

Gamma print options

The gamma print allows a special output of the principal emitters in each energy group, controlled by setting the "principal_step" keyword to a specific step index in the case, with the "principal_cutoff" keyword used to set the minimum percent of the total a nuclide must have to be considered a principal emitter. For the gamma print there is a warning that can be enabled with "unbinned_warning=yes" if some gamma lines fall outside the user group structure and thus are not included.

Alpha print options

The "alpha" print options are

- "summary" (yes/no) controls the printing of a source strength summary and
- "spectra" (yes/no) controls the printing of the spectra (energy group-wise).

Beta print options

The "beta" print options are

- "summary" (yes/no) controls the printing of a source strength summary and
- "spectra" (yes/no) controls the printing of the spectra (energy group-wise).

The beta print also allows a special output of the principal emitters by setting the "principal_step" keyword to a specific step index in the case, with the "principal_cutoff" keyword used to set the minimum percent of the total a nuclide must have to be considered a principal emitter.

The special printing options are shown in Example 5.1.10.

Example 5.1.10: ORIGEN special "print" options

```
% defaults special printing options
print{
    absfrac_sublib = ALL %print absorption fractions for
                         %a specific sublib (LT,AC,FP)
                         %or ALL sublibs (DEFAULT)

    absfrac_step = 7  %if absfrac active, step to print
                      % default is last step

    fisrate = NONE    %print fission rates (default NONE)
                      %absolute (ABS) or relative (REL)

    kinf = no         %print fission/absorption (yes/no)
}
```

Saving Results (save)

Saving the results to an ORIGEN binary file (f71) is requested with the "save" block, which specifies both the name of the file and the step indices to save, as shown in Example 5.1.11. The default for the filename is "file=ft71f001" and default for the steps is the special "steps=ALL" which saves all isotopics and spectra as a shortcut to having to specify "steps=[0 1 2 3 . . . LAST]". The step index "0" corresponds to the initial isotopics and the step index "LAST" may be used as a shortcut for the last case index. There is a special rule for copying f71 files from SCALE's temporary/working directory. If the file name "ft71f001" exists in the directory when SCALE finishes, it is copied to the user's "${OUTDIR}" as "${BASENAME}.f71", e.g. if my.inp produces "ft71f001" in the temporary/working directory then it will be copied to the same location as the main output file ("my.out") as "my.f71". Note that f71 files are always appended to by ORIGEN. To save with the defaults, the shortcut "save=yes" is provided. The default is "save=no".

Example 5.1.11: ORIGEN "save" block.

```
case{
    mat{ ... }
    time=[ 1 10 100 1000 ] % 4 steps to 1000 days
    save{
          file="short.f71"   % file name
          steps=[0 2 4]      % save initial (0) and isotopics
                                 % end-of-step 2 (10 days)
                                 % end-of-step 4 (1000 days)
    }

    save{
          file="ft71f001"    % file name (DEFAULT)
          steps=ALL          % save ALL steps (DEFAULT)
    }
    save=yes               % equivalent to the above

    save{
          file="last.f71"    % file name
          steps=[LAST]       % save only last (LAST=4 here)
          time_offset=1000   % write time - time_offset
          time_units=DAYS    % units of time_offset
    }
}
```

In order to change the time values written to the f71 file, use "time_offset=T_0" which will write the current cumulative time minus T_0 to the file. The "time_offset" is convenient, for example, when time *since discharge* is desired instead of the absolute, cumulative time. The "time_units" entry specifies the units of the "time_offset", with the same units available in the "time" block. The default is "time_units=DAYS".

Decay Emission Calculations (alpha, beta, gamma, neutron)

A decay emission calculation is initiated with the appropriate block inside the calculation "case." The group structure for any emission spectra result is determined by the energy bounds provided as described in Sect. 5.1.3.1.9. Each type of emission calculation is activated by the existence of a calculation block named "alpha", "beta", "gamma", or "neutron" for those respective types of calculations. Alternatively, to turn on an emission calculation with defaults, use "alpha=yes", "beta=yes", "gamma=yes", or "neutron=yes" in a "case" block.

Neutron source calculation

The neutron calculation (Sect. 5.1.3.1.10) is activated by the "neutron" calculation block. All neutron calculation options are to control the (α, n) calculation.

Three (α, n) options can be indicated with the "alphan_medium": a UO_2 fuel matrix (alphan_medium=UO2), a borosilicate glass matrix (alphan_medium=BOROSILICATE), and the problem-dependent matrix defined by the user input compositions (alphan_medium=CASE). The numeric options 0, 1, and 2 are also valid for the UO2, BOROSILICATE, and CASE options, respectively. Note that the UO_2 and borosilicate glass matrix options assume that the neutron source nuclides reside in these respective matrices, *regardless of the actual composition of the material in the problem.*

For oxide fuels, a significant neutron source can be produced from ^{17}O (α, n) and ^{18}O (α, n) reactions in the oxygen compounds of the fuel. For this reason, the UO_2 matrix option (enabled by alphan_medium=UO2) is provided with natural isotopic distribution of ^{17}O and ^{18}O. This includes the impact of oxygen isotopes on the neutron source without having to include oxygen in the initial composition.

Another common use case is fuel storage in a borosilicate glass matrix (enabled by `alphan_medium=BOROSILICATE`), listed in Table 5.1.5 from [ORIGEN-DJPB86]. Elements with atomic number less than 17 have (α, n) yields.

Table 5.1.5: Elemental composition used in the borosilicate glass option.

Atomic number	**Element symbol**	**Wt %**	**Atom %**
3	Li	2.18	6.296
5	B	2.11	3.913
8	O	46.4	58.138
9	F	0.061	0.0644
11	Na	7.65	6.671
12	Mg	0.49	0.404
13	Al	2.18	1.620
14	Si	25.4	18.130
17	Cl	0.049	0.0277
20	Ca	1.08	0.540
25	Mn	1.83	0.668
26	Fe	8.61	3.091
28	Ni	0.70	0.239
40	Zr	0.88	0.193
82	Pb	0.049	0.0047
Total		99.669	100.000

In the last most rigorous option, the (α, n) neutron source and spectra are calculated using the source, target, and constituents determined using the material compositions in the problem *at a particular time,* dictated by "alphan_step" in the "neutron" print options. For spent fuel neutron source calculations, there are a large number of potential source, target, and constituent nuclides in the (α, n) calculation and in order to remove low-importance nuclides from the calculation, the "alphan_step" and "alphan_cutoff" parameters are used. Only those nuclides with an α-decay activity exceeding the product of "alphan_cutoff" times the total α activity are included as source nuclides in the (α, n) neutron calculation. Additionally, only those nuclides with a constituent or target atom fraction less than "alphan_cutoff" are included unless the concentration is greater than 1 ppm, in which case it will be retained regardless of the cutoff. The "blend" array is particularly useful for creating a problem-dependent medium composed of fuel and another material.

Example 5.1.12: ORIGEN "neutron" calculation block

```
case{
    ...
    neutron{
          alphan_medium=CASE %0/UO2 for UO2
                             %1/BOROSILICATE
                             %  for Borosilicate glass
                             %2/CASE for case-specific (DEFAULT)
          alphan_bins=200    %use 200 bins for
                             %alpha slowing down calculation
          alphan_cutoff=0.0  %cutoff for alpha,n calculation
          alphan_step=LAST   %step index in this case
    }
    %alternatively, to enable neutron calculation with defaults
    %neutron=yes
}
```

Gamma source calculation

The gamma (photon) calculation described in Sect. 5.1.3.1.13 is activated with the "gamma" block, as shown in Example 5.1.13. The gamma block includes a host of options where the default should be appropriate in most cases.

- The "sublib" option (default "ALL") affects the sublibraries included in the gamma emission calculation. The "immediate" option (default "yes") includes immediate gamma and x-rays.
- The "spont" option (default "yes") includes photons emitted during spontaneous fission and (α, n).
- The "continuum" option (default "yes") enables mapping of continuum data stored artificially as lines to a continuum across the user-defined energy bins.

The following options may need to be modified in some scenarios.

- The "brem_medium" option (default "UO2") includes the photons emitted by beta particles as they slow down in a medium (Bremsstrahlung) in the gamma calculation. A problem-dependent medium is not available for "brem." The only options are "NONE," "H2O" (for Bremsstrahlung in water) and "UO2" (for uranium dioxide).
- The "conserve_line_energy" option (default "no") modifies the intensity of each gamma line within a group to conserve energy according to *Ig = Ia Ea/Eg*, where *Ia* is the original line intensity, *Ea* the original line energy and *Eg* is the group energy (simple midpoint energy). Note that this option modifies intensities and thus results in number of particles not being conserved.
- The "adjust_for_missing" option (default "no") accounts for the scenario where a nuclide has a known gamma decay mode with known recoverable energy release, but the spectral data do not exist in the available emission resource. If "adjust_for_missing=yes," the *entire spectrum* is scaled up to include the missing energy. If the missing energy is more than 1% of the total, a warning is printed.
- The "split_near_boundary" option (default "no") enables a splitting of a line when it appears very close to a bin boundary. If "split=yes" and a photon line is within 3% of an interior energy boundary, then the intensity is split equally between the adjacent groups.

Example 5.1.13: ORIGEN "gamma" calculation block

```
case{
    ...
    gamma{
         sublib=ALL      %LT, FP, or AC - single sublib
                         %ALL - all sub-libraries
         brem_medium=UO2 %assume Bremsstrahlung in UO2
         continuum=yes   %expand continuum data stored as
                         %lines into proper continuua
         immediate=yes   %load lines for immediate gamma/x-rays
         spont=yes       %include photons from
                         %spontaneous fission
                         %and alpha,n reactions

         conserve_line_energy=no %conserve line energy instead
                                 %of line intensity
         adjust_for_missing=no   %adjust photon intensity
                                 %for energy of missing spectra
         split_near_boundary=no  %lines within 3% of a
                                 %group boundary are split
                                 %into two bins
    }
    %alternatively, to enable gamma calculation with defaults
    %gamma=yes
}
```

Alpha and beta source calculation

The alpha calculation (Sect. 5.1.3.1.11) has no options and the beta calculation (Sect. 5.1.3.1.12) only has a single option to choose the nuclide sublibraries included (see Example 5.1.14). Note that the alpha and beta calculations determine *sources* and do not include any slowing down physics for charged particles.

Example 5.1.14: ORIGEN "alpha" and "beta" calculation blocks.

```
case{
    ...
    alpha{
        %no options
    }
    %alternatively, to enable alpha calculation with defaults
    %alpha=yes

    beta{
       sublib=ALL  %LT, FP, or AC - single sublib
                   %ALL - all sub-libraries
    }
    %alternatively, to enable beta calculation with defaults
    %beta=yes
}
```

Processing Options (processing)

The processing options are contained inside the "processing" block (inside a "case") and have two ways to modify the isotopics vector: the "removal" block and the "retained" array (see ::numref::*fig-origen-processing-block*). Both operate on *elements* instead of nuclides.

- Each "removal" block specifies a list of elements and a rate of continuous removal in units *(1/s)*. This removal is an artificial increase of the decay constant, from Eq. (5.1.4), and it can be used to model any continuous system of removal, such as filtration. Note that more than one removal block may be specified.
- The "retained" array specifies a list of elements and the mole/atom fraction to be retained at the start of this case. **Note that all unspecified elements become zero.**

Example 5.1.15: ORIGEN "processing" block

```
case{
    ...
    processing{
        %rate is in 1/s
        removal{ rate=1e-2 ele=[H Xe Ar] }
        removal{ rate=1e-7 ele=[U Pu] }
        %id=frac
        retained=[ u=1.0 pu=0.5 ]
    }
}
```

Bounds Block

The "bounds" block appears outside all cases and is valid for the entire input, dictating the energy bins (i.e., groups) for emission spectra and source calculations (see Example 5.1.16). The array shortcuts-"L" for logarithmically spaced intervals and "I" for linearly-spaced intervals-can be particularly useful for specifying the bounds (see Table 5.1.4). The energy bounds are in units *(eV)* and can be given in increasing or decreasing order with the output convention of decreasing order. Additionally, neutron and gamma energy bounds can be read from a standard SCALE cross section library file by specifying the path to the file instead of the array bounds.

Example 5.1.16: ORIGEN "bounds" block.

```
=origen
bounds{
    neutron=[1e6 1e3 1] %2-group with 1MeV, 1keV, 1eV
    gamma=[100L 1.0e7 1.0e-5] %101 logarithmically spaced bins
    alpha=[1e6 2e7] %high-energy bin between 1 and 20 MeV
    beta=[22I 1 100] %23 linear bins between 1 and 100 eV
    %read neutron bounds from SCALE multi-group library file
    %neutron="scale.rev04.xn252v7.1"
}
case{ ... }
end
```

Solver Block

The solver block controls high-level solver options, the most important of which is the actual solver kernel used, either the default MATREX (Sect. 5.1.3.1.3, Example 5.1.17) or CRAM (Sect. 5.1.3.1.7, Example 5.1.18).

Example 5.1.17: ORIGEN "solver" block for MATREX

```
=origen
solver{
    type=MATREX    %(DEFAULT)
    opt{
        terms=21   %number of expansion terms (DEFAULT)
        maxp=100   %maximum number of short-lived precursors
                   % for a long-lived nuclide (DEFAULT)
        substeps=1 %number of time step divisions (DEFAULT)
    }
}
case{ ... }
end
```

Example 5.1.18: ORIGEN "solver" block for CRAM

```
=origen
solver{
    type=CRAM
    opt{
        order=16   %order of the method (DEFAULT)
        substeps=1 %number time step divisions (DEFAULT)
    }
}
case{ ... }
end
```

The MATREX kernel contains two parameters that are generally sufficient with the default values. The minimum number of "terms" is $n_{terms} = 21$, which overrides the heuristic in Eq. (5.1.20). Experience indicates that high flux levels (e.g., greater than 10^{16} n/cm^2-s) may require more terms. The second parameter, "maxp," is generally sufficient, describing the amount of storage available for the short-lived precursors of a long-lived isotope.

The CRAM kernel has a single parameter that impacts the numerical error: the "order." Both the CRAM and MATREX solver include the "substeps" parameter that adds substeps to each user-defined time step. Testing the same input with a different number of substeps (e.g., "substeps=1," "substeps=2," "substeps=4") can be a simple way to check that the time grid is sufficient.

Options Block

The "options" block contains miscellaneous global options that apply to all cases, as shown in Example 5.1.19. The "print_xs" option enables output of the transition matrix **A** used in each case, the "digits" option enables high-precision output when set to 6, and the "fixed_fission_energy" allows 200 MeV per fission to be used everywhere instead of the nuclide-dependent energy release being used by default.

Example 5.1.19: ORIGEN "options" block

```
=origen
options{
    print_xs=no             %print cross sections
    digits=4                %digits=6 is high-precision
    fixed_fission_energy=no %set to yes to use 200 MeV/fission
}
case{...}
end
```

Library Building Block `build_lib`

This block allows one to create an ORIGEN library within an ORIGEN input. The basic usage is to create a library, "A.f33" in the example below, and then use this library in a calculation.

```
=origen
build_lib("A.f33"){
  ...
}
case{
  lib{ file="A.f33" pos=1 }
  ...
}
end
```

The build_lib block is composed of three main sub-blocks:

- nuclide{ ... }: nuclides to consider,
- decay{ ... }: decay data data to use, and
- neutron{ ... }: incident neutron data to use.

These blocks are designed to be extensible and will gain additional capabilities in future versions. Note that the obiwan command-line utility (see Sect. 5.2) is useful in the view mode to understand the contents of your library file.

Note: Previously, this was the purpose of the COUPLE module, which is deprecated as of SCALE 6.3.

Nuclide Block build_lib/nuclide

The build_lib/nuclide block contains the information on which nuclides to consider in the final library. One can read data from any source for any nuclide, and this will be echoed in the output file; however, only the nuclides defined here will be allowed in the final ORIGEN library. The default and only options for the nuclide block are type=NAME_SET with the nuclide set complete_v6.2, which is defined as the set of 2237 nuclides (including sub-libraries) used in SCALE v6.2.

```
=origen
build_lib("A.f33"){
  nuclide{
    type=NAMED_SET
    list="complete_v6.2"
  }
  ...
}
...
end
```

Note: The entire build_lib/nuclide block will default to the above for SCALE 6.3.

Decay Data Block build_lib/decay

The build_lib/decay block contains the fundamental decay data with which to build the library. With this block, there are currently two options:

- type=ORIGEN_LIBRARY, which loads decay data from an existing ORIGEN library specified with the file= input.
- type=ENDF_DECAY`, which loads decay data from any supported decay resource using the ``resource= input—both ORIGEN card-image formats (legacy) and ENDF-formatted data.

Creating a basic ORIGEN decay library is now trivial.

```
=origen

build_lib("my_decay.f33") {
  nuclide {
    type=NAMED_SET
```

(continues on next page)

(continued from previous page)

```
    list="complete_v6.2"
  }
  decay {
    type=ENDF_DECAY
    resource="${DATA}/origen_data/origen.rev03.decay.data"
  }
}

end
```

Note: All ORIGEN inputs that expect paths (e.g., `resource`) now expand SCALE environment variables such as `${DATA}` and `${INPDIR}`, which can be very convenient as it avoids having to use `=shell` to copy needed files to the working directory.

Updating specific decay coefficients

One final input is available in the `decay` block, `coeff_update`, which allows one to specify specific decay channels and yields to update.

The specific format is

```
coeff_update[
  parent "channel(yield)" value
  ...
]
```

where

- `parent` is the parent nuclide id or name,
- `channel` is the numeric identifier for the decay channel,
- `yield` (optional) is the isomeric state of the daughter (e.g., 0 or 1), and
- `value` is the actual value for the decay constant or yield.

As an example, the following input changes the alpha decay data for ^{238}U, after it loads the nominal data from the decay resource.

```
=origen

build_lib("end7dec.f33") {
  nuclide {
    type=NAMED_SET
    list="complete_v6.2"
  }
  decay {
    type=ENDF_DECAY
    resource="${DATA}/origen_data/origen.rev03.decay.data"
    coeff_update [
        u238  "5"        1e-12   % alpha decay (5) in 1/s
        u238  "5(0)"     0.9000  % alpha decay (5) yield to residual in ground (0)
        u238  "5(1)"     0.1000  % alpha decay (5) yield to residual in 1st meta (1)
    ]
  }
}

end
```

The decay channel identifiers are shown in the following table.

Table 5.1.6: Decay channel identifiers.

Channel Identifier	Decay Mode
1	Gamma
2	Beta Minus
3	Beta Plus
4	Isomeric Transition
5	Alpha
6	Neutron
7	Spontaneous Fission
8	Proton
9	Unknown

Note that a decay channel can be composed of multiple, chained modes. For example, a $2\beta_- + n$ decay channel would have identifier `226`.

Notes on Spontaneous Fission

Spontaneous fission has been historically accounted for in ORIGEN in an unexpected way. The spontaneous fission mode is included in the overall decay constant but the individual fission product yields are not modeled by default. With the `coeff_update` block it is possible to add spontaneous fission, but one must be careful: no matter what the input is, the yields will be renormalized to sum to 2.0, as in a physically meaningful fissioning system. For this reason, it may be beneficial to introduce a fake target for most spontaneous fissions if one's interest is just in modeling a few key fission products.

With spontaneous fission, the "yield" identifier is the target fission product nuclide identifier in IZZZAAA format.

For example, consider this fake spontaneous fission data which uses ^{12}C as a fake target nuclide.

```
=origen

build_lib("end7dec.f33") {
  nuclide {
    type=NAMED_SET
    list="complete_v6.2"
  }
  decay {
    type=ENDF_DECAY
    resource="${DATA}/origen_data/origen.rev03.decay.data"
    coeff_update [
        cf252 "7(6012)"  1.80000 % spontaneous fission (7) to fake target C-12 because yields must normalize␣
↪to 2.0
        cf252 "7(54135)" 0.1600  % 8% of FPs are Xe135
        cf252 "7(53135)" 0.0400  % 2% of FPs are  I135
    ]
  }
}
```

Neutron Data Block `build_lib/neutron`

The `build_lib/neutron` block contains the fundamental incident neutron data on which to build the library. With this block, there is currently only a single option, `type=ENDF_ENERGY_DEPENDENT`, which allows the following inputs.

- `reaction_resource`, a path to the reaction resource to load, sometimes referred to as a "JEFF" library because the current data originates from JEFF/3.0-A.
- `fp_yield_resource`, a path to the fission product yield resource, which may be an ORIGEN card-image format (legacy) or ENDF-formatted data.

With these resources loaded, one must add a flux spectrum to determine energy-integrated values for all reactions and fission product yields.

For example, the following input creates a complete library (decay and neutron data) using the 56-group reaction resource with a constant flux spectrum, 1.0 for all groups.

```
=origen

build_lib("B.f33") {

  nuclide {
    type=NAMED_SET
    list="complete_v6.2"
  }

  decay {
    type=ENDF_DECAY
    resource="${DATA}/origen_data/origen.rev03.decay.data"
  }

  neutron(1) {
    type=ENDF_ENERGY_DEPENDENT
    reaction_resource="${DATA}/origen.rev01.jeff56g"
    fp_yield_resource="${DATA}/origen_data/origen.rev05.yield.data"

    spectrum {
      type=MULTIGROUP
      flux=[56R 1.0]
    }

  }
}

end
```

The `spectrum` block currently has two types.

- `type=MULTIGROUP` to provide a `flux` in the same group structure as the reaction resource. Future versions will have additional flexibility to provide the flux spectrum in any group structure.
- `type=AMPX_LIBRARY` to provide the spectrum from an AMPX library, which requires both `file` and `mixture` inputs, which specify the AMPX library file and the mixture number, respectively.

The following input is an example of loading a spectrum directly from an AMPX multigroup library.

```
=origen

build_lib("B.f33") {
```

(continues on next page)

(continued from previous page)

```
  nuclide {
    type=NAMED_SET
    list="complete_v6.2"
  }

  decay {
    type=ENDF_DECAY
    resource="${DATA}/origen_data/origen.rev03.decay.data"
  }

  neutron(1) {
    type=ENDF_ENERGY_DEPENDENT
    reaction_resource="${DATA}/origen.rev01.jeff56g"
    fp_yield_resource="${DATA}/origen_data/origen.rev05.yield.data"

    spectrum {
      type=AMPX_LIBRARY
      file="sysin.microWorkLib_0.f44"
      mixture=1
    }
  }
}

end
```

Note that this assumes that `sysin.microWorkLib_0.f44` exists in the temporary working directory and contains mixture 1 data.

The `neutron/spectrum` is all that is needed to collapse to one-group reactions and interpolate fission product yields. However, this will neglect important cross section self-shielding effects. In general, self-shielding is very important for thermal systems and where there is a significant amount of a strong resonance absorber which will depress the flux spectrum and implicitly reduce the cross section in that energy range. This is the main purpose of the `xs_update` block which allows one to read multi-group cross sections from a file, collapse those with the flux spectrum from the `spectrum` block, and use those preferentially over any "unshielded" data from the reaction resource.

In the following input, the AMPX library is used in both the spectrum and cross section update. Note the `spectrum` and `xs_update` blocks are identical for the `AMPX_LIBRARY` type.

```
=origen

build_lib("B.f33") {

  nuclide {
    type=NAMED_SET
    list="complete_v6.2"
  }

  decay {
    type=ENDF_DECAY
    resource="${DATA}/origen_data/origen.rev03.decay.data"
  }

  neutron(1) {
    type=ENDF_ENERGY_DEPENDENT
    reaction_resource="${DATA}/origen.rev01.jeff56g"
    fp_yield_resource="${DATA}/origen_data/origen.rev05.yield.data"

    spectrum {
      type=AMPX_LIBRARY
      file="sysin.microWorkLib_0.f44"
```

(continues on next page)

(continued from previous page)

```
      mixture=1
    }
    xs_update {
      type=AMPX_LIBRARY
      file="sysin.microWorkLib_0.f44"
      mixture=1
    }
  }
}

end
```

The final optional component of the neutron block is the `coeff_update` list, which can directly set the values of transition coefficients both in terms of cross section channels and daughter yields.

The specific format is

```
coeff_update[
  parent "channel(yield)" value
  ...
]
```

where

- `parent` is the parent nuclide id or name,
- `channel` is the numeric identifier for the reaction channel, commonly called "MT" numbers (see Sect. 10.1.3.1 for details),
- `yield` (optional) is the isomeric state of the daughter (e.g., 0 or 1) or the actual nuclide identifier in the case of fission (channel=18), and
- `value` is the actual value for the one-group reaction cross section or yield.

For example, consider this code excerpt which modifies the fraction of ^{241}Am (n,γ) which results in ^{242m}Am and ^{242}Am. Note that an appropriate energy-integrated value was determined from the reaction resource and the flux spectrum, and this update overwrites that value.

```
=origen

build_lib("B.f33") {

  ...

  neutron(1) {
    ...
    coeff_update[
      am241 "102(1)" 0.7 % 70% of n,gamma go to Am-242m
      am241 "102(0)" 0.3 % 30% of n,gamma go to Am-242
    ]

  }
}

end
```

Note: Yields are always renormalized to sum to 1.0 as a final step before creating the ORIGEN library, except for fission (channel=18), which is renormalized to sum to 2.0. In the above example, if you provided only one yield (e.g., the 0.70 to ^{242m}Am), then the final yields will be renormalized to sum to 1.0, and your final yield will not be 0.70.

Data Cutoffs

The legacy COUPLE library building (see Sect. 5.1.9.4) included an internal cutoff for decay constants of approximately 2.0E-26 and for cross sections of approximately 1e-20 barns. It makes sense to have cutoffs from the perspective of computational efficiency; however, one issue with cutoffs is that as the system evolves, there may be reactions below the cutoff (and not included) that then should be included. Although we could allow for this flexibility eventually, the ability to change the nuclides and transitions considered during a depletion calculation is not allowed, and it must be fixed from the beginning. For this reason, `allow_zero` options exist in both the decay and neutron blocks to change this legacy cutoff to zero.

See example Example 5.1.32 for details.

Sensitivity Calculation Block `sens`

Sensitivity calculations in ORIGEN allow one to determine the change in a particular nuclide or group of nuclides to all nuclear data involved in the calculation. The implementation here follows that of [ORIGEN-Wil86] for the specific case where perturbations in data do not alter the neutron energy spectrum.

Consider the following complete case, which irradiates 1 gram of ^{238}U using the `transition.def` ORIGEN library file. This is a useful library to use for testing but has assumed a uniform flux spectrum and thus will not be accurate for any real scenario.

```
=origen
solver{
  type=cram
}

case(A){
  lib{ file="transition.def" }
  mat{ iso=[ u238=1.0 ] units=GRAMS }
  time=[ 20I 3 1000]
  flux=[ 22R 1e14 ]
  save=yes
}

sens{
  case=A
  response{ iso=[pu239=1.0] type=NUCLIDE step=LAST }
  threshold=1e-4

  dp_verify{
    nrank=3
    nrank_pert=1.001
  }
}

end
```

The components of the `sens` input are as follows.

- `case`: the name of the case to consider the forward calculation. Note that the default name for the first case is "1", second case is "2", etc.
- `response`: a block that defines the type of response (only `type=NUCLIDE` is currently supported).
  - `iso`: the realization vector for the nuclide response given in the same format as the `iso` array in the `mat` block.
  - `step`: the step index at which the relative change in the response will be analyzed.
- `threshold`: the minimum sensitivity coefficient to print in the output file.

- dp_verify: a block that defines "direct perturbation" cases to verify the sensitivities. Direct perturbations can point out useful issues with time steps too large for the forward/adjoint calculations or a response that is particularly difficult to integrate (e.g., rapidly changing amount over a time step). Even though the forward and adjoint solutions will be accurate on the time steps, the integration of the forward solution multiplied by the adjoint over all steps assumes a simple trapezoid rule and could be inaccurate.
  - nrank number of direct perturbations to perform per category of sensitivity (currently 6) for each sensitivity coefficient which exceeds the threshold.
  - nrank_pert perturbation factor to assume for the direct perturbation verification. For example, "1.01" would indicate to multiply the data of interest by 1.01 (i.e., increase by 1%).

Sensitivity output tables

The main output of the sensitivity calculation is sent to the .out file and is shown below.

```
------------------------------------------------------------------------
Sensitivity Calculation Summary (#1/1)
------------------------------------------------------------------------

Forward calculation case                          = A
Target step (from case)                           = 22
Corresponding target time (seconds)               = 86400000.000000
Sensitivity coefficient report threshold          = 0.000100
Response type                                     = nuclide
Number of direct perturbation checks per group    = 3
Direct perturbation factor (multiplier)           = 1.001000
Adjoint concentration file                        = adjoint.sens1-adj.f71

---------------------------------------------------------------------
Total Decay Loss Sensitivity
---------------------------------------------------------------------
-------------------------------------------------------------------------------------
| Nuclide ID | Adjoint Sens. Coeff. | Direct Sens. Coeff. | Direct F71 File         |
-------------------------------------------------------------------------------------
|      93239 |          7.83906e-02 |         1.10034e-03 |  adjoint.sens1-dps1.f71 |
|      94241 |          4.77040e-04 |         4.77023e-04 |  adjoint.sens1-dps2.f71 |
|      96242 |          2.49246e-04 |         2.51713e-04 |  adjoint.sens1-dps3.f71 |
-------------------------------------------------------------------------------------

---------------------------------------------------------------------
Decay Loss Sensitivity
---------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
| Nuclide ID |    Decay Mode   | Adjoint Sens. Coeff. | Direct Sens. Coeff. | Direct F71 File          |
-----------------------------------------------------------------------------------------------------------
|      93239 |           beta- |          7.83906e-02 |         1.10034e-03 |   adjoint.sens1-dps4.f71 |
|      94241 |           beta- |          4.77029e-04 |         4.77013e-04 |   adjoint.sens1-dps5.f71 |
|      96242 |           alpha |          2.49246e-04 |         2.51713e-04 |   adjoint.sens1-dps6.f71 |
-----------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------
Decay Yield Sensitivity
---------------------------------------------------------------------
No sensitivities exceed the threshold.

---------------------------------------------------------------------
Total Reaction Loss Sensitivity
---------------------------------------------------------------------
-------------------------------------------------------------------------------------
| Nuclide ID | Adjoint Sens. Coeff. | Direct Sens. Coeff. | Direct F71 File         |
-------------------------------------------------------------------------------------
```

(continues on next page)

(continued from previous page)

```
|      94239 |         -9.92411e-01 |        -9.89541e-01 |  adjoint.sens1-dps7.f71 |
|      92238 |          6.58620e-01 |         7.32985e-01 |  adjoint.sens1-dps8.f71 |
|      93239 |         -1.80910e-03 |        -1.95421e-03 |  adjoint.sens1-dps9.f71 |
|      94238 |          4.60255e-04 |                   - |                       - |
|      95241 |          2.95686e-04 |                   - |                       - |
|      94241 |         -2.14775e-04 |                   - |                       - |
-------------------------------------------------------------------------------------


----------------------------------------------------------------------
Reaction Loss Sensitivity
----------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------
| Nuclide ID |     Reaction    | Adjoint Sens. Coeff. | Direct Sens. Coeff. | Direct F71 File         |
-------------------------------------------------------------------------------------------------------------
|      92238 |         n,gamma |          6.59038e-01 |         7.33404e-01 | adjoint.sens1-dps10.f71 |
|      94239 |         fission |         -6.55423e-01 |        -6.53213e-01 | adjoint.sens1-dps11.f71 |
|      94239 |         n,gamma |         -3.37762e-01 |        -3.36715e-01 | adjoint.sens1-dps12.f71 |
|      93239 |         n,gamma |         -1.80585e-03 |                   - |                       - |
|      94238 |         n,gamma |          4.63905e-04 |                   - |                       - |
|      92238 |         fission |         -3.71955e-04 |                   - |                       - |
|      95241 |         n,gamma |          2.96838e-04 |                   - |                       - |
|      94241 |         fission |         -1.57067e-04 |                   - |                       - |
-------------------------------------------------------------------------------------------------------------


----------------------------------------------------------------------
Reaction Yield Sensitivity
----------------------------------------------------------------------
No sensitivities exceed the threshold.
```

The six groups of sensitivities are as follows.

1. Total decay loss sensitivity – the relative change in response per change in a nuclide's total decay constant.
2. Decay loss sensitivity – the relative change in response per change in specific decay channels (e.g., α-decay).
3. Decay yield sensitivity – the relative change in response per change in a decay yield—for example, the fraction of γ-decays that result in the ground state daughter versus a metastable daughter.
4. Total reaction loss sensitivity – the relative change in response per change in a nuclide's total loss cross section.
5. Reaction loss sensitivity – the relative change in response per change in specific reaction channels (e.g. (n,γ)).
6. Reaction yield sensitivity – the relative change in response per change in a reaction yield, (e.g., fission product yields).

Note that for each group, three direct perturbation calculations are performed because `nrank=3`. Thus, if an ORIGEN forward calculation only takes a second, and the adjoint calculation takes approximately the same, in this case there are 12 additional forward calculations performed. If there were any yield sensitivities, the number could be as high as 18. For this reason, small `nrank` is recommended.

According to the sensitivity output, the dominant sensitivities are related to reactions on ^{238}U and ^{239}Pu, which makes sense. The reaction loss sensitivity table shows eight reactions that exceeded the sensitivity coefficient threshold of `1e-4`. It is interesting to note that some comparisons of the adjoint sensitivity versus the direct sensitivity are very close, whereas the ^{238}U (n,γ) has an adjoint sensitivity of 0.66 and a direct of 0.73. This gap may decrease if the time steps for the forward calculations in case "A" are reduced; however, it may also simply be a non-linearity.

The adjoint solution utilized here is first-order perturbation theory, whereas the direct perturbation may capture additional orders of variation if the perturbation applied (nrank_pert) is "large." In this particular case, reducing the perturbation factor to 1.0001 had no effect, but adding an additional 20 time steps resulted in a closer adjoint value of 0.70 compared to an unchanged direct value of 0.73.

Sensitivity output files

A number of f71 files are produced as a result of the sensitivity calculation.

- ${BASENAME}-sens<i>-adj.f71 contains the adjoint solution for the <i>-th sensitivity calculation.
- ${BASENAME}-sens<i>-dps<k>.f71 contains the <k>-th direct perturbation solution for the <i>-th sensitivity calculation. Note the "Direct F71 File" column in the table indicates which one corresponds to which perturbation.

5.1.6 ARP MODULE

The ARP module performs multidimensional interpolation on a set of specially prepared ORIGEN libraries using interpolation methods discussed in Sect. 5.1.3.1.14, with available interpolators listed in Table 5.1.7. The ORIGEN reactor libraries distributed with this version of SCALE are described in the ORIGEN Reactor Libraries chapter, as well as details on how users can generate their own libraries. The ARP module has been validated extensively for light water reactor (LWR) spent fuel [ORIGEN-LHBP98]. Benchmarking studies for MOX fuel were also conducted [ORIGEN-Gau03].

Table 5.1.7: Interpolation options in ARP

Type	Interpolation keyword	Comments
Nearest value	nearest	Searches for closest value to the desired value
Linear interpolation	linear	Uses nearest two values bounding the desired value
Lagrangian interpolation	lagrange(N) *with order N from 1 to 4*	Uses N points near desired value and creates a polynomial of order N-1 using Eq. (5.1.54)
	lagrange *same as lagrange(4)*	The specification of lagrange(1) is equivalent to nearest and lagrange(2) to linear.
Standard cubic spline	stdspline	Standard, natural cubic spline (without monotonicity fix-up).
Monotonic cubic spline	spline	Natural cubic spline with a monotonicity fix-up designed to prevent nonphysical oscillations that in some cases may result in negative interpolated cross sections.

Parametrizations for three types of problems have been developed: uranium-based fuel, mixed-oxide (MOX) fuel, and general activation.

- The parametrization for uranium-based fuel (e.g., UO:sub:*2*), as would be found in most LWRs, can interpolate on
  - fuel enrichment,
  - moderator density, and
  - burnup.

- The parametrization for MOX fuel contains a mixture of plutonium and uranium oxide and can interpolate on
  - total plutonium content in the heavy metal,
  - plutonium isotopic vector (Pu vector) that defines the relative concentrations of the Pu isotopes,
  - moderator density, and
  - burnup.
- The parametrization for general activation problems has only one-dimensional interpolation on fluence.

Variation of the absorption cross sections was observed to be near linear as a function of Pu content. Interpolation on the Pu vector is more complex than the uranium enrichment for UO_2 fuel since the vector is composed of five different isotopes: ^{238}Pu, ^{239}Pu, ^{240}Pu, ^{241}Pu, and ^{242}Pu. Furthermore, the elements in the vector depend on one another and can therefore not be evaluated independently of one another since the entire vector must sum to 100%. The scheme developed for the Pu vector was based on an evaluation of a large database of plutonium compositions from actual MOX fuel assemblies of European origin. It might be expect edthat the parametrization would need to include all Pu isotopes. However, an evaluation of the MOX fuel database indicated that there is a strong correlation between ^{239}Pu and the other isotopes in the vector that permits cross sections for the MOX fuel to be determined to sufficient accuracy using only the ^{239}Pu concentration.

5.1.6.1 Input Description

ARP has a simple input scheme, a different line-by-line input expected for each of the three problem types—uranium, MOX, or activation—with the input required for each type shown in Table 5.1.8, Table 5.1.9, and Table 5.1.10. Available input depends on what is available in the relevant arpdata.txt file and the arplibs directory.

Table 5.1.8: Input description for uranium fuels

Entry #	**Data type**	**Entry requirements**	**Comment**
1	Data set name	Line 1 always required	Enter a uranium CONFIGNAM from the active arpdata.txt (see Table 5.1.11).
2	Enrichment	New line always	Enter the wt % ^{235}U in toal U
3	Number of cycles	Always	Enter the number of irradiation cycles N.
4	Fuel irradiation period	Always	Enter the irradiation time for each cycle in days ΔT_i, for $i = 1, 2, \ldots, N$.
5	Average power	Always	Enter the specific fission power (MW/MTHM) for each cycle P_i, for $i = 1, 2, \ldots, N$.
6	Data interpolations per cycle	Always	Enter the nummber of cross section sets to interpolate during each cycle m_i, for $i = 1, 2, \ldots, N$.
7	Moderator density	Always	Enter the moderator density (g/cm^3). Enter only one value
8	New library name	New line always	Enter the filename of the new ORIGEN library created from interpolation.

continues on next page

Table 5.1.8 – continued from previous page

Entry #	Data type	Entry requirements	Comment
9	Interpolation keyword	Optional	Enter the interpolation algorithm which will be used from Table 5.1.7 (**DEFAULT: spline**)

Table 5.1.9: Input description for MOX fuels

Entry #	Data type	Entry requirements	Comment
1	Data set name *(starts with MOX)*	Line 1 Always required	Enter a MOX CONFIGNAM from the active arpdata.txt (see Table 5.1.12)
2	Plutonium content	New line always	Enter the Pu content as wt % Pu in total heavy metal.
3	^{239}Pu isotopic vector	Always	Enter the ^{239}Pu isotopic concentration as wt % ^{239}Pu in total Pu.
4	Reserved parameter (not used)	Always	Enter a dummy value (e.g., 1.0)
5	Number of cycles	Always	Enter the number of irradiation cycles N.
6	Fuel irradiation period (days)	Always	Enter the irradiation time for each cycle days ΔT_i, for $i = 1,\ 2, \ldots, N$.
7	Average power (MW/MTHM)	Always	Enter the specific fission power (MW/MTHM) for each cycle, P_i, for $i = 1,\ 2, \ldots, N$.
8	Data interpolations per cycle	Always	Enter the number of cross section sets to interpolate during each cycle, m_i for $i = 1,\ 2, \ldots, N$.
9	Moderator density	Always	Enter the water moderator density (g/cm^3). Enter only one value.
10	New library name	New line always	Enter the name of the new interpolated library created by ARP.
11	Interpolation keyword	Optional	Enter the interpolation algorithm which will be used from Table 5.1.7 (**DEFAULT: spline**)

Table 5.1.10: Input description for activaiton problems

Entry no.	Data type	Entry requirements	Comment
1	Data set name *(starts with ACT)*	Line 1 always required	Enter an activation CONFIGNAM from the active arpdata.txt see Table 5.1.13
2	Dummy parameter	Always	Enter 1.

continues on next page

Table 5.1.10 – continued from previous page

Entry no.	Data type	Entry requirements	Comment
3	Number of cycles	Always	Enter the number of irradiation cycles N.
4	Fuel irradiation period	Always	Enter the irradiation time for each cyce time in days ΔT_i, $i = 1, 2, \ldots, N$.
5	Average neutron flux	Always	Enter the average flux level (n/cm^2-s) for each cycle, Φ_i, for $i = 1, 2, \ldots, N$.
6	Data interpolations per cycle	Always	Enter the number of cross section sets to interpolate during each cycle, m_i, for $i = 1, 2, \ldots, N$.
7	Flux type (flag)	Always	Enter 1.
8	New library name	New line always	Enter the name of the new interpolated library created by ARP.
9	Interpolation keyword	Optional	Enter the interpolation algorithm which will be used from Table 5.1.7 (**DEFAULT: spline**)

5.1.6.2 ARPDATA.TXT listing file

In addition to the user input file, ARP also reads a file named arpdata.txt when it runs. This file describes the parametrization of the ORIGEN libraries. The file is required because the cross section libraries contain no imbedded information on the reactor type, fuel type, or irradiation conditions. Both the file arpdata.txt and the directory of ORIGEN libraries named arplibs is searched for, first in the working directory so that a user can override the default libraries, and then to the SCALE data directory. An example arpdata.txt file is shown in Example 5.1.20

Example 5.1.20: Examples of arpdata.txt entries.

```
!ce14x14
6 1 11
1.5 2.0 3.0 4.0 5.0 6.0
0.7332
'ce14_e15.f33' 'ce14_e20.f33' 'ce14_e30.f33'
'ce14_e40.f33' 'ce14_e50.f33' 'ce14_e60.f33'
0. 1500. 4500. 7500. 10500. 13500.
16500. 31500. 46500. 58500. 70500.


!mox_bw15x15
3 5 1 1 10
4.0000 7.0000 10.0000
50.0000 55.0000 60.0000 65.0000 70.0000
1.0
0.7135
'mox_bw15_e40v50.f33' 'mox_bw15_e70v50.f33' 'mox_bw15_e10v50.f33'
'mox_bw15_e40v55.f33' 'mox_bw15_e70v55.f33' 'mox_bw15_e10v55.f33'
'mox_bw15_e40v60.f33' 'mox_bw15_e70v60.f33' 'mox_bw15_e10v60.f33'
'mox_bw15_e40v65.f33' 'mox_bw15_e70v65.f33' 'mox_bw15_e10v65.f33'
'mox_bw15_e40v70.f33' 'mox_bw15_e70v70.f33' 'mox_bw15_e10v70.f33'
0.00 1040.00 3000.00 5000.00 7500.00


!w17x17
6 1 11
```

(continues on next page)

(continued from previous page)

```
1.5 2.0 3.0 4.0 5.0 6.0
0.723
'w17_e15.f33' 'w17_e20.f33' 'w17_e30.f33'
'w17_e40.f33' 'w17_e50.f33' 'w17_e60.f33'
0. 1500. 4500. 7500. 10500. 13500.
16500. 31500. 46500. 58500. 70500.
```

As shown in Example 5.1.20, the `arpdata.txt` is simply a list of entries, each beginning with a `!CONFIGNAM`, where CONFIGNAM is the name to be used to reference the entire data set. Whether the entry is for a uranium, MOX, or activation problem is dictated by the actual CONFIGNAM. If it begins with MOX, it is a MOX entry, and if it begins with ACT, it is an activation entry. Otherwise it is uranium. The ORIGEN libraries listed must reside next to `arpdata.txt`:, in a directory called arplibs. Each type of entry is described fully in Table 5.1.11, Table 5.1.12, and Table 5.1.13 for uranium, MOX, and activation, respectively.

Table 5.1.11: ARPDATA.TXT uranium-type entry

Line no.	Data name	Description	Comments
1	CONFIGNAM	Data set name	Must begin with "!" in column one, followed by the alphanumeric name this data will be referenced by.
		(40-character maximum)	May not begin with ACT or MOX
2	N1	Number of enrichments	Entries pertain to the number of parameterized cross section data points for each variable type.
	N2	Number of water densities	
	N3	Number of burnup steps	
3	ENR	Enrichment values (wt % ^{235}U); values at which ARP libraries were generated	N1 entries defining the discrete enrichment values for each library
4	DENS	Water density values (g/cm^3)	N2 entries defining the discrete moderator density values for each library
5	FILES	Filenames of ORIGEN libraries for this fuel assembly type	N1 × N2 entries
		(Enclose each filename in single quotes with at least one space between each name.)	Filenames are ordered first by density values, then by enrichment values.
6	BURN	Burnups (MWd/MTU) corresponding to each position on the ORIGEN library	N3 entries

continues on next page

Table 5.1.11 – continued from previous page

Line no.	Data name	Description	Comments
			Each set of burnup-dependent cross sections is stored within a single ORIGEN binary library file (the first burnup is usually zero).
NOTE: Repeat all of the above entries for each fuel assembly configuration type			

Table 5.1.12: ARPDATA.TXT MOX-type entry

Line no	Data name	Description	Comments
1	CONFIGNAM	Data set name (40-character maximum)	Must begin with "!" in column one, followed by the alphanumeric name by which this data set will be referenced.
			Must begin with MOX (e.g., !mox_bw15x15).
2	N1	Number of Pu content values	Entries pertain to the number of separate cross section sets generated for each parameter.
	N2	Number of ^{239}Pu values	
	N3	Not used (enter 1)	
	N4	Number of water densities	
	N5	Number of burnup steps	
3	PU	Pu content values (wt % Pu in heavy metal)	N1 entries
4	VECT	^{239}Pu | N2 entries vector values | (wt % ^{239}Pu/Pu) |	
5	RESRV	Not used (enter 1)	N3 entries; dummy entry required.
6	DENS	Water density | N4 entries values | (g/cm^3) |	
7	FILE	Filenames of ORIGEN libraries for this fuel assembly type. Enclose each filename in single quotes with at least one space between each name.	N1 × N2 × N3 × N4 entries
			Increment FILE names in the order of N1, then N2, then N3, and then N4 values
8	BURN	Burnups (MWd/MTU) corresponding to each position on the ORIGEN library	N5 entries
			(first burnup is usually zero)
NOTE: Repeat all of the above entries for each fuel assembly configuration type			

Table 5.1.13: ARPDATA.TXT activation-type entry

Line no.	Data name	Description	Comments
1	CONFIGNAM	Data set name	Must begin in columnn one followed by the alphanumeric name by which this data set will referenced.
		(40-character maximum)	Must begin with ACT (e.g., !actcntlrod).
2	N1	Reserved (enter 1)	The first two entries pertain to the number of separate cross section sets generated for each variable parameter.
	N2	Not used (enter 1)	
	N3	Number of fluence values	These are usually set to 1.
			The variable N3 corresponds to the number of fluence-dependent cross section sets available in the library.
3	RESRV	Not used (enter 1)	Enter 1.
4	FTYPE	Neutron flux type (flag)	Enter 1.
5	FILES	Filenames of ORIGEN library. Enclose filename in single quotes.	Generally only one one library name is required.
6	FLUENCE	Neutron fluence values (n/cm:sup`2`) at each of the ORIGEN libraries	N3 entries
			The fluence values are reduced by the factor 10^{-24} to avoid numerical problems during the interpolation
			(First value is usually zero.)
NOTE: Repeat all of the above entries for each fuel assembly configuration type			

5.1.7 OPUS MODULE

The analysis of ORIGEN results often involves evaluating large amounts of output data that may include time-dependent nuclide and element inventories, radiological decay properties, and neutron, alpha, beta and photon source spectra. Visualization of the output data from ORIGEN often provides a means of rapidly evaluating the dominant nuclides in a problem, identifying important trends in the results, and providing insight into the problem that is not easily obtained otherwise. The OPUS utility program has been developed to read and process ORIGEN binary concentration results file (f71) into a format easily entered into graphics-plotting packages.

5.1.7.1 Key Features

The output of OPUS is a PLT file with extension ".plt" which can be easily read by most graphics packages, including the SCALE GUI Fulcrum. An example of the PLT file is shown in Figure 5.1.28.

Figure 5.1.28. Example of the OPUS PLT file.

The various types of data that can be returned by OPUS fall into two separate classes, briefly described as (1) dominant or selected isotopes or elements and (2) photon, alpha, beta and neutron source spectra. These requested data may be extracted for either irradiation or decay time periods of interest. One key feature of the program is that by default, it will automatically extract the 40 most dominant nuclides or elements in the problem, rank them using the output response and time periods of interest as specified by the user, and give the total response. The returned data may be selected, ranked, and plotted according to the group (types) of nuclides in the problem. The groups include the actinides, fission products, activation products, fission products and actinides, or all nuclides in the problem. The totals are printed for all nuclides in the specified group, and a subtotal is generated for all nuclides in the printed list. The user may also specify any other nuclide or element to be included in the output, regardless of its importance to the problem. The nuclides may be ranked using different output units of mass, density, atomic density, activity, toxicity, decay heat, or isotopic fractions.

Default values are assigned to most input variables, allowing a comprehensive visual analysis of returned data with relatively little user input.

5.1.7.2 Input Description

OPUS input records are limited to a maximum of 80 columns. A single data entry may be entered anywhere in a record but cannot be divided between two records; however, array-data entries may be divided over many records. A data entry (from left to right) is composed of a keyword, an equal sign (=), and numeric or alphanumeric data. A data entry is illustrated in the following example:

```
title="PWR 33 GWD/MTU 3.3 wt % U-235"
```

The program identifies keywords either by using only the first four characters in the keyword name or the full keyword name. In the example above, "titl" may be used, omitting the "e."[1] Alphanumeric data (as titles or axis labels) must be enclosed in double quotes, with nuclide or element symbols being the notable exception.

Floating-point data may be entered in various forms; for example, the value 12340.0 may be entered as 12340, 12340.0, 1.234+4, 1.234E+4, 1.234E4, or 1.234E04. Also, the value 0.012 may be entered as 12E-3, 12-3, 1.2-2, etc. Numeric data must be followed immediately by one or more blanks.

Array data (alphanumeric floating-point and integer) for an array-type parameter must always be terminated with an END that does not begin in column one.

Input parameters in a single OPUS sequence input (i.e., between "=opus" and "end") may be entered in any order. Usually each sequence input produces a single plot file with the name "${OUTBASENAME}N.plt" where ${OUTBASENAME} is the base of the input file name, e.g., my.inp has the base name "my," and "N" is an 18-digit integer that is incremented with each opus plot.

The OPUS input uses keywords. Keywords, descriptions, and allowed choices for each entry are listed in Table 5.1.14 for plot setup options and Table 5.1.15 for response units and nuclide selection. Input keywords and their choices occupy the first column. The next two columns indicate the type of plot in which the keyword may be entered. The last column is a description of the input keyword.

[1] Unlike in the previous OPUS versions, it is not permitted to enter four characters followed by any alphanumeric characters: only full keyword names or four-letter short-hands are allowed.

Table 5.1.14: OPUS input for plot setup

Input keywords	Nuclide / element plot	Spectra plot	Keyword description
LIBRARY= (“”)	*	*	ORIGEN binary cross section library (.f33). Only required in order to perform reaction rate calculations (see ABSO and FISS units below).
DATA= (“ft71f001”)	*	*	ORIGEN concentrations file (.f71). If enetered, must be enclosed in double quotes.
TITLE= (“”)	*	*	Title of plot. If entered, must be enclosed in double quotes.
TYPARAMS=			Inventory data to be plotted
NUCLIDES	*	NA	individual nuclides (DEFAULT)
ELEMENTS	*	NA	elements
ASPECTRUM	NA	*	alpha spectrum
BSPECTRUM	NA	*	beta spectrum
GSPECTRUM	NA	*	photon spectrum
TYPARAMS=			Spectral data to be plotted
NSPECTRUM	NA	*	total neutron spectrum from all sources
SFSPEC	NA	*	spontaneous fission
ANSPEC	NA	*	(α, n)
DNSPEC	NA	*	delayed neutrons
FACTOR=	*	NA	User-defined multiplicative factor for all concentrations
VOLUME=	*	NA	Material volume in cm ^{3}; required when plotting in units of grams/cm ^{3} or atoms/barn-cm
XLABEL=	*	*	X- and Y-axis labels. If this parameters is not entered, a label is generated automatically. If entered, it must be enclosed in double quotes
YLABEL=			
COLS= (YES)/NO			Print nuclide edits by columnn (one nuclide per column)
ROWS= (YES)/NO			Print nuclide edits by columnn (one nuclide per row)
TIME=	*	NA	Unit of time to be plotted
SECONDS	*	NA	seconds
MINUTES	*	NA	minutes
HOURS	*	NA	hours
DAYS	*	NA	days (DEFAULT)
YEARS	*	NA	years

Table 5.1.15: OPUS input for response units and nuclide selection

Input keywords	Nuclide / element plot	Spectra plot	Keyword description
NRANK=(40)	*	N/A	Total number of nuclides or elements to be returned for the plot. If user-selected nuclides or elements are requested (see SYMNUC), output will include these nuclides plus any remaining ones with the highest rankings for the quantity specified. The total and subtotal are always printed. Default is not used if SYMNUC is entered.
UNITS= (GATO) (INTENSITY)	*	N/A	Requested data units Gram-atoms are default for nuclide / element plots, and particles / MeV-sec are default for spectrum plots
ABSORPTIONS	*	N/A	Absorption / removal rate (using removal XS on library)
AIRM**3	*	N/A	Radiotoxicity; cubic meters of air to dilute to RCG_a
APELEM	*	N/A	Atom % of element; isotopic atom percentages of all elements specified in SYMNUC
ATOMS / B-CM	*	N/A	Atoms/barn-cm; requires VOLUME entry
BECQUERELS	*	N/A	Radioactivity, Bq.
CAPTURES	*	N/A	Capture reaction rate (removal minus fission; not equivalent to radiative capture)
CURIES	*	N/A	Radioactivity, Ci
FISSIONS	*	N/A	Fission reaction rate (using XS on library).
GAMWATTS	*	N/A	Gamma-ray thermal power, watts
GATOMS	*	N/A	Gram atoms, gram-atomic weights or moles
GPERCM**3	*	N/A	Partial density, grams/cm^3; requires VOLUME entry
GRAMS	*	N/A	Mass, grams
H2OM**3	*	N/A	Radiotoxicity; cubic meters of water to required to dilute to to RCG_w
KILOGRAMS	*	N/A	Mass, kilograms
WPELEM	*	N/A	Weight % of element; isotopic weight percentages of all elements specified in SYMNUC
WATTS	*	N/A	Total thermal power, watts
PARTICLES	N/A	*	Print spectra in units of particles/sec
INTENNSITY	N/A	*	Print spectra in units of particles/MeV-sec (normalized to energy bin width)
ENERGY	N/A	*	Print spectra in units of MeV/MeV-s (normalized to energy bin width; multiplied by mean bin energy)
LIBTYPE=	*	N/A	Selects sublibrary(ies) for nuclide/elements
LITE	*	N/A	Include light-nuclide sublibrary
ACT	*	N/A	Include actinide sublibrary
FISS	*	N/A	Include fission product sublibrary
FISSACT	*	N/A	Include actinide and fission products
ALL	*	N/A	Include all (DEFAULT)

continues on next page

Table 5.1.15 – continued from previous page

Input keywords	Nuclide / element plot	Spectra plot	Keyword description
SYMNUC=	*	N/A	Symbolic notation of nuclides or elements requested in the plot data. The list is an array terminated with an END. Nuclide entry is by chemical symbol and mass number, separated by a dash "–". Metastable states are indicated with an "m" immediately following the mass number (e.g., ^{242m}Am). Element entry is by chemical symbol only (e.g., Pu) For nuclide entry, it is also allowed to use element identifiers; in such cases, all isotopes of the element found in the ORIGEN library are used.
SORT=	*	N/A	Option to sort user-requested nuclides input via the SYMNUC array by descending order of importance. If SORT is not given or is SORT=NO, the order of the nuclides in the plot table is the same order as in the input. Options that can be given to SORT are the same as for UNITS. If SORT=YES, the same units provided in the UNITS entry are used.
RESPONSE=	*	N/A	Optional, user-specified arbitrary response conversion factor array. These factors are applied to the individual nuclides after conversion to quantities in the units requested by UNITS. Entries are in nuclide ID-response factor pairs. Nuclide identifiers are entered as legacy ORIGEN ZZAAAI 6-digit integer IDs, where the isomeric state is the last digit. The array must be terminated with an END.

The default values are given in parentheses after the input keyword definitions. Only the first four characters of any keyword (the letters are in bold) plus the "=," are required; they can be used instead of the full keyword name. However, unlike in previous versions of OPUS, the only two options accepted are the full keyword name or its first four letters. The keyword default value is listed in parentheses, and values are provided for almost all numeric input and axis labels. For example, the user has the option to enter axis labels using XLABEL= and YLABEL= entries, or "built-in" axis labels based on the time units and the type of data units can be used.

The remaining sections give more detail on the nuclide/element plots, the spectra plots, and time range, and case selection to further filter the results on an f71 file.

Nuclide/element plots

Various responses may be returned either by nuclide or by element, as a function of time, for a specified time interval. By default, the top nuclides/elements contributing to the response of interest over the time range are returned. An explicit list of nuclides/elements can also be entered, in which case these nuclides will always be included, regardless of their ranking. Results for nuclides in specific sublibraries may be filtered (e.g., to only see output of ^{155}Gd the fission product (FP) sublib and not the light nuclide version [LT] sublib. The default behavior is to sum over all sublibs.

Response totals are always generated for each plot. These totals represent the sum over all nuclides or elements in the specified library type. In addition, a subtotal is printed (both in the printout and the plot file) that represents the sum over the printed nuclides or elements. The subtotal can be very useful in identifying whether the nuclides in the printed list include all of the dominant nuclides in the problem.

OPUS allows the user to sort using any of the provided concentration units. For each nuclide/element, approximate integral value over all selected time steps is calculated in the selected units; nuclides/elements are then ranked according to this value. The NRANK cutoff is applied after the sort has been performed.

It is permitted to enter SORT=YES, which is equivalent to setting SORT= to the same value as UNITS=.

If the SORT key is not entered, nuclides/elements are sorted according to their position in the SYMNUC array.

Spectra plots

Alpha, beta, photon and neutron energy spectra calculated by ORIGEN may also be stored on the f71 file, and plotted by OPUS. One difference in the units of the spectra returned by OPUS and those printed by ORIGEN is that the intensities are typically converted from units of photons/s (stored on f71 file) to *particles/s-MeV* by dividing by the energy-group width. This conversion produces intensities that are more easily comparable across different energy groups and energy-group structures. Energy-intensity spectra in units of *MeV/s-MeV* and intensity spectra in units of *particles/s* may also be requested.

Selection of plotting data

OPUS provides four basic options for selecting a subset of the data on an f71 file:

1. a range of times with TMIN and TMAX (floating point numbers),
2. a range of positions on the f71 file with MINPOSITION and MAXPOSITION (integers),
3. a single CASE selection, and
4. a list of positions in the NPOSITION list.

A full overview of keywords that can be entered for time point selection is given in Table 5.1.16.

If no selection inputs are used, all available steps on the concentration file are printed. Range bounds (CASE, TMIN, TMAX, MINPOSITION, MAXPOSITION) are used as independent constraints, so any combination of those can be supplied, in which case all of the specified constraints are applied.

In case invalid positions are specified in the NPOSITION array, an error is raised, and no valid output is produced; the same happens if no steps fit into the specified selection.

The case number is useful for ORIGEN stacked cases. Furthermore, TRITON stores the mixture number (with 0 for the sum of all depletion materials and -1 for the sum of selected materials) as the case number, so this option can be used to obtain a plot of the desired depletion mixture without actually knowing the time

step numbers=. When an ORIGEN f71 file is accessed, a table of contents is printed in the output (.out) file. This can be very useful for understanding what is contained on the f71 at each position.

Table 5.1.16: OPUS input keywords for data set selection

Input keywords	Type	Keyword description
NPOSITION=	Array	List of all requested position numbers of data (on NUMUNIT), in ascending order. No other entries are required since NPOSITION is a unique data set identifier. Terminate with END. NPOSITIONS cannot be specified together with any of TMIN, TMAX, MINPOSITION or MAXPOSITION. If none of the position selectors is entered, all available positions on the data file will be returned.
CASE=	Variable	Number of ORIGEN case (or, a mixture number for TRITON-produced f71 files).
MINPOSITION=	Variable	Minimum position number, used to request a range of positions
MAXPOSITION=	Variable	Maximum position number All positions between MINPOSITION and MAXPOSITION, inclusive, are requested. MINPOSITION and/or MAXPOSITION cannot be specified together with NPOSITIONS, but they can be specified at the same time as TMIN and/or TMAX. Up to all four conditions (TMIN, TMAX, MINPOSITION, MAXPOSITION) will be used to constraint the selected time points. If none of the position selectors is entered, all available positions on the data file will be returned.
TMAX= **TMIN**=	Variable Variable	Minimum and maximum time for selection of positions on the data file. The units of TMAX and TMIN are the same as for TIME. TMIN and/or TMAX cannot be specified together with NPOSITIONS, but they can be specified at the same time as MINPOSITION and/or MAXPOSITION. Up to all four conditions (TMIN, TMAX, MINPOSITION, MAXPOSITION) will be used to constraint the selected time points. If none of the position selectors is entered, all available positions on the data file will be returned.

5.1.7.3 Plot File Formats

The plot file produced by OPUS is a text file that contains plot information, including the title, axis labels, plot type, and the time-dependent data for inventory-type plots or energy-dependent data for spectral-type plots. The plot file is a free-format reading, so the column positions have no particular significance. However, the plot files created by OPUS are aligned by column to facilitate reading of the files by other graphics programs.

File header information

The format of the first five records of the plot file is common to all plot types. They are:

Record 1: TITL - problem title (maximum 32 characters)

Record 2: XHEA - x-axis label (maximum 20 characters)

Record 3: YHEA. y-axis label (maximum 20 characters)

Record 4: Plot type - The type of plot is selected by the words "nuclide," "element," "case," or "spec." The nuclide and element entries have the same effect and produce time-dependent plots of individual nuclides or elements in the specified units. The "case" entry is used to identify case comparisons of total quantities or individual nuclides. The "spec" entry identifies spectral-type plots.

The remaining records and formats depend on the plot type specified in Record 4.

Nuclide and element plot types

Record 5: NPTONC, KTOT - the number of time points (x-axis) for the plot data KTOT - the number of nuclides or elements in the plot; includes the last record that contains the totals, if present

Record 6: TIMES(NPTONC) - NPTONC entries for the times associated with each data point; units are in the units specified in Record 2 (XHEA)

Record 7: SYMBOL, (X(N,I), I=1, NPTONC) - symbol is the alphanumeric symbol for the nuclide or element. No spaces are permitted in PlotOPUS. This symbol entry is followed by NPTONC entries containing the plot data for the nuclide or element SYMBOL with index N.

Record 8: Record 7 is repeated for all KTOT nuclides or elements.

Record 7+KTOT: Last record - contains the totals or other related quantity for the plot. The record has the same format as Record 7.

Spectral plot types

Record 5: NPTONC, NGP - the number of entries that follow to construct the spectral histogram, where NGP=(NG+1)*2 and NG is the number of energy groups. The histogram is made using a series of connecting lines, and therefore does not require any special histogram plotting capability in the graphics code.

Record 6: EN, YY - Each record contains a pair of entries: EN is the energy (MeV), and YY is the value for energy EN.

Repeat Record 6 NGP times.

Case comparison plot types

Record 5: NCAS - the negative value of the total number of cases in the plot

Record 6: NPTONC - the number of time points (x-axis) for the plot data for the case (non-user-entered data)

Record 7: INPOIN the number of time points (x-axis) for the plot data for user-entered data

Record 8: XINP(INPOIN) - INPOIN entries for the times associated with each data point of the user-entered data

Record 9: YINP(INPOIN) - INPOIN entries containing the plot data for the user-entered data

Record 10: LEGEND - legend for the user-entered data case.

The remaining records contain the case comparison data for the non-user-input data. Repeat records 7, 8, 9, and 10 for the case comparison plots, NPTONC times (same as case comparisons with no user-entered data).

5.1.8 EXAMPLES

The main problems solvable by the ORIGEN family of codes are enumerated below (with relevant components in parentheses).

1. Decay (ORIGEN)
2. Activation (ORIGEN)
3. Fuel irradiation (ARP+ORIGEN or ORIGAMI)
4. Emission spectra from decay (ORIGEN)
5. Processing, including batch/continuous chemical removal, isotopic feed, and stream blending (ORIGEN)
6. Unit conversions (OPUS)

Examples of the six variations above are contained in the following sections, except for fuel irradiation problems with ORIGAMI, as described in its own chapter.

5.1.8.1 Decay of ^{238}U

Example 5.1.21: Decay of ^{238}U

```
=origen
case{
    % use ENDF/VII-based decay library
    lib{ file="end7dec" }

    % create a material with 1 gram U-238
    mat{
        units=GRAMS
        iso=[u238=1.0]
    }
    time=[20L 1.0 1e9] %default units are days

    % save all information to f71
    save=yes
}
end
```

Example 5.1.21 illustrates using the "end7dec" binary decay library and decaying one gram of ^{238}U for 10^9 days using the logarithmic array shortcut to put 20 logarithmically spaced values between 1.0 and 1e9 days.

5.1.8.2 ^{252}Cf neutron Emission Spectrum

Example 5.1.22: ^{252}Cf neutron emission spectrum.

```
=origen

bounds{
    neutron=[ 2.000000e+07 6.376300e+06 3.011900e+06 1.826800e+06
              1.422700e+06 9.071800e+05 4.076200e+05 1.110900e+05 1.503400e+04
              3.035400e+03 5.829500e+02 1.013000e+02 2.902300e+01 1.067700e+01
              3.059000e+00 1.855400e+00 1.300000e+00 1.125300e+00 1.000000e+00
              8.000000e-01 4.139900e-01 3.250000e-01 2.250000e-01 1.000000e-01
              5.000000e-02 3.000000e-02 1.000000e-02 1.000000e-05 ]
}

case{
    title="Cf-252 decay"

    lib{ file="end7dec" pos=1 }

    time{
       units=YEARS
       t=[ 0.01 0.03 0.1 0.3 1 3 10 ]
    }

    mat{
       units=CURIES
       iso=[cf252=1.0]
    }

    %perform neutron calculation with defaults
    neutron=yes

    print{
       neutron{
           summary=yes
           spectra=yes
           detailed=yes
       }
    }
}
end
```

Example 5.1.22 illustrates decay of 1 Ci of ^{252}Cf for ten years with calculation of the time- and energy-dependent neutron source. The case uses the binary decay library "end7dec." The neutron energy group structure is defined in the bounds block with array "neutron."

5.1.8.3 Simple Fuel Irradiation Plus Decay

Example 5.1.23 illustrates the input for irradiation of a fuel assembly for 200 days at 15 MW in case "irrad," followed by decay for 5 years in the next case, "decay." The ORIGEN library (f33) was prepared by ARP to have a single position with a burnup of 1500 MWd/MTU. The input material has been specified to have 1 MTU total with an enrichment of 4.0% in ^{235}U. Note that there are consistency requirements between ARP and ORIGEN that cannot be checked by either module. Namely, the enrichment specified in ARP should be equivalent to the effective enrichment specified in the ORIGEN input, and the burnup-dependent cross section data on the ORIGEN library should be for the *midpoint burnup of the case*, which requires consistency between the operating history (power and time) and the initial isotopics heavy metal loading. In typical fuel depletion calculations, it is most convenient to specify a metric ton (10^6 grams) of initial heavy metal, such that the power in MW may be interpreted as MW/MTIHM. In the above example, the power history does not need to be constant but when combined with the time values should produce an average burnup of 1500

MWd/MTU in order that the cross sections interpolated by ARP in position 1 are valid.

Example 5.1.23: Simple fuel irradiation plus decay.

```
=arp
'library type
w17x17
'wt%
4.0
'number of cycles
1
'number of days per cycle
200.0
'cycle-average specific power (MW/MTU)
15.0
'number of interpolated cross section sets generated per cycle
1
'moderator density (g/cc)
0.723
'interpolated output ORIGEN library
w17x17_100d.f33
end
=origen
case(irrad){
    % use xs data at pos=1 corresponds to midpoint burnup (200 d * 15MW/MTU)/2
    lib {
         file="w17x17_100d.f33" pos=1
    }
    % 1 MT of enriched uranium
    mat {
        units=GRAMS
        iso=[u234=356 u235=40000 u236=184 u238=959460]
    }
    % power history (at least 4 steps for MATREX)
    time=[ 50 100 150 200 ] %default time in days
    power=[ 15 15 15 15 ] %power in MW
}
case(decay){
    time{
        units=YEARS
        start=0 %start time at 0 in this case for ease of input for t[]
        t=[0.1 0.3 0.9 1 2 3 4 5] %observe rule of threes
    }
    save{ file="discharge.f71" steps=[0 LAST] } %only save begin and end
}
end
```

It is important to note that with the MATREX solver, used by default, the recommendation for irradiation and decay of spent fuel is to use no fewer than four steps for the irradiation and begin the decay period with a time step on the order of weeks or a month, increasing the interval for each subsequent step by no more than a factor of three. In many continuation cases, such as the decay case described here, it is convenient to specify times starting from zero for the case with “start=0” in the time block.

5.1.8.4 Three Cycles of Irradiation Plus Decay

Example 5.1.24 is similar to the previous one, except there are three sets of burnup-dependent transition cross sections (positions) generated by ARP and used in ORIGEN, and there are three cases, one corresponding to each cycle. Neutron and gamma sources are generated and saved to the f71 file for the final decay case. The maximum burnup achieved is 60 GWd/MTIHM.

Example 5.1.24: Three cycles of irradiation plus decay

```
=arp
w17x17
4.0
3
500 500 500
40.0 40.0 40.0
1 1 1
0.723
ft33f001
end

=origen
bounds{ neutron="xn27g19v7.0"
        gamma=[1e+7 8e+6 6.5e+6 5e+6 4e+6 3e+6 2.5e+6 2e+6 1.66e+6 1.33e+6 1e+6
               8e+5 6e+5 4e+5 3e+5 2e+5 1e+5 5e+4 1e+4]}
case(c1){
    lib{ file="ft33f001" pos=1 }
    time=[8i 50 500]
    power=[10r40]
    mat{ iso=[u235=4e3 u238=960e3] }
}
case(c2){
    lib{ pos=2 }
    time=[8i 550 1000]
    power=[10r40]
}
case(c3){
    lib{ pos=3 }
    time=[8i 1050 1500]
    power=[10r40]
}
case(cool){
    time{ start=0 t=[20L 0.001 100] units=YEARS }
    save{ file="snf.f71" time_offset=1500 }
 gamma{ sublib=ALL brem_medium=UO2 }
 print{ neutron{ spectra=yes } }
 neutron{ alphan_medium=UO2 }
 }
 end
```

5.1.8.5 Load Isotopics from an f71 File

The "mat" block allows isotopics to be loaded from any position on an existing f71 file, as seen in Example 5.1.25. A table of contents is printed in the output file every time an f71 is read or written. Inspection of this table can help identify the appropriate position on the f71, as shown in Example 5.1.26, extracted from the output file.

Example 5.1.25: Load isotopics from an f71 file using the "mat" block and perform a follow-on decay.

```
'copy existing f71 file from input file directory to working directory
=shell
```

(continues on next page)

(continued from previous page)

```
cp ${INPDIR}/discharge.f71 discharge2.f71
end

=origen
case(restart){
    % decay only library
    lib {
        file="end7dec" pos=1
    }
    % must know correct position
    mat {
        load{ file="discharge2.f71" pos=2 }
    }
    % continue timeline from previous case ending at 5 years
    time{
        units=YEARS
        start=5
        t=[10 20 40 80 160]
    }
    % append to file
    save{
        file="discharge2.f71" steps=[1 2 3 4 5]
    }
}
end
```

Example 5.1.26: Table of contents printed when accessing the f71 file.

```
================================================================================
= Restart F71 File for case 'restart' (#1/1) =
--------------------------------------------------------------------------------
Data taken from position: 2
index time power flux fluence burnup libpos case step DCGNAB
    1 0.00000e+00 0.00000e+00 0.00000e+00 0.00000e+00 3.00000e+03 1 2 0 DC----
    2 1.57800e+08 0.00000e+00 0.00000e+00 0.00000e+00 3.00000e+03 1 2 8 DC----
D - state definition present
C - concentrations present
G - gamma emission spectra present
N - neutron emission spectra present
A - alpha emission spectra present
B - beta emission spectra present
```

The simple input in Example 5.1.27 can be used to print contents of an existing f71 file. The file is renamed so that it does not have extension ".f71." This prevents the automatic copy back from the working directory to the input file directory. Note that the same is true with any ORIGEN libraries that have the extension ".f33." There is a rule in place that *any* file in the working directory with extension ".f71" or ".f33" is copied back to the input file's directory. To prevent this from occurring, the ".f71" or ".f33" extensions must not be used in the filename, or unneeded files with a shell command at the end must be explicitly deleted.

Example 5.1.27: Isotopics from an f71 or f33 file.

```
'avoid .f71 extension to prevent automatic copy back
=shell
cp /path/to/unknown.f71 f71
end

=origen
case(test){
    lib{ file="end7dec" pos=1 }
```

(continues on next page)

(continued from previous page)

```
    mat{ load{ file="f71" pos=1 } }
    time=[1]
}
end

'remove any *.f71 or *.f33 in the working directory to prevent automatic copy back
'(not necessary in this example, but for reference)
=shell
rm -f *.f71
rm -f *.f33
end
```

5.1.8.6 Continuous Feed and Removal

Both continuous feeding of nuclides into a system and chemical removal of elements from a system are required to simulate molten fuel systems such as the molten salt reactor (MSR). Simulation of removal can be used to represent other physical processes such as purification systems (i.e., removal of chemical species by filtration or ion-exchange columns) and ventilated systems in which the removal can be represented using a rate constant (1/s). Example 5.1.28 applies to simulation of a molten salt reactor system, and it uses both continuous feed of ^{232}Th and simultaneous removal of fission products according to their removal process. To actually simulate this system properly, an ORIGEN reactor library appropriate for the MSR must be generated.

There are 11 groups of nuclides in this example, each with the same removal constant. The groups, by element and removal rate in units of (1/s), are

- Group 1: Ca to As (3.37×10^{-9});
- Group 2: Y, La, Ce, Pr, Nd, Pm, Sm, Gd (2.31×10^{-7});
- Group 3: Eu (2.31×10^{-8});
- Group 4: Se, Nb, Mo, Tc, Ru, Rh, Pd, Ag, Sb, Te (5×10^{-2});
- Group 5: Zr, Cd, In, Sn (5.79×10^{-8}) ;
- Group 6: Kr, Xe (5×10^{-2});
- Group 7: Br, I (1.93×10^{-7});
- Group 8: Rb, Sr, Cs, Ba (3.37×10^{-9});
- Group 9: Th, Li, Be, F (3.37×10^{-9});
- Group 10: Pa (3.86×10^{-6}); and
- Group 11: Np, Pu, Am, Cm, Bk, Cf (1.98×10^{-9}).

A ^{232}Th feed rate of 2.0×10^{-2} grams/s is used.

Example 5.1.28: Demonstration of continuous feed and removal within an ORIGEN irradiation case.

```
=origen
case{
    title="Single fluid MSR depletion calculation"

    lib{
       file="msr.f33"
```

(continues on next page)

(continued from previous page)

```
      pos=1
   }
   time{
       units=YEARS
       t=[ 8i 0.05 1.0 ]
   }

   power=[ 10r30 ] %30 MW

   %initial material
   mat{
      %FLiBe with pure Li7 and 1 MTIHM loading
      units=GRAMS
      iso =[f=1e7 li7=5e6 be=1e6 th232=9.5e5 u233=0.5e5]

      %continuous feed of th-232
      feed=[th232=2e-2] %g/s
   }

   %continuous removal by element using atomic numbers
   processing{ removal{rate=3.37e-9 ele=[12i 20 33]}
               removal{rate=2.31e-7 ele=[39 57 58 59 60 61 62 64]}
               removal{rate=5.79e-8 ele=[40 48 49 50]}
               removal{rate=5e-2 ele=[36 54]}
               removal{rate=1.93e-7 ele=[35 53]}
               removal{rate=3.37e-9 ele=[37 38 55 56]}
               removal{rate=3.37e-9 ele=[90 3 4 9]}
               removal{rate=3.86e-6 ele=[91]}
               removal{rate=1.98e-9 ele=[93 94 95 96 97 98]}
   }

   print{
        cutoffs=[ GRAMS=0.1 ] %do not show grams < 0.1% of total
        nuc{ units=[GRAMS] total=yes }
        kinf=yes   %print k-infinity summary
        absfrac_sublib=LT %absorption fractions for light nuclides in FLiBe
   }
} %end case
end
```

The absorption rates, fission rates, and k-infinity values are printed during irradiation, which can be used to evaluate the influence of the feed rates and removal constants on the time-dependent reactor performance. For the MSR in particular, the ability to self-sustain can be assessed from the "k-infinity" summary output, enabled by "kinf=yes" in print block (as well as the absorption fractions in the light nuclides in FLiBe), and enabled by "absfrac_sublib=LT" in the print block (Example 5.1.29). This example uses pure Li-7, whereas if natural Li is used, one will see a much larger fraction of absorptions in Li-6 and much lower "k-infinity."

Example 5.1.29: Continuous feed and removal–history overview.

```
================================================================================================
= History overview for case '1' (#1/1)
= Single fluid MSR depletion calculation
------------------------------------------------------------------------------------------------
   step          t0          t1          dt           t        flux     fluence       power      energy
    (-)         (y)         (y)         (s)         (s)   (n/cm2-s)     (n/cm2)        (MW)       (MWd)
      1      0.0000      0.0500  1.5780E+06  1.5780E+06  1.9782E+13  3.1217E+19  3.0000E+01  5.4792E+02
      2      0.0500      0.1556  3.3313E+06  4.9093E+06  2.0112E+13  9.8216E+19  3.0000E+01  1.7046E+03
      3      0.1556      0.2611  3.3313E+06  8.2407E+06  2.0575E+13  1.6676E+20  3.0000E+01  2.8613E+03
      4      0.2611      0.3667  3.3313E+06  1.1572E+07  2.1057E+13  2.3691E+20  3.0000E+01  4.0181E+03
      5      0.3667      0.4722  3.3313E+06  1.4903E+07  2.1557E+13  3.0872E+20  3.0000E+01  5.1748E+03
      6      0.4722      0.5778  3.3313E+06  1.8235E+07  2.2075E+13  3.8226E+20  3.0000E+01  6.3315E+03
```

(continues on next page)

(continued from previous page)

```
      7      0.5778      0.6833  3.3313E+06  2.1566E+07  2.2613E+13  4.5759E+20  3.0000E+01  7.4882E+03
      8      0.6833      0.7889  3.3313E+06  2.4897E+07  2.3171E+13  5.3478E+20  3.0000E+01  8.6449E+03
      9      0.7889      0.8944  3.3313E+06  2.8229E+07  2.3750E+13  6.1390E+20  3.0000E+01  9.8016E+03
     10      0.8944      1.0000  3.3313E+06  3.1560E+07  2.4351E+13  6.9502E+20  3.0000E+01  1.0958E+04

              step - step index within this case
                t0 - time at beginning-of-step in input units
                t1 - time at end-of-step in input units
                dt - length of step in seconds
                 t - end-of-step cumulative time in seconds
              flux - flux in neutrons/cm^2-sec (CALCULATED)
           fluence - cumulative end-of-step fluence in neutrons/cm^2 (CALCULATED)
             power - power in mega-watts (INPUT)
            energy - cumulative end-of-step energy released in mega-watt-days (INPUT)
================================================================================================



================================================================================================
=   Overall neutron balance for case '1' (#1/1)
=   Single fluid MSR depletion calculation
------------------------------------------------------------------------------------------------
                      0.0E+00y    5.0E-02y    1.6E-01y    2.6E-01y    3.7E-01y    4.7E-01y
n-production          2.9579E+18  2.9272E+18  2.9110E+18  2.9123E+18  2.9138E+18  2.9154E+18
n-absorption          2.6018E+18  2.6178E+18  2.6858E+18  2.7707E+18  2.8583E+18  2.9489E+18
k-inf                 1.1369E+00  1.1182E+00  1.0838E+00  1.0511E+00  1.0194E+00  9.8864E-01
================================================================================================

================================================================================================
=   Fraction of absorption rate for light elements for case '1' (#1/1)
=   Single fluid MSR depletion calculation
------------------------------------------------------------------------------------------------
            0.0E+00y    5.0E-02y    1.6E-01y    2.6E-01y    3.7E-01y    4.7E-01y
f-19        9.5276E-02  9.4189E-02  9.2293E-02  9.0506E-02  8.8782E-02  8.7113E-02
be-9        9.4833E-02  9.3751E-02  9.1862E-02  9.0083E-02  8.8366E-02  8.6704E-02
li-7        6.3293E-02  6.2572E-02  6.1312E-02  6.0125E-02  5.8980E-02  5.7871E-02
li-6        0.0000E+00  5.6409E-04  1.7104E-03  2.8000E-03  3.8355E-03  4.8190E-03
he-3        0.0000E+00  2.9340E-08  7.1009E-07  3.0708E-06  7.9374E-06  1.5939E-05
o-16        0.0000E+00  4.3552E-08  1.3502E-07  2.2606E-07  3.1678E-07  4.0728E-07

*{list continues}*
```

5.1.8.7 Calculate Fuel (α, n) Emissions in a Glass Matrix

Batch processing options are provided to separate various components of the nuclide compositions into different streams and to recombine the streams to form new compositions. This example applies to the irradiation of typical commercial fuel and subsequent storage of separated fission products and actinides from fuel in a glass matrix. The matrix is important in determining the $(alpha, n)$ component of the neutron source because the alpha particles interact with the light element constituents in the matrix, with $(alpha, n)$ yields corresponding to the medium containing the α-emitting nuclides. Therefore, an accurate calculation of the neutron source in a glass matrix requires combining the oxide fuel compositions after irradiation with the defined glass matrix. The calculation could be performed by creating a case by manually entering the calculated nuclide activities and the matrix composition. However, this is only practical if the number of source nuclides is small. This example applies batch processing and blending options to generate the required compositions.

The blending option is used to combine two streams: one stream from irradiated fuel, and the other stream defining the glass matrix composition.

An irradiation case—"irrad"—is performed first to generate spent nuclear fuel compositions. The next case "spent" decays the results for one year. At the start of the decay, only selected elements are retained in the

stream by performing processing with the "retained" array. In this example, all elements are removed except for Se (99.8%); Rb, Sr, Te, Cs, Ba, Dy (77.8%); and U, Np, Pu, Am, Cm (1%).

The glass matrix compositions are then defined in the third case, "glass."

The final case—"blend"—blends 10% of each the last step's isotopics from the "spent" and "glass" cases. To test the dependence on the time when the blend is performed, the blend can be changed to "blend=[spent(N)=0.1 glass=0.1]," where *N* is the index of the step from which to take isotopics from the spent case.

Example 5.1.30: Continuous feed and removal–blending option.

```
=arp
w17x17
3
1
360
40
1
0.723
fuellib
end

=origen
bounds{
    neutron=[ 1.00E-05 1.00E-02 3.00E-02 5.00E-02 1.00E-01 2.25E-01
              3.25E-01 4.14E-01 8.00E-01 1.00E+00 1.13E+00 1.30E+00
              1.86E+00 3.06E+00 1.07E+01 2.90E+01 1.01E+02 5.83E+02
              3.04E+03 1.50E+04 1.11E+05 4.08E+05 9.07E+05 1.42E+06
              1.83E+06 3.01E+06 6.38E+06 2.00E+07 ]
}
case(irrad){
    title="Fuel Stream 1 Irradiation"
    lib{ file="fuellib" pos=1 }
    time=[ 8I 36 360 ]
    power=[ 10r40 ]
    mat{
        units=GRAMS
        iso=[u234=534 u235=60000 u236=276 u238=939190]
    }
}
case(spent){
    title="Fuel Stream 1 Decay"
    time{
        t=[ 0.1 0.3 1 3 10 30 100 300 360 ] start=0 %enter times from 0
    }
    processing {
        retained=[se=0.998 rb=0.778 sr=0.778 te=0.778 cs=0.778 ba=0.778
                  dy=0.998 u=0.010 np=0.010 pu=0.010 am=0.010 cm=0.010]
    }
}
case(glass){
    title="100 kg glass" time{ t=[ 1 ] start=0 }
    mat{
        units=GRAMS
        iso=[li=2.18e3 b=2.11e3 o=46.4e3 f=0.061e3 na=7.65e3 mg=0.49e3 al=2.18e3
             si=25.4e3 cl=0.049e3 ca=1.08e3 mn=1.83e3 fe=8.61e3 ni=0.70e3
             zr=0.88e3 pb=0.049e3 ]
    }
}
case(blend){
    title="final blended case"  time=[ 1.01 3 10 30 100 ] %continue previous
    mat{
        blend=[ spent=0.1 glass=0.1 ] %blend factors of 0.1 for each
```

(continues on next page)

(continued from previous page)

```
   }
 neutron{ alphan_medium=CASE } %use this case's isotopics
 print{ ele{ total=yes units=[GRAMS] }
        neutron{ summary=yes spectra=yes detailed=yes }
 }
}
end
```

5.1.8.8 Create an ORIGEN Decay Library from a Decay Resource

The following ORIGEN input (Example 5.1.31) is all that is required to produce a binary decay library from the ORIGEN decay resource file. Note that ORIGEN decay resources can also be raw ENDF formatted data files.

Example 5.1.31: Creation of an ORIGEN decay library from the decay resource

```
=origen

build_lib("origen.end7dec") {
  nuclide {
    type=NAMED_SET
    list="complete_v6.2"
  }
  decay {
    type=ENDF_DECAY
    resource="${DATA}/origen_data/origen.rev03.decay.data"
  }
}

end
```

5.1.8.9 Create an ORIGEN Reaction Library

Example 5.1.32 creates an ORIGEN reaction library from an AMPX library created using the SCALE `t-depl-1d` sequence. This particular sequence is used to create the ORIGEN library which contains all possible reactions and decays between the standard set of nuclides used for all Polaris and TRITON calculations. For this reason, `allow_zero=yes` is important because it makes sure no reactions are eliminated. The spectrum in this case is not important because Polaris and TRITON will overwrite the values for each depletable zone at each time step.

Example 5.1.32: Creation of an ORIGEN reaction library.

```
=t-depl-1d parm=(addnux=4,bonami)
PWR pincell calculation to get transition definition skeleton
v7-56

read comp
'Fuel
uo2  1 den=10.412  1  900 92234 0.04 92235 4.11  92238 95.85   end
wtptzirc 25 6.44 4 40000 97.91 26000 0.5 50116 0.86 50120 0.73 1.0 600  end
h2o  26 den=0.6798  1  593 end
wtptbor 26 0.6798 1 5000 100 500e-6 593 end
end comp

read celldata
  latticecell squarepitch hpitch=.83116 26 fuelr=0.47815 1 cladr=0.5588 25 end
end celldata
```

(continues on next page)

(continued from previous page)

```
read depletion
  1
end depletion

read burndata
  power=40  burn=0 end
end burndata

read keep
  microlibfile
end keep

read model
Infinite lattice PWR pin cell
read parm
 prtflux=no collapse=yes prtmxsec=no
 sn=4 inners=5 outers=100 epsouter=1e-6
 epsglobal=1e-6
end parm
read materials
 mix=1 pn=0 com='4.11% enriched fuel' end
 mix=25 pn=0 com='cladding' end
 mix=26 pn=0 com='water' end
end materials
read geom
  geom=cylinder leftBC=refl rightBC=white
  zoneIDs 1 25 26 end zoneIDs
  zoneDimensions .47815 .55880 .83116 end zoneDimensions
  zoneIntervals 3r8 end zoneIntervals
end geom
end model
end

=origen

build_lib("origen.transition.def") {

  nuclide {
    type=NAMED_SET
    list="complete_v6.2"
  }

  decay {
    type=ENDF_DECAY
    resource="${DATA}/origen_data/origen.rev03.decay.data"
  }

  neutron(1) {
    type=ENDF_ENERGY_DEPENDENT
    reaction_resource="${DATA}/origen.rev01.jeff56g"
    fp_yield_resource="${DATA}/origen_data/origen.rev05.yield.data"

    %%%%%%%%%%%%%%
    allow_zero=yes
    %%%%%%%%%%%%%%

    spectrum {
      type=MULTIGROUP
      flux=[56R 1.0]
    }

    xs_update {
      type=AMPX_LIBRARY
      file="sysin.microWorkLib_0.f44"
```

(continues on next page)

(continued from previous page)

```
      mixture=1
    }

  }
}

end

=shell
cp origen.transition.def ${OUTDIR}/pwr.rev04.orglib
end
```

5.1.8.10 Create an ORIGEN Activation Library

Activation calculations typically do not require self-shielding, so an ORIGEN "activation" library is a means to refer to a library that is using infinitely dilute cross sections from the reaction resource. An ORIGEN activation library can be created very easily as long as the flux spectrum is available in one of the reaction resource group structures.

In the example below, we create a fast spectrum library based on the 200-group reaction resource and irradiate one gram of iron for 10 days at 1e15 flux.

Example 5.1.33: Creation of an ORIGEN activation library.

```
=origen
build_lib("ff.f33") {
    neutron(1) {
        type=ENDF_ENERGY_DEPENDENT
        reaction_resource="n200.reaction.data"
        spectrum {
          type=MULTIGROUP
          flux=[20r 1.0 180r 0.0]
        }
    }
}
case {
  lib {file="ff.f33"}
  mat { iso=[fe=1.0] units=GRAMS }
  time=[10]
  flux=[1e15]
}
end
```

5.1.8.11 Create an ORIGEN Library with User-Supplied Cross Sections

It is possible to change specific cross sections on ORIGEN libraries. The following input uses all default values for all data (the flux spectrum from the reaction resource will be used) but changes the U-238 fission (18) and n,gamma (102) reactions to 20 and 10 barns, respectively.

Example 5.1.34: Creation of an ORIGEN library with user-supplied cross sections.

```
=origen

build_lib("my.f33") {
  neutron(1) {
    type=ENDF_ENERGY_DEPENDENT
    reaction_resource="n252.reaction.data"
    coeff_update [
```

(continues on next page)

(continued from previous page)

```
        u238 18  20
        u238 102 10
    ]
  }
}

end
```

5.1.8.12 Printing library cross-section values

Print the Cross Section Values on an ORIGEN Library with OBIWAN

The `obiwan` utility was introduced in SCALE 6.3 to help users interrogate the data contained in ORIGEN binary files. From the command line, one can view all coefficients on the library with the following. See Sect. 5.2 for details.

```
obiwan view my.f33
```

Print the Cross Section Values on an ORIGEN Library in ORIGEN

Cross sections can also be printed when the library is used in ORIGEN by setting "print_xs=yes" in the options block, shown as Example 5.1.35.

Example 5.1.35: Printing cross section values on an ORIGEN library using ORIGEN.

```
=shell
cp ${INPDIR}/my_rx.f33 ft33f001
end

=origen
options{ print_xs=yes }
case{
    lib{ file="ft33f001" pos=1 }
    time=[1] %dummy time
    mat{ iso=[u235=1] } %dummy iso
}
end
```

5.1.8.13 Ranking Contribution to Toxicity

The OPUS input in Example 5.1.36 creates a plot of the volume of various nuclides with the maximum permissible concentration (MPC) in water using the "snf.f71" produced by the "Three cycle plus decay" example (Example 5.1.24). Only positions 6–16 are plotted using "minposition" and "maxposition." Three nuclides (^{242}Cm, ^{137m}Ba, and ^{99}Tc) are forced to be included via the "symnuc" list, with 17 more nuclides (for total "nrank=20") included according to their average rank in terms of the MPC in water. The total number of nuclides requested is 20. The x-axis label is set to "TIME (YEARS)," and the title is "SPENT FUEL AT 60 GWD/MTHM."

Example 5.1.36: Using OPUS to produce a ranked contribution to radiotoxicity

```
=shell
  cp ${INPDIR}/snf.f71 f71
end

=opus
```

(continues on next page)

(continued from previous page)

```
data="f71"
typarams=nucl
units=h2om**3
time=year
libtype=fisact
minposition=6  maxposition=16   nrank=20
title="SPENT FUEL AT 60 GWD/MTIHM "
xlabel="TIME (YEARS)"
symnuc=cm-242 ba-137m tc-99 end
end
```

5.1.8.14 Spectrum plots with OPUS

Photon Spectrum Plot

The OPUS input shown in Example 5.1.37 creates a plot of the photon spectrum for all times between 1 and 5 years, using "tmin=1" and "tmax=5" with "time=years."

Example 5.1.37: Plotting the photon spectrum using OPUS.

```
=shell
cp ${INPDIR}/snf.f71 f71
end

=opus
data="f71"
typarams=gspectrum
units=intensity
tmin=1 tmax=5 time=years
end
```

Example 5.1.38 shows the output of the time-dependent gamma spectrum extracted from the f71 file by OPUS.

Example 5.1.38: Photon spectrum plot from OPUS– output of the time-dependent gamma spectrum.

1/(s.MeV)	1.25y	2.15y	3.73y
1.00e+01 -- 8.00e+00	2.7961e+11	3.0661e+03	3.0661e+03
8.00e+00 -- 6.50e+00	7.1201e+13	1.9436e+04	1.9436e+04
6.50e+00 -- 5.00e+00	8.9725e+15	2.5300e+10	5.3041e+09
5.00e+00 -- 4.00e+00	3.7075e+16	9.1643e+12	1.9001e+12
4.00e+00 -- 3.00e+00	1.1734e+17	1.3562e+14	2.8824e+13
3.00e+00 -- 2.50e+00	3.4321e+17	5.5945e+15	4.8216e+15
2.50e+00 -- 2.00e+00	5.2847e+17	1.1338e+16	5.9433e+15
2.00e+00 -- 1.66e+00	8.3528e+17	3.3015e+16	1.8382e+16
1.66e+00 -- 1.33e+00	1.8060e+18	2.7233e+17	2.4658e+17
1.33e+00 -- 1.00e+00	2.4156e+18	1.4238e+17	9.2264e+16
1.00e+00 -- 8.00e-01	5.0662e+18	3.2193e+17	2.6845e+17
8.00e-01 -- 6.00e-01	6.4083e+18	2.1010e+18	1.8903e+18
6.00e-01 -- 4.00e-01	7.1803e+18	1.1822e+18	1.0301e+18
4.00e-01 -- 3.00e-01	8.7297e+18	1.7856e+18	1.6414e+18
3.00e-01 -- 2.00e-01	1.9481e+19	1.0658e+19	9.7018e+18
2.00e-01 -- 1.00e-01	2.9105e+19	1.5198e+19	1.3996e+19
1.00e-01 -- 5.00e-02	6.2160e+19	9.1756e+18	8.3611e+18
5.00e-02 -- 1.00e-02	1.1219e+20	3.4628e+19	3.1458e+19

Neutron Spectrum Plot

The OPUS input in Example 5.1.39 creates a plot of the total neutron spectrum at all times.

Example 5.1.39: Neutron spectrum plot using OPUS

```
=opus
data="f71"
typarams=nspe
units=intensity
end
```

5.1.8.15 Isotopic Weight Percentages for Uranium and Plutonium During Decay

The isotopic distributions in uranium and plutonium, in weight percent, may be plotted with the OPUS input in Example 5.1.40.

Example 5.1.40: Isotopic weight percentages for uranium and plutonium during decay.

```
=opus
data="f71"
units=wpel
libtype=act
symnuc=u pu end
end
```

5.1.8.16 User-Specified Response Function in OPUS

In Example 5.1.41, response conversion factor data are entered in the RESPONSE= array as nuclide-response factor pairs, with the nuclide id being the legacy nuclide ID (ZZAAAI), with ZZ two digits of atomic number, AAA three digits of mass number, and I one digit of isomeric state. For example, ^{235m}U would be given as 922351 and H-1 as 10010. In this example, it is assumed that the response factors are activity based (response/Bq of nuclide), so the units of Becquerels are requested using the UNITS keyword. The user-supplied response conversion factors are applied to the Becquerel units for all nuclides in the RESPONSE= array, and the results for any nuclide for which no response factors are provided are zeroed.

Example 5.1.41: Example of a user-specified response function in OPUS.

```
=opus
data="f71"
units=becq
time=year
typarams=nucl
response=
 270600 1.92027E+00  962460 2.27711E-01  481151 1.64222E-01
 822100 5.19246E-12  962470 1.75728E-09  501230 3.81237E-02
 832101 9.43341E-09  962480 1.05672E+00  511240 1.53438E+02
 882260 1.51600E-05  962500 1.60583E+02  511250 1.81023E-04
 892270 1.67080E-05  982490 2.21420E-04  511260 5.13360E+00
 902280 2.06832E-02  982500 2.87199E+02  end
end
```

5.1.9 APPENDICES

5.1.9.1 PRISM

Warning: PRISM is deprecated and replaced by Sampler (see Sect. 6.4 for details).

PRISM is a utility that reads a single input template file containing generic parameter flags and replaces them with specific values designated by the user to generate any number of files containing desired combinations of specific parameter values. PRISM provides a procedure to convert a generic input file for a particular fuel assembly design into a large number of input files containing combinations of specific fuel enrichment and moderator densities and/or other parameters for generating basic cross-section libraries. The program was designed in a general manner so PRISM can be used to generate multiple files from any generic file.

The input description for PRISM is presented in Table 5.B.1. The input format is free form. The user input includes the name of the template file to be read; the pattern for the name of the output files to be generated, using the generic parameter flags; the number of generic parameter flags; the number of files to be generated; each generic parameter flag and the specific values to be substituted in each output file.

The template file contains generic parameter flags. PRISM creates copies of the template file and substitutes specific values for the generic flags. **Note that the character length of each specific value must be the same as that of the associated generic flag.**

An example using PRISM to generate input files for six fuel enrichments is presented in Example 5.1.42 and Example 5.1.43. In this example a TRITON input file for a Westinghouse 17 × 17 fuel assembly (Example 5.1.43) is processed by PRISM using the input file for PRISM listed in Example 5.1.42. The case generates 6 specific input files from the template file.

Table 5.1.17: PRISM input description

Line No.	Parameter	Description	Comments
1	TEMPLATE	Template file name	80 characters maximum
2	OUT_TEMPLATE	Pattern for output file names	80 characters maximum Must contain enough generic parameter names to create unique filename for each output file
3	NUMPARMS	Number of generic parameter types	
	NUMFILES	Number of output files to be generated	
	NOTE: Repeat the following data for each generic parameter type (i.e. a total of NUMPARMS times).		
4	PARAM_NAME	Generic parameter name as it appears in template file	80 characters maximum
5[1]	PARAMETERS	Specific values of generic parameter for each output file	NUMFILES entries required. Length of value must be same as length of PARAM_NAME

[1] May be continued on subsequent lines as needed.

Example 5.1.42: PRISM input example to generate TRITON input files.

```
=shell
  cp $RTNDIR/w17x17_template.input .
end
=prism
  w17x17_template.input
  w17_u235.inp
  5 6
  u234wt%
  0.01200 0.01639 0.02543 0.03473 0.04423 0.05389
  u235
  1.50 2.00 3.00 4.00 5.00 6.00
  u236wt%
  0.00690 0.00920 0.01380 0.01840 0.02300 0.02760
  u238wt%
  98.4811 97.9744 96.9608 95.9469 94.9328 93.9185
  namelibrary
  w17_e15.lib
  w17_e20.lib
  w17_e30.lib
  w17_e40.lib
  w17_e50.lib
  w17_e60.lib
end
=shell
  cp w17*.inp $RTNDIR
end
```

Example 5.1.43: Generic TRITON input template for PRISM.

```
=t-depl
   PWR Westinghouse 17x17, 1/4 assembly model
   44groupndf5
   ' ----------------------------------------------------------------
   ' template to generate libraries for ORIGEN-S
   ' parameters are: u235 - wt% U-235
   ' u234wt% - wt% U-234
   ' u236wt% - wt% U-236
   ' u238wt% - wt% U-238
   ' namelibrary - name of ORIGEN library created
   ' ----------------------------------------------------------------
   ' Mixture data
   ' ----------------------------------------------------------------
   read comp
   ' fuel
   uo2 1 den=10.412 1 900 92234 u234wt%
   92235 u235
   92236 u236wt%
   92238 u238wt% end
   '
   ' clad
   zirc4 2 1 622 end
   ' water moderator with 630 ppm B
   h2o 3 den=0.723 1 575.5 end
   Figure .. Generic TRITON input template for PRISM.
   arbmb 0.723 1 1 0 0 5000 100 3 630e-06 575.5 end
   ' gap
   n 4 den=0.00125 1 622 end
   ' guide tube
   zirc4 5 1 575.5 end
   '
```

(continues on next page)

(continued from previous page)

```
end comp
' ----------------------------------------------------------------
' Cell data
' ----------------------------------------------------------------
read celldata
latticecell squarepitch pitch=1.259 3
fueld=0.805 1
gapd=0.822 4
cladd=0.95 2
end
end celldata
' ----------------------------------------------------------------
' Depletion data
' ----------------------------------------------------------------
read depletion
1
end depletion
' ----------------------------------------------------------------
' Burn data
' ----------------------------------------------------------------
read burndata
power=40.0 burn=1e-15 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
power=40.0 burn=75 down=0 end
end burndata
' ----------------------------------------------------------------
' NEWT model data
' ----------------------------------------------------------------
read model
Westinghouse 17x17
read parm
cmfd=yes xycmfd=4
run=yes echo=yes drawit=no
end parm
read materials
1 1 ! fuel ! end
2 1 ! clad ! end
3 2 ! water ! end
4 0 ! gap ! end
5 1 ! guide tube ! end
Figure 5.B.2. Generic TRITON input template for PRISM. (continued)
end materials
read geom
```

(continues on next page)

(continued from previous page)

```
unit 1
com='regular fuel rod'
cylinder 10 .4025
cylinder 20 .411
cylinder 30 .475
cuboid 40 4p0.6295
media 1 1 10
media 4 1 20 -10
media 2 1 30 -20
media 3 1 40 -30
boundary 40 4 4
unit 5
com='guide tube'
cylinder 10 .57175
cylinder 20 .6121
cuboid 40 4p0.6295
media 3 1 10
media 5 1 20 -10
media 3 1 40 -20
boundary 40 4 4
unit 11
com='right half of fuel rod'
cylinder 10 .4025 chord +x=0
cylinder 20 .411 chord +x=0
cylinder 30 .475 chord +x=0
cuboid 40 0.6295 0.0 2p0.6295
media 1 1 10
media 4 1 20 -10
media 2 1 30 -20
media 3 1 40 -30
boundary 40 2 4
unit 12
com='top half of fuel rod'
cylinder 10 .4025 chord +y=0
cylinder 20 .411 chord +y=0
cylinder 30 .475 chord +y=0
cuboid 40 2p0.6295 0.6295 0.0
media 1 1 10
media 4 1 20 -10
media 2 1 30 -20
media 3 1 40 -30
boundary 40 4 2
unit 51
com='right half of guide tube'
cylinder 10 .5715 chord +x=0
cylinder 20 .6121 chord +x=0
cuboid 40 0.6295 0.0 2p0.6295
media 3 1 10
media 5 1 20 -10
media 3 1 40 -20
boundary 40 2 4
unit 52
com='top half of guide tube'
cylinder 10 .5715 chord +y=0
cylinder 20 .6121 chord +y=0
cuboid 40 2p0.6295 0.6295 0.0
media 3 1 10
media 5 1 20 -10
media 3 1 40 -20
boundary 40 4 2
unit 53
com='1/4 instrument tube'
cylinder 10 .5715 chord +x=0 chord +y=0
cylinder 20 .6121 chord +x=0 chord +y=0
cuboid 40 0.6295 0.0 0.6295 0.0
```

(continues on next page)

(continued from previous page)

```
media 3 1 10
media 5 1 20 -10
media 3 1 40 -20
boundary 40 2 2
global unit 10
cuboid 10 10.7015 0.0 10.7015 0.0
array 1 10 place 1 1 0 0
media 3 1 10
boundary 10 34 34
end geom
read array
ara=1 nux=9 nuy=9 typ=cuboidal
fill
53 12 12 52 12 12 52 12 12
11 1 1 1 1 1 1 1 1
11 1 1 1 1 1 1 1 1
51 1 1 5 1 1 5 1 1
11 1 1 1 1 1 1 1 1
11 1 1 1 1 5 1 1 1
51 1 1 5 1 1 1 1 1
11 1 1 1 1 1 1 1 1
11 1 1 1 1 1 1 1 1
end fill
end array
read bounds
all=refl
end bounds
end model
end
=shell
  cp ft33f001.cmbined $RTNDIR/namelibrary
end
```

5.1.9.2 ARPLIB

Warning: ARPLIB is deprecated and replaced by the command line utility OBIWAN (see Sect. 5.2 for details).

ARPLIB is a utility program designed to read a burnup-dependent binary ORIGEN-ARP cross-section library and copy the cross-section data from only the desired burnup positions to create a new ORIGEN-ARP cross-section library.

The input for ARPLIB is described in Table 5.1.18. A new library (OUTLIB) is created by listing the positions from one or more existing libraries to copy to the new library.

An example of the input to ARPLIB is given in Example 5.1.44, showing how to use ARPLIB to reduce the number of cross sections sets on a library by creating a new library with only certain positions retained from the old library.

Table 5.1.18: ARPLIB input description

Parameter	Description	Comments
OUTLIB	Filename of output library to create	This library should not already exist

continues on next page

Table 5.1.18 – continued from previous page

NLIB	Number of input libraries to read	
	For each input library, i up to NLIB	
LIB-NAME[i]	Filename of input library to read	
NPOS[i]	Number of positions to read from this i-th library	>0
p1 p2 ...	The list of position indices from this i-th library to put on the output library	NPOS[i] position indices are read from a single line (all position indices are >0)

Example 5.1.44: ARPLIB example input to reduce size of ORIGEN cross-section libraries.

```
'get an ORIGEN library files for testing
=shell
  cp "${DATA}/arplibs/w17_e30.f33" 1_f33
end
'create a new library (ft33f001) with every-other burnup points
=arplib
  ft33f001
  1
  1_f33
  8
  1 3 5 7 9 11 13 15
 end
```

5.1.9.3 XSECLIST

Warning: ARPLIB is deprecated and replaced by the command line utility OBIWAN (see Sect. 5.2 for details).

The XSECLIST program is intended to provide an interpreted listing of any ORIGEN-ARP cross-section library. This utility program allows users to list the absorption and/or fission cross sections of any or all nuclides in the library as a function of burnup. The absorption cross sections are given for light elements, actinides, and fission products. Some of the light-element isotopes in the library may appear also as fission products; therefore, some isotopes may be listed twice. Fission cross sections may be listed for any or all actinides for which nonzero values of the cross sections exist. ORIGEN-S cross sections are typically normalized to thermal flux, rather than the total flux.

The nuclide ID numbers used in the library listings have the form IZ*10000+IA*10+IS, where

IZ = the atomic number;

IA = the atomic weight;

IS = 0, for ground state;

IS = 1, for metastable state.

The XSECLIST input is described in Table 5.1.19. The input is free format. The user specifies the library filename, the total number of burnup positions in the library, and the burnup values (GWd/MTU)

corresponding to each burnup position. The user then indicates whether the cross-section data listings are for absorption, fission, or both; and whether the listings are for all nuclides or only certain specified ones.

An example input file for XSECLIST is shown in Example 5.1.45. This example contains two cases: The first case lists both absorption and fission cross sections for ^{240}Pu in the ORIGEN-ARP 5 wt % enriched PWR 14 × 14 basic cross-section library. The output listing for this case is displayed in Example 5.1.46. These are microscopic cross sections listed in units of barns. The second case lists the fission cross sections for all nuclides in the ORIGEN-ARP 1.5 wt % enriched PWR 14 × 14 basic cross-section library. The output from this case is not presented here because of its size.

Table 5.1.19: XSECLIST input description

Line No.	**Parameter**	**Description**	**Comments**
1	FILENAME	Library filename	30-character maximum
2	NL	Number of burnup positions in library	
3[2]	BURN	Burnup (GWd/MTU) of each burnup position in library	NL entries required
4	CHARD	Cross-section data to be printed	a = absorption f = fission b = both
5	CHARL	List entire library (all nuclides)	y = yes n = no
	NOTE: The following optional data are entered only if CHARL = n.		
6	MT	Number of nuclides for which cross-section listings are desired	
7[?]	MTRD	Nuclide IDs[3]	MT entries required

Example 5.1.45: XSECTLIB input example

```
=xseclist
ce14_e50.arplib
10
0.0 1.5 4.5 7.5 10.5 13.5 16.5 31.5
46.5 58.5
b
n
1
942400
end
=xseclist
ce14_e15.arplib
10
0.0 1.5 4.5 7.5 10.5 13.5 16.5 31.5
46.5 58.5
f
y
end
```

[2] May be continued on subsequent lines as needed.

[3] Nuclide ID = Atomic No. * 10000 + Atomic wt * 10 + IS, where IS = 0 for ground state and 1 for metastable state.

Example 5.1.46: XSECLIST listing of ^{240}Pu data.

```
\*****\* absorption cross sections \*****\*
----------- light elements -----------
------- end of light elements --------
------------- actinides --------------
material= 942400 (pu240 )
burnup xsec
0.00000E+00 1.65356E+03
1.50000E+00 1.69928E+03
4.50000E+00 1.60593E+03
7.50000E+00 1.47163E+03
1.05000E+01 1.34200E+03
1.35000E+01 1.22895E+03
1.65000E+01 1.13326E+03
3.15000E+01 8.27437E+02
4.65000E+01 6.82870E+02
5.85000E+01 6.15974E+02
--------- end of actinides -----------
--------- fission products ----------
------ end of fission products -------
\****\* end of absorption cross sections \***\*
\*****\* fission cross sections \*****\*
material= 942400 (pu240 )
burnup xsec
0.00000E+00 4.51353E+00
1.50000E+00 4.71392E+00
4.50000E+00 4.82682E+00
7.50000E+00 4.89713E+00
1.05000E+01 4.93990E+00
1.35000E+01 4.96174E+00
1.65000E+01 4.96700E+00
3.15000E+01 4.82935E+00
4.65000E+01 4.56642E+00
5.85000E+01 4.34749E+00
\****\* end of fission cross sections \****\*
```

5.1.9.4 COUPLE Module

Warning: COUPLE is deprecated as of SCALE 6.3 with the ORIGEN `build_lib` capability replacing it. All of the capabilities of COUPLE are available and more. The legacy FIDO COUPLE input described here is still supported but a completely new computational engine is used to create the library, the same as is used for the `build_lib` block. See Sect. 5.1.5.2.21 for details.

COUPLE is a coupling code that prepares the transition matrix **A** from Eq. (5.1.2), which contains the decay and cross section transition rate constants according to the procedures defined in Sect. 5.1.3.1.1. The transition matrix and other important data are stored on an ORIGEN library (f33) file for use by other modules. COUPLE has two distinct modes of operation:

1. to create a new decay-only ORIGEN library from an ORIGEN decay resource, and
2. to add new or to update existing reaction transitions yield resource, reaction resource, and optionally an AMPX working library containing multigroup cross sections.

Details on the decay, yield, and reaction resources may be found in the ORIGEN Data Resources chapter.

Key Features

This section briefly highlights some key features in COUPLE and describes how they are used.

AMPX multi-group libraries

AMPX multigroup libraries contain multigroup cross sections by nuclide and material-zone identifiers. If the working library is the result of a multiregion transport calculation, then it is important to specify the correct zone identifier, e.g. corresponding to the fuel in a problem with moderator, clad, and fuel zones. The neutron flux is also stored on the AMPX library associated with a nuclide and a zone as are the cross sections. An AMPX library flux can be used to perform the cross section collapse as an alternative to providing a flux spectrum in the COUPLE input. New transitions may be added to the ORIGEN binary library for all reactions for which there are data in the weighted AMPX library if both the target and product nuclides are present in the ORIGEN library.

Nuclide Specification

In COUPLE, the following nuclide identifier is used:

```
Nuclide identifier = Z * 10000 + A * 10 + I
```

where

Z = atomic number,

A = mass number,

I = metastable/isomeric state (0 is ground/1 is first metastable)

Examples include 922350 for ^{235}U and 952421 for ^{242m}Am. Note that this varies from the identifiers used in other ORIGEN-related modules in which the isomeric state *I* comes first, as in 1095242 for ^{242m}Am.

Adding new transitions and user-defined transitions

The use of a transition matrix in ORIGEN allows any nuclide to transition to any other nuclide. By default, when the reaction data on the library is updated, then the transition matrix's sparse storage is expanded to include the new reaction transition if both the target and the reaction product nuclide are in the library. The user may request that the code does not add new transitions by setting Block1 `JADD=0` (`1$$ a4 0`). This option ensures that the matrix structure on the input library is identical to that of the output library. The user may explicitly set one-group transition coefficients by setting Block1 `LBUP=1` (`1$$ a3 1`) and entering Block6 and Block8 data.

Unit numbers and Aliases

In COUPLE, a unit number is used instead of a full file name to specify files, where unit number XY links to the data file "ftXYf001" in the working directory. For example, unit number 33 means file ft33f001. There are several predefined unit numbers that are controlled by a special "origen_filenames" file, which creates an alias for the local file "ftXYf001" to a file in the data directory. Table 5.1.20 shows the basic COUPLE unit numbers, their aliases, and a description of the file.

Table 5.1.20: Basic COUPLE unit numbers

Unit	Alias	Description
17	YIELDS	ORIGEN Yield Resource
21	END7DEC	ORIGEN library *ENDF/B-VII-based decay transitions only*
27	DECAY	ORIGEN Decay Resource
80	JEFF252G	ORIGEN Reaction Resource (252 groups)

An "origen_filenames" list which maps unit number 21 to alias "END7DEC" could link unit 21 to the file "${DATA}/origen.rev04.end7dec," where ${DATA} is the path to the SCALE data directory. To override this association, COUPLE must find a file named "ft21f001" in the working directory. The entire set of unit numbers is given in the ORIGEN Data Resources chapter**.**

Input Description

COUPLE uses the FIDO input system, except for title entries. The input is arranged in blocks, with each block containing one or more arrays, followed by the FIDO block terminator "t." Each input parameter is named and defined below in the order in which it appears, with the index of the parameter in the array. Some options have been deprecated over time and thus the first available entry may not correspond to index "1" and some indices may be skipped. Default values are given in parentheses. In the SCALE code system, COUPLE input appears between "=couple" and "end."

Block1: titles, unit numbers, and case controls.

TITLE — Title lines

Title lines can provide information about the ORIGEN library created and printed when the library is used. The input Block1 1\$\$ NUMA allows title lines to be copied from the input library to the output library.

The first blank line terminates the title.

A maximum of 40 lines can be included in the library.

A special title of "DONE" in the first four columns marks the completion of a COUPLE input case.

0\$\$ Array – Logical Unit Assignments

1. NOUT – Printed output unit number (6)
2. LIBDEC – Unit number of ORIGEN decay resource (27)

Only used if 1\$\$ LBIN=1

3. JD – Unit number of ORIGEN reaction resource (80)
4. ND – Input ORIGEN binary library file (21)

Only used if 1\$\$ LBIN=0

5. LD – Unit number of AMPX multigroup library file (0)

Only used if LD>0; energy group structure must be consistent with that on ORIGEN reaction resource (JD)

6. MD – Unit number for output ORIGEN library file (33)
8. NY – Unit number of ORIGEN yield resource (17)

1\$\$ Array – Control Constants [19 entries]

1. LBIN – 1/0 – Decay library creation/reaction update mode (0)

In decay library creation mode with LBIN=1, the reaction resource (0\$\$ JD) is not used, any input associated with reaction processing is ignored, and Block2 and Block8 may not be entered. In reaction update mode with LBIN=0, Block3 may not be entered.

2. PRT – 1/0 – Suppress all informational output / print informational output (0).

3. LBUP – 1/0 – Update from user input cross sections (Block6 and Block8 Arrays) / no user update (0).

4. JADD – 1/0 – Add/do not add new transitions to the library (1).

5. JEDT – 1/0 – Edit input library only/normal library generation case (0).

6. NXX – 1/0 – Allow/do not allow transitions with zero cross section (0).

7. NMO – Current month (as integer) for output library (0).

8. NDAY – Current day for output library (0).

9. NYR – Current year for output library (two digits) (0).

12. IDREF – Nuclide ID in AMPX multigroup library (0\$\$ LD) containing neutron flux weighting spectrum to use in cross section collapse (0).
If IDREF=0, uses first nuclide found in NZONE. Only used if NWGT=0.

13. NZONE – Zone ID (usually a mixture ID) in AMPX multigroup library (0\$\$ LD)

from which to add/update transitions (0).

If NZONE=0, the AMPX library must not contain zone IDs.

14. IEDOU – 1/0 – Edit/no edit of transition cross sections (0)

15. NFISW – Number of nuclides with fission yields (-1)

-1 fission yields included for all fissionable nuclides

0 no yields added

N input N nuclides with fission yields (Block2 7\$\$ Array)

16. NUMA – Number of title lines to copy from the input ORIGEN library (0\$\$ ND) to the output ORIGEN library (0\$\$ MD) (0).

18. NWGT – Flux spectrum source (0).

0 flux spectrum from AMPX multigroup library (IDREF)

N input N-group flux spectrum (Block2, 9** Array)

T – Block1 terminator.

Block2: nuclides with fission yields and weighting flux spectrum

This block is only read if in reaction update mode (Block1 `0\\$\\$` LBIN=0).

7\$\$ Array – Nuclide IDs with fission yields [Block1 1\$\$ NFISW entries]

9** Array – Weighting flux spectrum [Block1 1\$\$ NWGT entries]

The flux spectrum must be given in order of descending neutron energy according to the convention that group 1 is the highest energy group. The group structure (number of groups and group boundaries) must be consistent with the ORIGEN reaction resource (Block1 0\$\$ JD).

T – Block2 terminator.

Block3: array dimensions for decay library creation

This block is only read if in decay library creation mode (Block1 0\$\$ LBIN=1). The default values usually apply. The values are used only internally for memory allocation and may be set to a larger value than is required.

3\$\$ Array – Library constants

18. ITMAX – Total number of nuclides in library (2600)
19. ILMAX – Number of activation product nuclides (1000)

-1, omits light-element library

20. IAMAX – Number of actinide nuclides (200)

-1, omits actinide library

21. IFMAX —Number of fission-product nuclides (1400)

-1, omits fission-product library

22. IXMAX – Total number of decay transitions from one nuclide to another (40,000)

T – Block3 terminator.

Block6: number of user-defined transition coefficients

This block is only read if user-defined transition coefficients have been specified in decay library creation mode (Block1 1\$\$ LBUP=1).

15\$\$ Array – Number of user update nuclides

1. LBU – Total number of transitions to be entered in Block8 71\$\$, 72\$\$, and 73** Arrays (0)

T – Block6 terminator

Block8: user-defined transition coefficients

Block8 is only required only if a nonzero value is entered in the Block6 15\$\$ array. The three arrays (71\$\$, 72\$\$, and 73**) represent the parent, daughter, and coefficients for Block6 LBU user-defined transitions, or the quantity $f_{ij}\sigma_j$ for a given parent j and daughter i from Eq. (5.1.4).

71\$\$ Array – Parent Nuclides [LBU entries]

ISN1 – Parent Nuclide ID

72\$\$ Array – Daughter Nuclides or Reaction MT number [LBU entries]

ISN2 – Daughter Nuclide ID for the reaction product of the corresponding entry in ISN1

or reaction MT number

Note: The reaction transition will be added if it does not already exist by setting `Block1 1$$ JADD=1`. Otherwise, new transitions are omitted.

73** Array – Reaction Cross Sections [LBU entries].

SIGMA– Reaction cross section (in barns) for the reaction described by ISN1 and ISN2

There are two special rules to facilitate modifying fission cross sections σ_{fj} and removal cross sections σ_j.

if ISN1=ISN2, the removal cross section is set equal to the corresponding SIGMA. Note that this overrides the automatic calculation of the removal cross section as the sum of all transition cross sections.

if ISN1=-ISN2, the fission cross section is set equal to the corresponding SIGMA.

T -Block8 terminator.

This concludes the input for a single case in COUPLE. COUPLE allows for multiple cases in a single input and will automatically begin processing the next case's Block1 TITLE unless "DONE" (without quotes) is entered as the TITLE entry.

5.2 OBIWAN: A TOOL FOR VIEWING AND MANIPULATING ORIGEN BINARY FILES

S. E. Skutnik and W. A. Wieselquist

OBIWAN (**O**rigen **B**inary **I**nterrogation **W**ithout **A** scale i**N**put) is a command-line utility for viewing, manipulating, and converting ORIGEN binary output files.

OBIWAN can be called directly from the command line, from `${SCALE}/bin/obiwan`; in addition, it can be called within a SCALE input within a =*shell* block, i.e.:

Example 5.2.1: An OBIWAN command within a shell block

```
=shell
${PGMDIR}/obiwan [command] [file(s)]
end
...
```

OBIWAN supports the following modes:

- **convert** convert between versions of files
- **diff** diff/compare two files
- **info** print basic info about multiple files
- **interp** interpolate files
- **patch** patch data in files
- **tag** manipulate tags on a file
- **view** view a file

Each of these methods (including supported capabilities and syntax) are discussed in the following sections. OBIWAN also emits command-line help if you run it without any files/arguments.

5.2.1 OBIWAN INFO

Displays information about an on-disk binary Origen file.

Usage: `obiwan info [-type=TYPE] FILE`

`-type` `[f33 / hdf5 / f71 / yld ]`
Optionally specify OBIWAN to load only a particular file type

Options are:

- **f33**: Origen::Library
- **hdf5**: Origen::Archive
- **f71**: Origen::StateSet
- **yld**: Origen::YieldResource

By default, OBIWAN will attempt to infer the filetype by trial-and-error

`-verbose`
Verbose mode; prints extra output to STDERR

Table 5.2.1: File information output by `obiwan info`

Column	**Description**
fileName	path without the directory
dataType	type/class of the object
numSets	number of "sets" of data (e.g. burnups)
fileFormat	the format of the dataType in this file
fileType	"binary" or "text"
file/sysEndian	endian-ness for the file itself and the system (mismatches have I/O performance penalties)
appVersion	version of the application which wrote this file (some formats cannot store this and it appears as "-")
ms	time in milliseconds to load the file
directory	directory portion of path

5.2.2 OBIWAN VIEW

Displays detailed contents of ORIGEN results and data files.

Usage: `obiwan view [-type=TYPE -format=FORMAT options] FILE`

The `-type=TYPE` flag tells the loader to assume a data type for `FILE`. If not provided, trial and error is used.

`-type` `[f33 / hdf5 / f71 / yld ]`

- **f33**: *Origen::Library*

- **hdf5**: *Origen::Archive*
- **f71**: *Origen::StateSet*
- **yld**: *Origen::YieldResource*

The `-format=FORMAT` flag tells the printer to output data a certain way. If not provided, specific to the type of data.

5.2.2.1 Output formatting

`obiwan view` offers several options for formatting output which are specific to the file type being viewed (i.e., corresponding to `obiwan view -type`); these are discussed further within this section.

Note: All types and formats do not support all options.

Note: `True`/`False` values (boolean) should be specified as 1/0.

ORIGEN nuclide ID formatting

The most general form for an ORIGEN nuclide id is known as a *SIZZZAAA*, an 8-digit integer composed of four parts.

S - 1 digit for sublib integer
I - 1 digit for isomeric state
ZZZ - 3 digits of atomic number
AAA - 3 digits of mass number

Only the "sublib" is a non-standard nuclide parameter, specific to a design decision in ORIGEN where it is allowed to track two different versions of a nuclide, e.g. Gd-155 as both a fission product and fuel dopant. Below is a short description of the typical sublibs.

Table 5.2.2: Origen sublibrary definition

S	TY	DESCRIPTION
1	LT	light nuclides, typically the result of activation of common elements and structural materials, e.g. O-16, Li-6
2	AC	fissionable actinides and their immediate decay and activation products, e.g. U-235, Pu-239, Cm-244
3	FP	fission products and their decay and activation products, e.g. Te-135, Cs-135, Xe-135

When *S=0* (in most cases), this implies the sum over all versions of a nuclide.

Format options for Origen data resources

Available format options for Origen data resources (including `Origen::Library`, `Origen::Archive`, and `Origen::YieldResource`) are as-follows:

`-format`

- **coeff**: output tabular transition coefficients
- **coefft**: coeff but with transition ids
- **matrix**: output full transition matrix
- **json**: output in JSON format
- **tags**: output file tags
- **nucl**: nuclide list
- **nprodr**: neutron production ratio
- **genneu**: neutron generation
- **loxs**: loss xs
- **fisxs**: fission xs
- **burnups**: output burnup list
- **reaction**: allowed reactions
- **graphviz**: output dot file transitions to STDOUT

Default: `-format=coefft`

Format options for Origen concentrations files

`-format`

- **info**: information table
- **csv**: comma-separated values
- **ii.json**: inventory interface in JSON
- **tags**: tags stored on each state

Default: `-format=info`

5.2.2.2 Data filter options

`-pos=`

Position in `Origen::Library` or `Origen::StateSet` to output. (0 is all positions). < 0 means **normalize** output to that position (useful for viewing cross-section data across many scales).

Note: Not available for all types / formats.

Default `pos=1`

-cases=[...]
Only consider these cases (-type=f71 only)

-sizzzaaa=
Restrict data to this nuclide; e.g. u-235 is 20092235

allow_s=[...]
Allow only certain sublibs, e.g. -allow_s=[2,3] (allows only actinides and fission product sublibraries).

-allow_i=[...]
Allow only certain isomeric states, e.g. -allow_i=[1]

-allow_z=[...]
Allow only certain atomic numbers, e.g. -allow_z=[92]

-allow_a=[...]
Allow only certain mass numbers, e.g. -allow_a=[135,136]

-min_z=
Allow only atomic numbers greater or equal to Z (integer)

-tid=
Restrict data to this transition id:

dk mode (<0) or reaction MT (>0)

(use -format=coefft to see ids)

Note: Valid for -format=graphviz only

-allow_decay
Whether to display decay transitions (boolean)

Default: yes

-allow_fission=
Whether to display fission transitions (boolean)

Default: no

-allow_reaction=
Whether to display reaction transitions (boolean)

Default: yes

-include_byproducts=
Whether to include byproduct reactions arrows (boolean)

Default: no

5.2.2.3 Output control options

-prec=

Number of significant digits to output

Default: 4

-output_zeros=

Whether to output zero values (boolean)

Default: `yes`

-transpose=

Whether to transpose table (boolean)

Default: `no`

Note: Only available for some tables

-symbols=

Whether to output numeric ids (1001) or symbols (H-1)

Tip: Same as `-idform='{:TY}:{:ee}{:AAA}{:m}'`

-idform=

Controls printout of ids, the following are subsituted:

{:S} - sublib integer

{:TY} - sublib name

{:I} - isomeric state integer

{:ZZZ} - up to 3 digits of atomic number (also Z, ZZ)

{:AAA} - up to 3 digits of mass number (also A, AA)

{:EE} - up to 2 characters elemental symbol (also Ee,ee for different case)

{:M} - metastable indicator 'M' for first, 'M2' for second, etc. (also m)

Default:

- With `-symbols=0 -idform='{:S}{:I}{:ZZZ}{:AAA}'`
- With `-symbols=1 -idform='{:TY}:{:ee}{:AAA}{:m}'`

-units

Units used for printing

Note: Only applies to `-type=f71` and `-format=csv`

Available units:

- abso (total absorption xs)
- fiss (total fission xs)
- capt (total capture xs)
- airm (raditoxicity index in air, m^3)
- apel
- atom
- becq (activity in becquerels)
- curi (activity in curies)
- gamw (thermal output from gammas in watts)
- gamm
- gato (mass in gram-atoms)
- gper (isotopic mass percent by element)
- gram (mass in grams)
- h2om (radiotoxicity index in water, m^3)
- kilo (mass in kilograms)
- wpel (weight percent by element)
- watt (total thermal output in watts)
- mevs
- part
- inte
- ener

-verbose=

Display verbose output (includes echo of all input data; boolean)

Default: `-verbose=no`

5.2.2.4 Data manipulation options

-flux=

If `flux>=0`, then multiply reaction matrix coeff (one-group xs in barns) by the factor flux(n/cm^2s) * 1e-24(barns/cm) in order to display reactions/second units

---

Tip: Useful for comparing the magnitude of decay and reaction channels

---

Default: `flux=-1.0` (no flux scaling)

-lib=

Filepath to a binary Origen data library (.f33)

Tip: Needed for certain `Origen::StateSet` (.f71) unit conversions

See also:

`obiwan view -dec`

-dec=

Filepath to an on-disk Origen DecayResource (e.g., `origen.rev*.decay.data`)

Tip: Needed for certain `Origen::StateSet` (.f71) unit conversions

Default: Search for origen.rev*.decay.data in each of the following locations (in order):

1. `./obiwan/../data/origen_data/`
2. `${SCALE}/data/origen_data/`
3. `${DATA}/data/origen_data/`

See also:

`obiwan view -lib`

Note: Only one of `-lib` or `-dec` should be specified for F71 file unit conversions

---

5.2.2.5 Usage examples

Displaying a table of loss cross-section data

Note here that `pos=0` corresponds to all positions (burnups) on a library.

Example 5.2.2: Using OBIWAN to view loss cross-sections on a library

```
$ obiwan view -format=loxs -pos=0 ./a10_e15w01.bof 2>/dev/null

 nuclide  0.0000e+00  1.5000e+03  4.5000e+03  7.5000e+03  1.0500e+04  1.3500e+04
10001001  2.5724e-01  2.6172e-01  2.6315e-01  2.6400e-01  2.6582e-01  2.6731e-01
10001002  8.9254e-03  9.6637e-03  9.8350e-03  9.9177e-03  1.0278e-02  1.0590e-02
10001003  2.3025e-04  2.4988e-04  2.5428e-04  2.5635e-04  2.6587e-04  2.7415e-04
10001004  1.2445e+11  1.3507e+11  1.3744e+11  1.3856e+11  1.4371e+11  1.4818e+11
...
```

While the output can be piped through a command such as **`grep`** to filter on a specific nuclide, e.g.:

```
$ obiwan view -format=loxs -pos=0 ./a10_e15w01.bof | grep 20092235
```

in many cases, the transpose table (combined with a nuclide filter using `obiwan view -sizzzaaa` is typically a better option, such as shown below. (Note the normalization to the first position value).

Example 5.2.3: Using OBIWAN to view U-235 loss cross-sections, relative to the first burnup position

```
$ obiwan view -format=loxs -sizzzaaa=20092235 -pos=-1 -transpose=1 ./a10_e15w01.bof 2>/dev/null

         nuclide    20092235
      0.0000e+00  1.0000e+00
      1.5000e+03  9.9738e-01
      4.5000e+03  1.0177e+00
      7.5000e+03  1.0398e+00
      1.0500e+04  1.0466e+00
      1.3500e+04  1.0517e+00
      1.6500e+04  1.0560e+00
      3.1500e+04  1.0677e+00
      4.6500e+04  1.0709e+00
      5.8500e+04  1.0711e+00
      7.0500e+04  1.0707e+00
```

Output transitions to a graphical format

The following example demonstrates dumping a transition matrix set to a node-based graph (.dot) file; the **neato** program from **graphviz** works best for viewing this output.:

```
$ obiwan view -format=graphviz -allow_s="[1,2,3]" -include_byproducts=0 \
    $DATA/origen_library/pwr.rev03.orglib >transitions.dot

$ neato -n1 transitions.dot -T pdf >transitions.pdf
```

Output a JSON-based inventory interface file

The following example outputs a inventory interface file restricted to actinides and fission products:

```
$ obiwan view -format=ii.json -allow_s="[2,3]" my.f71 >my.ii.json
```

Here note that the *ii.json* output is normally directed to STDOUT; in this case, it is redirected to the file `my.ii.json`.

5.2.3 OBIWAN TAG

obiwan tag is used to add, modify, and delete "tags" on Origen binary files. These "tags" come in two types: "ID" tags used for categorical identification, and "interp" tags representing real values used to denote state points for interpolation. (Examples of typical "interpolation" tags might include features such as initial enrichment, average moderator density, etc.; i.e., characteristics that are expected to influence the shape of the neutron energy spectrum and thus the resulting one-group cross-sections.)

-idtags=

Key-value pairs corresponding to "categorical" information about a data set on a binary file. Each key is a string mapped to a string value; ID keys are used to identify common characteristics of different sets of data (e.g., assembly type, size, and other characteristic features used to identify and group together similar sets of data).

One or more values are given as key-value pairs within an array, e.g.:

`-idtags=[ tag1=val1, tag2=val2, ... ]`

-interptags=

Key-value pairs corresponding to "interpolable" information about the position on the binary file. Each

key is a string mapped to a real, floating-point point value; these interpolation values can then be used as "knots" on an interpolation grid to interpolate `Origen::Library` and `Origen::StateSet` data.

One or more values are given as an array, e.g.:

`-interptags=[ tag1=val1, tag2=val2, ... ]`

5.2.3.1 Viewing tags on on a file

To view tags present on an `Origen::Library`, call `obiwan tag [filename]` without specyfing any ID or interpolation tags, e.g.:

Example 5.2.4: Using OBIWAN to view all existing tags on a file

```
obiwan view libs/w17_15_tagged.arplib

    libs/w17_new.arplib: {
       idtags: {
          size: 17,
          vendor: Westinghouse }
       interptags: {
          enr: 1.5,    }
    }
```

Note: For `Origen::Archive`, `obiwan tag` viewing will display the "lowest common denominator" set of tags, i.e., exclusively the tags present on all `Library` objects on the `Archive`.

Note: **`obiwan tag`** modifies tags on an Origen binary file in-place.

5.2.3.2 Adding tags to a file

To add a tag to a file, specify the tag name and value as a key-value pair; multiple tags can be added simultaneously by separating each key-value pair by a comma. The entire series of key-value pairs must be enclosed by a pair of single quotes, e.g.:

```
$ obiwan tag idtags='tag1=val1, tag2=val2, ...'
```

If a tag name already exists on the file, the tag value is automatically updated to the new value provided by the user; otherwise, the tag is created and added to the `Library`.

Example 5.2.5: Adding tags to multiple `Origen::Library` files simultaneously; note that all output is directed to the file "log"

```
cp $DATA/arplibs/w17* new
obiwan tag new/\* -idtags='size=17,vendor=W' 2>log
obiwan tag new/\*e60.arplib -interptags='enr=6.0' 2>>log
obiwan tag new/\*e50.arplib -interptags='enr=5.0' 2>>log
obiwan tag new/\*e40.arplib -interptags='enr=4.0' 2>>log
obiwan tag new/\*e30.arplib -interptags='enr=3.0' 2>>log
obiwan tag new/\*e20.arplib -interptags='enr=2.0' 2>>log
obiwan tag new/\*e15.arplib -interptags='enr=1.5' 2>>log
```

The newly-created tags can be similarly viewed using a wildcard operator:

Example 5.2.6: Viewing newly-added tags on multiple files

```
$ obiwan tag new/\*.arplib

    new/w17_e15.arplib: {
       idtags: {
          size: 17,
          vendor: W   }
       interptags: {
          enr: 1.5,    }
    }

    new/w17_e20.arplib: {
       idtags: {
          size: 17,
          vendor: W   }
       interptags: {
          enr: 2,    }
    }
    ...
```

5.2.3.3 Deleting tags on a file

To delete a tag from a file, simply specify the name of an existing tag on the file with an empty value, e.g.:

```
$ obiwan tag -idtags='assembly='
```

5.2.4 OBIWAN CONVERT

Converts between different file formats for `Origen::Library` (reactor data libraries) and `Origen::StateSet` (stored concentrations).

-version=
Target output version (file format)

- **6.1**: sets -format=s61
- **6.2**: sets -format=bof

-format=
Target output format (mutually exclusive with `-version`)

- **s61**: SCALE 6.1 fortran binary
- **s62b**: SCALE 6.2 beta fortran binary
- **bof**: SCALE 6.2+ Binary Object Format
- **hdf5**: SCALE 6.3+ HDF5 format (EXPERIMENTAL)

Warning: The HDF5 format for `Origen::Library` is currently under development and may change in future releases; at present, it is provided as a convenience method for users to import `Origen::Library` data from external sources and to export data into HDF5 format for expedited viewing & processing.

-type=
Filter loading to a particular file type

- **f33**: `Origen::Library`
- **f71**: `Origen::StateSet`
- **yld**: `Origen::YieldResource`
- **dec**: `Origen::DecayResource`

-dir=
Existing directory for converted files

Default: `-dir=./` (current directory)

-i
Modify existing files in place

(By default, obiwan will not overwrite existing files)

-remove_mass
Remove masses on a library

-remove_abund
Remove abundances on a library

-verbose
Verbose mode (extra output to STDERR)

-endian=
Request for endianness of converted files

- **big**: big endian
- **little**: little endian
- otherwise: system-native endian

(will not be possible in all scenarios)

-thin=
Enable / disable library thinning (1 = thin)

"Thinning" library by interpolating to a user-specified series of burnups

Default: `thin=0` (no thinning)

interp=
Time / burnup interpolation method

near: Interpolate to nearest time / burnup point **lin**: Linearly interpolate between time / burnup points **cubic**: Cubic spline interpolaiton between time / burnup points

Default: `near` (nearest neighbor)

tvals=
List of times (MWd/MTHM) to keep/interpolate to in JSON array format, e.g. `tvals=[0,100,250,1000,50000]`

rvar=

Response variable to base thinning on: e.g., loss xs (*loxs*), fission xs (*fisxs*)

loxs: Total loss cross-section **fisxs**: Fission cross-section

Default: `rvar='loxs'`

rnuc=

Nuclide response to use for response variable (e.g., pu240) for interpolation / thinning

Default: `pu240`

-setbu=

Array of burnups to set on the library (i.e., burnup points to interpolate to)

Example: `-setbu=[1500.0, 3025.0, 10000.0]`

5.2.5 OBIWAN DIFF

Usage: `obiwan diff -format=String(y) [opts] file1 file2`

-reltol=

Relative tolerance in percent (%)

Default: `reltol=0.01`

-pos=

Position in file to output (0 corresponds to all positions)

Default: `-pos=1`

-format=

Controls output format

Options include:

- `y`: smart side-by-side
- `genneu`: neutron generation
- `eq`: fast exact match (no output)
- `approx_eq`: fast aprox match (no output)

Default: `-format=y` (smart side-by-side)

5.2.5.1 Obiwan diff examples

Take the difference between two files at the first file position, display differences using smart side-by-size, while redirecting STDERR to the file `log` and STDOUT to the file `out`:

```
$ obiwan diff -pos=1 -format=y \
              $DATA/arplibs/a10_e15w01.arplib \
              $DATA/arplibs/a10_e20w01.arplib 2>log >out
$ echo $? #return value should be 3 indicating difference
```

Same as before, but now allow for a 30% relative tolerance for reporting differences:

```
$ obiwan diff -pos=1 =format=y -reltol=30 \
              $DATA/arplibs/a10_e15w01.arplib \
              $DATA/arplibs/a10_e20w01.arplib 2>log >out
$ echo $? #return value should be 0 indicating no difference >30%
```

5.2.6 OBIWAN INTERP

obiwan interp interpolates between ORIGEN libraries based on a specified series of tag attributes to generate a new library.

Usage: `obiwan interp [-idtags='tag1=[v1],tag2=[v2]'] [-interptags='tag3=[v3],tag4=[v4]'] file1 file2 ...`

-idtags=

List of ID tags to down-select for interpolation. (i.e., Libraries without **all** of the matching ID tags are excluded from interpolation).

Accepts an array of key-value pairs separated by commas, e.g. `-idtags='tag1=[v1],tag2=[v2],...]`

Values are interpreted as strings

-interptags=

List of interpolable values to interpolate the new library to.

Accepts an array of key-value pairs separated by commas, e.g. `-interptags='tag1=[v1],tag2=[v2],...]`

Here, values are floating-point real numbers

Note: Required

5.2.6.1 Interpolation usage examples

Example 5.2.7 illustrates a basic interpolation case, selecting all libraries with ID tags `Assembly Type` matching `w17x17` and `Fuel Type` matching `UOX`, interpolating along dimensions `Enrichment` and `Moderator Density` to values of 4.5% and 0.55 g/cc, respectively.

Example 5.2.7: Interpolate all libraries with ID tags `Assembly Type=w17x17` and `Fuel Type=UOX` to an enrichment of 4.5% and moderator density of 0.55 g/cc.

```
obiwan interp
       -idtags='Assembly Type=w17x17,Fuel Type=UOX'
       -interptags='Enrichment=4.5,Moderator Density=0.55' /path/to/libraries
```

Note: When interpolating, obiwan will produce a new library file called `InterpLib.bof`.

5.2.7 OBIWAN PATCH

Patch an ORIGEN file with data from another file

Usage: `obiwan patch -from=SOURCE -data="path/to/data" [opts] file`

-from=

File to import patch data from; existing data or `NULL`

Default: `-from=ft33f001`

-data=

A special a hierarchical data description using "/" separators.

Data paths supported:

- `coeff[POS]/n_production_xs`
- `coeff[POS]/nu`

`POS=1,2,...,N` where `N` is the total number of positions

`POS=*` can be used to specify all positions

`POS=1|2|...` can be used to specify a list of positions

Note: `coeff/` and `coeff[*]/` are identical

-pos=

Position on <`file`> to apply patch

`0`: attempt to apply to all positions

-type=

Load only a particular type

Options:

- `f33`: Library

-verbose

Verbose mode (extra output to STDERR)

5.2.7.1 obiwan patch usage examples

The following examples illustrate how to use **`obiwan patch`** to patch data on Origen Library files.

Example 5.2.8 illustrates patching all neutron production cross-sections from the first position of an existing Library file.

Example 5.2.8: Patch all neutron production cross sections with a single data set (position 1 on `my.f33`)

```
$ obiwan patch -from=my.f33 -data=coeff[1]/n_production_xs patched.f33
$ obiwan view -format=genneu -type=f33 -pos=1 patched.f33
```

Example 5.2.9 nullifies (replaces with zeroes) all neutron production cross-sections as position 2 on the Library file.

Example 5.2.9: Zero out neutron production cross-sections at position #2

```
$ obiwan patch -from=NULL -data=coeff/n_production_xs -pos=2 patched.f33
```

Example 5.2.10 demonstrates replacing data on a three-position library, using data from source position 1 for the first two positions on the patched library and from source position 2 on the third position of the patched library.

Example 5.2.10: Patch a 3-position library with data from positons 1, 1, and 2 on `my.f33`, respectively

```
$ obiwan patch -from=my.f33 -data=coeff[1|1|2]/n_production_xs patched.f33
```

Example 5.2.11 and Example 5.2.11 demonstrate synonymous ways of sequentially replacing three positions on a patched library with the sequential positions from a source Library. (Note that the source library `my.f33` must have the same number of positions as the Library being patched.)

Example 5.2.11: Patch 3-position library with new data from `my.f33` in sequential order. (Note that `my.f33` must have 3 positions.)

```
$ obiwan patch -from=my.f33 -data=coeff[*]/n_production_xs patched.f33
```

Example 5.2.12: Patch 3-position library with new data from `my.f33` in sequential order. (Note that `my.f33` must have 3 positions.)

```
$ obiwan patch -from=my.f33 -data=coeff/n_production_xs patched.f33
```

Finally, Example 5.2.13 replaces only the nu portion (i.e., average neutron production per incident neutron) of the actinide neutron (i.e., average neutron production per incident neutron production data, assuming it is dominant).

Example 5.2.13: Just correct the nu portion of actinides production, assuming it is dominant, i.e. `nu=from_prodxs/from_fisxs --> to_prodxs=nu*to_fisxs`

```
$ obiwan patch -from=my.f33 -data=coeff/nu patched.f33
```

5.3 ORIGEN REACTOR LIBRARIES

B. Hiscox, B. R. Betzler, B. J. Ade

5.3.1 DESCRIPTION OF ORIGEN REACTOR LIBRARIES IN SCALE

The ARP code creates burnup-dependent ORIGEN cross-section libraries by interpolating over reactor cross-section libraries generated in advance using reactor physics transport methods. The reactor cross-section libraries distributed in SCALE include many classes of commercial power reactor designs and a range of fuel assembly designs.

Cross-section libraries suitable for use with ORIGEN are available in SCALE for the following reactor and fuel assembly designs, summarized in Table 5.3.1.

- BWR 7 × 7, 8 × 8-1, 8 × 8-2, 9 × 9-2, 9 × 9-9, 10 × 10-9, 10 × 10-8, SVEA-64, SVEA-96, and SVEA-100;
- PWR 14 × 14, 15 × 15, 16 × 16, 17 × 17, 18 × 18;
- CANDU reactor (19-, 28-, and 37-element bundle designs);
- Magnox graphite reactor;
- Advanced Gas-Cooled Reactor (AGR);
- VVER 440 and VVER 1000;

- RBMK;
- IRT;
- MOX BWR 7 × 7, 8 × 8-1, 8 × 8-2, 9 × 9-2, 9 × 9-9, 10 × 10-9, 10 × 10-8, SVEA-64, SVEA-96, and SVEA-100;
- MOX PWR 14 × 14, 15 × 15, 16 × 16, 17 × 17, 18 × 18.

All of the libraries distributed with SCALE were developed using ENDF/B-VII.1-based 252-group cross section libraries and the SCALE 6.3 TRITON module. The assembly design and the identifier for each library are summarized in Table 5.3.2. All TRITON templates used to generate these libraries are provided with the release; these templates have been developed to reflect the most recent publicly available data and to adhere to current SCALE best practices. MOX templates use the same set of BWR and PWR lattices, with MOX compositions replacing the UO:sub:*2* fuel.

Additionally, a Python script (`slig.py`) has been developed to semi-automate the process of generating sets of libraries from information in these templates. This script calls another Python script to automatically generate a PDF documentation file with LATEX. This SLIG script is made available to the user; more details are available in the SLIG manual.

Table 5.3.1: Summary of ORIGEN libraries (1470 total libraries).

Reactor type	**Assembly Design Description**	**Library Name(s)**
PWR LEU and MOX	Babcock & Wilcox 15 × 15	bw15x15, mox_bw15x15
	Siemens 14 × 14	s14x14, mox_s14x14
	Siemens 18 × 18	s18x18, mox_s18x18
	Westinghouse 14 × 14	w14x14, mox_w14x14
	Westinghouse 15 × 15	w15x15, mox_w15x15
	Westinghouse 17 × 17	w17x17, mox_w17x17
	Westinghouse 17 × 17-OFA	w17x17_ofa, mox_w17x17_ofa
	Westinghouse CE 14 × 14	ce14x14, mox_ce14x14
	Westinghouse CE 16 × 16	ce16x16, mox_ce16x16
BWR LEU and MOX	ABB 8 × 8-1	abb8x8-1, mox_abb8x8-1
	ATRIUM 9 × 9-9	atrium9x9-9, mox_atrium9x9-9
	ATRIUM 10 × 10-9	atrium10x10-9, mox_atrium10x10-9
	General Electric 7 × 7-0	ge7x7-0, mox_ge7x7-0
	General Electric 8 × 8-1	ge8x8-1, mox_ge8x8-1
	General Electric 8 × 8-2	ge8x8-2, mox_ge8x8-2
	General Electric 9 × 9-2	ge9x9-2, mox_ge9x9-2
	General Electric 10 × 10-8	ge10x10-8, mox_ge10x10-8
	SVEA 64(8 × 8-1)	svea64-1, mox_svea64-1
	SVEA 96(10 × 10-4)	svea96-0, mox_svea96-0
	SVEA 100(10 × 10-0)	svea100-0, mox_svea100-0
VVER	VVER-440 flat enrichment	vver440
	VVER-440 radial enr. profile, avg. 3.82%	vver440_3.82
	VVER-440 radial enr. profile, avg. 4.25%	vver440_4.25
	VVER-440 radial enr. profile, avg. 4.38%	vver440_4.38

continues on next page

Table 5.3.1 – continued from previous page

	VVER-1000	vver1000
RBMK	RBMK 1000, flat enrichment	rbmk1000
CANDU	19-element bundle design	candu19
	28-element bundle design	candu28
	37-element bundle design	candu37
Magnox	Magnox graphite reactor (Calder Hall)	magnox
AGR	Advanced gas-cooled reactor	agr
IRT	IRT-2M (3 tube)	irt2m3tube36enrich, irt2m3tube
	IRT-2M (4 tube)	irt2m4tube36enrich, irt2m4tube
	IRT-3M (6 tube)	irt3m6tube36enrich, irt3m6tube90enrich
	IRT-3M (8 tube)	irt3m8tube36enrich, irt3m8tube90enrich
	IRT-4M (6 tube)	irt4m6tube
	IRT-4M (8 tube)	irt4m8tube
LEU = Low-enriched uranium **MOX** = Mixed-oxide		

The libraries for BWR and RBMK assembly designs all include variable coolant density. Table 2 lists the number of cross-section files associated with each individual assembly design; this number is often a product of the number of enrichments (or Pu contents for MOX assemblies) and the number of coolant density values. Unless otherwise noted in Table 5.3.1, the PWR and BWR LEU libraries were generated for ten enrichment values: 0.5, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7,0, 8.0, and 8.5 wt% ^{235}U. The maximum enrichment and burnups of most of the BWR and PWR libraries were extended in SCALE 6.3 from their historical maximum values of 6.0 wt% enrichment and 70.5 GWd/MTIHM burnup to 8.5 wt% and 82.5 GWd/MTIHM, respectively.

The cross sections are represented at several burnup values in the range from 0 to the maximum burnup listed. The number of burnup values on the library has been optimized with a post-processing script to minimize the size of the data libraries without losing significant fidelity in the cross section data. Note that these optimized burnup steps are not the same burnup steps used in the TRITON calculations; more burnup steps are used to calculate the depletion of the fuel. The fuel assembly designations used for many of the BWR designs include the basic lattice type, followed optionally by the number of empty (non fuel) lattice positions. For example, the 9 × 9-9 designation refers to the Atrium design with a 9 × 9 assembly lattice with 9 non-fuel (water) locations. Additional data on the reactor libraries is available in the library information PDF file released with SCALE.

This document includes details on the fuel lattice type, the fuel vendor (where design information is specific to a vendor design), the assembly model, basic fuel design and reactor operating data, and the range of the variable parameters associated with the libraries (e.g., enrichment range, burnup range, etc.). This document references separate reports from which a more detailed description of the data for these libraries can be found.

Table 5.3.2: Additional information on ORIGEN libraries

Assembly type	Enrichments [wt. %^{235}U]	Coolant/moderator densities [g/cm^3]	Maximum burnup [GWd/MTIHM]	Number of Libraries
PWR LEU	0.5, 1.5, 2, 3, 4, 5, 6, 7, 8, 8.5	~0.73	82.5	10

continues on next page

Table 5.3.2 – continued from previous page

BWR LEU	0.5, 1.5, 2, 3, 4, 5, 6, 7, 8, 8.5	0.1, 0.3, 0.5, 0.7, 0.9	82.5	50
PWR MOX	*	~0.73	82.5	15
BWR MOX		0.1, 0.3, 0.5, 0.7, 0.9	82.5	75
AGR	0.5, 1.5, 2, 3, 4, 5	1.65	48.75	6
CANDU	0.711	0.8445	13.725	3
IRT	19.75, 36, 80, 90	0.989	159.3	12
Magnox	0.7, 0.8, 0.9, 1	1.628	13.725	4
RBMK	1.8, 2.2, 2.6, 3	0.15, 0.28, 0.41, 0.54, 0.67, 0.8	24.375	24
VVER-440	1.6, 2.4, 3.6, profiled	0.73	82.5	6
VVER-1000	0.5, 1.5, 2, 3, 4 5, 6, 7, 8, 8.5	0.7145	82.5	10
* **Pu contents [% heavy metal]:** 4, 7, 10; ^{239}**Pu contents [% of Pu]:** 50, 55, 60, 65, 70				

5.3.2 HOW TO GENERATE ORIGEN CROSS-SECTION LIBRARIES

This section describes the basic procedures one needs to follow to generate ORIGEN cross-section libraries for fuel types not included in SCALE. Note that the extension of the methods and data beyond 5 wt% enrichment has only been recently investigated by ORNL [ORIGEN-RX-LIBSCSHW21, ORIGEN-RX-LIBSHCWS21]. While no unexpected or anomalous trends were observed in trends beyond the limits of previous reactor data libraries (i.e., 5.0 wt% enrichment and 60 GWd/MTIHM), there is nonetheless a current lack of accessible validation data for these higher-burnup domains. In general, effects of higher enrichment were found to offset higher burnup limits. Users are encouraged to carefully test and validate applications of these libraries in these extended domains.

The following discussion and examples relate primarily to generating commercial LWR fuel libraries. Note that the parameter ranges used as examples are not necessarily appropriate to non-LWR applications, and will need to be modified for different reactor types and fuel designs.

The first step is to construct a physics model of the fuel lattice with the descriptions of the reactor assembly under consideration. For a given initial fuel enrichment, a TRITON depletion calculation is performed using one of the depletion analysis sequences of SCALE. TRITON can use either an explicit 2-D representation of the fuel assembly using the NEWT discrete ordinates transport code, or a 3-D representation of the assembly using the KENO Monte Carlo transport code. TRITON allows multiple fuel types to be defined and depleted independently.

The depletion analysis sequences are used to simulate irradiation and depletion of the fuel over the required irradiation history. A burnup analysis is typically modeled using a series of time intervals, or cycles. During the simulation, cross-sections that are representative of the mid-point of each cycle are created and saved in the library by the depletion sequence. In addition, cross sections that are representative of fresh fuel conditions (zero burnup) are now automatically saved in the library, without the need for the user to add a small burnup step to represent initial cross sections, as required in earlier versions of SCALE.

Past experience in creating LWR fuel libraries has indicated that depletion simulations with burnup steps of 3,000 MWd/MTU are generally adequate to represent the cross section variations with burnup. Each set of burnup-dependent cross sections is stored within the single library, and each set is accessed sequentially by its position in the library. The position 1 set contains fresh-fuel cross sections and the remaining cross-section

sets (positions) correspond to burnup levels characterizing the midpoint of each burnup step in the depletion sequence calculation.

For fuels with multiple enrichment values, the above procedure is repeated for each enrichment. For fuels with a single enrichment value (e.g., natural uranium), only one depletion case is required. Cross-section changes with enrichment are generally represented using approximately 1.0 wt % increments, e.g., 0.5, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, and 8.5 wt % of ^{235}U. Because the cross-section variation is generally well behaved and smooth with changes in enrichment, it may be possible to maintain accuracy using fewer enrichment points. For BWR designs, or reactor designs involving considerable moderator density variations, one has also to consider the effect of water density variation on the cross sections. The recommended water density values for BWRs are typically 0.1 to 0.9 g/cm^3, with increments of about 0.2 g/cm^3. Again, fewer points may be adequate for many applications. Water density variation has typically not been included for PWR libraries developed at ORNL because of the relatively small variation in pressurized water reactors. When the variation in moderator density is considered, the libraries must be calculated for each combination of enrichment and moderator density. This may involve a considerable number of depletion analysis simulations to prepare all of the required cross-section libraries.

The SLIG utility has been created to help automate the creation of input files that represent each of the discrete parameter combinations. Instead of having to manually create input files for each parameter value, a single template input file is constructed with generic flags substituted for the parameters that vary from case-to-case (i.e., enrichment and water density). The specific values required for each parameter are specified in the header of the template file; SLIG reads the template and substitutes the values for the generic flags in the input file. SLIG generates a unique subdirectory and input file for each parameter combination. A description of SLIG and examples are provided in the SLIG manual.

Another utility provides an optional procedure that can be used to reduce the size of the cross-section libraries by eliminating some of the burnup-dependent cross-section sets in the libraries in regimes where the cross sections do not vary appreciably with burnup. In general, the change in cross-section values is more pronounced early in irradiation, and tends to approach asymptotic values at higher burnup. Therefore, interpolation of the cross sections using fewer data points may yield acceptable accuracy in this range. This utility is currently unavailable in SCALE 6.3.

As an example of this procedure, the basic steps in creating a set of cross-section libraries for a typical PWR assembly are illustrated. In this example the burnup is calculated up to 72 GWd/MTU. To generate these basic cross-section libraries, the following steps are performed for each fuel enrichment value:

1. Create a 2-D or 3-D reactor physics model using TRITON for the fuel assembly design and fuel type being considered.

2. Run a depletion analysis calculation to the maximum burnup under consideration. For example, set a maximum burnup at 72 GWd/MTU, and simulate using 24 intervals (cycles) with one library per cycle. Each of these cycles is 3 GWd/MTU in size. The library generated by this analysis will contain 25 sets of cross sections: initial fresh fuel cross sections plus 24 burnup-dependent cross sections.

3. Include as many extra nuclides in the fresh fuel composition specification as is practical in trace concentrations (e.g., 1E-20 atoms/barn-cm). This procedure ensures that updated cross sections for each nuclide added will be used in creating the ORIGEN library. The depletion analysis sequences automatically add many of the most important actinides and fission products. However, the addition of more nuclides by the user will result in more complete updating of the cross-section data. The addition of trace nuclides is handled automatically in TRITON using the ADDNUX= parameter entry.

4. The above procedure is repeated for each enrichment value. If moderator density variation is to be included, a separate calculation is required for each enrichment-moderator density value combination.

5. Finally, the ARPDATA.TXT file must be updated to include the data for the new cross-section libraries created. This file contains information on the library, including the number of values for each variable parameter, the parameter values, the burnup values for each library position, and the actual library file names. The burnup values for each set of cross sections in the library are printed in the TRITON output file to assist the user in determining the appropriate library entries. Once the library information is registered in the ARPDATA.TXT file, it can be used directly by ORIGEN.

5.3.3 APPENDICES

5.3.3.1 SCALE/ORIGEN Library Generator (SLIG)

B. R. Betzler

This Python script (`slig.py`) semi-automates the process of generating sets of cross-section libraries for ORIGEN calculations for spent fuel depletion, decay, and source term analysis. SLIG performs several tasks:

1. reads standard-format template files, obtaining information (e.g, enrichments, burnups, moderator densities, etc.) from the file header
2. generates a set of input files according to this header information;
3. builds a directory tree to house these input files;
4. writes the addition to the `arpdata.txt` file;
5. moves the final libraries to a new directory; and
6. reads the burnup list for each set of libraries and writes them in the addition to the `arpdata.txt` file.

This manual is divided into five main sections based on the desired application:

Quick Start Directions for running SCALE on a local machine or a computing cluster without a queue system;

Advanced Options for running SCALE on a cluster with a queue system;

Template File Rules and Examples for editing and writing template files;

Troubleshooting for issues running SLIG; and

Code Information for making changes to the code.

Quick Start Directions

The SLIG package is located in the SCALE source directory in `./packages/etc/slig/`. Because it is necessary to run SCALE to generate the appropriate cross-section libraries, SLIG runs in a disjointed, three-step process:

1. Copy the contents of the `./slig/src/` directory into `./slig/testing/`. Move to the `./slig/testing/` directory and use **`./slig.py -g`** to perform tasks 1) through 4). If SLIG returns an error, follow the directions in the error message (see Sect. 5.3.3.1.7). After a successful run, three new items will appear in the current directory:

runSpace/ the directory tree containing the input files,

arpLibList.txt a list of the locations of the libraries in the directory tree, and

addToArpData.txt additional lines for the arpdata.txt file.

2. Run SCALE with each input within the directory tree (runSpace/). Each input is located within the directory tree in the location ./runSpace/templateName/inputID/, where templateName is the prefix of the template file (i.e., templateName_template.inp) and inputID is a string identifying the enrichment (eNN), moderator density (wMM, if applicable), and plutonium vector (vPP, if applicable) for the input.
3. After all SCALE calculations are complete, use **./slig.py -f** to perform task 5) and 6). SLIG will identify any missing libraries. After a successful run, two changes will be made in the current directory:

   newLibraries/ this directory containing the new libraries will appear, and

   addToArpData.txt the burnup lists will be written on the this file.

The final result is an addition to the arpdata.txt file (addToArpData.txt) and a directory containing all generated libraries (newLibraries/). Do not delete any files associated with slig.py unless it directs you to do so. The following options are available to the user (see Sect. 5.3.3.1.1 for more options):

-p <PATH>, **-path**=<PATH>
: This specifies the location of the template files. SLIG will search PATH and all its subdirectories for template files. The default path is the current directory.

-x <XSLIB>, **-xsections**=<XSLIB>
: This specifies the cross section library used in all calculations. The default is v7-252.

Advanced Options

The following are more advanced options available to the user:

-e <EXT>, **-extension**=<EXT>
: This specifies the file extension that identifies a template file. By default, SLIG searches for files ending in _template.inp).

-a, -add
: This flag is used with the -g option (**./slig.py -ga**) to add input files to the current directory tree (runSpace/). It also adds library information to addToArpData.txt and adds the location of the libraries in the directory tree to file:*arpLibList.txt*.

  This flag is used with the -f option (**./slig.py -fa**) to add newly generated libraries to the current library directory (newLibraries/).

-d, -document
: This flag turns on documentation routines that automatically generate a LaTeX file (libraryInformation.tex) with data about each template file using information on the file header. This also generates a .pdf file (libraryInformation.pdf) using pdflatex (required for this to function properly). Additionally, any figures referenced within the templates must be located within the current directory or any of its subdirectories.

-s, -submit
: This flag turns on routines that generate a PBS submit script for each input file for running on clusters with a queue system. It also generates a shell script to submit these jobs (with **qsub**). The PBS submit script template (run1proc.pbs) must be placed in the current directory.

The first two steps of the three-step process for running SLIG (see Sect. 5.3.3.1.1) change:

1. Use **./slig.py -gs** to perform tasks 1) through 4). After a successful run, four new items will appear in the current directory:

   runSpace/ the directory tree containing the input files,

   arpLibList.txt a list of the locations of the libraries in the directory tree,

   addToArpData.txt additional lines for the arpdata.txt file, and

   submitSLIGjobs a shell script for submitting jobs to the queue.

2. Run the shell script (`./submitSLIGjobs`) to submit jobs to the queue. Note that this may submit a very large number of jobs to the queue.

When using **./slig.py -ga**, SLIG will check for documentation and submit scripts and will proceed with the same settings that were used for the initial run of SLIG. For example, if the user initially runs **./slig.py -gds** to generate documentation and submit scripts then later uses **./slig.py -ga**, SLIG will proceed with adding documentation for the additional templates and generating submit scripts. When SLIG generates additional submit scripts, a separate numbered file is generated for each new template (e.g., `submitSLIGjobs1`).

Template File Rules and Examples

Example templates are provided with the SCALE distribution. For creating new templates, it is best to use these templates as starter files and make incremental changes as necessary for the new application. The header at the top of each template file contains the information that will be used to generate input files and documentation. Each line of this header must start with an apostrophe; each line is a comment line according to the SCALE standard input. There are two main sections in this header:

1. The **template header** (see Example 5.3.1 and Example 5.3.2). This header has a parameter list and an option list:
   a. The *parameter list* identifies the strings within the template file that SLIG will replace as it generates input files. These parameters are found throughout the template file; SLIG will replace these with values as it generates each input files. SLIG identifies the parameter list starting from the first line containing the 'parameter' string and each line thereafter that is indented more than five blank spaces (not including the apostrophe). The following is a list of rules for entering parameters:
      - Each parameter is entered on a separate line in the form "parameter-description".
      - Each parameter must be a unique string of characters.
      - Each parameter is a single string without spaces.
      - SLIG uses the "description" to characterize the parameter. If these descriptions are changed, SLIG may not be able to properly characterize each parameter. Thus, it is suggested that the descriptions in the following list are not changed.
      - There is some flexibility in changing the parameter strings.

   b. The *option list* identifies the quantities or values that SLIG uses to determine the values that will replace the parameters in each input file. SLIG identifies the option list starting from the first line containing the 'option' string and each line thereafter that is indented more than five blank spaces (not including the apostrophe).

      The following is a list of rules for entering options:

- Each option is entered on a separate line in the form `option - values`.
- SLIG uses the "option" to recognize what to do with the values. If this is changed, SLIG may not be able to properly characterize each option. Thus, it is suggested that the options (to the left of the dash) in the following list are not changed.
- For options with multiple values, each value must be separated with a comma.
- For values to continue onto the next line, the last value in a line must end with a comma and the next line must be indented and must begin with the next value.
- The values may be changed.

2. The **documentation header** (see Example 5.3.3). The documentation header has two purposes. First, it is used to track information about the template (e.g., author, date created, and methodology) and the source documents that contributed to the template's creation. This simplifies updating input files for new versions of SCALE. Second, SLIG uses this header information to generate documentation files. SLIG identifies the documentation header starting from the first line containing 'Documentation' to the line starting with an apostrophe followed by a space and five dashes (`' -----`).

Example 5.3.1: A typical BWR template header.

```
' ------------------------------------------------------------------
'  template to generate libraries for ORIGEN-S
'  parameters are: u235wt%       - wt% U235
'                  u234wt%       - wt% U234
'                  u236wt%       - wt% U236
'                  u238wt%       - wt% U238
'                  ddd           - coolant density (g/cc)
'                  dancoff1      - dancoff factor 1
'                  dancoff2      - dancoff factor 2
'                  namelibrary   - name of generated ORIGEN library
'                  specpow       - average specific power
'                  daystoburn    - depletion interval in days
'  options are:    name          - g10_
'                  enrichment    - 0.5, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0
'                  cool. density - 0.1, 0.3, 0.5, 0.7, 0.9
'                  dancoff1      - 0.5041, 0.3937, 0.3182, 0.2631,
'                                  0.2211
'                  dancoff2      - 0.3229, 0.2541, 0.2069, 0.1722,
'                                  0.1455
'                  spec. power   - 25.0
' ------------------------------------------------------------------
```

Example 5.3.2: A typical BWR MOX template header.

```
' ------------------------------------------------------------------
'  template to generate libraries for ORIGEN-S
'  parameters are: IcontentPu    - wt% plutonium: inner
'                  IcontentU     - wt% uranium: inner
'                  IEcontentPu   - wt% plutonium: inside edge
'                  IEcontentU    - wt% uranium: inside edge
'                  EcontentPu    - wt% plutonium: edge
'                  EcontentU     - wt% uranium: edge
'                  CcontentPu    - wt% plutonium: corner
'                  CcontentU     - wt% uranium: corner
'                  pu238wt%      - wt% Pu238
'                  pu239wt%      - wt% Pu239
'                  pu240wt%      - wt% Pu240
'                  pu241wt%      - wt% Pu241
```

(continues on next page)

(continued from previous page)

```
'                  pu242wt%      - wt% Pu242
'                  densityAm     - americium density (g/cc)
'                  ddd           - coolant density (g/cc)
'                  dancoff1      - dancoff factor 1
'                  dancoff2      - dancoff factor 2
'                  namelibrary   - name of generated ORIGEN library
'                  specpow       - average specific power
'                  daystoburn    - depletion interval in days
'  options are:    name          - mox_g10_
'                  pu content    - 4.0, 7.0, 10.0
'                  pu vector     - 50.0, 55.0, 60.0, 65.0, 70.0
'                  cool. density - 0.1, 0.3, 0.5, 0.7, 0.9
'                  dancoff1      - 0.5041, 0.3937, 0.3182, 0.2631,
'                                  0.2211
'                  dancoff2      - 0.3229, 0.2541, 0.2069, 0.1722,
'                                  0.1455
'                  spec. power   - 25.0
'                  burnups       - 0, 1, 2, 3, 4.5, 6, 7.5, 9,
'                                  10.5, 12, 13.5, 15, 16.5, 18,
'                                  19.5, 21, 24, 27, 30, 33, 36,
'                                  39, 42, 45, 48, 51, 54, 57, 60,
'                                  63, 66, 69, 72
'                  pin_zone      - 26, 16, 24, 12
'                  pin_gad       - 14
'                  avg_pin_dens. - 10.4
' ------------------------------------------------------------------
```

Example 5.3.3: A typical MOX BWR documentation header.

```
' ------------------------------------------------------------------
' Documentation and Notes (empty fields are auto-populated):
'  [Change Log]
'    Rev 0: Generated by J. Doe |
'    Rev 1: Generated by B. R. Betzler, June 2014 |
'    Rev 2: Generated by B. R. Betzler, September 2015
'  [Author(s)] B. R. Betzler
'  [SCALE Version] SCALE 6.2
'  [Reactor Type] Mixed Oxide Boiling Water Reactor General Electric 10x10-8
'  [Model Info] 2D t-depl full assembly model (see Figure \ref{fi:mox_ge10x10-8}), xsLib cross-section library
'  [Sources]
'    1. B. J. Ade, ``Generation of Collapsed Cross Sections for Hatch 1 Cycles 1-3 and Generation of Generic BWR␣
↪Reflector Cross Sections", ORNL/LTR-2012/559, Oak Ridge National Laboratory, 2012. |
'    2. H. Smith, J. Peterson, and J. Hu, ``Fuel Assembly Modeling for the Modeling and Simulation Toolset",␣
↪ORNL/LTR-2012-555 Rev. 1, Oak Ridge National Laboratory, 2013. |
'    3. I. C. Gauld, ``MOX Cross-Section Libraries for ORIGEN-ARP", ORNL/TM-2003/2, Oak Ridge National Laboratory,
↪ Oak Ridge, Tennessee, 2003. |
'    4. U. Mertyurek and I. C. Gauld, ``Development of ORIGEN Libraries for Mixed Oxide (MOX) Fuel Assembly␣
↪Designs", to be published, 2015.
'    5. H. Smith, J. Peterson, and J. Hu, ``Fuel Assembly Modeling for the Modeling and Simulation Toolset",␣
↪ORNL/LTR-2012-555 Rev. 1, Oak Ridge National Laboratory, 2013.
'  [Data Range]
'  [Libraries]
'  [Power]
'  [Other Info]
'    Channel box data, fuel/gap/channel moderator densities, and temperatures from Reference 1.
'    All other dimensions, materials, etc. from Reference 2.
'    Gad layout altered according to best engineering judgement.
'    MOX isotopic vector information from Reference 3.
'    MOX zoning pattern from section 4.1 of Reference 4 (see Table 2, Eq.~3, and Eq.~4).
'    Specific power from Reference 5.
'  figure{mox_ge10x10-8.pdf: MOX BWR GE 10x10-8.}
' ------------------------------------------------------------------
```

UOX fuel parameters

The following (Example 5.3.4) is a list of some of the parameters available to the user for working with most PWR and BWR lattices:

Example 5.3.4: SLIG input options for UOX-based templates

```
u235wt% - wt% U235
u234wt% - wt% U234
u236wt% - wt% U236
u238wt% - wt% U238

ddd - coolant density (g/cc)

dancoff1 - dancoff factor 1
dancoff2 - dancoff factor 2

namelibrary - name of generated ORIGEN library

specpow - average specific power
daystoburn - depletion interval in days
```

Example 5.3.5: Example of a fully-specified UOX input.

```
name - abb_

enrichment - 0.5, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0

cool. density - 0.1, 0.3, 0.5, 0.7, 0.9 (or mod. density - 1.65)

dancoff1 - 0.4686, 0.3429, 0.2651, 0.2122, 0.1742
dancoff2 - 0.3103, 0.2316, 0.1823, 0.1484, 0.1237

spec. power - 25.0

burnups - 0, 1, 2, 3, 4.5, 6, 7.5, 9,
          10.5, 12, 13.5, 15, 16.5, 18,
          19.5, 21, 24, 27, 30, 33, 36,
          39, 42, 45, 48, 51, 54, 57, 60,
          63, 66, 69, 72
```

MOX fuel parameters

For mixed oxide (MOX) PWR and BWR assemblies, there is a different set of parameters available to the user. SLIG will recognize a file as a MOX template by searching for the Pu-239 parameter. SLIG automatically generates a MOX zoning pattern (see Fig. 5.3.1) for BWR and PWR lattices according to the values of `pin_zone`, `pin_gad`, and `avg_pin_dens.`.

Example 5.3.6: SLIG input options for MOX-based templates.

```
IcontentPu - wt% plutonium: inner
IcontentU - wt% uranium: inner

IEcontentPu - wt% plutonium: inside edge
IEcontentU - wt% uranium: inside edge

EcontentPu - wt% plutonium: edge
EcontentU - wt% uranium: edge
```

(continues on next page)

(continued from previous page)

```
CcontentPu - wt% plutonium: corner
CcontentU - wt% uranium: corner

pu238wt% - wt% Pu238
pu239wt% - wt% Pu239
pu240wt% - wt% Pu240
pu241wt% - wt% Pu241
pu242wt% - wt% Pu242

densityAm - americium density (g/cc)

ddd - coolant density (g/cc)

dancoff1 - dancoff factor 1
dancoff2 - dancoff factor 2

namelibrary - name of generated ORIGEN library

specpow - average specific power

daystoburn - depletion interval in days
```

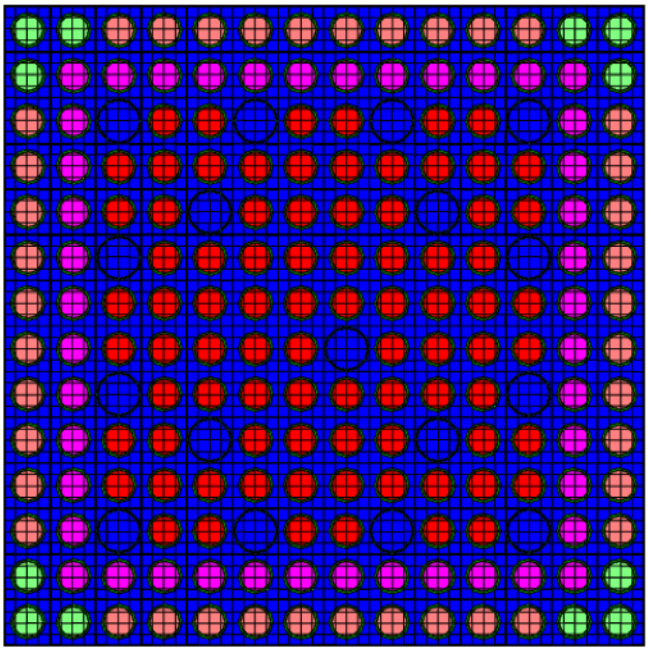


Fig. 5.3.1: MOX zoning layout for a Westinghouse 14 × 14 PWR assembly showing the corner (green), edge (salmon), inside edge (magenta), and inner (red) pin zones.

Example 5.3.7: Example of a fully-specified MOX input.

```
name - mox_abb_

pu content - 4.0, 7.0, 10.0
pu vector - 50.0, 55.0, 60.0, 65.0, 70.0

cool. density - 0.1, 0.3, 0.5, 0.7, 0.9
```

(continues on next page)

(continued from previous page)

```
dancoff1 - 0.4686, 0.3429, 0.2651, 0.2122, 0.1742
dancoff2 - 0.3103, 0.2316, 0.1823, 0.1484, 0.1237

spec. power - 25.0

burnups - 0, 1, 2, 3, 4.5, 6, 7.5, 9,
          10.5, 12, 13.5, 15, 16.5, 18,
          19.5, 21, 24, 27, 30, 33, 36,
          39, 42, 45, 48, 51, 54, 57, 60,
          63, 66, 69, 72

pin_zone - 11, 20, 16, 12

pin_gad - 4

avg_pin_dens. - 10.4
```

The following is an explanation of each option:

name
: This is the prefix for the name of the final generated libraries. It should be unique to the template (i.e., other templates should have a different name).

enrichment
: This is a list of the ^{235}U enrichments [%] for which SLIG will create separate inputs.

cool. density, mod. density
: This is a list of the coolant or moderator densities [g/cm^3] for which SLIG will create separate inputs.

dancoffN
: This is a list of dancoff factors that correspond to pin N. The length of this list must match the length of the coolant/moderator density list (the dancoff factor varies significantly with this density).

spec. power
: This is the specific power [MW/MTU] of the assembly.

burnups
: This is a list of cumulative burnup steps [GWd/MTU] that will be used to create the depletion steps. If a small step is not included, SLIG will print a warning and automatically insert one at the beginning of the calculation (for Xe equilibrium).

pu content
: This is a MOX-specific parameter that is a list of the average content of plutonium [%] in the assembly for which SLIG will create separate inputs. These quantities are calculated as a percentage of the total heavy metal loading in the assembly.

pu vector
: This is a MOX-specific parameter that is a list of the enrichments of ^{239}Pu [%] for which SLIG will create separate inputs. These quantities are calculated as a percentage of the total plutonium in the assembly and are used to calculate the entire plutonium vector [ORIGEN-Gau03].

pin_zone
: This is a MOX-specific parameter that lists of the number of pins in the inner, inside edge, edge, and corner zones (see Figure 5.A.3). This varies for each assembly.

pin_gad
: This is a MOX-specific parameter that specifies the number of Gd-bearing pins in the assembly. SLIG uses this quantity to calculate ensure that the specified Pu content of the assembly is correctly represented [ORIGEN-Gau03].

avg_pin_dens.
: This is a MOX-specific parameter that specifies the average density [g/cm^3] of the pins in the assembly. SLIG uses this quantity to calculate ensure that the specified Pu content of the assembly is correctly represented.

Documentation header

The documentation header has a free form-style entry, where sections are specified by `[Section Name]` and the text for each section follows afterward. Figures are referenced as `figure{figureName.pdf: Figure caption.}`. Any reference to the figure within the header must be in LaTeX-like form; for this, the figure is labeled as `fi:figureName`. The section name keywords `[Data Range]`, `[Libraries]`, and `[Power]` are reserved to be auto-populated by SLIG according to information provided within the vars list. All information within this header is transferred to the documentation file.

The parameters in the file headers are located within the template file in the appropriate locations. See the example templates for the proper usage. The burndata card in the input file should only be three lines:

```
read burndata
power=specpow burn=daystoburn down=0 end
end burndata
```

because it will be populated by the burnup points listed in the file header. There should also be a shell command at the end of the input file to save the cross-section library:

```
=shell
  cp ft33f001.cmbined $RTNDIR/namelibrary
end
```

Troubleshooting

If at any time an error occurs, SLIG will exit after printing the reason for the error and offer a possible solution. Following these instructions should resolve most issues with SLIG. The following is a list of factors to consider when having trouble running SLIG.

- SLIG uses the module argparse.py (see https://docs.python.org/3/library/argparse.html) to handle command line arguments; SLIG will crash if it does not have access to it. For versions of Python that do not inherently support this module, download the `argparse.py` file and include it in the directory with `slig.py`.
- If SLIG is crashing, there may be an issue with the version of Python. SLIG was developed to be functional with Python 2.7.6.
- After running **`./slig.py -g`**, the `addToArpData.txt` file will have dummy placeholders instead of the burnup lists. These placeholders are replaced with the burnup lists that are read off of the SCALE output files. The burnup lists in the documentation file are the burnup lists specified in the template files.

Code Information

SLIG consists of a `main`, one SCALE-specific class (`manageTemplate`), and three generic classes (`messenger`, `manageDirectory`, and `manageFile`):

- `main` reads and validates arguments, and contains a loop to make inputs and collect libraries
- `manageTemplate` manages templates (reads/sorts options, calculates concentrations)
- `messenger` manages all prints to the screen
- `manageDirectory` manages external directories (searching, making, etc.)
- `manageFile` manages external files (reading, writing, searching, copying, etc.)

SLIG calls a separate Python script (`collectinfov04.py`) to perform documentation functions. This script uses a template (`basedoc.v04.tex`) to generate the documentation file.

5.4 ORIGAMI: A CODE FOR COMPUTING ASSEMBLY ISOTOPICS WITH ORIGEN

ORIGAMI computes detailed isotopic compositions for light water reactor assemblies containing UO_2 fuel by using the ORIGEN transmutation code with pre-generated ORIGEN libraries, for a specified assembly power distribution. The fuel may be modeled using either lumped or pinwise representations with the option of including axial zones. In either case, ORIGAMI performs ORIGEN burnup calculations for each of the specified power regions to obtain the spatial distribution of isotopes in the burned fuel. Multiple cycles with varying burn-times and downtimes may be used. ORIGAMI produces several types of output files, including one containing stacked ORIGEN binary output data ("ft71 file") for each depletion zone; files with nuclide concentrations at the last time-step for each axial depletion region, in the format of SCALE standard composition input data or as MCNP material cards; a file containing the axial decay heat at the final time-step; and gamma and neutron radiation source spectra.

5.4.1 ACKNOWLEDGMENTS

ORIGAMI is based on the PinDeplete code developed by Steve Skutnik and also includes techniques taken from the Orella code written by Ian Gauld (retired). Support for development of ORIGAMI was provided by the U.S. Department of Energy, Office of Nuclear Energy, Nuclear Fuels Storage and Transportation Planning Project.

5.4.2 INTRODUCTION

ORIGAMI (**ORIG**EN **A**sse**m**bly **I**sotopics) provides the capability to perform isotopic depletion and decay calculations for a light water reactor fuel assembly model using one or more ORIGEN calculations. The assembly may be modeled using either lumped or pinwise representations with the option of including axial zones. ORIGAMI automates the performance of ORIGEN depletion calculations for each region and thus simulates zero-, one-, two-, and three -dimensional (0D, 1D, 2D, and 3D) modeling of a fuel assembly. Multiple cycles with different specific powers and exposure and decay times may be treated, and the power distribution is described in terms of fractional pin powers in the XY plane and axial distributions along the Z axis, which define the burnup regions for the ORIGEN computations. ORIGAMI allows for easy and flexible material composition specification through the standard SCALE mixture processor for composition input, the same as in TRITON (see XSPROC chapter). While ORIGAMI cannot presently treat axially non-uniform lattice features (e.g. axially varying enrichment or the partial-length rods found in many boiling water reactor designs) within a single input, these problems can still be easily modeled by splitting the problem across sequential ORIGAMI input cases residing on the same file.

The ORIGEN calculations performed by ORIGAMI use the methodology originally established for the SCALE sequence ORIGEN-ARP (see the ARP section). This approach provides an efficient mechanism to perform stand-alone reactor depletion calculations using pre-generated ORIGEN libraries which contain self-shielded, collapsed one-group cross sections as a function of selected independent variables, such as burnup, enrichment, and moderator density, for different reactor systems. Typically the data in these libraries are obtained from 2D, multigroup lattice transport calculations (e.g., TRITON) coupled with depletion calculations for burnup. The library cross sections may be flux-weighted over the lattice to obtain data representative of the entire homogenized assembly for lumped depletion; or alternatively, it is also possible to generate multiple ORIGEN libraries corresponding to individual or groups of pins within the lattice for multi-pin depletion. The burnup-dependent ORIGEN libraries are analogous to the parameterized cross section data produced by lattice physics codes for reactor core simulators, except that data for many more nuclides and reactions are included to allow ORIGEN to compute detailed isotopics for more than 2200 nuclides.

ORIGAMI extends the capabilities previously provided by ORIGEN-ARP to perform a suite of ORIGEN calculations in order to represent the isotopic distribution of fuel within an assembly in more detail. The pre-generated ORIGEN libraries provided with SCALE tabulate the assembly-average one-group cross sections, in order to accurately reproduce assembly-average isotopics. When performing pin-by-pin calculations in ORIGAMI, users can increase the fidelity with respect to proximity to features such as assembly edges, water holes, burnable poison rods, etc. by creating and employing zone-specific libraries for different pins. By specifying the individual library assignments for each pin, users can capture these local spectrum changes in the ORIGAMI calculation through the use of one-group libraries based on these local conditions. Currently, the specification of individual libraries is limited to pin-level specification only (i.e., the same library is used for all axial zones corresponding to a pin for 3D cases) with an allowed axial moderator density distribution and radial and axial power distributions.

ORIGAMI can produce the following output files in addition to the standard ORIGEN output for each depletion zone:

- isotopics in ORIGEN binary concentration (ft71) files
  - in each depletion zone at times specified by the **`options`** block, `ft71` key
  - in each axial zone (summed over all pins at a particular axial level) at the final time;
- nuclide concentrations by axial zone, written as a SCALE "standard composition block" that can be used directly as input for SCALE transport codes such as the KENO Monte Carlo criticality code;
- axially-dependent decay heat source for input to a thermal analysis code such as COBRA, so that the temperature distribution within a storage cask can be computed;
- nuclide concentrations for each axial zone, given in the format of MCNP material cards;
- space-dependent radiation source energy spectra and magnitudes in a simple text file.

ORIGAMI is tightly integrated with the SCALE Graphical User Interface, Fulcrum. Using Fulcrum and the "UO2 express form (configurable)", one can create a simple UO_2 assembly depletion case in seconds (see Fig. 5.4.1). Finally, ORIGAMI has the ability to perform the depletion/decay calculations for each zone in parallel using the MPI (Message Passing Interface), however this requires a special SCALE installation built with MPI in order to do so [ORIGAMISHDLG13].

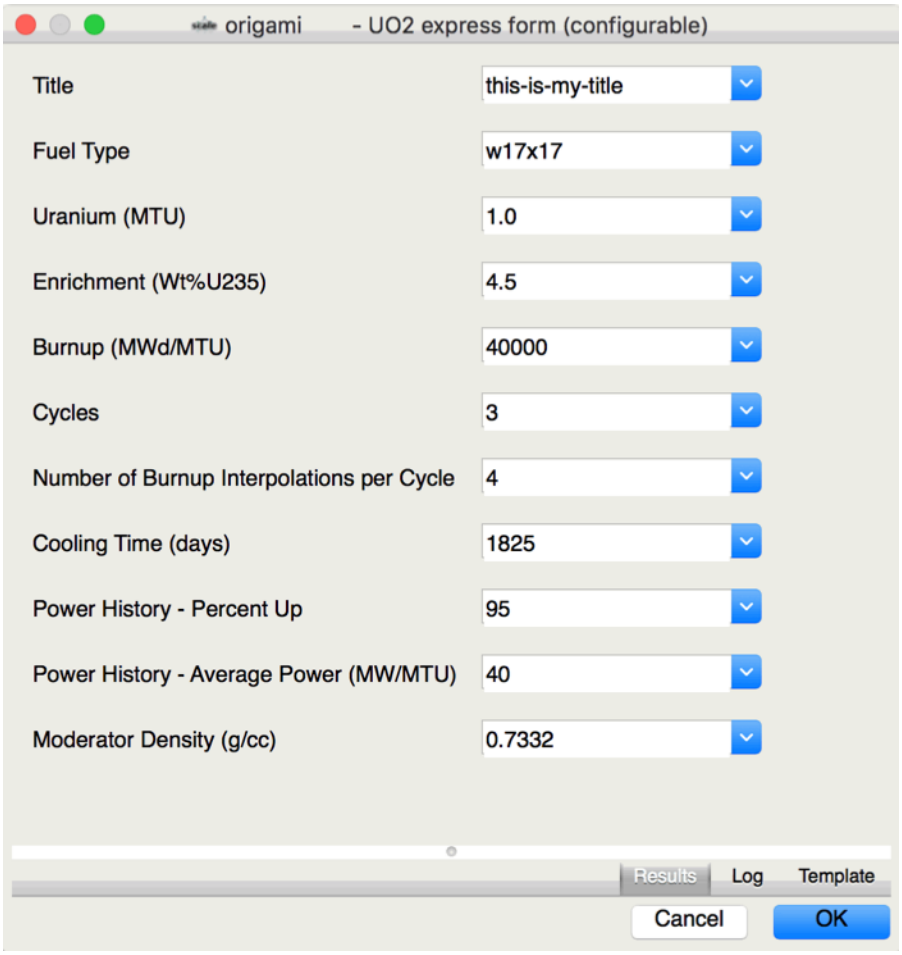


Fig. 5.4.1: Fulcrum UO_2 express form for creating ORIGAMI input.

5.4.3 COMPUTATIONAL METHODS

5.4.3.1 ORIGAMI assembly model

The basic model for ORIGAMI is a fuel assembly, which may be modeled in several ways with varying degrees of complexity. The most primitive model represents the assembly materials as a single mass lump that is depleted using the value of the specific power input in the power-history block. In this case, a single ORIGEN calculation is performed to obtain isotopics representing the entire assembly. This 0D model is equivalent to the current ORIGEN-ARP procedure. A more detailed model applies an input axial power profile to the (radially) lumped assembly materials. This lumped axial depletion model produces a 1D axially varying burnup distribution, but no allowance is made for variations in the relative pin powers within the assembly. Thus, if the axial power distribution is defined by N_Z axial zones, ORIGEN calculations are performed for N_Z different depletion regions. The 1D axial depletion model has been found to be adequate for most criticality and decay heat analysis of spent fuel assemblies [ORIGAMIRGlW12]. Note that both the 0D and 1D modes are fully consistent with the 2D TRITON calculations used to generate ORIGEN reactor libraries distributed with SCALE, in that these modes employ spatially-homogenized cross-sections to represent assembly-averaged flux and cross-sections. For 2D and 3D depletion models (wherein individual pin-specific libraries may optionally be specified), the user is advised that the ORIGEN reactor data libraries distributed with SCALE are representative of an assembly axial plane as a whole; in as much, the user is

advised to generate their own zone-specific libraries (i.e., based on individual material zones) within TRITON if they wish to capture regional neutronic effects within the assembly (such as proximity to water holes, burnable absorbers, etc.)

By specifying a radial pin-power map, a 2D or 3D calculation may be performed. Currently the axial and radial power shapes are fixed for the entire calculation but do still result in a fully 3D isotopic distribution [ORIGAMISGRT12, ORIGAMISHDLG13]. If there are N_P pins in the assembly and each has N_Z axial zones, ORIGAMI will perform ORIGEN calculations for $N_P \times N_Z$ depletion regions. For example, an assembly with a 17×17 array with 264 fuel pins and $N_Z = 24$ axial zones requires 6336 independent ORIGEN calculations. For these types of simulations, the parallel mode with MPI is highly recommended.

5.4.3.2 Definition of initial composition

The initial mass in metric tons of heavy metal is M_{mtu}, set by the input parameter `mtu`. The default value of `mtu` is equal to 1.0, so that by default the ORIGEN calculations are performed on the basis of "per metric ton of heavy metal". Given that the sum over all zones must have the total heavy metal content (M_{mtu}), one arrives at zone-wise heavy metal masses of:

$$M_{xy,z} = \frac{M_{\text{mtu}}}{\sum_{z=1}^{N_z} f_z \sum_{xy=1}^{N_P} m_{xy}} \cdot f_z \cdot m_{xy} \tag{5.4.1}$$

where the relative amount of heavy metal in each radial position, :m_{xy}, is calculated from the mixture specification; fractional axial height, f_z, from the zone specification; and N_P and N_Z are the total number of fuel pins and axial zones, respectively. Note that some pin locations in the assembly may not contain fuel, and these are not included in the value of N_P. The fractional axial height is given

$f_z = \frac{\Delta Z}{Z_{\text{tot}}}$ is the fraction of the active fuel height occupied by axial zone Z ΔZ is the length of axial zone

Z_{tot} is the total length of the active fuel.

Whenever an axial zone mesh is input (with array `meshz`), the value of f_Z is computed from the values of the zone boundaries (see input description in Sect. 5.4.4.6). If an axial mesh array is not input, the axial zones are assumed to be uniformly distributed. In this case, the axial zones all have the same height, so that $f = \frac{1}{N_Z}$, where N_Z is the number of uniform axial zones in the assembly.

The uranium mass in a single axial zone for all N_P fuel pins in the assembly (M_z) is thus:

$$M_Z = N_P \times M_{XY,Z} = M_{\text{mtu}} \times f_Z \tag{5.4.2}$$

In addition to the fuel mixture in an assembly, non-fuel materials (e.g., structural materials) may also be present. These materials contribute to the overall power production due to the energy produced by neutron capture reactions.. For a given value of the total assembly power, this reduces the power from the fuel mass and thus may slightly alter the fuel burnup and isotopics. In addition, activation of non-fuel materials produces additional radiation source terms in the spent fuel, which contribute to the decay heat and activity. Therefore ORIGAMI provides an option for including the non-fuel elements in the input array, `nonfuel`. The units of the non-fuel element masses are kg per MTU, and the materials are distributed uniformly within all fuel depletion zones. Note that the input non-fuel materials should not include oxygen in UO_2 if UO_2 is specified as the fuel material, as oxygen is already included in proportion to the uranium mass basis. Finally, because ORIGAMI accesses the StdComp library, any SCALE StdComp composition, e.g. "zirc4" for reactor cladding material Zircaloy-4, may be used in either structural or fuel materials.

5.4.3.3 Restart cases

ORIGAMI also allows the initial nuclide concentrations to be obtained from a previously produced ORIGEN binary output file. A restart case is indicated by setting **restart=yes** in the parameter array. The restart file has the name `assembly_restart.f71` and must be copied (or linked) to the SCALE temporary directory used for calculations. The restart file is normally obtained from an earlier ORIGAMI calculation, which always produces an ORIGEN restart file named `$`*OUTBASENAME*`.assm.f71`, where `${OUTBASENAME}` is an output prefix defined by the name of the input file and any user-specified prefix with the **prefix** key. Generally the restart file from ORIGAMI contains stacked concentrations, corresponding to each axial zone and then a final entry for the lumped assembly concentrations; hence, the initial composition for a restart case varies with axial zone, unlike the case for fresh fuel. ORIGAMI does not currently allow pin-dependent restart calculations. A restart case may be useful for performing decay-only calculations of spent fuel inventory, using the burned fuel composition previously computed for the assembly exposure during reactor operation. For decay-only cases, a value for the input parameter `nz` must be input in order to indicate the number of axial depletion regions in the previous burnup calculation.

5.4.3.4 Definition of power distribution

The radial power distribution is defined by the XY fractional pin powers in the input array `pxy`, and the axial fractional powers in the input array `pz`. The input values in arrays `pxy` and `pz` are normalized to unity by the code. The fractional power for a fuel pin "XY" is designated here to be r_{XY}, with the normalization $\sum_{XY=1}^{N_P} r_{XY}$. Similarly the fractional axial power for an axial zone Z is a_Z, which is normalized to $\sum_{Z=1}^{N_Z} a_Z$. The shapes of both the radial XY and axial Z distributions must be obtained prior to the ORIGAMI calculation, either from neutron transport calculations or experimental measurements. The input distributions remain constant during the ORIGEN burn calculations for all cycles; but in reality, the power distributions may vary with time—for example, the initial axial power distribution tends to flatten after a period of burnup since the higher power zones deplete the fuel faster. For this reason it is strongly recommended to use the relative burnup distribution (at final discharge) rather than the relative power density distribution for the input values. The burnup shape corresponds to the shape of the time-averaged flux distribution during the exposure period. This ensures that the final burnup distribution matches the desired shape.

For a given cycle, the assembly-specific power $P^{(SP)}$ is equal to the value of input variable `power`, read in the power-history block (see Sect. 5.4.4.4). The assembly-specific power has units of megawatts per MTU (MW/MTU). Therefore, the total power produced by the fuel assembly is:

$$P_{\text{tot}} = P_A^{(\text{SP})} \cdot M_{\text{mtu}} \tag{5.4.3}$$

where P_{tot} is the assembly total power, and $P_A^{(SP)}$ is the specific power for the assembly, read from input.

The absolute power (MW) in fuel pin "XY" is:

$$P_P = P_{\text{tot}} \cdot r_{xy} = P_A^{(\text{SP})} \cdot M_{\text{mtu}} \cdot r_{xy} \tag{5.4.4}$$

and the power produced in axial zone Z of this fuel pin XY is:

$$P_{XY,Z} = P_{\text{tot}} \cdot r_{xy} \cdot a_Z = P_A^{(\text{SP})} \cdot M_{\text{mtu}} \cdot r_{xy} \cdot a_Z \tag{5.4.5}$$

The absolute power produced in a single axial zone Z for all pins is:

$$P_Z = \sum_{XY=1}^{N_P} P_{XY,Z} = P_{\text{tot}} \times a_Z = P_A^{(\text{SP})} \times M_{\text{mtu}} \times a_Z \tag{5.4.6}$$

The ORIGEN depletion calculations are performed with the absolute powers defined in Eq. (5.4.4) and Eq. (5.4.6) for each depletion region in the 2D/3D pin-wise or 0D/1D axial depletion models, respectively. However, cross sections in the ORIGEN libraries are parameterized as a function of burnup, which depends on the specific power rather than absolute power for a given depletion region. The specific power (MW/MTU) in axial zone Z of pin XY is equal to:

$$P_{XY,Z}^{(SP)} = \frac{P_{XY,Z}}{M_{XY,Z}} \tag{5.4.7}$$

Substituting Eq. (5.4.1) and Eq. (5.4.5) into Eq. (5.4.7) gives:

$$P_{XY,Z}^{(SP)} = \frac{P_A^{(SP)} \cdot r_{\text{xy}} \cdot a_Z \cdot N_P}{f_Z} \tag{5.4.8}$$

In a similar manner, it can be shown that the specific power for all fuel pins in axial plane Z is:

$$P_Z^{(SP)} = \frac{P_A^{(SP)} \cdot a_Z}{f_Z} \tag{5.4.9}$$

ORIGAMI permits two modes for user-specified power distributions along the axial and radial meshes: *absolute* fractions (i.e., where powers along the axial mesh points are expressed as fractions of the total assembly power in MW) and *relative* normalization (i.e., in which *specific powers*— in MW/MTU — of axial zones are expressed as a relative modifiers of the assembly specific powers input in the power history block). Relative power shape modifiers assume that the specific powers expressed in the power history block represent the *average* assembly specific power(s) thus, ORIGAMI will convert these factors into axial & pin power *fractions* — i.e., the factors r_{xy} and a_{z} found in Eq. (5.4.4) and Eq. (5.4.6) used to calculate the absolute pin power and axial zone power, respectively. The conversion from *relative* specific power modifiers to *absolute* power fractions is accomplished through the following normalization procedure Eq. (5.4.10):

$$(a_Z)_i = \frac{(R_Z)_i \cdot M_{\text{MTU}} \cdot \left(\frac{\Delta Z}{Z_{\text{tot}}}\right)_i}{\sum (R_Z)_i \cdot M_{\text{MTU}} \cdot \left(\frac{\Delta Z}{Z_{\text{tot}}}\right)_i} = \frac{(R_Z)_i \cdot \left(\frac{\Delta Z}{Z_{\text{tot}}}\right)_i}{\sum (R_Z)_i \cdot \left(\frac{\Delta Z}{Z_{\text{tot}}}\right)_i} \tag{5.4.10}$$

where $(a_Z)_i$ is the axial power fraction for axial zone i and $(R_Z)_i$ is the relative axial zone specific power modifier for axial zone i. Obviously, for a uniformly-spaced axial mesh, the conversion from relative specific powers (using relative power modifiers) is precisely the same as that for absolute fractional axial zone powers; i.e., the relative power modifiers simply become axial power fractions by virtue of the fact that the term $\left(\frac{\Delta Z}{Z_{\text{tot}}}\right)_i$ becomes a constant, thereby reducing Eq. (5.4.10) back to a direct calculation of the fractional axial power based on a relative power modifier following normalization.

Because it is assumed that the assembly mass is uniformly distributed across the pins, it can similarly be shown that the use of relative power modifiers for the XY pin map $(r_{XY})_i$ will always produce the same result as using pre-normalized absolute fractional powers in the pin map, i.e. Eq. (5.4.11):

$$(r_{XY})_i = \frac{(R_{XY})_i \cdot \frac{M_{\text{MTU}}}{N_P}}{\sum (R_{XY})_i \cdot \frac{M_{\text{MTU}}}{N_P}} = \frac{(R_{XY})_i}{\sum (R_{XY})_i} \tag{5.4.11}$$

This option is provided as the `relnorm` option in the parameters block (discussed further in Sect. 5.4.4.2). The motivation for providing an alternative normalization for axial power shape factors is twofold. First, it is

generally assumed that information on the axial power shape is obtained from axial measurements relative to an assembly-average value (i.e., axial gamma scans to determine the burnup profile based upon the gross gamma intensity or isotopic ratios of burnup indicators such as ^{134}Cs / ^{137}Cs, etc.). Therefore, by using the relative normalization option (i.e., treating axial power shape factors as *relative* modifiers of the assembly specific power), users can directly input shape factors obtained from techniques such as non-destructive analysis (NDA) fuel measurements into ORIGAMI to model assembly isotopic distributions.

The second motivation for the relative normalization option comes from potential problems that can arise if treating axial power shape factors as absolute fractional powers (**`relnorm=no`**) in conjunction with non-uniform axial mesh spacing defined by the user in the **`z`** array (see Sect. 5.4.4.7 for details).

Important: If using the **`relnorm=no`** option, the fractional axial powers **must** be consistent with the axial mesh sizes defined or else **incorrect** zone-specific powers will result from Eq. (5.4.9), therefore leading to incorrect results and likely causing the ARP sequence to fail (and therefore the ORIGAMI calculation to halt) due to calculated burnup values for the depletion zone being out of the library range.

Users are thus **strongly cautioned** when using absolute fractional axial powers (**`relnorm=no`**) to ensure proper consistency between the axial power fractions and the axial mesh sizes.

For this reason, relative power shape factor normalization is **turned on** by default (**`relnorm=yes`**).

5.4.3.5 Computation of neutron and gamma energy spectra

ORIGAMI includes an option to generate multi-group neutron and gamma source spectra due to radioactive decay, for each depletion zone. Multi-group values are calculated by binning the discrete line and continuum spectra produced by radioactive decay and nuclear reactions into arbitrary energy group structures defined by user input. Whenever neutron energy group boundaries are input in array `ngrp`, neutron source spectra due to spontaneous fission, delayed neutron emission, and (α, n) reactions are calculated.

Similarly, gamma source spectra are computed if gamma energy group bounds are input in array `ggrp`. The gamma source includes photons produced by all types of radioactive decays, and also may include bremsstrahlung radiation produced by beta interactions. Input options can specify the type of nuclides included in the source term (i.e., light elements, actinides, fission products, or all nuclides), and the materials used for (α, n) reactions and bremsstrahlung production. If source spectra are calculated, the values are always included in the ORIGEN output ft71 binary file; and optionally the source spectra may also be output in a text file. The source text file only includes the average over all pins for each axial zone, while the ft71 file includes sources for all pins and axial zones.

The source spectra output by ORIGAMI are calculated in ORIGEN using the expression outlined in Eq. (5.4.12):

$$S_{\mathrm{Z.g}}^{(p)} = \sum_{i=1}^{\mathrm{itot}} Y_{i,g}^{(p)} \lambda_i \frac{M_Z^{(i)}}{A^{(i)}} \cdot N_A \tag{5.4.12}$$

where

$S_{\mathrm{Z.g}}^{(p)}$ = source spectrum (p/s) in energy group g for particles of type p and axial zone Z;

$Y_{i,g}^{(p)}$ = number of particles of type p emitted per decay of nuclide i; with energy in group g;

$M_Z^{(i)}$ = mass (g) of nuclide i in axial zone Z, obtained from ORIGEN calculation;

N_A = Avogadro's number (number atoms of nuclide i per mole);

$A^{(i)}$ = mass (g) of 1 mole of nuclide i;

λ_i = decay constant (s^{-1}) for nuclide i,

itot = total number of nuclides in burned fuel.

More details on the ORIGEN calculation of the source spectra can be found in the ORIGEN section (Sect. 5.1.5.2.7) of the SCALE documentation.

5.4.4 ORIGAMI INPUT DESCRIPTION

ORIGAMI uses free-form, keyword-driven input with the SCALE Object Notation (SON) syntax also used for ORIGEN input, and is described in more detail there. The general outline of ORIGAMI input is as follows.

(a) Case Identifier

(b) Options

(c) Fuel Composition

(d) Power-History

(e) Source-Options

(f) Output-Print Options

(g) Input Data

The above input data may be entered in any order. Data blocks and parameters which are not needed, or for which default values are desired, can be omitted. Example 5.4.1 provides a template containing all of the ORIGAMI input data blocks and arrays, with example values assigned. Note that much of the information shown in the template is optional, and typically is not needed for many cases. The following subsections provide a more detailed description of the input.

Example 5.4.1: Template for ORIGAMI input data

```
=origami
% Case identifier information
  title= 'input template example'
  prefix= example
  asmid=1
% Parameter options
  options{
    pitch= 19.718
    mtu= 0.4
    decayheat=yes
    fracnf=0.08
    nburn=15
    ndecay=12
    temper=300.0
    stdcomp=yes
    restart=no
    interp=spline
    output=cycle
    ft71=all
  }
% Array containing ORIGEN library names
  libs=[ ce14x14 ce16x16 ]
```

(continues on next page)

(continued from previous page)

```
% Fuel Composition
  fuelcomp{
    uox(fuel1){ enrich=3.21 }
    uox(fuel2){ enrich=3.50 }
    uox(fuel3){ enrich=2.80 }
    mix(1){ comps[ fuel1=98.2 Gd2O3=1.8 ] }
    mix(2){ comps[ fuel2=100 ] }
    mix(3){ comps[ fuel2=97.5 Gd2O3=2.5 ] }
    mix(4){ comps[ fuel3=96.9 Gd2O3=3.1 ] }
  }
% Map ORIGEN library names to XY pin layout
  libmap=[ 1 2
           2 1 ]
% Map individual compositions XY pin layout
  compmap=[ 1 2
            3 4 ]
% XY relative power distribution (code renormalizes to unity)
  pxy=[ 0.2 0.3
        0.4 0.5 ]
% Z-axial relative power distribution (code renormalizes to unity)
  pz=[ 0.6 0.4 ]
% Axial interval boundaries (for MTU mass distribution & plotting)
  meshz=[ 0.0 15.0 30.0 ]
% Non-fuel nuclides distributed within fuel material
  nonfuel=[ cr=3.366 mn=0.1525 fe=6.309 co=0.0302
            ni=2.366 zr=516.3 sn=8.412 gd=2.860 ]
% Axial variation of moderator density fraction
  modz=[ 0.73 0.715 ]
% Irradiation/decay information
  hist[
    cycle{ power=35.0 burn=200.0 nlib=7 down=50.0 }
  ]
% Optional neutron/gamma source information
  ggrp=[ 10.0e6 2.0e6 1.0e6 0.5e6 0.01 ]
  ngrp=[ 20.0e6 1.0e6 1.0e5 1.0e4 1.0e3 10.0 0.01 ]
  srcopt{ sublib=ac brem_medium=uo2 alphan_medium=case print=yes }
% Output edit options
  print{
    nuc{sublibs=[lt ac] total=no units=[grams] }
  }
% Nuclides included in comp file (OPTIONAL: overrides default)
  nuccomp=[
     92232 92233 92234 92235 92236 92237 92238 92239 92240
     92241 93235 93236 93237 93238 93239 94236 94237 94238
     94239 94240 94241 94242 94243 94244 94246 95241 95242
     95243 95244 95246 96241 96242 96243 96244 96245 96246
     96247 96248 96249 96250 97249 97250 98249 98250 98251
     98252 98253 98254 99253 99254 99255
  ]
end
```

5.4.4.1 Case and identifier information

ORIGAMI has three optional identifiers for the case. The `title` is included as a descriptor in the printed output file. The character string `prefix` is added to the front of the output file names described in Sect. 5.4.5 and in Table 5.4.8. Finally, the integer variable `asmid` is an arbitrary assembly identifier used in defining mixture numbers in the SCALE standard composition output file. Eq. (5.4.13) in Sect. 5.4.5.1 describes how the mixture ID is determined.

`title`= <string>

Title (up to 50 characters) describing the case. Enclosed in quotes if using embedded blanks.

(**Default:** none)

prefix= <string>
Prefix (up to 16 characters) to append to output file names.

(**Default:** none)

asmid= <integer>
Integer used to identify mixture ID in generated SCALE standard composition block [see Eq. (5.4.13)].

(**Default:** 1)

Table 5.4.1: Keywords for case identifier :class: longtable :widths: 13 65 12 :name: tab-origami-id-kw

Keyword	Description	Default
title=	up to 50 characters describing the case title, quoted if embedded blanks	blank
prefix=	up to 16 characters (no embedded blanks) appended to output file names	blank
asmid=	integer used to identify mixture ID in generated SCALE standard composition block [see Eq. (5.4.13)]	1

5.4.4.2 Options block

The options block has the following form:

options {... keyword blocks ...}
The options block allows the user to control problem features such as the total mass basis (mtu), non-fuel mass (fracnf), axial power normalization (relnorm), exercise fine-grained control over depletion calculations (solver, interp, option:*nburn*, ndecay), perform restart calculations from a prior ORIGAMI run (restart), specify the number of axial zones (nz), specify optional parameters used for visualization and post-processing (pitch, temper, fdens), and control which outputs to generate (small, mcnp, stdcomp, decayheat).

Each of the allowable parameter keywords is explained below. An example parameter block would be:

```
options{ stdcomp=yes decayheat=yes }
```

mtu= <number>
Metric tons of heavy metal in the assembly.

(**Default:** *1.0*)

fracnf= <number>
Total non-fuel mass in the assembly, given as a fraction of the heavy metal mass defined in mtu.

(**Default:** *none*)

nz= <integer>
Number of axial intervals. If not input, nz is equal to the number of entries in the input axial power array pz.

Required for decay-only restarts.

(**Default:** Determined by code via pz)

nburn= <integer>
Number of substeps used in ORIGEN burn calculations

(**Default:** 10)

ndecay= <integer>
Number of substeps used in ORIGEN decay calculations

(**Default:** 10)

pitch= <real number>
Assembly pitch (cm), if > 0.0. Only used to define XY mesh in viewing results. If this parameter is input, array pxy must also be entered.

(**Default:** 0.0)

temper= <real number>
Temperature (in degrees Kelvin) for mixtures output to the SCALE Standard Composition output (compBlock).

(**Default:** 293.0)

fdens= <real number>
Fuel density in g/cm^3.

(**Default:** 10.4)

offsetz= <integer>
Axial numbering offset; used for sequential ORIGAMI cases to uniquely identify axial zones (i.e., such as when using sequential cases to modify changing axial geometry).

(**Default:** 0)

relnorm= <yes | no>
Normalization of axial power shaping factors (pz) to be used

no → axial power shape factors treated as absolute fractions (does not normalize all axial burnups to 1.0)

yes → axial power shape factors treated as relative modifiers of assembly specific power (i.e., power= entries in the power history block)

(**Default:** *yes*)

mcnp= <yes | no>
Generate MCNP input stubs containing data on material concentrations and/or gamma and neutron emissions for each depletion node in the problem.

(**Default:** *yes*)

stdcomp= <yes | no>
Generate a text-based standard composition file containing burnup-credit nuclide number densities for each axial zone.

(**Default:** *no*)

decayheat= <yes | no>

Produce a decay heat file containing decay powers (in W) for each axial zone.

(**Default:** *no*)

restart= <yes | no>
Perform a restart calculation using initial compositions from a previously-generated ORIGEN ft71 file.

(**Default:** *no*)

solver= <matrex | cram>
Use the standard ("MATREX") solver or the Chebyshev Rational Approximation Method (CRAM) solver.

(**Default:** *matrex*)

small= <yes | no>
keep .out file small by suppressing all spectra and concentrations output except for lumped, assembly-averaged concentrations and spectra

---

Note: Full results are still written to other relevant files

(**Default:** *no*)

interp= <lagrange | spline>
Method for interpolating cross sections in ARP; Lagrangian polynomial (*lagrange* or monotonic cubic spline *spline*)

(**Default:** *lagrange*)

ft71=<last,cycle,all>, **output**=<last,cycle,all>
Controls output of saved / printed output concentrations.

last saves / prints results only for the substeps in last step of the last cycle (**default**)

cycle saves results for substeps in the last irradiation and decay steps in every cycle

all saves results for all substeps of all irradiation and decay steps in every cycle

(**Default:** last)

Table 5.4.2: Keywords in ORIGAMI options

Keyword	Description	Default
mtu=	Metric tons of heavy metal in the assembly	1.0
fracnf=	Total non-fuel mass in assembly, given as fraction of heavy metal mass defined by input *mtu*= . See description of input array nonfuel	none
nz=	Number of axial intervals. If not input, *nz* is equal to the number of entries in the input axial power array pz. Required for decay-only restarts.	Determined by code
nburn=	Number of substeps used in ORIGEN burn calculations	10
ndecay=	Number of substeps used in ORIGEN decay calculations	10
pitch=	Assembly pitch (cm), if > 0.0. Only used to define XY mesh in viewing results. If this parameter is input, array pxy must also be entered.	0.0
temper=	Temperature (Kelvin) assigned to materials in standard composition file	293.0

continues on next page

Table 5.4.2 – continued from previous page

Keyword	Description	Default
offsetz=	Axial numbering offset; used for sequential ORIGAMI cases to uniquely identify axial zones (i.e., such as when using sequential cases to modify changing axial geometry). [integer]	0
relnorm=	Normalization of axial power shaping factors (`pz`) to be used **no:** axial power shape factors treated as absolute fractions (does not normalize all axial burnups to 1.00) **yes:** axial power shape factors treated as **relative** modifiers of assembly specific power (i.e., *power*= entries in the power history block) [yes/no]	Yes
mcnp=	no/yes → do not / do generate an MCNP material and gamma/neutron file	Yes
stdcomp=	no/yes → do not / do generate a standard composition file containing burnup-credit nuclide number densities for each axial zone.	No
decay-heat=	no/yes → do not / do produce a decay heat file containing decay powers (in W) for each axial zone.	No
restart=	no/yes → do not / do restart using initial compositions from a previously-generated ORIGEN ft71 file.	No
solver=	matrex/cram → use the standard ("MATREX") solver or the Chebyshev Rational Approximation Method (CRAM) solver.	Matrex
small=	no/yes → keep .out file small by suppressing all spectra and concentrations output except for lumped, assembly-averaged concentrations and spectra (**Note:** all results are still written to other relevant files).	No
interp=	lagrange/spline → method for interpolating cross sections in ARP	Lagrange
output=	last/cycle/all → time steps for output print edits	Last
ft71=	last/cycle/all → time steps included in output ft71 file	Last

Additional notes on input parameters:

(a) `pitch` is only used for visualization of the results, and may be omitted if this is not of interest;

(b) `mtu` is discussed in Sect. 5.4.4.2

(c) `nz` is not required except decay-only restart cases; it must equal the number of entries in the array `pz`;

(d) `nburn` and `ndecay` are discussed in Sect. 5.4.4.4;

(e) `fracnf` is discussed in Sect. 5.4.4.7, where the input array of non-fuel materials is described;

(f) `relnorm` is discussed in Sect. 5.4.3.4, in the definition of the assembly power distribution;

(g) `stdcomp`, `fdens`, and `temper` are discussed in Sect. 5.4.5;

(h) `offsetz` is an optional feature designed to allow for ORIGAMI cases to be split across multiple inputs to capture axially-dependent features (such as partial-length rods); its use is discussed in further detail in the context of output generation in Sect. 5.4.5;

(i) `decayheat` is discussed in Sect. 5.4.5.3;

(j) `restart` is discussed in Sect. 5.4.3.3.

(k) `output`, `ft71`, are discussed in Sect. 5.4.3.4.

5.4.4.3 Fuel composition block

The purpose of the `fuelcomp` block is to create a set of mixtures (via the **`mix`** blocks inside) to specify the pin-wise distribution of initial isotopics. The example below, defines three mixtures (with IDs 1, 2, and 3); these are referenced in the `compmap` array for this 2x2 array of fuel pins.

`fuelcomp``= { mixture blocks }`
: Specifies fuel mixtures to be used by ORIGAMI in the `compmap` array. *Numbered* `mix` blocks are used by `compmap`, which can be composed of other named mixtures.

`mix``= { SCALE standard composition }`
: Mixture blocks identify specific pin-wise composition to be used by ORIGAMI, using the standard SCALE mixture composition syntax. Mixtures must be given an integer identifier (e.g., `mix(1)`, `mix(2)`, etc.)

`compmap``= [ mixture IDs ]`
: Specifies the distribution of fuel compositions / mixtures for each pin for 2-D and 3-D depletion cases. Mixture ID numbers correspond to those in the `fuelcomp` block.

  Required if `libmap` is explicitly specified beyond one element.

  (**Default:** `[1]`)

Example 5.4.2: Example specification of uranium oxide-based fuel mixtures in ORIGAMI, including 1) Mixed urania-gadolina fuel, 2) 4% enriched UO_2 fuel, and 3) 2% enriched UO:sub:2 fuel.

```
fuelcomp{
   uox(fuel_3pct){ enrich=3.20 dens=10.42 }
   uox(fuel_4pct){ enrich=4.00 dens=10.45 }
   uox(fuel_2pct){ enrich=2.10 dens=10.43 }
   mix(1){ comps[ fuel_3pct=99.0 Gd2O3=1.0 ] }
   mix(2){ comps[ fuel_4pct=100] }
   mix(3){ comps[ fuel_2pct=100] }
}
compmap=[ 1 2
          2 3 ]
```

The *mix* block defines an array of compositions by their weight %. For example, in the case of mix 2 and 3, it is 100% the "fuel_4pct" and "fuel_2pct" compositions defined on the *uox* blocks above. In the case of mix 1, it is 99% by weight *fuel_3pct* and 1% by weight the SCALE StdComp *Gd2O3* (gadolinia). Each mixture number (defined by numbered *mix* objects) is then referenced in the *compmap* array to define an individual pin composition. For UO_x-based fuels, ORIGAMI automatically calculates the pin enrichment for cross-section library interpolation via ARP. (Interpolation for MOX-based fuels is not supported by ORIGAMI at this time.)

The *uox* keyword is an ORIGAM-specific shortcut to allow for easy specification of UO_2-based fuels along with their enrichment; ORIGAMI automatically expands the *uox* keyword into a SCALE StdComp block with a UO_2 base and explicitly-calculated uranium isotopics per Table 5.4.3. For example, the *uox* block "fuel_3pct"expands to the following (Example 5.4.3):

Example 5.4.3: Equivalent explicit expansion of the "fuel_3pct" block

```
stdcomp(fuel_3pct){
   base=uo2
   iso[92234=0.02848 92235=3.2 92236=0.01472 92238=96.7568]
}
```

For *uox*-based entries, the uranium isotopic distribution is calculated from the user-specified enrichment per the formula outlined in Table 5.4.3 [ORIGAMIOWHR94, ORIGAMIRGI10]:

Table 5.4.3: Uraniumm isotope dependent on X wt% ^{235}U

Isotope	**Isotope wt%**
^{234}U	0.0089 X
^{235}U	1.0000 X
^{236}U	0.0046 X
^{238}U	100 - 1.0135 X

Users may also specify materials directly using SCALE mixture processor conventions; for example, the user could simply enter fuel mixture 2 directly as a StdComp as shown in Example 5.4.4 and Example 5.4.5:

Example 5.4.4: Direct specification of materials in ORIGAMI (i.e., within the mixture block)

```
mix(2){
  stdcomp(fuel_4pct){
    base=uo2
    iso[92234=XXX 92235=XXX 92236=XXX 92238=XXX]
   }
}
```

Or similarly, one can refer to a composition by its alias:

Example 5.4.5: Indirect specification of fuel material mixtures (outside the mixture block)

```
stdcomp(fuel_4pct){
   base=uo2
   iso[92234=XXX 92235=XXX 92236=XXX 92238=XXX]
}
mix(2){ comps[ fuel_4pct=100.0 ] }
```

The *uox* keyword is thus useful when a user wishes to quickly specify a UO_2-based fuel; however, in cases where the user wishes to specify the isotopic fractions of each uranium isotope, the use of a StdComp object is recommended.

Caution: The mixture composition system in ORIGAMI is very flexible but the user is cautioned that ORIGAMI does not rigorously check that the specified composition is neutronically similar to that used to generate the ORIGEN library used in the calculation.

For example, use of gadolinia burnable absorbers in the ORIGAMI input will yield incorrect results if the ORIGEN library was generated without gadolinia, due to the extreme thermal flux depression that gadolinia creates. It is therefore **up to the user** to verify that the libraries specified for the depletion zone are matched neutronically to the compositions specified.

5.4.4.4 Power history block

The data contained in the power history block is the same as in the *BURNDATA* block of the TRITON lattice physics depletion sequence in SCALE (see the TRITON chapter, BURNDATA block). The power-history block describes the burnup and decay of the assembly and has the following general form:

Example 5.4.6: Origami power history block

```
hist[
   cycle{ keywords for cycle-1 }
   cycle{ keywords for cycle-2 }
   ... *(repeat for total number of cycles) ...*
]
```

Because the cycles must be processed in order, the array syntax with "[]" is used for the "hist" block. (The block syntax "{}" implies no order for its contents.) The "hist" array consists of one or more "cycle" blocks, each describing the assembly irradiation and/or decay for some period of time. Each cycle is defined by (a) the assembly total specific power; (b) number of exposure days at this power; (c) the number of ORIGEN library burnup interpolations during the exposure period; and (d) number of days of decay following the exposure period.

The keywords defining this information are:

power= <real number>
Assembly specific power (MW/MTU) for the cycle

(**Default:** 0.0)

burn= <real number>
Length of cycle exposure period in days

(**Default:** 0.0)

down= <real number>
Downtime (decay) in days following exposure

(**Default:** 0.0)

nlib= <integer>
Number of ORIGEN library burnup-interpolations during the cycle

(**Default:** 1)

Table 5.4.4: Keywords in the power history (hist) block hist-repeat

Keyword	Description	Default
power=	assembly specific power (MW/MTU) for the cycle	0.0
burn=	length of the cycle exposure period in days	0.0
nlib=	number of ORIGEN library burnup-interpolations during the cycle	1
down=	downtime in days following the exposure	0.0

Example 5.4.7 demonstrates the use of the power-history block for four cycles:

Example 5.4.7: Example of the ORIGAMI "hist" block for irradiation cycle history

```
hist[
  cycle{ power=35.6 burn=400 nlib=6 down=30 }
  cycle{ power=38.2 burn=350 nlib=6 down=30 }
  cycle{ power=30.0 burn=200 nlib=4 down=30 }
  cycle{ down=10000 }
]
```

ORIGAMI discretizes time intervals first by *cycles* (composed of a fixed power over a set burn time interval and / or decay time), where each *cycle* is composed of a number of *substeps*. The power-history block, along with values of `nburn` and `ndecay` from the input parameter block, define various types of nested time intervals (substeps) for the ORIGEN calculations. The entire time period for an ORIGAMI case is first of all divided into the cycles defined within the power-history block. Each cycle is divided into an exposure interval (`burn`) and a decay (`down`) interval. The exposure interval has a constant specific power, but it is further subdivided into a number of equally spaced burnup steps defined by `nlib` in the power-history block. This parameter specifies the number of burnup-dependent ORIGEN libraries to use during the exposure interval. Cross section values for each burnup step are interpolated using the burnup at the midpoint of the step and remain constant throughout the burnup step. The burnup period associated with a single ORIGEN library, or a decay period, is called a time "step." Finally, each burnup step, as well as the entire decay step, is divided into a number of computational "substeps"—the actual time steps used in the ORIGEN solver kernel. The number of substeps in each burnup step is given by the value of `nburn`, while the number of decay substeps is equal to the value `ndecay`. The default number of substeps for both burnup and decay is equal to 10. The substeps for irradiation are equally spaced but for decay follow the rule of threes, i.e. each substep increases in duration by a factor of three over the previous substep.

For the example given above, there are four cycles. The first three cycles include both exposure and decay intervals, while the last cycle is decay only. In the first cycle, the assembly-specific power is 35.6 MW/MTU, which remains constant over the 400-day exposure interval; therefore, the total burnup for the exposure period is 400*35.6 = 14240 MWD/MTU. This exposure period is divided into six burnup steps of 66.67 days, each with a cross-section library based on the midpoint burnup of that step. Thus, ORIGEN libraries are interpolated at 1186.7, 3560.0, 5933.3, 8306.7, 10680.0, and 13053.3 MWD/MTU. Each of the six burnup steps is further subdivided into 10 computational substeps. Likewise, the decay interval of 30 days is divided into 10 computational substeps.

5.4.4.5 Source options block

This block defines options used in computing neutron and gamma sources. The block is only used if the input energy group boundary arrays `ggrp` or `ngrp` is given, which indicates that radiation decay source spectra are to be computed. The general form of this block is:

`srcopt` `{ ... keyword-value pairs ... }`

Where the following blocks are permitted:

- `sublibs`
- `brem_medium`
- `alphan_medium`
- `print`

The following (Example 5.4.8) is an example of the `srcopt` input block:

Example 5.4.8: Template of the ORIGAMI "srcopt" block options

```
srcopt{
   sublib= ...
   brem_medium= ...
   alphan_medium= ...
   print= ...

}
```

If *print=yes*, then text files with axial neutron and gamma sources are created.

sublib= [lt / fp / ac / all]
Gamma sources from light elements / fission products / actinides / all nuclides.

(**Default:** all)

brem_medium= [H2O / UO2 / none]
Medium for Bremsstrahlung production based on water (H2O), uranium oxide (UO2), or no Bremsstrahlung calculation (none)

(**Default:** UO2)

alphan_medium= [UO2 / borosilicate / case]
Target medium used for (α, n) source caclulation; UO_2, borosilicate glass, or case-specific mixture.

(**Default:** case)

print= [yes / no]
Write text-based output file containing source information / only write radiation source terms to binary ft71 file.

(**Default:** no)

Table 5.4.5: Keywords in the ORIGAMI source options (srcopt) block

Keyword	**Description**	**Default**
sublib=	*lt / fp / ac / all* → gamma sources from: light elements / fission products / actinides / all nuclides	all
brem_medium=	*none / H2O / UO2 /* → bremsstrahlung production based on: no bremsstrahlung / water / UO_2	uo2
alphan_medium=	*UO2 / borosilicate/ case* → (alpha,n) source computed for: UO_2/ borosilicate glass / case-specific mixture	case
print=	*yes / no* → write output text file containing sources / only write sources in binary output ft71 file	no

5.4.4.6 Output print-options block

This block defines the desired ORIGEN output response edits to be printed by ORIGAMI.

The following is an example input which edits response values for the mass in grams, activities in Curies, and concentrations in atoms/barn-cm, for all nuclides (isotopes) broken down by actinides or fission products as well as curies by element, totaled over all nuclide sub-libraries (sublibs).

Example 5.4.9: Example of Origami's "print" block for specifying output print options

```
print{
   nuc{ units=[grams curies atoms-per-barn-cm] sublibs=[fp ac] }
   ele{ units=[curies] total=yes }
}
```

nuc= { }, **ele**={ }
: Block to specify print options for output by individual nuclides / elements

units= [moles / gram-atoms / grams / curies / becquerels / watts / g-watts / m3_air / m3_water / weight_ppm / atoms_ppm / atoms-per-barn-cm]

: Output concentrations in units of gram-atoms (moles), grams, curies, becquerels, total thermal power (alpha, beta, and gamma), thermal power from gammas only, radiotoxicity / dilution factors for air and water, mass fraction (in ppm), atom fraction (in ppm), atoms / barn-cm^2, respectively.

: One or more output units may be specified, separated by commas.

: (**Default:** gram-atoms)

sublibs= [le / fp / ac / all]
: Output concentration units for light element sublibrary, fission product sublibrary, actinide sublibrary, or all nuclides.

: (**Default:** all)

total= [no / yes]
: Print out total concentration for nuclides / elements for each selected unit type.

: (**Default:** yes)

Table 5.4.6: Keywords in ORIGAMI "print" block

Keyword	Description	Default
nuc / ele	Specify print options for output by individual nuclides / elements	N/A
units=	*moles / gram-atoms / grams / curies / becquerels / watts / g-watts / m3_air / m3_water / weight_ppm / atoms_ppm / atoms-per-barn-cm* Output concentrations in units of gram-atoms (moles), grams, curies, becquerels, total thermal power (alpha, beta, and gamma), thermal power from gammas only, radiotoxicity / dilution factors for air and water, mass fraction (in ppm), atom fraction (in ppm), atoms / barn-cm, respectively.	all

continues on next page

[2] Requires volume input

Table 5.4.6 – continued from previous page

Keyword	Description	Default
sublibs=	*le / fp / ac / all* → output concentration units for light element / fission product / actinide sub-libraries	all
total=	*yes / no* → print out total concentration for nuclides / elements for each output unit type	yes

5.4.4.7 Input data arrays

For all other input arrays in ORIGAMI, the input values are entered in either of the general forms (with or without =):

```
array[ ... values ... ]

array=[ ... values ... ]
```

The array `libs`, which defines the ORIGEN library files, is the only one that is strictly required for all cases. Cases that simulate 0D or 1D lumped-assembly models typically only require one entry for a single ORIGEN library (assuming uniform axial enrichment), while the simulated 3D depletion model may utilize multiple libraries if specific ORIGEN libraries are pre-generated for different pin locations (e.g., adjacent to a water hole, Gd rods, etc.). If multiple libraries are used, the array `libmap` is required to identify the pin locations associated with the input libraries. The numbering of these libraries in the libmap array corresponds to the ordering of libraries in the `libs` array; i.e., a "1" corresponds to the first library specified, "2" to the second, and so on. A zero-value entry in the array indicates that the location is not to be depleted (i.e., a non-fuel region, such as a water hole or guide tube).

For **single** array values, the array bracket syntax is not required. For example, each of the following is equivalent:

```
compmap=[1]

compmap[1]

compmap=
```

Note that the assignment operator (=) is likewise optional for arrays when using the square-bracket syntax.

Unless the 0D lumped-assembly model (i.e., lumped mass with no axial power variation) is used, at least one of the arrays (`pz`, `pxy`) describing the power variations must also be entered. The 1D axial depletion model requires that the `pz` array be entered, while the pin-wise depletion model additionally requires the array `pxy` . The data in arrays `pxy` and `pz` correspond to the variables r_{xy} and a_z, respectively, described in Sect. 5.4.3.4. The axial and XY power distributions are normalized to unity inside the code, so that only the ratios of the input array values are significant. As discussed in Sect. 5.4.3.4, it is generally recommended to use the final burnup distributions rather than the relative power distributions for the values in the `pxy` and `pz` arrays.

The array `nuccomp` defines the nuclides to be included in the output compBlock file, described in more detail in Sect. 5.4.5. The nuclides in the array are identified by their seven digit IZZZAAA identifier defined as ID = I * 1000000 + Z * 1000 + A, where Z is the atomic number; A is the mass number, and I is the isomeric state (I=0 for ground; I=1 for first metastable; etc.). For example, identifiers for ^{16}O and ^{242m}Am are 8016 and 1095242, respectively. If this array is omitted, the nuclides in Table 5.4.9 are used. This is described in more detail in Sect. 5.4.5.1.

The optional array describing the non-fuel elements in the assembly contains pairs of values (element, mass), where "element" is the chemical symbol for a particular element, and "mass" is the mass of the element in kilograms per MTU. For example, the:

```
nonfuel=[ zr=520.3 sn=8.4 ]
```

indicates that the assembly contains 520.3 kilograms of zirconium and 8.4 kilograms of tin for each metric ton of uranium (MTU) in the assembly. Note that elemental masses are specified — the isotopic masses are computed internally by the code using natural abundances in the data library. It is also possible to normalize the total mass of non-fuel elements to a specified fraction of the MTU mass using the parameter `fracnf` in the parameter block. In this case, only the relative amounts of each non-fuel element are needed for the `nonfuel` array. Non-fuel masses are distributed uniformly among all the fuel depletion regions.

libs= [...]
: List of ORIGEN one or more library file names for fuel in assembly

  Required

libmap= [integer(s)]
: XY map of library identifiers associated with each pin in assembly. Library identifiers correspond to the order of the ORIGEN libraries entered in the `libs` array (i.e., index positions)

  (**Default:** [1])

  See also:

  libs

commap= [integer(s)]
: XY map of mixture identifiers that correspond to the mixture ID in the fuelcomp block.

  (**Default:** [1])

  See also:

  fuelcomp

pxy= [real number(s)]
: XY map of pin power shaping factors / fractional powers. Must be a square array (e.g., 15×15). Defaults to lumped assembly model (no individual pins).

  (**Default:** [1.0])

pz= [real number(s)]
: Axial (Z) power shaping factors / fractional power distribution for the assembly.

  (**Default:** [1.0])

meshz= [real number(s)]
: Axial mesh boundaries (cm) for the axial relative power zones. Only required to define axial mesh for viewing results; but if entered, it must be consistent with axial power shape. The number of entries should be one greater than number of entries in pz array.

  (**Default:** *none*)

  See also:

  pz

modz= [real number(s)]
Axial variation in water density (g/cc) corresponding to the axial power zones.

(**Default:** [0.723])

nonfuel= [key-value pairs]
Non-fuel materials contained in assembly. Values are entered in pairs of element-symbol=mass (kg per mtu of HM). If parameter fracnf is input, mass of non-fuel materials is normalized to this fraction of fuel mtu.

Note:

Oxygen mass in UO_2 should not be entered here (i.e., this is pre-supplied by ORIGAMI).

(**Default:** *None*)

---

ggrp= [real numbers]
Energy boundaries (eV) for defining decay gamma source spectrum, in monotonically increasing order.

(**Default:** *None*)

ngrp= [real numbers]
Energy boundaries (eV) for defining (α, n) and fission neutron source spectrum.

(**Default:** Nuclides in Table 5.4.9)

nuccomp= [IZZZAAA values]
User-specified list of nuclides (in IZZZAAA format) to be included in the compBlock file.

(**Default:** Nuclides specified in Table 5.4.9).

Table 5.4.7: Description of ORIGAMI input arrays

Array Name	**Description**	**Default**
libs *	List of ORIGEN library file names for fuel in assembly. [characters]	None
libmap	XY map of library identifiers associated with each pin in assembly. Library identifiers correspond to the order of the ORIGEN libraries entered in the *libs* block. [integers]	1
compmap	XY map of mixture identifiers that correspond to the mixture ID in the fuelcomp block. [integers]	1
pxy	XY map of pin power shaping factors / fractional powers. Must be a square array (e.g., 15×15). Defaults to lumped assembly model (no individual pins). [real numbers]	1.0
pz	Axial (Z) power shaping factors / fractional power distribution for the assembly. [real numbers]	1.0
meshz	Axial mesh boundaries (cm) for the axial relative power zones. Only required to define axial mesh for viewing results; but if entered, it must be consistent with axial power shape. The number of entries should be one greater than number of entries in pz array. [real numbers]	None

continues on next page

Table 5.4.7 – continued from previous page

Array Name	Description	Default
modz	Axial variation in water density (g/cc) corresponding to the axial power zones. [real numbers]	0.723
nonfuel	Non-fuel materials contained in assembly. Values are entered in pairs of (element-symbol=mass(kg) per mtu of HM). If parameter *fracnf* is input, mass of non-fuel materials is normalized to this fraction of fuel mtu. **NOTE:** Oxygen mass in UO_2 should **not** be entered here (i.e., this is pre-supplied by ORIGAMI). [character / real number pairs]	None
ggrp	Energy boundaries (eV) for defining decay gamma source spectrum. [real numbers]	None
ngrp	Energy boundaries (eV) for defining (α, n) and fission neutron source spectrum. [real numbers]	None
nuccomp	List of nuclide IZZZAAAs to be included in output *compBlock* file.	Table 5.4.9 Nuclides
* indicates required		

5.4.5 ORIGAMI INPUT/OUTPUT FILES

Table 5.4.8 gives the input and output files for ORIGAMI. ORIGAMI produces printed output results as well as several optional output files described in this section. In order to reduce the potentially voluminous amount of printout, by default ORIGAMI only prints the concentrations in grams for selected actinides in each axial zone of every pin, and only for the last time step (e.g., decay step) of the last cycle in the power-history block. Time-dependent results are given for all substeps in the last step (i.e, there are `nburn` and `ndecay` substeps within a burn step or decay step, respectively) In addition, the blended actinide concentrations over all pins are printed for each axial zone, and for the entire lumped assembly. Additional types of printed output can be specified in the `print` block. The concentrations, as well as optional neutron and gamma source spectra information, for all nuclides, in all pins and axial zones are also stored in the ORIGEN binary output file, often called an "ft71" file. The contents and format of the binary file are described in the ORIGEN documentation of the SCALE manual. The binary file information can be edited by the OPUS module in SCALE. Like the printed output, the ft71 file is written by default only for the last step of the last cycle. However, both the printed output and binary file results can be obtained at additional time steps by specifying the input variables output and ft71, respectively, in the OPTIONS input block. These input parameters can have the keywords:

The output files are written in the user output directory for the calculation (i.e., the same directory where the printed output file is written — the default is the directory from where the case was submitted). File names are prefixed by an extension consisting of the input file base-name appended to an optional character string given by the input keyword `prefix` . For example, if the ORIGAMI input file is named file:*ORIGAMICase.inp*, the base-name is `ORIGAMICase`. Thus, if the keyword `prefix` is not included in the input, the file containing the axial decay heat results is named file:*ORIGAMICase_AxialDecayHeat*. On the other hand, if the input contains the keyword **`prefix=CE16X16`**, the file is named `ORIGAMICase_CE16X16_AxialDecayHeat`.

In order to capture axially dependent features of an assembly (such as partial-length rods), users may elect to construct sequential ORIGAMI cases that modify the XY pin map features (e.g., library and enrichment maps) between cases. In order to allow for these types of "continuation" cases (in which the sequential case

represents an adjacent axial span of the assembly), the `offsetz` feature is provided, which adjusts the axial numbering for ORIGAMI outputs (such as for MCNP materials & spectra cards, axial decay heat, etc.). The `offsetz` parameter offsets the axial numbering for these output files, where the (integer) value provided corresponds to the *last* axial zone number calculated by ORIGAMI (default: 0). For more details on the syntax of the **`options`** block, see Sect. 5.4.3.4.

Table 5.4.8: ORIGAMI input/output files

File Name[3]	**Description**	**Type**	**Format**
`compBlock`	Mixture compositions in standard composition format for input to SCALE codes such as KENO	out	text
`MCNP_matls.inp`	Nuclide identifiers and weight fractions in format for MCNP material cards	out	text
`MCNP_gamma.inp`	Total gamma source intensity in MCNP source format. Only output if gamma energy group boundaries are entered in input array `ggrp`	out	text
`MCNP_neutron.inp`	Total neutron source intensity in MCNP source format. Only output if neutron energy group boundaries are entered in input array `ngrp`	out	text
`AxialGammaSpec`	Gamma spectrum (photons/sec) by axial zone, enabled by "srcopt{ print=yes }".	out	text
`AxialNeutSpec`	Neutron spectrum (neutron/sec) by axial zone, enabled by "srcopt{ print=yes }".	out	text
`AxialDecayHeat`	Decay heat source (watts) by axial zone, enabled by "options{ decayheat=yes }"	out	text
`assm.f71`	Output stacked ORIGEN ft71 files for each axial zone	out	binary
`assembly_restart.f71`	Input stacked ORIGEN ft71 files for each axial zone	in	binary
`.f71`	Output of stacked ORIGEN ft71 files for each pin and axial zone	out	binary
`actinideMesh.3dmap`	Binary MeshView file of selected actinide masses by depletion cell	out	binary
`actinideMesh.ASCII.txt`	Plaintext MeshView file of selected actinide masses by depletion cell	out	text
`fpMesh.3dmap`	Binary MeshView file of selected fission product masses by depletion cell	out	binary
`fpMesh.ASCII.txt`	Plaintext MeshView file of selected fission product masses by depletion cell	out	text
`burnupMesh.3dmap`	Binary MeshView file of depletion node burnups	out	binary

[3] Note that all file names are prefixed by an identifier `${OUTBASENAME}`, where `${OUTBASENAME}` is a prefix constructed from the input file base name followed by the character string given by input keyword `prefix= *.*` For example, the input file named "my.inp" with **`prefix=sample`** would give an output prefix `my_sample`; e.g., `my_sample.f71`, `my_sample.assm.f71`, `my_sample_MCNP_matls.inp`, etc.

5.4.5.1 Generation of SCALE standard composition data file

If input parameter `stdcomp=yes` is specified, ORIGAMI produces a text file containing a SCALE standard composition description for each axial interval. The file is written in the form of a `stdcomp` block that can be directly used as input to any SCALE module that requires a composition block. If a 1D axial depletion model is used for the assembly, the composition for each axial zone is given a unique mixture number defined for an axial node "*Z*" as:

$$\text{(1D axial model)} \quad \text{mix} = 1000 + (asmid - 1) \times N_Z + Z \tag{5.4.13}$$

where N_Z is the number of axial zones and `asmid` is the input identifier. For example, if there are 12 axial zones and the input for `asmid` is 20, then the mixture number associated with axial zone number 1 is mix = 1229, and the mixture for zone 12 is mix= 1240. If an assembly is represented by a 3D multiple-pin model, the mixture number is defined,

$$\text{(3D model)} \quad \text{mix} = 1000 + (asmid - 1) \times N_Z + Z + X \times 100000 + Y \times 10000000 \tag{5.4.13}$$

where X and Y correspond to the row and column numbers of the pin.

The nuclides components of the mixtures may be specified in the input array `nuccomp`, or by default the mixture may consist of the nuclides given in Table 5.4.9, which are the nuclides recommended in [ORIGAMIRGlW12] for burnup credit analysis, plus ^{16}O.

The temperatures of the mixtures are set by the value of parameter `temper`, which defaults to a value of 293 Kelvin. The number densities of the nuclides in the mixtures are calculated using the following expressions:

$$\begin{aligned} N_Z^{(i)} &= \rho \frac{M_Z^{(i)}}{M_Z \cdot 10^6} \cdot \frac{N_A}{A^{(i)}} \cdot 0.8814 \cdot 10^{-24} \\ &= \rho \frac{M_Z^{(i)}}{M_Z \cdot A^{(i)}} \cdot 5.309 \cdot 10^{-7} \end{aligned} \tag{5.4.13}$$

Where:

$N_Z^{(i)}$ = number density of nuclide "i" in zone Z, in units of atoms of "i" per barn-cm of UO_2;

ρ = density of UO_2 (g/cc), defined by the input parameter `fdens` (default is 10.4 g/cc);

$M_Z^{(i)}$ = mass (g) of nuclide i in axial zone Z , obtained from ORIGEN calculation;

$M_Z \cdot 10^6$ = mass (g) of uranium in axial zone Z, where M_Z is given by Eq. (5.4.2);

$A^{(i)}$ = mass (g) of 1 mole of nuclide i;

0.8814 = weight fraction of uranium in UO_2;

10^{-24} = cm^3 per barn-cm.

The definitions of other parameters appearing in this equation are given in Sect. 5.4.3.4. An example of the standard composition file produced by ORIGAMI is given in Sect. 5.4.7, Example 5.4.12 (illustrated in sample problem 2, Example 5.4.11).

Table 5.4.9: Default burnup credit nuclides in Standard Composition output

Nuclide	ZAID	Nuclide type
^{16}O	8016	light element
^{234}U	92234	actinide
^{235}U	92235	actinide
^{236}U	92236	actinide
^{238}U	92238	actinide
^{237}Np	93237	actinide
^{238}Pu	94238	actinide
^{239}Pu	94239	actinide
^{240}Pu	94240	actinide
^{241}Pu	94241	actinide
^{242}Pu	94242	actinide
^{241}Am	95241	actinide
^{243}Am	95243	actinide
^{95}Mo	42095	fission product
^{99}Tc	43099	fission product
^{101}Ru	44101	fission product
^{103}Rh	45103	fission product
^{109}Ag	47109	fission product
^{133}Cs	55133	fission product
^{143}Nd	60143	fission product
^{145}Nd	60145	fission product
^{147}Sm	62147	fission product
^{149}Sm	62149	fission product
^{150}Sm	62150	fission product
^{151}Sm	62151	fission product
^{152}Sm	62152	fission product
^{151}Eu	63151	fission product
^{153}Eu	63153	fission product
^{155}Gd	64155	fission product

5.4.5.2 MCNP data files

If the input parameter `mcnp=yes` is set in the `options` block, the computed weight fractions for the materials in each axial zone also are output in a file in the format of MCNP material cards. These material cards are designed to be coupled to a corresponding MCNP assembly geometry using the same numbering convention for the depletion zones. Sect. 5.4.7 shows an example of the MCNP material information produced by ORIGAMI. The numbering convention of the MCNP materials cards works by combining the axial and pin numbers into a material card, where pins are counted sequentially by row, starting with the bottom-left row of input, counting from left to right across each row to the top-right pin (i.e., the bottom-left pin is pin #1, etc.). The pin numbers reset with each axial zone, starting from the bottom zone, counting up from 1. The naming convention for materials cards is thus the pin number (1-999) followed by the zone number (1-99); for example, pin #15 of axial zone #12 would be **m1512**. Accompanying each material card is a list of ZAID numbers and final concentrations (following depletion/decay) for the cell expressed in weight fractions. The weight fractions are given as negative values in accordance with MCNP convention. The fuel density, which

may be used in the MCNP cell card, is equal to the value of the input parameter `fdens`.

When parameter `mcnp=yes` is set, ORIGAMI also produces output files containing the fuel assembly radiation source magnitude by depletion zone, to support modeling with MCNP. The gamma/neutron source term cards correspond to the total gamma or neutron intensity (particles/s) from each respective depletion region, using the same numbering convention as that for the MCNP material cards. The source magnitude is computed by summing over the MG source spectra defined in Eq. (5.4.12).

$$S_Z^{(p)} = \sum_g S_{Z,g}^{(p)} \tag{5.4.14}$$

Where:

$S_Z^{(p)}$ = total source magnitude (p/s) for particles of type p;

$S_{Z,g}^{(p)}$ = multigroup source magnitude (p/s) for energy group g, and particles of type p

More details on the ORIGEN calculation of the source terms can be found in the ORIGEN section of SCALE documentation.

5.4.5.3 Decay heat calculation

When input parameter `decayheat=yes` is specified in the input, a text file containing the decay heat source by axial zone, summed over all pins, is generated as output. The decay heat in zone Z is given in watts and is computed from the

$$H_Z = \sum_{i=1}^{\text{itot}} Q_i \lambda_i \frac{M_Z^{(i)}}{A^{(i)}} \cdot 1.602 \cdot 10^{-13} \cdot N_A = 9.65 \cdot 10^{10} \sum_{i=1}^{\text{itot}} Q_i \lambda_i \frac{M_Z^{(i)}}{A^{(i)}} \tag{5.4.15}$$

where:

Q_i = decay energy in MeV for nuclide i;

λ_i = decay constant (s^{-1}) for nuclide i;

$M_Z^{(i)}$ = mass (g) of nuclide i in axial zone Z, obtained from ORIGEN calculation;

$A^{(i)}$ = mass (g) of 1 mole of nuclide i;

itot = total number of nuclides in burned fuel,

1.602×10^{-19} = number of joules per MeV.

An example output decay heat file produced by ORIGAMI is shown in Sect. 5.4.7, Example 5.4.13 (from sample problem 2).

5.4.5.4 ORIGEN results files

The ORIGEN computation for each depletion region produces an ORIGEN binary concentrations output file, historically called an "ft71" because it was written on "Fortran tape" number 71. The file named `$`*`OUTBASENAME`*`.f71` contains the concentrations for all depletion regions, stacked within a single binary file, where `${OUTBASENAME}` is the base of the output file name, e.g. the "my" in `my.out`.

The order of stored cases on the f71 file corresponds to the order in which ORGAMI processes individual depletion cases, starting with the bottom-left row in the user-supplied power map (pin #1) and looping left to right, progressively up through the series of rows. This process repeats for each axial zone, starting from the bottom of the assembly and working upward (i.e., starting with pin #1, axial zone #1, looping through each pin on axial zone #1, and then proceeding to pin #1 on axial zone #2, etc.). This convention is the same as that used for TRITON arrays.

In addition, the compositions are blended over all pins for each axial zone to obtain the axially-dependent compositions for the lumped assembly, stored in a file named `$`*`OUTBASENAME`*`.assm.f71`. If saved, this file may be input as a restart file, as discussed in Sect. 5.4.3.3.

5.4.5.5 Plotting features

ORIGAMI creates three separate mesh summaries of material inventories for individual depletion regions, useful for 3D visualization and inspection. These include maps of (1) depletion region burnups, (2) selected actinide concentrations (including isotopes of U, Pu, A m, and Cm), and (3) selected fission products typically used for burnup evaluation, including isotopes of Cs, Y, Ag, Rh, Ru, Eu, Sm, Nd, Gd, and others). Additionally, ORIGAMI outputs a separate mesh tally of individual node burnups. These outputs are described in Table 5.4.8.

Note: The mesh files are only created if the user specifies the (optional) input arguments for assembly pitch (`pitch`) and axial zone locations (`meshz`).

These output mesh-dependent maps can be visualized using the Java-based Mesh File Viewer program included with SCALE. An example MeshView visualization of one of these outputs is shown in Fig. 5.4.2 and Fig. 5.4.3. MeshView is installed in `$`*`SCALE`*`/Meshview`, where `${SCALE}` is the installation directory. A script to run MeshView is located at `$`*`SCALE`*`/cmds/meshview`.

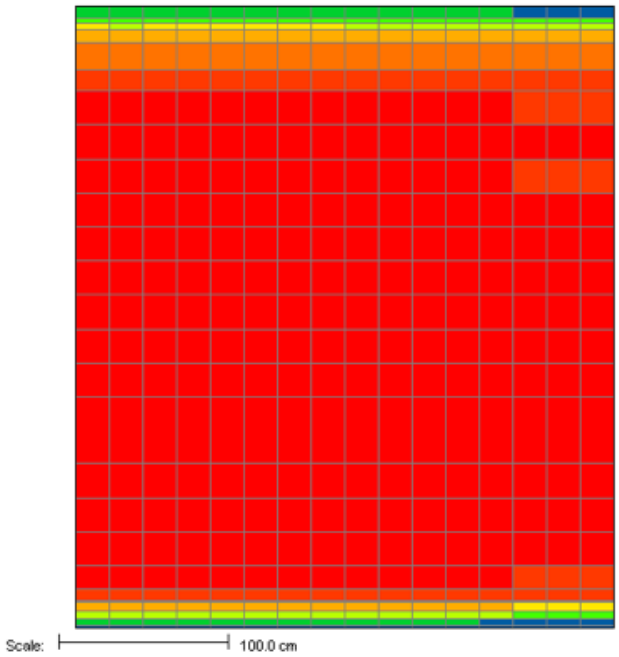


Fig. 5.4.2: MeshView plot of total plutonium content in the 3D depletion regions (XZ plane)

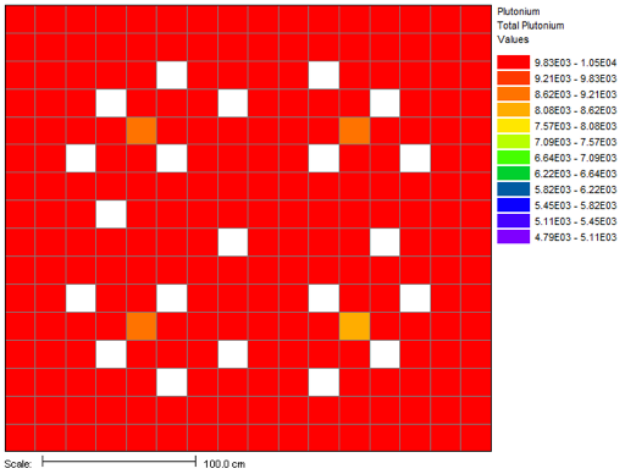


Fig. 5.4.3: MeshView plot of total plutonium content in the 3D depletion regions (XY plane).

5.4.6 PARALLEL EXECUTION ON LINUX CLUSTERS

For large 3D depletion problems it is advantageous to execute the ORIGEN calculations for different depletion regions in parallel. This can be done on Linux clusters using MPI. When parallel execution mode is enabled, ORIGAMI distributes the individual depletion cases across the pin rows, columns, and axial zones across several processors; the depletion calculation is thus split across several processors. ORIGAMI then collects the inventories from each calculation node and concatenates the output.

To execute ORIGAMI in parallel mode, a parallel-enabled MPI build of SCALE must be used and ORIGAMI should be invoked with the percent (%) prefix:

```
=%

<normal ORIGAMI input follows>
```

Additionally, for parallel jobs spanning multiple computational nodes (as opposed to those just using multiple processors on the same node, it is recommended to use the **-T** option to specify a common temporary directory (such as a network-mounted directory accessible to all nodes). This is due to the way ORIGAMI divides the problem space in parallel mode; each computational node stores its respective binary dump file of the individual pin/zone concentrations. Upon completion of execution, the master node must be able to locate these individual problem node-generated binary dump files; thus, by using a common temporary directory, ORIGAMI can correctly re-assembly the individual pinwise dumpfiles into a single consolidated "master" dump file.

The following is a typical execution command line to execute ORIGAMI in parallel.

scalerte -N [number of nodes] -M [machine file] -T [tmpdir] [input_file.inp]

For more information on executing SCALE in parallel, see the SCALE Readme file.

5.4.7 SAMPLE PROBLEMS

This section shows sample problems for each of the three types of simulated assembly models: 0D fully lumped, 2D lumped axial depletion, and 3D pinwise depletion, and also demonstrates a restart case.

5.4.7.1 Sample problem 1: fully lumped assembly model

The first example, Example 5.4.10, corresponds to a fully-lumped assembly model in which the materials are depleted with a space-independent (i.e., assembly average) flux distribution. The assembly contains 0.38 MTU, and the fuel is 2.8 wt% enriched. The assembly also includes several non-fuel materials corresponding to cladding and other structural materials. Note that the non-fuel concentrations are specified in kg/MTU, and thus are not the actual total non-fuel masses in the 0.38 MTU assembly. The assembly is depleted for three cycles with specific powers of 40.0, 38.6, and 25.2 MW/MTU, respectively. The ORIGEN library data are interpolated for eight different burnup steps during the irradiation periods of the first two cycles, and for six burnup steps in the last cycle.

Table 5.4.10 gives the calculated actinide concentrations at the end of the third cycle.

Example 5.4.10: Input for ORIGAMI sample problem 1

```
=origami
  title='fully lumped assembly model'
  libs=[ ce14x14 ]
  fuelcomp{
    stdcomp(fuel){
       base=uo2 iso[92234=0.02848 92235=3.2 92236=0.01472 92238=96.7568] }
       mix(1){ comps[fuel=100] }
   }
   options{ mtu=0.38 ft71=all }
   nonfuel=[ cr=3.366 mn=0.1525 fe=6.309 co=0.0302
             ni=2.366 zr=516.3 sn=8.412 gd=2.860 ]
   hist[
      cycle{ power=40 burn=284 nlib=4 down=54 }
      cycle{ power=38.6 burn=300 nlib=4 down=28 }
      cycle{ power=25.2 burn=250 nlib=3 down=30 }
   ]
   print{
```

(continues on next page)

(continued from previous page)

```
    nuc {
        sublibs=[ac fp]
        units=[grams moles]
        total=no }
    }
end
```

Table 5.4.10: Calculated Actinide inventories for sample problem 1

Nuclide *	Mass (g)
^{234}U	6.820E+01
^{235}U	3.621E+03
^{236}U	1.487E+03
^{238}U	3.598E+05
^{237}Np	1.348E+02
^{238}Pu	3.862E+01
^{239}Pu	1.919E+03
^{240}Pu	7.820E+02
^{241}Pu	3.960E+02
^{242}Pu	1.394E+02
^{241}Am	1.474E+01
^{243}Am	2.491E+01
^{242}Cm	2.663E+00
^{244}Cm	5.698E+00
TOTAL	6.820E+01

* Actinides with concentrations less than 0.0001 are not shown.

5.4.7.2 Sample problem 2: lumped axial depletion assembly model

The second example has the same lumped assembly and power history as sample problem 1, except in this case an axial power distribution is provided for eight zones, so that the fuel burnup will vary axially; the ORIGAMI input for this case is provided as Example 5.4.11. Also, the options to generate standard composition and decay output files are requested.

Table 5.4.11 gives the computed actinide concentrations in grams for the first four of the eight axial zones. Since the input axial power distribution is symmetrical about the assembly midplane, the last four zones have identical concentrations as the first four. The last column in the table shows actinide masses for the entire assembly.

Example 5.4.12 is a listing of the contents of the compBlock file, which contains standard composition input for the eight axial zones in the assembly at the end of cycle 3. A complete description of the SCALE standard composition input format is given in the XSPROC chapter The first entry on each line in Example 5.4.12 corresponds to the SCALE nuclide identifier. Only the default burnup credit analysis are included. The second entry is the mixture number associated with a particular axial zone. The mixture number for an axial zone is obtained using Eq. (5.4.13). The third entry is always zero in this file, and the fourth entry corresponds to the number density in atoms per barn-cm. The next entry on the line is the temperature, which has the default value of 293.0 since the input parameter `temper` was not specified. The final entry is an "end" statement. The information in this file can be used as the `read comp` input block for any SCALE module.

Example 5.4.13 shows a listing of the file AxialDecayHeat, which contains the heat source at the end of the third cycle. The entries in the file correspond to the decay power in watts for the eight axial zones, which are computed using Eq. (5.4.15).

Example 5.4.11: Input for ORIGAMI sample problem 2

```
=origami
title= 'lumped axial-deplete assembly model'
libs=[ ce14x14 ]
fuelcomp{
   uox(fuel2){ enrich=3.2 }
   mix(1){ comps[fuel2=100] }
}
options{
   mtu=0.38 stdcomp=yes decayheat=yes
 }
pz=[ 1.0 2.0 3.0 4.0 4.0 3.0 2.0 1.0 ]
nonfuel=[ cr=3.366 mn=0.1525 fe=6.309 co=0.0302
          ni=2.366 zr=516.3 sn=8.412 gd=2.860 ]
hist[
   cycle{ power=40 burn=284 nlib=4 down=54 }
   cycle{ power=38.6 burn=300 nlib=4 down=28 }
   cycle{ power=25.2 burn=250 nlib=3 down=30 }
]
end
```

Table 5.4.11: Calculated actinide inventories by axial zone for sample problem 2

Nuclide	Axial Zone 1 Mass (g)	Axial Zone 2 Mass (g)	Axial Zone 3 Mass (g)	Axial Zone 4 Mass (g)	TOTAL Mass (g)
^{234}U	1.1384E+01	9.4326E+00	7.6666E+00	6.11E+00	6.92E+01
^{235}U	9.7324E+02	5.9453E+02	3.3795E+02	1.78E+02	4.17E+03
^{236}U	1.0392E+02	1.6544E+02	2.0053E+02	2.15E+02	1.37E+03
^{238}U	4.5604E+04	4.5202E+04	4.4752E+04	4.43E+04	3.60E+05
^{237}Np	4.7874E+00	1.2622E+01	2.0984E+01	2.83E+01	1.33E+02
^{238}Pu	5.5445E-01	2.9024E+00	7.1722E+00	1.26E+01	4.65E+01
^{239}Pu	1.7378E+02	2.2968E+02	2.4438E+02	2.46E+02	1.79E+03
^{240}Pu	3.3124E+01	7.7894E+01	1.1457E+02	1.40E+02	7.30E+02
^{241}Pu	1.2974E+01	3.8401E+01	5.8714E+01	7.09E+01	3.62E+02
^{242}Pu	1.4543E+00	1.0115E+01	2.6393E+01	4.75E+01	1.71E+02
^{241}Am	5.3938E-01	1.5121E+00	2.0433E+00	2.12E+00	1.24E+01
^{243}Am	9.5029E-02	1.4341E+00	5.6093E+00	1.29E+01	4.00E+01
^{242}Cm	< 0.0001	2.0213E-01	4.7744E-01	7.56E-01	2.93E+00
^{244}Cm	< 0.0001	2.4685E-01	1.6315E+00	5.54E+00	1.49E+01
^{245}Cm	< 0.0001	< 0.0001	5.8836E-02	2.40E-01	6.11E-01
TOTAL	4.6920E+04	4.6346E+04	4.5780E+04	4.5221E+04	3.6854E+05

Example 5.4.12: Sample Problem 2: Standard composition file (default burnup credit nuclides)

```
o-16   1001 0 4.6395E-02 293.0 end
u-234  1001 0 5.6529E-06 293.0 end
```

(continues on next page)

(continued from previous page)

```
u-235  1001 0 4.8120E-04 293.0 end
u-236  1001 0 5.1165E-05 293.0 end
u-238  1001 0 2.2263E-02 293.0 end
np-237 1001 0 2.3470E-06 293.0 end
pu-238 1001 0 2.7067E-07 293.0 end
pu-239 1001 0 8.4481E-05 293.0 end
pu-240 1001 0 1.6036E-05 293.0 end
pu-241 1001 0 6.2547E-06 293.0 end
pu-242 1001 0 6.9823E-07 293.0 end
am-241 1001 0 2.6003E-07 293.0 end
am-243 1001 0 4.5435E-08 293.0 end
mo-95  1001 0 1.5700E-05 293.0 end
tc-99  1001 0 1.7475E-05 293.0 end
ru-101 1001 0 1.5272E-05 293.0 end
rh-103 1001 0 9.4877E-06 293.0 end
ag-109 1001 0 7.6473E-07 293.0 end
cs-133 1001 0 1.8563E-05 293.0 end
nd-143 1001 0 1.4504E-05 293.0 end
nd-145 1001 0 1.0445E-05 293.0 end
sm-147 1001 0 1.5451E-06 293.0 end
sm-149 1001 0 8.0469E-08 293.0 end
sm-150 1001 0 3.3359E-06 293.0 end
sm-151 1001 0 3.5230E-07 293.0 end
sm-152 1001 0 1.7469E-06 293.0 end
eu-151 1001 0 1.5582E-09 293.0 end
eu-153 1001 0 9.1077E-07 293.0 end
gd-155 1001 0 1.3966E-09 293.0 end
o-16   1002 0 4.6394E-02 293.0 end
u-234  1002 0 4.6837E-06 293.0 end
u-235  1002 0 2.9395E-04 293.0 end
u-236  1002 0 8.1452E-05 293.0 end
u-238  1002 0 2.2067E-02 293.0 end
np-237 1002 0 6.1878E-06 293.0 end
pu-238 1002 0 1.4169E-06 293.0 end
pu-239 1002 0 1.1165E-04 293.0 end
pu-240 1002 0 3.7710E-05 293.0 end
pu-241 1002 0 1.8513E-05 293.0 end
pu-242 1002 0 4.8561E-06 293.0 end
am-241 1002 0 7.2896E-07 293.0 end
am-243 1002 0 6.8566E-07 293.0 end
mo-95  1002 0 2.9706E-05 293.0 end
tc-99  1002 0 3.3424E-05 293.0 end
ru-101 1002 0 3.0467E-05 293.0 end
rh-103 1002 0 1.8486E-05 293.0 end
ag-109 1002 0 2.2784E-06 293.0 end
cs-133 1002 0 3.5311E-05 293.0 end
nd-143 1002 0 2.4869E-05 293.0 end
nd-145 1002 0 1.9393E-05 293.0 end
sm-147 1002 0 2.4415E-06 293.0 end
sm-149 1002 0 1.0154E-07 293.0 end
sm-150 1002 0 7.3182E-06 293.0 end
sm-151 1002 0 4.4543E-07 293.0 end
sm-152 1002 0 3.4915E-06 293.0 end
eu-151 1002 0 1.1064E-09 293.0 end
eu-153 1002 0 2.5199E-06 293.0 end
gd-155 1002 0 2.5157E-09 293.0 end
o-16   1003 0 4.6392E-02 293.0 end
u-234  1003 0 3.8068E-06 293.0 end
u-235  1003 0 1.6709E-04 293.0 end
u-236  1003 0 9.8728E-05 293.0 end
u-238  1003 0 2.1847E-02 293.0 end
np-237 1003 0 1.0287E-05 293.0 end
pu-238 1003 0 3.5014E-06 293.0 end
pu-239 1003 0 1.1880E-04 293.0 end
pu-240 1003 0 5.5465E-05 293.0 end
```

(continues on next page)

(continued from previous page)

```
pu-241 1003 0 2.8306E-05 293.0 end
pu-242 1003 0 1.2671E-05 293.0 end
am-241 1003 0 9.8507E-07 293.0 end
am-243 1003 0 2.6819E-06 293.0 end
mo-95  1003 0 4.2205E-05 293.0 end
tc-99  1003 0 4.7742E-05 293.0 end
ru-101 1003 0 4.5361E-05 293.0 end
rh-103 1003 0 2.6134E-05 293.0 end
ag-109 1003 0 4.1770E-06 293.0 end
cs-133 1003 0 5.0069E-05 293.0 end
nd-143 1003 0 3.1519E-05 293.0 end
nd-145 1003 0 2.6993E-05 293.0 end
sm-147 1003 0 2.8511E-06 293.0 end
sm-149 1003 0 1.2277E-07 293.0 end
sm-150 1003 0 1.1691E-05 293.0 end
sm-151 1003 0 5.2472E-07 293.0 end
sm-152 1003 0 5.0112E-06 293.0 end
eu-151 1003 0 9.0563E-10 293.0 end
eu-153 1003 0 4.4446E-06 293.0 end
gd-155 1003 0 4.0054E-09 293.0 end
o-16   1004 0 4.6390E-02 293.0 end
u-234  1004 0 3.0343E-06 293.0 end
u-235  1004 0 8.7951E-05 293.0 end
u-236  1004 0 1.0588E-04 293.0 end
u-238  1004 0 2.1605E-02 293.0 end
np-237 1004 0 1.3864E-05 293.0 end
pu-238 1004 0 6.1605E-06 293.0 end
pu-239 1004 0 1.1946E-04 293.0 end
pu-240 1004 0 6.7542E-05 293.0 end
pu-241 1004 0 3.4172E-05 293.0 end
pu-242 1004 0 2.2786E-05 293.0 end
am-241 1004 0 1.0210E-06 293.0 end
am-243 1004 0 6.1463E-06 293.0 end
mo-95  1004 0 5.3319E-05 293.0 end
tc-99  1004 0 6.0395E-05 293.0 end
ru-101 1004 0 5.9831E-05 293.0 end
rh-103 1004 0 3.2151E-05 293.0 end
ag-109 1004 0 6.2338E-06 293.0 end
cs-133 1004 0 6.2779E-05 293.0 end
nd-143 1004 0 3.4995E-05 293.0 end
nd-145 1004 0 3.3350E-05 293.0 end
sm-147 1004 0 2.9202E-06 293.0 end
sm-149 1004 0 1.4469E-07 293.0 end
sm-150 1004 0 1.6101E-05 293.0 end
sm-151 1004 0 6.0099E-07 293.0 end
sm-152 1004 0 6.3684E-06 293.0 end
eu-151 1004 0 8.1899E-10 293.0 end
eu-153 1004 0 6.4006E-06 293.0 end
gd-155 1004 0 5.5315E-09 293.0 end
o-16   1005 0 4.6390E-02 293.0 end
u-234  1005 0 3.0343E-06 293.0 end
u-235  1005 0 8.7951E-05 293.0 end
u-236  1005 0 1.0588E-04 293.0 end
u-238  1005 0 2.1605E-02 293.0 end
np-237 1005 0 1.3864E-05 293.0 end
pu-238 1005 0 6.1605E-06 293.0 end
pu-239 1005 0 1.1946E-04 293.0 end
pu-240 1005 0 6.7542E-05 293.0 end
pu-241 1005 0 3.4172E-05 293.0 end
pu-242 1005 0 2.2786E-05 293.0 end
am-241 1005 0 1.0210E-06 293.0 end
am-243 1005 0 6.1463E-06 293.0 end
mo-95  1005 0 5.3319E-05 293.0 end
tc-99  1005 0 6.0395E-05 293.0 end
ru-101 1005 0 5.9831E-05 293.0 end
```

(continues on next page)

(continued from previous page)

```
rh-103 1005 0 3.2151E-05 293.0 end
ag-109 1005 0 6.2338E-06 293.0 end
cs-133 1005 0 6.2779E-05 293.0 end
nd-143 1005 0 3.4995E-05 293.0 end
nd-145 1005 0 3.3350E-05 293.0 end
sm-147 1005 0 2.9202E-06 293.0 end
sm-149 1005 0 1.4469E-07 293.0 end
sm-150 1005 0 1.6101E-05 293.0 end
sm-151 1005 0 6.0099E-07 293.0 end
sm-152 1005 0 6.3684E-06 293.0 end
eu-151 1005 0 8.1899E-10 293.0 end
eu-153 1005 0 6.4006E-06 293.0 end
gd-155 1005 0 5.5315E-09 293.0 end
o-16   1006 0 4.6392E-02 293.0 end
u-234  1006 0 3.8068E-06 293.0 end
u-235  1006 0 1.6709E-04 293.0 end
u-236  1006 0 9.8728E-05 293.0 end
u-238  1006 0 2.1847E-02 293.0 end
np-237 1006 0 1.0287E-05 293.0 end
pu-238 1006 0 3.5014E-06 293.0 end
pu-239 1006 0 1.1880E-04 293.0 end
pu-240 1006 0 5.5465E-05 293.0 end
pu-241 1006 0 2.8306E-05 293.0 end
pu-242 1006 0 1.2671E-05 293.0 end
am-241 1006 0 9.8507E-07 293.0 end
am-243 1006 0 2.6819E-06 293.0 end
mo-95  1006 0 4.2205E-05 293.0 end
tc-99  1006 0 4.7742E-05 293.0 end
ru-101 1006 0 4.5361E-05 293.0 end
rh-103 1006 0 2.6134E-05 293.0 end
ag-109 1006 0 4.1770E-06 293.0 end
cs-133 1006 0 5.0069E-05 293.0 end
nd-143 1006 0 3.1519E-05 293.0 end
nd-145 1006 0 2.6993E-05 293.0 end
sm-147 1006 0 2.8511E-06 293.0 end
sm-149 1006 0 1.2277E-07 293.0 end
sm-150 1006 0 1.1691E-05 293.0 end
sm-151 1006 0 5.2472E-07 293.0 end
sm-152 1006 0 5.0112E-06 293.0 end
eu-151 1006 0 9.0563E-10 293.0 end
eu-153 1006 0 4.4446E-06 293.0 end
gd-155 1006 0 4.0054E-09 293.0 end
o-16   1007 0 4.6394E-02 293.0 end
u-234  1007 0 4.6837E-06 293.0 end
u-235  1007 0 2.9395E-04 293.0 end
u-236  1007 0 8.1452E-05 293.0 end
u-238  1007 0 2.2067E-02 293.0 end
np-237 1007 0 6.1878E-06 293.0 end
pu-238 1007 0 1.4169E-06 293.0 end
pu-239 1007 0 1.1165E-04 293.0 end
pu-240 1007 0 3.7710E-05 293.0 end
pu-241 1007 0 1.8513E-05 293.0 end
pu-242 1007 0 4.8561E-06 293.0 end
am-241 1007 0 7.2896E-07 293.0 end
am-243 1007 0 6.8566E-07 293.0 end
mo-95  1007 0 2.9706E-05 293.0 end
tc-99  1007 0 3.3424E-05 293.0 end
ru-101 1007 0 3.0467E-05 293.0 end
rh-103 1007 0 1.8486E-05 293.0 end
ag-109 1007 0 2.2784E-06 293.0 end
cs-133 1007 0 3.5311E-05 293.0 end
nd-143 1007 0 2.4869E-05 293.0 end
nd-145 1007 0 1.9393E-05 293.0 end
sm-147 1007 0 2.4415E-06 293.0 end
sm-149 1007 0 1.0154E-07 293.0 end
```

(continues on next page)

(continued from previous page)

```
sm-150 1007 0 7.3182E-06 293.0 end
sm-151 1007 0 4.4543E-07 293.0 end
sm-152 1007 0 3.4915E-06 293.0 end
eu-151 1007 0 1.1064E-09 293.0 end
eu-153 1007 0 2.5199E-06 293.0 end
gd-155 1007 0 2.5157E-09 293.0 end
o-16   1008 0 4.6395E-02 293.0 end
u-234  1008 0 5.6529E-06 293.0 end
u-235  1008 0 4.8120E-04 293.0 end
u-236  1008 0 5.1165E-05 293.0 end
u-238  1008 0 2.2263E-02 293.0 end
np-237 1008 0 2.3470E-06 293.0 end
pu-238 1008 0 2.7067E-07 293.0 end
pu-239 1008 0 8.4481E-05 293.0 end
pu-240 1008 0 1.6036E-05 293.0 end
pu-241 1008 0 6.2547E-06 293.0 end
pu-242 1008 0 6.9823E-07 293.0 end
am-241 1008 0 2.6003E-07 293.0 end
am-243 1008 0 4.5435E-08 293.0 end
mo-95  1008 0 1.5700E-05 293.0 end
tc-99  1008 0 1.7475E-05 293.0 end
ru-101 1008 0 1.5272E-05 293.0 end
rh-103 1008 0 9.4877E-06 293.0 end
ag-109 1008 0 7.6473E-07 293.0 end
cs-133 1008 0 1.8563E-05 293.0 end
nd-143 1008 0 1.4504E-05 293.0 end
nd-145 1008 0 1.0445E-05 293.0 end
sm-147 1008 0 1.5451E-06 293.0 end
sm-149 1008 0 8.0469E-08 293.0 end
sm-150 1008 0 3.3359E-06 293.0 end
sm-151 1008 0 3.5230E-07 293.0 end
sm-152 1008 0 1.7469E-06 293.0 end
eu-151 1008 0 1.5582E-09 293.0 end
eu-153 1008 0 9.1077E-07 293.0 end
gd-155 1008 0 1.3966E-09 293.0 end
```

Example 5.4.13: Sample problem 2: Decay heat file with axial decay heat by zone (Watts)

```
7.11397E+02
1.43786E+03
2.18631E+03
2.95428E+03
2.95428E+03
2.18631E+03
1.43786E+03
7.11397E+02
```

5.4.7.3 Sample problem 3: restart decay calculation for lumped axial depletion assembly model

The third example shows a restart decay-only calculation, using the ORIGEN ft71 binary file obtained from sample problem 2. This case calculates the composition of the burned fuel produced in sample problem 2 after 100,000 additional days of decay. The input this problem is given in Example 5.4.15. Because the input parameter **`restart=yes`** is specified, the initial composition of the assembly is obtained from a file named `assembly_restart.f71`. The shell input that precedes the ORIGAMI input in Example 5.4.15 copies the output ft71 file produced in sample problem 2, which was named `assembly_dump.f71`, into a file named `assembly_restart.f71` in the temporary directory for SCALE calculations. The restart file contains the complete inventory of nuclide compositions for eight axial zones. Because this restart case is decay only (i.e., `power` value is not given in the power-history block), it is necessary to provide the input parameter **`nz=8`** because this value is used to determine how many axial zones were used in the previous burnup calculations.

Table 5.4.12 shows the actinide composition of the first four axial of the (symmetrical) eight zones after 100,000 days of decay. The initial masses of these nuclides before decay are the values given in numref:*ex-origami-prob2-stdcmp*. The last column in Table 5.4.12 shows actinide masses for the entire assembly after the decay period.

Example 5.4.14: Sample problem 3: restart decay for a lumped axial depletion model

```
=origami
   title= 'lumped axial-deplete assembly model'
   libs=[ ce14x14 ]
   fuelcomp{
      %3.2 w/o
      uox(fuel){ enrich=3.2 }
      mix(1){ comps[fuel=100] }
   }
   options[ mtu=0.38 relnorm=no ]
   pz=[ 1.0 2.0 3.0 4.0 4.0 3.0 2.0 1.0 ]
   nonfuel=[ cr=3.366 mn=0.1525 fe=6.309 co=0.0302
             ni=2.366 zr=516.3 sn=8.412 gd=2.860 ]
   hist[
      cycle{ power=40 burn=284 nlib=8 down=54 }
      cycle{ power=38.6 burn=300 nlib=8 down=28 }
      cycle{ power=25.2 burn=250 nlib=6 down=30 }
   ]
   end
   =shell
      mv \*.assm.f71 assembly_restart.f71
   end
   =origami
      title= 'restart decay'
      asmid= 22
      libs=[ ce14x14 ]
      prefix=origam3
   options{
      stdcomp=yes decayheat=yes relnorm=no restart=yes nz=8
   }
   pz=[ 1.0 2.0 3.0 4.0 4.0 3.0 2.0 1.0 ]
   hist[
      cycle{ down=100000 }
   ]
   end
   =shell
      rm assembly_restart.f71
      rm ${OUTDIR}/\*origam3\*
   end
```

Table 5.4.12: Calculated actinide inventories by axial zone for sample problem 3

Nuclide	Axial Zone 1 Mass (g)	Axial Zone 2 Mass (g)	Axial Zone 3 Mass (g)	Axial Zone 4 Mass (g)	TOTAL Mass (g)
^{234}U	1.1889E+01	1.2133E+01	1.4319E+01	1.7738E+01	1.1216E+02
^{235}U	9.7454E+02	5.9616E+02	3.3963E+02	1.7955E+02	4.1798E+03
^{236}U	1.0486E+02	1.6764E+02	2.0378E+02	2.1907E+02	1.3907E+03
^{238}U	4.5604E+04	4.5202E+04	4.4752E+04	4.4256E+04	3.5963E+05
^{237}Np	9.2242E+00	2.5723E+01	4.0929E+01	5.2246E+01	2.5625E+02
^{238}Pu	6.8535E-02	3.6104E-01	8.8478E-01	1.5422E+00	5.7132E+00
^{239}Pu	1.7243E+02	2.2794E+02	2.4262E+02	2.4413E+02	1.7742E+03

continues on next page

Table 5.4.12 – continued from previous page

Nuclide	Axial Zone 1 Mass (g)	Axial Zone 2 Mass (g)	Axial Zone 3 Mass (g)	Axial Zone 4 Mass (g)	TOTAL Mass (g)
^{240}Pu	3.2192E+01	7.5957E+01	1.1297E+02	1.4095E+02	7.2412E+02
^{241}Pu	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001
^{242}Pu	1.4538E+00	1.0107E+01	2.6379E+01	4.7456E+01	1.7079E+02
^{241}Am	8.9945E+00	2.6564E+01	4.0451E+01	4.8623E+01	2.4927E+02
^{243}Am	9.2577E-02	1.3965E+00	5.4616E+00	1.2516E+01	3.8934E+01
^{242}Cm	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001
^{244}Cm	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001
^{245}Cm	< 0.0001	< 0.0001	5.7288E-02	2.3382E-01	5.9479E-01
TOTAL	4.6920E+04	4.6346E+04	4.5780E+04	4.5220E+04	3.6853E+05

5.4.7.4 Sample problem 4: Simplified 3D multi-pin model

The fourth example is a simulation of a simplified 3D depletion model. The ORIGAMI 3D model normally includes all fuel pins within the assembly, such as a 14×14 array. However to keep this example case simple and the execution time low, only a 2×2 array of four individual pins is considered for illustrative purposes. An axial fractional power distribution is also specified for two axial zones. Therefore the total number of depletion regions will be eight – two axial for each of the four pins. Two different ORIGEN libraries are used to obtain cross sections for the four pins. This is done whenever fuel pins in different locations in the assembly have significantly different neutron spectra, such as if some pins are adjacent to a control rod. In this sample problem, ORIGEN libraries for two different types of assembly designs CE (Combustion Engineering) 14×14 and 16×16 assembly designs, respectively are used to demonstrate the use of pin-dependent libraries, although in reality the ORIGEN libraries normally would be pre-generated for different pin locations within a single type of assembly configuration. The values specified in `libmap` indicate which library is to be used for each pin location. Example 5.4.15 shows the input for this sample problem.

Example 5.4.15: Input for ORIGAMI Sample Problem 4: a simplified multi-pin, multi-axial model

```
=origami
 title= 'multi-pin; multi-library pin-deplete model'
 prefix= origam4
 libs=[ ce14x14 ce16x16 ]

 fuelcomp{
    %3.2 w/o
    stdcomp(fuel){ base=uo2 iso[92234=0.028569 92235=3.21 92236=0.014766
                                92238=96.746665] }
    mix(1){ comps[fuel=100] }
 }

 options{ mtu=0.4 decayheat=yes }
 libmap=[ 1 1
          2 2 ]
 pxy=[ 0.284 0.283
       0.218 0.215 ]
 pz=[ 0.55 0.45 ]
 hist[
   cycle{ power=39.78 burn=284.0 nlib=2 down=54.0 }
 ]
end
```

Table 5.4.13 shows selected actinide compositions for the first row of two pins, that is, locations (1,1) and (1,2), for each of the two axial zones. The blended compositions over all fuel pins, for the two axial zones,

are given in Table 5.4.14. The output decay heat file for the assembly is shown in Table 5.4.15, as a function of axial zone, summed over all pins. Note that this file has the prefix "sample4_" appended to the standard file name, since *prefix=sample4* is specified in the input.

Table 5.4.13: Actinide inventories by axial zone for pins (1,1) and (1,2) in sample problem 4

Nuclide	pin (1,1) Axial Zone 1 Mass (g)	pin (1,1) Axial Zone 2 Mass (g)	pin (1,2) Axial Zone 1 Mass (g)	pin (1,2) Axial Zone 2 Mass (g)
^{234}U	1.2115E+01	1.2493E+01	1.2143E+01	1.2516E+01
^{235}U	1.0701E+03	1.1528E+03	1.0762E+03	1.1581E+03
^{236}U	1.0383E+02	8.9286E+01	1.0277E+02	8.8348E+01
^{237}U	1.0664E-03	< 0.0001	1.0428E-03	< 0.0001
^{238}U	4.8003E+04	4.8074E+04	4.8009E+04	4.8078E+04
^{237}Np	4.8000E+00	3.6009E+00	4.7055E+00	3.5303E+00
^{239}Np	5.8742E-07	4.4485E-07	5.7555E-07	4.3692E-07
^{238}Pu	4.7198E-01	2.9453E-01	4.5684E-01	2.8512E-01
^{239}Pu	1.8855E+02	1.6706E+02	1.8705E+02	1.6560E+02
^{240}Pu	3.3235E+01	2.5225E+01	3.2625E+01	2.4736E+01
^{241}Pu	1.3906E+01	9.2953E+00	1.3537E+01	9.0304E+00
^{242}Pu	1.3733E+00	7.3260E-01	1.3161E+00	7.0081E-01
^{241}Am	2.3603E-01	1.5773E-01	2.2978E-01	1.5322E-01
^{243}Am	8.4143E-02	< 0.0001	7.9365E-02	< 0.0001
TOTAL	4.9432E+04	4.9535E+04	4.9440E+04	4.9541E+04

Table 5.4.14: Blended actinide inventories by axial zone (all pins) for sample problem 4

Nuclide	Axial Zone 1 Mass (g)	Axial Zone 2 Mass (g)	TOTAL Mass (g)
^{234}U	4.7388E+01	4.9082E+01	9.6469E+01
^{235}U	4.0115E+03	4.3778E+03	8.3894E+03
^{236}U	4.5851E+02	3.9537E+02	8.5387E+02
^{238}U	1.9184E+05	1.9216E+05	3.8399E+05
^{237}Np	2.2821E+01	1.7132E+01	3.9953E+01
^{238}Pu	2.5739E+00	1.6049E+00	4.1789E+00
^{239}Pu	7.8246E+02	6.9963E+02	1.4821E+03
^{240}Pu	1.5436E+02	1.1801E+02	2.7238E+02
^{241}Pu	6.7918E+01	4.6455E+01	1.1437E+02
^{242}Pu	8.1693E+00	4.4411E+00	1.2610E+01
^{241}Am	1.1476E+00	7.8696E-01	1.9346E+00
^{243}Am	6.0150E-01	2.5995E-01	8.6145E-01
TOTAL	1.9740E+05	1.9787E+05	3.9526E+05

Table 5.4.15: Sample problem 4: Decay heat by axial zone (Watts)

6.96472E+03	5.72539E+03

5.4.7.5 Sample problem 5: PWR 3D assembly model

This sample problem shows the input for a simulated full 3D pressurized water reactor (PWR) assembly-depletion model, which corresponds to a 16×16 lattice with 26 axial zones. Example 5.4.16 shows the ORIGAMI input for this case. The arrays `pxy` and `pz` define the 3D XY-Z fractional power distribution. Four different pre-processed ORIGEN libraries are used to describe the pin-averaged ORIGEN cross-sections for the assembly. The array `libmap` assigns these libraries to the appropriate pin locations. It can be seen that the `libmap` array contains values of zero at 21 locations. These correspond to non-depleting (i.e., zero power) locations. The input includes the information (parameter `pitch`, and array z=) necessary to generate 3D mesh summary maps for subsequent visualization. In this case, the axial mesh is not uniform. Since a total of 6110 ORIGEN cases are executed for the depletable pins, this ORIGAMI calculation was performed in parallel using MPI. Figure 5.4.3 shows a plot of the axial burnup distribution, summed over all pins.

Example 5.4.16: Input for ORIGAMI Sample Problem 4

```
=%origami
title= 'PWR 3D deplete model'
prefix= pwr
options{ pitch= 19.816 }
fuelcomp{
uox(fuel){ enrich=3.5 }
   mix(1){ comps[fuel=100] }
}
libs=[ lib1 lib2 lib3 lib4]
libmap=[
   3 2 2 2 2 2 2 2 2 2 2 2 2 2 3
   2 1 1 1 1 4 1 1 1 4 1 1 1 1 2
   2 1 1 4 4 0 4 4 1 4 0 4 4 1 1 2
   2 1 4 0 4 4 4 0 4 1 4 4 0 4 1 2
   2 1 4 4 1 4 1 4 1 1 4 1 4 4 1 2
   2 4 0 4 4 0 4 1 1 4 0 4 4 0 4 2
   2 1 4 4 1 4 1 1 1 1 4 1 1 4 1 2
   2 1 4 0 4 1 1 4 1 1 1 1 4 1 1 2
   2 1 1 4 1 1 4 0 4 1 1 4 0 4 1 2
   2 1 4 1 1 4 1 4 1 1 4 1 4 4 1 2
   2 4 0 4 4 0 4 1 1 4 0 4 4 0 4 2
   2 1 4 4 1 4 1 4 1 1 4 1 4 4 1 2
   2 1 4 0 4 4 4 0 4 1 4 4 0 4 1 2
   2 1 1 4 4 0 4 4 1 4 0 4 4 1 1 2
   2 1 1 1 1 4 1 1 1 1 4 1 1 1 1 2
   3 2 2 2 2 2 2 2 2 2 2 2 2 2 3 ]
pxy=[
   0.99 0.98 0.98 0.99 0.99 0.99 0.99 0.99
   0.99 0.99 0.99 0.99 0.99 0.98 0.98 0.97 0.98
   0.99 0.99 0.99 1.00 1.01 1.02 1.00 1.00
   1.00 1.01 1.02 1.00 0.99 0.98 0.98 0.98
   1.00 1.00 1.01 1.03 1.03 0.00 1.03 1.01
   1.03 1.04 0.00 1.03 1.02 1.00 0.99 0.98
   1.01 1.01 1.03 0.00 1.04 1.04 1.02 0.00
   1.03 1.04 1.04 1.04 0.00 1.02 1.00 0.99
   1.01 1.02 1.02 1.05 0.73 1.04 1.02 1.02
   1.03 1.03 1.04 0.72 1.04 1.04 1.01 1.00
   1.02 1.04 0.00 1.05 1.04 0.00 1.03 1.01
   1.01 1.03 0.00 1.03 1.04 0.00 1.02 1.00
   1.02 1.03 1.02 1.05 1.04 1.04 1.02 1.01
   1.01 1.02 1.03 1.02 1.02 1.03 1.01 1.00
   1.01 1.02 1.04 0.00 1.04 1.02 1.02 1.03
```

(continues on next page)

(continued from previous page)

```
   1.01 1.01 1.01 1.02 1.03 1.02 1.00 1.00
   1.00 1.01 1.02 1.03 1.02 1.02 1.03 0.00
   1.02 1.01 1.01 1.03 0.00 1.02 0.99 0.98
   1.00 1.01 1.03 1.02 1.02 1.03 1.03 1.03
   1.01 1.01 1.03 1.02 1.03 1.03 1.00 0.99
   1.01 1.02 0.00 1.04 1.03 0.00 1.03 1.01
   1.01 1.02 0.00 1.03 1.03 0.00 1.01 0.98
   1.00 1.01 1.04 1.04 0.72 1.04 1.03 1.03
   1.01 1.01 1.02 0.71 1.03 1.02 0.99 0.98
   1.00 1.00 1.02 0.00 1.04 1.04 1.04 0.00
   1.02 1.01 1.03 1.03 0.00 1.01 0.98 0.97
   0.99 0.99 1.01 1.03 1.04 0.00 1.04 1.03
   1.01 1.02 0.00 1.02 1.01 0.99 0.97 0.97
   0.99 0.99 0.99 1.00 1.01 1.03 1.01 1.00
   1.00 1.00 1.01 1.00 0.99 0.97 0.97 0.97
   1.00 0.99 1.00 1.00 1.01 1.01 1.01 1.00
   0.99 0.99 0.99 0.99 0.98 0.97 0.97 0.97 ]

pz=[0.486645842
    0.510544887
    0.641121243
    0.798557507
    0.931372279
    1.063949280
    1.173174524
    1.178015382
    1.241701554
    1.247451593
    1.203231683
    1.228462686
    1.237668911
    1.221002529
    1.191997899
    1.231513011
    1.222065701
    1.172711869
    1.200902470
    1.164812132
    1.083204453
    0.931028309
    0.810656652
    0.700324838
    0.611466339
    0.516416427 ]

meshz=[ 0.0 2.0 6.0 10.0 16.5 23.0 37.0 57.0 77.0 97.0 16.0
        136.0 156.0 176.0 196.0 216.0 236.0 256.0 276.0 296.0
        316.0 328.5 344.0 352.0 355.5 359.0 366.0 ]
hist[
   cycle{ power=49.395 burn=385 nlib=3 down=52 }
   cycle{ power=43.772 burn=360 nlib=2 down=7673 }
]
end
```

6. SENSITIVITY AND UNCERTAINTY ANALYSIS

Introduction by B. T. Rearden and J. D. McDonnell

SCALE provides a suite of computational tools for sensitivity and uncertainty analysis to (1) identify important processes in safety analysis and design, (2) provide a quantifiable basis for neutronics validation for criticality safety and reactor physics analysis based on similarity assessment, and (3) quantify the effects of uncertainties in nuclear data and physical parameters for safety analysis [SU-RWJ+11, SU-WIJ+13].

Sensitivity Analysis and Uncertainty Quantification

Sensitivity analysis provides a unique insight into system performance in that the predicted response of the system to a change in some input process is quantified. Important processes can be identified as those that cause the largest changes in the response per unit change in the input. In neutron transport numerical simulations, calculating important responses such as k_{eff}, reaction rates, and reactivity coefficients requires many input parameters, including material compositions, system geometry, temperatures, and neutron cross section data. Because of the complexity of nuclear data and its evaluation process, the response of neutron transport models to the cross section data can provide valuable information to analysts. The SCALE sensitivity and uncertainty (S/U) analysis sequences-known as the Tools for Sensitivity and Uncertainty Analysis Methodology Implementation (TSUNAMI)-quantify the predicted change in k_{eff}, reaction rates, or reactivity differences due to changes in the energy-dependent, nuclide-reaction–specific cross section data, whether continuous-energy or multigroup.

Uncertainty quantification is useful for identifying potential sources of computational biases and highlighting parameters important to code validation. When applying uncertainties in the neutron cross section data, the sensitivity of the system to the cross section data can be applied to propagate the uncertainties in the cross section data to an uncertainty in the system response. Additionally, SCALE provides the ability to stochastically sample uncertainties in nuclear data or any other model input parameter (e.g., dimensions, densities, temperatures) and propagate these input uncertainties to uncertainties not only as traditional TSUNAMI responses of k_{eff}, reaction rates, and reactivity, but also in any general output quantity such as burnup isotopics, dose rates, etc. Additionally, where the same input quantities are used in multiple models, the sampling techniques can be applied to quantify the correlation in uncertainties of multiple systems due to the use of the same uncertain parameters across these systems.

Validation of Codes and Data

Modern neutron transport codes such as the KENO Monte Carlo codes in the SCALE code system can predict k_{eff} with a high degree of precision. Still, computational biases of one percent or more are often found when using these codes to model critical benchmark experiments. The primary source of this computational bias is believed to be errors in the cross section data as bounded by their uncertainties. These errors can be tabulated in cross section covariance data. To predict or bound the computational bias for a design system of interest, the *American National Standards for Nuclear Criticality Safety in Operations with Fissionable Material Outside Reactors* (ANSI/ANS-8.1-1998) [SU-AmericanNSSCWGA8198] and the *American National Standard for Validation of Neutron Transport Methods for Nuclear Criticality Safety Calculations* (ANSI/ANS-8.24-2007) [SU-AmericanNSSCWGA82407] allow calculations to be used to determine subcritical limits for the design of fissionable material systems. The standards require validation of the analytical methods and data used in nuclear criticality safety calculations to quantify any computational bias and the uncertainty in the bias. The validation procedure must be conducted through comparison of computed results with experimental data, and the design system for which the subcritical limit is established must fall within the area of applicability of the experiments chosen for validation. The ANS-8.1 standard defines the area(s) of applicability as

"the limiting ranges of material compositions, geometric arrangements, neutron-energy spectra, and other relevant parameters (e.g., heterogeneity, leakage, interaction, absorption, etc.) within which the bias of a computational method is established."

TSUNAMI Techniques for Code Validation

The TSUNAMI software provides a unique means to determine the similarity of nuclear criticality experiments to safety applications [SU-BRH+04]. The TSUNAMI validation techniques are based on the assumption that computational biases are primarily caused by errors in cross section data, the potential for which are quantified in cross section covariance data. TSUNAMI provides two methods to establish the computational bias introduced through cross section data.

For the first method, instead of using one or more average physical parameters to characterize a system, TSUNAMI determines the uncertainties in the computed response that are shared between two systems due to cross section uncertainties. These shared uncertainties directly relate to the bias shared by the two systems. To accomplish this, the sensitivity to each group-wise nuclide-reaction–specific cross section is computed for all systems considered in the analysis. Correlation coefficients are developed by propagating the uncertainties in neutron cross section data to uncertainties in the computed response for experiments and safety applications through sensitivity coefficients. The bias in the experiments, as a function of correlated uncertainty with the intended application, is extrapolated to predict the bias and bias uncertainty in the target application. This correlation coefficient extrapolation method is useful where many experiments with uncertainties that are highly correlated to the target application are available.

For the second method, data adjustment or data assimilation techniques are applied to predict computational biases, and more general responses, including but not limited to k_{eff}, can be addressed simultaneously [SU-BRH+04]. This technique uses S/U data to identify a single set of adjustments to nuclear data and experimental responses, taking into account their correlated uncertainties, which would improve the agreement between the response values from the experimental results and computational simulations. The same data adjustments are then used to predict an unbiased response (e.g., k_{eff}) value for the application and an uncertainty on the adjusted response value. The difference between the originally calculated response value and the new post-adjustment response value represents the bias in the original calculation, and the uncertainty in the adjusted value represents the uncertainty in this bias. If experiments are available to validate the use of a particular nuclide in the application, the uncertainty of the bias for this nuclide may be reduced. If similar experiments are not available, the uncertainty in the bias for the given nuclide is high. Thus, with a complete set of experiments to validate important components in the application, a precise bias with a small uncertainty can be predicted. Where the experimental coverage is lacking, a bias can be predicted with an appropriately large uncertainty. The data assimilation method presents many advantages over other techniques in that biases can be projected from an agglomeration of benchmark experiments, each of which may represent only a small component of the bias of the target application. Also, contributors to the computational bias can be analyzed on an energy-dependent nuclide-reaction–specific basis. However, this technique requires additional data that are not generally available and must be quantified or approximated by the analyst, specifically the correlation coefficients that quantify the relative independence of experimental measurements that use the same equipment, whether nuclear fuel, reactivity devices, or measurement tools.

Sensitivity and Uncertainty Analysis Tools in SCALE

The **TSUNAMI-1D** and **TSUNAMI-3D** analysis sequences compute the sensitivity of k_{eff} and reaction rates to energy-dependent cross section data for each reaction of each nuclide in a system model. The one-dimensional (1D) transport calculations are performed with XSDRNPM, two-dimensional (2D) transport calculations are preformed using NEWT, and the three-dimensional (3D) calculations are performed with KENO V.a or KENO-VI. The Monte Carlo capabilities of TSUNAMI-3D provide for S/U analysis from either

continuous-energy or multigroup neutron transport, where the deterministic capabilities of TSUNAMI-1D only operate in multigroup mode. SAMS (Sensitivity Analysis Module for SCALE) is applied within each analysis sequence to provide the requested S/U data. Whether performing a continuous-energy or multigroup calculation, energy-dependent sensitivity data are stored in multigroup-binned form in a sensitivity data file (SDF) for subsequent analysis. Additionally, these sequences use the energy-dependent cross section-covariance data to compute the uncertainty in the response value due to the cross section-covariance data.

TSAR (Tool for Sensitivity Analysis of Reactivity Responses) computes the sensitivity of the reactivity change between two k_{eff} calculations, using SDFs from TSUNAMI-1D, and/or TSUNAMI-3D. TSAR also computes the uncertainty in the reactivity difference due to the cross section covariance data.

TSUNAMI-IP (TSUNAMI Indices and Parameters) uses the SDFs generated from TSUNAMI-1D, TSUNAMI-3D, or TSAR for a series of systems to compute correlation coefficients that determine the amount of shared uncertainty between each target application and each benchmark experiment considered in the analysis. TSUNAMI-IP offers a wide range of options for more detailed assessment of system-to-system similarity. Additionally, TSUNAMI-IP can generate input for the **USLSTATS** (Upper Subcritical Limit Statistical Software) [SU-LBDH97] trending analysis and compute a penalty, or additional margin, needed for the gap analysis. USLSTATS is distributed as a graphical user interface with SCALE, but its use is documented in the TSUNAMI Primer [SU-RMB+09], not in this documentation chapter.

TSURFER (Tool for S/U Analysis of Response Functions Using Experimental Results) is a bias and bias uncertainty prediction tool that implements the generalized linear least-squares (GLLS) approach to data assimilation and cross section data adjustment that also uses the SDFs generated from TSUNAMI-1D, TSUNAMI-3D, or TSAR. The data adjustments produced by TSURFER are not used to produce adjusted cross section data libraries for subsequent use; rather, they are used only to predict biases in application systems.

The TSUNAMI Primer also documents the use of the graphical user interfaces for TSUNAMI, specifically ExSITE (Extensible SCALE Intelligent Text Editor) that facilitates analysis with TSUNAMI-IP, TSURFER, TSAR, and USLSTATS as well as VIBE (Validation, Interpretation and Bias Estimation) for examining SDF files, creating sets of benchmark experiments for subsequent analysis, and gathering additional information about each benchmark experiment.

Sampler is a "super-sequence" that performs general uncertainty analysis by stochastically sampling uncertain parameters that can be applied to any type of SCALE calculation, propagating uncertainties throughout a computational sequence. Sampler treats uncertainties from two sources: (1) nuclear data and (2) input parameters. Sampler generates the uncertainty in any result generated by any computational sequence through stochastic means by repeating numerous passes through the computational sequence, each with a randomly perturbed sample of the requested uncertain quantities. The mean value and uncertainty in each parameter is reported, along with the correlation in uncertain parameters where multiple systems are simultaneously sampled with correlated uncertainties.

Used in conjunction with nuclear data covariances available in SCALE, Sampler is a general technique to obtain uncertainties for many types of applications. SCALE includes covariances for multigroup neutron cross section data, as well as for fission product yields, and radioactive decay data, which allow uncertainty calculations to be performed for most MG computational sequences in SCALE. At the present time, nuclear data sampling cannot be applied to SCALE CE Monte Carlo calculations, although the fundamental approach is still valid.

Used in conjunction with uncertainties in input data, Sampler can determine the uncertainties and correlations in computed results due to uncertainties in dimensions, densities, distributions of material compositions, temperatures, or any quantities that are defined in the user input for any SCALE computational sequence.

This methodology was developed to produce uncertainties and correlations in criticality safety benchmark experiments, [SU-MR15] but it has a wide range of applications in numerous scenarios in nuclear safety analysis and design. The input sampling capabilities of Sampler also include a parametric capability to determine the response of a system to a systematic variation of an input parameter.

6.1 TSUNAMI-1D: CONTROL MODULE FOR ONE-DIMENSIONAL CROSS-SECTION SENSITIVITY AND UNCERTAINTY

B. T. Rearden, M. A. Jessee, L. M. Petrie, and M. L. Williams

6.1.1 ABSTRACT

TSUNAMI-1D (**T**ools for **S**ensitivity and **Un**certainty **A**nalysis **M**ethodology **I**mplementation in **One** **D**imension) is a SCALE control module that facilitates the application of sensitivity and uncertainty analysis theory to nuclear systems analyses. TSUNAMI-1D was originally developed to provide sensitivity and uncertainty analysis of k_{eff} for criticality safety applications, and subsequent updates provide for analysis of system responses other than k_{eff}, using generalized perturbation theory. TSUNAMI-1D provides for automated processing of material input, processing of cross-section data, calculation of neutron transport solutions, calculation of sensitivity coefficients, and the calculation of uncertainties in system responses due to cross-section-covariance data. The XSDRNPM module is used for the transport solver. XSDRNPM uses the method of discrete ordinates to calculate k_{eff} for applications that are appropriate for 1D modeling. The SAMS module is used to determine the sensitivities of the calculated value of k_{eff} and other system responses to the nuclear data used in the calculation as a function of nuclide, reaction type, and energy. The uncertainties in the calculated value of k_{eff} and other system responses, resulting from uncertainties in the basic nuclear data used in the calculation, are estimated using energy-dependent cross-section-covariance matrices. The implicit effects of the resonance self-shielding calculations are predicted using BONAMIST.

6.1.2 ACKNOWLEDGMENTS

The authors acknowledge Bryan Broadhead of Oak Ridge National Laboratory, and R. L. Childs, formerly of the Oak Ridge National Laboratory, for their assistance with this work. The support and encouragement of Calvin Hopper, Cecil Parks, and Don Mueller of Oak Ridge National Laboratory is also appreciated. Additionally, the authors wish to acknowledge Debbie Weaver and Sheila Walker for their assistance in preparing this document.

6.1.3 INTRODUCTION

TSUNAMI-1D (**T**ools for **S**ensitivity and **Un**certainty **A**nalysis **M**ethodology **I**mplementation in **One** **D**imension) is a SCALE control module that facilitates the application of sensitivity and uncertainty theory to nuclear system analyses. The data computed with TSUNAMI-1D are the sensitivity of a system response, such as k_{eff}, to each constituent cross-section data component used in the calculation. The sensitivity data are coupled with cross-section uncertainty data, in the form of multigroup covariance matrices, to produce an uncertainty in the system response due to uncertainties in the underlying nuclear data. The group-wise sensitivity data computed with TSUNAMI-1D are stored in a sensitivity data file (.sdf file) that is suitable for use in assessing system similarity for code validation purposes using TSUNAMI-IP, (see TSUNAMI-IP chapter), and for advanced bias assessment using TSURFER, see the TSURFER chapter.

This manual is intended to provide the user with a detailed reference on code input options and provide some examples of the application of TSUNAMI-1D to generate sensitivity and uncertainty data. A detailed description of code input is provided in Sect. 6.1.4, three sample problems are given in Sect. 6.1.5 the techniques used in each computational sequence are described in Sect. 6.1.3.1, and additional information is

provided in the appendices. A new user may wish to begin by reviewing the sample problems, and then refer to the input details in Sect. 6.1.4 to customize an input for his specific needs.

TSUNAMI-1D provides automated, problem-dependent cross sections using the same methods and input as the **C**riticality **S**afety **A**nalysis **S**equences (CSAS). The BONAMIST code computes the sensitivity of resonance self-shielded cross to the input data, the so-called "implicit sensitivities."

After the cross sections are processed, the TSUNAMI-1D sequence performs two XSDRNPM criticality calculations, one forward and one adjoint. Finally, the sequence calls the SAMS module to calculate the sensitivity coefficients that indicate the sensitivity of the calculated values to changes in the cross sections and the uncertainty in the calculated value due to uncertainties in the basic nuclear data. SAMS prints energy-integrated sensitivity coefficients and their statistical uncertainties to the SCALE output file and generates a separate data file containing the energy-dependent sensitivity coefficients.

In addition to the sensitivity and uncertainty analysis sequence, the TSUNAMI-1DC sequence can be used to verify the accuracy of the TSUNAMI-1D calculations with direct perturbation criticality calculations. The verification of computed sensitivity coefficients is imported for systems where the cell-weighted material is not the only material used in the model. By default, TSUNAMI-1DC performs the same functions as the TSUNAMI-1D sequence with PARM=CENTRM, except that it does not perform the adjoint XSDRNPM calculation and does not call the SAMS module.

6.1.3.1 TSUNAMI-1D Techniques

TSUNAMI-1D is a SCALE control module. As such, its primary function is to control a sequence of calculations that are performed by other codes. The input for each of the TSUNAMI-1D sequences is very similar to that used for CSAS1, with the addition of the system model description and optional sensitivity calculation data. TSUNAMI-1D uses the same material and cell data input as all other SCALE sequences. The control sequences available in TSUNAMI-1D are summarized in Table 6.1.1, where the functional modules executed by each control sequence are also shown. A general flow diagram of TSUNAMI-1D is shown in Fig. 6.1.1.

Table 6.1.1: TSUNAMI-1D control sequences.

Control module	**Functional modules executed by the control module**				
TSUNAMI-1D	XSProc	XSDRNPM forward	XSDRNPM adjoint*	BONAMIST	SAMS*
TSUNAMI-1DC	XSProc	XSDRNPM forward			
* The XSDRNPM adjoint calculation and SAMS calculation on are repeated for each system response defined by the user.					

TSUNAMI-1D and many other SCALE sequences apply a standardized procedure to provide appropriate cross sections for the calculation. This procedure is carried out by routines of the XSProc that generate number densities and related information, prepare geometry data for resonance self-shielding and flux-weighting cell calculations, and create data input files for the cross-section processing codes.

By default, the TSUNAMI-1D sequence performs cross-section processing with XSProc, exercising all available options there, performs the forward and adjoint XSDRNPM calculations, calls BONAMIST to produce implicit sensitivity coefficients, then calls SAMS to produce sensitivity and uncertainty output and *sdf* files. Optional sequence level parameters can be used to change methods applied in resonance self-shielding

and exclude the implicit sensitivity calculation, which detailed later in this document. If additional system responses are requested in the input, TSUNAMI-1D executes additional generalized adjoint XSDRNPM and SAMS calculations for each system response.

The input requirements for the model description are very similar to those used for multiregion cell descriptions in the cell data section of input. The definition of system responses other than k_{eff} requires both the `DEFINITIONS` and `SYSTEMRESPONSE` block of input data. These blocks of data are described in Sect. 6.1.4.4. TSUNAMI-1D also reads and prepares inputs for the SAMS calculation. The additional input blocks for the SAMS module are optional. The input format of the SAMS blocks of data are described in the SAMS chapter.

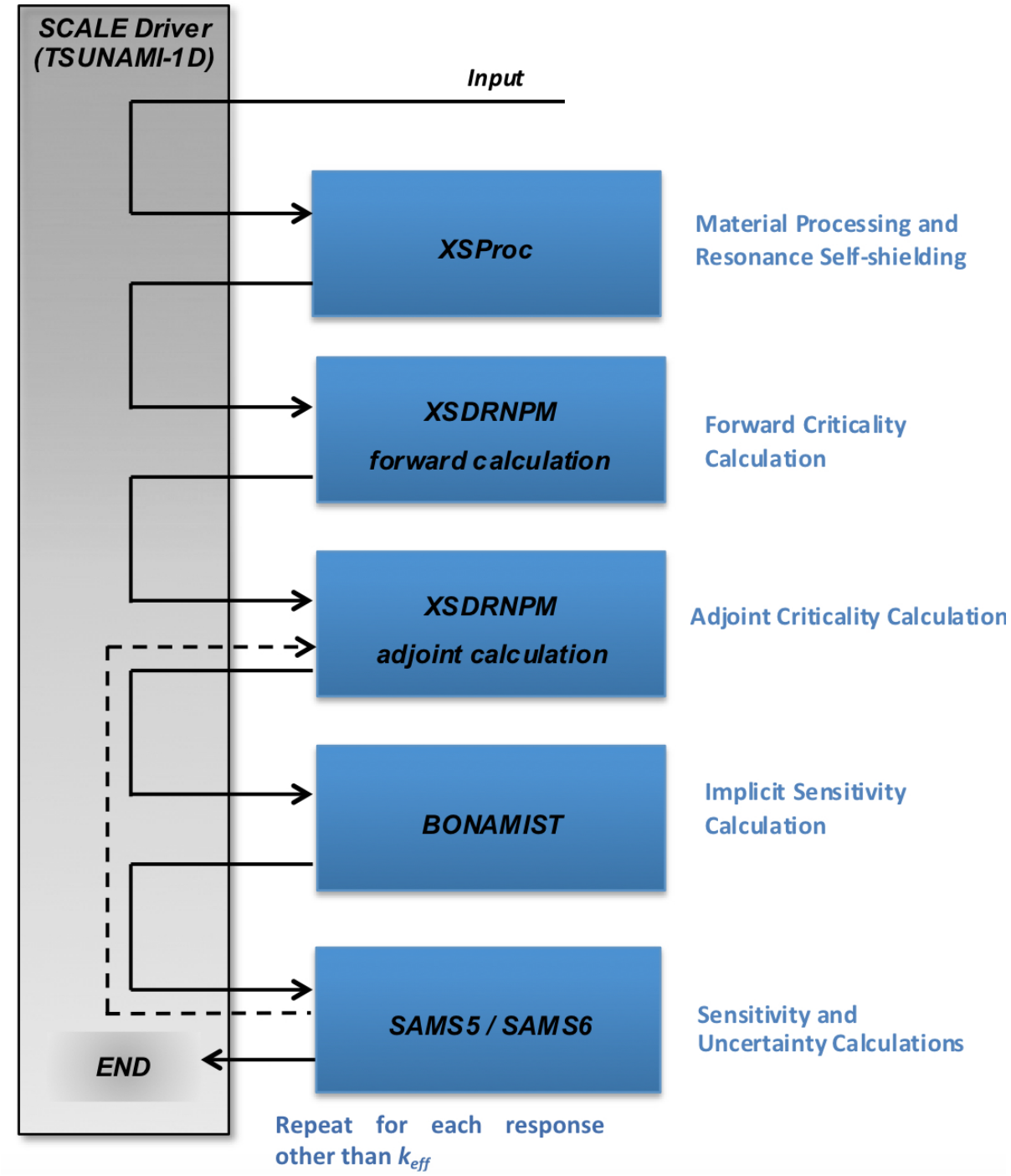


Fig. 6.1.1: General flow diagram of TSUNAMI-1D.

6.1.4 TSUNAMI-1D INPUT DESCRIPTION

The input to TSUNAMI-1D consists of a SCALE Analytical Sequence Specification Record, SCALE XSProc data, model problem data, optional sensitivity and uncertainty calculation data, and optional system response characterization data. The data for each of these segments are entered using the SCALE free-form format, allowing alphanumeric data, floating-point data, and integer data to be entered in an unstructured manner. The input is not case sensitive, so either upper- or lowercase letters may be used. A maximum of 252 columns per line may be used for input. Data can usually start or end in any column with a few exceptions. As an example, the word END beginning in column 1 and followed by two blank spaces will end the problem, any data following will be ignored. Each data entry must be followed by one or more blanks to terminate the data entry. For numeric data, either a comma or a blank can be used to terminate each data entry. Integers may be entered for floating values. For example, 10 will be interpreted as 10.0. Imbedded blanks are not allowed within a data entry unless an E precedes a single blank as in an unsigned exponent in a floating-point number. For example, 1.0E 4 would be correctly interpreted as 1.0×10^4. A comment is initiated with a single quote, `''`, and continues until the end of the input line.

6.1.4.1 Analytical sequence specification record

The analytical sequence specification begins in column 1 of the first line of the input file and must contain one of the following:

=TSUNAMI-1D
: This sequence is used for sensitivity and uncertainty calculations.

=TSUNAMI-1DC
: This sequence allows more flexibility than CSAS1 and is used for criticality calculations where the criticality problem description contains more detail than that specified in a single unit cell description.

Optional keyword input may be entered, starting after column 10 of the analytical sequence specification record. These keywords are

PARM=CHECK
: This option allows the input data to be read and checked without executing any functional modules.

PARM=CHK
: Alias for `PARM=CHECK`.

PARM=SIZE=n
: The amount of memory requested in four-byte words may be set with this entry. The default value for n is 20000000. This value only affects calculations in BONAMIST, where this value of the `SIZE` parameter is used for allocation of storage for the derivatives. Please see the documentation on BONAMIST in the Sensitivity Utility Modules chapter for more details. All other codes use dynamic memory allocation and this value has no effect.

PARM=BONAMIST
: This is the default configuration for MG TSUNAMI-1D calculations. XSProc with BONAMI and CENTRM is used for cross-section processing, and implicit sensitivities are produced with BONAMIST.

PARM=CENTRM
: XSProc with BONAMI and CENTRM is used for cross-section processing, but BONAMIST is not run. **TSUNAMI-1D sequence with `PARM=CENTRM` does not produce the implicit portions of the sensitivity coefficients, and should be used with caution.**

PARM=BONAMI

XSProc with BONAMI is used for cross-section processing, but BONAMIST is not run. **TSUNAMI-1D sequence with PARM=BONAMI does not produce the implicit portions of the sensitivity coefficients, and should be used with caution.**

PARM=2REGION

XSProc with BONAMI and CENTRM are run where Dancoff factors are to compute the escape probabilities for an accelerated, yet more approximate, CENTRM calculation. Implicit sensitivities are computed with BONAMIST.

Multiple parameters can be used simultaneously by enclosing them in parentheses and separating them with commas such as PARM=(SIZE=2000000, CHECK).

Multiple parameters can be used simultaneously by enclosing them in parentheses and separating them with commas such as PARM=(SIZE=2000000, CHECK).

6.1.4.2 XSProc

XSProc reads the standard composition specification data and the unit cell geometry specifications. It then produces the mixing table and unit cell information necessary for processing the cross sections. The XSProc chapter provides a detailed description of the input data for the Material Information Processor.

6.1.4.3 Model problem data

The model problem data are used by the TSUNAMI-1D sequences to prepare input for the XSDRNPM transport calculation. This input section consists of two data blocks, one block contains a geometry description and one contains optional parameters.

Geometry data

The TSUNAMI-1D geometry data block begins with the keywords `READ GEOM` and ends with the keywords `END GEOM`. This data block is always required. The following data is contained within this data block:

1. A line containing the geometry and boundary conditions for the XSDRNPM criticality case. The first entry on this line describes the geometry and must be SLAB, CYLINDRICAL, or SPHERICAL. The second entry is optional and describes the right-boundary condition. The default value for the right-boundary condition is VACUUM. The third entry on this line is optional and describes the left-boundary condition. The default value for the left boundary condition is REFLECTED. The last entry on this line is END. Valid entries for the boundary conditions are the following:

   VACUUM – No return at boundary

   REFLECTED – Specular (mirror-like) return at boundary

   PERIODIC – Infinite array of cells in slab geometry

   WHITE – Isotropic return at boundary

2. A line containing the following two entries for each zone of the XSDRNPM case:

   a. mixture number in the zone and

   b. zone outer dimension (in cm).

Mixture numbers and zone dimensions are entered in pairs until the entire geometry is defined. The mixture numbers must be defined in the material input processor input. Mixture 0 is used for voids, and a mixture number defined with CELLMIX= in the MIP section of the input may be used here. It should be noted that, due to a restriction in XSDRNPM, the mixture number identified with CELLMIX= may not appear in the output file, even though it is input in this section. TSUNAMI-1D automatically renumbers the cell mixed mixture to the next available mixture number for use in XSDRNPM. A message is printed in the output identifying this change. TSUNAMI-1D uses the same techniques as CSAS1X to automatically prepare a spatial mesh appropriate for the input materials and dimensions.

Parameter data

An optional data block may be entered to change parameters of the XSDRNPM forward and adjoint calculations. This data block begins with the keywords `READ PARA` or `READ PARM` and must end with either `END PARA` or `END PARM`, corresponding to the read keyword. In this data block, the user may enter optional lines that contain entries for selected XSDRNPM input parameters. A list of the parameters and their default values are found in Table 6.1.2.

Table 6.1.2: Optional parameter input for the criticality problem data.

Name	Default	Meaning
ISN=	16	Order of angular quadrature
IIM=	20	Inner-iteration maximum
ICM=	100	Outer-iteration maximum
ID1=	-1	Flux-edit option:
		-1 no flux print
		0 scalar flux print
		1 scalar and angular flux print
SCT=	5	Order of Legendre expansion for cross sections
PRT=	-2	Cross-section print option:
		-2 no cross-section print
		-1 print 1-D cross sections
		0/N print 2-D cross sections through order N
PBT=	0	Balance table print option:
		-1 no balance table print
		0 fine group balance table print
EPS=	1.E-6	Outer-iteration convergence criteria
PTC=	1.E-6	Inner-iteration convergence criteria
DY=	0	First-transverse dimension (cm) for buckling correction (i.e., height of cylinder or slab)
DZ=	0	Second-transverse dimension (cm) for buckling correction (i.e., depth of slab)
SZF=	1.5	Size factor of spatial computational mesh intervals. Increasing this number will cause the forward and adjoint XSDRNPM calculations to be conducted with larger mesh intervals and fewer mesh points. 0.0<SZF<1.5 gives a finer mesh, SZF>1.5 gives a coarser mesh.

6.1.4.4 Sensitivity and uncertainty calculation data

The data blocks for controlling the sensitivity and uncertainty calculation are optional. The optional data blocks include the SAMS block, the HTML block, the COVARIANCE block, the DEFINITIONS block, and the SYSTEMRESPONSES block. These data blocks begin with the keywords READ BLOCKNAME and end with the keywords END BLOCKNAME, where BLOCKNAME is one of SAMS, HTML, COVARIANCE, DEFINITONS, or SYSTEMRESPONSES. These data blocks can be input in any order with the following two exceptions. First, all five data blocks must appear in the input file *after* the composition and cell data blocks of data. Second, if a SAMS block is specified, the HTML and COVARIANCE data blocks must come *after* the SAMS block, if they are to be specified. In addition, both the DEFINITONS and SYSTEMRESPONSES data blocks must be present for additional analysis of system responses other than k_{eff}. If only one or both of the data blocks are omitted, then analysis is only performed for k_{eff}. The following sub-sections describe these blocks of data in detail.

Response definition data

The DEFINITIONS and SYSTEMRESPONSES blocks are used to define system responses for additional sensitivities and uncertainty analysis in SAMS. For criticality calculations, the sensitivities of system responses other than k_{eff} are calculated in TSUNAMI-1D using generalized perturbation theory (GPT). The details of the GPT methodology are provided in *General Perturbation Theory* section of the SAMS chapter. Using GPT, a *system response,* denoted R, is defined as a ratio such as:

$$R = \frac{\sum_g \int d\bar{r} H_{N,g}(\bar{r}) \phi_g(\bar{r})}{\sum_g \int d\bar{r} H_{D,g}(\bar{r}) \phi_g(\bar{r})} \tag{6.1.1}$$

In this equation, $\phi_g(\bar{r})$ is the space-dependent multi-group scalar flux and $H_{N,g}(\bar{r})$,$H_{D,g}(\bar{r})$ are referred to as the space-dependent, multi-group *response functions*. In TSUNAMI-1D, the *response functions* are specified in the DEFINITIONS data block and the *system responses* are defined in the SYSTEMRESPONSES data block. Responses (other k_{eff}) treated in TSUNAMI-1D MUST be ratios.

The DEFINITIONS data block is used by TSUNAMI-1D similarly to that of the MAVRIC and MONACO modules in SCALE. The format of the DEFINITIONS block is as follows:

```
read definitions
  response I1
    (specifications for response I1)
  end response
  response I2
    (specifications for response I2)
  end response
  ...
end definitions
```

The DEFINITIONS block of data begins with READ DEFINITIONS and terminates with END DEFINITIONS. Likewise, each response function definition begins with RESPONSE - followed by a unique, positive integer identifier - and terminates with END RESPONSE. The keyword entries summarized in Table 6.1.3 are allowed for each response specification. Keywords ending with = must be followed by the value to be assigned to the corresponding variable. All keywords are optional and can be entered in any order. However certain keywords are required depending one of the seven basic types of response functions described in the following subsections. The required keywords are summarized for each of the seven basic response function types in Table 6.1.3 at the end of this section.

Table 6.1.3: Response function keywords in DEFINITIONS block.

Keyword	Type	Default value	Description
title=	String	” “	Response function title. The title must begin and end with quotes and have a maximum of 256 characters.
macro	Logical	F	Macroscopic cross-section flag. If specified, macroscopic cross-sections are used to define the response function.
micro	Logical	T	Microscopic cross-section flag. If specified, microscopic cross-sections are used to define the response function.
nuclide= or zaid=	Integer or string	Undefined	Nuclide identifier for which cross-sections are used to define the response function. The nuclide can be specified in integer format (92235) or in character string format (u-235).
reaction= or mt=	Integer or string	Undefined	Reaction identifier for which cross-sections are used to define the response function. The reaction can be specified as an MT number (18) or as a character string (fission). Supported reaction types are listed below.
material= or mixture=	Integer	Undefined	Mixture identifier for which cross-sections are used to define the response function.
multimix ... end or multimat ... end	Integer array	Undefined	Array of mixture identifiers for which cross-sections cross-sections are used to define the response function.
unity	Logical	F	Flux response function flag. If specified, cross-sections are not used to define the response function.
multiplier	Real	1.0	Response function multiplier.
ehigh=	Real	10^{25}	Upper energy (eV) boundary of the response function.
elow=	Real	0.0	Lower energy (eV) boundary of the response function.
ehightransfer=	Real	10^{25}	Upper energy (eV) boundary used for cross-sections with secondary particle distributions (elastic, inelastic, scatter, and n,2n).
elowtransfer=	Real	0.0	Lower energy (eV) boundary used for cross-sections with secondary particle distributions (elastic, inelastic, scatter, and n,2n).

Single-mixture flux response function

A single-mixture flux response is simply the integration of the neutron flux wherever a specified mixture is defined in the problem geometry. Therefore, the response function $H_g(\bar{r})$ for a single mixture-j is defined as:

$$H_g(\bar{r}) = c^* \delta_g * \delta_j(\bar{r}) \tag{6.1.2}$$

where

$$\delta_g = \begin{cases} 1.0 & E_{\text{High}} > E_g^{\text{Lower}}, E_{\text{Low}} < E_g^{Upper} \\ 0 & \text{otherwise} \end{cases}$$

and (6.1.3)

$$\delta_j(\bar{r}) = \begin{cases} 1.0 & \text{mixture jis used at } \bar{r} \\ 0 & \text{otherwise} \end{cases}$$

In this expression, the constant c is the scalar multiplier defined by the `multiplier=` keyword. For example, the "fast" and "thermal" flux responses for mixture 5 would be:

In this example, the energy cutoff between the fast group and the thermal group is defined as 0.625 eV. Response 3 reflects the total energy-integrated flux-scaled by a factor of 2.0-because the default values of `eHigh` and `eLow` are used.

For single-mixture flux responses, keywords `unity` and `mixture` are required; `multiplier`, `eHigh`, and `eLow` are optional; `title`, `nuclide`, `reaction`, `micro`, `macro`, `eHighTransfer`, and `eLowTransfer` are optional but are not used; and `multimix` is not allowed. These keyword dependencies are outlined for each response type in Table 6.1.4.

Multiple-mixture flux response

A multiple-mixture flux response is the integration of the neutron flux wherever a set of mixtures are defined in the problem geometry. Therefore, the response function $H_g(\bar{r})$ for a set of mixtures "S" is defined as:

$$H_g(\bar{r}) = c^* \delta_g * \sum_{j \in S} \delta_j(\bar{r}) \tag{6.1.4}$$

For example, the following definition is for the energy-integrated flux response, and spatially-integrated wherever mixtures 5, 7, and 9 are used:

```
read definitions
  response 1 unity multimix 5 7 9 end end response
end definitions
```

or alternatively,

```
read definitions
  response 1 multimat 5 7 9 end unity end response
end definitions
```

For multiple-mixture flux responses, keywords `unity` and `multimix` are required; `multiplier`, `eHigh`, and `eLow` are optional; `title`, `nuclide`, `reaction`, `micro`, `macro`, `eHighTransfer`, and `eLowTransfer` are optional but are not used; and `mixture` is not allowed.

Single-mixture, single-nuclide, microscopic cross-section response

A single-mixture, single-nuclide, microscopic cross-section response is the integration of the neutron flux multiplied by a microscopic cross-section. The microscopic cross-section used in the integral is defined by a specific mixture, nuclide, and reaction type. Therefore, the response function $H_g(\bar{r})$ is defined as:

$$H_g(\bar{r}) = c^* \delta_g^* \delta_j(\bar{r})^* \sigma_{x,g}^{j,n} \tag{6.1.5}$$

In this expression, $\sigma_{x,g}^{j,n}$ is the microscopic cross-section for mixture-j, nuclide-n, reaction type-x, and energy group-g. For transfer reaction types-scatter, elastic, inelastic, and n,2n-the expression above is slightly modified so the user can define the energy range of the secondary particles, i.e.,

$$H_g(\bar{r}) = c^* \delta_g * \delta_j(\bar{r})^* \sum_{g'} \delta_{g'}^{Transfer} \sigma_{x,g\to g'}^{j,n} \tag{6.1.6}$$

where

$$\delta_g^{\text{Transfer}} = \begin{cases} 1.0 & E_{\text{HighTranser}} > E_g^{\text{Lower}}, E_{\text{LowTransfer}} < E_g^{Upper} \\ 0 & \text{otherwise} \end{cases} \tag{6.1.7}$$

Likewise, the expression for $H_g(\bar{r})$ is also modified for fission distribution responses (chi), which are usually integrated by the energy-integrated neutron production rate rather than the neutron flux:

$$H_g(\bar{r}) = c^* \delta_j(\bar{r})^* \bar{\nu}_{f,g}^{j,n} * \sigma_{f,g}^{j,n} * \sum_{g'} \delta_{g'}^* \chi_{g'}^{j,n} \tag{6.1.8}$$

For examples of this response type, the following `DEFINITION` block has response definitions for

- total nu-fission rate of U-235 in mixture 1,
- "fast" n,gamma capture rate of U-238 in mixture 1 (energy cutoff is 0.625 eV),
- downscatter rate of H-1 in mixture 2, and
- number fission neutrons born from Pu-239 fissions in mixture 1 in the intermediate energy range (E>0.625 eV and E<25 keV)

```
read definitions
  response 1
    reaction=nu-fission mixture=1 nuclide=92235
  end response
  response 2
    reaction=n,gamma mixture=1 nuclide=u-238 eLow=0.625
  end response
  response 3
    mt=0   mixture=2 zaid=1001  eLow=0.625 eHighTransfer=0.635
  end response
  response 4
    mt=chi mixture=1 zaid=pu-239 eHigh=25.0e3 eLow=0.625
  end response
end definitions
```

For single-mixture, single-nuclide microscopic cross-section responses, keywords `mixture`, `nuclide`, and `reaction` are required; `multiplier`, `eHigh`, `eLow`, `eHighTransfer`, `eLowTransfer`, and `micro`, are optional; `title` is optional but not used; and `multimix`, `macro`, and `unity` are not allowed. A list of supported cross-section reaction types is provided at the end of this section in Table 6.1.5.

Single-mixture, single-nuclide, macroscopic cross-section response

A single-mixture, single-nuclide, macroscopic cross-section response is the integration of the neutron flux multiplied by a macroscopic cross-section. The macroscopic cross-section used in the integral is defined by a specific mixture, nuclide, and reaction type. The response function $H_g(\bar{r})$ is defined as:

$$H_g(\bar{r}) = c^* \delta_g * \delta_j(\bar{r})^* \Sigma_{x,g}^{j,n} \tag{6.1.9}$$

In this expression, $\Sigma_{x,g}^{j,n}$ is the macroscopic cross-section ($N^{j,n} * \sigma_{x,g}^{j,n}$) for mixture-$j$, nuclide-$n$, reaction type-$x$, and energy group-$g$. The modifications to this expression for transfer reactions and chi are similar to that of single-mixture, single-nuclide, microscopic cross-section responses. Using the same example as above, the single-mixture, single-nuclide, macroscopic cross-section responses are given as:

```
read definitions
  response 1
    reaction=nu-fission mixture=1 nuclide=92235 macro
  end response
  response 2
    reaction=n,gamma mixture=1 nuclide=u-238 eLow=0.625 macro
  end response
  response 3
    mt=0   mixture=2 zaid=1001  eLow=0.625 eHighTransfer=0.635 macro
  end response
  response 4
    mt=chi mixture=1 zaid=pu-239 eHigh=25.0e3 eLow=0.625 macro
  end response
end definitions
```

For single-mixture, single-nuclide macroscopic cross-section responses, keywords `mixture`, `nuclide`, `macro`, and `reaction` are required; `multiplier`, `eHigh`, `eLow`, `eHighTransfer`, and `eLowTransfer`, are optional; `title` is optional but not used; and `multimix`, `micro`, and `unity` are not allowed.

Single-mixture, multiple-nuclide, macroscopic cross-section response

A single-mixture, multiple-nuclide, macroscopic cross-section response is the integration of the neutron flux multiplied by a macroscopic cross-section. The macroscopic cross-section used in the integral is defined by a specific mixture, and reaction type. The response function $H_g(\bar{r})$ is defined as:

$$H_g(\bar{r}) = c^* \delta_g * \delta_j(\bar{r})^* \Sigma^j_{x,g} \tag{6.1.10}$$

In this expression, $\Sigma^j_{x,g}$ is the mixture macroscopic cross-section defined as $\sum_n N^{j,n} * \sigma^{j,n}_{x,g}$ for mixture-j, reaction type-x, and energy group-g. The modifications to this expression for transfer reactions is similar to that defined in previous subsections. For mixture chi responses, $H_g(\bar{r})$ is rewritten as

$$H_g(\bar{r}) = c^* \delta_j(\bar{r})^* \sum_n \bar{\nu}^{j,n}_{f,g} * \Sigma^{j,n}_{f,g} * \sum_{g'} \delta_{g'} * \chi^{j,n}_{g'} \tag{6.1.11}$$

For examples of this response type, the following `DEFINITIONS` block has response definitions for

- total nu-fission rate in mixture 1,
- "fast" n,gamma capture rate in mixture 1 (energy cutoff is 0.625 eV),
- downscatter rate in mixture 2, and
- number fission neutrons born in mixture 1 in the intermediate energy range (E>0.625 eV and E<25 keV)

```
read definitions
  response 1
    reaction=nu-fission mixture=1 macro
  end response
  response 2
    reaction=n,gamma mixture=1 eLow=0.625 macro
  end response
  response 3
    mt=0   mixture=2 eLow=0.625 eHighTransfer=0.635 macro
  end response
  response 4
    mt=chi mixture=1 eHigh=25.0e3 eLow=0.625 macro
  end response
end definitions
```

For single-mixture, multiple-nuclide macroscopic cross-section responses, keywords `mixture`, `macro`, and `reaction` are required; `multiplier`, `eHigh`, `eLow`, `eHighTransfer`, and `eLowTransfer`, are optional; `title` is optional but not used; and `multimix`, `micro`, `nuclide`, and `unity` are not allowed.

Multiple-mixture, single-nuclide, macroscopic cross-section response

A multiple-mixture, single-nuclide, macroscopic cross-section response is the integration of the neutron flux multiplied by a macroscopic cross-section over a set of mixtures defined in the problem geometry. The macroscopic cross-section used in the integral is defined by a specific mixture, nuclide, and reaction type. The response function $H_g(\bar{r})$ is defined as:

$$H_g(\bar{r}) = c^* \delta_g * \sum_{j \in S} \delta_j(\bar{r})^* \Sigma_{x,g}^{j,n} \tag{6.1.12}$$

In this expression, $\Sigma_{x,g}^{j,n}$ is the macroscopic cross-section ($N^{j,n} * \sigma_{x,g}^{j,n}$) for mixture-$j$, nuclide-$n$, reaction type-$x$, and energy group-$g$. The summation of mixtures in this expression is for a set of user-defined mixtures, denoted S. The modifications to this expression for transfer reactions and chi are applied similarly to previously defined response types above.

For examples of this response type, the following `DEFINITIONS` block has response definitions for

- total nu-fission rate of U-235 in the fuel mixtures (mixtures 1,3,5)
- "fast" n,gamma capture rate of U-238 in the fuel mixtures
- downscatter rate of H-1 in the moderator mixtures (mixtures 2,4)
- number fission neutrons born in the intermediate energy range (E>0.625 eV and E<25 keV) in Pu-239 in the fuel mixtures

```
read definitions
  response 1
    reaction=nu-fission multimix 1 3 5 end macro zaid=92235
  end response
  response 2
    reaction=n,gamma multimix 1 3 5 eLow=0.625 macro zaid=u-238
  end response
  response 3
    mt=0   multimix 2 4 end eLow=0.625 eHighTransfer=0.635 macro zaid=h-1
  end response
  response 4
    mt=chi multimix 1 3 5 end eHigh=25.0e3 eLow=0.625 macro zaid=pu-239
  end response
end definitions
```

For multiple-mixture, single-nuclide macroscopic cross-section responses, keywords `multimix`, `nuclide`, `macro`, and `reaction` are required; `multiplier`, `eHigh`, `eLow`, `eHighTransfer`, and `eLowTransfer`, are optional; `title` is optional but not used; and `mixture`, `micro`, and `unity` are not allowed.

Multiple-mixture, multiple-nuclide, macroscopic cross-section response

A multiple-mixture, multiple-nuclide, macroscopic cross-section response is the integration of the neutron flux multiplied by a macroscopic cross-section over a set of mixtures defined in the problem geometry. The macroscopic cross-section used in the integral is defined by a specific mixture, and reaction type. The response function $H_g(\bar{r})$ is defined as:

$$H_g(\bar{r}) = c^* \delta_g * \sum_{j \in S} \delta_j(\bar{r})^* \Sigma_{x,g}^{j} \tag{6.1.13}$$

In this expression, $\Sigma_{x,g}^{j}$ is the mixture macroscopic cross-section for mixture-j and reaction type-x, and energy group-g. The summation of mixtures in this expression is for a set of user-defined mixtures, denoted S. The

modifications to this expression for transfer reactions and chi are applied similarly to the previously defined response types above.

For examples of this response type, the following `DEFINITIONS` block has response definitions for

- total nu-fission rate in the fuel mixtures (mixtures 1,3,5)
- "fast" n,gamma capture rate in the fuel mixtures
- downscatter rate in the moderator mixtures (mixtures 2,4)
- number fission neutrons born in the intermediate energy range (E>0.625 eV and E<25 keV) in the fuel mixtures

```
read definitions
  response 1
    reaction=nu-fission multimix 1 3 5 end macro
  end response
  response 2
    reaction=n,gamma multimix 1 3 5 eLow=0.625 macro
  end response
  response 3
    mt=0   multimix 2 4 end eLow=0.625 eHighTransfer=0.635 macro
  end response
  response 4
    mt=chi multimix 1 3 5 end eHigh=25.0e3 eLow=0.625 macro
  end response
end definitions
```

For multiple-mixture, multiple-nuclide macroscopic cross-section responses, keywords `multimix`, `macro`, and `reaction` are required; `multiplier`, `eHigh`, `eLow`, ``eHighTransfer`, and `eLowTransfer`, are optional; `title` is optional but not used; and `mixture`, `micro`, `nuclide`, and `unity` are not allowed.

Table 6.1.4: Keyword dependencies for the `DEFINITIONS` block.

Response type	**Required keywords**	**Unallowed keywords**	**Optional keywords**	**Optional, but not used keywords**
Single-mixture flux	unity, mixture	multimix	multiplier, eHigh, eLow	title, nuclide, reaction, micro, macro, eHighTransfer, eLowTransfer
Multiple-mixture flux	unity, multimix	mixture	multiplier, eHigh, eLow	title, nuclide, reaction, micro, macro, eHighTransfer, eLowTransfer
Single-mixture, single-nuclide, microscopic cross-section	mixture, nuclide, reaction	unity, macro, multimix	multiplier, eHigh, eLow, micro, eHighTransfer[a], eLowTransfer[a]	title
Single-mixture, single-nuclide, macroscopic cross-section	mixture, nuclide, reaction,macro	unity, micro, multimix	multiplier, eHigh, eLow, eHighTransfer[a], eLowTransfer[a]	title

continues on next page

Table 6.1.4 – continued from previous page

Response type	Required keywords	Unallowed keywords	Optional keywords	Optional, but not used keywords
Single-mixture, multiple-nuclide, macroscopiccross-section	mixture, reaction,macro	unity, micro, multimix, nuclide	multiplier, eHigh, eLow, eHighTransfer[a] , eLowTransfer[a]	title
Multiple-mixture, single-nuclide, macroscopic cross-section	multimix, nuclide reaction,macro	unity, micro, mixture	multiplier, eHigh, eLow, eHighTransfer[a] , eLowTransfer[a]	title
Multiple-mixture, multiple-nuclide, macroscopic cross-section	multimix, reaction,macro	unity, micro, mixture, nuclide	multiplier, eHigh, eLow, eHighTransfer[a] , eLowTransfer[a]	title
[a]Keywords `eHighTransfer` and `eLowTransfer` are only used for the following reaction types: scatter (mt=0), elastic (mt=2), inelastic (mt=4), and n,2n (mt=16) For all other reaction types, these keywords are optional, but not used				

Table 6.1.5: Supported Reaction Types in DEFINITIONS block.

MT	Reaction	String Identifier
1	total	Total
2	elastic scattering	Elastic
4	inelastic scattering	Inelastic
16[a]	effective n,2n	n,2n
0	sum of scattering (2+4+16)	Scatter
18	fission	Fission
102	n, *gamma*	n,gamma
103	n,p	n,p
104	n,d	n,d
105	n,t	n,t
106	n,^{3}he	n,he-3
107	n, *alpha*	n,alpha
101	Neutron disappearance (102+103+104+105+106+107)	capture
452	$\bar{\nu}$	nubar
1452	$\bar{\nu}$ times fission	nu-fission
1018	χ	chi

[a]The effective n,2n is defined by the summation of transfer matrices of the following reaction types: (n,2n), (n,2n+α), (n,2n+2α), (n,3n), (n,3n+α), and (n,4n). The individual transfer matrices are scaled by the number of exit channel neutrons, i.e., 2, 3, or 4.

System response definition data

The SYSTEMRESPONSES block is used to define the set of system responses for which TSUNAMI-1D will perform sensitivity and uncertainty analysis additional to k_{eff}. For SCALE 6.1, only system response ratios are supported in TSUNAMI-1D. The system response ratios are defined from the response function definitions created in the DEFINITIONS block. The format of the SYSTEMRESPONSES block is as follows:

```
read systemresponses
  ratio I1
    (specifications for response ratio I1)
  end ratio
  ratio I2
    (specifications for response ratio I2)
  end ratio
  ...
end systemresponses
```

The SYSTEMRESPONSES block of data begins with READ SYSTEMRESPONSES and terminates with END SYSTEMRESPONSES. Likewise, each system response ratio definition begins with RATIO - followed by a unique, positive integer identifier - and terminates with END RATIO. For each response ratio definition, the keywords title=, numer, and denom are allowed in any order. The title= specification is optional. However, if specified, the title must be begin and end with quotes and have a maximum of 20 characters. If omitted, the title of the ratio is "rsp ratio NNNNNNNNNN" where NNNNNNNNNN is a zero-padded 10-digit integer that is equal to the ratio identifier. The title is used as labels in both the TSUNAMI-1D text and html output. The title is also used by SAMS to generate the filename for the sensitivity data file for the ratio system response discussed further below.

The numer array is a list of integers that correlate to response function identifiers defined in the DEFINITIONS block. These response functions are added together to form the composite response function used in the numerator of the ratio. Likewise, the denom array is a list of integers that correlate to response function identifiers defined in the DEFINITIONS block. These response functions are added together to form the composite response function used in the denominator of the response ratio. Multiple response function ratios can be defined in a single input file.

For a simple example of the SYSTEMRESPONSES block, suppose the ratio system response of interest is the resonance escape probability for a given system. Using 2-group theory, this is equivalent to the following expression:

$$p = \frac{\langle \Sigma_{s,1\to 2} \rangle}{\langle \Sigma_{r,1} \rangle} = \frac{\int d\bar{r} \sum_{g\in 1} \phi_g(\bar{r}) \sum_{g'\in 2} \Sigma_{s,g\to g'}(\bar{r})}{\int d\bar{r} \sum_{g\in 1} \phi_g(\bar{r}) \Sigma_{r,g}(\bar{r})} \tag{6.1.14}$$

where $\Sigma_{r,g}(\bar{r})$ is the removal cross-section defined as the total cross-section minus the within group cross-section — $\Sigma_{t,g}(\bar{r}) - \Sigma_{s,g\to g}(\bar{r})$. The TSUNAMI-1D model uses three mixtures whose ids are 6, 7, and 10. The thermal energy cutoff is 0.625 eV.

This ratio can be defined in multiple ways. First, the ratio can be defined with three response function definitions:

```
read definitions
  response 1 title="DownScatter"
     reaction=scatter
     multimix 6 7 10 end
     macro
     eLow=0.625 eHighTransfer=0.625
  end response
  response 2 title="Fast Total"
     reaction=total
     multimix 6 7 10 end
     macro
     eLow=0.625
  end response
  response 3 title="Fast Within Group (times -1)"
     reaction=scatter
     multimix 6 7 10 end
     macro
     eLow=0.625 eLowTransfer=0.625
     factor=-1.0
  end response
end definitions
read systemresponses
  ratio 100
     title="Res Escape"
     numer 1 end
     denom 2 3 end
  end ratio
end systemresponses
```

In the above input, the numerator of the response ratio is defined by a single response function (id=1), which represents the rate at which neutrons slow down from fast energies to slow energies. The denominator of the response ratio is defined by two response functions (id=2 and id=3). The addition of these two response functions represents the "total minus within group scattering" calculation to formulate the fast neutron removal rate. In this input, the title of the response ratio is set to "Res Escape". Because only one response ratio is defined, TRITON will invoke SAMS twice, first for the k_{eff} sensitivity and uncertainty analysis and second for the analysis of the resonance escape probability. SAMS will generate two .sdf files, the first will be jobname.sdf for k_{eff} sensitivities and the second will be jobname.Res_Escape.sdf. jobname is the

name of the input file. An underscore is used to replace blanks and special characters in the response ratio title in the sdf filename.

Similarly, the resonance escape probability can be defined in a variety of different ways. For example, the numerator response function can be expressed as the sum of individual mixture downscattering rates:

```
read definitions
  response 2 title="Fast Total"
     reaction=total
     multimix 6 7 10 end
     macro
     eLow=0.625
  end response
  response 3 title="Fast Within Group (times -1)"
     reaction=scatter
     multimix 6 7 10 end
     macro
     eLow=0.625 eLowTransfer=0.625
     factor=-1.0
  end response
  response  6 mt=0 mixture= 6 macro eLow=0.625 eHighTransfer=0.625
  end response
  response  7 mt=0 mixture= 7 macro eLow=0.625 eHighTransfer=0.625
  end response
  response 10 mt=0 mixture=10 macro eLow=0.625 eHighTransfer=0.625
  end response
end definitions
read systemresponses
  ratio 100
    numer 6 7 10 end
    denom 2 3        end
  end ratio
end systemresponses
```

In this input, the numerator of the response ratio is defined by adding the individual mixture downscattering rates together. Because a title was not given for the response ratio, SAMS will generate the filename of the response ratio sdf file as `jobname.rsp_ratio_0000000100.sdf`.

SAMS data

The `SAMS` block is used for controlling certain aspects of the sensitivity and uncertainty calculation. This data block begins with the keywords `READ SAMS` and ends with the keywords `END SAMS`. Any of the optional SAMS input data may be entered in free form format between the `READ SAMS` and `END SAMS` keywords. This optional SAMS input data is shown in:

Table 6.1.6, with the default values specific to TSUNAMI-1D. Parameters used to specify default covariance data to supplement or correct values on the files specified by `coverx=` are shown in Table 6.1.7. A more detailed explanation of the SAMS parameters may be found in the SAMS chapter.

Table 6.1.6: SAMS input keywords

Keyword	Default value	Description
binsen	F	Produces SENPRO formatted binary sensitivity data file on unit 40
coverx=	56groupcov7.1	Name of covariance data file to use for uncertainty analysis
largeimp=	100.0	Value for the absolute value of implicit sensitivities, which if exceeded, will be reset to 0.0 and print a warning message.

continues on next page

Table 6.1.6 – continued from previous page

nocovar	T	Flag to cause uncertainty edit to be turned off (sets print_covar to F)
nohtml	F	Flag to cause HTML output to not be produced.
nomix	F	Flag to cause the sensitivities by mixture to be turned off
pn=	3	Legendre order for moment calculations
prtgeom	F	Flag to cause the sensitivities to be output for each geometry region
prtimp	F	Prints explicit sensitivities coefficients, implicit sensitivity coefficients and complete sensitivity coefficients
prtvols	F	Flag to cause the volumes of the regions to be printed by SAMS
unconstrainedchi	F	Flag to generate pre-SCALE 6 unconstrained chi (fission spectrum) sensitivities

Table 6.1.7: SAMS input keywords for default covariance data.

Keyword	**Default value**	**Description**
use_dcov	F	Use default covariance data
use_icov	F	Use user-input covariance data
cov_fix	F	Correct covariance data if the uncertainty is large >1000% or zero
large_cov	10.0	Relative Standard deviation to apply `cov_fix`
return_work_cov	F	Create a new covariance data file with only the cross-section covariance data used in the analysis.
udcov=	0.05	User-defined default value of standard deviation in cross-section data to use for all groups for nuclide-reaction pairs for which cross-section-covariance data are too large or not available on input covariance data library.
udcov_corr=	1.0	User-defined default correlation value to use for nuclide-reaction pairs for which cross-section-covariance data are not available on the input covariance library.
udcov_corr_type=	zone	User-defined default correlation to use for nuclide-reaction pairs for which cross-section-covariance data are not available on the input covariance library. Allowed values are `long`, `zone`, and `short`. See the table *Input Data for Covariance Block of TSAR Input* in the TSAR chapter for details on `long`, `zone`, and `short`.
udcov_therm=	0.0	User-defined default value of standard deviation in cross-section data to use for thermal data for nuclide-reaction pairs for which cross-section-covariance data are too large or not available on input covariance data library. If input, the `udcov_therm` overrides the `udcov` value in the thermal range (i.e. neutron energies below 0.625 eV).

continues on next page

Table 6.1.7 – continued from previous page

udcov_inter=	0.0	User-defined default value of standard deviation in cross-section data to use for intermediate data for nuclide-reaction pairs for which cross-section-covariance data are too large or not available on input covariance data library. If input, the udcov_inter overrides the udcov value in the intermediate range (i.e. neutron energies above 0.625 eV and below 25 keV).
udcov_fast=	0.0	User-defined default value of standard deviation in cross-section data to use for fast data for nuclide-reaction pairs for which cross-section-covariance data are too large or not available on input covariance data library. If input, the udcov_fast overrides the udcov value in the fast range (i.e. neutron energies above 25 keV).

HTML and user-input covariance data

User-defined covariance data can be specified for individual nuclides and reactions using the COVARIANCE data block. This data begins with the keywords READ COVARIANCE and ends with the keywords END COVARIANCE. Any of the optional COVARIANCE input data may be entered in free form format between the READ COVARIANCE and END COVARIANCE keywords. The specifications for the COVARIANCE data block are described in *User Input Covariance Data* of the TSUNAMI Utility Modules chapter.

As the SAMS module generates HTML output, the optional HTML data block will provides user control over some formats of the output. This data begins with the keywords READ HTML and ends with the keywords END HTML. Any of the optional HTML input data may be entered in free form format between the READ HTML and END HTML keywords. The specifications for the HTML data block are described in the TSUNAMI Utility Modules manual.

6.1.4.5 Input termination

The input specification for all TSUNAMI-1D sequences must terminate with a line containing END in column 1. This END terminates the control sequence.

6.1.5 EXAMPLE PROBLEMS

Nine TSUNAMI-1D sample problems are included in the SCALE package to verify successful installation and to provide examples for users. They are provided in the smplprbs directory of the software distribution. Three example problems are presented in this section and comparisons among the different methods for cross-section processing are discussed. The first problem presented is a variant of the TSUNAMI-1D1 k_{eff} sensitivity sample problem with some addition input parameters in the READ SAMS data block and using INFHOMMEDIUM unit cell type. The second example problem presented in this section generates k_{eff} sensitivities using the MULTIREGION unit cell type. The third example problem is similar to the TSUNAMI-1D5 sample problem that demonstrates the GPT capabilities. The five sample problems in the software package are designed to run quickly and test most code features. The three examples presented here are designed to produce accurate results, but may require more computational resources.

For all problems the validity of the sensitivity coefficients should be confirmed through the use of direct perturbation sensitivity calculations. For each sensitivity coefficient examined by direct perturbation, the k_{eff} of the system is computed first with the nominal values of the input quantities, then with a selected input

value increased by a certain percentage, and then with the value decreased by the same percentage. The direct perturbation sensitivity coefficient of k_{eff} to some input value α is computed as

$$S_{k,\alpha} = \frac{\alpha}{k} \times \frac{dk}{d\alpha} = \frac{\alpha}{k} \times \frac{k_{\alpha^+} - k_{\alpha^-}}{\alpha^+ - \alpha^-}, \tag{6.1.15}$$

whereα^+ and α^- represent the increased and decreased values, respectively, of the input quantity α and k_{α^+} and k_{α^-} represent the corresponding values of k_{eff}.

The use of direct perturbation calculations to confirm the validity of sensitivity coefficients is strongly encouraged. Inconsistent modeling between the resonance-self shielding input and the criticality problem description can lead to erroneous sensitivity results. These erroneous results can be revealed through the use of direct perturbation confirmation of the energy-integrated sensitivity results for the total cross section. The total cross-section sensitivities are equivalent to number density sensitivities on an energy-integrated basis.

The results shown here were generated with a previous version of SCALE, so current data libraries and code implementations may product different results. However, the techniques demonstrated are applicable to the current version of TSUNAMI-1D.

6.1.5.1 INFHOMMEDIUM sample problem

The selected sample problem with INFHOMMEDIUM cross-section processing is based on an unreflected rectangular parallelepiped consisting of a homogeneous mixture of UF_4 and paraffin with an enrichment of 2% in ^{235}U. The H/^{235}U atomic ratio is 294:1. The dimensions of the experiment were 56.22 cm × 56.22 cm × 122.47 cm. [TS-1D-CLP86]. For the purposes of this exercise, this experiment was modeled as a sphere with a critical radius of 38.50 cm. This model is consistent with SCALE sample problem TSUNAMI-1D1, which utilizes the 238-group ENDF/B-VII cross-section library, and the default cross-section processing with BONAMIST and CENTRM/PMC/WORKER.

An annotated TSUNAMI-1D1 input for this experiment is shown in Sect. 6.1.3.1. The composition data is input as number densities for each nuclide. Because the material is treated as INFHOMMEDIUM, no explicit unit cell model is necessary, and the READ CELL block is omitted. The criticality description contains optional parameter data to change the default S_{16} angular quadrature set to S_8. The change in angular quadrature is made only to demonstrate the input capabilities of TSUNAMI-1D and has little effect on this calculation. The criticality problem geometry uses a spherical coordinate system with the default boundary conditions (reflected left, vacuum right). The system consists of a single material zone containing mixture 1 with a radius of 38.50 cm. The optional sensitivity calculation data block was entered to request the extended edit of sensitivity by material zone (`prtgeom`), the extended edits of the explicit, implicit and complete sensitivity coefficients (`prtimp`), and corrections in the cross-section covariance data (`use_dcov`, `cov_fix`).

Prior to producing the output of the functional modules, TSUNAMI-1D produces output from the XSProc routines as it is processing the user input and creating internal inputs for the resonance processing codes. TSUNAMI-1D also prints information regarding the criticality description.

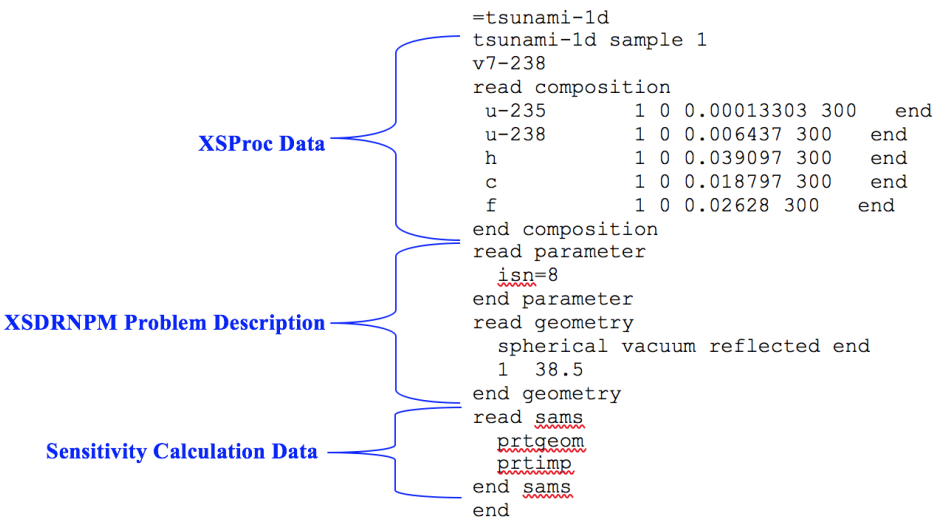


Fig. 6.1.2: TSUNAMI-1D INFHOMMEDIUM sample problem input.

For this problem, direct perturbation results were obtained for the number densities of each nuclide. In these calculations, the number density of each nuclide was perturbed by ±2% and the calculation was repeated using the TSUNAMI-1DC sequence. The sensitivity of k_{eff} to the number density is equivalent to the sensitivity of k_{eff} to the total cross section, integrated over energy. The direct perturbation sensitivity coefficients were computed by using the k_{eff} values from the unperturbed and perturbed cases in Eq. (6.1.15). To demonstrate the importance of the sensitivity to the resonance processing implicit sensitivity computed by BONAMIST, the same model shown in Fig. 6.1.2 was run with TSUNAMI-1D with `PARM=CENTRM`. The results from the `INFHOMMEDIUM` sample problem are given in Table 6.1.8. The TSUNAMI-1D results using the default codes for resonance processing show good agreement with the direct perturbation results for all nuclides. Due to omission of the implicit terms, the TSUNAMI-1D results with `PARM=CENTRM` do not show good agreement with the direct perturbation for this thermal system. The maximum difference between the direct perturbation results and the TSUNAMI-1D results occurs for ^{238}U with a magnitude of 1.5%. The maximum difference between the direct perturbation results and the TSUNAMI-1D with `PARM=CENTRM` results occurs for ^{238}U with a magnitude of 19%. Thus, the use of the default `PARM=BONAMIST` is recommended.

Table 6.1.8: Energy- and region-integrated sensitivity coefficients from TSUNAMI-1D INFHOMMEDIUM sample problem.

Isotope	**Reaction**	**Direct perturbation**	**TSUNAMI-1D**	**TSUNAMI-1D PARM=**CENTRM**
^{1}H	total	2.20E-01	2.18E-01	2.52E-01
^{1}H	scatter		3.19E-01	3.53E-01
^{1}H	elastic		3.19E-01	3.53E-01
^{1}H	capture		-1.01E-01	-1.01E-01
^{1}H	n, *gamma*		-1.01E-01	-1.01E-01
^{12}C	total	2.41E-02	2.38E-02	2.76E-02
^{12}C	scatter		2.45E-02	2.83E-02
^{12}C	elastic		2.43E-02	2.80E-02

continues on next page

Table 6.1.8 – continued from previous page

Isotope	Reaction	Direct perturbation	TSUNAMI-1D	TSUNAMI-1D PARM=CENTRM**
^{12}C	n,n'		2.20E-04	2.20E-04
^{12}C	capture		-6.83E-04	-6.83E-04
^{12}C	n, *gamma*		-4.98E-04	-4.98E-04
^{12}C	n,p		-3.53E-08	-3.53E-08
^{12}C	n,d		-7.33E-08	-7.33E-08
^{12}C	n, *alpha*		-1.85E-04	-1.85E-04
^{19}F	total	4.10E-02	4.06E-02	4.47E-02
^{19}F	scatter		4.62E-02	5.03E-02
^{19}F	elastic		2.94E-02	3.34E-02
^{19}F	n,n'		1.58E-02	1.58E-02
^{19}F	n,2n		2.89E-06	2.89E-06
^{19}F	capture		-5.59E-03	-5.59E-03
^{19}F	n, *gamma*		-2.39E-03	-2.39E-03
^{19}F	n,p		-2.37E-04	-2.37E-04
^{19}F	n,d		-1.27E-05	-1.27E-05
^{19}F	n,t		-2.72E-06	-2.72E-06
^{19}F	n, *alpha*		-2.96E-03	-2.96E-03
^{235}U	total	2.52E-01	2.52E-01	2.53E-01
^{235}U	scatter		4.32E-04	5.03E-04
^{235}U	elastic		2.02E-04	2.73E-04
^{235}U	n,n'		2.13E-04	2.13E-04
^{235}U	n,2n		1.70E-05	1.70E-05
^{235}U	fission		3.64E-01	3.65E-01
^{235}U	capture		-1.13E-01	-1.12E-01
^{235}U	n, *gamma*		-1.13E-01	-1.12E-01
^{235}U	nubar		9.50E-01	9.50E-01
^{235}U	χ		8.52E-08	8.52E-08
^{238}U	total	-2.08E-01	-2.05E-01	-2.47E-01
^{238}U	scatter		4.81E-02	2.46E-02
^{238}U	elastic		3.46E-02	1.10E-02
^{238}U	n,n'		1.25E-02	1.25E-02
^{238}U	n,2n		1.02E-03	1.02E-03
^{238}U	fission		3.35E-02	3.35E-02
^{238}U	capture		-2.86E-01	-3.05E-01
^{238}U	n, *gamma*		-2.86E-01	-3.05E-01
^{238}U	nubar		5.02E-02	5.02E-02
^{238}U	χ		4.54E-09	4.54E-09

The uncertainty information from SAMS for the INFHOMMEDIUM sample problem is shown in Example 6.1.1. Based on the 44GROUPCOV covariance data library, documented in the COVLIB chapter, the uncertainty in k_{eff} due to these covariance data is 0.6064% $\Delta k/k$. A more detailed description of the uncertainty information is given in Chapter 6.3. Some plots of the energy-dependent sensitivity data were generated with Fulcrum. The energy-dependent data is available in the sensitivity data file, which is returned

to the same directory as the input file and given the same name as the user's input file with the extension .sdf. Energy-dependent sensitivity profiles for ^{235}U fission and ^{1}H elastic scattering are shown in Fig. 6.1.3. The ^{238}U capture sensitivity profiles from TSUNAMI-1D and TSUNAMI-1D with PARM=CENTRM are shown in Fig. 6.1.4. The effect of the implicit component of the sensitivity coefficients can be seen in the resonance region in the difference between the TSUNAMI-1D and TSUNAMI-1D PARM=CENTRM profiles.

Example 6.1.1: INFHOMMEDIUM sample problem.

```
-----------------------------
   Uncertainty Information
-----------------------------

 the relative standard deviation of k-eff (% delta-k/k) due
 to cross-section covariance data is:

   0.6064 % delta-k/k

  contributions to uncertainty in k-eff (% delta-k/k) by
  individual energy covariance matrices:

                        covariance matrix
      nuclide-reaction         with        nuclide-reaction            % delta-k/k due to this matrix
 -----------------------------       -----------------------------  ---------------------------------
        u-238 n,gamma                       u-238 n,gamma                3.8595E-01
        u-235 nubar                         u-235 nubar                  2.8506E-01
        u-238 n,n'                          u-238 n,n'                   2.1331E-01
        u-235 n,gamma                       u-235 n,gamma                1.5963E-01
         f-19 elastic                        f-19 elastic                1.3392E-01
        u-238 elastic                       u-238 n,n'                  -1.2469E-01
        u-235 fission                       u-235 n,gamma                1.2396E-01
        u-235 fission                       u-235 fission                1.2185E-01
          h-1 elastic                         h-1 elastic                1.1625E-01
         f-19 elastic                        f-19 n,n'                  -1.1598E-01
         f-19 n,n'                           f-19 n,n'                  1.1072E-01
        u-235 chi                           u-235 chi                    8.4524E-02
        u-238 elastic                       u-238 elastic                6.8573E-02
        u-238 nubar                         u-238 nubar                  5.8699E-02
          h-1 n,gamma                         h-1 n,gamma                5.0686E-02
        u-238 elastic                       u-238 n,gamma                4.9596E-02
         f-19 n,alpha                        f-19 n,alpha                1.9853E-02
        u-238 fission                       u-238 fission                1.7402E-02
            c elastic                           c elastic                1.5259E-02
        u-238 n,2n                          u-238 n,2n                   1.3655E-02
         f-19 n,gamma                        f-19 n,gamma                9.7725E-03
            c n,n'                              c elastic                -8.8958E-03
            c n,n'                              c n,n'                   8.4710E-03
         f-19 elastic                        f-19 n,alpha                6.6444E-03
        u-238 chi                           u-238 chi                    5.6329E-03
        u-235 elastic                       u-235 n,gamma                4.4651E-03
        u-235 elastic                       u-235 fission                -3.2889E-03
        u-238 fission                       u-238 n,gamma                2.7666E-03
         f-19 n,p                            f-19 n,p                    2.0768E-03
        u-238 elastic                       u-238 n,2n                   -1.8932E-03
        u-238 elastic                       u-238 fission                -1.8189E-03
            c n,alpha                           c n,alpha                1.6172E-03
            c n,gamma                           c n,gamma                1.4880E-03
        u-235 n,n'                          u-235 n,n'                   1.3414E-03
        u-235 elastic                       u-235 n,n'                   -8.6275E-04
         f-19 elastic                        f-19 n,p                    5.8397E-04
         f-19 elastic                        f-19 n,gamma                4.5179E-04
        u-235 elastic                       u-235 elastic                4.3646E-04
         f-19 n,d                            f-19 n,d                    2.8169E-04
        u-235 n,2n                          u-235 n,2n                   1.5476E-04
            c n,n'                              c n,alpha                -1.4865E-04
```

(continues on next page)

(continued from previous page)

f-19 elastic	f-19 n,2n	-7.0280E-05
f-19 elastic	f-19 n,d	6.6324E-05
f-19 n,t	f-19 n,t	6.5613E-05
u-235 elastic	u-235 n,2n	-2.7763E-05
f-19 n,2n	f-19 n,2n	2.2764E-05
f-19 n,n'	f-19 n,2n	-1.9276E-05
f-19 elastic	f-19 n,t	1.4593E-05
c n,n'	c n,gamma	6.9724E-06
c n,d	c n,d	8.5422E-07
c n,p	c n,p	4.5780E-07
c n,n'	c n,d	-3.2157E-07
c n,n'	c n,p	-1.5591E-07

Note: relative standard deviation in k-eff can be computed from individual values by adding the square of the values with positive signs and subtracting the square of the values with negative signs, then taking the square root

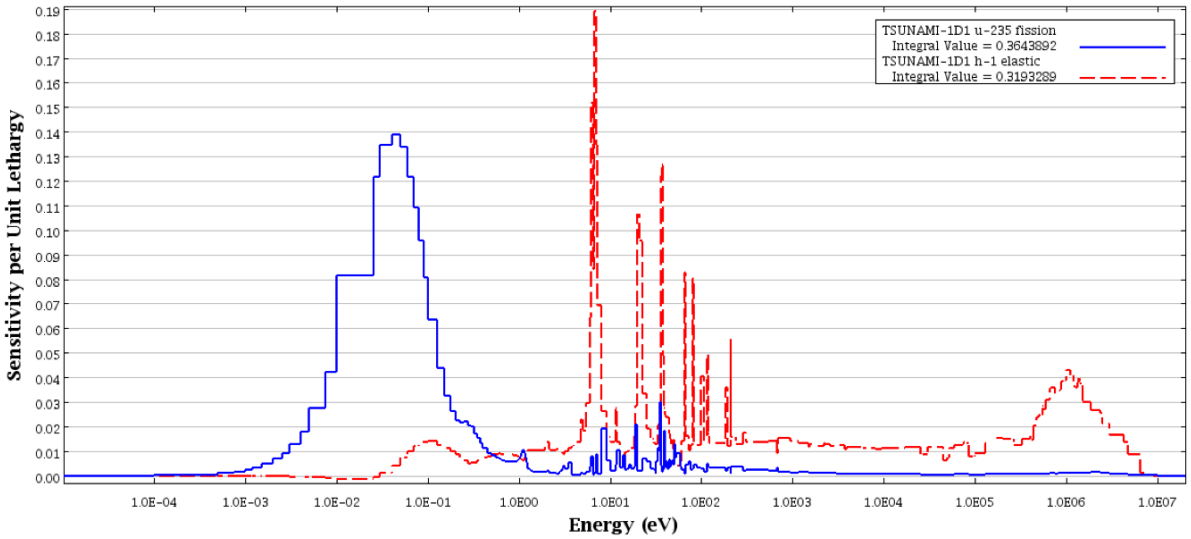


Fig. 6.1.3: Energy-dependent sensitivity profiles from TSUNAMI-1D for INFHOMMEDIUM sample problem.

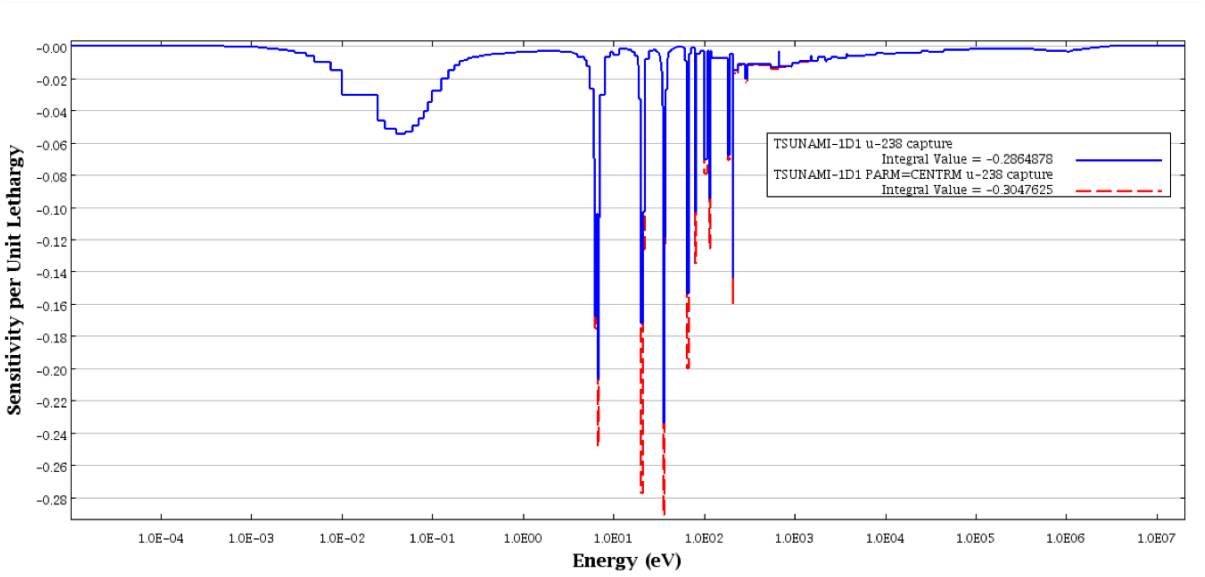


Fig. 6.1.4: Comparison of 238U capture sensitivities from TSUNAMI-1D and TSUNAMI-1D with PARM=CENTRM for INFHOMMEDIUM sample problem.

6.1.5.2 Multiregion sample problem

The sample problem selected to demonstrate the use of TSUNAMI-1D with MULTIREGION cross-section processing is the FLATTOP-25 metal system from the Cross-Section Evaluation Working Group benchmark specifications. [TS-1D-AKPZ74]. The system consists of a 6.116-cm sphere of 93%-enriched uranium with a natural uranium reflector. The outer radius of the reflector is 24.13 cm. The system is used for sample problems TSUNAMI-1D4 – TSUNAMI-1D7. For this example, input for TSUNAMI-1D4 was modified to use the SCALE 238-group ENDF/B-VII library with multiregion cell data as shown in Fig. 6.1.5. The multiregion cell data processes the cross sections in the same geometry as the criticality model. Therefore, the dimensions of the system are input twice in this model: once in the unit cell specification portion of the input and once in the criticality portion of the input. The unit cell specification geometry is used to generate input for BONAMIST and CENTRM/PMC/WORKER, and the criticality model is used to generate input for the forward and adjoint XSDRNPM calculations. The optional sensitivity calculation data block was entered to request the extended edit of sensitivity by material zone (`prtgeom`), the extended edits of the explicit, implicit and complete sensitivity coefficients (`prtimp`), and to allow larger implicit sensitivity values to be computed without producing warning messages (`largeimp=1000`).

This model was executed with TSUNAMI-1D and also with TSUNAMI-1D with PARM=CENTRM. Direct perturbation sensitivity results were obtained for the number densities of all nuclides, which correspond to the sensitivity of k_{eff} to the total cross section, integrated over energy. The energy-integrated sensitivity results are shown in Table 6.1.9. The TSUNAMI-1D results agree well with the direct perturbation results for this system. The maximum difference occurs for ^{238}U in the reflector region with a magnitude of 0.9%. Because this is a fast system, the effect of the resonance processing on the sensitivity coefficients is minimal. Thus, the TSUNAMI-1D PARM=CENTRM results are almost identical to the default TSUNAMI-1D results with BONAMIST.

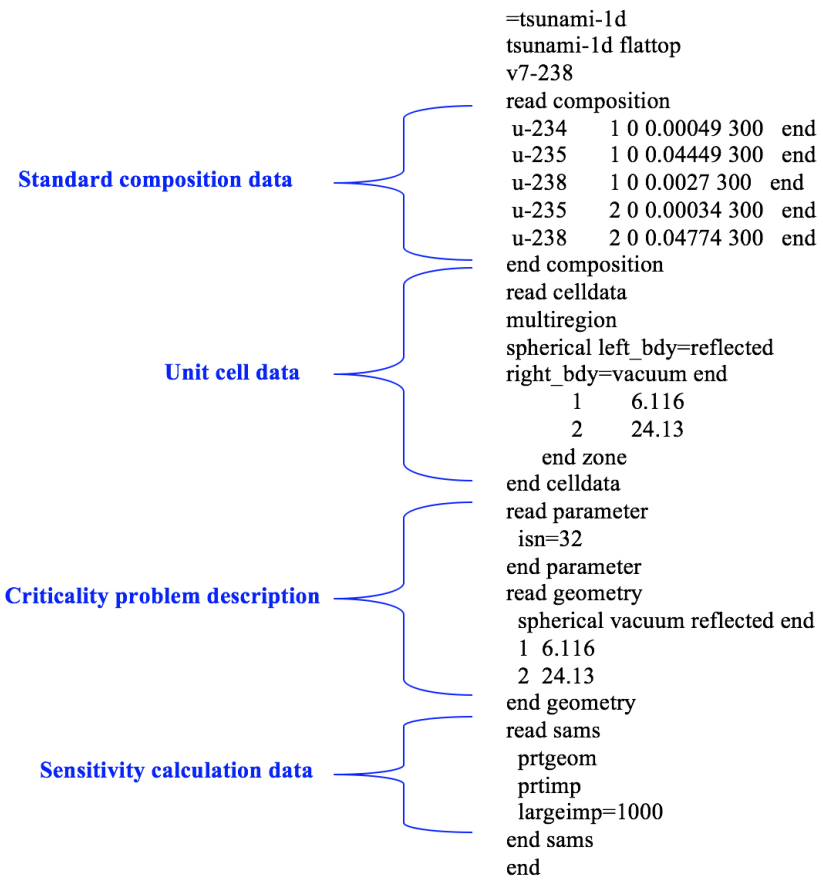


Fig. 6.1.5: TSUNAMI-1D MULTIREGION sample problem input.

Table 6.1.9: Energy- and region-integrated sensitivity coefficients from TSUNAMI-1D MULTIREGION sample problem.

PLOT Format: READ PLOT plot parameters END PLOT plot parameters must be entered for each plot that is to be made.
See sections Plot data and Color plots.

KEYWORD	DEFAULT	DEFINITION
TTL=	prob. title	delim ptitl delim delim is a one-character delimiter (i.e. ", ', *, ^, :, or !) that signals the beginning and end of the title.
		ptitl is the plot title (max. 132 char.)
PIC=	MAT	Type of plot: MIXTURE NO., UNIT NO. or BIAS ID NO.
		MIXTURE ----------MAT MIX MIXT MIXTURE MEDI MEDIA
		UNIT NO. ----------UNT UNIT UNITTYPE
		BIAS ID NO. -------IMP BIAS BIASID WTS WEIG WEIGHTS WGT WGTS
XUL=	prev. plot	X coord. of upper left corner of plot
YUL=	prev. plot	Y coord. of upper left corner of plot
ZUL=	prev. plot	Z coord. of upper left corner of plot
XLR=	prev. plot	X coord. of lower right corner of plot
YLR=	prev. plot	Y coord. of lower right corner of plot
ZLR=	prev. plot	Z coord. of lower right corner of plot

KEYWORD	DEFAULT	DEFINITION
UAX=	prev. plot 0 If VAX OR WAX is read	X component of direction cosine for the AX axis of the plot (across)
VAX=	prev. plot 0 IF UAX OR WAX is read	Y component of direction cosine for the AX axis of the plot (across)
WAX=	prev. plot 0 IF UAX OR VAX is read	Z component of direction cosine for the AX axis of the plot (across)
UDN=	prev. plot 0 IF VDN OR WDN is read	X component of direction cosine for the DN axis of the plot (down)
VDN=	prev. plot 0 IF UDN OR WDN is read	Y component of direction cosine for the DN axis of the plot (down)
WDN=	prev. plot 0 IF UDN OR VDN is read	Z component of direction cosine for the DN axis of the plot (down)
DLX=		Horizontal spacing between points on plot
DLD=		Vertical spacing between points on plot
NAX=		No. of intervals to be printed across page
NDN=		No. of intervals to be printed down page
LPI=	8.0 (character plots) 10 (color plots)	Vertical to horizontal scaling factor for plot proportionality.
RUN=	YES	YES allows the problem to execute NO terminates problem after data checking
PLT=	YES	YES allows the plot(s) to be made NO allows reading the plot data without making a plot
SCR=	YES	Display plot method SCR=YES utilizes color plot display SCR=NO utilizes printer plot display
NCH=	CHRS[c]	delim CHRS delim a one character delimiter[b] signals the beginning and end of the character string
CLR=	See Table on default color specifications for the color plot display method	num(i) red(num(i)) green (num(i)) blue (num(i)) num(i) defines mix. no., unit no., or bias ID next 3 entries define red, green, and blue components of the color representing num(i).

PLOT ORIGIN:
(1) SINGLE UNIT - coincides with origin of geometry description.
(2) BASE ARRAY - at the most negative point of the global array
(3) REFLECTED ARRAY - coincides with the origin of the CORE or ARRAY description of the global array.

[c] Default values of CHRS are given below:

MEDIA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
CHRS		1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	I	J	K	L	M	
MEDIA	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42				
CHRS		N	O	P	Q	R	S	T	U	V	W	X	Y	Z	@		$	.	<	(	+			
MEDIA	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58								
CHRS	&	>	:	;	,	-	%	*	"	=	!	(	@	<	/	0								

[b] Acceptable delimiters are ", ', *, ^, or !.

The uncertainty information from SAMS HTML output for the multiregion sample problem is shown in Fig. 6.1.6. Based on the 44GROUPCOV covariance data file, the uncertainty in k_{eff} due to these covariance data is 1.2743% $\Delta k/k$. The contributions to this uncertainty are listed by nuclide. These data are explained in more detail in the SAMS chapter.

Sensitivity profiles from TSUNAMI-1D for ^{235}U fission in zone 1 (core) and zone 2 (reflector) were generated with Fulcrum and are shown in Fig. 6.1.7. Additionally, sensitivity profiles for ^{238}U capture in zone 1 and zone 2 are shown in Fig. 6.1.8. Note that the capture sensitivities are negative, such that the lower curve has the greater magnitude. In ^{235}U and ^{238}U sensitivity profiles, the effect of the differing enrichments in the core and the reflector of this system are demonstrated with the much greater sensitivity to ^{235}U fission in the core and to ^{238}U capture in the reflector.

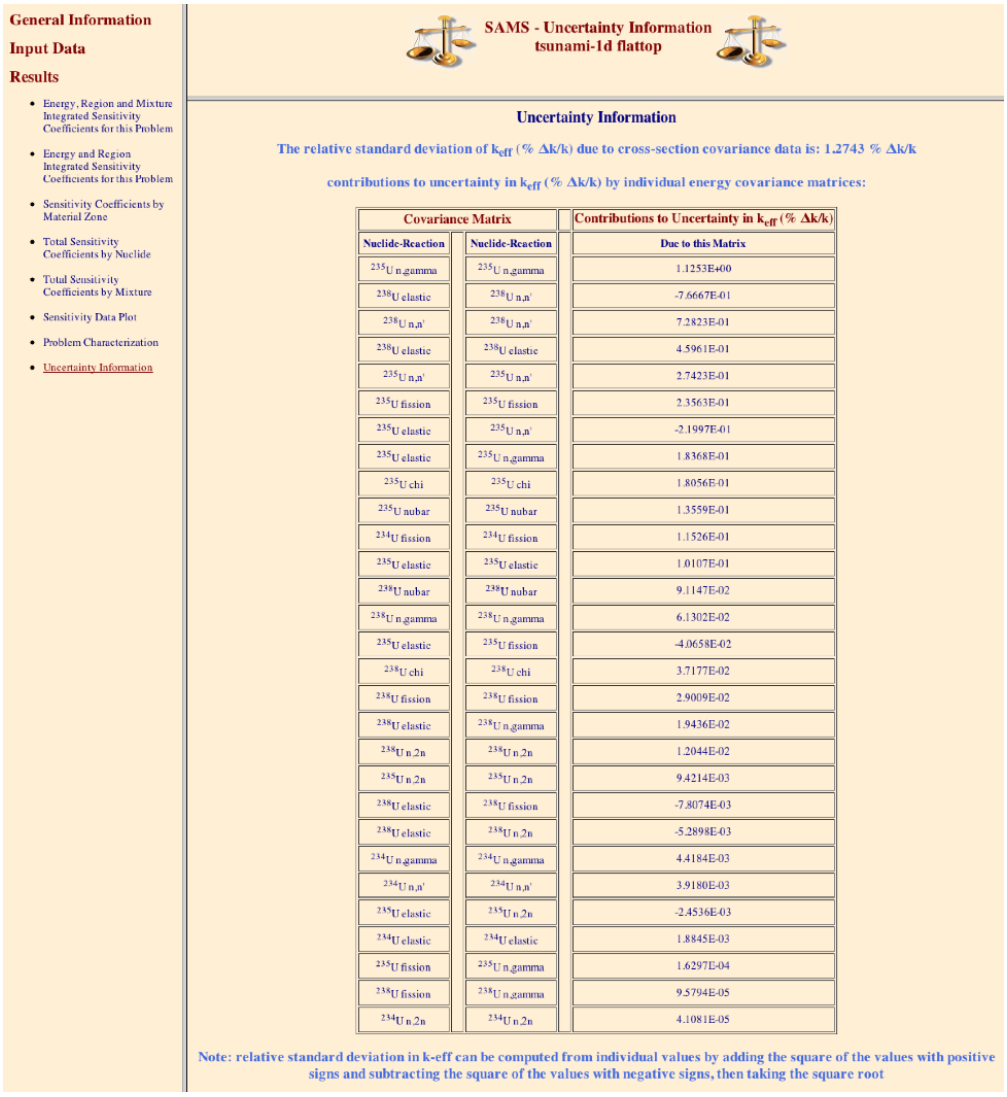


Fig. 6.1.6: Uncertainty information in HTML output from MULTIREGION sample problem.

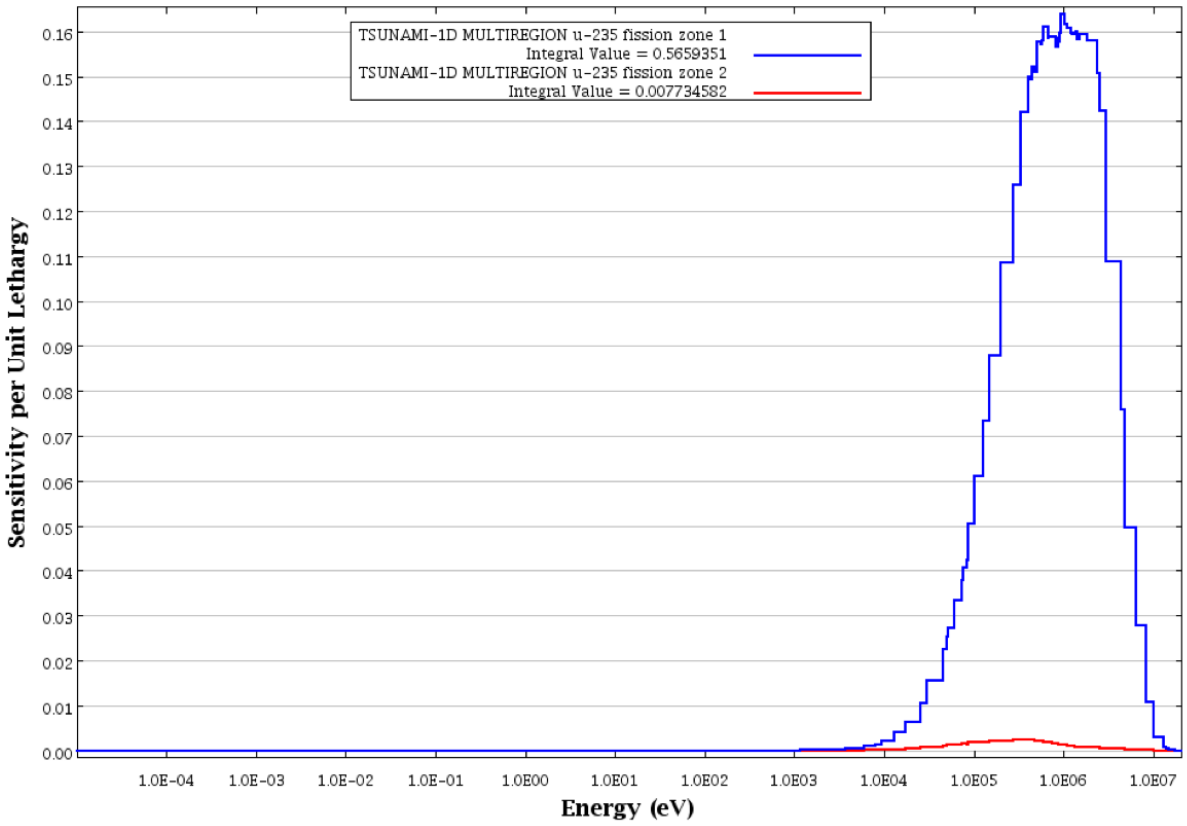


Fig. 6.1.7: Sensitivity profiles from TSUNAMI-1D for ^{235}U fission in zone 1 and zone 2 of MULTIREGION sample problem.

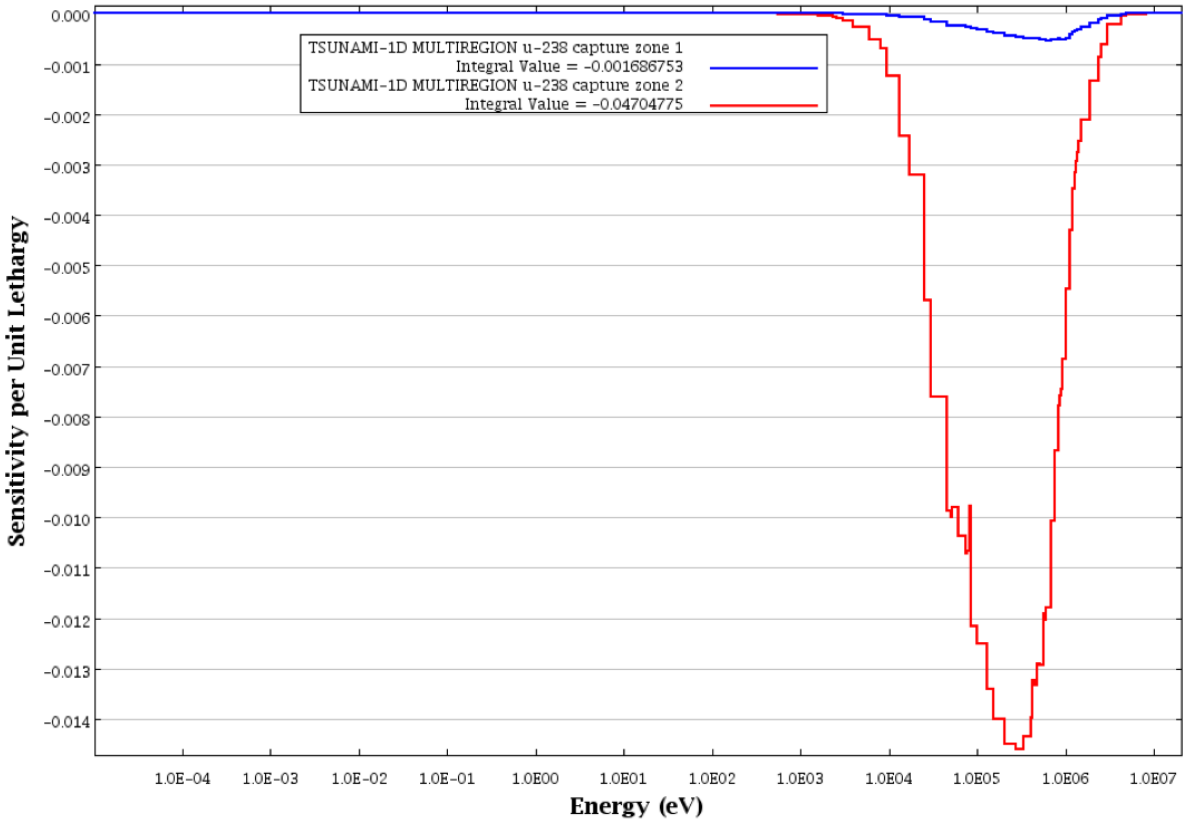


Fig. 6.1.8: Sensitivity profiles from TSUNAMI-1D for ^{238}U capture in zone 1 and zone 2 of MULTIREGION sample problem.

6.1.5.3 GPT sample problem

The sample problem selected to demonstrate the use of TSUNAMI-1D with Generalized Perturbation theory is from the OECD LWR Uncertainty Analysis in Modeling benchmark specification [TS-1D-IAKS07]. The system consists of a 4.85% enriched uranium PWR fuel pin modeled at 551 K. This system is used for sample problem TSUNAMI-1D9. For this example, the `DEFINITIONS` and `SYSTEMRESPONSES` blocks are used to define six additional response ratios for sensitivity and uncertainty analysis. The requested responses in the benchmark were for the energy-integrated fission and absorption microscopic cross-sections for ^{234}U, ^{235}U, and ^{238}U. The input for this sample problem is provided in Example 6.1.2. For this sample, seven separate sensitivity data files are generated, one for each of the six defined responses in addition to k_{eff}. Selected sensitivity profiles are shown in Fig. 6.1.9 for the ^{238}U (n,γ) cross-section. This figure shows the negative sensitivity of k_{eff} due to ^{238}U resonance absorption in the blue profile. The red profile shows the positive sensitivity of the energy-integrated ^{238}U absorption cross-section due to the multigroup ^{238}U (n,γ) cross-section. The large positive magnitude of this sensitivity is predominantly due to the presence of the ^{238}U (n,γ) cross-section directly in the definition of the response ratio. In contrast, the black sensitivity profile shows the negative sensitivity of the energy-integrated ^{235}U fission cross-section due to the multigroup ^{238}U (n,γ) cross-section. In this response, positive perturbations to the ^{238}U (n,γ) multigroup cross-sections induce changes in the flux spectra that lead to a decrease in the energy-integrated ^{235}U fission cross-section. These indirect sensitivity effects are determined by the solution of the generalized adjoint calculations.

Example 6.1.2: Input for TSUNAMI-1D9 sample problem.

```
=tsunami-1d
PWR Unit Cell
v7-238
read comp
'fuel
uo2    10 den=10.283   1 551.0 92235  4.85 92234 0.045 92238 95.105 end
zirc4 20               1 551.0 end
h2o    30 den=0.766    1 551.0 end
he     40 den=0.00125 1 551.0 end
end comp
read celldata
latticecell squarepitch pitch=1.4427 30 fueld=0.9391 10 cladd=1.0928 20 gapd=0.9582 40  end
end celldata
read geom
cylindrical white reflected end
10 .46955 40 .4791 20 .5464 30 .813956
end geom
read definitions
  response 1 nuclide=92234 mt=102 mixture=10 micro end response
  response 2 nuclide=92234 mt= 18 mixture=10 micro end response
  response 3 nuclide=92235 mt=102 mixture=10 micro end response
  response 4 nuclide=92235 mt= 18 mixture=10 micro end response
  response 5 nuclide=92238 mt=102 mixture=10 micro end response
  response 6 nuclide=92238 mt= 18 mixture=10 micro end response
  response 7 unity multimix 10 20 30 40 end end response
end definitions
read systemresponses
  ratio 1 numer 1 2 end denom 7 end title='U234-abs' end ratio
  ratio 2 numer   2 end denom 7 end title='U234-fis' end ratio
  ratio 3 numer 3 4 end denom 7 end title='U235-abs' end ratio
  ratio 4 numer   4 end denom 7 end title='U235-fis' end ratio
  ratio 5 numer 5 6 end denom 7 end title='U238-abs' end ratio
  ratio 6 numer   6 end denom 7 end title='U238-fis' end ratio
end systemresponses
end
```

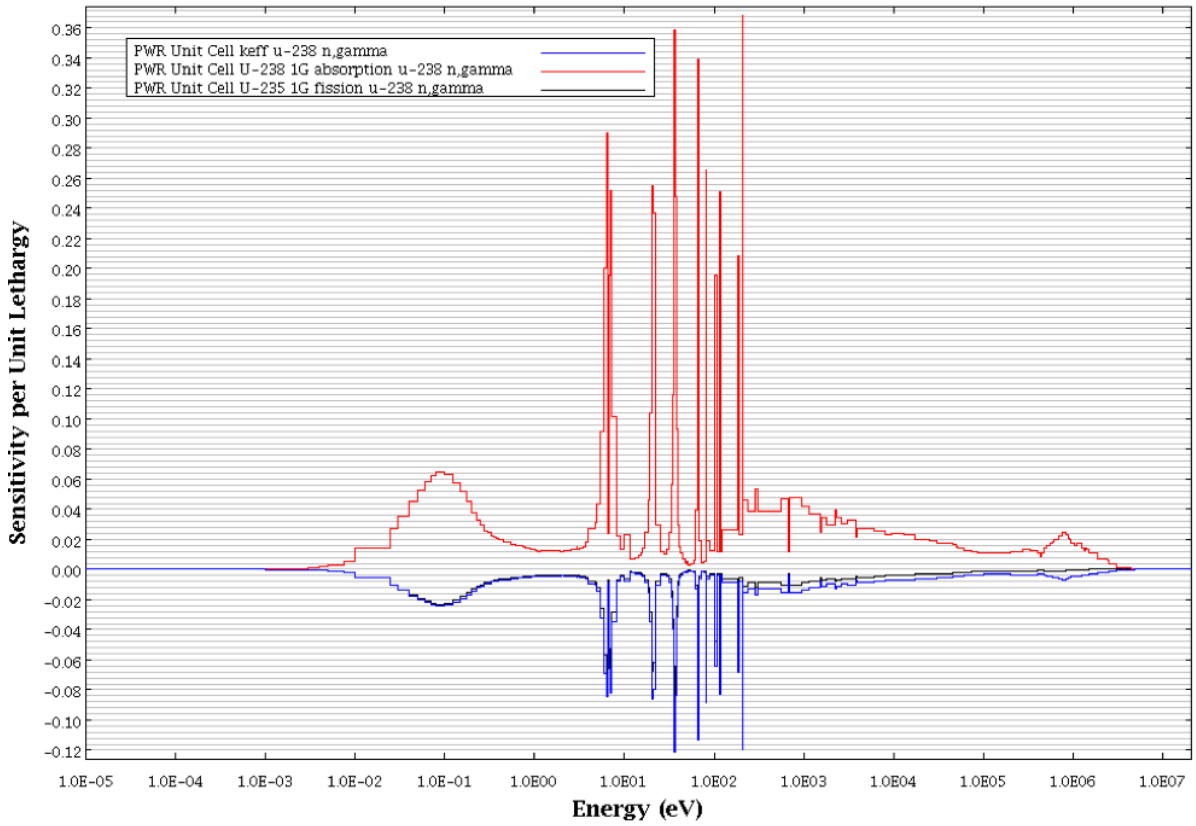


Fig. 6.1.9: Response Sensitivities to ^{238}U n,gamma cross section for the TSUNAMI-1D9 sample problem.

6.1.6 APPENDICES

6.1.6.1 XSDRNPM Data File Formats

As part of the development of the sensitivity capabilities in SCALE, the XSDRNPM module was modified to allow the writing of a new interface file containing information needed for sensitivity calculations. This same data file is used by TSUNAMI-1D. This information is written on unit NTD in the XSDRNPM input, when NTD is positive. When TSUNAMI-1D creates the inputs for the forward and adjoint XSDRNPM criticality calculations, NTD is set to 31 for the forward case and 32 for the adjoint case. The files written are ft31f001 and ft32f001, respectively.

6.1.6.2 XSDRNPM forward output file

For a forward case, XSDRNPM writes the following unformatted records on unit NTD:

RECORD 1: IZM,IM,MXX,MS,ISCT,MM,JT,IGM

IZM - Number of zones
IM - Number of spatial intervals
MXX - Number of compositions (mixtures)

MS - Length of the XSDRNPM mixing table
ISCT - Order of the Legendre scattering expansion
MM - Number of angles in the angular quadrature
JT - Number of flux moments
IGM - Number of energy groups

RECORD 2: IGE,IBL,IBR,ISN,IFTG,MMT,NT1,T

IGE - Geometry: 1/2/3 = plane/cylinder/sphere
IBL - Left-boundary condition: 0/1/2/3 = vacuum/reflected/periodic/white
IBR - Right-boundary condition
ISN - S_n quadrature order
IFTG - First thermal group
MMT - Number of neutron groups
NT1 - Unit number of working cross-section library
T - Problem title containing 80 characters

RECORD 3: V,R

V(IM) - Volumes of the spatial mesh cells (single-precision)
R(IM+1) - Boundaries of the spatial mesh cells (single-precision)

RECORD 4: W,PNC

W(MM) - Weights in the angular quadrature (single-precision)
PNC(MM,JT) - Scattering constants used to obtain flux moments from angular fluxes (single-precision)

RECORD 5: MA,MZ

MA(IM) - Zone number by interval
MZ(IZM) - Mixture number by zone

RECORD 6: MB,MC,XMD

MB(MS) - Mixture number in the cross-section mixing table
MC(MS) - Component (nuclide) in the cross-section mixing table
XMD(MS) - Atom density in the cross-section mixing table (single-precision)

RECORD 7: CHI,FISNU

CHI(IGM,MXX) - χ for each mixture (single-precision)
FISNU(IGM,MXX) - $\bar{\nu}$ times the fission cross section for each mixture (single-precision)

RECORD 8: EIGEN

EIGEN k_{eff} (single-precision)

NEXT IGM RECORDS: XNDC

XNDC(IM,MM) - Mesh cell centered angular flux for one group (double-precision)

LAST RECORD: TLEAKAGE

TLEAKAGE(IGM) - Total leakage from the system (single-precision)

6.1.6.3 XSDRNPM adjoint output file

For an adjoint case, XSDRNPM writes the following unformatted records on unit NTD, containing the following information:

RECORD 1: EIGEN

EIGEN - k_{eff} value (single-precision)

NEXT IGM RECORDS: XNDC

XNDC(IM,MM) - Mesh cell centered angular flux for one group (double-precision)

The adjoint angular fluxes are reversed in direction such that each angular flux is the importance for that direction in the forward case. This reversal is done by using the reflected angle. Also, the records are written in forward order such that the first record corresponds to the highest-energy group.

6.2 TSUNAMI-3D: CONTROL MODULE FOR THREE-DIMENSIONAL CROSS SECTION SENSITIVITY AND UNCERTAINTY ANALYSIS FOR CRITICALITY

K. B. Bekar, T. M. Greene, S. R. Johnson, B. Langley, J. D. McDonnell, W. J. Marshall, C. M. Perfetti, B. T. Rearden, W. A. Wieselquist

ABSTRACT

TSUNAMI-3D (**T**ools for **S**ensitivity and **Un**certainty **A**nalysis **M**ethodology **I**mplementation in **Three** **D**imensions) is a SCALE control module that facilitates the application of sensitivity and uncertainty analysis theory to criticality safety analysis. In multigroup (MG) mode, TSUNAMI-3D provides for resonance self-shielding of cross section data, calculation of the implicit effects of resonance self-shielding calculations, calculation of forward and adjoint Monte Carlo neutron transport solutions, and calculation of eigenvalue sensitivity coefficients. In continuous-energy (CE) mode, sensitivity coefficients are computed in a single forward Monte Carlo neutron transport calculation for either eigenvalue or generalized reaction rate ratio responses. In both MG and CE modes, the KENO V.a or KENO-VI module is used for transport solvers, and the SAMS module is used to compute and/or create edits of the sensitivity of the calculated responses to the nuclear data used in the calculation as a function of nuclide, reaction type, and energy. SAMS also computes the uncertainty in the calculated responses resulting from uncertainties in the basic nuclear data used in the calculation through energy-dependent cross section covariance matrices. In both MG and CE calculations, a sensitivity data file is produced for use in subsequent analysis.

Version Information

Version 6.2 (2016)

Code Responsible(s): B. T. Rearden, C. M. Perfetti, and L. M. Petrie

The following section acknowledgements appeared in the SCALE 6.2 manual.

The author wishes to acknowledge M. L. Williams, B. L. Broadhead, and K. B. Bekar, of Oak Ridge National Laboratory (ORNL), and R. L. Childs, formerly of ORNL, for their assistance with this work. The support and encouragement of C. M. Hopper, formerly of ORNL, and C. V. Parks and D. E Mueller of ORNL is also appreciated. The continued support of the U.S. Department of Energy Nuclear Criticality Safety Program and the ORNL Laboratory-Directed Research and Development Program is gratefully acknowledged.

6.2.1 INTRODUCTION

TSUNAMI-3D (**T**ools for **S**ensitivity and **Un**certainty **A**nalysis **M**ethodology **I**mplementation in **Three** **D**imensions) is a SCALE control module that facilitates the application of sensitivity and uncertainty theory to criticality safety analysis. The data computed with TSUNAMI-3D are the sensitivity of computed responses (e.g. k_{eff} and ratios of reaction rates) to each constituent nuclear data component used in the calculation. The sensitivity data are coupled with cross section uncertainty data, in the form of multigroup (MG) covariance matrices, to produce an uncertainty in the computed responses due to uncertainties in the underlying nuclear data. The group-wise sensitivity data computed with TSUNAMI-3D are stored in a sensitivity data file format (.sdf file) that is suitable for use in data visualization with Fulcrum, similarity assessment with TSUNAMI-IP, and bias assessment with TSURFER.

This manual is intended to provide the user with a detailed reference on code input options and some examples of the application of TSUNAMI-3D to generate sensitivity and uncertainty data. The techniques used in the MG and continuous-energy (CE) versions of TSUNAMI-3D are described in Sect. 6.2.2 and Sect. 6.2.3, respectively. The input for TSUNAMI-3D is presented in Sect. 6.2.4, and the use of TSUNAMI-3D to obtain accurate sensitivity coefficients for several sample problems is given in Sect. 6.2.5. Additional information is provided in the appendix. A new user may wish to review the sample problems and then refer to the input description in Sect. 6.2.4 to obtain specific guidance on preparing input for specific models.

TSUNAMI-3D provides automated, problem-dependent cross sections using the same methods and input as the **C**riticality **S**afety **A**nalysis **S**equences (CSASs). Although CE calculations with TSUNAMI do not require resonance self-shielding calculations because of its use of CE cross sections, MG TSUNAMI-3D uses self-shielding codes that are similar to those used in MG CSAS calculations. The BONAMIST code computes the sensitivity of resonance self-shielded cross to the input data, the so-called "implicit sensitivities" for MG calculations.

After the cross sections are processed, the MG TSUNAMI-3D-K5 sequence performs two KENO V.a criticality calculations, one forward and one adjoint; the MG TSUNAMI-3D-K6 sequence performs two KENO-VI calculations. Finally, the sequence calls the SAMS module to calculate the sensitivity coefficients that indicate the sensitivity of the computed responses to changes in the cross sections and the uncertainty in the computed responses that is due to uncertainties in the basic nuclear data. SAMS prints energy-integrated sensitivity coefficients and their statistical uncertainties to the SCALE output file and generates a separate data file containing the energy-dependent sensitivity coefficients. CE calculations in TSUNAMI do not require a separate adjoint criticality calculation; instead, they calculate sensitivity coefficients and produce a sensitivity data file during a single forward simulation. The SAMS module is used to print user output for CE calculations.

Choosing poor values for any of several adjustable parameters in TSUNAMI inputs may result in inaccurate sensitivity coefficient estimates; thus, users are advised to always compare their calculated sensitivity coefficients to reference values to verify the suitability of their TSUNAMI input parameters. Sect. 6.2.5 describes the direct perturbation approach for generating reference sensitivity coefficients for the total cross

section of nuclides in a system. It is difficult to calculate high fidelity, energy-dependent, reference sensitivity coefficients using direct perturbation (although possible using stochastic sampling of cross section data), but at the minimum users are advised to verify the accuracy of TSUNAMI-produced total nuclide sensitivity coefficients for all key nuclides in their application.

6.2.2 MULTIGROUP TSUNAMI-3D TECHNIQUES

TSUNAMI-3D is a SCALE control module. As such, its primary function is to control a sequence of calculations that are performed by other codes. A thorough theoretical development of MG eigenvalue sensitivity theory is described in the SAMS section of the SCALE documentation. Currently, two computational sequences are available with TSUNAMI-3D: TSUNAMI-3D-K5 and TSUNAMI-3D-K6. The input for TSUNAMI-3D-K5 is very similar to that used for CSAS5 and the input for TSUNAMI-3D-K6 is very similar to that of CSAS6. TSUNAMI-3D uses the same material and cell data input as all other SCALE sequences. TSUNAMI-3D can calculate eigenvalue sensitivity coefficients using either MG or CE Monte Carlo simulations, but the theoretical approaches for each calculation mode differ greatly. MG TSUNAMI-3D techniques will be discussed in this section, and CE TSUNAMI-3D calculations will be discussed in Sect. 6.2.3. The control sequences available in MG TSUNAMI-3D are summarized in Table 6.2.1, where the functional modules that are executed are also shown. A general flow diagram of MG TSUNAMI-3D is shown in Fig. 6.2.1.

Table 6.2.1: Multigroup TSUNAMI-3D control sequences.

Control sequence	**Functional modules executed by the control module**				
TSUNAMI-3 D-K5	XSProc	KENO V.a forward	KENO V.a adjoint	BONAMIST	SAMS5
TSUNAMI-3 D-K6	XSProc	KENO-VI forward	KENO-VI adjoint	BONAMIST	SAMS6

TSUNAMI-3D and many other SCALE sequences apply a standardized procedure to provide appropriate cross sections for the calculation. This procedure is carried out by routines of the XSProc module, which generate number densities and related information, prepare geometry data for resonance self-shielding and flux-weighting cell calculations, and create data input files for the cross section processing codes.

By default, the MG TSUNAMI-3D sequence performs cross-section processing with XSProc, exercising all available options there, performs the forward and adjoint KENO calculations, calls BONAMIST to produce implicit sensitivity coefficients, then calls SAMS to produce sensitivity and uncertainty output and *sdf* files. Optional sequence level parameters can be used to change methods applied in resonance self-shielding and exclude the implicit sensitivity calculation, which are detailed later in this document.

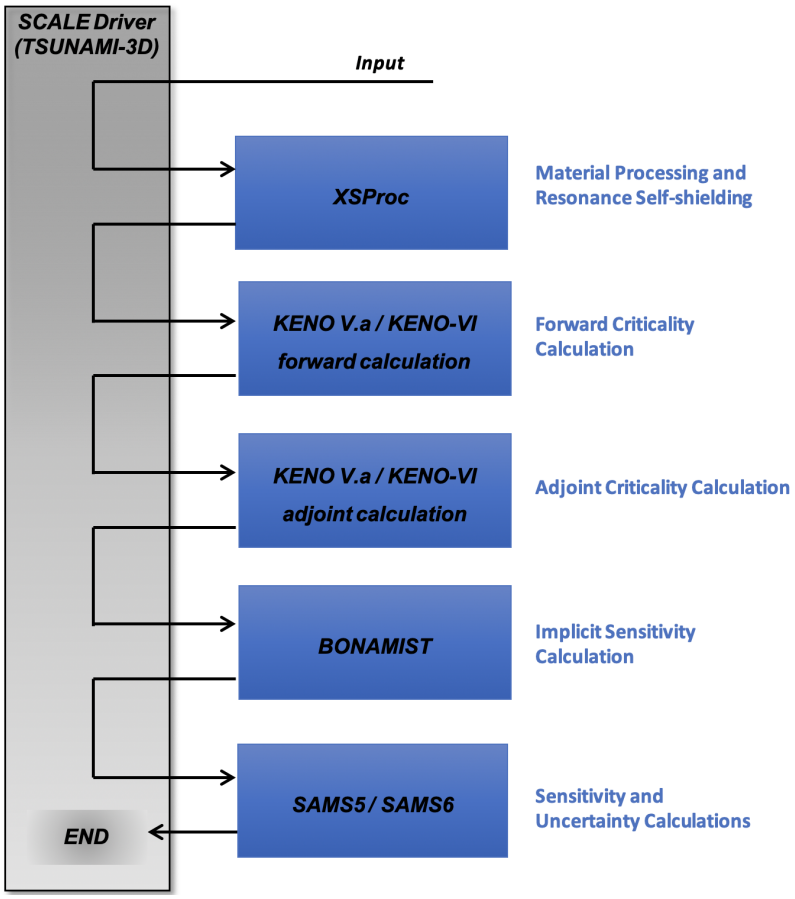


Fig. 6.2.1: General flow diagram of MG TSUNAMI-3D.

Once the appropriate AMPX libraries are prepared, TSUNAMI-3D prepares KENO V.a or KENO-VI inputs for forward and adjoint calculations from the criticality model provided by the user. The input requirements for the KENO V.a input sections are identical to those for the CSAS5 sequence, with some optional additional data. Also, the input requirements for the KENO-VI are identical to those for CSAS6, with some optional additional data. Additional input is prepared for the SAMS module using an optional user-defined input block for SAMS. TSUNAMI-3D executes forward and adjoint KENO calculations, generates implicit sensitivity data with BONAMIST, and then executes the SAMS module to compute the sensitivity and uncertainty data using the data accumulated from the codes previously executed in the sequence. Details concerning calculation of sensitivity and uncertainty data using MG forward and adjoint calculations are provided in the SAMS chapter of the SCALE manual. Of particular interest, the *filename.sdf* file, where *filename* is the name of the input file less any extensions, contains energy-dependent sensitivity coefficients. SCALE returns this file to the same directory as the input file.

The XSProc module is responsible for reading the standard composition data and other engineering-type specifications, including volume fraction or percent theoretical density, temperature, and isotopic distribution as well as the unit cell data. The techniques used in the XSProc module and their applications and limitations are discussed in the XSProc chapter. The input data for XSProc is the same for all analytical sequences available through TSUNAMI-3D, TSUNAMI-1D, CSAS, and many other SCALE sequences.

6.2.3 CONTINUOUS-ENERGY TSUNAMI-3D TECHNIQUES

Like MG TSUNAMI-3D, the CE TSUNAMI-3D capability is a control module that uses codes within the SCALE code package to calculate eigenvalue sensitivity coefficients and other information for models of eigenvalue problems. The CE and MG TSUNAMI-3D calculations differ dramatically in their approach for calculating sensitivity coefficients and as a result have different user interfaces. CE TSUNAMI calculations are automatically enabled when the user selects a CE cross-section library.

6.2.3.1 CE TSUNAMI methodology

CE TSUNAMI currently contains two separate approaches for performing eigenvalue sensitivity coefficient calculations: Iterated Fission Probability (IFP) approach and Contributon-Linked eigenvalue sensitivity/Uncertainty estimation via Tracklength importance Characterization (CLUTCH) approach. Both IFP and CLUTCH calculate sensitivity coefficients during a single forward Monte Carlo (KENO) simulation. Unlike MG TSUNAMI-3D, IFP and CLUTCH do not require the simulation of adjoint histories, calculation of angular flux moments using a flux mesh, volume calculations, or treatment of implicit sensitivity effects. The theoretical background of each method is discussed in detail in the following sections. IFP is often easier to use than CLUTCH, but CLUTCH has greater computational efficiency and a lower memory footprint.

IFP methodology

The IFP methodology, developed by Hurwitz in the 1940s and 1950s, determines the importance of events during a particle history using the notion that an event's importance is proportional to the number of neutrons descended from the original event that are present in the system during some future generation. [3d1], [3d2] In practice, the IFP method requires storing reaction rate tallies for particles for some number of generations until the average population of their descendants in the system, the "asymptotic population," is obtained. This process is illustrated in Fig. 6.2.2. Once obtained, the asymptotic population of the original neutron is used to weight reaction rate tallies for that neutron and to produce sensitivity coefficient estimates via the first-order perturbation equation. The number of fission neutrons created by progenitor p's asymptotic population,

$$\pi_p = \sum_{\tau \in p} \sum_{s \in \text{paths}} w_s \bar{\nu} \Sigma_f l_s, \tag{6.2.1}$$

takes the place of the adjoint flux ϕ^* in the first-order perturbation equations.

A number of generations, referred to as the "latent generations," must be skipped before calculating the asymptotic population for an event to guarantee that the progeny of the event have had sufficient time to impact all regions in the system and to converge to a true estimate of the asymptotic population. The number of latent generations required to calculate accurate sensitivity coefficients varies based on the complexity of the system and the desired sensitivity coefficient fidelity, but in general 20 generations is a conservative number of latent generations to ensure convergence to the asymptotic population. (Note that the current default in TSUNAMI-3D is 5 latent generations.) The IFP method is useful for benchmarking the accuracy of other sensitivity coefficient methods and is very easy to use because the only assumption of the IFP method (besides the standard CE Monte Carlo and first order perturbation theory assumptions) is that the asymptotic population that is reached after the chosen number of latent generations is representative of the importance of the original event. Thus a user who is new to sensitivity methods can assume a conservative number of latent generations and can use the IFP method to accurately calculate sensitivity coefficients for a system so long as the user's computer has sufficient computational memory for the simulation.

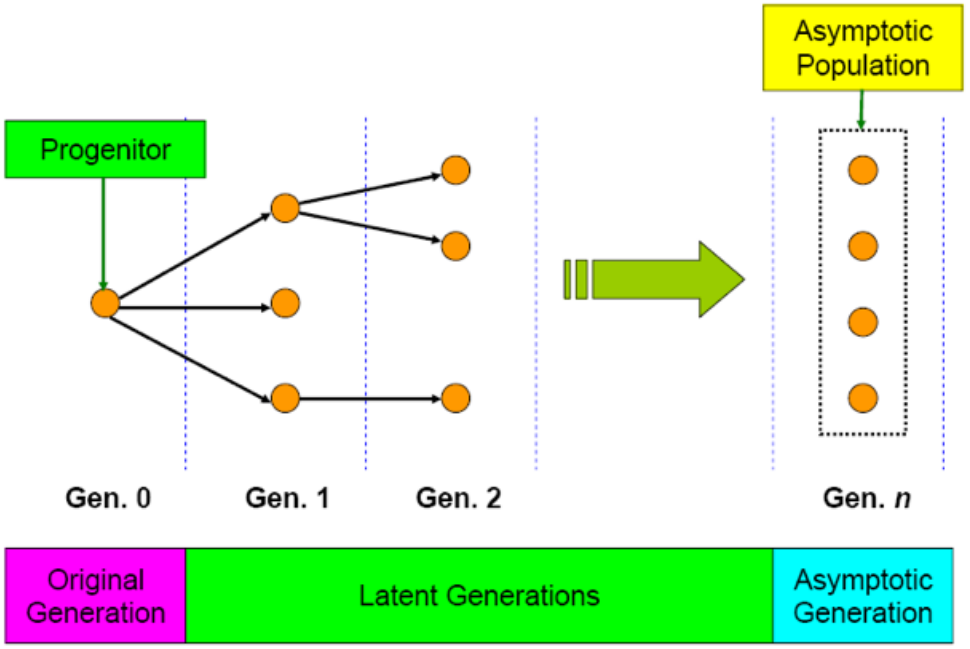


Fig. 6.2.2: Illustration of the Iterated Fission Probability process.

The IFP method requires storage of region-, isotope-, reaction-, and energy-dependent reaction rates for every particle for the specified number of latent generations. Complex problems can require simulating tens of thousands or hundreds of thousands of particles during each generation, and IFP simulations for these systems can easily require many gigabytes of computational memory storage. The IFP implementation in TSUNAMI-3D makes use of dynamic memory allocation to minimize memory requirements, but the method frequently produces large memory footprints despite these optimizations. The IFP memory requirements are proportional to the number of latent generations used in the calculations, and perhaps the best approach for minimizing the memory requirements of the IFP method is to minimize the number of latent generations used in a calculation. Twenty is typically a conservative number of latent generations, and it is recommended that the user always use as few latent generations as possible for IFP simulations. The five latent generations that IFP assumes by default is a reasonable starting guess, but users should compare IFP nuclide sensitivity coefficients with direct perturbation sensitivities to determine if they need to use more or fewer latent generations. Determining the adequate number of latent generations can be done by starting with a small number of latent generations and slowly increasing the number until the IFP-calculated sensitivity coefficients agree well with reference sensitivities. The memory requirements of the IFP method are also proportional to the number of particles used in each generation (NPG), and high-memory simulations can decrease NPG to decrease the memory requirements of the IFP method. This should be done with caution because making NPG too small can affect the fission source convergence and thus can produce poor tally Monte Carlo variance estimates.

The integration of the Shift Monte Carlo transport code has made possible the parallelization of the IFP method, so that the large amount of memory required can be distributed among several available processors. Note that this parallelism for IFP is only available when using the Shift transport code, and it is not available through KENO.

CLUTCH methodology

The CLUTCH method was developed by Perfetti in 2012, based in part on the Contributon theory explored by M. L. Williams of ORNL, to produce an accurate and efficient means for calculating eigenvalue sensitivity coefficients with a small computational memory footprint. [3d3] Like the IFP method, CLUTCH calculates sensitivities during a single forward Monte Carlo calculation and does not require tallying angular flux moments. Instead, the CLUTCH method calculates the importance of events during a particle's lifetime by examining how many fission neutrons are created by that particle after those events occur. Consider a neutron source Q that is equal to the fission source of a system:

$$Q = \lambda F \phi. \tag{6.2.2}$$

Multiplying this source by the adjoint flux and integrating over phase space gives

$$\langle \phi^* Q \rangle = \lambda \langle \phi^* F \phi \rangle \tag{6.2.3}$$

Consider now a neutron emitted in phase space τ_s such that $Q(\tau_s) = Q_s\, \delta(\tau - \tau_s)$. This source definition reduces Eq. (6.2.3) and allows the importance of the neutron in phase space τ_s to be calculated by

$$\phi^*(\tau_s) = \frac{\lambda}{Q_s} \int_V G(\tau_s \rightarrow r)\, F^*(r) dr \tag{6.2.4}$$

where the transfer function $G(\tau_s \rightarrow r)$ is equal to the expected number of fission neutrons generated in all energies and directions at r due to a neutron emitted at phase space τ_s and is given by

$$G(\tau_s \rightarrow r) = \frac{1}{Q_s} \int_E \int_\Omega \nu \Sigma_f(r, E) \phi(r, E, \Omega \mid Q(\tau_s))\, d\Omega dE \tag{6.2.5}$$

where $\phi(r, E, \Omega \mid Q(\tau_s))$ is the flux created in phase space (r, E, Ω) given the source $Q(\tau_s)$. The weighting function $F^*(r)$ is defined to be equal to the expected importance generated by a fission neutron emitted at r and is given by

$$F^*(r) = \int_E \int_\Omega \frac{\chi(r, E)}{4\pi} \phi^*(r, E, \Omega) d\Omega dE. \tag{6.2.6}$$

In practice, the CLUTCH method calculates the integral of $G(\tau_s \rightarrow r)$, weighted by $F^*(r)$, to calculate the importance of every event in a particle's lifetime. For example, the importance of a scattering event would be determined by tallying how many fission neutrons, weighted by the value of $F^*(r)$ at the sites where they are born, are created by the neutron that emerges from the scattering collision. The CLUTCH method cannot calculate the importance of events until after the particle that caused these events dies, which requires that CLUTCH store reaction rate information for every collision that a particle undergoes during its lifetime. This is a manageable amount of information because these tallies are not energy dependent (a particle's energy is constant between any two collisions) and because this information can be deleted once the particle dies. In contrast, the IFP method requires much more memory because it stores energy-dependent reaction rate tallies for every particle for the number of latent generations.

The only assumption made by the CLUTCH method (besides the standard CE Monte Carlo assumptions) is that $F^*(r)$ provides an accurate estimate of the average importance of fission neutrons at r. The current approach for calculating $F^*(r)$ takes advantage of the definition of the unconstrained fission spectrum sensitivity coefficient, which is given by

$$S_{k,\chi}(r) = \frac{1}{D} \frac{1}{k} \int_0^\infty \int_{4\pi} \bar{\nu} \Sigma_f(r, E) \phi(r, E, \Omega) d\Omega dE \int_0^\infty \int_{4\pi} \frac{\chi(r, E')}{4\pi} \phi^*(r, E', \Omega')\, d\Omega' dE' \tag{6.2.7}$$

where D is the adjoint-weighted fission source for the system. The right-most integral of Eq. (6.2.7) is recognized as the definition of $F^*(r)$ in Eq. (6.2.6) and the terms in Eq. (6.2.7) can be rearranged to define $F^*(r)$ as

$$F^*(r) = \frac{D \times S_{k,\chi}(r)}{\int_0^\infty \int_{4\pi} \frac{\overline{\upsilon}\Sigma_f(r,E)\phi(r,E,\Omega)}{k} d\Omega dE} \tag{6.2.8}$$

This approach assumes that the energy spectrum of neutrons emitted from a fission event is not strongly dependent on the parent isotope or the energy of the neutron causing the fission. The current CLUTCH implementation tallies the unconstrained chi sensitivity in the numerator of Eq. (6.2.8) during the inactive generations to estimate $F^*(r)$ for the active generation sensitivity coefficient calculations. The spatial dependence of $F^*(r)$ is currently accounted for by calculating and storing $F^*(r)$ on a spatial mesh, although kernel density estimators might be used in the future to store $F^*(r)$ using spatially continuous functional representations. [3d4] Because $F^*(r)$ is only nonzero for regions containing fissionable material, the $F^*(r)$ mesh used in a CE TSUNAMI calculation must only encompass all fissionable material in the system, rather than the entire system as required by MG TSUNAMI. The D term in Eq. (6.2.8) can be ignored because it is constant for all regions in a problem and is cancelled out by the presence of $F^*(r)$ terms in both the numerator and denominator of the first-order perturbation equation. The denominator in Eq. (6.2.8) is simply the total weight of fission neutrons born in each $F^*(r)$ mesh region, which must also be tallied during the inactive generations.

Because $F^*(r)$ describes the contribution to the chi sensitivity that is created per fission neutron born at a point, a converged fission source is not required to begin $F^*(r)$ calculations; thus, $F^*(r)$ tallies begin during the inactive generations of Monte Carlo simulations to obtain useful information from the typically discarded inactive particle histories. The fission source must converge well enough during the inactive generations so that $F^*(r)$ is tallied in all fissile regions to a desired statistical uncertainty, and it is sometimes necessary to simulate extra inactive generations to allow $F^*(r)$ tallies to fully converge; however, the ability to begin $F^*(r)$ tallies before the fission source is converged essentially provides "free" tallies while the fission source is converging.

Although several approaches exist for calculating the unconstrained chi sensitivity coefficients needed to calculate $F^*(r)$, an IFP-based approach has been determined to be the best approach. [3d3] Although the IFP method can produce large memory footprints for full sensitivity coefficient calculations, the amount of memory required by the method to calculate unconstrained chi sensitivity coefficients and $F^*(r)$ is generally negligible. The spatial dependence of $F^*(r)$ is currently described using a spatial mesh, and an interval of 1 to 2 cm mesh is typically sufficiently refined to obtain accurate $F^*(r)$ estimates. Users should simulate at least on the order of 50 to 100 inactive histories per mesh interval to allow for sufficient $F^*(r)$ convergence; sometimes this necessitates simulating a large number of additional inactive histories/generations.

6.2.3.2 CE TSUNAMI Generalized Perturbation Theory capability

The CLUTCH and IFP methods were combined in 2013 by Perfetti to enable sensitivity calculations for generalized neutronic response ratios. [3d5] This Generalized Perturbation Theory (GPT) capability is known as the GEneralized Adjoint Responses in Monte Carlo (GEAR-MC) method. Applications for GPT sensitivity coefficients differ from the traditional criticality safety applications of TSUNAMI, and may include S/U analyses for multigroup cross sections that are produced by a continuous-energy Monte Carlo code, the relative power of pins in a LWR, or ratios of foil activities in an irradiation experiment.

Two approaches for performing the GEAR-MC method have been included in this release as demonstration capabilities. One implementation relies heavily on the IFP method for sensitivity calculations and is therefore subject to the long runtimes and large memory footprint that may be encountered when using the IFP method.

The other implementation does not use the IFP method when calculating sensitivities. Instead, it uses only the CLUTCH method with an $F^*(r)$ mesh that has been modified for performing generalized response sensitivity calculations. Because it does not use the IFP method except for calculating the $F^*(r)$ function, this approach typically produces a significantly lower memory footprint than the first GEAR-MC implementation and can be performed in a parallel environment. [3d6] Both approaches are experimental capabilities and have been included in this release to demonstrate the expanding SCALE S/U capabilities. For the CLUTCH-only GPT, the current format for inputting response information allows for the sensitivity calculations for a single response in each sensitivity calculation. For GPT via CLUTCH and IFP, multiple responses may be defined.

Rather than calculating eigenvalue sensitivities, the GEAR-MC method calculates the sensitivity of a response, R, to perturbations or uncertainties in the system parameters. The generalized response sensitivity coefficient for the system parameter Σ_x is defined as

$$S_{R,\Sigma_x} = \frac{\delta R/R}{\delta\Sigma_x/\Sigma_x} \tag{6.2.9}$$

The GEAR-MC method calculates sensitivities of responses that are defined as ratios of neutron reaction rates integrated over some region of phase space such that

$$R = \frac{\langle\Sigma_1\phi\rangle}{\langle\Sigma_2\phi\rangle} \tag{6.2.10}$$

where Σ_1 and Σ_2 are nuclear cross sections. The reaction rates in Eq. (6.2.10) can be isotope- or material-dependent reaction rates and can also represent neutron flux responses if $\Sigma = 1$. The fractional change in R due to a perturbation $\delta\Sigma_x$ to the system parameter Σ_x is given by

$$\frac{\delta R}{R} = \left\langle \frac{1}{R}\frac{\partial R}{\partial\Sigma_x}\delta\Sigma_x + \frac{1}{R}\frac{\partial R}{\partial\phi}\frac{\partial\phi}{\partial\Sigma_x}\delta\Sigma_x \right\rangle \tag{6.2.11}$$

The first term in Eq. (6.2.11) is known as the "direct effect term" and describes how perturbations in Σ_x affect the response function of the response reaction rates. The second term, known as the "indirect effect term," describes how perturbations affect the neutron flux spectrum in the response region. [3d7] Calculating the sensitivity of the response to the direct effect term is relatively simple and involves tallying the fraction of the total numerator and denominator responses that is generated for each energy, region, isotope, and material in the response region(s). For example, consider a response that is defined as the ratio of the energy-integrated fission rate to the energy-integrated capture rate in a uranium fuel pin. The direct effect sensitivity of this response to the thermal fission cross section is simply the fraction of the fission reaction rate in the pin that is caused by neutrons with thermal energies.

The indirect effect term in Eq. (6.2.11) cannot be calculated as simply as the direct effect term and has historically posed a greater challenge. The GEAR-MC method offers an approach for calculating the indirect effect term during a single, unperturbed Monte Carlo transport calculation. The neutron balance equation for an eigenvalue problem is given by

$$L\phi \; - \; \lambda P\phi = 0 \tag{6.2.12}$$

where L is the neutron loss operator and P is the fission neutron production operator. The change induced in the neutron balance equation in response to a first-order perturbation is given by

$$(L - \lambda P)\,\delta\phi = \; \delta\lambda P\phi + (\lambda\delta P - \delta L)\phi \tag{6.2.13}$$

Consider now the generalized adjoint balance equation

$$(L^* - \lambda P^*)\Gamma^* = S^* \tag{6.2.14}$$

where L^* is the adjoint loss operator, P^* is the adjoint fission neutron production operator, S^* is a source of importance for the response that is defined such that $\langle \phi S^* \rangle = 0$, and Γ^* is the generalized importance function that provides the solution to this equation.[7] Multiplying Eq. (6.2.13) and Eq. (6.2.14) by Γ^* and $\delta\phi$, respectively, and taking the inner product gives, respectively,

$$\langle \Gamma^*(L - \lambda P)\delta\phi \rangle = \delta\lambda \langle \Gamma^* P\phi \rangle + \langle \Gamma^*(\lambda\delta P - \delta L)\phi \rangle \tag{6.2.15}$$

and

$$\langle \delta\phi \, (L^* - \lambda P^*) \, \Gamma^* \rangle = \langle \delta\phi S^* \rangle \tag{6.2.16}$$

The source of adjoint importance in Eq. (6.2.16) is defined to conveniently provide an expression for the indirect effect term. Defining S^* as

$$S^* \equiv \frac{1}{R}\frac{\delta R}{\delta \phi} = \frac{\Sigma_1}{\langle \Sigma_1 \phi \rangle} - \frac{\Sigma_2}{\langle \Sigma_2 \phi \rangle} \tag{6.2.17}$$

and applying the adjoint property allows Eq. (6.2.15) and Eq. (6.2.17) to be combined to express the indirect effect term as

$$\left\langle \frac{1}{R}\frac{\delta R}{\delta \phi}\delta\phi \right\rangle = \langle \delta\lambda \Gamma^* P\phi \rangle + \langle \Gamma^*(\lambda\delta P - \delta L)\phi \rangle \tag{6.2.18}$$

Eq. (6.2.18) is usually equal to zero because Γ^* is typically orthogonal to $P\phi$. The effect of this orthogonality can be interpreted in a more physical manner by realizing that perturbations to the eigenvalue of a system do not alter the steady-state neutron flux shape or spectrum of the system. As a result, perturbations affect the response numerator and denominator terms equally.

The GEAR-MC methodology uses Eq. (6.2.14) and Eq. (6.2.18) to calculate the generalized importance function Γ^* for neutrons during a single forward Monte Carlo simulation, thus enabling the calculation of the indirect effect term in Eq. (6.2.11). and thus sensitivity coefficients for generalized responses using GPT. The approach developed for calculating Γ^* is similar to the approach used by the CLUTCH method for calculating eigenvalue sensitivity coefficients.

Assuming that the fission production term, $\lambda P\phi$, in Eq. (6.2.12) is the sole source of neutron production in a system, Q, multiplying Eq. (6.2.12) and Eq. (6.2.14) by Γ^* and ϕ, respectively, and integrating over all phase space gives

$$\langle \Gamma^* L\phi \rangle = \langle \Gamma^* Q \rangle \tag{6.2.19}$$

and

$$\langle \phi L^* \Gamma^* \rangle = \lambda \langle \phi P^* \Gamma^* \rangle + \langle \phi S^* \rangle \tag{6.2.20}$$

Nonfission neutron production reactions, such as (*n,Xn*) reactions, are included in the L^* adjoint loss term in Eq. (6.2.20). Combining Eq. (6.2.19) and Eq. (6.2.20). and using the adjoint property gives

$$\langle \Gamma^* Q \rangle = \lambda \langle \Gamma^* P\phi \rangle + \langle \phi S^* \rangle \tag{6.2.21}$$

The terms in Eq. (6.2.21) are all equal to zero in inner product space, but it can be used to extract information about the importance of events by considering the neutron source to be a single neutron traveling through the phase space τ_s, such as a neutron entering or leaving a collision at some point. This concept is used similarly in Williams' Contributon theory for calculating eigenvalue sensitivity coefficients and assumes that

$$Q = Q_s \delta(\tau - \tau_s) \tag{6.2.22}$$

where Q_s is the source strength for this neutron. [3d8], [3d9] Substituting Eq. (6.2.22) into Eq. (6.2.21) produces an expression for the generalized importance function at τ_s:

$$\begin{aligned} \Gamma^*(\tau_s) &= \frac{1}{Q_S} \langle S^*(r)\phi(\tau_s \to r)\rangle + \frac{\lambda}{Q_s} \langle \Gamma^*(r) P\phi(\tau_s \to r)\rangle \\ &= \frac{1}{Q_s} \left\langle \frac{1}{R}\frac{\delta R}{\delta \phi}(r)\phi(\tau_s \to r)\right\rangle + \frac{\lambda}{Q_s} \langle \Gamma^*(r) P\phi(\tau_s \to r)\rangle \end{aligned} \tag{6.2.23}$$

where $\phi(\tau_s \to r)$ is the neutron flux created at r by the neutron originating at τ_s. The two terms on the right-hand side of Eq. (6.2.21) and Eq. (6.2.23) represent the intragenerational and intergenerational effects of an event on the importance of a particle, respectively. The intragenerational effect term describes how much importance the neutron in phase space τ_s generates in the response region(s) during its lifetime; the intergenerational effect term describes how many fission neutrons this neutron creates and how much importance these fission neutrons will generate in future generations. The intragenerational term can be determined by tallying the amount of flux generated in the response region(s) and weighted by $S^*(r)$ from Eq. (6.2.17) from the time the particle enters phase space τ_s until its death. Thus the intragenerational term is given by

$$\langle S^*(r)\phi(\tau_s \to r)\rangle = \frac{\Sigma_1 \phi(\tau_s \to r)}{\langle \Sigma_1 \phi\rangle} - \frac{\Sigma_2 \phi(\tau_s \to r)}{\langle \Sigma_2 \phi\rangle} \tag{6.2.24}$$

The approach for calculating the intragenerational importance term in Eq. (6.2.24) is similar to the approach used by the CLUTCH method during eigenvalue sensitivity coefficient calculations and requires storing tracklength information for each collision that a particle enters and determining the importance of that collision after the particle dies. The presence of both positive and negative terms in Eq. (6.2.24) allows a single event to generate either a positive or negative importance. The intergenerational contribution to the importance function can be calculated by tallying the cumulative score of $S^*(r)\,\phi(\tau_s \to r)$ that is generated by the particle's daughter fission neutrons, or "progeny," over some number of generations. The GEAR-MC method estimates the intergenerational importance by summing the intragenerational importance, Γ_i^*, generated by the fission production, F_i, of neutrons in the ith generation of a fission chain over some number of generations:

$$\lambda \langle \Gamma^*(r) P\phi(\tau_s \to r)\rangle = \Gamma_1^* F_1 + \Gamma_2^* F_2 + \Gamma_3^* F_3 + \ldots + 0 \tag{6.2.25}$$

This approach is used similarly by the IFP approach for calculating the importance of events during eigenvalue sensitivity calculations, except that the IFP method tallies the importance only one time after the daughter neutrons have established an asymptotic population in the system. The $\delta\lambda$ term in Eq. (6.2.18). demands that the $\langle \Gamma^* P\phi\rangle$ term be equal to zero, which causes the $\Gamma_i^* F_i$ terms in Eq. (6.2.25) to approach zero as i approaches infinity; therefore, the intergenerational importance term is obtained by taking the sum of the $\Gamma_i^* F_i$ terms as they asymptotically approach zero.

6.2.3.3 CE TSUNAMI sequence description

The code flow of CE calculations with TSUNAMI is significantly simpler than the flow of MG TSUNAMI because CE Monte Carlo does not require resonance self-shielding of MG cross sections or the calculation of implicit sensitivity coefficients and calculates sensitivity coefficients during a single forward calculation. The CE control sequences available in TSUNAMI are summarized in Table 6.2.2, where the functional modules executed are also shown. A general flow diagram of a CE calculation with TSUNAMI is shown in Fig. 6.2.3.

Table 6.2.2: CE TSUNAMI-3D control sequences.

Control sequence	Functional modules executed by the control module	
TSUNAMI-3D-K5	KENO V.a	SAMS5
TSUNAMI-3D-K5-SHIFT	Shift	SAMS5
TSUNAMI-3D-K6	KENO-VI	SAMS6
TSUNAMI-3D-K6-SHIFT	Shift	SAMS6

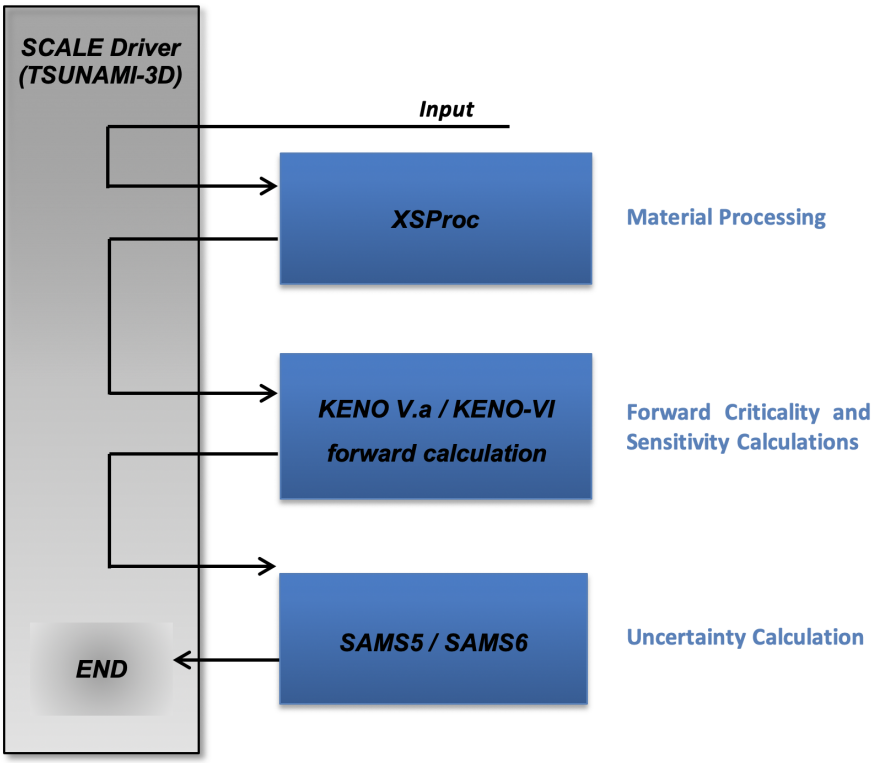


Fig. 6.2.3: General flow diagram of CE TSUNAMI-3D.

Eigenvalue sensitivity coefficients are calculated during the KENO or Shift Monte Carlo transport calculation, and the energy-dependent sensitivity coefficients are summarized in a sensitivity data file (sdf). The SAMS module then uses the *.sdf* file produced and cross section covariance data to complete the eigenvalue uncertainty analysis for the problem and estimate the data-induced eigenvalue uncertainty.

6.2.4 TSUNAMI-3D INPUT DESCRIPTION

The input for TSUNAMI-3D is designed to be very compatible with those used for the CSAS criticality safety analysis sequences. Given a CSAS input, MG TSUNAMI-3D calculations only require several input modifications to obtain adequate flux solutions, and CE TSUNAMI calculations require as little as one additional parameter. Additional optional input may be added to control the sensitivity calculations.

The input to TSUNAMI-3D consists of an input title, SCALE analytical sequence specification record, SCALE XSProc data, KENO V.a or KENO-VI input descriptions with some additional optional parameter data, and optional sensitivity and uncertainty data. These data are processed using the SCALE free-form reading routines, which allow alphanumeric data, floating-point data, and integer data to be entered in an unstructured manner. The input is not case sensitive, so either upper- or lowercase letters may be used. A maximum of 252 columns per line may be used for input, although some exceptions for this rule exist, such as the 80 character title data. Data can usually start or end in any column with a few exceptions. As an example, the word END beginning in column 1 and followed by two blank spaces will end the problem, any data following will be ignored. Each data entry must be followed by one or more blanks to terminate the data entry. For numeric data, either a comma or a blank can be used to terminate each data entry. Integers may be entered for floating values. For example, 10 will be interpreted as 10.0. Imbedded blanks are not allowed within a data entry unless an E precedes a single blank as in an unsigned exponent in a floating-point number. For example, 1.0E 4 would be correctly interpreted as 1.0×10^4.

6.2.4.1 Analytical sequence specification record

The analytical sequence specification begins in column 1 of the first line of the input file and must contain the following:

=TSUNAMI-3D-K5

This sequence is used for sensitivity and uncertainty calculations with KENO V.a.

=TSUNAMI-3D-K6

This sequence is used for sensitivity and uncertainty calculations with KENO-VI.

Either analytical sequence specification may have `-shift` appended in order to use the Shift Monte Carlo transport code. (Namely, `tsunami-3d-k5-shift` or `tsunami-3d-k6-shift`.)

Optional keyword input may be entered after the analytical sequence specification record. These keywords are

PARM=CHECK PARM=CHK	This option allows the input data to be read and checked without executing any functional modules.

Note: The following PARM settings only apply to MG calculations and are ignored for CE calculations:

`PARM=SIZE=n`

The amount of memory requested in four-byte words may be set with this entry. The default value for n is 20000000. This value only affects calculations in BONAMIST, where this value of the SIZE parameter is used for allocation of storage for the derivatives. Please see the documentation on BONAMIST in the Sensitivity Utility Modules chapter for more details. All other codes use dynamic memory allocation and this value has no effect.

PARM=BONAMIST

This is the default configuration for MG TSUNAMI-3D calculations. XSProc is used with BONAMI and CENTRM for cross section processing. Implicit sensitivities are produced with BONAMIST.

PARM=CENTRM

XSProc is used with BONAMI and CENTRM for cross section processing, but BONAMIST is not run. **MG TSUNAMI-3D calculations with PARM=CENTRM do not account for contributions by implicit sensitivity effects, and should be used with caution.**

PARM=BONAMI

XSProc is used with BONAMI for cross section processing, but BONAMIST is not run. **MG TSUNAMI-3D calculations with PARM=BONAMI do not account for contributions by implicit sensitivity effects, and should be used with caution.**

PARM=2REGION

XSProc (with BONAMI and CENTRM) use Dancoff factors to compute neutron escape probabilities for an accelerated, yet more approximate, CENTRM calculation. Implicit sensitivities are computed with BONAMIST.

Multiple parameters can be used simultaneously by enclosing them in parentheses and separating them with commas such as PARM=(SIZE=2000000, CHECK).

6.2.4.2 Title data

A *title*, a character string, must be entered as the second line of the input file. The syntax for the title is a string of characters with a length of up to 80 characters, including blanks.

6.2.4.3 XSProc Execution

XSProc reads the standard composition specification data for MG and CE and the unit cell geometry specifications for MG resonance self-shielding calculations. CE TSUNAMI calculations do not require the specification of unit cells. Please see chapters on the Material Information Processor for input specifications. The cross section data library that is to be used by TSUNAMI must also be entered as the third line of the TSUNAMI input; a list of the currently available libraries are listed in the table *Standard SCALE cross section libraries* of the XSLIB chapter.

6.2.4.4 KENO V.A or KENO-VI problem description

The KENO V.a or KENO-VI problem description follows the Material Input Processor section in the TSUNAMI-3D input. The input for KENO V.a and KENO-VI in TSUNAMI-3D is very similar to that described in section *KENO V.a Data Guide* in chapter KENO V.a or section *KENO-VI Data Guide* in the KENO-VI chapter, with only a few differences in the default values for the parameter data and a few additional parameters to control the adjoint criticality calculation. Otherwise, geometry, array, biasing, boundary, start, and plot data are entered exactly as described for KENO. Default parameter values for MG TSUNAMI-3D that are different from those used for other MG KENO calculations are shown in Table 6.2.3. Parameters that are unique to TSUNAMI-3D and are used to control to the MG adjoint calculation are shown in Table 6.2.4. These adjoint parameters are optional and can be entered with other parameters in the standard *READ PARAMETER* input block in KENO problem description. Parameters that are unique to CE TSUNAMI-3D are shown in Table 6.2.5; these parameters can also be entered in the KENO *READ PARAMETER* input block.

Several features were added to KENO V.a and KENO-VI to allow the calculation of sensitivity coefficients from the MG Monte Carlo analysis. One significant addition is the calculation of neutron flux moments and/or angular fluxes, both of which give directional components to the neutron flux solution required to compute

the sensitivity of k_{eff} to scattering cross sections. Another significant addition is the ability to compute fluxes that are subdivided over a cubic mesh or Cartesian spatial grid. These mesh fluxes simplify the accurate computation of the product of the forward and adjoint flux solutions, sometimes called the "inner product." Both of the flux moments and the spatial flux mesh options must be correctly applied to obtain accurate sensitivity coefficients. Generally, the refinement of the flux mesh to sufficiently small intervals to capture relevant spatial effects while maintaining a manageable memory footprint is the most challenging aspect of performing MG calculations with TSUNAMI-3D.

Important: It is important to note that TSUNAMI-3D provides no default mesh for these flux tallies, and the user must input a mesh using either the MSH parameter or a GridGeometry input to access this feature.

New input data that aid in the accurate calculation of sensitivity coefficients are the parameter inputs: NQD, PNM, MFX, MSH, TFM, APG, AGN, ABK, ASG, CET, CFP, CGD, FST, NNC, NMA, NMT, DNC, DMA, DMT, NMX, NMN, DMX, and DMN. Thus, these parameters may require modification when using TSUNAMI-3D. The *READ GRID* block of KENO input is used with MSH to input the planar grid for MG TSUNAMI-3D calculations and also to input the spatial grid for the calculation and storage of $F^*(r)$ for CE TSUNAMI calculations. Also, if using the coordinate transform for angular flux or flux moment calculations, use of the optional *CENTER* modifier in the KENO geometry input may be required. Users should note that when using the *xlinear*, *ylinear*, or *zlinear* options in the *READ GRID* block that KENO checks to determine whether the boundaries of each grid coincide with the global unit boundaries. If any planes in the grid boundaries are identical to the planes used in the global boundary, then KENO will extend the outermost plane in that grid by a distance equal to one-tenth of the grid mesh interval to ensure that the grid mesh covers the entire geometry.

Some KENO parameters (e.g., SCD, MFX, CDS, GFX, and CGD) may be used to assign user provided grid definitions for use with specific tallies. These are described in Sect. 8.1.3.3 of the SCALE KENO chapter. Grid IDs in the READ GRID blocks should be limited to no more than 4 integer characters (i.e., $1 \leq \text{ID} \leq 9999$).

Table 6.2.3: Default values KENO parameters in MG TSUNAMI-3D

Parameter	**Default value for TSUNAMI-3D**	**Default value for KENO in CSAS Sequences or as stand-alone code**	**Description**
CFX	YES	NO	Collect fluxes
GEN	550	203	Number of generations to be run
NSK	50	3	Number of generations to be omitted when collecting results
PNM	3	0	Highest order of flux moments tallies
TFM	YES	NO	Perform coordinate transform for flux moment and angular flux calculations

Table 6.2.4: Default values of TSUN
MG adjoint calculation.

Parameter	Default value for TSUNAMI-3D	Corresponding KENO parameter	Description
ABK	APG × 2	NBK	Number of positions in the neutr
AGN	GEN - NSK + ASK	GEN	Number of generations to be run
APG	NPG × 3	NPG	Number of particles per generati
ASG	SIG (default SIG=0)	SIG	if > 0.0, this is the standard devi
ASK	NSK × 3	NSK	Number of generations to be om

MG TSUNAMI sensitivity coefficient estimates can be very sensitive to the values of a problem's input parameters, and users should always check the accuracy of their sensitivity coefficients by comparing them with reference direct perturbation sensitivities, discussed in Sect. 6.2.5.1. The default values for parameters in Table 6.2.3 and Table 6.2.4 generally serve as good starting values for a MG TSUNAMI calculation, but users may need to use a higher order of flux moments to better capture the angular dependence of the flux (typically PNM=5 is sufficient), and may need to simulate more histories/generations to reduce the uncertainty in sensitivity coefficient estimates. The dimensions of the flux mesh are another important MG TSUNAMI parameter, and a reasonable starting guess for the width of the flux mesh intervals is $1/10^{\text{th}}$ of the diameter of the fuel-containing region. Users should take care when increasing the order of the flux moment tallies and the number of intervals in the spatial flux mesh as the memory footprint of a MG TSUNAMI calculation increases quickly as these parameters increase.

CE TSUNAMI calculations are in many respects simpler than MG TSUNAMI calculations because they use state-of-the-art sensitivity methodologies that typically require less user input to perform sensitivity calculations. CE TSUNAMI calculations do not use any of the input parameters in Table 6.2.3 or Table 6.2.4 and do not require flux moment tallies, flux mesh tallies (except for calculating $F^*(r)$), or the simulation of adjoint histories. CET, CFP, and CDG are the three parameters that control CE TSUNAMI eigenvalue sensitivity calculations. CET specifies which CE sensitivity method (either CLUTCH, IFP, or GEAR-MC) will be used in the CE TSUNAMI calculation. CFP specifies how many latent generations will be used by the IFP method for either calculating sensitivity coefficients (CET=2/5) or for calculating $F^*(r)$ during the inactive generations (CET=1/4); if CET=1 or 4 and CFP= -1, then CE TSUNAMI will perform a CLUTCH calculation assuming that $F^*(r)$ is equal to one everywhere for CET=1 and zero everywhere for CET=4. The number of latent generations (CFP) and the number of generations skipped for fission source convergence (NSK) control very different things.

The typical workflow for generating an IFP-based CE TSUNAMI k_{eff} sensitivity input is given below:

1) Set CET=2.
2) Set your number of latent generations using CFP=# (usually between 5 and 10).

The typical workflow for generating a CLUTCH-based CE TSUNAMI k_{eff} sensitivity input is given below:

1) Set CET=1.
2) Set your number of latent generations using CFP=# (usually between 5 and 10).
3) Specify a mesh size for the $F^*(r)$ mesh with CGD=yes and MSH=<value>; the width of the mesh voxels is usually between 1 and 2 cm in the X-, Y-, and Z-dimensions. Ensure that the mesh covers all fisionable regions of the problem. If the user desires to construct a specific $F^*(r)$ mesh, specify a GridGeometry with CGD=# the ID of the GridGeometry intended.

4) Consider simulating extra inactive generations to allow the $F^*(r)$ mesh to converge — most $F^*(r)$ mesh calculations require between 10 and 100 inactive histories per mesh voxel to sufficiently converge.

   Ex: A problem using a 2×30×40 $F^*(r)$ mesh contains 24,000 voxels. Assuming at least 10 inactive histories per mesh interval means this problem will require 240,000 inactive histories for the $F^*(r)$ mesh to converge. If the problem uses 1,000 particles per generation (NPG=1000), then the user should use at least 240 skipped generations (NSK=240) to allow $F^*(r)$ mesh tallies to converge.

When performing CLUTCH calculations using $F^*(r)$ (i.e., when CET=1 or 4 and CFP is not -1) a spatial grid for $F^*(r)$ is needed. CGD=yes may be used with a specification of uniform mesh size with MSH=<value>, or CGD=# to specify the ID of this a GridGeometry $F^*(r)$ grid. Failure to specify a grid results in an error message. The entries in the $F^*(r)$ grid can be printed to a 3dmap file by setting the parameter FST=YES, which is the default. This information is printed to a file with the same name as the input but with a _FStar_map.3dmap extension (i.e., *problem.inp* prints information to *problem_FStar_map.3dmap*). This .3dmap file can be viewed using the Fulcrum interface, as shown for a CLUTCH k_{eff} sensitivity test problem in Fig. 6.2.4. Values for $F^*(r)$ are set by default to one/zero in CLUTCH k_{eff}/GPT sensitivity calculations, respectively, for regions that do not contain fissionable material and/or did not generate any $F^*(r)$ tallies. When doing GPT sensitivity calculations using $F^*(r)$ CE TSUNAMI will produce two $F^*(r)$ meshes: one for the numerator term in the response of interest and another for the denominator term; therefore, 3dmaps that are produced from CE TSUNAMI GPT $F^*(r)$ calculations will contain two meshes, as shown in Fig. 6.2.5.

Table 6.2.5: CE TSUNAMI-3D parameters and default values.

Parameter	**Default value for TSUNAMI-3D**	**Description**
CET	1	Mode for CE TSUNAMI 0 = No sensitivity calculations 1 = CLUTCH sensitivity calculation 2 = IFP sensitivity calculation 4 = GEAR-MC calculation (with CLUTCH only) 5 = GEAR-MC calculation (with CLUTCH+IFP) 7 = Undersampling metric calculation
CFP	5	Number of latent generations used for IFP sensitivity or $F^*(r)$ calculations. If CET=1 and CFP= -1 then $F^*(r)$ is assumed to equal one everywhere. If CET=4 and CFP= -1 then $F^*(r)$ is assumed to equal zero everywhere.
CGD	NONE	ID of the gridgeom mesh used for CLUTCH $F^*(r)$ calculations.
FST	YES (when calculated)	Print the $F^*(r)$ grid values to a .3dmap file.

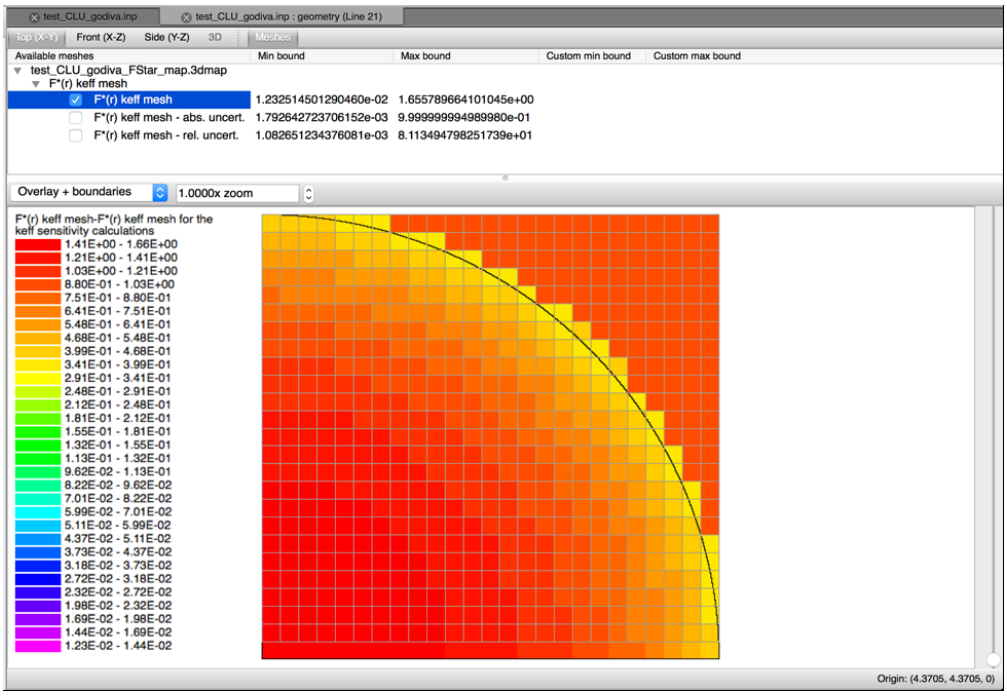


Fig. 6.2.4: $F^*(r)$ mesh from a sample CLUTCH eigenvalue sensitivity calculation.

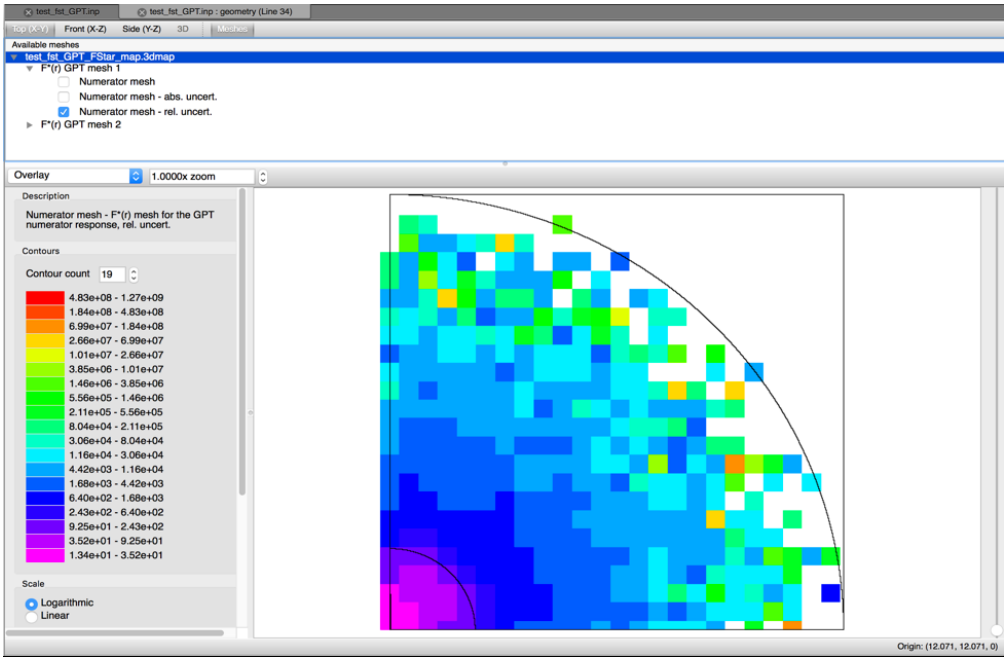


Fig. 6.2.5: $F^*(r)$ meshes from a sample CLUTCH GPT sensitivity calculation.

Setting CET=7 will activate an experimental capability for detecting computational biases due to the undersampling of fission sites and particle histories during a Monte Carlo simulation. This capability does not calculate sensitivity coefficients and does not require selecting a number of latent generations or building an F^* (r) mesh. Instead, this approach scores flux and reaction rate tallies for various materials and nuclides in a

system and reports the tallies to a *.sdf* file after the simulation ends, along with scores for various statistical metrics that may predict undersampling biases in the reported tallies. An undersampling metric calculation will produce four *.sdf* files, as described below:

*input_name*_metric1.sdf = Reaction rate and flux tallies
*input_name*_metric2.sdf = Number of nonzero scores per generation for each tally
*input_name*_metric3.sdf = Tally Entropy scores for each tally
*input_name*_metric4.sdf = Heidelberger-Welch RHW scores for each tally

Each of these undersampling metrics is described in detail in Reference [3d10]. The values reported in the _metric2/3/4.sdf files correspond to each of the tallies scored in the _metric1.sdf file. As in typical sensitivity coefficient *.sdf* files, the undersampling metric *.sdfs* report energy-dependent and energy-integrated information for various reaction rates in every material and nuclide in a system; however, the undersampling metric calculations also report information for flux tallies within each material/nuclide under the name of the fictitious isotope H-111. This undersampling metric capability is only peripherally related to the scope and application of sensitivity coefficient calculations and was included with CE TSUNAMI-3D predominantly to take advantage of the established TSUNAMI tally scoring framework and sensitivity visualization/postprocessing tools (i.e., Fulcrum).

Performing GPT sensitivity calculations with CE TSUNAMI requires the user to enter some additional input to specify the GPT response being examined in the calculation. For the IFP+CLUTCH GPT implementation (CET=5) users can define GPT reaction rate ratios using the "Definitions" and "SystemResponses" blocks (please see *Sensitivity and uncertainty calculation data* in the TSUNAMI-1D chapter), which are used similarly in TSUNAMI-1D. Both TSUNAMI-3D GPT capabilities can currently only accept total cross section (MT=1), fission (MT=18), n,gamma (MT=102), nu-fission (MT=452), or flux reaction response definitions. The CLUTCH-only GPT implementation (CET=4) currently cannot accept GPT response input from the Definitions and SystemResponses blocks, and may only calculate GPT sensitivity coefficients for one response per TSUNAMI-3D simulation. The input parameters for defining the GPT response in these (CET=4) calculations are described in Table 6.2.6. The two GEAR-MC implementations (CET=4 or CET=5) both require the user to specify a value for CFP; the CLUTCH-only GEAR-MC implementation requires the user to specify an $F^*(r)$ mesh using the gridGeom block and the CGD parameter.

Table 6.2.6: CE TSUNAMI-3D GPT sensitivity parameters and default values.

GEOMETRY (region)

Format: READ GEOM enter geometry region data here END GEOM

See sections on Geometry, Geometry data, Multiple entries in geometry data, and Alternative Sample problem mockups.

GEOMETRY REGION DATA consist of SIMPLE GEOMETRY REGION DATA and EXTENDED GEOMETRY REGION DATA.

ENTER GEOMETRY DATA IN THE FOLLOWING FORM:

OPTIONAL GLOBAL SPECIFICATION

UNIT n

OPTIONAL GEOMETRY COMMENT

GEOMETRY REGION DATA and/or EXTENDED GEOMETRY REGION DATA

****

ENTER SIMPLE REGION DATA IN THE FOLLOWING FORM:

GLOBAL Enter only to specify this unit as the global unit.

UNIT n

COM=delim comment delim This optional comment can be up to 132 characters long. It must begin and end with a delimiter (i.e. “ , ‘ , * , ^ , or !)

fgeom mix no. bias ID dimensions optional origin data (ORIGIN coordinates) optional chord data (CHORD distance) optional center data (CENTER type X-offset Y-offset Z-offset)

Enter as many geometry description specifications as necessary to describe the unit and as many units as necessary to describe the system.

SIMPLE GEOMETRY REGION INPUT DATA REQUIREMENTS

TYPE OF DATA	TYPE 1 DATA	TYPE 2 DATA	TYPE 3 DATA	TYPE 4 DATA	TYPE 5 DATA	TYPE 6 DATA
fgeom	SPHERE	XCYLINDER	YCYLINDER	CYLINDER	CUBE	CUBOID
	HEMISPHERE	XHEMICYL+Y	YHEMICYL+X	ZCYLINDER		
	HEMISPHE+X	XHEMICYL–Y	YHEMICYL–X	ZHEMICYL+X		
	HEMISPHE–X	XHEMICYL+Z	YHEMICYL+Z	ZHEMICYL–X		
	HEMISPHE+Y	XHEMICYL–Z	YHEMICYL–Z	ZHEMICYL+Y		
	HEMISPHE–Y			ZHEMICYL–Y		
	HEMISPHE+Z					
	HEMISPHE–Z					
dimensions	R (radius)	R +X –X	R +Y –Y	R +Z –Z	+X –X	+X –X +Y –Y +Z –Z
optional origin coordinates*	Enter the X Y Z coordinates of origin	Enter the Y Z coordinates of centerline	Enter the X Z coordinates of centerline	Enter the X Y coordinates of centerline	omit	omit
optional chord data**	Enter the dist. to plane	Enter the dist. to plane	Enter the dist. to plane	Enter the dist. to plane	omit	omit
optional center data***	Center type	X-offset	X-offset	Z-offset	omit	omit

*Enter ORIG or ORIGIN for *fgeom*.

**Enter CHORD for *fgeom*.

***Center data are only applicable when flux moments (PNM>0) or angular fluxes (NQD>0) are requested with the coordinate transform (TFM) set to YES.

NOTE: Chord data are not applicable for SPHERE, XCYLINDER, YCYLINDER, CYLINDER, ZCYLINDER, CUBE, or CUBOID.
Origin data are not applicable for a CUBE or CUBOID.

6.2.4.5 Sensitivity calculation data

A data block for controlling the sensitivity calculation is optional. If included, this data block begins with the keywords *READ SAMS* and ends with the keywords *END SAMS*. Any of the optional SAMS input data may be entered in free-form format between the *READ SAMS* and *END SAMS* keywords. The optional SAMS input data are shown with the default values specific to TSUNAMI-3D in Table 6.2.7. Certain options in Table 6.2.8 are not available for CE calculations. Parameters used to specify default covariance data to supplement or correct values on the files specified by *coverx*= are shown in Table 6.2.8. A more detailed explanation of the SAMS parameters is provided in the SAMS chapter.

Table 6.2.7: SAMS input keywords.

Keyword	**Default value**	**Description**
binsen	F	Produces SENPRO formatted binary sensitivity data file on unit 40
coverx=	56group-cov7.1	Name of covariance data file to use for uncertainty analysis
largeimp=	100.0	Value for the absolute value of implicit sensitivities, which if exceeded, will be reset to 0.0 and print a warning message.
makeimp*	F	Flag to cause implicit sensitivity coefficients to be generated.
nocovar	T	Flag to cause uncertainty edit to be turned off (sets print_covar to F)
nohtml	F	Flag to cause HTML output to not be produced.
nomix	F	Flag to cause the sensitivities by mixture to be turned off
pn=*	3	Legendre order for moment calculations
prtgeom*	F	Flag to cause the sensitivities to be output for each geometry region
prtimp*	F	Prints explicit sensitivities coefficients, implicit sensitivity coefficients and complete sensitivity coefficients
prtvols*	F	Flag to cause the volumes of the regions to be printed by SAMS
useang*	F	Flag to cause the angular flux calculated in KENO to be used to calculate flux moments for sensitivity calculations. If angular fluxes were not computed by KENO, *useang* is set to false internally.
usemom*	T	Flag to cause the flux moments calculated with KENO to be used to in sensitivity coefficient generation. If flux moments were not computed by KENO, *usemom* is set to false internally.
unconstrainedchi	F	Flag to generate pre-SCALE 6 unconstrained chi (fission spectrum) sensitivities
sensitivity_format=	txt	Specifies desired format for the resulting sensitivity data file (SDF). May be 'txt' for text-based format or 'hdf5' for HDF5-based format.
***MG only**		

Table 6.2.8: SAMS input keywords for default covariance data.

MIXTURES	Format: READ MIXT xsec parameters mixing table data. END MIXT Multigroup mode: These data are entered only if an AMPX working format library is being used. (LIB=) in the parameter data, s ee section on Title and parameter data. Do not enter if a Monte Carlo mixed library is used, (XSC=) in the parameter data. Continuous energy mode: Only the mixing table data are entered. See sections on Mixing table and Mixing table data.			
	XSEC PARAMETERS	consists of keywords and associated values. These parameters, if entered, only need to be entered once.		
		KEYWORD	DEFAULT	DEFINITION
		SCT=	1	number of discrete scattering angles 0 is isotropic 1 is P1 2 is P3 3 is P5
		EPS=	0.1	cross section message cutoff value used to suppress message numbers K5-060 and K6-060.
	MIXING TABLE DATA consists of	(1) a keyword and mixture ID for the mixture The keyword is MIX= The desired mixture number follows the keyword (1a) optionally, a keyword and nuclide mixture ID for the rest of the current mixture The keyword is NCM= The desired nuclide mixture ID follows the keyword (1b) optionally, a keyword and temperature for the mixture (used for continuous energy problems) The keyword is TMP= or TEM= The desired temperature in Kelvin follows the keyword (2) nuclide ID** (2a) optionally a keyword and nuclide mixture ID for this nuclide The keyword is NCM= and the desired nuclide mixture ID follows the keyword (3) number density** (4) continuous energy cross section filename** ** The sequence (2) (3) (4) is repeated for each nuclide in the mixture. Entry (4) is optional and entered with XS= name where name is the filename. REPEAT the sequence (1) (2)s (3)s (4)s until all the mixtures have been described.		

Additionally, user-defined covariance data can be specified for individual nuclides and reactions using the COVARIANCE data block. This data block begins with the keywords *READ COVARIANCE* and ends with the keywords *END COVARIANCE*. Any of the optional *COVARIANCE* input data may be entered in free-form format between the *READ COVARIANCE* and *END COVARIANCE* keywords. The specifications for the COVARIANCE data block are described in the "User Input Covariance Data" section of the TSUNAMI-IP chapter of the TSUNAMI Utility Modules manual.

As the SAMS module generates HTML output, the optional HTML data block provides user control over some formats of the output. This data block begins with the keywords *READ HTML* and ends with the keywords *END HTML*. Any of the optional HTML input data may be entered in free-form format between the *READ HTML* and *END HTML* keywords. The specifications for the HTML data block are described in the "HTML Data" section or the TSUNAMI-IP chapter of the TSUNAMI Utility Modules manual.

6.2.4.6 Input termination

The input for TSUNAMI-3D must terminate with a line containing *END* in column 1. This *END* terminates the control sequence.

6.2.5 SAMPLE PROBLEMS

Five sample problems are given in this section. In each example, the use of a new feature is explained to guide the user in the proper definition of input models so that reliable sensitivity coefficients can be obtained. The user must ensure that the necessary options are employed for each model to obtain accurate results. Please note, **the results shown here were generated with a previous version of SCALE**, so current data libraries and code implementations may produce different results. However, the techniques demonstrated are applicable to the current version of TSUNAMI-3D.

6.2.5.1 Generating reference direct perturbation sensitivity coefficients

The accuracy of the energy-integrated sensitivity coefficients can be confirmed through the use of central difference direct perturbation sensitivity calculations. Through this technique, the sensitivity of k_{eff} to the number density of particular nuclide can be obtained. This sensitivity of k_{eff} to the number density is equivalent to the sensitivity of k_{eff} to the total cross section integrated over energy. Because the total cross section sensitivity coefficient tests much of the data used to compute all other sensitivity coefficients, it is considered an adequate test for verification. For each sensitivity coefficient examined by direct perturbation, the k_{eff} of the system is computed first with the nominal values of the input quantities, then with a selected nominal input value increased by a certain percentage, and then with the nominal value decreased by the same percentage. The direct perturbation sensitivity coefficient of k_{eff} to some input value α is computed as

$$S_{k,\alpha} = \frac{\alpha}{k} \times \frac{dk}{d\alpha} = \frac{\alpha}{k} \times \frac{k_{\alpha^+} - k_{\alpha^-}}{\alpha^+ - \alpha^-} \tag{6.2.26}$$

where α^+ and α^- represent the increased and decreased values, respectively, of the input quantity α and k_{α^+} and k_{α^-} represent the corresponding values of k_{eff}. In general, perturbations used for calculating direct perturbation sensitivities should be large enough to induce a statistically significant change (10 σ_{keff}) in the eigenvalue of the system but not large enough to induce second-order effects in the perturbed eigenvalue. Statistical uncertainties in the computed values of k_{eff} are propagated to uncertainties in direct perturbation sensitivity coefficients by standard error propagation techniques as

$$\sigma_S = \left(\left(\frac{\left(\sigma_{k^+}^2 + \sigma_{k^-}^2\right)}{(k^+ - k^-)^2} + \frac{\sigma_k^2}{k^2} \right) \times \left(\frac{k^+ - k^-}{k} \right)^2 \right)^{1/2} \times \frac{\alpha}{\alpha^+ - \alpha^-} \tag{6.2.27}$$

In MG TSUNAMI sensitivity calculations it is important to ensure that the k_{eff} value of the forward and adjoint solutions closely agree. If the k_{eff} values do not agree, then the quality of at least one of the transport calculations may be in question. Typically, the transport calculation of concern is the adjoint calculation. By default, TSUNAMI-3D triples the number of histories per generation requested for the forward case to produce the adjoint solution. Experience has shown that agreement to less than 0.5% difference in k_{eff} between the forward and adjoint calculations is adequate to obtain accurate sensitivity coefficients.

6.2.5.2 Simple MG Sample Problem

A simple sample problem with INFHOMMEDIUM MG cross section processing is based on an unreflected rectangular parallelepiped consisting of a homogeneous mixture UF_4 and paraffin with an enrichment of 2 wt% in ^{235}U. The H/^{235}U atomic ratio is 294:1. The dimensions of the experiment are 56.22 × 56.22 × 122.47 cm [3d11]. For consistency with a TSUNAMI-1D model of the same sample problem, this experiment was modeled as a sphere with a critical radius of 38.50 cm. This configuration is used in the TSUNAMI-3D_K5-1 and TSUNAMI-3D_K6-1 sample problems distributed with SCALE. An annotated TSUNAMI-3D-K5 input for this experiment is shown in Fig. 6.2.6. The composition data are input as number densities for each nuclide. Because the material is treated as INFHOMMEDIUM, no explicit unit cell model is necessary, and the READ CELL block is omitted. The KENO V.a problem description contains parameter data to request 10,000 generations (*gen=10000*) with 3000 neutrons per generation (*npg=3000*), deactivate the HTML output for KENO V.a (*htm=no*), stop the forward calculation when k_{eff} has converged to one standard deviation of 0.005 (*sig=0.005*), and to stop the adjoint calculation when k_{eff} has converged to one standard deviation of 0.010 (*asg=0.010*). The KENO V.a geometry consists of nine concentric spheres, up to an outer radius of 38.5 cm, with each sphere containing the material defined as mixture 1. The geometry subdivision is necessary to adequately resolve the spatial dependence of the angular moments of the forward and adjoint flux solutions. The optional sensitivity calculation data block is used to request edits of the sensitivity coefficients

for each region (*prtgeom*) and edits of the explicit, implicit, and complete sensitivity coefficients *(prtimp*). The code output from each functional module is not given here, but is described in the manual section for each functional module.

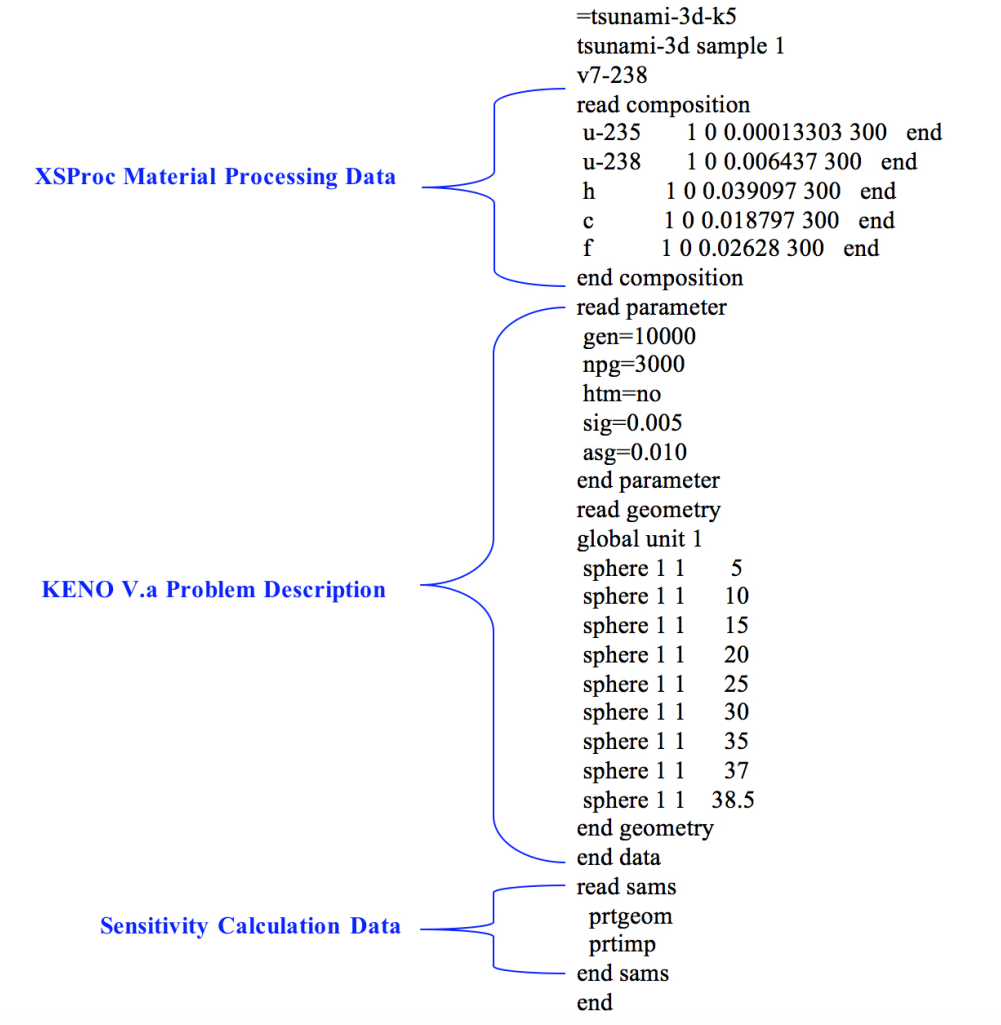


Fig. 6.2.6: TSUNAMI-3D-K5 simple sample problem input.

For this problem, direct perturbation results were obtained for the number densities of each nuclide using an equivalent 1D model. In these calculations, the number density of each nuclide was perturbed by ± to the number density is equivalent to the sensitivity of k_{eff} to the total cross section, integrated over energy. The direct perturbation sensitivity coefficients were computed by using the k_{eff} values from the unperturbed and perturbed cases in Eq. (6.2.27).

This experiment was modeled with the nine-region model shown in Fig. 6.2.6, and as a single computational region (not shown). Flux moments were expanded to third order in both cases, which is the default configuration. TSUNAMI-3D-K5 automatically increases the number of particles per generation by a factor of three for the adjoint analysis. For the single region case, the k_{eff} values for the forward and adjoint cases are in good agreement at 1.00682 +/- 0.00094, and 1.0013 +/- 0.0049, respectively. The sensitivity results shown in Table 6.2.9 were extracted from the output file from the edit titled “Energy, Region, and Mixture

Integrated Sensitivity Coefficients for this Problem." The uncertainty in the sensitivity coefficients represents one standard deviation and is present due to the use of Monte Carlo methods to compute the fluxes and k_{eff}. These results indicate similarity with the direct perturbation results for some nuclides but not for others. Differences between the TSUNAMI-3D-K5 results and the direct perturbation results vary 1% for ^{238}U up to 16% for ^{1}H. The results from the TSUNAMI-3D-K5 analysis with the model divided into nine spherical shells, with all other parameters held constant, are also shown in Table 6.2.9. These results compare much more favorably with the direct perturbation results. For this model, all TSUNAMI-3D-K5 sensitivities agree with the direct perturbation values within 0.1% for ^{1}H up to a maximum difference of 1.5% for ^{238}U.

The differences in the results from the two TSUNAMI-3D models, one region and nine regions, are due to the summation of the product of the forward and adjoint fluxes over the regions in the problem. For a region in which the flux moments vary greatly by position, subdividing the geometry will provide better resolution of the variation of the flux across the system and will produce more accurate results. The number of regions necessary for accurate computation of the sensitivity coefficients was determined through an iterative process. Models divided into more regions produce the equivalent results to those produced by the nine-region model. Increasing the number of computational regions increases the run time for this problem by about 10%.

The sensitivity results from the nine-region model using TSUNAMI-3D-K5 with PARM=CENTRM, which does not include the contributions from the implicit sensitivity coefficients, are also shown in Table 6.2.9. The differences between the TSUNAMI-3D-K5 PARM=CENTRM and the direct perturbation results are 16% for ^{1}H and 19% for ^{238}U. The use of the default cross section processing with the sensitivity versions of the resonance processing codes is strongly recommended. However, TSUNAMI-3D-K5 with PARM=CENTRM should produce accurate results for fast systems where resonance self-shielding is not important. This is illustrated with the second sample problem for TSUNAMI-1D and will not be repeated here.

Table 6.2.9: Energy- and region-integrated sensitivity coefficients from TSUNAMI-3D UF_4 sample problem.

PLOT Format: READ PLOT plot parameters END PLOT plot parameters must be entered for each plot that is to be made.
See sections Plot data and Color plots.

KEYWORD	DEFAULT	DEFINITION
TTL=	prob. title	delim ptitl delim delim is a one-character delimiter (i.e. " , ' , * , ^ , or !) that signals the beginning and end of the title.
		ptitl is the plot title (max. 132 char.)
PIC=	MAT	Type of plot: MIXTURE NO., UNIT NO. or BIAS ID NO.
		MIXTURE ---------MAT MIX MIXT MIXTURE MEDI MEDIA
		UNIT NO. ----------UNT UNIT UNITTYPE
		BIAS ID NO. ------IMP BIAS BIASID WTS WEIG WEIGHTS WGT WGTS
XUL=	prev. plot	X coord. of upper left corner of plot
YUL=	prev. plot	Y coord. of upper left corner of plot
ZUL=	prev. plot	Z coord. of upper left corner of plot
XLR=	prev. plot	X coord. of lower right corner of plot
YLR=	prev. plot	Y coord. of lower right corner of plot
ZLR=	prev. plot	Z coord. of lower right corner of plot

KEYWORD	DEFAULT	DEFINITION
UAX=	prev. plot 0 If VAX OR WAX is read	X component of direction cosine for the AX axis of the plot (across)
VAX=	prev. plot 0 IF UAX OR WAX is read	Y component of direction cosine for the AX axis of the plot (across)
WAX=	prev. plot 0 IF UAX OR VAX is read	Z component of direction cosine for the AX axis of the plot (across)
UDN=	prev. plot 0 IF VDN OR WDN is read	X component of direction cosine for the DN axis of the plot (down)
VDN=	prev. plot 0 IF UDN OR WDN is read	Y component of direction cosine for the DN axis of the plot (down)
WDN=	prev. plot 0 IF UDN OR VDN is read	Z component of direction cosine for the DN axis of the plot (down)
DLX=		Horizontal spacing between points on plot
DLD=		Vertical spacing between points on plot
NAX=		No. of intervals to be printed across page
NDN=		No. of intervals to be printed down page
LPI=	8.0 (character plots) 10 (color plots)	Vertical to horizontal scaling factor for plot proportionality.
RUN=	YES	YES allows the problem to execute NO terminates problem after data checking
PLT=	YES	YES allows the plot(s) to be made NO allows reading the plot data without making a plot
SCR=	YES	Display plot method SCR=YES utilizes color plot display SCR=NO utilizes printer plot display
NCH=	CHRS[a]	delim CHRS delim a one character delimiter[b] signals the beginning and end of the character string
CLR=	See Table on default color specifications for the color plot display method	num(i) red(num(i)) green (num(i)) blue (num(i)) num(i) defines mix. no., unit no., or bias ID next 3 entries define red, green, and blue components of the color representing num(i).

PLOT ORIGIN:

(1) SINGLE UNIT - coincides with origin of geometry description.

(2) BASE ARRAY - at the most negative point of the global array

(3) REFLECTED ARRAY - coincides with the origin of the CORE or ARRAY description of the global array.

[a] Default values of CHRS are given below:

MEDIA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
CHRS		1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	I	J	K	L	M
MEDIA	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42			
CHRS	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	#	.	$	-	+	)	|		
MEDIA	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58							
CHRS	&	>	:	;	.	-	%	*	"	=	!	(	@	<	/	0							

[b] Acceptable delimiters are " , ' , * , ^ , or !.

continues on next page

Table 6.2.10 – continued from previous page

Table 6.2.10: Energy- and region-integrated sensitivity coefficients from TSUNAMI-3D UF_4 sample problem (continued).

Isotope	Reaction	Direct perturbation	TSUNAMI-3D one computational region	TSUNAMI-3D nine computational regions	TSUNAMI-3D PARM=CENTRM nine computational regions
^{235}U	total	2.52E-01	2.31E-01 ± 8.79E-05	2.50E-01 ± 4.36E-05	5.13E-04 ± 1.22E-07
^{235}U	scatter		5.10E-04 ± 9.11E-08	4.42E-04 ± 1.22E-07	2.76E-04 ± 1.08E-07
^{235}U	elastic		2.57E-04 ± 4.65E-08	2.05E-04 ± 1.08E-07	2.20E-04 ± 4.62E-08
^{235}U	n,n'		2.36E-04 ± 6.35E-08	2.20E-04 ± 4.62E-08	1.73E-05 ± 4.49E-09
^{235}U	n,2n		1.65E-05 ± 8.98E-09	1.73E-05 ± 4.49E-09	3.63E-01 ± 5.30E-05
^{235}U	fission		3.47E-01 ± 1.10E-04	3.63E-01 ± 5.30E-05	-1.13E-01 ± 1.27E-05
^{235}U	capture		-1.16E-01 ± 2.86E-05	-1.13E-01 ± 1.27E-05	-1.13E-01 ± 1.27E-05
^{235}U	n,γ		-1.16E-01 ± 2.86E-05	-1.13E-01 ± 1.27E-05	9.50E-01 ± 1.65E-04
^{235}U	nubar		9.51E-01 ± 3.64E-04	9.50E-01 ± 1.65E-04	8.52E-08 ± 9.89E-05
^{235}U	chi		8.54E-08 ± 2.18E-04	8.52E-08 ± 9.89E-05	-2.47E-01 ± 2.33E-05
^{238}U	total	-2.08E-01	-2.11E-01 ± 5.14E-05	-2.05E-01 ± 2.39E-05	2.52E-02 ± 5.17E-06
^{238}U	scatter		5.29E-02 ± 7.93E-06	4.88E-02 ± 5.85E-06	1.13E-02 ± 3.86E-06
^{238}U	elastic		3.82E-02 ± 6.53E-06	3.49E-02 ± 4.73E-06	1.29E-02 ± 2.95E-06
^{238}U	n,n'		1.37E-02 ± 3.83E-06	1.29E-02 ± 2.95E-06	1.03E-03 ± 3.01E-07
^{238}U	n,2n		9.85E-04 ± 5.90E-07	1.03E-03 ± 3.01E-07	3.35E-02 ± 5.49E-06
^{238}U	fission		3.24E-02 ± 1.18E-05	3.35E-02 ± 5.49E-06	-3.06E-01 ± 2.14E-05
^{238}U	capture		-2.96E-01 ± 4.82E-05	-2.87E-01 ± 2.16E-05	-3.06E-01 ± 2.14E-05
^{238}U	n,γ		-2.96E-01 ± 4.82E-05	-2.87E-01 ± 2.16E-05	5.02E-02 ± 1.11E-05
^{238}U	nubar		4.85E-02 ± 2.37E-05	5.02E-02 ± 1.11E-05	4.53E-09 ± 5.18E-06
^{238}U	chi		4.38E-09 ± 1.10E-05	4.53E-09 ± 5.18E-06	5.13E-04 ± 1.22E-07

continues on next page

The uncertainty information from SAMS for the UF_4 sample problem is shown in Example 6.2.1. Based on the 44GROUPCOV covariance data file, the uncertainty in k_{eff} due to these covariance data is 0.6110% $\Delta k/k$. A more detailed description of the uncertainty information is given in the SAMS chapter.

The energy-dependent sensitivity data are available in the sensitivity data file, which is returned to the same directory as the input file and given the same name as the user's input file with the extension `.sdf`. In the case of the nine-region model, the sensitivity data file contains 495 individual sensitivity profiles with varying reaction types, each in the 238-group energy structure. There are 45 profiles that are integrated over all regions, one for each reaction of each nuclide in the system. The sum of the sensitivity coefficients for the same nuclide in all mixtures is printed unless *nomix* is entered in the SAMS data block, so there are an additional 45 profiles, one for each reaction of each nuclide in mixture 1. Additionally, because *prtgeom* was entered in the SAMS data block, each reaction of each nuclide for each region in the system model is represented with a sensitivity profile. There are nine regions in the model, each with 45 sensitivity profiles, making for a total of 495 sensitivity profiles on the data file and a total of 117,810 energy-dependent sensitivity coefficients.

Some plots of the energy-dependent sensitivity data from the nine-region model of the sample problem were generated with the plotting capabilities of Fulcrum. Energy-dependent sensitivity profiles for ^{235}U fission and ^{1}H elastic scattering are shown in Fig. 6.2.7. The error bars represent one standard deviation for the statistical uncertainty due to the use of Monte Carlo methods to compute the fluxes and k_{eff}.

Example 6.2.1: Uncertainty information from UF_4 sample problem.

```
-----------------------------
  Uncertainty Information
-----------------------------


the relative standard deviation of k-eff (% delta-k/k)
due to cross-section covariance data is:

  0.6110 +/- 0.0000 % delta-k/k

 contributions to uncertainty in k-eff (% delta-k/k) by
 individual energy covariance matrices:

 covariance matrix

        nuclide-reaction        with        nuclide-reaction            % delta-k/k due to this matrix
 ------------------------------      ------------------------------  ----------------------------------
         u-238 n,gamma                       u-238 n,gamma                3.8714E-01 +/- 6.2871E-06
         u-235 nubar                         u-235 nubar                  2.8509E-01 +/- 7.9001E-06
         u-238 n,n'                          u-238 n,n'                   2.2073E-01 +/- 7.7594E-06
         u-235 n,gamma                       u-235 n,gamma                1.6006E-01 +/- 1.7559E-06
          f-19 elastic                        f-19 elastic                1.3624E-01 +/- 5.0707E-06
         u-238 elastic                       u-238 n,n'                  -1.2828E-01 +/- 1.7674E-06
         u-235 fission                       u-235 n,gamma                1.2387E-01 +/- 8.3076E-07
         u-235 fission                       u-235 fission                1.2134E-01 +/- 1.2085E-06
           h-1 elastic                         h-1 elastic                1.1972E-01 +/- 2.1606E-06
          f-19 elastic                        f-19 n,n'                  -1.1793E-01 +/- 3.1965E-06
          f-19 n,n'                           f-19 n,n'                   1.1286E-01 +/- 3.8652E-06
         u-235 chi                           u-235 chi                    8.8178E-02 +/- 1.5583E-05
         u-238 elastic                       u-238 elastic                6.9520E-02 +/- 1.1586E-06
         u-238 nubar                         u-238 nubar                  5.8614E-02 +/- 5.4192E-07
           h-1 n,gamma                         h-1 n,gamma                5.0829E-02 +/- 1.6728E-07
         u-238 elastic                       u-238 n,gamma                5.0286E-02 +/- 1.7408E-06
```

(continues on next page)

(continued from previous page)

f-19 n,alpha	f-19 n,alpha	1.9795E-02 +/- 1.0127E-07
u-238 fission	u-238 fission	1.7394E-02 +/- 3.4394E-08
c elastic	c elastic	1.5520E-02 +/- 5.5754E-08
u-238 n,2n	u-238 n,2n	1.3981E-02 +/- 1.2056E-07
f-19 n,gamma	f-19 n,gamma	9.7994E-03 +/- 6.0845E-09
c n,n'	c elastic	-9.0325E-03 +/- 3.2330E-08
c n,n'	c n,n'	8.6479E-03 +/- 5.6289E-08
f-19 elastic	f-19 n,alpha	6.6750E-03 +/- 1.2243E-08
u-238 chi	u-238 chi	5.8854E-03 +/- 6.7274E-08
u-235 elastic	u-235 n,gamma	4.4783E-03 +/- 8.4753E-09
u-235 elastic	u-235 fission	-3.3039E-03 +/- 1.0089E-08
u-238 fission	u-238 n,gamma	2.7661E-03 +/- 8.1090E-10
f-19 n,p	f-19 n,p	2.0897E-03 +/- 1.3810E-09
u-238 elastic	u-238 n,2n	-1.9405E-03 +/- 1.8719E-09
u-238 elastic	u-238 fission	-1.8278E-03 +/- 3.9798E-10
c n,alpha	c n,alpha	1.6271E-03 +/- 1.2097E-09
c n,gamma	c n,gamma	1.4922E-03 +/- 1.4387E-10
u-235 n,n'	u-235 n,n'	1.3833E-03 +/- 2.5316E-10
u-235 elastic	u-235 n,n'	-8.8072E-04 +/- 6.2537E-11
f-19 elastic	f-19 n,p	5.9136E-04 +/- 2.5027E-10
f-19 elastic	f-19 n,gamma	4.4592E-04 +/- 4.9929E-10
u-235 elastic	u-235 elastic	4.3974E-04 +/- 3.1277E-11
f-19 n,d	f-19 n,d	2.7814E-04 +/- 4.7485E-11
u-235 n,2n	u-235 n,2n	1.5578E-04 +/- 1.1389E-11
c n,n'	c n,alpha	-1.2708E-04 +/- 7.1578E-10
f-19 elastic	f-19 n,2n	-6.9247E-05 +/- 1.2530E-11
f-19 elastic	f-19 n,d	6.6212E-05 +/- 1.0124E-11
f-19 n,t	f-19 n,t	6.3390E-05 +/- 3.8183E-12
u-235 elastic	u-235 n,2n	-2.7965E-05 +/- 3.3228E-13
f-19 n,2n	f-19 n,2n	2.1907E-05 +/- 1.0175E-12
f-19 n,n'	f-19 n,2n	-1.9896E-05 +/- 5.7304E-12
f-19 elastic	f-19 n,t	1.4406E-05 +/- 5.0542E-13
c n,n'	c n,gamma	7.0412E-06 +/- 1.1010E-14
c n,d	c n,d	6.9094E-07 +/- 9.1360E-15
c n,p	c n,p	3.8587E-07 +/- 2.0164E-15
c n,n'	c n,d	-2.8704E-07 +/- 0.0000E+00
c n,n'	c n,p	-1.4206E-07 +/- 0.0000E+00

Note: relative standard deviation in k-eff can be computed from
individual values by adding the square of the values with positive signs and
subtracting the square of the values with negative signs, then taking the square root

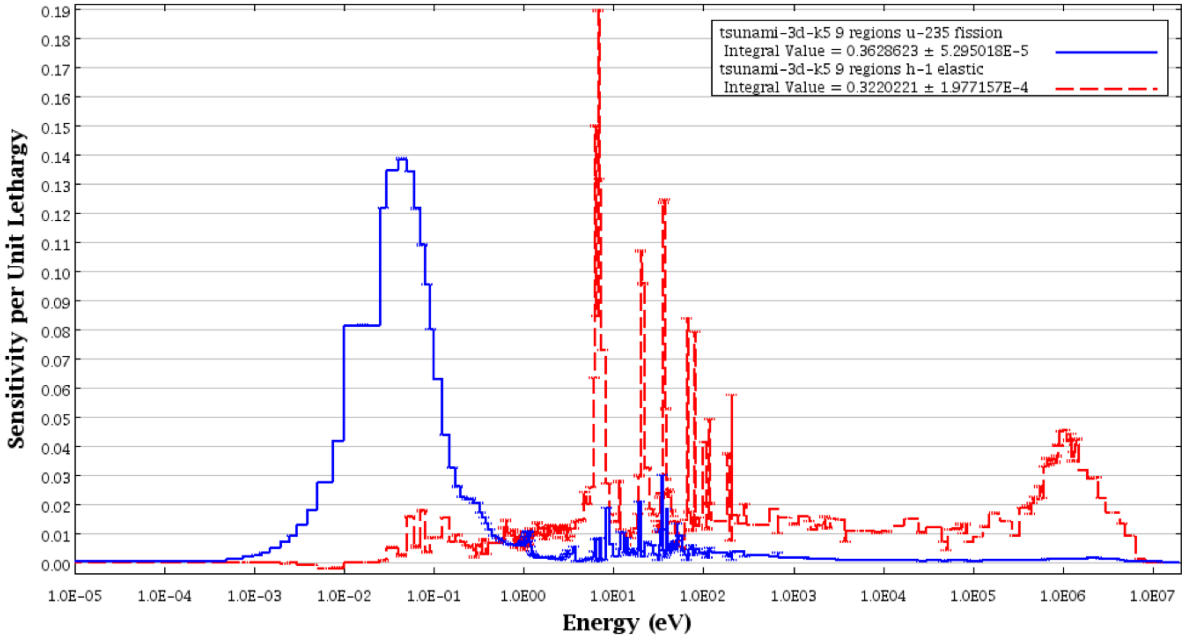


Fig. 6.2.7: Energy-dependent sensitivity profiles from TSUNAMI-3D-K5 for simple sample problem.

Multigroup sample problem with spatial flux mesh

In the previous example, subdivision of the system geometry was necessary to obtain adequate resolution of the flux solution to obtain the appropriate product of the forward and adjoint fluxes necessary for the sensitivity calculations. To simplify the geometry refinement procedure, the meshing of KENO V.a or KENO-VI is used where fluxes are tallied in a cubic mesh that is superimposed over each region of the system geometry. If mesh fluxes are generated in the KENO solution, they are automatically used by the SAMS module in the calculation of the sensitivity coefficients.

To demonstrate the use of the mesh flux option in TSUNAMI-3D-K5, another simple system has been selected. This system is an unreflected rectangular parallelepiped consisting of a homogeneous mixture of UF_4 and paraffin with an enrichment of 2 wt% in ^{235}U. The H/^{235}U atomic ratio is 972:1. The dimensions of the experiment are 81.45 × 86.70 × 88.22 cm. This system is identified as LEU-COMP-THERM-033 case 45 from the *International Handbook of Evaluated Criticality Safety Benchmark Experiments* (IHECSBE). [3d11] The model provided in the IHECSBE was converted to a TSUNAMI-3D-K5 input and is shown in Example 6.2.2. Here the experiment is modeled as a single cuboid. This model is included as an example and is not distributed as a TSUNAMI-3D sample problem.

Direct perturbation sensitivity results were obtained for ^{235}U, ^{238}U, and ^{1}H with a ± value converged to a standard deviation of 0.0001 (*sig=0.0001*). For the model shown in Example 6.2.2, the forward and adjoint k_{eff} values agreed well at 0.99250 ± 0.00056 and 0.9905 ± 0.0049, respectively. The direct perturbation results and the TSUNAMI-3D-K5 results are shown in Table 6.2.11. When modeled as a single region, the TSUNAMI-3D-K5 results differed from the direct perturbation results by 2.6% for ^{238}U, 3.1% for ^{235}U, and 15% for ^{1}H.

To improve the agreement in the values, the model was divided into 6 cuboids nested inside of each other. The input for this model is given in Example 6.2.3 and is illustrated in Fig. 6.2.8. The results for this model

are also given in Table 6.2.10. Here the TSUNAMI-3D-K5 sensitivity coefficients agree with the direct perturbation within 0.3% for ^{238}U, 1.7% for ^{235}U, and 0.8% for ^{1}H.

To simplify the generation of a refined geometrical representation of this system, the same geometry as the initial model was used with an 8 cm mesh for the flux tallies. The input for this model is shown in Example 6.2.4. The mesh is illustrated in Fig. 6.2.9, using a larger (15 cm) mesh interval for illustrative purposes. The mesh flux option is activated by entering *mfx=yes* in the input, and the size of the mesh is defined with the *msh=8* entry. The 8 following *msh=* indicates that a cubic grid with a length of 8 cm on the side of each cube will be superimposed on each geometry region. For this system, the flux will be tallied in 1728 mesh intervals. When the forward and adjoint mesh flux solutions are processed by the SAMS module, the product of the forward and adjoint solutions are produced for each mesh interval, then summed for each region. This technique provides a simple input parameter to produce accurate sensitivity coefficients. The results from TSUNAMI-3D-K5 with an 8 cm mesh flux are given in Table 6.2.11. Similar to the results with the manual subdivision, sensitivity coefficients agree with the direct perturbation within 0.7% for ^{238}U, 2.1% for ^{235}U, and 0.03% for ^{1}H.

The sensitivity results for ^{235}U and ^{238}U show little variation with modifications in the geometry subdivision. The products of the forward and adjoint flux moments, which are derived from the angular flux solution, are most impacted by the mesh flux. The flux moments are used to compute the scattering terms of the sensitivity coefficients. For isotopes with limited scattering cross sections, such as ^{235}U and ^{238}U, the impact of refinement of the flux solution is reduced.

Example 6.2.2: TSUNAMI-3D-K5 input for LEU-COMP-THERM-033 case 45.

```
=tsunami-3d-k5
uf4 paraffin mixture u2f4-6
v6-238
read composition
 h-poly      6 0 0.060586 300   end
 f           6 0 0.012302 300   end
 c           6 0 0.029128 300   end
 u-235       6 0 6.2282e-05 300   end
 u-238       6 0 0.0030126 300   end
 u-234       6 0 6.2548e-07 300   end
end composition
read parameter
 gen=10000
 npg=10000
 sig=0.0001
end parameter
read geometry
global unit 1
 cuboid 6 1   40.725  -40.725    43.35   -43.35    44.11   -44.11
end geometry
end data
read sams
  prtgeom prtimp
end sams
end
```

continues on next page

Table 6.2.11 – continued from previous page

Table 6.2.11: Energy- and region-integrated sensitivity coefficients from LEU-COMP-THERM-033 case 45 sample problem.

Isotope	Reaction	Direct perturbation			TSUNAMI-3D one computational region			TSUNAMI-3D six computational regions			TSUNAMI-3D 8 cm mesh (1728 mesh regions)		
$^{1}_{H}$	total	-1.00E-01	±	4.74E-03	-8.46E-02	±	7.44E-05	-1.01E-01	±	2.70E-04	-1.00E-01	±	1.51E-03
$^{1}_{H}$	scatter				1.93E-01	±	5.26E-05	1.79E-01	±	2.65E-04	1.79E-01	±	1.50E-03
$^{1}_{H}$	elastic				1.93E-01	±	5.26E-05	1.79E-01	±	2.65E-04	1.79E-01	±	1.50E-03
$^{1}_{H}$	capture				-2.77E-01	±	5.12E-05	-2.80E-01	±	3.20E-05	-2.79E-01	±	5.44E-05
$^{1}_{H}$	n,γ				-2.77E-01	±	5.12E-05	-2.80E-01	±	3.20E-05	-2.79E-01	±	5.44E-05
$^{12}_{C}$	total	1.84E-02	±	5.46E-03	2.34E-02	±	4.86E-06	1.52E-02	±	1.82E-05	1.65E-02	±	1.04E-04
$^{12}_{C}$	scatter				2.50E-02	±	4.85E-06	1.68E-02	±	1.82E-05	1.81E-02	±	1.04E-04
$^{12}_{C}$	elastic				2.46E-02	±	4.84E-06	1.65E-02	±	1.82E-05	1.78E-02	±	1.04E-04
$^{12}_{C}$	n,n'				3.60E-04	±	1.35E-07	3.36E-04	±	1.15E-07	3.47E-04	±	6.72E-07
$^{12}_{C}$	capture				-1.62E-03	±	2.67E-07	-1.64E-03	±	1.69E-07	-1.63E-03	±	3.17E-07
$^{12}_{C}$	n,γ				-1.36E-03	±	2.50E-07	-1.37E-03	±	1.56E-07	-1.37E-03	±	2.66E-07
$^{12}_{C}$	n,p				-4.87E-08	±	4.71E-11	-2.75E-08	±	3.05E-11	-4.52E-08	±	8.03E-10
$^{12}_{C}$	n,d				-1.32E-07	±	1.54E-10	-5.57E-08	±	8.13E-11	-1.17E-07	±	2.67E-09
$^{12}_{C}$	n,α				-2.63E-04	±	9.55E-08	-2.68E-04	±	6.33E-08	-2.58E-04	±	1.73E-07
$^{19}_{F}$	total	1.18E-02	±	5.26E-03	1.03E-02	±	1.97E-06	6.16E-03	±	6.04E-06	6.90E-03	±	3.54E-05
$^{19}_{F}$	scatter				1.35E-02	±	2.03E-06	9.42E-03	±	6.02E-06	1.01E-02	±	3.53E-05
$^{19}_{F}$	elastic				8.86E-03	±	1.57E-06	5.71E-03	±	5.71E-06	6.22E-03	±	3.31E-05
$^{19}_{F}$	n,n'				4.66E-03	±	6.67E-07	3.71E-03	±	1.07E-06	3.90E-03	±	6.34E-06
$^{19}_{F}$	n,2n				1.69E-06	±	5.84E-10	1.53E-06	±	5.20E-10	1.70E-06	±	4.82E-09
$^{19}_{F}$	capture				-3.24E-03	±	5.40E-07	-3.26E-03	±	3.39E-07	-3.23E-03	±	6.03E-07
$^{19}_{F}$	n,γ				-1.76E-03	±	3.00E-07	-1.78E-03	±	1.88E-07	-1.77E-03	±	3.19E-07
$^{19}_{F}$	n,p				-1.03E-04	±	3.50E-08	-1.04E-04	±	2.23E-08	-1.01E-04	±	4.57E-08
$^{19}_{F}$	n,d				-4.97E-06	±	1.63E-09	-4.99E-06	±	1.10E-09	-4.85E-06	±	3.69E-09
$^{19}_{F}$	n,t				-1.13E-06	±	4.63E-10	-1.10E-06	±	3.33E-10	-1.10E-06	±	1.55E-09
$^{19}_{F}$	n,α				-1.36E-03	±	4.23E-07	-1.37E-03	±	2.66E-07	-1.35E-03	±	4.79E-07
$^{234}_{U}$	total	1.51E-03	±	5.79E-03	-1.13E-03	±	2.10E-07	-1.15E-03	±	1.32E-07	-1.14E-03	±	2.31E-07
$^{234}_{U}$	scatter				1.68E-06	±	1.95E-10	1.25E-06	±	1.70E-10	1.34E-06	±	2.19E-09
$^{234}_{U}$	elastic				1.02E-06	±	1.14E-10	6.53E-07	±	1.21E-10	7.19E-07	±	1.84E-09
$^{234}_{U}$	n,n'				6.38E-07	±	1.14E-10	5.79E-07	±	8.78E-11	5.95E-07	±	5.33E-10

(continued)

Isotope	Reaction	Direct perturbation			TSUNAMI-3D one computational region			TSUNAMI-3D six computational regions			TSUNAMI-3D 8 cm mesh (1728 mesh regions)		
$^{234}_{U}$	n,2n				1.03E-08	±	3.49E-12	1.05E-08	±	2.16E-12	1.01E-08	±	4.95E-12
$^{234}_{U}$	fission				1.59E-05	±	2.12E-09	1.57E-05	±	1.27E-09	1.57E-05	±	2.13E-09
$^{234}_{U}$	capture				-1.15E-03	±	2.10E-07	-1.16E-03	±	1.32E-07	-1.16E-03	±	2.31E-07
$^{234}_{U}$	n,γ				-1.15E-03	±	2.10E-07	-1.16E-03	±	1.32E-07	-1.16E-03	±	2.31E-07
$^{234}_{U}$	nubar				2.68E-05	±	4.50E-09	2.65E-05	±	2.73E-09	2.65E-05	±	4.46E-09
$^{234}_{U}$	chi				5.64E-11	±	3.20E-09	5.59E-11	±	2.06E-09	5.59E-11	±	3.30E-09
$^{235}_{U}$	total	3.94E-01	±	5.09E-03	4.06E-01	±	8.85E-05	4.01E-01	±	5.42E-05	4.02E-01	±	9.57E-05
$^{235}_{U}$	scatter				1.18E-04	±	1.52E-08	8.51E-05	±	3.97E-08	9.08E-05	±	2.36E-07
$^{235}_{U}$	elastic				7.04E-05	±	9.54E-09	4.37E-05	±	3.68E-08	4.81E-05	±	2.29E-07
$^{235}_{U}$	n,n'				4.53E-05	±	7.94E-09	3.88E-05	±	7.75E-09	4.02E-05	±	4.28E-08
$^{235}_{U}$	n,2n				5.06E-06	±	1.73E-09	5.09E-06	±	1.06E-09	4.99E-06	±	2.21E-09
$^{235}_{U}$	fission				4.93E-01	±	1.02E-04	4.89E-01	±	6.26E-05	4.90E-01	±	1.07E-04
$^{235}_{U}$	capture				-8.69E-02	±	1.55E-05	-8.78E-02	±	9.71E-06	-8.76E-02	±	1.65E-05
$^{235}_{U}$	n,γ				-8.69E-02	±	1.55E-05	-8.78E-02	±	9.71E-06	-8.76E-02	±	1.65E-05
$^{235}_{U}$	nubar				9.78E-01	±	2.60E-04	9.79E-01	±	1.60E-04	9.79E-01	±	2.61E-04
$^{235}_{U}$	chi				5.20E-07	±	1.38E-04	5.20E-07	±	8.48E-05	5.20E-07	±	1.38E-04
$^{238}_{U}$	total	-1.63E-01	±	5.62E-03	-1.59E-01	±	2.48E-05	-1.63E-01	±	1.57E-05	-1.62E-01	±	3.07E-05
$^{238}_{U}$	scatter				1.88E-02	±	2.25E-06	1.70E-02	±	2.07E-06	1.74E-02	±	1.01E-05
$^{238}_{U}$	elastic				1.47E-02	±	2.10E-06	1.34E-02	±	1.96E-06	1.36E-02	±	9.24E-06
$^{238}_{U}$	n,n'				3.67E-03	±	6.42E-07	3.12E-03	±	6.09E-07	3.24E-03	±	3.38E-06
$^{238}_{U}$	n,2n				4.84E-04	±	1.79E-07	4.89E-04	±	1.12E-07	4.76E-04	±	2.49E-07
$^{238}_{U}$	fission				1.44E-02	±	3.21E-06	1.42E-02	±	1.93E-06	1.41E-02	±	3.20E-06
$^{238}_{U}$	capture				-1.92E-01	±	2.42E-05	-1.94E-01	±	1.52E-05	-1.93E-01	±	2.67E-05
$^{238}_{U}$	n,γ				-1.92E-01	±	2.42E-05	-1.94E-01	±	1.52E-05	-1.93E-01	±	2.67E-05
$^{238}_{U}$	nubar				2.15E-02	±	6.43E-06	2.13E-02	±	3.89E-06	2.12E-02	±	6.34E-06
$^{238}_{U}$	chi				-1.54E-08	±	3.07E-06	-1.52E-08	±	1.85E-06	-1.52E-08	±	3.02E-06

continues on next page

Table 6.2.11 – continued from previous page

Example 6.2.3: TSUNAMI-3D-K5 input LEU-COMP-THERM-033 case 45 with manual geometrical subdivision.

```
=tsunami-3d-k5
uf4 paraffin mixture u2f4-6
v6-238
read composition
 h-poly      6 0 0.060586 300   end
 f           6 0 0.012302 300   end
 c           6 0 0.029128 300   end
 u-235       6 0 6.2282e-05 300   end
 u-238       6 0 0.0030126 300   end
 u-234       6 0 6.2548e-07 300   end
end composition
read parameter
 gen=10000
 npg=10000
 sig=0.0001
end parameter
read geometry
global unit 1
 cuboid 6 1       10      -10       10      -10       10      -10
 cuboid 6 1       20      -20       20      -20       20      -20
 cuboid 6 1       30      -30       30      -30       30      -30
 cuboid 6 1       35      -35       35      -35       35      -35
 cuboid 6 1       38      -38       41      -41       42      -42
 cuboid 6 1   40.725  -40.725    43.35   -43.35    44.11   -44.11
end geometry
end data
read sams
  prtgeom prtimp
end sams
end
```

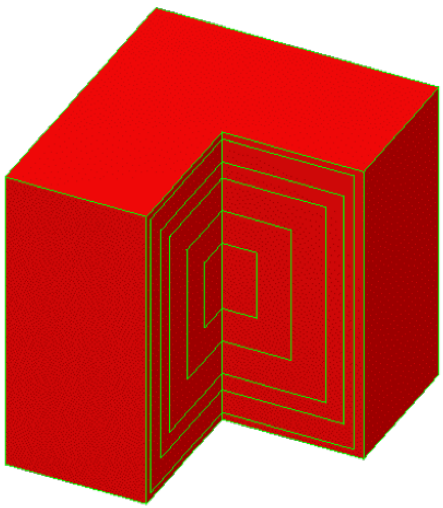


Fig. 6.2.8: Cutaway view of LEU-COMP-THERM-033 case 45 with manual subdivision.

Example 6.2.4: TSUNAMI-3D-K5 input for LEU-COMP-THERM-033 case 45 with 8 cm automated mesh.

```
=tsunami-3d-k5
uf4 paraffin mixture u2f4-6
v6-238
read composition
 h-poly      6 0 0.060586 300   end
 f           6 0 0.012302 300   end
 c           6 0 0.029128 300   end
 u-235       6 0 6.2282e-05 300   end
 u-238       6 0 0.0030126 300   end
 u-234       6 0 6.2548e-07 300   end
end composition
read parameter
 gen=10000
 npg=10000
 sig=0.0001
 mfx=yes
 msh=8
end parameter
read geometry
global unit 1
 cuboid 6 1   40.725  -40.725    43.35   -43.35    44.11   -44.11
end geometry
end data
read sams
 prtimp prtgeom
end sams
end
```

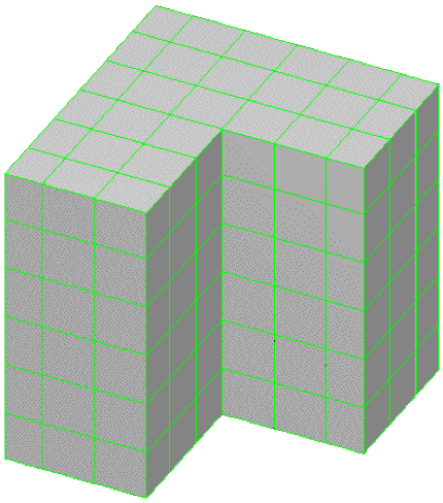


Fig. 6.2.9: TSUNAMI-3D-K5 input for LEU-COMP-THERM-033 case 45 with 8 cm automated mesh.

6.2.5.3 Complex sample problem

A more complex sample problem is a critical assembly of 4.31 wt%-enriched UO_2 fuel rods with a pitch of 2.54 cm in clusters that are separated by copper plates. This system is identified as LEU-COMP-THERM-009 case 10 from the IHECSBE. This system is used for sample problem TSUNAMI-3D_K5-2 that is distributed with SCALE. The TSUNAMI-3D_K5-2 sample problem input is shown in Example 6.2.5, and the geometry is illustrated in Fig. 6.2.10. The KENO V.a input section for this model is essentially the same as that presented in the IHECSBE except that the parameter data have been modified for the TSUNAMI calculation. Also, in the IHECSBE model, all water was assigned to the same mixture. For this model, the water in the reflector region is entered as a separate mixture from the water in the pin cell to generate separate sensitivity coefficients in these regions.

For this system, direct perturbation results were obtained for ^{235}U, ^{238}U, and ^{1}H in the pin cell model (mixture 2) and for ^{1}H in the reflector (mixture 7). Direct perturbation sensitivity coefficients can be difficult to obtain for large systems using Monte Carlo techniques. For a small change in a number density (1% or 2%), the effect of k_{eff} may not be significant enough to be observed outside of statistical uncertainties. If a larger perturbation (5% or 10%) is used, the effect on k_{eff} may not be linear and may produce misleading results. Also, if the magnitude of the difference in k_{eff} between the baseline value and the increased density perturbed model is different from the magnitude of the difference in k_{eff} between the baseline model and the decreased density perturbed model, the response may be nonlinear. In these calculations, the amount of the perturbation was carefully chosen to produce approximately 10 standard deviations of change in k_{eff} between the perturbed and unperturbed case. The amount of perturbation for each nuclide is shown in Table 6.2.12.

TSUNAMI-3D-K5 was executed with no flux mesh, a 15 cm flux mesh, and a 5 cm flux mesh using the default coordinate transform setting (*tfm=yes*). Also, the 15 cm and 5 cm mesh cases were run with the coordinate transform turned off (*tfm=no*).

Where a region is repeatedly used in KENO, the flux data are averaged over all occurrences of a region. In this model, only one fuel pin is explicitly modeled. This single pin is used repeatedly in arrays to create the system model. The flux data for this single pin are averaged over all occurrences throughout the system to create the sensitivity data. Using the mesh flux option, the flux is accumulated for the fuel pin in each mesh interval. For the model with no mesh, the flux in the UO_2 portion of the fuel pin is averaged over a single region. With the 15 cm mesh, the flux in UO_2 is accumulated in 126 separate mesh intervals distributed across the core. For the 5 cm mesh, the flux in UO_2 is accumulated in 2280 mesh intervals. For all regions of the model with no mesh, the flux is only stored for 17 unique locations, one for each region. For the 15 cm mesh, the flux is stored in 1968 unique mesh intervals. For the 5 cm mesh mode, the flux is stored in 34,298 unique mesh intervals.

Example 6.2.5: TSUNAMI-3D LEU-COMP-THERM-009 case 10 sample problem input.

```
=tsunami-3d-k5
tsunami-3d sample 2
v7-238
read composition
 u-234       1 0 5.1835e-06 295   end
 u-235       1 0 0.0010102 295   end
 u-236       1 0 5.1395e-06 295   end
 u-238       1 0 0.022157 295   end
 o           1 0 0.046753 295   end
 h           2 0 0.066675 295   end
 o           2 0 0.033338 295   end
 al          3 0 0.058433 295   end
 cr          3 0 6.231e-05 295   end
```

(continues on next page)

(continued from previous page)

```
 cu          3 0 6.3731e-05 295   end
 mg          3 0 0.00066651 295   end
 mn          3 0 2.2115e-05 295   end
 ti          3 0 2.5375e-05 295   end
 cu          3 0 3.0967e-05 295   end
 si          3 0 0.00034607 295   end
 fe          3 0 0.00010152 295   end
 c           4 0 0.043562 295   end
 h           4 0 0.058178 295   end
 ca          4 0 0.002566 295   end
 s           4 0 0.0004782 295   end
 si          4 0 9.636e-05 295   end
 o           4 0 0.012461 295   end
 h           5 0 0.056642 295   end
 c           5 0 0.035648 295   end
 o           5 0 0.014273 295   end
 c           6 0 0.0015194 295   end
 cu          6 0 0.084128 295   end
 fe          6 0 3.8444e-06 295   end
 mg          6 0 4.4168e-06 295   end
 na          6 0 4.6695e-06 295   end
 o           6 0 0.00010064 295   end
 si          6 0 3.8223e-05 295   end
 s           6 0 3.3474e-06 295   end
 h           7 0 0.066675 295   end
 o           7 0 0.033338 295   end
end composition
read celldata
  latticecell squarepitch fuelr=0.6325 1 gapr=0.6415 0 cladr=0.7075 3 hpitch=1.27 2 end
  inf 4 end
  inf 5 end
  inf 6 end
  inf 7 end
end celldata
read parameter
 npg=10000
 gen=10000
 sig=0.0002
 tfm=yes
 msh=15
end parameter
read geometry
unit 1
com='fuel pin '
 zcylinder 1 1  0.6325   92.075        0
 zcylinder 0 1  0.6415   92.075        0
 zcylinder 4 1  0.6415  94.2975  -2.2225
 zcylinder 3 1  0.7075  94.2975  -2.2225
 cuboid 2 1     1.27    -1.27     1.27    -1.27  94.2975  -2.2225
```

Figure 6.2.14. TSUNAMI-3D LEU-COMP-THERM-009 case 10 sample problem input.

```
unit 2
com='array of fuel pins '
  array 1       0        0  -2.2225
 replicate 5 1       0        0        0        0        0     2.54 1
 replicate 7 1       0        0     7.64     7.64        0        0 1
unit 3
com='water between clusters 7.422 cm '
 cuboid 7 1    7.422        0    27.96    -7.64  94.2975  -2.2225
 cuboid 5 1    7.422        0    27.96    -7.64  94.2975  -4.7625
unit 4
com='cu poison plate between clusters,0.646 cm wide '
 cuboid 6 1    0.646        0    27.96    -7.64     91.5        0
 cuboid 7 1    0.646        0    27.96    -7.64  94.2975  -2.2225
```

(continues on next page)

(continued from previous page)

```
 cuboid 5 1    0.646        0    27.96    -7.64  94.2975  -4.7625
unit 5
com='water between clusters,0.084cm wide '
 cuboid 7 1    0.084        0    27.96    -7.64  94.2975  -2.2225
 cuboid 5 1    0.084        0    27.96    -7.64  94.2975  -4.7625
global unit 6
com='clusters with water between '
  array 2       0        0  -2.2225
 replicate 7 1    30.5     30.5    22.86    22.86  12.7775     15.3 1
end geometry
read array
ara=1 nux=15 nuy=8 nuz=1
 fill
    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
 end fill
ara=2 nux=9 nuy=1 nuz=1
 fill
    2    3    4    5    2    5    4    3    2
 end fill
end array
end data
read sams
  prtgeom
  prtimp
end sams
end
```

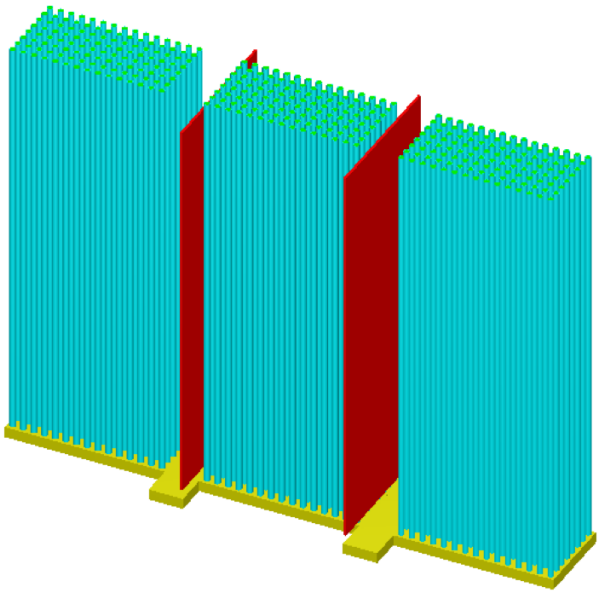


Fig. 6.2.10: Graphical representation of KENO V.a geometry for LEU-COMP-THERM-009 case 10 sample problem.

Sensitivity coefficients and their statistical uncertainties are shown in Table 6.2.12. For ^{235}U, the TSUNAMI-

3D and direct perturbation results agree within one standard deviation for all cases. For ^{238}U, TSUNAMI-3D agrees within two standard deviations for all cases except the no-mesh case, which disagrees by more than four standard deviations. Similarly for ^{1}H in the pin cell, all meshed cases agree with the direct perturbation results to less than one standard deviation, but the case with no geometrical subdivision disagrees by five standard deviations. For ^{1}H in the reflector, again, the meshed cases all agree well with the direct perturbation results, but the non-mesh case shows large discrepancy of 21 standard deviations.

Also, only small differences are observed with and without the transform where an adequate flux mesh is used. Because of the computational resources required for computing the transform while performing neutron tracking, setting *tfm=no* can decrease runtime by ~50%. However, in cases where an adequate computational mesh is not used, setting *tfm=no* can lead to erroneous results.

The energy-dependent sensitivity profiles for ^{1}H elastic scattering for the three TSUNAMI-3D models are shown in Fig. 6.2.11. The differences in the groupwise sensitivity coefficients are most pronounced in the intermediate- to high-energy regions.

Table 6.2.12: Energy- and region-integrated total sensitivity coefficients for LEU-COMP-THERM-009 case 10.

KEYWORD	Format: READ VOLUME enter volume data here END VOLUME The default type for volume calculation type is NONE. See section Volume Data.			
	TYPE	REQUIRED DATA	OPTIONAL DATA	DEFAULT
TYPE=	NONE	none	none	
	RANDOM	none	BATCHES=	500
			POINTS=	1000
			XP=	global unit boundary
			XM=	global unit boundary
			YP=	global unit boundary
			YM=	global unit boundary
			ZP=	global unit boundary
			ZM=	global unit boundary
			SAMPLE_DEN=	points sampled per cubic cm
	TRACE	none	NRAYS=	100,000
			XP=	global unit boundary
			XM=	global unit boundary
			YP=	global unit boundary
			YM=	global unit boundary
			ZP=	global unit boundary
			ZM=	global unit boundary
			IFACE=	smallest face
READVOL=		File name	none	

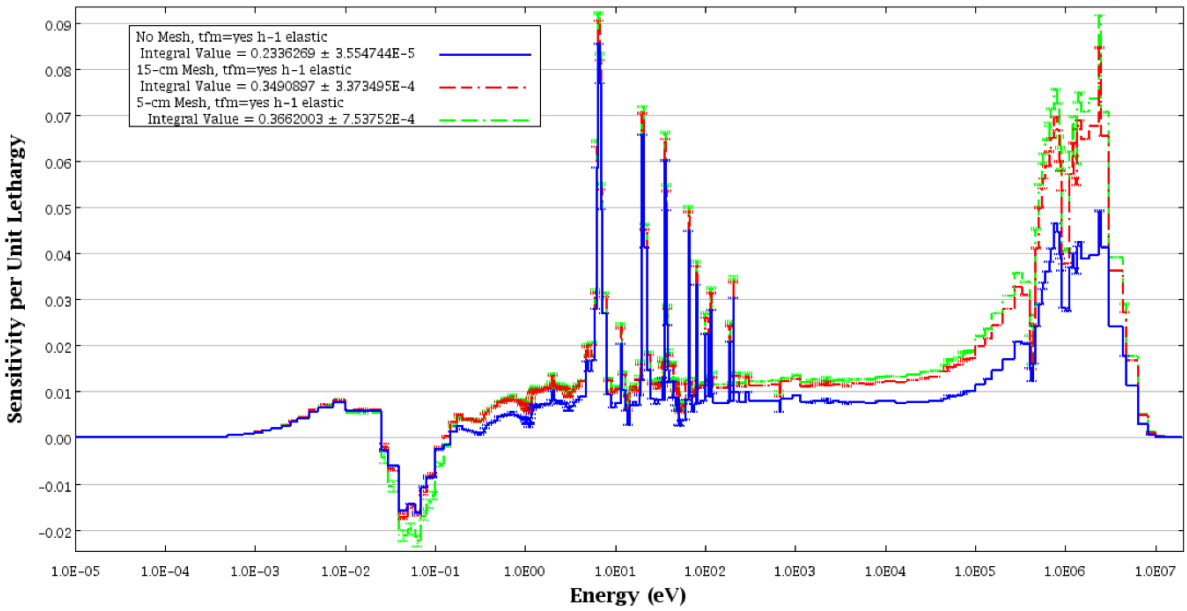


Fig. 6.2.11: Sensitivity profiles from TSUNAMI-3D for ^{1}H elastic scattering in mixture 2 of LEU-COMP-THERM-009 case 10 sample problem.

Loosely coupled sample problem

Another sample problem demonstrates the use of the *GRID* input block to input a nonuniform mesh. This sample problem consists of three tanks containing plutonium nitrate solution that are suspended in air inside of a large concrete room. This system is identified as PU-SOL-THERM-014 case 19 in the IHECSBE. A TSUNAMI-3D-K5 input for this system is shown in Example 6.2.6. The system geometry is illustrated in Fig. 6.2.12, where the concrete room is shown in light blue and the solution tanks are shown in dark blue. A single tank is shown in Fig. 6.2.13, where the tank wall is shown in dark blue and the plutonium nitrate solution is shown in gray.

The KENO V.a option to compute the matrix k_{eff} by hole was used (*mkh=yes)* to give the self-multiplication of each tank. This analysis revealed a system k_{eff} of 1.0049 ± 0.0003 for the forward case and 1.002 ± 0.002 for the adjoint case. The k_{eff} of a single tank for the forward solution was found to be 1.0044 ± 0.0017. Thus the interaction between the tanks has a limited effect on the system multiplication factor.

Direct perturbation sensitivity coefficients were generated for ^{239}Pu, ^{240}Pu, and ^{1}H in the solution using number density perturbations to produce a 20 standard deviation change in k_{eff}, where the requested k_{eff} convergence for the CSAS direct perturbation calculation was 10^{-4}. These values are shown in Table 6.2.13 along with the corresponding TSUNAMI-3D results. The results generated by the model shown in Example 6.2.6, which has no geometric subdivision, agree well with the direct perturbation results for ^{239}Pu, but the ^{1}H values differ by more than six standard deviations, indicating poor agreement.

Example 6.2.6: TSUNAMI-3D-K5 input for sample problem PU-SOL-THERM-014 case 19.

```
=tsunami-3d-k5
PU-SOL-THERM-014-019
v7-238
```

(continues on next page)

(continued from previous page)

```
read comp
PU-238 1 0 2.0965E-8 300.0 END
PU-239 1 0 2.7672E-4 300.0 END
PU-240 1 0 1.2209E-5 300.0 END
PU-241 1 0 9.0317E-7 300.0 END
PU-242 1 0 4.5817E-8 300.0 END
AM-241 1 0 1.1013E-7 300.0 END
N      1 0 2.3837E-3 300.0 END
O      1 0 3.7011E-2 300.0 END
H      1 0 6.0930E-2 300.0 END
FE     1 0 2.5125E-6 300.0 END
CA     1 0 1.3839E-6 300.0 END
CR     1 0 6.6711E-7 300.0 END
NI     1 0 5.3151E-7 300.0 END
FE     2 0 5.8686E-2 300.0 END
CR     2 0 1.6469E-2 300.0 END
NI     2 0 8.1061E-3 300.0 END
MN     2 0 1.7319E-3 300.0 END
SI     2 0 1.6939E-3 300.0 END
C      2 0 1.5857E-4 300.0 END
P      2 0 6.1439E-5 300.0 END
S      2 0 4.4518E-5 300.0 END
H      3 0 1.0350E-2 300.0 END
B-10   3 0 1.6020E-6 300.0 END
O      3 0 4.3470E-2 300.0 END
AL     3 0 1.5630E-3 300.0 END
SI     3 0 1.4170E-2 300.0 END
CA     3 0 6.4240E-3 300.0 END
FE     3 0 7.6210E-4 300.0 END
END COMP
READ PARA
TME=500.0 GEN=15000 NPG=5000  NSK=5 TBA=10.0
RUN=YES AMX=NO FLX=NO FDN=NO FAR=NO PLT=NO sig=0.0001
END PARA
READ GEOM
UNIT 1
COM='CYLINDER + SOLUTION'
CYLINDER 1 1 14.7 -10.7555 -50.5855
CYLINDER 0 1 14.7 50.5855 -50.5855
CYLINDER 2 1 15.0 51.7855 -51.9145
GLOBAL
UNIT 2
COM='CONCRETE BUILDING + ARRAY'
CUBOID 0 1 2P605.0 2P440.0 2P500.0
HOLE 1   -310.0      55.0        -344.0855
HOLE 1   -160.0   -95.0        -344.0855
HOLE 1   -310.0   -95.0        -344.0855
CUBOID 3 1 2P750.0 2P585.0 570.0 -540.0
END GEOM
READ STAR
NST=1
END STAR
END DATA
end
```

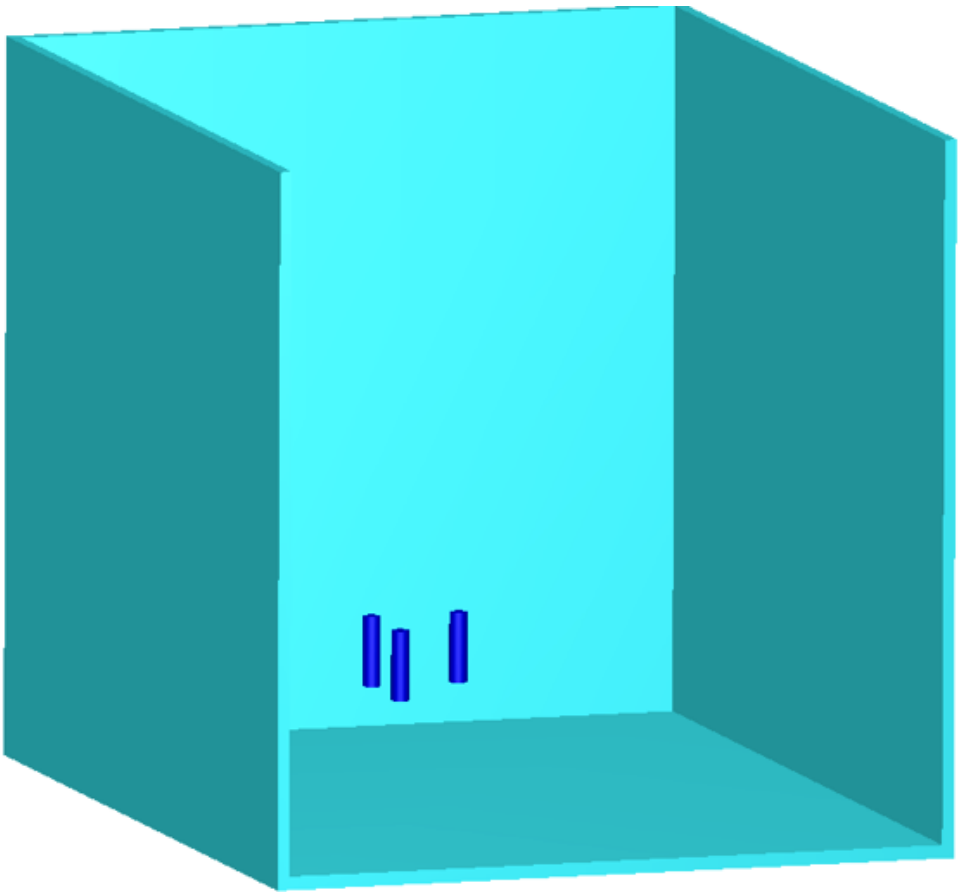


Fig. 6.2.12: Cutaway view of PU-SOL-THERM-014 case 19 sample problem.

Fig. 6.2.13: Cutaway view of single solution tank from PU-SOL-THERM-014 case 19 sample problem.

Table 6.2.13: Energy- and region-integrated total sensitivity coefficients for PU-SOL-THERM-014 case 19.

	^{1}H	^{239}Pu
Percent change in number density for direct perturbation	0.32	2.39
Direct perturbation	5.99E-01 ± 2.56E-02	7.57E-02 ± 8.29E-03
TSUNAMI-3D-K5 base model	4.34E-01 ± 1.82E-05	7.58E-02 ± 4.42E-06
TSUNAMI-3D-K5 grid geometry model	6.05E-01 ± 3.42E-04	7.84E-02 ± 3.80E-05

Because of the large size of the room included in this model relative to the size of the fuel tanks, use of the uniform mesh flux proved to be impractical due to excessive memory requirements. The directional dependence of the flux needs to be computed separately for each tank through the use of adequate spatial refinement with the flux mesh through the use of the grid geometry input.

The *GRID* block in KENO V.a and KENO-VI enables the use of a nonuniform mesh for this problem that is coarse throughout the room to save memory, yet refined within the fueled regions to obtain accurate results. As with previous examples, the use of a sufficiently fine mesh provides accurate results without the use of the transform (*tfm=no*). The TSUNAMI-3D-K5 input listing for this model using the user-defined *GRID* is shown in Example 6.2.7, where the transform is disabled with *tfm=no*, the mesh fluxes are requested with *mfx=yes*, and the grid is defined in the *read grid* block. The *msh* parameter that defines a uniform mesh is not used. The input requirements for the grid are described in the READ GRID section of the KENO documentation and will not be repeated here. It is required that the user-defined mesh completely encloses the geometry of the system. In the model in Example 6.2.11, the grid extends past the geometry by 1 cm in each direction to ensure that potential round-off errors do not cause the grid to appear to terminate inside the geometry at any point.

When the user-defined grid is used without the transform, the TSUNAMI-3D-K5 and direct perturbation sensitivities agree within one standard deviation for both ^{1}H and ^{239}Pu, as shown in Table 6.2.13.

Example 6.2.7: TSUNAMI-3D-K5 input with *GRID* input for sample problem PU SOL-THERM-014 case 19.

```
=tsunami-3d-k5
PU-SOL-THERM-014-019
v7-238
read comp
PU-238 1 0 2.0965E-8 300.0 END
PU-239 1 0 2.7672E-4 300.0 END
PU-240 1 0 1.2209E-5 300.0 END
PU-241 1 0 9.0317E-7 300.0 END
PU-242 1 0 4.5817E-8 300.0 END
AM-241 1 0 1.1013E-7 300.0 END
N      1 0 2.3837E-3 300.0 END
O      1 0 3.7011E-2 300.0 END
H      1 0 6.0930E-2 300.0 END
FE     1 0 2.5125E-6 300.0 END
CA     1 0 1.3839E-6 300.0 END
CR     1 0 6.6711E-7 300.0 END
NI     1 0 5.3151E-7 300.0 END
FE     2 0 5.8686E-2 300.0 END
CR     2 0 1.6469E-2 300.0 END
NI     2 0 8.1061E-3 300.0 END
MN     2 0 1.7319E-3 300.0 END
SI     2 0 1.6939E-3 300.0 END
C      2 0 1.5857E-4 300.0 END
P      2 0 6.1439E-5 300.0 END
S      2 0 4.4518E-5 300.0 END
```

(continues on next page)

(continued from previous page)

```
H       3 0 1.0350E-2 300.0 END
B-10    3 0 1.6020E-6 300.0 END
O       3 0 4.3470E-2 300.0 END
AL      3 0 1.5630E-3 300.0 END
SI      3 0 1.4170E-2 300.0 END
CA      3 0 6.4240E-3 300.0 END
FE      3 0 7.6210E-4 300.0 END
END COMP

READ PARA
TME=500.0 GEN=10000 NPG=10000 SIG=0.0005 ASG=0.001 NSK=5 TBA=10.0
RUN=YES AMX=NO FLX=NO FDN=NO FAR=NO PLT=NO
tfm=no  mfx=yes
END PARA
READ GEOM
UNIT 1
COM='CYLINDER + SOLUTION'
CYLINDER 1 1 14.7 -10.7555 -50.5855
CYLINDER 0 1 14.7 50.5855 -50.5855
CYLINDER 2 1 15.0 51.7855 -51.9145
GLOBAL
UNIT 4
COM='CONCRETE BUILDING + ARRAY'
CUBOID 0 1 -140 -330 75 -115 -290 -405
HOLE 1   -310.0      55.0        -344.0855
HOLE 1   -160.0   -95.0        -344.0855
HOLE 1   -310.0   -95.0        -344.0855
CUBOID 3 1 2P750.0 2P585.0 570.0 -540.0
END GEOM
read grid
1
xplanes=-751, -500, -400, -350,
-331, -311, -291, -271, -261, -241, -221, -201, -180, -160, -139,
-325, -322, -319, -316, -313, -310, -307, -304, -301, -298, -295, -292,
-175, -172, -169, -166, -163, -160, -157, -154, -151, -148, -145, -142
-90, 0, 100, 300, 500, 751 end
yplanes=-586, -400, -200, -150,
-116, -96, -76, -56, -36, -16, 4, 24, 44, 64, 84,
40, 43, 46, 49, 52, 55, 58, 61, 64, 67, 70,
-110, -107, -104, -101, -98, -95, -92, -89, -86, -83, -80, -77,
-25, 100, 300, 586 end
zplanes=-541, -450, -406, -386, -366, -346, -326, -306, -286,
-345, -342, -339, -336, -333, -330, -327, -324, -321, -319, -316, -313, -310, -307, -304,
-250, -100, 100, 300, 571 end
end grid
END DATA
read sams prtimp prtgeom end sams
end
```

6.2.5.4 CE TSUNAMI Sample Problem

The last sample problem focuses on performing CE TSUNAMI sensitivity calculations starting from a CE KENO-VI model. The model used in this example is a highly enriched uranium sphere (Godiva) using one-eighth symmetry; this case is identified as HEU-MET-FAST-001 in IHECSBE [3d11]. CE TSUNAMI inputs are in many ways easier to prepare than MG TSUNAMI inputs because CE TSUNAMI does not require flux moment calculations, a flux mesh (except for CLUTCH $F^*(r)$ calculations), implicit sensitivity calculations, or an adjoint transport calculation, and thus requires much less user input. Example 6.2.8 shows the CE TSUNAMI input for this system. The input lines that differ from those in the CE KENO-VI input are highlighted in blue. This simulation uses the IFP sensitivity method (cet=2) and assumes that daughter fission neutrons diffuse through the system and produce an asymptotic population after five latent generations (cfp=5). In general, IFP simulations require a cfp between 5 and 10 generations to produce accurate sensitivity estimates, although IFP sensitivity calculations can produce accurate sensitivities for this

system using as few as 2 latent generations. The runtime and memory requirements of the IFP method are proportional to cfp, and users should minimize cfp when performing IFP sensitivity calculations to maximize calculation efficiency and minimize calculation memory footprint.

Example 6.2.8: CE TSUNAMI-3D-K6 input for the Godiva system using the IFP method.

```
=tsunami-3d-k6
Godiva sample problem
ce_v7_endf
read composition
 u-234       1 0 0.000491995 300   end
 u-235       1 0 0.0449996 300   end
 u-238       1 0 0.002498 300   end
end composition
read parameter
 gen=1200
 npg=10000
 nsk=200
 htm=no
 cet=2
 cfp=5
end parameter
read geometry
global unit 1
com="global unit 1"
 sphere 1    8.741  chord +x=0  chord +y=0  chord +z=0
 cuboid 2    8.741        0    8.741        0    8.741        0
 media 1 1 1
 media 0 1 -1 2
 boundary 2
end geometry
read bnds
  body=2
    surface(1)=vacuum
    surface(2)=mirror
    surface(3)=vacuum
    surface(4)=mirror
    surface(5)=vacuum
    surface(6)=mirror
  end bnds
end data
end
```

Example 6.2.9 shows the CE TSUNAMI input for the Godiva system using the CLUTCH sensitivity method (cet=1). This input is different from the IFP input in that the CLUTCH method requires a spatial grid for tallying the CLUTCH $F^*(r)$ function. This grid is specified in the "read gridGeometry" block and the ID of the grid specified in this block is passed to CLUTCH in the parameter block (cgd=11). In general, CLUTCH calculations are much faster than IFP calculations and produce a significantly smaller memory footprint, but CLUTCH calculations must have an accurate $F^*(r)$ mesh to obtain accurate sensitivity coefficients. The $F^*(r)$ mesh is currently calculated during inactive generations using the IFP method, and the cfp=5 card specifies how many latent generations are used in this calculation. Using more latent generations will increase the accuracy of the $F^*(r)$ mesh and will not significantly affect the memory footprint of the CLUTCH method, but it will increase the variance of the $F^*(r)$ mesh values. In general an $F^*(r)$ mesh with 1 cm to 2 cm mesh intervals is sufficiently refined to obtain accurate sensitivity coefficients. $F^*(r)$ is calculated during the inactive generations, and in general users should simulate at least on the order of 10 to 100 inactive histories per mesh interval to allow for sufficient $F^*(r)$ convergence. This sometimes necessitates simulating a large number of additional inactive histories/generations. Fission source convergence is not necessary to begin accurate $F^*(r)$ estimates. Therefore, $F^*(r)$ can be tallied during the inactive generations while the fission source is still converging.

Example 6.2.9: CE TSUNAMI-3D-K6 input for the Godiva system using the CLUTCH method.

```
=tsunami-3d-k6
Godiva sample problem
ce_v7_endf
read composition
 u-234       1 0 0.000491995 300   end
 u-235       1 0 0.0449996 300   end
 u-238       1 0 0.002498 300   end
end composition
read parameter
 gen=1200
 npg=10000
 nsk=200
 htm=no
 cet=1
 cfp=5
 cgd=11
end parameter
read geometry
global unit 1
com="global unit 1"
 sphere 1    8.741  chord +x=0  chord +y=0  chord +z=0
 cuboid 2    8.741        0    8.741        0    8.741        0
 media 1 1 1
 media 0 1 -1 2
 boundary 2
end geometry
read bnds
  body=2
    surface(1)=vacuum
    surface(2)=mirror
    surface(3)=vacuum
    surface(4)=mirror
    surface(5)=vacuum
    surface(6)=mirror
  end bnds
read gridGeometry 11
    title="Mesh for collecting fission source distribution"
    xLinear   10  -0.01 8.741
    yLinear   10  -0.01 8.741
    zLinear   10  -0.01 8.741
end gridGeometry
end data
end
```

It is possible to perform a CE TSUNAMI CLUTCH sensitivity calculation using the assumption that $F^*(r)$ equals one everywhere. This will not produce accurate sensitivity coefficient estimates for systems where the importance of fission neutrons varies significantly as a function of space, but it will produce accurate sensitivity coefficients for systems with relatively flat importance functions, such as an infinitely reflected model of a single fuel pin or an infinite media problem. This calculation mode requires no additional inactive generations for the calculation of $F^*(r)$ and is useful for estimating the runtime or memory requirements of a CLUTCH calculation and for obtaining rough estimates of the sensitivity coefficients of complicated systems. Example 6.2.10 shows the CE TSUNAMI input for the Godiva system using the CLUTCH sensitivity method (cet=1). The $F^*(r)$ mesh calculation is disabled using cfp=-1, making the "read gridGeometry" and "cgd=###" inputs not required.

Example 6.2.10: CE TSUNAMI-3D-K6 input for the Godiva system using the CLUTCH method with no F*(r) mesh.

```
=tsunami-3d-k6
Godiva sample problem
ce_v7_endf
read composition
 u-234       1 0 0.000491995 300   end
 u-235       1 0 0.0449996 300   end
 u-238       1 0 0.002498 300   end
end composition
read parameter
 gen=1200
 npg=10000
 nsk=200
 htm=no
 cet=1
 cfp=-1
end parameter
read geometry
global unit 1
com="global unit 1"
 sphere 1    8.741  chord +x=0  chord +y=0  chord +z=0
 cuboid 2    8.741        0    8.741        0    8.741        0
 media 1 1 1
 media 0 1 -1 2
 boundary 2
end geometry
read bnds
  body=2
    surface(1)=vacuum
    surface(2)=mirror
    surface(3)=vacuum
    surface(4)=mirror
    surface(5)=vacuum
    surface(6)=mirror
  end bnds
end data
end
```

The three previously described CE TSUNAMI inputs were simulated. The total nuclide sensitivity coefficients from these runs are compared with reference direct perturbation and MG TSUNAMI-3D sensitivities in Table 6.2.14. The difference between the calculated and reference sensitivities, in terms of the effective number of standard deviations, is given in the table in parentheses below each calculated sensitivity coefficient. Although the reference direct perturbation sensitivity estimates had uncertainties that were a bit large (it is recommended that their relative uncertainties are less than 5%), the four TSUNAMI calculations all produced sensitivity coefficients that seem to agree well with the reference sensitivities.

Table 6.2.14: CE TSUNAMI Godiva sensitivity coefficient comparison.

Nuclide	Direct Perturbation	MG TSUNAMI-3D	IFP	CLUTCH	CLUTCH (no mesh)
^{234}U	0.00646 ± 0.00050	0.00727 ± 0.000005 (1.62 σ_{eff})	0.00725 ± 0.00030 (1.35 σ_{eff})	0.00738 ± 0.00006 (1.83 σ_{eff})	0.00708 ± 0.00006 (1.23 σ_{eff})
^{235}U	0.80959 ± 0.06937	0.79648 ± 0.00048 (-0.19 σ_{eff})	0.80378 ± 0.00289 (-0.08 σ_{eff})	0.80359 ± 0.00062 (-0.09 σ_{eff})	0.77705 ± 0.00059 (-0.47 σ_{eff})
^{238}U	0.01896 ± 0.00178	0.01768 ± 0.00002 (-0.72 σ_{eff})	0.01816 ± 0.00064 (-0.42 σ_{eff})	0.01792 ± 0.00013 (-0.58 σ_{eff})	0.01633 ± 0.00012 (-1.47 σ_{eff})

Two sample inputs have been included to illustrate how a user can run GEAR-MC sensitivity calculations starting from the CE KENO-VI Godiva model. The first, CLUTCH-only GPT input requires specifying a number of latent generations for the GPT $F^*(r)$ calculation (cfp=4), creating a gridGeom for the $F^*(r)$ mesh, and specifying reaction rates, materials, and energy ranges for the numerator and denominator response terms. The generalized response examined in this problem is the ^{235}U fast (> 0.6 MeV and < 20 MeV) fission cross section; in other words:

$$R = \frac{\left\langle \Sigma_f^{U-235} \phi \right\rangle^{\text{fast}}}{\langle \phi \rangle^{\text{fast}}}. \tag{6.2.28}$$

Therefore, the reaction rate in the numerator term of this response is the ^{235}U fast fission reaction rate in material 1 (`nnc=92235`, `nma=1`, `nmt=18`, `nmx=20e7`, and `nmn=600000`) and the denominator term is the fast flux in material 1 (`dnc=-1`, `dma=1`, `dmt=0`, `dmx=20e7`, and `dmn=600000`).

Example 6.2.11: CE TSUNAMI-3D-K6 input for the Godiva system using GEAR-MC with only the CLUTCH method.

```
=tsunami-3d-k6
Godiva sample problem
ce_v7_endf
read composition
 u-234       1 0 0.000491995 300    end
 u-235       1 0 0.0449996 300    end
 u-238       1 0 0.002498 300    end
end composition
read parameter
 gen=1200
 npg=10000
 nsk=200
 htm=no
 cet=4
 cfp=5
```

(continues on next page)

(continued from previous page)

```
 cgd=11
 nnc=92235
 nma=1
 nmt=18
 dnc=-1
 dma=1
 dmt=0
 nmx=20e7
 nmn=600000
 dmx=20e7
 dmn=600000
end parameter
read geometry
global unit 1
com="global unit 1"
 sphere 1    8.741  chord +x=0  chord +y=0  chord +z=0
 cuboid 2    8.741        0    8.741        0    8.741        0
 media 1 1 1
 media 0 1 -1 2
 boundary 2
end geometry
read bnds
  body=2
    surface(1)=vacuum
    surface(2)=mirror
    surface(3)=vacuum
    surface(4)=mirror
    surface(5)=vacuum
    surface(6)=mirror
  end bnds
read gridGeometry 11
    title="Mesh for collecting fission source distribution"
    xLinear   10  -0.01 8.741
    yLinear   10  -0.01 8.741
    zLinear   10  -0.01 8.741
end gridGeometry
end data
end
```

This problem can be specified using the input option `cet=5`, which uses the IFP method to estimate the intergenerational effect term on the fly rather than tallying the effect in an $F^*(r)$ mesh. For this cet=5 case the Definitions and SystemResponses blocks (found in the SystemResponses block sections in the TSUNAMI-1D chapter of the SCALE manual) are used to specify the GPT responses rather than using KENO input parameters.

Example 6.2.12: CE TSUNAMI-3D-K6 input for the Godiva system using GEAR-MC with the CLUTCH and IFP methods.

```
=tsunami-3d-k6
Godiva sample problem
ce_v7_endf
read composition
 u-234       1 0 0.000491995 300    end
 u-235       1 0 0.0449996 300    end
 u-238       1 0 0.002498 300    end
end composition
read parameter
 gen=1200
 npg=10000
 nsk=200
 htm=no
 cet=5
```

(continues on next page)

(continued from previous page)

```
 cfp=5
end parameter
read definitions
  response 10
    mixture=1
    nuclide=92235
    reaction=fission
    macro
    ehigh=20e7
    elow=6e5
  end response
  response 30
    mixture=1
    unity
    ehigh=20e7
    elow=6e5
  end response
end definitions
read systemResponses
  ratio 5
    title="u-235 fast xs"
    numer 10  end
    denom 30  end
  end ratio
end systemResponses
read geometry
global unit 1
com="global unit 1"
 sphere 1    8.741  chord +x=0  chord +y=0  chord +z=0
 cuboid 2    8.741        0    8.741        0    8.741        0
 media 1 1 1
 media 0 1 -1 2
 boundary 2
end geometry
read bnds
  body=2
    surface(1)=vacuum
    surface(2)=mirror
    surface(3)=vacuum
    surface(4)=mirror
    surface(5)=vacuum
    surface(6)=mirror
  end bnds
end data
end
```

REFERENCES

6.3 TSUNAMI UTILITY MODULES

B. T. Rearden, M. A. Jessee, and J. D. McDonnell

ABSTRACT

Several modules have been developed to assist with the sensitivity and uncertainty analysis techniques included in SCALE. TSUNAMI-IP (**T**ools for **S**ensitivity and **U**ncertainty **A**nalysis **M**ethodology **I**mplementation-**I**ndices and **P**arameters) uses sensitivity data generated by TSUNAMI-1D, -2D, or -3D to propagate the effect of uncertainty in nuclear data to a response of interest, and to generate several relational parameters and indices that predict the degree of similarity between two systems. Additionally, the formats of files used by SCALE for sensitivity analyses are described herein.

ACKNOWLEDGMENTS

The author wishes to acknowledge C. M. Hopper, B. L. Broadhead of the Oak Ridge National Laboratory (ORNL), S. Goluoglu, and R. L. Childs, formerly of the ORNL, and J. J. Wagschal of The Hebrew University of Jerusalem, for their assistance with the development of TSUNAMI-IP and L. M. Petrie, I. C. Gauld, and J. E. Horwedel of the Oak Ridge National Laboratory for their assistance in the development of SENLIB and BONAMIST.

6.3.1 TSUNAMI-IP

TSUNAMI-IP (**T**ools for **S**ensitivity and **U**ncertainty **A**nalysis **M**ethodology **I**mplementation-**I**ndices and **P**arameters) uses sensitivity data generated by TSUNAMI-1D, -2D, or -3D and cross section covariance data described in the SCALE Nuclear Data Covariance Library chapter to generate several relational parameters and indices that can be used to determine the degree of similarity between two systems. TSUNAMI-IP combines many techniques developed over several years into a comprehensive package with simplified input. Most of the techniques developed for TSUNAMI-IP are documented in Ref. [refTsip1], other techniques are documented in Refs. [refTsip2], [refTsip3] and [refTsip4], and others are originally presented in this document.

Depending on the user selected options, voluminous output can be generated by TSUNAMI-IP. To assist with review of these data, TSUNAMI-IP was the first SCALE module to offer HTML formatted output. This document first describes the indices and parameters that are computed, presents a description of the user input, then finally a sample problem is presented where the standard text output and the HTML formatted output are described.

TSUNAMI-IP was introduced in SCALE 5.0 and improved in SCALE 5.1. In SCALE 6.0, TSUNAMI-IP now includes an individual c_k to better quantify uncertainty-based similarity on a nuclide-reaction specific level. The E_{sum} parameter has since been replaced with a sensitivity-only parameter, simply called E, which is based on more rigorous mathematical definition. The input format has been expanded with additional keywords to identify sensitivity data and a *RESPONSE* data block that allows users to define applications and experiments within the same input block. Uncertainty and penalty edits can also be presented with relative or absolute values, where previously only relative values were available. The input of user-defined covariance data now allows much greater flexibility, and the covariance data used in the problem can now be exported to a COVERX formatted data file for subsequent plotting in Fulcrum.

6.3.1.1 Global integral indices

Three global integral indices that assess the similarity of a particular application and a single experiment on a system-wide basis for all nuclides and reactions are defined in this section. The integral indices are c_k, E, and G. Each of these indices is defined in subsequent subsections, and TSUNAMI-IP input options used to generate these indices and produce specific output edits are explained. Each integral index is normalized such that a value of 1.0 indicates complete similarity between the application and the experiment and a value of 0.0 indicates no similarity.

Integral index c_k

A rigorous approach to assessing the similarity of two systems for purposes of criticality code validation is the use of uncertainty analysis, which propagates the tabulated cross section uncertainty information to the calculated k_{eff} value of a given system via the energy-dependent sensitivity coefficients.X1X Mathematically, the system uncertainty is computed with a quadratic product of the group-wise sensitivity profile vectors by nuclide and reaction type with the group-wise cross section uncertainty matrices by nuclide and reaction type. The result of this procedure is not only an estimate of the uncertainty in the system k_{eff} due to cross sections, but also an estimate of the correlated uncertainty between systems. These correlated uncertainties

can be represented by correlation coefficients, which represent the degree of correlation in the uncertainties between the two systems. This correlation coefficient index, denoted as c_k, not only has the desirability of a single quantity relating the two systems, but also measures the similarity of the systems in terms of related uncertainty. These correlation coefficients are particularly useful when used in traditional trending analyses for criticality safety validation in that the correlation coefficient relates the degree in which the uncertainties in the critical benchmarks are coupled with the uncertainties in the application of interest. This coupling with the common uncertainties in the various systems is expected to closely mimic the coupling in predicted biases between the various systems, since they should both be related to the cross section uncertainties.

The cross section covariance data are read from a COVERX formatted data file identified by *coverx*= in the *PARAMETER* input. The cross section covariance data files distributed with SCALE are discussed in the SCALE Nuclear Data Covariance Library chapter, and the format of the COVERX data file is presented in COVERX format. A prerequisite in the uncertainty analysis approach is that cross section uncertainty data for all nuclides and reactions of interest have been evaluated and processed for use by these procedures. However, evaluated cross section uncertainty data are not available for all nuclide-reaction pairs. Nuclide-reaction pairs without data are omitted from this analysis, but it is assumed that either the cross section data values from these pairs are well known (i.e., small uncertainties), or that the sensitivity of the system k_{eff} to these nuclide-reaction pairs is small. Where these assumptions hold, the nuclide-reaction pairs without cross section-uncertainty data present a negligible contribution to the uncertainty-based analysis. For situations where this negligible contribution assumption is judged to be invalid, the use of uncertainty analysis is not appropriate. However, the *COVARIANCE* data block can be used to input uncertainty values for the cross section data for particular nuclide-reaction pairs to assess the impact of additional covariance data. To utilize the covariance data generated by user input in the *COVARIANCE* data block, the keyword *use_icov* must be entered in the *PARAMETER* data block. Additionally, default uncertainty values can be assigned for all unknown covariance data. This default uncertainty data is input in the *PARAMETER* data block and the keyword *use_dcov* must be entered to activate its use. Warning messages are printed to identify substituted covariance matrices.

When *use_dcov* and/or *use_icov* and *cov_fix* are specified in the *PARAMETER* data block, and a reaction has zero or large (standard deviation > 1000%) values on the diagonal of the covariance matrix, these values are replaced with the square of the user input or default standard deviation values, and the corresponding off-diagonal terms are substituted according to the user input or default correlation values. Warning messages are printed to identify which values were replaced and which standard deviation value was used in the replacement.

The mathematical development of the integral index c_{k} is presented here based on the development given in [refTsip1]. The nuclear data parameters (i.e., group wise nuclide-reaction specific cross sections) are represented by the vector $\alpha \equiv (\alpha_m), m = 1, 2, \ldots, M$, where M is the number of nuclide-reaction pairs × the number of energy groups. The corresponding symmetric $M \times M$ matrix containing the relative variances (diagonal elements) and covariances (off-diagonal elements) in the nuclear data are

$$\mathbf{C}_{\alpha\alpha} \equiv \left[\frac{\text{COV}(\alpha_m, \alpha_p)}{\alpha_m \alpha_p}\right], m = 1, 2, \ldots, M; p = 1, 2, \ldots, M, \tag{6.3.1}$$

where

$$\text{COV}(\alpha_m, \alpha_p) = \left\langle \delta\alpha_m \delta\alpha_p \right\rangle \tag{6.3.2}$$

where $\delta\alpha_m$ and $\delta\alpha_p$ represent the difference between the values and expectation values of the nuclear data parameters and represents integration over the ranges of α_m and α_p weighted with a probability density function. A rigorous definition of the cross section covariance data are given in [refTsip5].

The matrix containing sensitivities of the calculated k_{eff} to the α parameters is given as

$$\mathbf{S_k} \equiv \left[\frac{\alpha_m}{k_i} \frac{\partial k_i}{\partial \alpha_m} \right], i = 1, 2, \ldots, I; m = 1, 2, \ldots, M, \tag{6.3.3}$$

where I is the number of critical systems being considered. In TSUNAMI-IP, the elements of S_k are the sensitivity coefficients read from the sensitivity data files identified in the *APPLICATIONS* and *EXPERIMENTS* data blocks. The uncertainty matrix for the system k_{eff} values, C_{kk}, is given as

$$\mathbf{C_{kk}} = \mathbf{S_k} \mathbf{C}_{\alpha\alpha} \mathbf{S}_\mathbf{k}^\dagger \tag{6.3.4}$$

where † indicates a transpose.

S_k is an I × M matrix; $C_{\alpha\alpha}$ is an M × M matrix; and the resulting C_{kk} matrix is of dimension I × I. The C_{kk} matrix consists of relative variance values, σ_i^2, for each of the systems under consideration (the diagonal elements), as well as the relative covariance between systems, σ_{ij}^2 (the off-diagonal elements). These off-diagonal elements represent the shared or common variance between two systems. The off diagonal elements are typically divided by the square root of the corresponding diagonal elements (i.e., the respective standard deviations) to generate a correlation coefficient matrix. Thus, the correlation coefficient is defined as

$$c_k = \frac{\sigma_{ij}^2}{\left(\sigma_i \sigma_j\right)} \tag{6.3.5}$$

such that the single c_k value represents the correlation coefficient between uncertainties in system i and system j.

These correlations are primarily due to the fact that the uncertainties in the calculated k_{eff} values for two different systems are related, since they contain the same materials. Cross section uncertainties propagate to all systems containing these materials. Systems with the same materials and similar spectra would be correlated, while systems with different materials or spectra would not be correlated. The interpretation of the correlation coefficient is the following: a value of 0.0 represents no correlation between the systems, a value of 1.0 represents full correlation between the systems, and a value of -1.0 represents a full anti-correlation.

To request the computation of c_k for each application identified in the *APPLICATIONS* section of the input in relation to each experiment in the *EXPERIMENTS* portion of the input, simply enter the input *c* in the *PARAMETER* section of the input. Additionally, the *values* table must be requested to output the full listing of c_k values. The *csummary* edit prints a listing of all experiments that have a c_k value exceeding the criteria value set by *cvalue*=.

Integral index $\mathbf{c}_r$

The integral index c_k is intended for investigative analysis and is defined as c_k with user-defined reactions removed from consideration. The *EXCLUSIONS* data block is used to identify which reactions will be excluded from consideration in the calculation of c_k. If the user identifies no reactions, then c_r will compute the same value as c_k. Using c_r is equivalent to removing all sensitivities for a given reaction or series of reactions for all nuclides from the sensitivity data file for all applications and all experiments considered in the analysis.

To request the computation of c_r for each application identified in the *APPLICATIONS* section of the input in relation to each experiment in the *EXPERIMENTS* portion of the input, simply enter the input *cr* in the *PARAMETER* section of the input. Additionally, the *values* table must be requested to output the full listing of c_r values. The *crsummary* edit prints a listing of all experiments that have a c_r value exceeding the criteria value set by *crvalue*=.

***Integral index* E**

The E index is a replacement for the previous E_{sum} index, which was present in TSUNAMI-IP for SCALE 5.0 and 5.1.1. If the group-wise sensitivity data for all nuclides and reactions for each system is thought of as a vector, then the integral index E is the cosine of the angle between the two sensitivity vectors for the analyzed systems. If the two sensitivity vectors are parallel, i.e., proportional, the systems are similar. E does not require cross section covariance data and is normalized such that an E value of 0.0 indicates the systems are totally dissimilar, and an E value of 1.0 indicates the two systems are the same. Mathematically, an E value as low as -1.0 could be generated, but this would be the result of a rare combination of system sensitivity coefficients such that the sensitivity of the respective system responses would have to be exactly proportional in magnitude and opposite in sign, which seems not to be physically feasible. The E parameter is considered global in nature because its single quantity assesses similarity between two systems based on the magnitude and shape of all sensitivity profiles. The vector $\mathbf{S_i}$ is defined as the sensitivity *vector* (not matrix) for a particular application or experiment "i." The magnitude of the sensitivity vector corresponds to the L2 norm: $|\mathbf{S_i}| = \sqrt{\mathbf{S_i^T S_i}}$. The E value for a given application a with experiment e is then

$$E \equiv \frac{\mathbf{S_a^T S_e}}{|\mathbf{S_a}|\,|\mathbf{S_e}|} \tag{6.3.6}$$

The similarity of systems in terms of their sensitivities to only the fission, capture or scatter reactions can also be evaluated as

$$E_x \equiv \frac{\mathbf{S_{x,a}^T S_{x,e}}}{\left|\mathbf{S_{x,a}}\right|\left|\mathbf{S_{x,e}}\right|}, \tag{6.3.7}$$

where the vectors $S_{x,a}$ and $S_{x,e}$ represent the sensitivity of application and experiment to fission, capture, or scatter reaction x.

To request the computation of E for each application identified in the *APPLICATIONS* section of the input in relation to each experiment in the *EXPERIMENTS* portion of the input, simply enter the input e in the *PARAMETER* section of the input. Additionally, the *values* table must be requested to output the full listing of E values. The reaction components of E (i.e., E_{f}, E_{c} and E_{s}) are included in the *values* table if *prtparts* is entered in the input. The reaction-specific components of E are each normalized between -1 and 1, the same as E itself. The *esummary* edit prints a listing of all experiments that have an E value exceeding the criteria value set by *evalue*=.

***Integral index* G**

The G index assesses the similarity of two systems based on normalized differences in the energy-dependent sensitivity data for fission, capture, and scatter [refTsip2]. The similarity measure used for G is based on the concept of coverage of the application by an experiment. A physical interpretation of the G index is the ratio of the sum of the sensitivity coefficients of the application that are covered by the experiment to the sum of the sensitivity coefficients for the application. The G index, sometimes referred to as "big G" is defined as:

$$G = 1 - \frac{\sum_n \sum_x \sum_j \left(S_{x,j}^{a,n} - S_{x,j}^{e',n}\right)}{\sum_n \sum_x \sum_j S_{x,j}^{a,n}} \tag{6.3.8}$$

where,

$$S_{x,j}^{e',n} = \begin{cases} S_{x,j}^{e,n}, \text{ where} & \left|S_{x,j}^{a,n}\right| \geq \left|S_{x,j}^{e,n}\right| \text{ and } \frac{S_{x,j}^{a,n}}{\left|S_{x,j}^{a,n}\right|} = \frac{S_{x,j}^{e,n}}{\left|S_{x,j}^{e,n}\right|} \\ S_{x,j}^{a,n}, \text{ where} & \left|S_{x,j}^{a,n}\right| < \left|S_{x,j}^{e,n}\right| \text{ and } \frac{S_{x,j}^{a,n}}{\left|S_{x,j}^{a,n}\right|} = \frac{S_{x,j}^{e,n}}{\left|S_{x,j}^{e,n}\right|} \\ 0, & \text{otherwise} \end{cases} \tag{6.3.9}$$

the n summation is performed over all nuclides present in the application system,

the x summation is performed over fission, capture and scatter reactions (f, c, and s) as appropriate for each nuclide, and

the j summation is performed over all energy groups.

The use of 1 minus the normalized difference makes the range of this index consistent with c_k and E_{sum}. Hence, a G of 1 indicates complete similarity and a G value of 0 indicates no similarity.

The definition of $S_{x,j}^{e',n}$ restricts the coverage of the application by the experiment to the portion of the experiment's sensitivity coefficient that does not exceed that of the application in magnitude. Additionally, the application's sensitivity coefficient and that of the experiment must have the same sign. The coverage for ^{1}H scatter for an example application and experiment is illustrated in Fig. 6.3.1 where the energy-dependent sensitivity profiles for the application, the experiment and the coverage of the application by the experiment are shown in green, red and blue, respectively. Because the sensitivity coefficients for the application and the experiment have opposite signs at energies just above 1×10^{-2} eV, the application provides no coverage for the experiment for these groups. Also, for several groups, the sensitivity of the application exceeds that of the experiment and only partial coverage is provided. Partial coverage is illustrated where the application data (green) exceeds the coverage (blue). In other groups, the sensitivity of the experiment exceeds that of the application. In these groups, full coverage is provided, but the coverage is not allowed to exceed the sensitivity of the application. This is illustrated in groups where the experiment data (red) exceeds the coverage data (blue). With the limitation of the coverage as the value of the applications sensitivity coefficient, the experiment cannot provide "extra credit" in coverage for sensitivity coefficients that exceed those of the application.

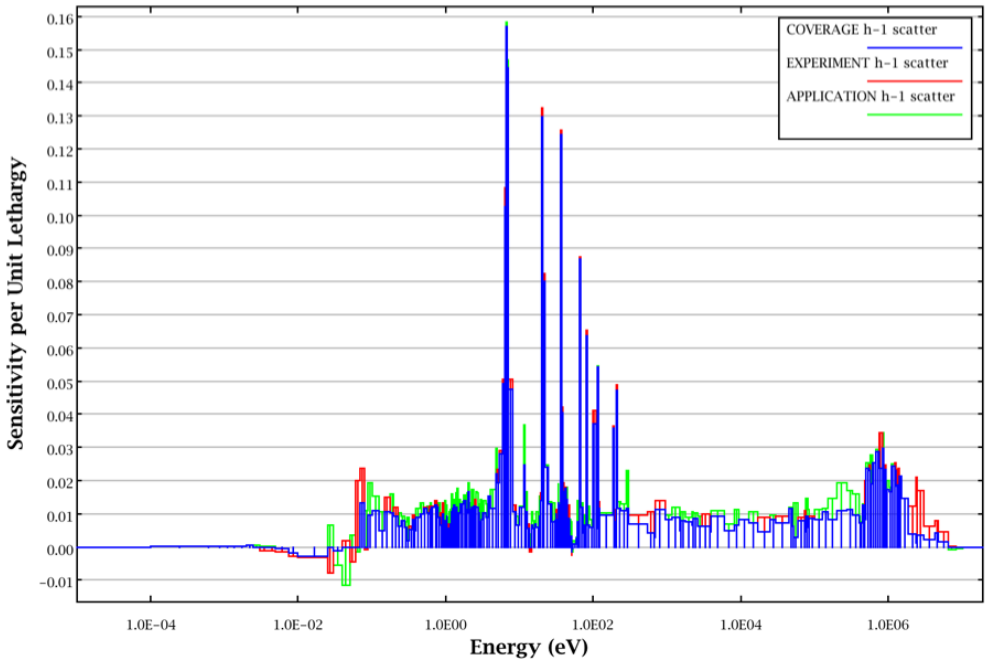


Fig. 6.3.1: Illustration of coverage for ^{1}H scatter.

The coverage for ^{10}B capture for an example application and experiment is illustrated in Fig. 6.3.2. In this figure, the sensitivity coefficients are all negative. The magnitudes of the sensitivity coefficients for the experiment far exceed those of the application at thermal energies. However, coverage is only provided to the magnitudes of the sensitivity coefficients of the application.

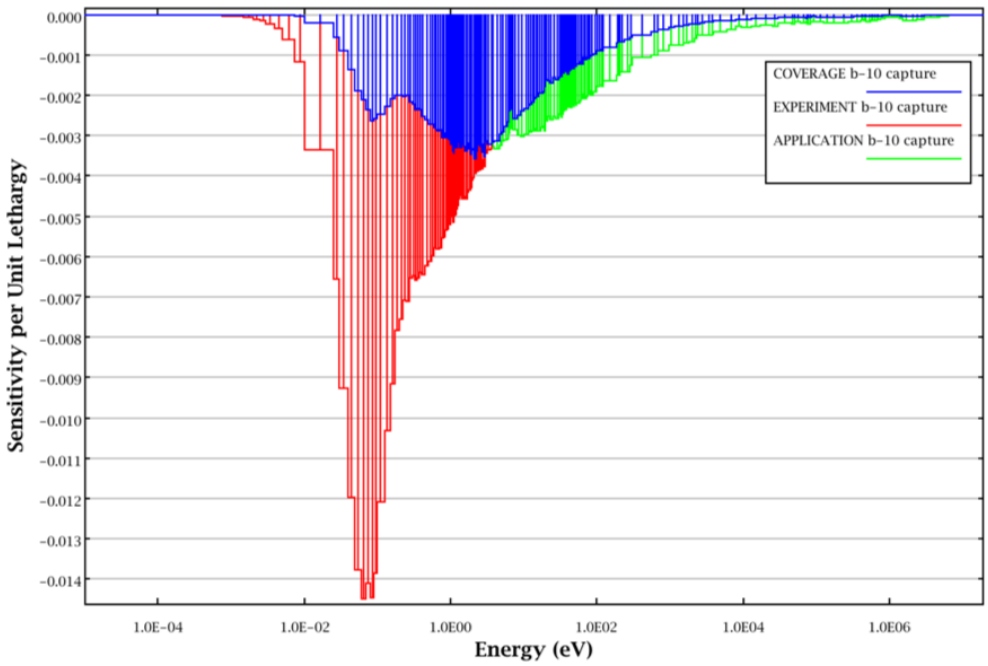


Fig. 6.3.2: Illustration of coverage for ^{10}B capture.

Assessment of similarity over a particular reaction type (fission, capture or scatter) can be made with a partial G value. A G_x value can be computed by eliminating the sum over reactions, x in Eq. (6.3.11), as

$$G_x = 1 - \frac{\sum_n \sum_j \left(S_{x,j}^{a,n} - S_{x,j}^{e',n} \right)}{\sum_n \sum_j S_{x,j}^{a,n}} \tag{6.3.10}$$

where

x is f, c, or s.

To request the computation of G for each application identified in the *APPLICATIONS* section of the input in relation to each experiment in the *EXPERIMENTS* portion of the input, simply enter the input g in the *PARAMETER* section of the input. Additionally, the *values* table must be requested to output the full listing of G values. The reaction components of G (i.e., G_f, G_c and G_s) are included in the *values* table if *prtparts* is entered in the input. The G_x indices are normalized such they also have a range of 0 to 1, where a G_x of 1 indicates complete similarity of sensitivity coefficients for that particular reaction type x between the application and experiment. The *gsummary* edit prints a listing of all experiments that have a G value exceeding the criteria value set by *gvalue*=.

6.3.1.2 Nuclide-reaction specific integral indices

The global integral indices described in Sect. 6.3.1.1 assess system similarity for all nuclides and reactions in the application system. It is also possible and sometimes desirable to produce values analogous to G, E, and c_k for each nuclide-reaction pair, such that similarity can be assessed on a nuclide-reaction-specific level.

***Nuclide-reaction specific integral index* g**

The nuclide-reaction specific integral index based on the same coverage criteria as G is denoted g, and sometimes referred to as "little g." [2] It is defined in terms of the normalized differences of the group wise sensitivity coefficients for a particular nuclide, n, and reaction, x, summed over all energy groups, j, as

$$g_x^n = 1 - \frac{\sum_j \left(S_{x,j}^{a,n} - S_{x,j}^{e',n} \right)}{\sum_j S_{x,j}^{a,n}} \tag{6.3.11}$$

where,

$$S_{x,j}^{e',n} = \begin{cases} S_{x,j}^{e,n}, \text{ where} & \left| S_{x,j}^{a,n} \right| \geq \left| S_{x,j}^{e,n} \right| \text{ and } \frac{S_{x,j}^{a,n}}{\left| S_{x,j}^{a,n} \right|} = \frac{S_{x,j}^{e,n}}{\left| S_{x,j}^{e,n} \right|} \\ S_{x,j}^{a,n}, \text{ where} & \left| S_{x,j}^{a,n} \right| < \left| S_{x,j}^{e,n} \right| \text{ and } \frac{S_{x,j}^{a,n}}{\left| S_{x,j}^{a,n} \right|} = \frac{S_{x,j}^{e,n}}{\left| S_{x,j}^{e,n} \right|} \\ 0, & \text{otherwise} \end{cases} \tag{6.3.12}$$

and the j summation is performed over all energy groups.

The criteria for g are the same as those explained in Sect. 6.3.1.1.3. The only difference in g and G is that the summations over nuclide and reaction have been removed. As with other integral indices, the g index is normalized such that a g value of 1 indicates complete coverage of the application by the experiment for the particular nuclide-reaction pair. A g value of 0 indicates no coverage of the application by the experiment for the particular nuclide-reaction pair.

To request the computation of g ("little g") for each application identified in the APPLICATIONS section of the input in relation to each experiment in the EXPERIMENTS portion of the input for fission, capture and scatter reactions for each nuclide in the application, simply enter the input *lg* in the PARAMETER section of the input. The *lg* data appears in the nuclide-reaction specific edit for each application and only includes nuclide-reaction pairs with energy integrated sensitivity coefficients with at least the magnitude of *sencut*. Additionally, g values are printed for each nuclide-reaction pair specified in the REACTIONS data block. The *lgall* edit prints all the g values for fission, capture and scatter for all nuclides in the application in relation to all experiments regardless of the magnitude of the energy-integrated sensitivity value. The *lgsum* edit prints a listing of all g values exceeding the criteria value set by *lgvalue*= for fission, capture and scatter reactions for each nuclide. The *lggroups* edit prints a long table listing the numbers of experiments that are at least as sensitive as the application for each group for each nuclide-reaction pair requested in the REACTIONS data block.

Extended c_k

The extended c_k edit lists the contribution of each nuclide-reaction-to-nuclide-reaction energy covariance matrix to the global integral index c_k. A single entry in this edit is computed by utilizing a subset of the $\mathbf{C}_{\alpha\alpha}$ matrix and the $\mathbf{S_k}$ sensitivity vectors for the application and the experiment. Thus, the extended c_k for a nuclide-reaction pair in the application in relation to a nuclide-reaction pair in the experiment is defined as

$$c^{contribution}_{k,(n-j)_a,(m-k)_e} = \frac{\sigma^2_{(n-j)_a,(m-k)_e}}{(\sigma_a \sigma_e)} \tag{6.3.13}$$

where,

> $(n\text{-}j)_a$ represents the nuclide n and reaction j of the application a, $(m\text{-}k)_e$ represents the nuclide m and reaction k of the experiment e, $\sigma^2_{(n-j)_a,(m-k)_e}$ represents the covariance between application a and experiment e due to this nuclide-reaction pairs, σ_a is the standard deviation in keff for the application due to all cross section covariance data, and σ_e is the standard deviation in keff for the experiment due to all cross section covariance data.

The global integral index c_k can be reconstructed from the extended c_k components by summing the extended c_k over all nuclide-reaction-to-nuclide-reaction values. Thus, the extended c_k allows the user to determine the contribution of nuclide-reaction-to-nuclide-reaction covariances to c_k. All covariance data, default covariance data and user-input covariance data used in the calculation of c_k, as outlined in Sect. 6.3.1.1, are also applied to the extended c_k data. Nuclide-reaction-to-nuclide-reaction covariances using default and user-input values are identified with one and two asterisks, respectively, in the text output, and are identified with unique colors in the HTML output. Nuclide-reaction-to-nuclide-reaction covariances using default and user-input values for *cov_fix* adjustments are identified with three and four asterisks, respectively, in the text output, and are identified with unique colors in the HTML output. Warning messages are printed to identify which zero values were replaced and which standard deviation value was used in the substitution.

The extended c_k edit also includes an individual c_k to examine the similarity of an application and experiment based only on a single nuclide-reaction pair. The individual c_k is similar to the c_k contribution from Eq. (6.3.13), except that it is normalized between -1 and 1 for each nuclide-reaction pair as

$$c^{individual}_{k,(n-j)_a,(m-k)_e} = \frac{\sigma^2_{(n-j)_a,(m-k)_e}}{\left(\sigma_{(n-j)_a} \sigma_{(m-k)_e}\right)} \tag{6.3.14}$$

Note that individual c_k values computed for a nuclide-reaction pair in an application to the same nuclide-reaction pair in the experiment. Although cross-reaction and cross-nuclide covariance data are available, the cross-relationship has no physical interpretation for assessing for assessing the similarity of systems for a specific nuclide-reaction pair.

To request the computation of extended c_k for each application identified in the APPLICATIONS section of the input in relation to each experiment in the EXPERIMENTS portion of the input, simply enter the input *c_long* in the PARAMETER section of the input.

Extended $\mathbf{c}_r$

Similar to the extended c_k edit, the extended c_r edit lists the contribution of each nuclide-reaction-to-nuclide-reaction energy covariance matrix to the global integral index c_k. An individual c_r evaluating the similarity of each application to each experiment in terms of only one nuclide-reaction pair, normalized from -1 to 1, is also available. The computation of the extended c_r is the same as that of extended c_k, but with user-defined reactions removed from consideration. The *EXCLUSIONS* data block is used to identify which reactions will be excluded from consideration in the calculation of c_r and its components as presented in extended c_r. If the user identifies no reactions, then c_r and extended c_r will be computed the same as c_k and extended c_k. The c_r parameter and extended c_r edit are equivalent to removing all sensitivities for a given reaction or series of reactions from the sensitivity data file for all applications and all experiments considered in the analysis.

As with extended c_k, nuclide-reaction-to-nuclide-reaction covariances using default and user-input values are identified with one and two asterisks, respectively, in the text output, and are identified with unique colors in the HTML output. Nuclide-reaction-to-nuclide-reaction covariances using default and user-input values for *cov_fix* adjustments are identified with three and four asterisks, respectively, in the text output, and are identified with unique colors in the HTML output. Warning messages are printed to identify which values were replaced and which standard deviation value was used in the substitution.

To request the computation of extended c_r for each application identified in the *APPLICATIONS* section of the input in relation to each experiment in the *EXPERIMENTS* portion of the input, simply enter the input *cr_long* in the *PARAMETER* section of the input.

6.3.1.3 Penalty assessment

A method is available to assess an additional margin to subcriticality, or penalty, where sufficient experiments are not available to provide complete coverage for a particular application [refTsip3]. This penalty is intended as an additional uncertainty component that can be added to the calculated value of k_{eff} to provide an added measure of safety for application systems where validation coverage is lacking. The penalty calculation is based on the criteria for coverage explained in Sect. 6.3.1.1.3. The criteria for coverage in this implementation of the penalty assessment is that if a single experiment that passes qualification tests for the particular application exhibits a sensitivity coefficient for a particular energy group for a particular nuclide-reaction pair that is at least as great in magnitude and has the same sign as the corresponding sensitivity coefficient for the application, then adequate coverage exists for the code validation of the application. For group-wise nuclide-reaction specific sensitivity coefficients for the application that are not fully covered by the experiments, the uncovered portion of the sensitivity coefficient is used to compute an uncertainty in k_{eff} though the cross section covariance data.

Any experiment used in the penalty assessment calculation must pass a qualification test to determine global similarity of the experiment, based on a global integral index (c_k, E, or G), and may also have to pass a nuclide-reaction specific qualification test based on the g integral index. Thus, only experiments that exhibit a certain degree of similarity to the application can be considered in this calculation, unless the tests are deactivated at the user's discretion. Additionally, a number of similar experiments may be required before any penalty assessment is produced.

To compute the penalty, a vector of the minimum differences in the sensitivity coefficients, $\mathbf{Z_a}$, for the application with respect to all experiments can be obtained as

$$\mathbf{Z_a} \equiv \left[Z_{x,j}^{a,n}\right] n = 1, \ldots, N, x = 1, \ldots, X, j = 1, .., J \tag{6.3.15}$$

where,

$Z^{a,n}_{x,j} = S^{a,n}_{x,j} - C^{a,n}_{x,j}$,

$C^{a,n}_{x,j}$ is a composite of the best available sensitivity data from all experiments and is defined as

$C^{a,n}_{x,j} = S^{e',n}_{x,j}$ for the experiment that satisfies min $\left|S^{a,n}_{x,j} - S^{e',n}_{x,j}\right|$, e′ =1,. . . ,E,

$$S^{e',n}_{x,j} = \begin{cases} S^{e,n}_{x,j}, \text{ where} & \left|S^{a,n}_{x,j}\right| \geq \left|S^{e,n}_{x,j}\right| \text{ and } \frac{S^{a,n}_{x,j}}{\left|S^{a,n}_{x,j}\right|} = \frac{S^{e,n}_{x,j}}{\left|S^{e,n}_{x,j}\right|} \\ S^{a,n}_{x,j}, \text{ where} & \left|S^{a,n}_{x,j}\right| < \left|S^{e,n}_{x,j}\right| \text{ and } \frac{S^{a,n}_{x,j}}{\left|S^{a,n}_{x,j}\right|} = \frac{S^{e,n}_{x,j}}{\left|S^{e,n}_{x,j}\right|} \\ 0, & \text{otherwise} \end{cases} \tag{6.3.16}$$

a represents a particular application,

e represents a particular experiment,

n represents a particular nuclide,

x represents a particular reaction,

j represents a particular energy group,

N = number of nuclides in the application system,

X = number of reactions for each nuclide,

J = number of energy groups, and

E = number of experiments meeting the qualification tests.

Once $\mathbf{Z_a}$ is computed, the portion of the sensitivity of the application that is not covered by the experiments can be used to propagate the uncertainty in the cross section data to a relative uncertainty in k_{eff} as

$$\Delta k_{\text{eff}}/k_{\text{eff}} = \sqrt{\mathbf{Z_a}\mathbf{C}_{\alpha\alpha}\mathbf{Z_a}^{\dagger}} \tag{6.3.17}$$

where † indicates a transpose and $\mathbf{C}_{\alpha\alpha}$ is the matrix of the cross section covariance data defined in Eq. (6.3.1). In the above equation, the elements of $Z_{\mathbf{a}}$ are each expressed in terms of $(\Delta k_{\text{eff}}/k_{\text{eff}})/(\Delta\Sigma/\Sigma)$, and the elements of $C_{\alpha\alpha}$ are expressed in terms of relative variances or covariances as $(\Delta\Sigma/\Sigma)^2$, so that the final penalty is expressed as a relative uncertainty in k_{eff} , $\Delta k_{\text{eff}}/k_{\text{eff}}$. This relative uncertainty in k_{eff} is written to the output file in the penalty calculations.

To request the penalty computation for each application identified in the *APPLICATIONS* section of the input in relation to all experiments in the *EXPERIMENTS* section of the input, simply enter the input *penalty* in the *PARAMETER* section of the input. Additionally, a list of the contribution each nuclide-reaction-to-nuclide-reaction covariance matrix to the total penalty can be viewed by entering *penlong* in the *PARAMETER* input. This creates an edit similar to the Uncertainty Information edit of the SAMS module. Each value shown in this output edit represents the relative penalty in percent (% $\Delta k/k$) due to the specified nuclide-reaction-to-nuclide-reaction covariance matrix. The values are sorted in order of descending magnitude. The cumulative penalty can be constructed by squaring the individual values, then adding those that had positive signs and subtracting those that had negative signs, then taking the square root. Negative values in the table result from covariance matrices that have anticorrelated values.

The qualification test for including or excluding experiments from the development of the penalty for a particular application is set by several input parameters. The purpose of the qualification test is to ensure that some relevant data are used in the calculation. Otherwise, in the limit that no relevant data are used in

the penalty calculation, the Z_a minimum difference sensitivity vector becomes S_a, the sensitivity vector for the application. In this case, the maximum penalty assessed is simply the uncertainty in the application's criticality calculation due to cross section covariance data.

A global qualification test, to test the similarity of a particular experiment to the given application based on a global integral index, is configured with the keywords: *penusec*, *penusee* and *penuseg*. These keywords are used to produce a penalty calculation that only includes experiments with c_k, *E,* or *G* values at least as great as *cvalue*, *evalue* or *gvalue*, respectively. The default value is *penusec*. If more than one of the keywords *penusec*, *penusee* or *penuseg* are included in the input, only the last one entered will be used. The penalty is only computed for a given application if the number of experiments passing the global integral index qualification test is at least as great as *penminx* (default = 10). If no global test is desired, enter *penminx=0*.

The *pencut*= keyword allows the user to set a discriminator for excluding nuclide-reaction pairs from the application with small sensitivity coefficients from the penalty calculation. If the sum of the absolute values of the group-wise sensitivity coefficients is below *pencut*, the nuclide reaction pair is excluded from the penalty calculation. The default value is 0.0, which includes all nuclide-reaction pairs in the penalty calculation. Additionally, *penlgv*= sets a discriminator that only includes experiments with a *g* value relative to the application for a given nuclide-reaction pair that meets or exceeds *penlgv*. Thus, using *penlgv* may allow some nuclide-reaction pairs from a given experiment to be included in the penalty calculation, but exclude others from the same experiment that do not meet the criteria. The default value is 0.0, which means include all nuclide-reaction pairs that pass the other qualification tests.

The keyword *penwarn* activates a penalty warning edit that details which experiments were excluded from the penalty calculation based on failing the global qualification test, and which nuclide-reaction pairs were excluded from the penalty calculation based on failing the nuclide-reaction specific tests.

The composite of the best available sensitivity data from all experiments that meet the requested criteria, as used in the calculation of the penalty, can be viewed with the composite sensitivity data for nuclides and reactions requested in the *REACTIONS* input data block with the input keyword *prtcomp* described in Sect. 6.3.1.4.3.

The $\mathbf{C}_{\alpha\alpha}$ matrix in Eq. (6.3.17) is the cross section covariance data read in the COVERX formatted data file identified by *coverx*=. As in the calculation of c_k, nuclide-reaction pairs without available covariance data are omitted from this analysis, but it is assumed that either the cross section data values from these pairs are well known (i.e., small uncertainties), or that the sensitivity of the system k_{eff} to these nuclide-reaction pairs is small. Where these assumptions hold, the nuclide-reaction pairs without cross section uncertainty data present a negligible contribution to the penalty calculation. For situations where this negligible contribution assumption is judged not to be valid, the use of uncertainty analysis for the computation of a penalty is not appropriate. However, the *COVARIANCE* data block can be used to input uncertainty values for the cross section data for particular nuclide-reaction pairs to assess the impact of additional covariance data. To utilize the covariance data generated by user input in the *COVARIANCE* data block, the keyword *use_icov* must be entered in the *PARAMETER* data block. Additionally, default uncertainty values can be assigned for all unknown covariance data. The default uncertainty data input in the *PARAMETER* data block and the keyword *use_dcov* must be entered to activate its use. In the extended penalty edit, nuclide-reaction-to-nuclide-reaction covariances using default and user-input values are identified with one and two asterisks, respectively, in the text output, and are identified with unique colors in the HTML output. Warning messages are also printed to identify substituted covariance matrices.

When *use_dcov* and/or *use_icov* and *cov_fix* are specified in the *PARAMETER* data block, and a reaction has zero or large (standard deviation > 1000%) values on the diagonal of the covariance matrix, these values are replaced with the square of the user input or default standard deviation values, and the corresponding

off-diagonal terms are substituted according to the user input or default correlation values. In the extended penalty edit, nuclide-reaction-to-nuclide-reaction covariances using default and user-input values for *cov_fix* adjustments are identified with three and four asterisks, respectively, in the text output, and are identified with unique colors in the HTML output. Warning messages are printed to identify which values were replaced and which standard deviation value was used in the substitution.

6.3.1.4 Other parameters

TSUNAMI-IP can produce a number of other parameters that are useful for analysis of systems. These are briefly explained in this section.

Uncertainty information

The keyword *uncert* activates the calculation of the uncertainty in k_{eff} due to the cross section covariance data read from the COVERX formatted data file identified by *coverx*=. The uncertainty is computed for each application and each experiment and is printed in the *values* table as a standard deviation value with its statistical uncertainty, if appropriate. The uncertainty value printed is a relative uncertainty in percent (i.e., $\Delta k_{\text{eff}}/k_{\text{eff}} \times 100\%$).

Nuclide-reaction pairs without available data are omitted from this analysis, but it is assumed that either the cross section data values from these pairs are well known (i.e., small uncertainties), or that the sensitivity of the system k_{eff} to these nuclide-reaction pairs is small. Where these assumptions hold, the nuclide-reaction pairs without cross section uncertainty data present a negligible contribution to the uncertainty-based analysis. For situations where this negligible contribution assumption is judged not to be valid, the use of uncertainty analysis is not appropriate. However, the *COVARIANCE* data block can be used to input uncertainty values for the cross section data for particular nuclide-reaction pairs to assess the impact of additional covariance data. To utilize the covariance data generated by user input in the *COVARIANCE* data block, the keyword *use_icov* must be entered in the *PARAMETER* data block. Additionally, default uncertainty data can be assigned for all unknown covariance data. This default data is input in the *PARAMETER* data block and the keyword *use_dcov* must be entered to activate its use.

To request a listing of the contributions nuclide-reaction-to-nuclide-reaction covariance matrix to the uncertainty in the k_{eff} value for each application identified in the *APPLICATIONS* section of the input enter the input keyword *uncert_long* in the *PARAMETER* section of the input. This creates an edit similar to the Uncertainty Information edit of the SAMS module. Each value shown in this output edit represents the relative uncertainty in percent ($\%\Delta k/k$) due to the specified nuclide-reaction-to-nuclide-reaction covariance matrix. The values are sorted in order of descending magnitude. The cumulative uncertainty can be constructed by squaring the individual values, then adding those that had positive signs and subtracting those that had negative signs, then taking the square root. Negative values in the table result from covariance matrices that have anticorrelated values.

In the extended uncertainty edit, nuclide-reaction-to-nuclide-reaction covariances using default and user-input values are identified with one and two asterisks, respectively, in the text output, and are identified with unique colors in the HTML output. Nuclide-reaction-to-nuclide-reaction covariances using default and user-input values for *cov_fix* adjustments are identified with three and four asterisks, respectively, in the text output, and are identified with unique colors in the HTML output. Warning messages are printed to identify which values were replaced and which standard deviation value was used in the substitution.

Completeness parameter

A parameter has been developed to assess the completeness of a set of experiments for the code validation of a given application. [4] The set of experiments is "complete" in the sense that it completely tests all the important cross section elements in the particular application of interest. The availability of sensitivity coefficients provides a key element in the definition of this completeness parameter.

The completeness parameter, R, is defined as follows:

$$R = S_a / S_t, \tag{6.3.18}$$

where

$$\begin{aligned} \mathrm{S_t} &= \sum_{\mathrm{n}} \sum_{\mathrm{x}} \sum_{\mathrm{j}} \left| \mathrm{S}^{\mathrm{a,n}}_{\mathrm{x,j}} \right| \\ S_a &= \sum_{n} \sum_{x} \sum_{j} \left| dS^{a,n}_{x,j} \right| \end{aligned} \tag{6.3.19}$$

and

$$d = \begin{cases} 1, \text{ if } N^n_{x,j} \geq nix\lim \\ 0, \text{ if } N^n_{x,j} < nix\lim \end{cases} \tag{6.3.20}$$

$N^n_{x,j}$ = number of systems for which $S^{e,n}_{x,j} >$ | senfac $\times S^{a,n}_{x,j}$|

e = experiment,

a = application,

$S^{e,n}_{x,j}$ = the sensitivity of k_{eff} of an experiment to the cross sections of the constituent material nuclide n, reaction x, and energy group j,

$S^{a,n}_{x,j}$ = the sensitivity of k_{eff} of the application to the cross sections of the constituent material nuclide n, reaction x, and energy group j,

nixlim = an integer, and

senfac = a real number such that $0.0 \leq senfac \leq 1.0$.

The completeness parameter is designed to give the effective fraction of the total sensitivity for each application system that is "covered" by the benchmark set. This coverage is defined by comparing the magnitude of each group-wise sensitivity coefficient for the application with respect to each of the corresponding sensitivities of the benchmark systems. The completeness parameter is computed for each application for each experiment if the keyword *cp* is entered in the *PARAMETER* data. The minimum coverage of the sensitivity coefficients for the experiment systems is defined as *senfac*$\times$ 100% of the application sensitivity in the definition of $N^n_{x,j}$. The value of *senfac* is set with the keyword input *senfac*=, and the default value is 0.9. Thus, the experiment must have a sensitivity coefficient at least 90% as great as that of the application for the particular nuclide, reaction and energy group to count in $N^n_{x,j}$. The number of experiments counted in $N^n_{x,j}$ must be at least *nixlim* to change d from 0 to 1, which indicates coverage for the particular nuclide, reaction and group of the application. The value of *nixlim* is set by the *nixlim*= keyword, and it has a default value of 10

Composite sensitivity data

A composite of the best available sensitivity data from all experiments included in the analysis, based on the coverage criteria used for the G integral index in Sect. 6.3.1.1.4 is produced for each application for each nuclide-reaction pair in the *REACTIONS* input if the keyword *prtcomp* is entered. The composite profile for a particular application a for nuclide n and reaction x is defined as the vector

$$\mathbf{C}_{\mathbf{x}}^{\mathbf{a,n}} \equiv \left[C_{x,j}^{a,n}\right], j = 1, .., J \tag{6.3.21}$$

where,

$C_{x,j}^{a,n}$ is $S_{x,j}^{e',n}$ for the experiment that satisfies $\min \left|S_{x,j}^{a,n} - S_{x,j}^{e',n}\right| \; e' = 1, \ldots, E$,

$$S_{x,j}^{e',n} = \begin{cases} S_{x,j}^{e,n}, \text{ where} & \left|S_{x,j}^{a,n}\right| \geq \left|S_{x,j}^{e,n}\right| \text{ and } \frac{S_{x,j}^{a,n}}{\left|S_{x,j}^{a,n}\right|} = \frac{S_{x,j}^{e,n}}{\left|S_{x,j}^{e,n}\right|} \\ S_{x,j}^{a,n}, \text{ where} & \left|S_{x,j}^{a,n}\right| < \left|S_{x,j}^{e,n}\right| \text{ and } \frac{S_{x,j}^{a,n}}{\left|S_{x,j}^{a,n}\right|} = \frac{S_{x,j}^{e,n}}{\left|S_{x,j}^{e,n}\right|} \\ 0, & \text{otherwise} \end{cases} \tag{6.3.22}$$

a represents a particular application,

e represents a particular experiment,

n represents a particular nuclide,

x represents a particular reaction,

j represents a particular energy group,

J = number of energy groups, and

E = number of experiments.

An example composite sensitivity profile for ^{1}H total is shown in Fig. 6.3.3. Here, the composite sensitivity profile comprised of the best available data from unidentified experiments 1–5 is shown in black. The sensitivity of the unidentified application is shown in red. The areas where the red application curve exceeds the black composite curve are considered uncovered. Areas where the experiment data exceed the application data are considered fully covered. Note that the composite data does not exceed the application data.

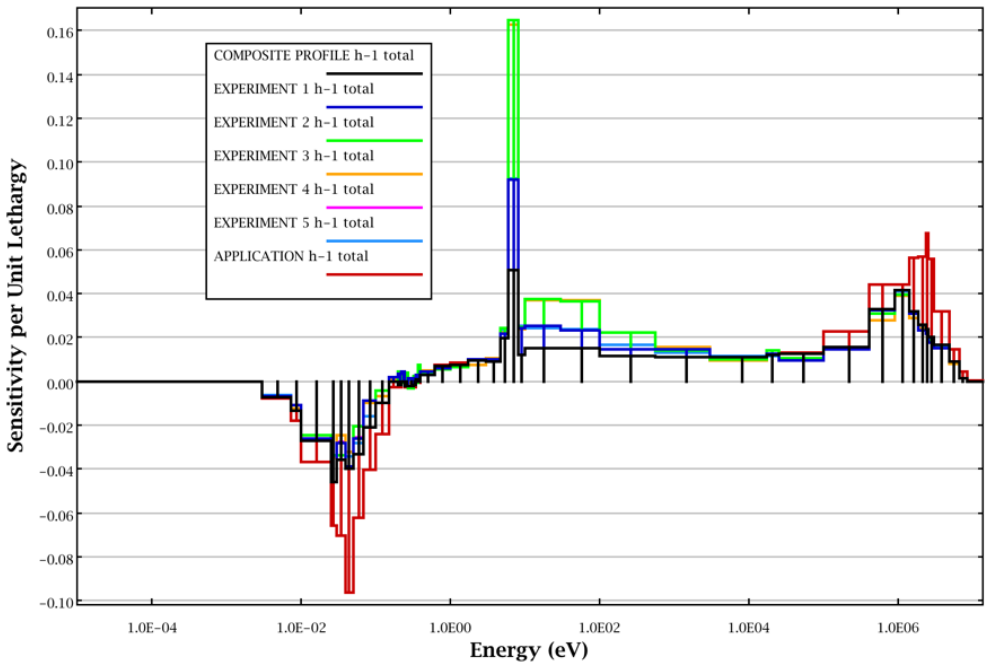


Fig. 6.3.3: Example composite sensitivity profile.

If the composite data for a particular nuclide-reaction pair for a particular energy group are added to the minimum difference data as defined in Eq. (6.3.15), the sensitivity of the application would result as

$$C_{x,j}^{a,n} + Z_{x,j}^{a,n} = S_{x,j}^{a,n} \tag{6.3.23}$$

A sensitivity data file containing the composite sensitivity data is generated to permit further analysis. Only data for those nuclides and reaction identified in the *REACTIONS* input block are included in the composite sensitivity data file. The data file is identified with the TSUNAMI-IP input file name with the extension ".sdf" and is suitable for use with Fulcrum. This data file is presented as an interactive plot in the HTML output. The following data are included on the composite sensitivity data file for each nuclide-reaction pair identified in the *REACTIONS* input block for each application: the sensitivity of the application identified with "Application" and the application number, the composite sensitivity data as defined in Eq. (6.3.21) , and the composite profile used in the definition of the minimum difference profile from the penalty calculation (Sect. 6.3.1.3), which may be reduced from the full composite by excluding experiments not meeting the cutoff criteria for the penalty calculation. The composite data as used in the penalty calculation are identified as "Cut Composite."

Non-coverage

A summary of non-coverage for the nuclide-reaction pairs entered in the *REACTIONS* section of the input can be produced if the keyword *prtnotcv* is entered in the *PARAMETER* data. The non-coverage edit gives the number of groups that are not fully covered according to the *G* criteria and gives the sum of the non-covered portion of the sensitivity coefficients. It also lists the group with largest sensitivity, regardless of whether or not it is covered, and gives the sensitivity value for this group. The experiment that best covers the group with the maximum sensitivity is given and the sensitivity of this group for the best matching experiment is given.

6.3.1.5 Miscellaneous options

Several input parameters that provide a wide range of options in TSUNAMI-IP are explained here. These keywords are entered in the *PARAMETER* data block.

VADER input files

VADER (formerly USLSTATS) is a data regression tool described in Ref. [refTsip6] and in the VADER chapter of the SCALE manual. The *uslstats* keyword requests input files for VADER to be generated from all integral indices computed by TSUNAMI-IP. The *uslsummary* keyword requests VADER input files to be generated from all integral indices exceeding their particular acceptance criteria (*cvalue, evalue,* and *gvalue*). The uncertainties in the k_{eff} values for VADER can be adjusted with two options. The default setting is that the uncertainty for each k_{eff} value included in the VADER input file consists of the square root of the sum of the squares of the Monte Carlo uncertainty and a uniform experimental uncertainty of 0.3%. To quadratically add the uncertainty in k_{eff} due to cross section covariance data to this uncertainty, enter the keyword *usl_uncert*. To modify the uniform experimental uncertainty, use the *usl_sigs*= keyword. When a positive value is entered for *usl_sigs*, this value is treated as the total uncertainty for each experiment, and no other contributions, Monte Carlo or cross section, are considered. When a negative value is entered for *usl_sigs*, this value is treated as the experimental uncertainty for each experiment and Monte Carlo and cross section uncertainties are added as appropriate. Currently, no option exists to input a unique experimental uncertainty for each experiment.

The VADER input files contain the extension `.usl` and the filenames are presented in a descriptive format as *title_xxxx_p_yyyy.usl*, where *title* is the *filename* of the TSUNAMI-IP input case, *xxxx* is the application number in the TSUNAMI-IP input, *p* is the name of the integral index (*ck* for c_k, *e* for *E*, and *g* for *G*), and *yyyyy* is the TSUNAMI-IP execution number (typically 0001). When the summary inputs are requested with *uslsummary*, the filename is of the format *title_xxxx_p_sum_yyyy.usl*, where *sum* denotes that this VADER input file contains only the summary of experiments that exceed the requested criteria for the specific integral index.

Covariance data directory listing

The *prtmtrix* keyword causes a listing of the energy covariances of the COVERX formatted cross section data file, identified by *coverx*=, to be printed.

HTML output

The keyword *html* causes TSUNAMI-IP to create an html formatted output. This output edit is accessed by opening the *root*.html file, where *root* is the name of the user's input file without extensions. Additional resources for the html output are placed in new directories called *root*.htmd and *applet_resources*. The *root*.html directory contains files needed to display data produced by the current case and the *applet_resources* directory contains Java applets for data plotting within the HTML interface and can be shared by multiple output files within the same directory. The *root*.html file and associated directories are placed in the same location as the user's input file. The html formatted output is color coded and more easily navigated than the standard plain text formatted output file. The html output can be customized with the *HTML* data block.

Case sensitive input

The *inptcase* keyword sets a flag in the SCALE free form input reader to prevent SCALE from translating all input to lower case. The purpose of this keyword is the identification of file and directory names that have upper case characters on a case-sensitive operating system. By default, the SCALE input reader translates all text to lower case. However, any input entered after *inptcase* is treated as case sensitive. All subsequent keyword entries and data block names must be entered in lower case, or errors will result.

Print filenames

The *usename* keyword causes TSUNAMI-IP to identify files with their file names in the code output. By default, TSUNAMI-IP identifies files according to the title on the TSUNAMI-A or TSUNAMI-B sensitivity data files. For files that have the same titles, or have long or non-descriptive titles, *usename* can provide for simplified interpretation of the output.

c_k and* E *differences

The keywords *cechk*= and *cediff*= allow the user to input discriminators that govern the output of warning messages when c_k and E values for a particular application and experiment differ by more than *cediff* and c_k and E are both at least as great as *cechk.*

Plots

The keyword *plot* causes Javapeño formatted plot (.plt) files of the global integral indices to be generated. A plot file containing all computed integral index values for each application as a function of experiment number is titled *rootxxxx.plt*, where *root* is the base name of the input file and *xxxx* is the TSUNAMI-IP execution number (typically 0001). Plot file containing the number of integral index values that exceeded the acceptance criteria for each application is titled *rootsummaryxxxx.plt*, where *summary* indicated it is a summary plot. Both plots are displayed in the HTML output or can be view with Fulcrum. If the .plt files are opened in Fulcrum with "Open Dataset…" each set of data can be manually added to the plot as desired by the user. If the .plt files are opened with "Open Plot…," all data are immediately displayed.

Absolute sensitivity option

All global indices and parameters can also be calculated using absolute sensitivities. This capability is useful when working with TSAR-generated reactivity sensitivity data files. TSAR creates a sensitivity data file with reactivity sensitivities tabulated in either relative format or absolute format $(\Delta\rho/|\rho|)/\Delta\sigma/\sigma)$. If keyword *relative* is specified in the PARAMETER block, any absolute reactivity sensitivities are internally converted to $(\Delta\rho)$ / $(\Delta\sigma/\sigma)$. (Note that the reactivity sensitivity is normalized by ρ rather than $|\rho$. Normalizing by $|\rho|$ can potentially lead to c_k and E indices with the wrong sign.) Likewise, if the keyword *absolute* is specified, any relative reactivity sensitivities are internally converted to $(\Delta\rho \cdot \rho/ (\Delta\sigma/\sigma)$, and the SAMS-generated relative *k*eff sensitivities are internally converted to $\left(\Delta k_{eff}/k_{eff}\right)$ / $(\Delta\sigma/\sigma)$. If both the *relative* and *absolute* keywords are specified, the last keyword entered in the PARAMETER block is used to determine the sensitivity format. If neither keyword is entered, the default relative format is applied.

It is important to note that the absolute sensitivity option has been added for user control over the format of the uncertainty edit and the extended uncertainty edit. Using absolute sensitivities, the absolute standard deviation in k_{eff} or ρ is edited. Using relative sensitivities, the relative standard deviation in k_{eff} or ρ is edited. The global indices c_k and E are the same regardless of the sensitivity format option. For other indices and parameters (i.e., G, penalty assessment, and completeness parameter), their respective values can change depending on sensitivity format option. These indices and parameters play an important role in nuclear criticality safety validation, and the default relative format option is recommended.

6.3.1.6 User input

The user input for TSUNAMI-IP is described in this section. The input consists of an optional title on a single line followed by one required and six optional blocks of data which are identified in Table 6.3.1 and individually described in subsequent subsections. These data blocks must begin with **READ KEYNAME**and end with **END KEYNAME**, where **KEYNAME** is the name of an individual data block. The required PARAMETER data block should be entered first, followed by the remaining blocks of data in any order. Note that this is different than the SCALE 5 and SCALE 5.1 versions of TSUNAMI-IP, which allowed the blocks of data to be in any order. The SCALE 6 version will continue if the PARAMETER data block is not entered first, but the keywords *absolute* and *use_diff_groups* (explained below) will have no effect.

Table 6.3.1: Keynames and descriptions for TSUNAMI-IP input data blocks*.

Keyname	Description	Required/Optional
PARAMETER	Indices and parameters to be computed are input in this section. Output edits are requested, and user-input criteria values for numeric data are entered.	Required
APPLICATIONS	File paths to sensitivity data files representing application systems for which validation by the experiments are assessed are input in this section.	Optional*
EXPERIMENTS	File paths to sensitivity data files representing experiments to be used in the analysis are input in this section.	Optional*
RESPONSE	File paths to sensitivity data files representing experiments or applications to be used in the analysis are input in this section.	Optional*
REACTIONS	Specific nuclide-reaction pairs for which analysis with certain indices and parameters are desired can be entered in this section.	Optional
COVARIANCE	User input standard deviation for nuclide-reaction pairs for which cross-section-covariance data are not available can be entered in this section.	Optional
EXCLUSIONS	User input reactions to exclude from c_r and Extended c_r calculations.	Optional
HTML	Parameters to customize the HTML formatted output can be entered in this section.	Optional

*Although the EXPERIMENTS, APPLICATIONS, and RESPONSE data blocks are optional, at least one application and one experiment must be specified on the TSUNAMI-IP input file. This can be done in a variety of ways explained below.

Parameter data

The *PARAMETER* data block is used to request the calculation of the various indices and parameters available in TSUNAMI-IP, request output edits and set criteria values. The parameter block must begin with *READ PARAMETER* and end with *END PARAMETER*. The data input to the parameter data block consist of numerous keywords that are shown, along with their default values and descriptions, in Table 6.3.2. A keyword that ends with = must be followed by numeric data. Keywords that do not end with = are Boolean flags that are used to turn on certain features of the code, such as the computation of certain data or certain output edits. If the keyword is present for a Boolean entry, the value is set to true. Otherwise, the Boolean flag is set to its default value. If no data are requested in the *PARAMETER* section using the Boolean flags, then no data will be produced by the code. The input is designed to maximize user control over the operation of the code. A more detailed description of the indices and parameters is given in Sect. 6.3.1.1 through Sect. 6.3.1.4.

Table 6.3.2: Input data for parameter block of TSUNAMI-IP input.

Keyword	**Default value**	**Description**
absolute	False	Use absolute sensitivities [e.g. $\left(\Delta k_{eff}/\Delta\sigma/\sigma\right)$] for all applications and experiments in the analysis. If *absolute* is entered, the keyword *relative* is set to *False*.
c	False	Compute c_k values for each application compared to each experiment.
c_long	False	Produces extended c_k output edit for each application compared to each experiment.
cechck=	0.5	Level of E and c_k values that trigger the *cediff* warning. If E or c_k are below this value, no warning is printed.
cediff=	0.1	If the E and c_k values for a given application and experiment differ by more than *cediff*, a warning message is printed.
cov_fix	False	Replace zero and large (standard deviation >1000%) values on diagonal of cross-section covariance data with user input values and default values.
coverx=	56group-cov7.1	Name of cross-section covariance data file to use in analysis.
cp	False	Compute and print completeness parameter for each application.
cr	False	Compute c_r values for each application compared to each experiment.
cr_long	False	Produces extended c_r output edit for each application compared to each experiment.

continues on next page

Table 6.3.2 – continued from previous page

inptcase	False	This sets the SCALE free form reader to leave the case of the input data as read. This is useful when sensitivity data file names have upper case letters. If this option is set, all other input keywords MUST be in lower case to be correctly interpreted by SCALE. This keyword must be entered in the input prior to the reading of any titles of the sensitivity data files with capital letters.
large_cov=	10	Cutoff fractional standard deviation value for *cov_fix*. Covariance data with uncertainties larger than *large_cov* are replaced with user-defined or default values. Default = 10, which is 1000% uncertainty.
lg	False	Compute *g* values for fission, capture and scatter for each nuclide for each experiment compared to each application and print them in a table if the application's sensitivity for the corresponding nuclide-reaction pair is greater than or equal to *sencut*. Also compute *g* values for reactions specified in the *REACTIONS* data block.
lgall	False	Print *g* values for fission, capture and scatter for all nuclides for all experiments for each application.
lggroups	False	Print a table listing the numbers of experiments that are at least as sensitive as the application for each group for each reaction requested in the *REACTIONS* data block.
lgsum	False	Print a summary table of *g* for each application for each experiment that exceeds *lgvalue* for each nuclide's capture, fission and scatter reactions
lgvalue=	0.9	Threshold value of *g* for inclusion in summary table.
nixlim=	10	Minimum number of experiments with group-wise values exceeding *senfac* times the group-wise value for the application for the group-wise value to be added to the completeness parameter.
penalty	False	Create penalty assessment based on differences in the application's sensitivity profile for a particular nuclide-reaction pair, and the corresponding composite profile for all qualifying experiments.

continues on next page

Table 6.3.2 – continued from previous page

pencut=	0.0	Cutoff value for excluding sensitivities from the penalty calculation. If the sum of the absolute values of the energy-dependent sensitivity data for a particular nuclide-reaction pair is below this number, the nuclide-reaction pair will not be included in the penalty calculation.
penlgv=	0.0	Use only nuclide-reaction pairs with g values exceeding *penlgv* in the penalty assessment.
penlong	False	Print detailed edits of components of penalty assessments showing each component of the penalty.
penminx=	10	Minimum number of qualifying experiments for each application for penalty calculation.
pensusec	True	Use only experiments with c_k values exceeding *cvalue* in the penalty assessment. If *penusec* is entered, the keywords *penusee* and *penuseg* are set to False.
penusee	False	Use only experiments with E values exceeding *evalue* in the penalty assessment. If *penusee* is entered, the keywords *penusec* and *penuseg* are set to False.
penuseg	False	Use only experiments with G values exceeding *gvalue* in the penalty assessment. If *penuseg* is entered, the keywords *penusec* and *penusee* are set to False.
penwarn	False	Print list of warning messages noting the experiments and nuclide-reaction pairs that were excluded from the penalty calculation.
plot	False	Produces Javapeño formatted plot (.plt) files for integral values and composite sensitivity data.
prtcomp	False	Print "composite" of experiment sensitivity profiles for reactions selected in *REACTIONS* data block. Also write data to sensitivity data file.
prtmtrix	False	Print directory of data available on the cross-section-covariance data library.
prtnotcv	False	Print a table summarizing the non-coverage for the nuclide-reaction pairs entered in the *REACTIONS* data block.
prtparts	False	Print the components of E and G for fission capture in scatter in the *values* table.

continues on next page

Table 6.3.2 – continued from previous page

relative	True	Use relative sensitivities [e.g., $\left(\Delta k_{eff} k_{eff}\right) / (\Delta\sigma/\sigma)$ for all applications and experiments in the analysis. If *relative* is entered, the keyword *absolute* is set to False.
return_work_cov	False	Copy the working covariance library to the return directory with the file name *job_name.wrk.cov,* where *job_name* is the name of the input file. If *return_work_cov* is not present, the working covariance library remains in the temporary working directory with the file name *job_name.wrk*.
sencut=	0.01	Cutoff value for ignoring low valued sensitivities in nuclide-reaction specific edit tables. If the absolute value of the sum of the energy-dependent sensitivity data for a particular nuclide-reaction pair is below this number, the nuclide-reaction pair will not be included in the edit.
senfac=	0.9	Value used in calculation of completeness parameter. Group-wise sensitivity for the application is counted as validated by the experiment if the sensitivity from the experiment is greater than the application sensitivity times *senfac*.
udcov=	0.05	User-defined default value of standard deviation in cross-section data to use for all groups for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_corr=	1.0	User-defined default correlation value to use for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_corr_type=	*zone*	User-defined default correlation in cross-section data to use for nuclide-reaction pairs for which covariance data are not available on the selected data file. Allowed values are *long*, *zone*, and *short.* (See Sect. 6.3.1.6.3 for details on *long*, *zone* and *short.*)
udcov_fast=	0.0	User-defined default value of standard deviation in cross-section data to use for fast data for nuclide-reaction pairs for which covariance data are too large or not available on the selected data file.

continues on next page

Table 6.3.2 – continued from previous page

ud-cov_inter=	0.0	User-defined default value of standard deviation in cross-section data to use for intermediate data for nuclide-reaction pairs for which covariance data are too large or not available on the selected data file.
ud-cov_therm=	0.0	User-defined default value of standard deviation in cross-section data to use for thermal data for nuclide-reaction pairs for which covariance data are too large or not available on the selected data file.
uncert	False	Computes uncertainty in k_{eff} due to covariance data
uncert_long	False	Prints extended table of uncertainty in k_{eff}due to covariance data.
use_dcov	False	Use user-defined default covariance data for nuclide reaction pairs not included on the covariance data file. The user-defined data will be used in the penalty assessment as well as the computation of c_{k} and uncertainty calculations.
use_diff_groups	True	Allow sensitivity data files to have different energy groups. Data files with different energy groups will be internally converted to the energy group structure of the covariance data file. This parameter is now always equal *true* and does not need to be set.
use_icov	False	Use user-defined data input in *COVARIANCE* input data block in place of default values for user input nuclide-reaction pairs that are not on the covariance data file. The user-defined data will be used in the penalty assessment as well as the computation of c_{k} and uncertainty calculations.
usename	False	Use the name of the sensitivity data file in place of its title in all output.
uslstats	False	Produces VADER input files for trending analysis with all experiments for each global integral index (c_{k}, E, and G) requested for each application
uslsummary	False	Produces VADER input files for trending analysis with experiments exceeding cutoff value (*cvalue*, *evalue* or *gvalue*) for each global integral index (c_{k}, E, and G) requested for each application

continues on next page

Table 6.3.2 – continued from previous page

usl_p=	0.9990	Value of P in VADER input files. P is the portion of the population falling above the lower tolerance level
usl_1-g=	0.9500	Value of 1-γ in VADER input files. 1-γ is the confidence on the fit.
usl_alpha=	0.9990	Value of α in VADER input files. α is the confidence on the proportion of P.
usl_xmin=	0.0000	Value of x(min) in VADER input files. x(min) is the minimum value of the parameter x.
usl_xmax=	1.0000	Value of x(max) in VADER input files. x(max) is the maximum value of the parameter x.
usl_sigs=	-0.0030	Value of σ_s in VADER input files. If a positive value is entered, it is applied as the total uncertainty for each experiment. If a negative or zero value is entered, it is used as the experimental measurement uncertainty and any Monte Carlo and cross section uncertainties are added quadratically to the absolute value of *usl_sigs* for each experiment.
usl_dkm=	0.0500	Value of Δk_m in VADER input files. Δk_m is the administrative margin used to ensure subcriticality.
usl_uncert	False	Includes uncertainty in k_{eff} due to cross section covariance data from k_{eff} uncertainty written to VADER input files.
values	False	Print all computed "global" indices (E, c_k, and G) in a table for each application. If *prtparts* is input, also include the partial values

Reaction data

The *REACTIONS* data block is used to specify nuclide-reaction pairs for the *lg* and *lggroups* edits. The reactions block must begin with *READ REACTIONS* and end with *END REACTIONS*. Data are entered in pairs with the nuclide number (e.g., 92235) followed by the reaction MT number (e.g., 18). Alphanumeric input is also accepted (e.g., u-235 fission) for the nuclide-reaction pairs. Mixed input is also acceptable (e.g., 92235 fission). Available reaction types are given in Table 6.3.4.

User input covariance data

The *COVARIANCE* data block allows the user to specify a covariance matrix for specific nuclide-reaction pairs for which covariance data are not present on the covariance data file or that have zero or large values on the diagonal. The *COVARIANCE* data block must begin with *READ COVARIANCE* and end with *END COVARIANCE*.

Table 6.3.3: Input options for user-defined covariance data.

Input Parameter	**Requirement**	**Default Value**	**Allowed Values**	**Description**
Nuclide	Required	none	Nuclide name or ZA number	Nuclide for which covariance data are to be entered
Reaction	Required	none	Reaction name or MT number	Reaction for which covariance data are to be entered. See Table 6.3.4 for available reaction types.
all=	Optional	0.0	any number	Fractional standard deviation value to be applied to all groups.
fast=	Optional	0.0	any number	Fractional standard deviation value to be applied to fast groups. The *fast* value overrides the *all* value in the fast groups.
therm=	Optional	0.0	any number	Fractional standard deviation value to be applied to thermal groups. The *therm* value overrides the *all* value in the thermal groups.
inter=	Optional	0.0	any number	Fractional standard deviation value to be applied to intermediate groups. The *inter* value overrides the *all* value in the intermediate groups.
corr=	Optional	1.0	any number from -1.0 to 1.0	Correlation between groups (see *corr_type* for use)
corr_type	Optional	*zone*	*long*, *short*, *zone*	Type of correlation applied form group-to-group covariance values. *long* – correlation is applied between all groups *short* – correlation is applied only between adjacent groups *zone* – correlation is applied within fast, intermediate and thermal groups, but no correlation is applied between zones
end	Required			Denotes end of input for current nuclide/reaction

Any MT number or reaction name will be treated as a valid input, but only those present on the sensitivity data

files will produce useful information. The reaction sensitivity types computed by SAMS from TSUNAMI-1D and TSUNAMI-3D are shown in Table 6.3.4. An energy-covariance matrix is created for the specified nuclide-reaction pair with the square of the entered standard deviation for the diagonal terms for all groups using the *all*= value. Groups in the fast, intermediate and thermal energies are then set to the square of the standard deviation value entered for *fast*=, *inter*=, and *therm*=, respectively. The off-diagonal terms of the energy matrix are generated according to the input for *corr*=, and *corr_type*=, with default settings of *1.0* and *zone.* Data entered in this block do not override data present on the covariance data file. The SCALE 5.1 input format is supported where data are entered in triplets with the nuclide name (e.g., u-235), then the reaction MT number or name (e.g., 18 or fission), and then a standard deviation value. In this case, the *end* card must not be entered. The standard deviation value is applied to all groups with default setting for correlations. These data are only used if *use_icov* is specified in the *PARAMETER* data block.

Table 6.3.4: Reaction sensitivity types computed by SAMS.

MT	**Reaction**	**SCALE identifier**
0	Sum of scattering	scatter
1	Total	total
2	Elastic scattering	elastic
4	Inelastic scattering	n,n'
16	n,2n	n,2n
18	Fission	fission
101	Neutron disappearance	capture
102	n,γ	n,gamma
103	n,p	n,p
104	n,d	n,d
105	n,t	n,t
106	n,^{3}He	n,he-3
107	n,α	n,alpha
452	$\bar{\nu}$	nubar
1018	χ	Chi

If *use_icov* is specified, these data are used for the calculation of the uncertainty in k_{eff} (*uncert* and *uncert_long* edits), calculation of c_k, Extended c_k, and the penalty assessment for each application. These user input values are only applied where an application has a sensitivity profile for which there is no corresponding covariance matrix on the covariance data file. When both *use_icov* and *cov_fix* are specified in the *PARAMETER* data block, and a reaction has zero or large (standard deviation > 1000%) values on the diagonal of the covariance matrix, these values are replaced with the square of the user input standard deviation value, and the corresponding off-diagonal terms are substituted according to the values of *corr* and *corr_type*.

Specifying user input covariance data for the summative reactions total, scatter and capture have no affect on the results of TSUNAMI-IP, as the summative reactions are not used in uncertainty analysis calculations.

Application, experiment, and response data

Sensitivity data files are designated as either application systems or experiment systems by the *APPLICATIONS, EXPERIMENTS,* or *RESPONSE* data blocks. For each application system, TSUNAMI-IP will calculate integral parameters against each experiment system. Each data block must begin with *READ KEYNAME* and end with *END KEYNAME* where *KEYNAME* can be *APPLICATIONS, EXPERIMENTS,* or *RESPONSE*.

Inside each data block, sensitivity data files are listed using response definition records. A response definition record is a single line of input that contains the sensitivity data filename and nine optional keyword specifications. The sensitivity data filename and optional keywords can be entered in any order, with the following format:

filename (name=N)(use=U)(type=T) (omit) (ev=E) (uv=U) (nu=P) (absolute or abs) (relative or rel)

where

filename = sensitivity data filename. The filename can include the file path.

N = A descriptive identifier for the sensitivity data file in TSUNAMI-IP output edits. The identifier is limited to 20 alphanumeric characters (spaces are not allowed).

U = The file usage. Allowed values are *appl, expt*, and *omit*, signifying the use of the file as an application or an experiment or to exclude the file from the analysis, respectively. The default values for the *use* keyword are described below.

T = 8-character alphanumeric identifier for the response type (e.g. '*keff*'). The response type is used in various output edits along with the *name=N* identifier.

omit - Optional keyword used to omit *filename* from the analysis, can be used independent of *use=*.

E = experimental value of the response (e.g. k_{eff}).

U = uncertainty value of the response.

P = number of uncertainty components to characterize the experiment uncertainty for this response.

absolute - Optional keyword that specifies absolute sensitivities will be used for *filename*.

relative - Optional keyword that specifies relative sensitivities will be used for *filename*.

By default, filenames listed in the *APPLICATIONS* data block are designated as application systems. Likewise, filenames listed in the *EXPERIMENTS* or *RESPONSE* data blocks are designated as experiment systems. The sensitivity data files must be in either the TSUNAMI-A or TSUNAMI-B file format, detailed in Appendix Data File Formats. (TSUNAMI-IP is not currently compatible with, for example, sensitivity data files in the "ABBN" format.) Only the region-integrated sensitivity coefficients from the sensitivity data files are used by TSUNAMI-IP. These data are energy-dependent, but any mixture-dependent or region-dependent data present on the data files are not used. Case-sensitive filenames and their file paths are allowed. However, **spaces are not allowed in the filenames or file paths.**

The nine optional keywords provide more user control over how each sensitivity data file is used in the analysis, and how each sensitivity data file is identified in the TSUNAMI-IP output. By default, TSUNAMI-IP identifies files in the output according to the title on the sensitivity data files. For files that have the same

titles, or have long or non-descriptive titles, the *usename* keyword in the *PARAMETER* data block can be used to identify sensitivity data files by their filename. Although file names are unique, they can also be non-descriptive. For this reason, the *name* keyword on the response definition record can be used to create a new identifier for the sensitivity data file in the TSUNAMI-IP output. Similarly, the *type* keyword can be used to identify the response type in the output. Because TSUNAMI-IP was initially intended for similarity assessment of critical systems, the default response type identifier is '*keff*'; the type identifier for Generalized Perturbation Theory reaction rate sensitivity responses is *'gpt'*. For reactivity sensitivity data files produced by the TSAR module (Chapter 6.4H), the default response type identifier is ρ. The *type* keyword can be used to specify a descriptive response type identifier, such as *type=k_sdm* or *type=alpha_t*.

The *use* keyword specifies how the sensitivity data file is used in the analysis. Allowed values are *expt, appl,* or *omit*. All sensitivity data files on response definition records with *use=expt* are designated as experiment systems, regardless of what data block it is in. Likewise, all sensitivity data files on response definition records with *use=appl* are designated as application systems. In addition, the user can omit the sensitivity data file from the analysis by entering either *use=omit* or simply *omit* on the response definition record.

The experiment value of the response (*ev*=) and the experiment uncertainty of the response (*uv*=) are used in generating VADER input files for trending analysis. If the experiment value is not provided, then the default value used in VADER is 1.0. If the experiment uncertainty value is not provided, then the default used in VADER is the absolute value of the *usl_sigs* value in the *PARAMETER* data block.

The *ev*= and *uv*= keywords do not impact the computed results in TSUNAMI-IP and are only used if the *uslstats* keyword is included in the *PARAMETER* data block. Experiment values of k_{eff} or reactivity only need to be provided for experiment systems. TSUNAMI-IP skips over *ev*= and *uv*= keyword specifications given for application or omitted systems.

In addition to the *uv*= keyword specification, TSUNAMI-IP allows for the experiment uncertainty value to be given in terms of uncertainty components. The *nu*= keyword defines the number of uncertainty components that characterize the experiment uncertainty. If the experiment response uncertainty is given in terms of uncertainty components, the *uv*= keyword specification is optional. An uncertainty component definition record follows the response definition record if the *nu*= keyword specification is given. The uncertainty component definition record has the following format:

uncmp1 val1 uncmp2 val2 …….. uncmpP valP

where

uncmp1 =4-character alphanumeric identifier for the 1st uncertainty component.

val1 =experiment uncertainty for component *uncmp1*.

uncmp2 =4-character alphanumeric identifier for the 2nd uncertainty component.

val2 =experiment uncertainty for component *uncmp2*.

uncmpP =4-character alphanumeric identifier for the Pth uncertainty component.

valP =experiment uncertainty for component *uncmpP*.

The uncertainty component definition record contains *nu=P* pairs of alphanumeric identifiers and numeric values. The experiment uncertainty value is determined by taking the square root of the sum of the squares of each uncertainty component value.

The final optional keywords *abs* and *rel* are used to determine the format of sensitivity and uncertainty data on the response definition record and the uncertainty component definition record. For a k_{eff} response, the following four input definitions produce equivalent uncertainties:

```
1)    name=exp_001 ev=1.001 uv=0.005000 rel C:\sensitivity\k_critical_a.sdf

2)    name=exp_001 ev=1.001 uv=0.005005 abs C:\sensitivity\k_critical_a.sdf

3)    name=exp_001 ev=1.001 nu=2        rel C:\sensitivity\k_critical_a.sdf
       enri 0.003000 sden 0.004000

4)    name=exp_001 ev=1.001 nu=2        abs C:\sensitivity\k_critical_a.sdf
       enri 0.003003 sden 0.004004
```

In the example above, the measured k_{eff} is 1.001 ± 0.5% or 1.001 ± 0.005005. The sensitivity data filename is given as *C:sensitivityk_critical_a.sdf*, and the experiment response is referred to as *exp_001* in the TSUNAMI-IP output. In 1), the *relative* format is used to specify the relative standard deviation of the measured response as 0.005. In 2), the *absolute* format is used to specify the absolute standard deviation of the measured response as 0.005005. Because the TSUNAMI-generated sensitivity data file is in relative format, TSUNAMI-IP internally renormalizes the relative sensitivities to absolute sensitivities. In 3), the *relative* format is used to specify the relative standard deviation of k_{eff} due to two components (*enri* and *sden*) as 0.003 and 0.004 respectively. Using the square root of the sum of the squares approach, the relative standard deviation of k_{eff} is computed to be 0.005. In 4), the *absolute* format is used to specify the absolute standard deviation of k_{eff} due to two components (*enri* and *sden*) as 0.003003 and 0.004004 respectively. The absolute standard deviation of k_{eff} is computed to be 0.005005. Like 2), the sensitivity data file is internally converted to contain absolute sensitivities.

Because of the high-degree of flexibility in designating application systems and experiment systems in the TSUNAMI-IP input, the following two methods are recommended. The first method is consistent with previous versions of TSUNAMI-IP. That is, application systems are listed in the APPLICATIONS data block and experiment systems are listed in the EXPERIMENTS data block. The second method would be to list both sets of systems in a single RESPONSE data block and use the *use=U* keyword specification to designate how each file is to be used. Sample input files for both methods are given for the TSUNAMI-IP example problem in Sect. 6.3.1.6.7.

Reactions excluded from c_r

The *EXCLUSIONS* data block contains lists of reaction types that will be removed from consideration in the calculation of the c_r integral index and in the extended c_r edit. Valid input for this block are reaction MT numbers or names (e.g., 18 or fission). Values must be separated by a space or a line break. Any MT number or reaction name will be treated as a valid input, but only those present on the sensitivity data files will produce useful information. The reaction sensitivity types computed by SAMS from TSUNAMI-1D and TSUNAMI-3D are shown in Table 6.3.4.

HTML data

The optional *HTML* data block is used to customize HTML formatted output. The *HTML* data block must begin with *READ HTML* and end with *END HTML*. The data input to the *HTML* data block consist of several keywords that are shown, along with their default values and descriptions, in Table 6.3.5. A keyword that ends with = must be followed by text data. For color entries, any valid html color name can be entered or the hexadecimal representation can be used if preceded by a # sign. For example, to change the background color of the html output to white, *bg_clr=white* and *bg_clr=#ffffff* have the same effect, because ffffff is the hexadecimal representation of white. An extensive list of available colors for customized output is shown in Appendix BH. Please note that not all features are supported by all browsers.

Table 6.3.5: Input data for html block of TSUNAMI-IP input.

Keyword	Default value	Description
bg_clr=	papayawhip	Background color.
h1_clr=	maroon	Color used for major headings.
h2_clr=	navy	Color used for sub-headings.
txt_clr=	black	Color for plain text.
lnk_clr=	maroon	Color for hyperlinks.
lnk_dec=	none	Decoration for hyperlinks. (none, underline, overline, line-through, blink).
vlnk_clr	navy	Color for visited hyperlinks.
max_clr=	maroon	Color for maximum values in tables.
cut_clr=	navy	Color for values in tables that exceed cutoff values.
ud_clr=	blue	Color for values in tables that use default covariance data.
ui_clr=	Red	Color for values in tables that use user-input covariance data.
udfix_clr=	royalblue	Color for values in tables that use default corrected covariance data.
uifix_clr=	green	Color for values in tables that use user-input corrected covariance data.

Example TSUNAMI-IP input

An example TSUNAMI-IP input listing is given in Example 6.3.1. In this example, the optional title for this analysis is entered as "tsunami-ip example". The parameter data are used to request that E, c_k, G, and g be computed (e, c, g, and lg, respectively). The integral indices E, c_k, and G will be listed in a table for each application containing the values of each index for each experiment (*values*). The cutoff values for E and c_k for inclusion in the summary tables are set to 0.8 (*evalue=0.8* and *cvalue=0.8*). Summary tables listing experiments with E values exceeding *evalue*, $c_k$values exceeding *cvalue* will be produced (*esummary*, and *csummary*, respectively). The *REACTIONS* data block requests that the n,γ reaction for ^{235}U and the elastic scattering reaction for ^{1}H be assessed for coverage with the g index.

Example 6.3.1: Example TSUNAMI-IP input listing.

```
=shell
copy c:\applications\1sen.sdf
copy c:\applications\2sen.sdf
copy c:\applications\3sen.sdf
end
=tsunami-ip
tsunami-ip example
read parameter
  e c g lg
  values
  evalue=0.8
  cvalue=0.8
  esummary csummary tesum tcsum
end parameter
read applications
  1sen.sdf
  2sen.sdf
  3sen.sdf
end applications
read experiments
  c:\sensitivity\bnwl2129t3-01sen.sdf
  c:\sensitivity\bnwl2129t3-02sen.sdf
  c:\sensitivity\bnwl2129t3-03sen.sdf
  c:\sensitivity\bnwl2129t3-04sen.sdf
  c:\sensitivity\bnwl2129t3-05sen.sdf
end experiments
```

(continues on next page)

(continued from previous page)

```
read reactions
  u-235 n,gamma
  h-1 elastic
end reactions
end
```

The application systems are represented by the sensitivity data files named 1sen.sdf, 2sen.sdf and 3sen.sdf. In this example input, these data files are copied (or they could be linked in UNIX) to the temporary directory where SCALE executes and writes its scratch files via system commands that are input to the SHELL utility at the top of the input file.

The experiments for which the indices will be computed for each application are listed in the *experiments* data block. Here, an explicit file path is specified in the input. This could also have been done in the APPLICATIONS data block. In this case, the 5 data files would have to exist in a directory called c:\sensitivity. An equivalent input file using a single *RESPONSE* data block is given in Example 6.3.2.

Example 6.3.2: Example TSUNAMI-IP input listing with Response Block.

```
=shell
copy c:\applications\1sen.sdf
copy c:\applications\2sen.sdf
copy c:\applications\3sen.sdf
end
=tsunami-ip
tsunami-ip example
read parameter
  e c g lg
  values
  evalue=0.8
  cvalue=0.8
  esummary csummary tesum tcsum
end parameter
read response
  1sen.sdf                              use=appl
  2sen.sdf                              use=appl
  3sen.sdf                              use=appl
  c:\sensitivity\bnwl2129t3-01sen.sdf use=expt
  c:\sensitivity\bnwl2129t3-02sen.sdf use=expt
  c:\sensitivity\bnwl2129t3-03sen.sdf use=expt
  c:\sensitivity\bnwl2129t3-04sen.sdf use=expt
  c:\sensitivity\bnwl2129t3-05sen.sdf use=expt
end response
read reactions
  u-235 n,gamma
  h-1 elastic
end reactions
end
```

6.3.1.7 Example PROBLEM

An example problem to demonstrate the use of TSUNAMI-IP is based on the TSUNAMI-1D and TSUNAMI-3D sample problems available with the SCALE distribution. The systems included in the sample problem are not intended to provide a rigorous demonstration of the methodologies of TSUNAMI-IP, but merely to demonstrate how to use the software and read the output. The problem does not correspond to the sample problem distributed with SCALE and is included for illustrative purposes only.

Input description

The example problem input listing is shown in Example 6.3.3. The input begins with the SCALE shell module, which executes a series of copy commands to move the sensitivity data files required for code execution from the user's directory (${INPDIR}) to the current working directory. Next TSUNAMI-IP is called with =tsunami-ip. The optional title card is entered as "tsunami-ip example 2." In the *PARAMETER* data block, the *values* table is requested and it will contain c_k, E and G, since *c e g* are entered on the next line of input. The cutoff value for inclusion of c_k values in the summary table is set at 0.8 with *cvalue=0.8* and the c_k summary table is requested with *csummary*. The nuclide-reaction specific integral indices extended c_k and g are requested with the input *c_long* and *lg*. The summary table of g values exceeding the cutoff is requested with *lgsum*, and the cutoff value for this table is set to 0.75 with *lgvalue=0.75.* The composite sensitivity profiles for the reactions entered in the *REACTIONS* input section are requested with *prtcomp*, and the summary of non-coverage edit is requested with *prtnotcv*. The uncertainty in the k_{eff} values for the experiments and applications due to cross section uncertainties is requested with *uncert* and the extended uncertainty edits for the applications are requested with *uncert_long*. The penalty assessment is requested with *penalty*, the minimum number of experiments exceeding the global integral index; the cutoff value is set to 0 with *penminx=0*. In this way, the limited number of experiments used in this sample problem will still produce a penalty edit. The detailed edit of the components of the penalty are requested with *penlong*. The user-defined default value for covariance data are set to 100% uncertainty for cross sections without covariance data with *udcov*=1.0. The *use_dcov* input keyword activates the use of the *udcov* value in the computation of c_k and the penalty assessment. The use of filenames in place of the titles on the data files is requested with *usename*.

This input defines two systems as applications in the *APPLICATIONS* data block. The application systems are defined as the system with sensitivity data in the data files tsunami-3d_k5-1.sdf and tsunami-3d_k5-2.sdf. Three experiments are defined in the *EXPERIMENTS* data block. These systems have sensitivity data in the data files tsunami-1d1.sdf, tsunami-1d4.sdf, and tsunami-3d_k5-3.sdf.

Example 6.3.3: TSUNAMI-IP sample problem input.

```
=shell
cp ${INPDIR}/tsunami-1d1.sdf .
cp ${INPDIR}/tsunami-3d_k5-1.sdf .
cp ${INPDIR}/tsunami-3d_k5-2.sdf .
cp ${INPDIR}/tsunami-1d4.sdf .
cp ${INPDIR}/tsunami-3d_k5-3.sdf .
end
=tsunami-ip
tsunami-ip example 2
read parameter
  values
  c e g
  cvalue=0.8 csummary
  c_long
  lg lgsum lgvalue=0.75
  prtcomp
  prtnotcv
  uncert uncert_long
  penalty penminx=0 penlong
  udcov=1.0
  use_dcov
  use_icov
  html usename
end parameter
read applications
  tsunami-3d_k5-1.sdf
```

(continues on next page)

(continued from previous page)

```
  tsunami-3d_k5-2.sdf
end applications
read experiments
  tsunami-1d1.sdf
  tsunami-1d4.sdf
  tsunami-3d_k5-3.sdf
end experiments
read reactions
  u-235 nubar u-238 capture
  h-1 total
  b-10 capture
end reactions
end
```

Output Description

The output listings shown here are for descriptive purposes only. The data generated from an actual execution of this problem may differ from that shown below. The Applications List and the Experiments List provide some details from the sensitivity data files. These include the title on the data files, the filename, the number of energy groups and the number of sensitivity profiles. After the Experiments List, Covariance Warnings are printed during the creation of the working (i.e., problem-dependent) covariance library. The nuclide-reactions pairs present on the sensitivity data files for which covariance data are not available on the *coverx* file are detailed. This edit is shown in Example 6.3.5.

The next output edit, shown in Example 6.3.6 details the global integral indices for the current application compared to all experiments. The edit is only present if *values* is entered in the *PARAMETER* data block. The first row of data in the table displays the application compared to itself. This is useful to confirm the normalization of the integral indices. The data detailing the application compared to itself is not displayed in the HTML output. The next rows display the values of the integral indices requested by the user for each experiment as compared to the current application. The values of the response (i.e., k_{eff} or reactivity) for the application and each experiment are also shown in this table. If the sensitivity data were computed with Monte Carlo methods, the statistical uncertainties are shown. In Example 6.3.6, the k_{eff} value for tsunami-1d1 does not show a statistical uncertainty because these sensitivity data were computed with TSUNAMI-1D. The integral values of the application compared to tsunami-1d1 do show statistical uncertainties because the application system was computed with TSUNAMI-3D. The column titles "xsec unc %" gives the relative uncertainty in the response due to cross section data uncertainties, in percent.

The next output edit is the extended uncertainty edit for the application and is shown in Example 6.3.7. This edit details the uncertainty in the response due to each nuclide-reaction to nuclide-reaction covariance matrix. Where default or user input values for covariance data are used, they are indicated in the output edit. The total uncertainty can be computed from the individual values by squaring each term, adding the values that had positive signs, subtracting the values that had negative signs, then taking the square root. The edit is present because *uncert_long* was included in the parameter input.

The next output edit is the extended c_k edit, and an example edit for Application 1 with Experiment 1 is shown in Example 6.3.8. This edit details the contribution to c_k due to each nuclide-reaction to nuclide-reaction covariance matrix. Where default or user input values for covariance data are used, they are indicated in the output edit. The edit is present because *c_long* was included in the parameter input. The c_k value can be computed by summing the individual values. Similar tables present the output for each application compared to each experiment.

The next output edit includes the nuclide-reaction specific integral indices and is shown in Example 6.3.9. In this sample problem, g indices for fission capture and scatter reactions for each nuclide-reaction pair with

a sensitivity coefficient exceeding *sencut* in magnitude are printed for each experiment compared to the application. The first table of values is for the application compared to itself. This initial edit is useful to test the normalization of the integral indices for the application. This initial table of the application compared with itself is not available in the HTML output. The energy-integrated sensitivity coefficient for the application is also shown for each reaction. Statistical uncertainties in these integral values are not available in this version of TSUNAMI-IP. This table is repeated for each experiment compared to the application. Because some reactions were requested in the *REACTIONS* section of the input, and the *g* integral index was computed, a table of the user requested *g* values for the application compared to each experiment is given. In this sample problem, ^{10}B was not present in the application, even though it was requested in the input. TSUNAMI-IP lists all requested reactions in this table, but leaves the values blank when the nuclide or reaction is not available on the sensitivity data file for the application.

The next output edit, shown in Example 6.3.10, gives the extended penalty assessment for application 1. This edit is present because *penlong* was entered in the parameter data block. The penalty edit utilizes the user defined default uncertainties defined by *udcov* because *use_dcov* was entered in the *PARAMETER* data block. Nuclide reaction pairs that utilized default uncertainty data are indicated with *. In this case, when the sensitivity coefficients of the application that are not covered by the experiments are propagated to an uncertainty in the application response with the cross section covariance data value of 0.0097% ± 0.00007% results. Next a listing of the contributions to the penalty from each nuclide-reaction to nuclide-reaction covariance matrix is given. As with the extended uncertainty edit, the cumulative penalty can be computed from the individual values by squaring each term, adding the values that had positive signs, subtracting the values that had negative signs, then taking the square root.

Each of the previous edits would repeat in the full output listing for each application. The next series of output edits, shown in Example 6.3.11, are the summary edits requested by the user. The c_k summary table, requested with the keyword *csummary*, indicates that one experiment has a c_k value at least as great as 0.8 compared to each application. The *g* summary table lists the *g* values that exceed *lgvalue* for the fission, capture and scatter reactions for each nuclide in the application with an energy-integrated sensitivity coefficient exceeding *sencut* in magnitude. The value of the sensitivity coefficient for the application is also given. The summary of non-coverage edit is present because *prtnotcv* was entered in the *PARAMETER* input data block. This edit lists a number of properties of each reaction input in the *REACTIONS* input data block. The final summary table is the penalty summary table, which is activated when *penalty* is listed in the *PARAMETER* data block. The penalties for all applications identified in the *APPLICATIONS* input data block are listed in the penalty summary table.

The composite sensitivity data are listed next in the TSUNAMI-IP output as shown in Example 6.3.12 The group-wise values for the composite profiles for each nuclide-reaction pair identified in the *REACTIONS* data block are listed here for each application identified in the *APPLICATIONS* data block. If a reaction is requested that is not present in the application, the reaction is left blank in the table, as is the case in this sample problem for ^{10}B capture. The composite sensitivity profiles are also written to a TSUNAMI-B formatted sensitivity data file. In this data file, the composite sensitivity profiles are written for each application in the order they were requested in the *REACTIONS* data block. If a requested reaction is not available in the application, it is not written to the data file. The application number is listed on the data file in the position of the material number but with a value that is the negative of the application number. All unit and region information are entered as zeros as is the k_{eff} value. This data file can be correctly read by Javapeño such that the composite data can be plotted. In Javapeño, the application number appears in the list of available sensitivity profiles identified as "composite." The application sensitivity data and the composite data as used in the penalty calculation are also included on the composite sensitivity data file. These new data are identified as "application" and "cut composite" respectively.

Example 6.3.4: Input echo from TSUNAMI-IP output.

```
------------------------------------------------------------------------------------------------------------------------
|                                                                                                                      |
|       TSUNAMI-IP:  Tools for Sensitivity and UNcertainty Analysis Methodology Implementation - Indices and Parameters |
|                                                                                                                      |
|                                                tsunami-ip example 2                                                  |
|                                                                                                                      |
------------------------------------------------------------------------------------------------------------------------
|                                                                                                                      |
|                                            TSUNAMI-IP Parameter Table                                                |
|                                                                                                                      |
|  PARAMETER      VALUE                                DESCRIPTION                                                     |
|                                                                                                                      |
|  absolute       false          Print uncertainty values and penalty assessments in absolute format.  This is the default |
|                                format. Relative format can be specified using the "rel" keyword in the APPLICATIONS,  |
|                                EXPERIMENTS, or RESPONSE input blocks.                                                |
|                                                                                                                      |
|  c              true           Compute c(k) values for each application compared to each experiment.                  |
|                                                                                                                      |
|  c_long         true           Produces extended c(k) output edit for each application compared to each experiment.   |
|                                Sets c to true.                                                                       |
|                                                                                                                      |
|  cechck=        .5000          Level of E and c(k) values that trigger the cediff warning.  If E or c(k) are below this |
|                                value, no warning is printed.                                                         |
|                                                                                                                      |
|  cediff=        .1000          If the E and c(k) values for a given application and experiment differ by more than   |
|                                cediff, a warning message is printed.                                                 |
|                                                                                                                      |
|  coverx=        56groupcov7.1  Name of cross section covariance data file.                                           |
|                                                                                                                      |
|  cov_fix        false          Replace zero and large values on diagonal of cross-section covariance data with user  |
|                                input values and dcov value.                                                          |
|                                                                                                                      |
|  cp             false          Compute and print completeness parameter for each application.                        |
|                                                                                                                      |
|  csummary       true           Print summary table of c(k) values that meet or exceed the cvalue limit.  Sets c to true. |
|                                                                                                                      |
|  cvalue=        .8000          Threshold value of c(k) for inclusion in the summary table.                           |
|                                                                                                                      |
|  cr             false          Compute c(r) values for each application compared to each experiment.                  |
|                                                                                                                      |
|  cr_long        false          Produces extended c(r) output edit for each application compared to each experiment.   |
|                                Sets cr to true.                                                                      |
|                                                                                                                      |
```

(continues on next page)

(continued from previous page)

crsummary	false	Print summary table of c(r) values that meet or exceed the crvalue limit. Sets cr to true.	
crvalue=	.9000	Threshold value of c(r) for inclusion in the summary table.	
e	true	Compute E(sum) values for each application compared to each experiment.	
esummary	false	Print summary table of E(sum) values that meet or exceed the evalue limit. Sets e to true.	
evalue=	.9000	Threshold value of E(sum) for inclusion in the summary table.	
g	true	Compute G values.	
gsummary	false	Print summary table of G values that meet or exceed the gvalue limit. Sets g to true.	
gvalue=	.9000	Threshold value of G for inclusion in summary table.	
html	true	Generate output in html format in addition to the standard text output.	
inptcase	false	Set the SCALE free form reader to leave the case of the input data as read.	
large_cov=	10.0000	Cutoff fractional standard deviation value for cov_fix. Covariance data with uncertainties larger than large_cov are replaced with user-defined or default values.	
lg	true	Compute g values for fission, capture and scatter for each nuclide for each experiment compared to each application and print them in a table if the application's sensitivity for the corresponding nuclide-reaction pair is greater than or equal to sencut.	
lgall	false	Print g values for fission, capture and scatter for all nuclides for all experiments for each application.	
lggroups	false	Print a table listing the number of experiments that are at least as sensitive as the application for each group for each reaction requested in the REACTIONS input block.	
lgsum	true	Prints a summary table of g for each application for each experiment that exceeds lgvalue for each nuclide's capture, fission and scatter reactions.	
lgvalue=	.7500	Threshold value of g for inclusion in summary table.	
nixlim=	10	Minimum number of experiments with group-wise values exceeding senfac times the group-wise value for the application for the group-wise value to be added to the completeness parameter.	
penalty	true	Creates penalty assessment based on differences in the application's sensitivity profile for a particular nuclide-reaction pair, and the corresponding composite profile for all	

(continues on next page)

(continued from previous page)

		qualifying experiments.
pencut=	0.0000E+00	Cutoff value for excluding sensitivities from the penalty calculation. If the sum of the absolute values of the energy-dependent sensitivity data for a particular nuclide-reaction pair is below this number, the nuclide-reaction pair will not be included in the penalty calculation.
penlgv=	0.0000E+00	Use only nuclide-reaction pairs with g values exceeding penlgv in the penalty assessment.
penlong	true	Print detailed edits of components of penalty assessments.
penminx=	0	Minimum number of qualifying experiments for each application for penalty calculation.
penusec	true	Use only experiments with c(k) values exceeding cvalue in the penalty assessment. Sets c to true.
penusee	false	Use only experiments with E(sum) values exceeding evalue in the penalty assessment. Sets e to true.
penuseg	false	Use only experiments with G values exceeding gvalue in the penalty assessment. Sets g to true.
penwarn	false	Print list of warning messages noting the experiments and nuclide-reaction pairs that were excluded from the penalty calculation.
plot	false	Produces Ptolemy formatted plot (.ptp) files for integral values and composite sensitivity data.
prtcomp	true	Print "composite" of experiment sensitivity profiles for reactions selected in REACTIONS input block and write data to sensitivity data file. Sets g to true.
prtmtrix	false	Prints directory of data available on the cross section covariance data library.
prtnotcv	true	Print a table summarizing the non-coverage for the nuclide-reaction pairs entered in the REACTIONS input block. Sets g to true.
prtparts	false	Print the components of E and G for fission capture in scatter in the values table.
relative	true	Print uncertainty values and penalty assessments in relative format. This is the default format. Absolute format can be specified using the "abs" keyword in the APPLICATIONS, EXPERIMENTS, or RESPONSE input blocks.
return_work_cov	false	Option to copy the working covariance data file back to the return directory.
sencut=	.0100	Cutoff value for ignoring low valued sensitivities in nuclide-reaction specific edit tables.

(continues on next page)

(continued from previous page)

senfac=	.9000	Value used in calculation of completeness parameter. Group-wise sensitivity for the application is counted as validated by the experiment if the sensitivity from the experiment is greater than the application sensitivity times senfac.
uncert	true	Computes uncertainty in response due to covariance data.
uncert_long	true	Prints extended table of uncertainty in response due to covariance data. sets uncert to true.
use_dcov	true	Use user-defined default covariance data, udcov, for nuclide reaction pairs not included on the covariance data file. The user-defined data will be used in the fourth penalty assessment as well as the computation of c(k) and uncertainty calculations.
udcov=	1.0000	User-defined default value of standard deviation in cross-section data to use for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_corr=	1.0000	User-defined default correlation value to use for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_corr_type	zone	User-defined default correlation in cross-section data to use for nuclide-reaction pairs for which covariance data are not available on the selected data file. (long, zone, short)
udcov_fast=	0.0000	User-defined default value of standard deviation in cross-section data to use for fast data for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_inter=	0.0000	User-defined default value of standard deviation in cross-section data to use for intermediate data for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_therm=	0.0000	User-defined default value of standard deviation in cross-section data to use for thermal data for nuclide-reaction pairs for which covariance data are not available on the selected data file.
use_diff_groups	true	Allow sensitivity data files to have different energy group structures.
use_icov	true	Use user-defined data input in COVARIANCE input data block in place of udcov value for user input nuclide-reaction pairs that are not on the covariance data file. The user-defined data will be used in the fourth penalty assessment as well as the computation of c(k) and uncertainty calculations.
usename	true	Use the name of the sensitivity data file in place of its title in all output.

(continues on next page)

(continued from previous page)

uslstats	false	Produces USLSTATS input files for trending analysis with all experiments for each global integral index (c(k), E, and G) requested for each application
uslsummary	false	Produces USLSTATS input files for trending analysis with experiments exceeding cutoff value (cvalue, crvalue, evalue or gvalue) for each global integral index (c(k), E, and G) requested for each application
usl_uncert	false	Includes uncertainty in k-eff due to cross section covariance data from k-eff uncertainty written to USLSTATS input files.
usl_p=	.9990	Value of P in USLSTATS input files. P is the portion of the population falling above the lower tolerance level
usl_1-g=	.9500	Value of 1-gamma in USLSTATS input files. 1-gamma is the confidence on the fit.
usl_alpha=	.9990	Value of alpha in USLSTATS input files. Alpha is the confidence on the proportion of P.
usl_xmin=	.0000	Value of x(min) in USLSTATS input files. X(min) is the minimum value of the parameter x.
usl_xmax=	1.0000	Value of x(max) in USLSTATS input files. X(max) is the maximum value of the parameter x.
usl_sigs=	-.0030	Value of sigma(s) in USLSTATS input files. If a positive value is entered, it is applied as the total uncertainty for each experiment. If a negative or zero value is entered, it is used as the experimental measurement uncertainty and any Monte Carlo and cross section uncertainties are added quadratically to the absolute value of usl_sigs for each experiment.
usl_dkm=	.0500	Value of delta-k(m) in USLSTATS input files. Delta-k(m) is the administrative margin used to ensure subcriticality.
values	true	Print all computed "global" indices (E, c(k), and G) in a table for each application. If prtparts is input, also include the partial values.

User Requested Reactions for g Analysis

NUCLIDE ID	NUCLIDE	REACTION	MT
92235	u-235	nubar	452
92238	u-238	capture	101
1001	h-1	total	1
5010	b-10	capture	101

(continues on next page)

(continued from previous page)

HTML Format Options

PARAMETER	VALUE	DESCRIPTION
bg_clr=	PapayaWhip	Background color
h1_clr=	Maroon	Color used for major headings
h2_clr=	Navy	Color used for sub-headings
txt_clr=	Black	Color for plain text
lnk_clr=	Navy	Color for hyperlinks
lnk_dec=	none	Decoration for hyperlinks (none, underline, overline, line-through, blink)
vlnk_clr=	Navy	Color for visited hyperlinks
max_clr=	Red	Color for maximum values in tables
cut_clr=	Blue	Color for values in tables that exceed cutoff values
ud_clr=	Blue	Color for values in tables that use default covariance data
ui_clr=	Red	Color for values in tables that use user-input covariance data
udfix_clr=	RoyalBlue	Color for values in tables that use default corrected covariance data
uifix_clr=	Green	Color for values in tables that use user-input corrected covariance data

Applications List

APPLICATION	DESCRIPTION
1	Title: tsunami-3d sample 1 Name: tsunami-3d_k5-1, Type: keff , Format: Relative 238 groups, 45 Sensitivity profiles
2	Title: tsunami-3d sample 2 Name: tsunami-3d_k5-2, Type: keff , Format: Relative 238 groups, 406 Sensitivity profiles

(continues on next page)

(continued from previous page)

```
---------------------------------------------------------------------------------------------------------------------------------
|                                                                                                                               |
|                                                      Experiments List                                                         |
|                                                                                                                               |
|  EXPERIMENT          DESCRIPTION                                                                                              |
|                                                                                                                               |
|      1        Title: tsunami-1d sample 1 - keff                                                                                |
|              Name: tsunami-1d1, Type: keff    , Format: Relative                                                              |
|              Filename: tsunami-1d1.sdf                                                                                        |
|              238 groups, 45 Sensitivity profiles                                                                              |
|                                                                                                                               |
|      2        Title: tsunami-1d sample 4                                                                                       |
|              Name: tsunami-1d4, Type: keff    , Format: Relative                                                              |
|              Filename: tsunami-1d4.sdf                                                                                        |
|              238 groups, 30 Sensitivity profiles                                                                              |
|                                                                                                                               |
|      3        Title: tsunami-3d sample 3                                                                                       |
|              Name: tsunami-3d_k5-3, Type: keff    , Format: Relative                                                          |
|              Filename: tsunami-3d_k5-3.sdf                                                                                    |
|              238 groups, 182 Sensitivity profiles                                                                             |
|                                                                                                                               |
|                                                                                                                               |
```

Example 6.3.5: Warning messages from TSUNAMI-IP output.

```
------------------------------------------------------------------------------------------
Covariance Warnings in creating working COVERX library
------------------------------------------------------------------------------------------

  Default data will be used for mg n,n'
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for mg n,2n
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for mg n,p
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for mg n,alpha
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for al-27 n,d
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for al-27 n,t
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for si n,2n
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for si n,d
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for s n,n'
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for s n,2n
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for s fission
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for s n,p
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for s n,d
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for s n,t
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for s n,alpha
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for ca n,n'
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for ca n,2n
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for ca n,p
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for ca n,d
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
  Default data will be used for ca n,t
    all= 1.0000 therm= 0.0000 inter = 0.0000 fast= 0.0000 corr= 1.0000 corr_type= zone
```

Example 6.3.6: Global integral indices values table from TSUNAMI-IP output.

```
---------------------------------------------------------------
Integral Values for Application #1
---------------------------------------------------------------

       Experiment          Type     Value      s.d.       xsec unc %    s.d.     c(k)   s.d.     E      s.d.  ␣
↪  G     s.d.
  ------------------------- -------- ---------- ---------  ---------- ---------  ------ ------  ------ ------  -
↪----- ------
     0 tsunami-3d_k5-1      keff      1.0078E+0 1.0000E-3  6.10990E-1 2.1762E-5  1.0000 0.0001  1.0000 0.0005 ␣
↪1.0000 0.0000
     1 tsunami-1d1          keff      1.0073E+0            6.06419E-1            0.9999 0.0000  0.9999 0.0003 ␣
↪0.9801 0.0008
     2 tsunami-1d4          keff      1.0040E+0            1.42001E+0            0.0988 0.0000  0.0164 0.0000 ␣
↪0.0542 0.0016
```

(continues on next page)

(continued from previous page)

```
     3 tsunami-3d_k5-3      keff      9.9575E-1 9.9300E-4  7.48243E-1 1.2500E-4  0.6442 0.0001  0.9647 0.0008 ␣
↪0.6754 0.0026
```

Example 6.3.7: Extended uncertainty edit from TSUNAMI-IP output.

```
 ---------------------------------------------------------------------------------------------------------------------
↪---------
    Extended Uncertainty Edit for Application 1 tsunami-3d1
 ---------------------------------------------------------------------------------------------------------------------
↪---------

 ---------------------------------------------------------------------------------------------------------------------
↪------------------

   Input covariance file: 56groupcov7.1
 Working covariance file: tsunami-ip.wrk


   The standard deviation of keff ( % dk/k ) is:
   6.110E-01 +/- 2.176E-05 percent
 ---------------------------------------------------------------------------------------------------------------------
↪------------------

    * indicates default covariance data

   contributions to uncertainty in keff ( % dk/k ) by individual energy covariance matrices:

                      covariance matrix
           nuclide-reaction    with    nuclide-reaction         % dk/k due to this matrix
        ----------------------      ----------------------   ----------------------------------
               u-238 n,gamma               u-238 n,gamma         3.8714E-01 +/- 6.2871E-06
               u-235 nubar                 u-235 nubar           2.8509E-01 +/- 7.9001E-06
               u-238 n,n'                  u-238 n,n'            2.2073E-01 +/- 7.7594E-06
               u-235 n,gamma               u-235 n,gamma         1.6006E-01 +/- 1.7559E-06
                f-19 elastic                f-19 elastic         1.3624E-01 +/- 5.0707E-06
               u-238 elastic               u-238 n,n'           -1.2828E-01 +/- 1.7674E-06
               u-235 fission               u-235 n,gamma         1.2387E-01 +/- 8.3076E-07
               u-235 fission               u-235 fission         1.2134E-01 +/- 1.2085E-06
                 h-1 elastic                 h-1 elastic         1.1972E-01 +/- 2.1606E-06
                f-19 elastic                f-19 n,n'           -1.1793E-01 +/- 3.1965E-06
                f-19 n,n'                   f-19 n,n'            1.1286E-01 +/- 3.8652E-06
               u-235 chi                   u-235 chi             8.8178E-02 +/- 1.5583E-05
               u-238 elastic               u-238 elastic         6.9520E-02 +/- 1.1586E-06
               u-238 nubar                 u-238 nubar           5.8614E-02 +/- 5.4192E-07
                 h-1 n,gamma                 h-1 n,gamma         5.0829E-02 +/- 1.6728E-07
               u-238 elastic               u-238 n,gamma         5.0286E-02 +/- 1.7408E-06
                f-19 n,alpha                f-19 n,alpha         1.9795E-02 +/- 1.0127E-07
               u-238 fission               u-238 fission         1.7394E-02 +/- 3.4394E-08
                   c elastic                   c elastic         1.5520E-02 +/- 5.5754E-08
               u-238 n,2n                  u-238 n,2n            1.3981E-02 +/- 1.2056E-07
                f-19 n,gamma                f-19 n,gamma         9.7994E-03 +/- 6.0845E-09
                   c n,n'                      c elastic        -9.0325E-03 +/- 3.2330E-08
                   c n,n'                      c n,n'            8.6479E-03 +/- 5.6289E-08
                f-19 elastic                f-19 n,alpha         6.6750E-03 +/- 1.2243E-08
```

Example 6.3.8: Extended c_k edit from TSUNAMI-IP output.

```
-------------------------------------------------------------------------------------------------------------------------
    Extended c(k) Edit for Application 1 tsunami-3d1 with Experiment 1 tsunami-1d1
-------------------------------------------------------------------------------------------------------------------------

-------------------------------------------------------------------------------------------------------------------------

   the c(k) value is:

     1.0000 +/- 0.0001
 -------------------------------------------------------------------------------------------------------------------------


   Input covariance file: 56groupcov7.1
 Working covariance file: tsunami-ip.wrk

    * indicates default covariance data

   contributions to c(k) by individual energy covariance matrices
   the c(k) value is the sum of the individual contributions

                      covariance matrix
          nuclide-reaction    with    nuclide-reaction       c(k) contribution from this matrix         individual c(k)
        ---------------------       ----------------------   ----------------------------------  ----------------------------
               u-238 n,gamma               u-238 n,gamma        4.0149E-01 +/- 1.4552E-05          1.0000E+00 +/- 3.6246E-05
               u-235 nubar                 u-235 nubar          2.1772E-01 +/- 1.8286E-05          1.0000E+00 +/- 8.3989E-05
               u-238 n,n'                  u-238 n,n'           1.3051E-01 +/- 1.7960E-05          1.0000E+00 +/- 1.3761E-04
               u-235 n,gamma               u-235 n,gamma        6.8631E-02 +/- 4.0642E-06          1.0000E+00 +/- 5.9217E-05
                f-19 elastic                f-19 elastic        4.9723E-02 +/- 1.1737E-05          1.0000E+00 +/- 2.3604E-04
               u-235 fission               u-235 fission        3.9440E-02 +/- 2.7972E-06          1.0000E+00 +/- 7.0924E-05
                 h-1 elastic                 h-1 elastic        3.8391E-02 +/- 5.0010E-06          1.0000E+00 +/- 1.3026E-04
                f-19 n,n'                   f-19 n,n'           3.4122E-02 +/- 8.9464E-06          1.0000E+00 +/- 2.6219E-04
               u-238 elastic               u-238 n,n'          -2.2039E-02 +/- 2.8927E-06
               u-238 n,n'                  u-238 elastic       -2.2039E-02 +/- 2.8927E-06
               u-235 chi                   u-235 chi            2.0828E-02 +/- 3.6069E-05          1.0000E+00 +/- 1.7317E-03
               u-235 fission               u-235 n,gamma        2.0550E-02 +/- 1.3597E-06
               u-235 n,gamma               u-235 fission        2.0550E-02 +/- 1.3597E-06
                f-19 elastic                f-19 n,n'          -1.8628E-02 +/- 5.2316E-06
                f-19 n,n'                   f-19 elastic       -1.8628E-02 +/- 5.2316E-06
               u-238 elastic               u-238 elastic        1.2947E-02 +/- 2.6818E-06          1.0000E+00 +/- 2.0714E-04
               u-238 nubar                 u-238 nubar          9.2031E-03 +/- 1.2543E-06          1.0000E+00 +/- 1.3629E-04
                 h-1 n,gamma                 h-1 n,gamma        6.9207E-03 +/- 3.8719E-07          1.0000E+00 +/- 5.5946E-05
               u-238 elastic               u-238 n,gamma        3.3869E-03 +/- 2.8491E-06
               u-238 n,gamma               u-238 elastic        3.3869E-03 +/- 2.8491E-06
```

(continues on next page)

(continued from previous page)

f-19 n,alpha	f-19 n,alpha	1.0496E-03 +/- 2.3439E-07	1.0000E+00 +/- 2.2331E-04
u-238 fission	u-238 fission	8.1042E-04 +/- 7.9610E-08	1.0000E+00 +/- 9.8233E-05
c elastic	c elastic	6.4522E-04 +/- 1.2905E-07	1.0000E+00 +/- 2.0001E-04
u-238 n,2n	u-238 n,2n	5.2362E-04 +/- 2.7904E-07	1.0000E+00 +/- 5.3290E-04

Example 6.3.9: Nuclide-reaction specific integral indices from TSUNAMI-IP output.

```
-----------------------------------------------
   NUCLIDE-REACTION SPECIFIC INTEGRAL INDICES
 -----------------------------------------------




     nuclide     reaction  sensitivity          g
   -------------  --------  -----------        ------
    h-1           capture    -1.02E-01         1.0000
    h-1           scatter     3.22E-01         1.0000
    c             scatter     2.48E-02         1.0000
    f-19          scatter     4.70E-02         1.0000
    u-235         fission     3.63E-01         1.0000
    u-235         capture    -1.13E-01         1.0000
    u-238         fission     3.35E-02         1.0000
    u-238         capture    -2.87E-01         1.0000
    u-238         scatter     4.88E-02         1.0000




     nuclide     reaction  sensitivity          g
   -------------  --------  -----------        ------
    h-1           capture    -1.02E-01         0.9971
    h-1           scatter     3.22E-01         0.9413
    c             scatter     2.48E-02         0.9491
    f-19          scatter     4.70E-02         0.9511
    u-235         fission     3.63E-01         0.9998
    u-235         capture    -1.13E-01         0.9967
    u-238         fission     3.35E-02         0.9978
    u-238         capture    -2.87E-01         0.9960
    u-238         scatter     4.88E-02         0.9773


     nuclide     reaction  sensitivity          g
   -------------  --------  -----------        ------
    h-1           capture    -1.02E-01         0.0000
    h-1           scatter     3.22E-01         0.0000
    c             scatter     2.48E-02         0.0000
    f-19          scatter     4.70E-02         0.0000
    u-235         fission     3.63E-01         0.0143
    u-235         capture    -1.13E-01         0.0072
    u-238         fission     3.35E-02         0.9996
    u-238         capture    -2.87E-01         0.0481
    u-238         scatter     4.88E-02         0.4112



 ------------------------------------------------------------------------------------------------------------------------
↪-------------
  Values for reactions with sensitivities greater than 1.00E-02 for application tsunami-3d_k5-1 with␣
↪experiment tsunami-
 ------------------------------------------------------------------------------------------------------------------------
↪-------------
     nuclide     reaction  sensitivity          g
   -------------  --------  -----------        ------
```

(continues on next page)

(continued from previous page)

```
   h-1           capture    -1.02E-01         0.9994
   h-1           scatter     3.22E-01         0.7755
   c             scatter     2.48E-02         0.0076
   f-19          scatter     4.70E-02         0.0000
   u-235         fission     3.63E-01         0.8622
   u-235         capture    -1.13E-01         0.9364
   u-238         fission     3.35E-02         0.7276
   u-238         capture    -2.87E-01         0.3955
   u-238         scatter     4.88E-02         0.4822



 ------------------------------------------------
  User requested g values for tsunami-3d_k5-1
 ------------------------------------------------

                   u-235    u-238    h-1      b-10
  Experiment       nubar    capture  total    capture
 ---------------- -------- -------- -------- --------
  tsunami-1d1       0.9991   0.9960   0.9445
  tsunami-1d4       0.0095   0.0481   0.0000
  tsunami-3d_k5-3   0.9403   0.3955   0.8433
```

Example 6.3.10: Extended penalty assessment from TSUNAMI-IP output.

```
-------------------------------------------------------------------------------------------------------------------------
↪--------
    Penalty Assessment for Application 1 tsunami-3d1
 ----------------------------------------------------------------------------------------------------------------------
↪---------

-------------------------------------------------------------------------------------------------------------------------
↪-----------------

   the standard deviation of keff due to uncovered sensitivity data is:

  9.747E-03 +/- 6.584E-05 ( % dk/k )
 ----------------------------------------------------------------------------------------------------------------------
↪------------------


   Input covariance file: 56groupcov7.1
 Working covariance file: tsunami-ip.wrk

    * indicates default covariance data

   contributions to uncertainty in keff ( % dk/k ) by individual energy covariance matrices:

                      covariance matrix
           nuclide-reaction    with    nuclide-reaction        % dk/k due to this matrix
        ----------------------      ----------------------   ----------------------------------
               u-238 n,n'                 u-238 n,n'           7.5820E-03 +/- 1.8458E-05
                 h-1 elastic                h-1 elastic         4.1794E-03 +/- 4.5264E-06
                f-19 elastic               f-19 elastic         3.9736E-03 +/- 1.0132E-05
               u-238 elastic              u-238 n,n'           -3.6600E-03 +/- 2.8032E-05
               u-235 chi                  u-235 chi             3.6588E-03 +/- 4.5806E-05
                f-19 n,n'                  f-19 n,n'           2.9919E-03 +/- 5.7578E-06
                f-19 elastic               f-19 n,n'           -2.9052E-03 +/- 3.0295E-05
               u-238 n,gamma              u-238 n,gamma         1.2407E-03 +/- 1.3961E-06
               u-238 elastic              u-238 elastic         1.1105E-03 +/- 9.2475E-07
               u-238 elastic              u-238 n,gamma         5.6491E-04 +/- 5.8032E-06
                   c elastic                  c elastic         4.8848E-04 +/- 1.0550E-07
               u-235 n,gamma              u-235 n,gamma         4.4181E-04 +/- 2.6790E-07
```

(continues on next page)

(continued from previous page)

```
                u-238 n,2n                u-238 n,2n           3.2664E-04 +/- 1.8609E-07
                u-238 chi                 u-238 chi            2.5281E-04 +/- 1.9474E-07
                u-235 nubar               u-235 nubar          1.7677E-04 +/- 2.0085E-07
                    c n,n'                    c elastic       -1.7595E-04 +/- 2.8179E-07
                    c n,n'                    c n,n'           1.7558E-04 +/- 7.7046E-08
                  h-1 n,gamma               h-1 n,gamma        1.4297E-04 +/- 2.6298E-08
                u-238 nubar               u-238 nubar          9.8421E-05 +/- 3.5167E-08
                 f-19 n,alpha              f-19 n,alpha        6.4564E-05 +/- 1.1052E-08
                u-238 elastic             u-238 n,2n          -4.7329E-05 +/- 2.0045E-08
                u-235 n,n'                u-235 n,n'           4.5469E-05 +/- 5.5758E-10
                 f-19 elastic              f-19 n,alpha        4.2446E-05 +/- 2.0604E-08
```

Example 6.3.11: Summary tables from TSUNAMI-IP output.

```
  ------------------------------
       c(k) SUMMARY TABLE
  ------------------------------




application # 1   tsunami-3d_k5-1 has 1 c(k) values >=  0.8000

1   tsunami-1d1          0.9999 +/-  0.0000

application # 2   tsunami-3d_k5-2 has 1 c(k) values >=  0.8000

1   tsunami-1d1          0.8175 +/-  0.0001

-----------------------------------------------------------------
Summary of Integral Values Meeting Acceptance Criteria
-----------------------------------------------------------------

Application         c(k) >=  0.8000
-----------------   ----------------
tsunami-3d_k5-1               1
tsunami-3d_k5-2               1

----------------
g  SUMMARY TABLE
----------------

application #1   tsunami-3d_k5-1

h-1 capture          ( sensitivity = -1.0173E-01 +/-  1.2526E-05 ) has 2 g values >=  0.7500
               1  tsunami-1d1       0.9971
               3  tsunami-3d_k5-3   0.9994
           h-1 scatter          ( sensitivity =  3.2202E-01 +/-  1.9772E-04 ) has 2 g values >=  0.7500
               1  tsunami-1d1       0.9413
               3  tsunami-3d_k5-3   0.7755
           c scatter            ( sensitivity =  2.4841E-02 +/-  1.4343E-05 ) has 1 g values >=  0.7500
               1  tsunami-1d1       0.9491
           f-19 scatter         ( sensitivity =  4.6979E-02 +/-  2.0034E-05 ) has 1 g values >=  0.7500
               1  tsunami-1d1       0.9511
           u-235 fission        ( sensitivity =  3.6286E-01 +/-  5.2950E-05 ) has 2 g values >=  0.7500
               1  tsunami-1d1       0.9998
               3  tsunami-3d_k5-3   0.8622
           u-235 capture        ( sensitivity = -1.1286E-01 +/-  1.2704E-05 ) has 2 g values >=  0.7500
               1  tsunami-1d1       0.9967
               3  tsunami-3d_k5-3   0.9364
           u-238 fission        ( sensitivity =  3.3498E-02 +/-  5.4895E-06 ) has 2 g values >=  0.7500
               1  tsunami-1d1       0.9978
               2  tsunami-1d4       0.9996
```

(continues on next page)

(continued from previous page)

```
          u-238 capture        ( sensitivity = -2.8728E-01 +-  2.1554E-05 ) has 1 g values >=  0.7500
              1  tsunami-1d1       0.9960

  application #2   tsunami-3d_k5-2
          h-1 capture          ( sensitivity = -1.9682E-01 +-  5.1773E-05 ) has 1 g values >=  0.7500
            3  tsunami-3d_k5-3   0.9921
          h-1 scatter          ( sensitivity =  3.5602E-01 +-  1.9469E-03 ) has 1 g values >=  0.7500
            3  tsunami-3d_k5-3   0.9160
          o-16 scatter         ( sensitivity =  6.9366E-02 +-  9.7179E-05 ) has 1 g values >=  0.7500
            3  tsunami-3d_k5-3   0.9229
          u-235 fission        ( sensitivity =  3.2845E-01 +-  1.3867E-04 ) has 2 g values >=  0.7500
            1  tsunami-1d1       0.9687
            3  tsunami-3d_k5-3   0.9751
          u-235 capture        ( sensitivity = -1.1817E-01 +-  3.5477E-05 ) has 2 g values >=  0.7500
            1  tsunami-1d1       0.9000
            3  tsunami-3d_k5-3   0.9841
          u-238 fission        ( sensitivity =  2.4663E-02 +-  9.8534E-06 ) has 3 g values >=  0.7500
            1  tsunami-1d1       1.0000
            2  tsunami-1d4       0.9998
            3  tsunami-3d_k5-3   0.9813
          u-238 capture        ( sensitivity = -1.1333E-01 +-  2.3953E-05 ) has 2 g values >=  0.7500
            1  tsunami-1d1       1.0000
            3  tsunami-3d_k5-3   0.9868

            ------------------------------
               SUMMARY OF NON-COVERAGE
            ------------------------------



------------------------------------------------------------------------------------------------------------------------
↪--------
Non-coverage for user-requested reactions for tsunami-3d_k5-1
------------------------------------------------------------------------------------------------------------------------
↪--------

     u-235            u-238            h-1              b-10
     nubar            capture          total            capture
---------------------------------- ---------------- ---------------- ---------------- ----------------
Number of groups not covered                    63              139               96
Sum of sensitivities not covered        3.9598E-04      -1.1226E-03       8.0684E-03
Group with largest sensitivity                 225              225              225
Largest sensitivity group value         2.0287E-01      -2.8184E-02      -2.1178E-02
Best experiment for max group          tsunami-1d1      tsunami-1d1  tsunami-3d_k5-3
Group sensitivity for best exp          2.0296E-01      -2.8154E-02      -3.8871E-02

------------------------------------------------------------------------------------------------------------------------
↪--------
Non-coverage for user-requested reactions for tsunami-3d_k5-2
------------------------------------------------------------------------------------------------------------------------
↪--------

     u-235            u-238            h-1              b-10
     nubar            capture          total            capture
---------------------------------- ---------------- ---------------- ---------------- ----------------
Number of groups not covered                    22                0               91
Sum of sensitivities not covered        3.7824E-03       0.0000E+00      -1.2898E-03
Group with largest sensitivity                 225              225              225
Largest sensitivity group value         1.9220E-01      -1.3369E-02      -4.6670E-02
Best experiment for max group          tsunami-1d1      tsunami-1d1  tsunami-3d_k5-3
Group sensitivity for best exp          2.0296E-01      -2.8154E-02      -3.8871E-02

------------------------------
     PENALTY SUMMARY TABLE
------------------------------
```

(continues on next page)

(continued from previous page)

```
Standard Deviation in Application Response Due to Uncovered Sensitivity Coefficients

Application    Type           Penalty          SD Units
-------------- -------- ---------------------- --------
tsunami-3d_k5-1  keff      9.7475E-3 +/- 6.5845E-5 % dk/k
tsunami-3d_k5-2  keff      2.3500E-1 +/- 1.2902E-4 % dk/k
```

Example 6.3.12: Composite sensitivity data from TSUNAMI-IP output.

```
------------------------------
COMPOSITE SENSITIVITY DATA
------------------------------



 ------------------------------------------------------------------
   Composite of user-requested reactions for tsunami-3d_k5-1
 ------------------------------------------------------------------

              u-235            u-238            h-1              b-10
  Group       nubar            capture          total            capture
 -------- ---------------- ---------------- ---------------- ----------------
     1       0.0000E+00       0.0000E+00       0.0000E+00
     2       8.3121E-08      -2.1269E-10      -4.8103E-07
     3       2.8391E-07      -1.1742E-09      -1.2188E-06
     4       5.1635E-07      -2.9305E-09      -1.4983E-06
     5       7.6669E-07      -6.5763E-09      -3.0912E-06
     6       1.2087E-05      -1.3860E-07      -4.2146E-05
     7       3.2436E-05      -2.5827E-07      -7.0385E-05
     8       9.4203E-05      -1.5178E-06       2.2338E-04
     9       1.7111E-04      -9.8460E-06       2.6160E-03
    10       9.5881E-05      -9.6314E-06       1.4498E-03
    11       4.6498E-04      -8.6324E-05       6.0348E-03
    12       3.3668E-04      -1.0881E-04       4.1605E-03
    13       1.0308E-04      -4.2404E-05       1.5022E-03
    14       4.9039E-04      -2.8588E-04       6.9219E-03
    15       4.1834E-04      -3.6981E-04       6.6199E-03
    16       1.3739E-04      -1.5075E-04       2.9036E-03
    17       6.1269E-05      -7.3358E-05       1.2879E-03
    18       5.3656E-05      -6.8243E-05       1.1029E-03
    19       9.3382E-05      -1.2782E-04       1.8069E-03
    20       7.2274E-05      -1.0611E-04       1.7937E-03
    21       1.6121E-04      -2.6852E-04       3.6259E-03
    22       1.5722E-04      -2.9405E-04       3.6598E-03
    23       1.6032E-04      -3.2036E-04       3.6221E-03
    24       3.6667E-05      -7.5670E-05       8.7492E-04
    25       4.6036E-05      -9.5868E-05       1.1263E-03
    26       2.4917E-05      -5.2550E-05       5.7950E-04
    27       7.3525E-05      -1.5597E-04       1.6697E-03
    28       1.3207E-04      -2.7919E-04       3.1426E-03
    29       1.4917E-04      -3.1499E-04       3.3212E-03
    30       1.9524E-05      -4.0694E-05       4.7128E-04
    31       1.5260E-04      -3.1113E-04       3.5947E-03
    32       5.9914E-05      -1.2004E-04       1.0086E-03
    33       5.4703E-05      -1.0849E-04       1.0609E-03
    34       1.2067E-04      -2.3813E-04       2.0801E-03
    35       7.4939E-05      -1.4661E-04       1.3644E-03
    36       7.5277E-05      -1.4645E-04       1.3422E-03
    37       4.5884E-05      -8.8788E-05       6.8414E-04
    38       5.1306E-05      -9.8727E-05       9.3662E-04
    39       1.9597E-04      -3.7831E-04       3.5664E-03
```

(continues on next page)

(continued from previous page)

40	1.8431E-04	-3.6387E-04	1.8026E-03
41	2.8555E-04	-5.5723E-04	4.2242E-03
42	2.6119E-04	-5.3510E-04	4.7957E-03
43	1.3711E-04	-2.9893E-04	1.8811E-03
44	1.9055E-04	-4.3852E-04	2.1220E-03
45	1.2409E-04	-3.1979E-04	1.1164E-03
46	3.6535E-05	-9.7941E-05	4.6072E-04
47	8.7249E-05	-2.3882E-04	8.8505E-04
48	2.5653E-05	-6.9180E-05	3.5130E-04
49	1.6739E-04	-5.0475E-04	2.2125E-03
50	1.1716E-04	-3.9760E-04	1.5983E-03
51	2.4526E-05	-9.6985E-05	1.5119E-04
52	6.4964E-05	-2.6803E-04	7.1621E-04
53	3.3839E-04	-1.4814E-03	4.7531E-03
54	1.3114E-04	-6.1020E-04	1.8245E-03
55	3.5173E-04	-1.7047E-03	5.1134E-03
56	2.4929E-04	-1.2514E-03	3.3044E-03
57	3.1251E-04	-1.5616E-03	3.2685E-03
58	1.7934E-04	-7.8059E-04	1.6846E-03
59	3.3768E-04	-1.5847E-03	3.0949E-03
60	5.7218E-04	-2.6453E-03	4.6208E-03
61	6.2408E-05	-2.1987E-04	2.8987E-04
62	3.4221E-04	-1.6321E-03	2.6850E-03
63	2.5176E-04	-1.0549E-03	1.7729E-03
64	2.1465E-04	-9.5589E-04	1.3821E-03
65	7.3062E-05	-3.8956E-04	5.2412E-04
66	3.7161E-04	-1.4107E-03	2.3877E-03
67	3.2093E-04	-1.3730E-03	1.6616E-03
68	5.8195E-05	-2.1647E-04	3.7317E-04
69	6.6370E-04	-2.5516E-03	3.5190E-03
70	4.5913E-04	-2.0723E-03	2.4579E-03
71	9.7426E-04	-4.1869E-03	4.9772E-03
72	1.0966E-04	-5.8274E-05	1.8589E-04
73	7.1828E-04	-2.6476E-03	2.7657E-03
74	2.4703E-03	-6.6484E-03	7.4726E-03
75	2.6381E-04	-1.3566E-03	1.1842E-03
76	1.1644E-03	-1.9283E-03	2.1884E-03
77	9.0059E-04	-1.9983E-03	2.0712E-03
78	1.3932E-05	-1.7300E-03	6.6140E-04
79	4.7174E-04	-3.7028E-04	5.4439E-04
80	1.2166E-04	-2.3242E-03	1.2190E-03
81	2.6867E-03	-3.1503E-03	5.5591E-03
82	1.7934E-04	-7.5046E-05	4.2063E-04
83	2.9797E-04	-3.1650E-03	1.6682E-03
84	1.8506E-04	-1.7962E-04	6.9532E-04
85	3.8625E-04	-5.4319E-03	3.0824E-03
86	6.6282E-04	-2.9944E-04	1.2789E-03
87	8.2978E-04	-4.5155E-04	1.1380E-03
88	1.1337E-04	-2.5098E-03	1.9443E-03
89	2.1511E-04	-6.1408E-05	6.1627E-04
90	5.9065E-04	-6.5874E-05	5.6950E-04
91	5.4779E-04	-3.0607E-04	1.0246E-03
92	3.7293E-05	-5.7959E-03	3.1081E-03
93	2.7815E-04	-6.1499E-04	8.7462E-04
94	2.5372E-04	-4.8609E-05	1.2161E-04
95	1.8329E-03	-8.5639E-05	4.8164E-04
96	3.9099E-04	-1.8614E-05	1.2128E-04
97	7.1230E-04	-1.9738E-05	1.5539E-04
98	2.4608E-04	-3.5551E-05	3.8414E-04
99	2.2019E-04	-1.5836E-05	1.7509E-04
100	3.1350E-04	-2.7103E-05	3.2313E-04
101	2.9855E-04	-6.8308E-05	5.6510E-04
102	3.5827E-04	-5.9559E-05	4.5259E-04
103	2.1822E-04	-1.0055E-04	8.6367E-04
104	3.8254E-04	-1.3022E-04	7.8610E-04

(continues on next page)

(continued from previous page)

105	2.6271E-04	-2.6041E-04	7.6712E-04
106	4.5665E-04	-1.6304E-04	3.4290E-04
107	1.3250E-04	-7.8605E-04	1.3860E-03
108	5.8291E-05	-3.1200E-03	3.3141E-03
109	2.5917E-04	-9.6788E-03	5.0386E-03
110	1.5727E-03	-9.5678E-04	5.9505E-04
111	5.1745E-04	-3.9567E-04	5.1849E-04
112	3.3689E-04	-1.4825E-04	2.9832E-04
113	5.5476E-04	-2.9394E-04	5.5464E-04
114	2.2143E-05	-6.9191E-05	1.6138E-04
115	1.9064E-04	-1.4094E-04	5.3270E-04
116	5.0989E-04	-2.2832E-04	1.2496E-03
117	1.1009E-03	-3.0055E-04	1.6754E-03
118	1.4392E-03	-8.5266E-04	3.3368E-03
119	2.4313E-04	-7.0828E-03	6.5606E-03
120	1.7047E-04	-8.4025E-03	5.1957E-03
121	2.2684E-03	-1.6808E-03	1.4346E-03
122	2.0681E-04	-3.2469E-04	4.1713E-04
123	5.3283E-04	-5.4572E-04	9.2483E-04
124	5.6983E-04	-2.1805E-04	5.6677E-04
125	3.4396E-04	-1.5890E-04	4.4437E-04
126	2.2707E-04	-1.1013E-04	4.5908E-04
127	8.3158E-04	-9.6956E-05	1.8533E-04
128	4.5995E-04	-1.2467E-04	4.7548E-04
129	1.7971E-03	-1.4955E-04	8.4077E-04
130	3.2110E-04	-6.5461E-05	9.1569E-04
131	6.1328E-04	-8.3373E-04	1.7860E-03
132	1.2347E-03	-3.2726E-04	1.4198E-03
133	4.8042E-03	-7.7205E-04	2.9930E-03
134	1.9295E-04	-3.7846E-03	8.5681E-03
135	3.2715E-04	-2.2262E-03	2.7745E-03
136	6.6065E-05	-7.5091E-03	6.8661E-03
137	1.3387E-05	-3.9274E-03	3.1875E-03
138	5.5488E-04	-6.5404E-03	5.8490E-03
139	3.2411E-04	-2.8026E-03	2.5508E-03
140	3.8712E-04	-2.8312E-03	2.6262E-03
141	1.4925E-04	-9.6337E-04	1.3378E-03
142	1.4422E-04	-4.5295E-04	1.1235E-03
143	1.5490E-04	-1.2351E-03	2.1938E-03
144	6.4744E-05	-3.3445E-04	7.5336E-04
145	8.2229E-04	-2.7165E-04	6.6269E-04
146	6.7978E-04	-4.1042E-04	7.9458E-04
147	3.2938E-04	-1.1761E-04	1.9318E-04
148	1.0859E-04	-5.9175E-05	1.3243E-04
149	5.9627E-05	-3.7701E-05	9.7112E-05
150	1.7015E-04	-1.2409E-04	2.8871E-04
151	1.3042E-04	-1.2721E-04	3.4573E-04
152	7.9303E-05	-1.3007E-04	3.5622E-04
153	8.0252E-05	-1.3289E-04	3.9765E-04
154	9.3218E-05	-1.3512E-04	3.3495E-04
155	9.3312E-05	-1.2361E-04	2.5897E-04
156	9.2711E-05	-1.1448E-04	3.3308E-04
157	1.1417E-04	-1.3297E-04	3.5854E-04
158	1.2864E-04	-1.3720E-04	3.7838E-04
159	3.0061E-04	-1.9035E-04	6.5148E-04
160	1.2085E-04	-9.9337E-05	3.2785E-04
161	1.4351E-04	-1.3909E-04	3.1532E-04
162	1.6682E-04	-1.6273E-04	4.2556E-04
163	1.8495E-04	-1.7393E-04	3.0967E-04
164	2.0345E-04	-1.8509E-04	4.6764E-04
165	2.2132E-04	-1.9666E-04	4.2683E-04
166	1.3542E-04	-1.1483E-04	2.8152E-04
167	1.5182E-04	-1.2110E-04	3.3331E-04
168	1.7660E-04	-1.2685E-04	2.7533E-04
169	2.1245E-04	-1.3208E-04	3.0394E-04

(continues on next page)

(continued from previous page)

170	2.6320E-04	-1.3841E-04	2.7320E-04
171	1.9419E-04	-7.1841E-05	1.6409E-04
172	2.7519E-04	-7.3711E-05	1.4999E-04
173	3.8550E-04	-7.5674E-05	1.4648E-04
174	5.0044E-04	-7.7661E-05	8.4438E-05
175	2.2717E-04	-3.1699E-05	3.8770E-05
176	2.2913E-04	-3.1410E-05	2.9601E-05
177	2.3709E-04	-3.2293E-05	4.2854E-05
178	2.4257E-04	-3.2709E-05	3.4193E-05
179	2.3003E-04	-3.2775E-05	2.9616E-05
180	2.2349E-04	-3.2936E-05	3.7056E-05
181	2.2113E-04	-3.3917E-05	3.6280E-05
182	2.1354E-04	-3.4383E-05	3.6401E-05
183	2.0719E-04	-3.4859E-05	5.0084E-05
184	2.0168E-04	-3.5283E-05	4.0429E-05
185	1.9414E-04	-3.5733E-05	5.6482E-05
186	1.8530E-04	-3.6287E-05	5.3350E-05
187	1.8060E-04	-3.6876E-05	4.6883E-05
188	1.7542E-04	-3.7408E-05	4.8868E-05
189	1.7875E-04	-3.8136E-05	1.7370E-05
190	4.1786E-04	-9.7698E-05	1.3305E-04
191	4.0193E-04	-1.0043E-04	1.4039E-04
192	4.0255E-04	-1.0398E-04	1.5921E-04
193	3.9780E-04	-1.0814E-04	1.2778E-04
194	8.0388E-04	-2.3098E-04	2.8322E-04
195	8.5863E-04	-2.5171E-04	1.3920E-04
196	9.2365E-04	-2.7401E-04	3.7939E-04
197	1.0456E-03	-3.0662E-04	4.0857E-04
198	1.1698E-03	-3.3550E-04	4.2186E-04
199	6.4963E-04	-1.8310E-04	2.6318E-04
200	7.1981E-04	-1.9880E-04	2.2883E-04
201	1.6158E-03	-4.2613E-04	3.6152E-04
202	2.0254E-03	-4.9966E-04	2.8211E-04
203	2.4707E-03	-5.6340E-04	2.8843E-04
204	3.3624E-03	-6.7630E-04	2.8969E-04
205	2.1844E-03	-3.8558E-04	-8.4664E-05
206	2.7103E-03	-4.3373E-04	1.1227E-04
207	3.2507E-03	-4.6733E-04	0.0000E+00
208	4.1765E-03	-5.5377E-04	1.9915E-05
209	4.6469E-03	-6.0428E-04	-6.5711E-05
210	5.4292E-03	-7.4815E-04	-5.8270E-05
211	5.9319E-03	-8.9850E-04	-4.1482E-05
212	6.8645E-03	-1.1352E-03	3.6827E-06
213	9.0697E-03	-1.5740E-03	9.5084E-05
214	1.3105E-02	-2.3116E-03	-6.2132E-04
215	2.0944E-02	-3.6390E-03	0.0000E+00
216	3.7206E-02	-6.2563E-03	-2.7238E-03
217	2.2374E-02	-3.6629E-03	-1.8224E-03
218	2.9838E-02	-4.7756E-03	-3.1442E-03
219	3.8551E-02	-6.0551E-03	-2.1233E-03
220	4.9816E-02	-7.6542E-03	-4.8489E-03
221	6.4973E-02	-9.7526E-03	-4.3945E-03
222	8.2484E-02	-1.2128E-02	-8.4682E-03
223	1.0354E-01	-1.4855E-02	-9.5965E-03
224	5.5503E-02	-7.8609E-03	-5.7398E-03
225	2.0287E-01	-2.8154E-02	-2.1178E-02
226	3.2324E-02	-4.4137E-03	-3.9011E-03
227	2.9803E-02	-4.0284E-03	-3.8972E-03
228	1.0735E-02	-1.4428E-03	-1.2831E-03
229	9.7841E-03	-1.3106E-03	-1.2401E-03
230	4.4405E-03	-5.9294E-04	-4.6629E-04
231	4.0564E-03	-5.4338E-04	-4.6124E-04
232	3.6568E-03	-4.8680E-04	-4.1461E-04
233	1.9509E-03	-2.5916E-04	-2.2200E-04
234	1.1574E-03	-1.5422E-04	-1.3472E-04

(continues on next page)

(continued from previous page)

```
  235       1.3127E-03      -1.7411E-04      -1.4945E-04
  236       1.1070E-03      -1.4705E-04      -1.1858E-04
  237       1.2172E-03      -1.6145E-04      -1.3854E-04
  238       1.0399E-04      -1.3798E-05      -1.1822E-05

------------------------------------------------------------------
  Composite of user-requested reactions for tsunami-3d_k5-2
------------------------------------------------------------------

            u-235           u-238           h-1             b-10
 Group      nubar           capture         total           capture
-------- ---------------- ---------------- ---------------- ----------------
    1       0.0000E+00       0.0000E+00       0.0000E+00
    2       1.2519E-07      -8.6672E-11       0.0000E+00
    3       3.3421E-07      -4.3407E-10       4.2389E-07
    4       7.2762E-07      -1.3902E-09       6.8757E-07
    5       1.3021E-06      -3.8278E-09       2.5489E-06
    6       1.4922E-05      -5.6904E-08       4.6252E-05
    7       4.7469E-05      -1.3730E-07       2.0214E-04
    8       1.4446E-04      -8.9470E-07       1.0914E-03
    9       2.8286E-04      -6.6636E-06       4.8035E-03
   10       1.5596E-04      -6.5378E-06       2.9555E-03
   11       7.5246E-04      -5.9712E-05       1.3113E-02
   12       5.8627E-04      -7.9447E-05       1.2557E-02
   13       1.7542E-04      -3.0813E-05       4.0300E-03
   14       7.3736E-04      -1.8899E-04       1.5889E-02
   15       6.2601E-04      -2.4990E-04       1.3073E-02
   16       2.0457E-04      -1.0331E-04       4.7828E-03
   17       9.0359E-05      -4.9631E-05       2.1468E-03
   18       7.3377E-05      -4.3188E-05       1.6032E-03
   19       1.3573E-04      -8.6428E-05       2.6278E-03
   20       1.1161E-04      -7.6638E-05       2.4900E-03
   21       2.2033E-04      -1.7115E-04       4.9231E-03
   22       1.8239E-04      -1.6205E-04       3.4382E-03
   23       1.9578E-04      -1.8691E-04       3.6221E-03
   24       4.7147E-05      -4.5670E-05       1.1450E-03
   25       6.3898E-05      -6.1821E-05       1.6513E-03
   26       3.7065E-05      -3.6333E-05       9.5190E-04
   27       1.1856E-04      -1.1649E-04       3.1639E-03
   28       2.2510E-04      -2.2146E-04       5.6953E-03
   29       2.1545E-04      -2.1077E-04       6.2475E-03
   30       2.7193E-05      -2.6428E-05       7.8781E-04
   31       2.1388E-04      -2.0401E-04       6.4215E-03
   32       8.3295E-05      -7.7147E-05       2.3947E-03
   33       7.1583E-05      -6.6441E-05       2.0713E-03
   34       1.5905E-04      -1.4732E-04       4.5168E-03
   35       9.2541E-05      -8.5375E-05       2.5578E-03
   36       8.1119E-05      -7.5870E-05       1.5017E-03
   37       4.9873E-05      -4.6393E-05       6.8414E-04
   38       6.3817E-05      -5.9216E-05       9.7337E-04
   39       2.7366E-04      -2.5218E-04       6.3032E-03
   40       2.7217E-04      -2.5472E-04       6.6936E-03
   41       3.4925E-04      -3.2612E-04       8.1596E-03
   42       2.8678E-04      -2.8720E-04       6.4855E-03
   43       1.4163E-04      -1.4926E-04       3.3112E-03
   44       2.1400E-04      -2.3936E-04       4.3949E-03
   45       1.2424E-04      -1.5558E-04       3.0408E-03
   46       2.7340E-05      -3.5790E-05       5.9417E-04
   47       7.2215E-05      -9.6257E-05       1.5233E-03
   48       2.2974E-05      -3.0302E-05       4.7674E-04
   49       1.5246E-04      -2.2541E-04       2.9677E-03
   50       1.1097E-04      -1.8360E-04       1.7649E-03
   51       2.8048E-05      -5.3701E-05       5.0807E-04
   52       7.6335E-05      -1.5167E-04       1.3984E-03
   53       2.9500E-04      -6.2994E-04       5.4989E-03
```

(continues on next page)

(continued from previous page)

54	1.3718E-04	-3.1099E-04	1.9844E-03
55	3.0486E-04	-7.2088E-04	5.0718E-03
56	2.2166E-04	-5.3923E-04	2.8143E-03
57	2.8217E-04	-6.7714E-04	3.5159E-03
58	1.6164E-04	-3.3247E-04	2.1026E-03
59	3.0000E-04	-6.5640E-04	3.1621E-03
60	5.0678E-04	-1.0792E-03	4.8527E-03
61	5.7681E-05	-8.7342E-05	4.7504E-04
62	3.0594E-04	-6.2686E-04	2.0945E-03
63	2.2535E-04	-4.0760E-04	1.7729E-03
64	1.9245E-04	-3.5872E-04	1.4543E-03
65	6.6674E-05	-1.3797E-04	4.3374E-04
66	3.3064E-04	-5.1874E-04	2.2962E-03
67	2.9095E-04	-4.5419E-04	1.8807E-03
68	5.1987E-05	-7.5088E-05	3.1526E-04
69	5.8854E-04	-8.5478E-04	3.0208E-03
70	4.1099E-04	-7.1338E-04	1.7490E-03
71	8.5819E-04	-1.3405E-03	4.3646E-03
72	9.7106E-05	-2.5341E-05	2.1843E-04
73	6.3698E-04	-7.7299E-04	2.5855E-03
74	2.1654E-03	-2.0489E-03	6.4043E-03
75	2.4547E-04	-3.7235E-04	6.9292E-04
76	1.0286E-03	-5.5024E-04	1.9443E-03
77	7.9496E-04	-5.2036E-04	2.0518E-03
78	1.1633E-05	-4.1199E-04	4.3688E-04
79	4.1893E-04	-1.6219E-04	8.4874E-04
80	1.0671E-04	-6.8021E-04	7.8182E-04
81	2.3203E-03	-9.9401E-04	4.6543E-03
82	1.5328E-04	-3.1823E-05	3.2999E-04
83	2.7539E-04	-8.2324E-04	1.0520E-03
84	1.6432E-04	-7.7786E-05	6.4772E-04
85	3.3128E-04	-1.4793E-03	1.9299E-03
86	5.8071E-04	-1.2966E-04	9.6470E-04
87	7.2897E-04	-1.8707E-04	1.0648E-03
88	1.0165E-04	-6.0134E-04	9.5099E-04
89	1.9705E-04	-2.7280E-05	5.4711E-04
90	5.1470E-04	-2.8447E-05	4.2588E-04
91	4.7360E-04	-1.2936E-04	8.4947E-04
92	2.6906E-05	-1.3329E-03	1.8984E-03
93	2.5613E-04	-2.6531E-04	6.5075E-04
94	2.3538E-04	-2.1992E-05	2.0144E-04
95	1.5775E-03	-3.7759E-05	5.4441E-04
96	3.3280E-04	-7.9568E-06	1.7036E-04
97	5.9346E-04	-8.3159E-06	6.5965E-05
98	2.2511E-04	-1.5893E-05	3.1849E-04
99	2.0225E-04	-7.0445E-06	1.9922E-04
100	2.7342E-04	-1.1609E-05	3.2313E-04
101	2.6957E-04	-3.0399E-05	4.6923E-04
102	3.3244E-04	-2.7060E-05	3.2659E-04
103	1.9461E-04	-4.2923E-05	5.9301E-04
104	3.2556E-04	-5.2598E-05	5.4021E-04
105	2.3730E-04	-1.1291E-04	5.1509E-04
106	3.4607E-04	-5.8981E-05	2.2766E-04
107	1.1425E-04	-2.9850E-04	7.9441E-04
108	4.1191E-05	-6.7330E-04	1.4840E-03
109	1.9773E-04	-1.6031E-03	2.7789E-03
110	1.1533E-03	-3.4513E-04	2.8717E-04
111	4.0019E-04	-1.5092E-04	4.0941E-04
112	2.7434E-04	-5.7168E-05	2.7772E-04
113	4.5490E-04	-1.2471E-04	4.5389E-04
114	2.0751E-05	-3.1118E-05	1.8411E-04
115	1.7034E-04	-5.9649E-05	5.3270E-04
116	4.5462E-04	-9.8437E-05	8.1464E-04
117	1.0112E-03	-1.3308E-04	9.1834E-04
118	1.2441E-03	-3.4617E-04	1.8797E-03

(continues on next page)

(continued from previous page)

119	1.4613E-04	-1.7858E-03	3.0512E-03
120	1.1927E-04	-1.4793E-03	3.4795E-03
121	1.6146E-03	-6.3939E-04	8.1885E-04
122	2.0288E-04	-1.4691E-04	2.7061E-04
123	4.9382E-04	-2.3581E-04	9.1055E-04
124	5.0753E-04	-9.0473E-05	4.6140E-04
125	3.2433E-04	-7.0258E-05	5.3142E-04
126	2.1587E-04	-4.8428E-05	4.5908E-04
127	7.7506E-04	-4.2735E-05	1.8533E-04
128	4.4479E-04	-5.6181E-05	4.7548E-04
129	1.3698E-03	-5.9501E-05	1.0153E-03
130	2.7637E-04	-2.5432E-05	7.1524E-04
131	5.7207E-04	-3.5591E-04	1.2909E-03
132	1.1350E-03	-1.4174E-04	7.8511E-04
133	3.6109E-03	-3.1864E-04	1.2579E-03
134	1.7268E-04	-1.4722E-03	3.9073E-03
135	2.5746E-04	-6.2532E-04	1.1511E-03
136	4.3943E-05	-1.0605E-03	2.9541E-03
137	5.7508E-06	-2.4973E-04	3.3501E-03
138	3.7473E-04	-1.2848E-03	2.3995E-03
139	2.8865E-04	-9.6386E-04	1.2706E-03
140	3.6819E-04	-1.1046E-03	2.0471E-03
141	1.4780E-04	-3.9461E-04	1.3134E-03
142	1.3985E-04	-1.8380E-04	1.0559E-03
143	1.5490E-04	-5.5020E-04	1.4747E-03
144	6.4744E-05	-1.5547E-04	7.5336E-04
145	8.1785E-04	-1.1830E-04	6.1400E-04
146	6.7829E-04	-1.7972E-04	9.0948E-04
147	3.2822E-04	-5.0617E-05	2.4536E-04
148	1.0819E-04	-2.5691E-05	1.1098E-04
149	6.4077E-05	-1.7786E-05	9.2894E-05
150	1.7080E-04	-5.5107E-05	2.8871E-04
151	1.3528E-04	-5.7736E-05	3.6300E-04
152	7.9303E-05	-5.8377E-05	3.3702E-04
153	8.0053E-05	-5.7765E-05	2.9098E-04
154	9.4301E-05	-6.1205E-05	3.5845E-04
155	9.5100E-05	-5.5160E-05	3.1008E-04
156	9.2093E-05	-4.9272E-05	3.3308E-04
157	1.1417E-04	-6.0554E-05	3.5854E-04
158	1.2913E-04	-6.2998E-05	3.6015E-04
159	3.0061E-04	-8.2329E-05	7.1490E-04
160	1.2367E-04	-4.5006E-05	2.9688E-04
161	1.4840E-04	-6.7230E-05	3.8590E-04
162	1.7126E-04	-7.3391E-05	4.2556E-04
163	1.8465E-04	-7.6909E-05	4.4451E-04
164	2.0831E-04	-8.3782E-05	4.5114E-04
165	2.2843E-04	-8.8648E-05	4.6432E-04
166	1.4190E-04	-5.2854E-05	2.4605E-04
167	1.5012E-04	-5.2102E-05	3.1876E-04
168	1.8061E-04	-5.6757E-05	3.1259E-04
169	2.1245E-04	-6.0593E-05	3.0542E-04
170	2.6343E-04	-6.3934E-05	3.4215E-04
171	1.9448E-04	-3.1752E-05	1.3489E-04
172	2.6013E-04	-3.0902E-05	1.1947E-04
173	3.7054E-04	-3.2340E-05	1.2321E-04
174	4.7482E-04	-3.2632E-05	1.1158E-04
175	2.1975E-04	-1.3672E-05	3.2459E-05
176	2.1632E-04	-1.3213E-05	4.0032E-05
177	2.2152E-04	-1.3437E-05	4.5984E-05
178	2.3849E-04	-1.4859E-05	1.8672E-05
179	2.1745E-04	-1.4007E-05	2.7954E-05
180	2.2646E-04	-1.4890E-05	3.7056E-05
181	2.1143E-04	-1.4534E-05	3.4536E-05
182	2.0606E-04	-1.4947E-05	6.3531E-05
183	1.9077E-04	-1.4574E-05	5.0864E-05

(continues on next page)

(continued from previous page)

```
   184       1.9528E-04      -1.5566E-05       4.4139E-05
   185       1.8839E-04      -1.5884E-05       4.6885E-05
   186       1.8455E-04      -1.6210E-05       4.8693E-05
   187       1.8045E-04      -1.6486E-05       1.6260E-05
   188       1.7154E-04      -1.6166E-05       4.8868E-05
   189       1.7634E-04      -1.7504E-05       2.2074E-05
   190       4.1297E-04      -4.3034E-05       1.3305E-04
   191       4.0310E-04      -4.5888E-05       1.4039E-04
   192       3.9172E-04      -4.5833E-05       1.5921E-04
   193       3.8144E-04      -4.6490E-05       1.0588E-04
   194       8.3313E-04      -1.0632E-04       3.3357E-04
   195       8.7545E-04      -1.1477E-04       2.7002E-04
   196       9.0294E-04      -1.1976E-04       3.7939E-04
   197       1.0334E-03      -1.3672E-04       3.6811E-04
   198       1.1556E-03      -1.4920E-04       4.2186E-04
   199       6.6467E-04      -8.4875E-05       1.4783E-04
   200       7.0922E-04      -8.7084E-05       2.2883E-04
   201       1.6056E-03      -1.9185E-04       4.5221E-04
   202       1.9989E-03      -2.2347E-04       4.2581E-04
   203       2.4690E-03      -2.5230E-04       4.1739E-04
   204       3.3101E-03      -3.0014E-04       0.0000E+00
   205       2.0584E-03      -1.6521E-04       1.0675E-04
   206       2.5065E-03      -1.8312E-04       0.0000E+00
   207       2.9577E-03      -1.9366E-04       0.0000E+00
   208       3.8129E-03      -2.3258E-04      -1.4373E-04
   209       4.2586E-03      -2.5331E-04       9.6694E-06
   210       4.8357E-03      -3.0345E-04      -1.5320E-05
   211       5.4913E-03      -3.8129E-04      -2.2865E-04
   212       6.4549E-03      -4.8716E-04      -3.0036E-04
   213       8.8549E-03      -7.0184E-04       0.0000E+00
   214       1.3538E-02      -1.0984E-03      -2.0716E-03
   215       2.2933E-02      -1.8610E-03      -4.7201E-03
   216       4.2566E-02      -3.3462E-03      -6.9812E-03
   217       2.5852E-02      -1.9896E-03      -4.0796E-03
   218       3.4808E-02      -2.6485E-03      -8.1183E-03
   219       4.5763E-02      -3.4270E-03      -6.2337E-03
   220       5.8912E-02      -4.3180E-03      -1.0982E-02
   221       7.5765E-02      -5.4456E-03      -1.7604E-02
   222       9.4178E-02      -6.7228E-03      -1.9553E-02
   223       1.1300E-01      -8.0852E-03      -2.5022E-02
   224       5.6986E-02      -4.0496E-03      -1.1570E-02
   225       1.9220E-01      -1.3369E-02      -3.8871E-02
   226       2.5046E-02      -1.7355E-03      -5.2696E-03
   227       2.0587E-02      -1.4265E-03      -3.8972E-03
   228       6.7126E-03      -4.6879E-04      -1.5915E-03
   229       5.4068E-03      -3.7844E-04      -8.8839E-04
   230       2.3059E-03      -1.6176E-04      -4.7061E-04
   231       1.8324E-03      -1.2931E-04      -3.5961E-04
   232       1.5083E-03      -1.0623E-04      -3.3234E-04
   233       7.3805E-04      -5.2455E-05      -8.8918E-05
   234       4.2877E-04      -3.0590E-05      -8.5943E-05
   235       3.7055E-04      -2.6389E-05      -6.6216E-05
   236       2.8675E-04      -2.0539E-05      -4.1094E-05
   237       2.4347E-04      -1.7796E-05      -4.2897E-05
   238       2.3790E-05      -1.7598E-06       0.0000E+00


 Producing composite datafile:

          6 profiles

-------------------------------------------------------------------------------------------
TSUNAMI-IP Execution Complete
-------------------------------------------------------------------------------------------
```

HTML output description

The HTML formatted output from the TSUNAMI-IP sample problem is described in this section. The HTML output is generated when the *html* keyword is included in the *PARAMETER* block keyword The input for this sample problem is named tsunami-ip.input. In this case, the HTML formatted output is stored in a file called tsunami-ip.html and additional resources are stored in directories called tsunami-ip.htmd and applet_resources. These sections contain example TSUNAMI-IP HTML formatted output only for demonstration of the interface. When tsunami-ip.html is opened in a web browser, the information shown in Fig. 6.3.4 is displayed. The title of the input file is displayed between the two SCALE logos. Because this SCALE input file only executed tsunami-ip, only a single output listing is available. The text "1. TSUNAMI-IP" is a hyperlink to view the output from TSUNAMI-IP. Clicking on the "1. TSUNAMI-IP" hyperlink will present the information shown in Fig. 6.3.5

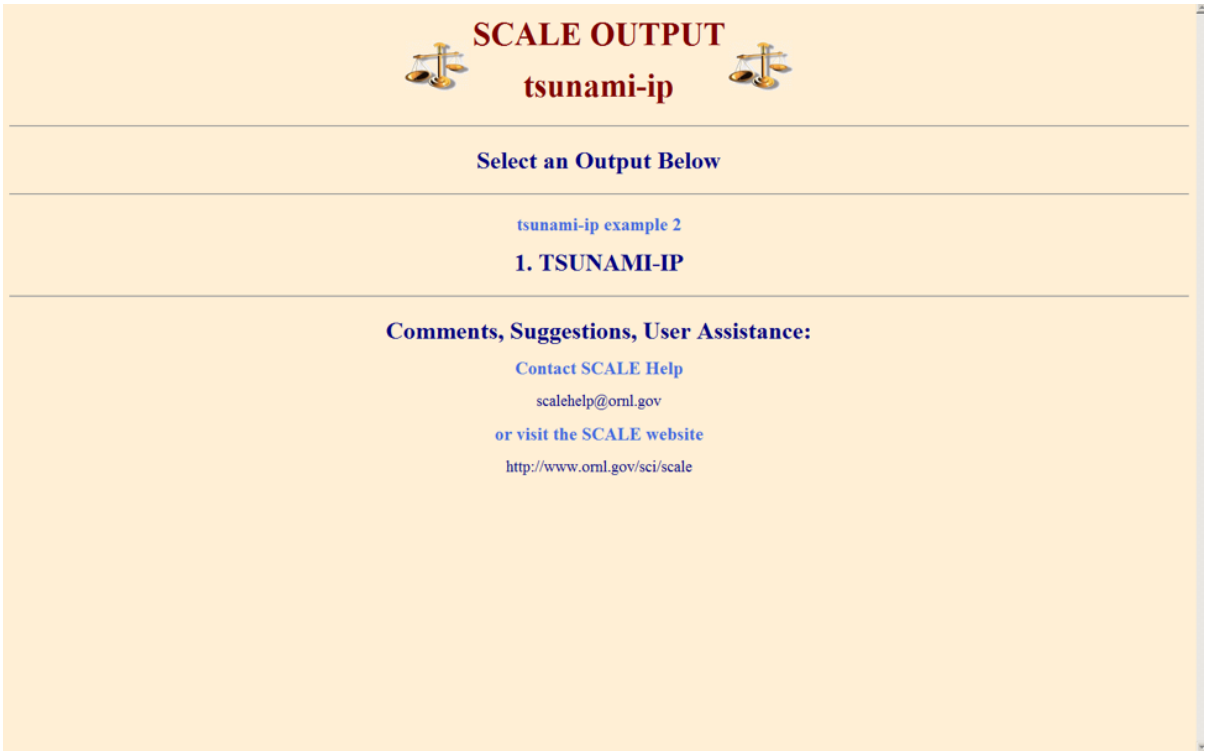


Fig. 6.3.4: Initial screen from TSUNAMI-IP HTML output.

The initial page of output from TSUNAMI-IP is shown in Fig. 6.3.5. Program verification information is shown in the table under the TSUNAMI logo. This table includes information about the code that was executed and the date and time it was run. The menu on the left side of the screen contains hyperlinks to specific portions of the code output. Echoes of the input data are available in the Input Data section. Any errors or warning messages are available in the Messages sections. Results from the code execution are shown in the results section.

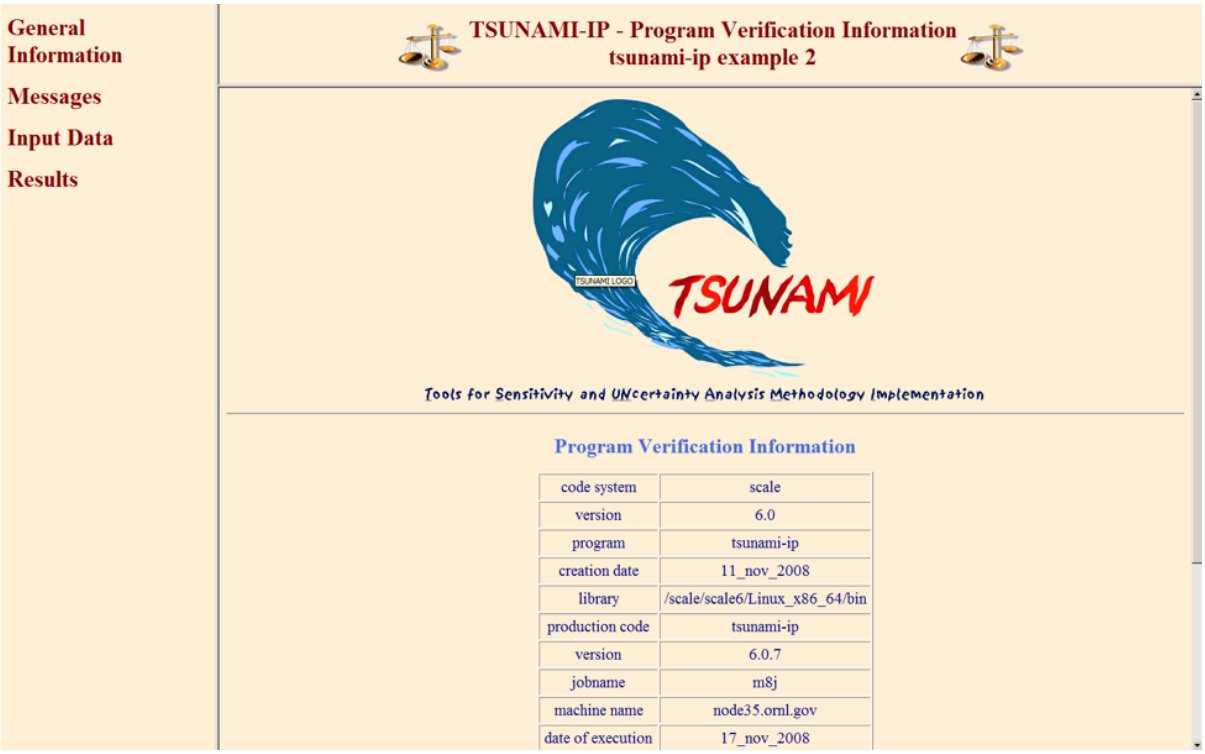
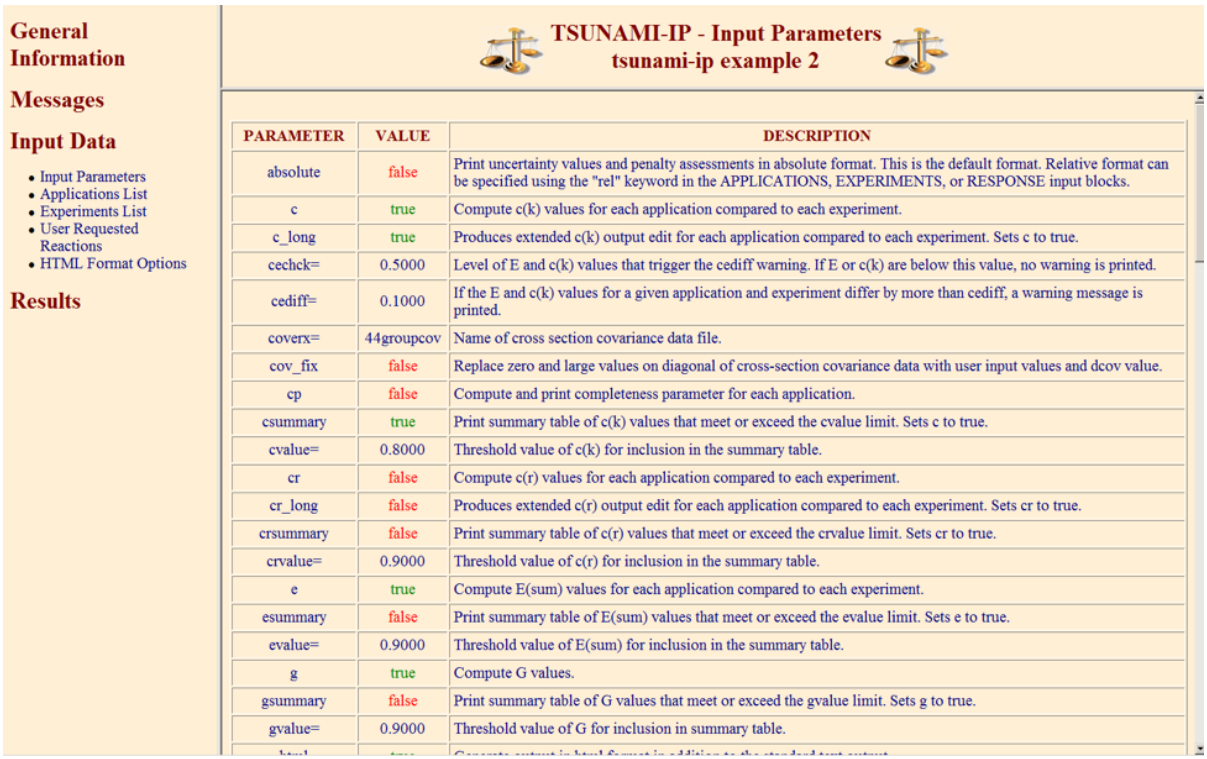


Fig. 6.3.5: Program verification screen from TSUNAMI-IP HTML output.

Selecting Input Parameters will reveal the menu of available input data. Selecting Input Parameters causes the table shown in Fig. 6.3.6 to be displayed. Other input data can also be displayed by selecting the desired data from the menu.



PARAMETER	VALUE	DESCRIPTION
absolute	false	Print uncertainty values and penalty assessments in absolute format. This is the default format. Relative format can be specified using the "rel" keyword in the APPLICATIONS, EXPERIMENTS, or RESPONSE input blocks.
c	true	Compute c(k) values for each application compared to each experiment.
c_long	true	Produces extended c(k) output edit for each application compared to each experiment. Sets c to true.
cechck=	0.5000	Level of E and c(k) values that trigger the cediff warning. If E or c(k) are below this value, no warning is printed.
cediff=	0.1000	If the E and c(k) values for a given application and experiment differ by more than cediff, a warning message is printed.
coverx=	44groupcov	Name of cross section covariance data file.
cov_fix	false	Replace zero and large values on diagonal of cross-section covariance data with user input values and dcov value.
cp	false	Compute and print completeness parameter for each application.
csummary	true	Print summary table of c(k) values that meet or exceed the cvalue limit. Sets c to true.
cvalue=	0.8000	Threshold value of c(k) for inclusion in the summary table.
cr	false	Compute c(r) values for each application compared to each experiment.
cr_long	false	Produces extended c(r) output edit for each application compared to each experiment. Sets cr to true.
crsummary	false	Print summary table of c(r) values that meet or exceed the crvalue limit. Sets cr to true.
crvalue=	0.9000	Threshold value of c(r) for inclusion in the summary table.
e	true	Compute E(sum) values for each application compared to each experiment.
esummary	false	Print summary table of E(sum) values that meet or exceed the evalue limit. Sets e to true.
evalue=	0.9000	Threshold value of E(sum) for inclusion in the summary table.
g	true	Compute G values.
gsummary	false	Print summary table of G values that meet or exceed the gvalue limit. Sets g to true.
gvalue=	0.9000	Threshold value of G for inclusion in summary table.

Fig. 6.3.6: Input parameters from TSUNAMI-IP HTML output.

Selecting Messages will reveal a menu of available messages. Selecting Warning Messages from the Messages section of the menu causes the information shown in Fig. 6.3.7 to appear. The Warning Messages edit contains all warning messages that were generated during the execution of the code. If errors were encountered in the code execution, an Error Messages item would have also been available in the menu under Messages.

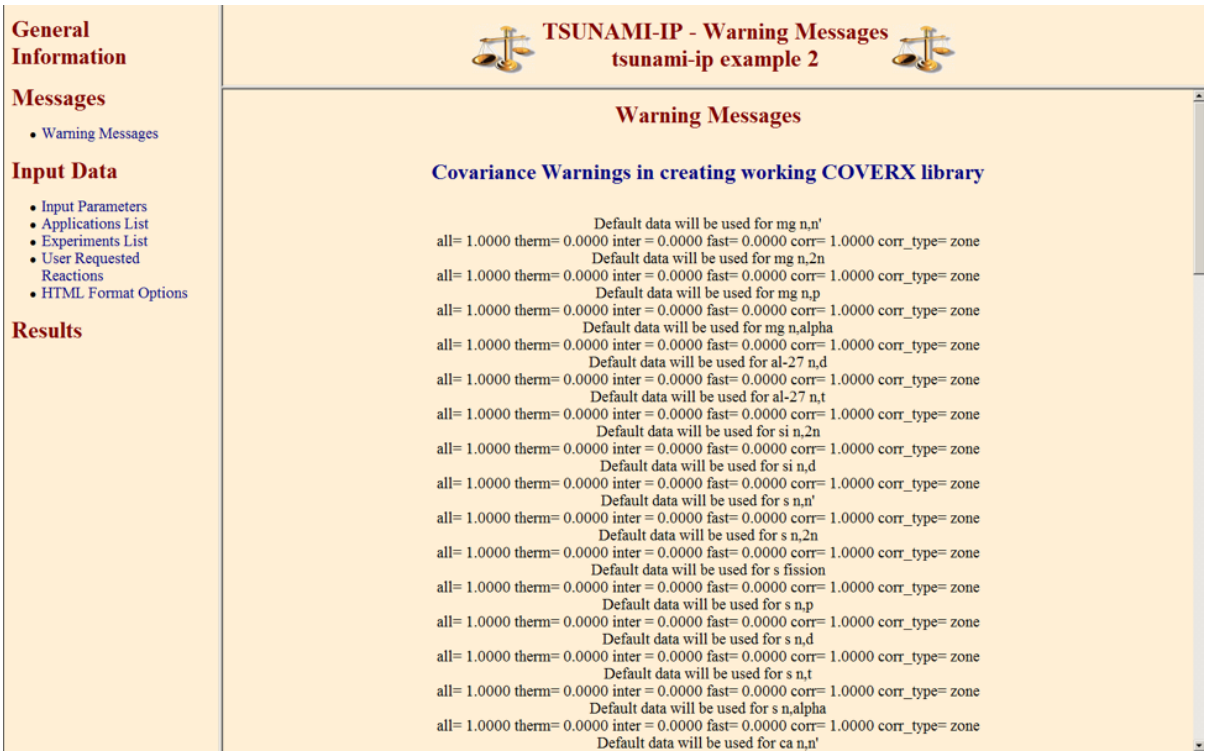


Fig. 6.3.7: Warning messages from TSUNAMI-IP HTML output.

Selecting Results causes a menu of available results to be revealed. From this menu, selecting Global Integral Indices causes a submenu of available global integral indices to be revealed. Selecting Integral Index Values from this submenu causes the information shown in Fig. 6.3.8 to appear. In certain results edits, a second menu appears on the right such that the information for a particular application can be quickly reached. The table shown in Fig. 6.3.8 corresponds to the *values* table from the standard output file. One advantage of the HTML output is the use of color coding. Values exceeding the cutoff value for a particular integral index are printed in the color that is set by the *HMTL* data block input *cut_clr*=. The maximum value for each index for each application is printed in the color that is set by the *HMTL* data block input *max_clr*=. In this case, the values that exceed the cutoff values are also the maximum values and are colored as such. Edits for each type of data requested in the *PARAMETER* data block are available in the results section by selecting the appropriate submenu.

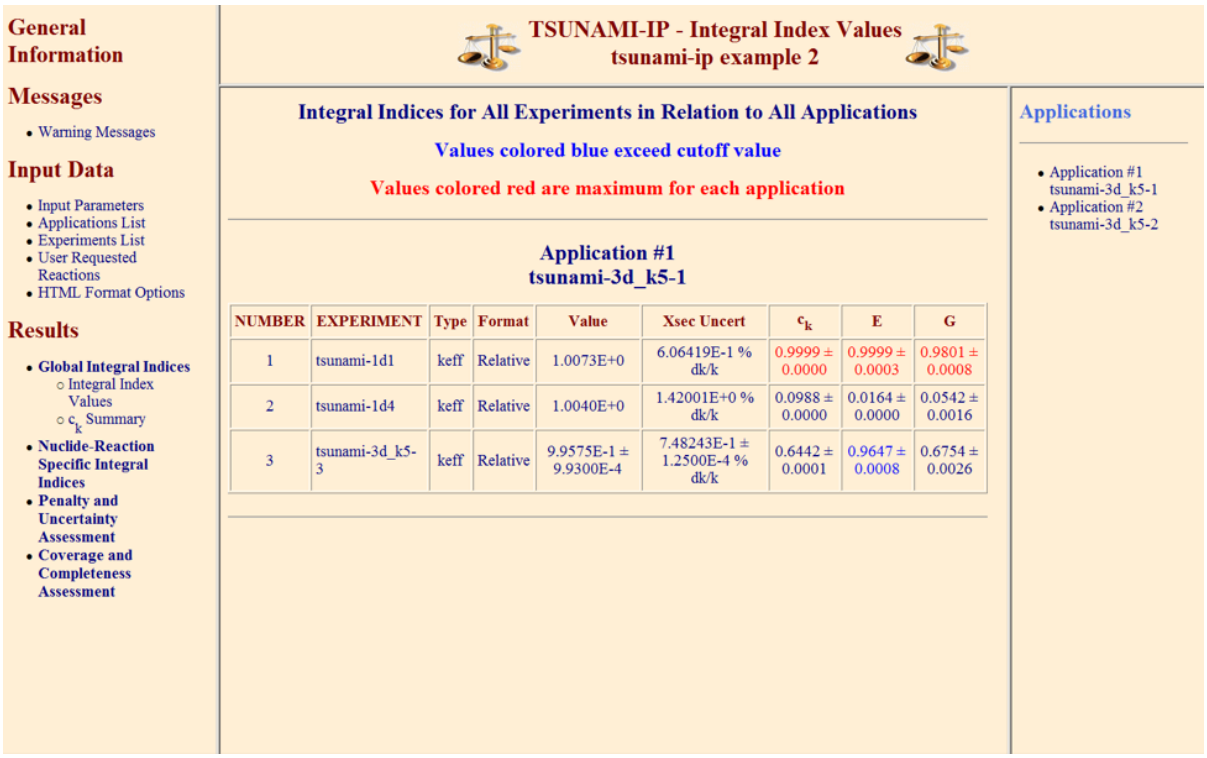


Fig. 6.3.8: Global integral indices from TSUNAMI-IP HTML output.

Data plots can also be directly viewed in the HTML output. The composite data can be viewed by selecting Coverage and Completeness Assessment then Composite Sensitivity Data Plot. A Java applet version of Javapeño will appear in the browser window with the appropriate datafile preloaded. Data can be added to the plot by double-clicking on the list of available data on the right side of Javapeño. The plot shown in Fig. 6.3.9 was produced with this procedure.

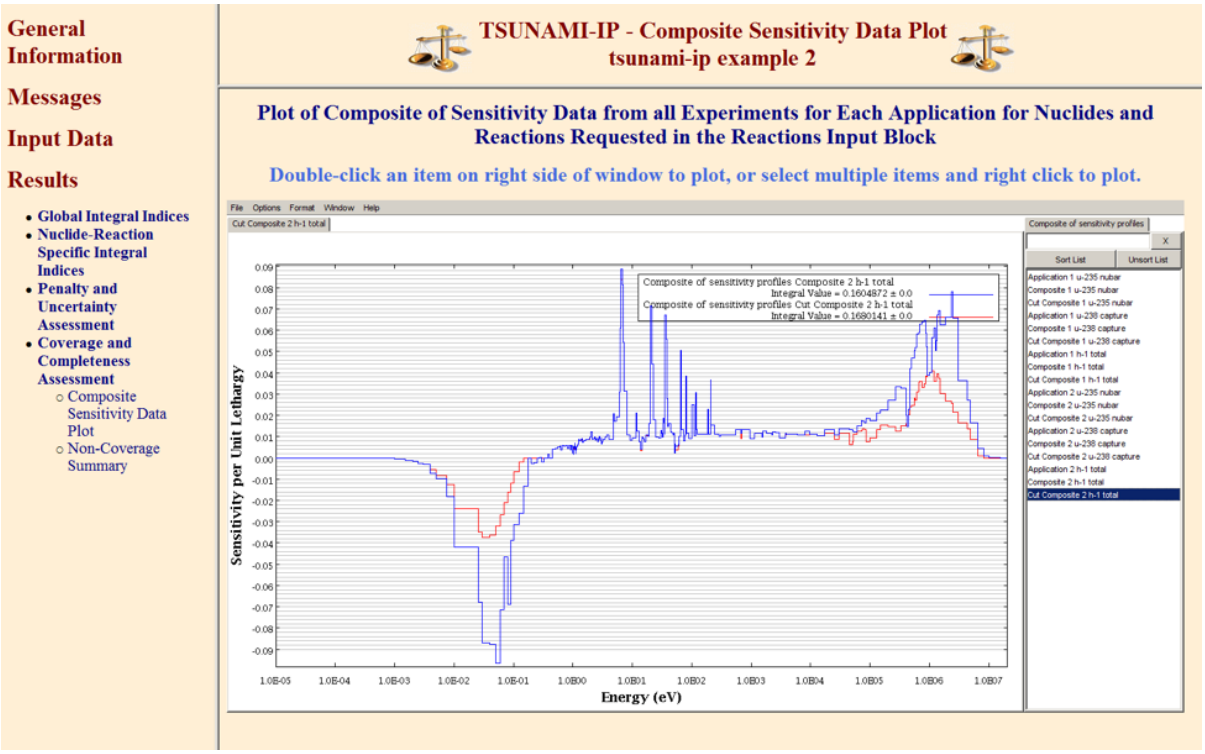


Fig. 6.3.9: Composite sensitivity data plot from TSUNAMI-IP HTML output.

6.3.2 BONAMIST

BONAMIST is a sensitivity version of the SCALE 4.4a BONAMI module. BONAMI performs Bondarenko calculations for resonance self-shielding. The entire FORTRAN 77 source for BONAMI was processed with the GRESS pre-compiler so that the sensitivities of the self-shielded cross sections to the data input to the code are computed. BONAMIST utilizes many of the routines from SCALELIB. BONAMIST writes the resonance self-shielded cross sections to an AMPX formatted data file, just as BONAMI does. However, an additional data file, *bonamist.sen*, is also written. The *bonamist.sen* file contains the sensitivities of resonance self-shielded cross sections generated by BONAMIST to the quantities input to BONAMIST.

The methodologies for computing the resonance self-shielded cross sections in BONAMIST are identical to those used in BONAMI, and equivalent cross section results are obtained. The *bonamist.sen* data file contains the sensitivity of the self-shielded cross sections to the number density of each nuclide and the extra cross section input to BONAMIST. The extra cross section is used by TSUNAMI-1D, TSUNAMI-3D, CSAS and other SCALE sequences to pass the Dancoff factor computed in MIPLIB or SENLIB to BONAMI. The format of the *bonamist.sen* data file is given in Format of BONAMIST.SEN data file.

The input, output, methodologies and program flow of BONAMIST are not described here. The user is referred to the BONAMI chapter. The input of BONAMIST is identical to that required for BONAMI. The sensitivities are automatically computed and the *bonamist.sen* data file is automatically generated without user intervention.

6.3.3 REFERENCES

6.3.4 APPENDICES

6.3.4.1 Data File Formats

Several data file formats that are used by the sensitivity and uncertainty analysis codes are presented in this appendix.

Format of TSUNAMI-A sensitivity data file

The format of the TSUNAMI-A sensitivity data file produced by SAMS for cases with deterministic transport solutions is given in Table 6.3.6. The occurrence of each entry in the data file is followed by an identification of the data contained on each line of the file and the FORTRAN edit descriptor denoting the format of each line. A brief description of each line is also presented.

A sample of the TSUNAMI-A data file for the Flattop-25 sample problem is provided in Example 6.3.13. Here, only two profiles out of the 130 computed are shown.

Table 6.3.6: Format specification for TSUNAMI-A sensitivity data file.

Occurrence	Data	Format	Description
Once at beginning of file.	title	a80	Title extracted from transport calculation
	number of neutron groups	i10	Number of neutron groups in calculation
	total number of profiles, text descriptor, number of total profiles that are region integrated	i10, a35, i10	Total number of sensitivity profiles in data file, separated by a text descriptor, then the number of profiles that contain region-integrated data
	k_{eff}	f10.6	Value of k_{eff} from the forward transport calculation
	'energy boundaries:'	a	Text
	energy boundary data	5es14.6 (repeats until all data is printed)	Values of boundaries of energy groups. Begins with upper value for highest energy group and ends with lower value of lowest energy group.
Repeats for each profile.	isotope name, sensitivity reaction name, nuclide ID, MT number, Zone number or negative of material number, zone volume	a12, 1x, a15, 3i12, es14.6	Provides data identifying the sensitivity data that follows. Note, if sensitivity data is region-integrated, zone number and zone volume are both 0. If data is integrated over all zones containing the same material, the material number is given in place of the zone number as a negative number.
	Energy integrated sensitivity coefficient, sum of absolute value of group-wise sensitivities, sum of the group-wise sensitivities with opposite sign as energy integrated value (osc)	3es14.6	Energy-integrated sensitivity coefficients for this profile.
	group-wise sensitivity coefficients	5es14.6 (repeats until all data is printed)	Energy-dependent sensitivity coefficients. Begins with highest energy group.
Once at end of file.			Block of file verification information.

Example 6.3.13: Truncated sensitivity data file for Flattop-25 sample problem.

```
flattop-25
       238   number of neutron groups
       130   number of sensitivity profiles          30 are region integrated
  1.002969    k-eff from the forward case
energy boundaries:
  2.000000E+07  1.733300E+07  1.568300E+07  1.455000E+07  1.384000E+07
  1.284000E+07  1.000000E+07  8.187300E+06  6.434000E+06  4.800000E+06
  4.304000E+06  3.000000E+06  2.479000E+06  2.354000E+06  1.850000E+06
  1.500000E+06  1.400000E+06  1.356000E+06  1.317000E+06  1.250000E+06
  1.200000E+06  1.100000E+06  1.010000E+06  9.200000E+05  9.000000E+05
  8.750000E+05  8.611000E+05  8.200000E+05  7.500000E+05  6.790000E+05
  6.700000E+05  6.000000E+05  5.730000E+05  5.500000E+05  4.995200E+05
  4.700000E+05  4.400000E+05  4.200000E+05  4.000000E+05  3.300000E+05
  2.700000E+05  2.000000E+05  1.500000E+05  1.283000E+05  1.000000E+05
  8.500000E+04  8.200000E+04  7.500000E+04  7.300000E+04  6.000000E+04
  5.200000E+04  5.000000E+04  4.500000E+04  3.000000E+04  2.500000E+04
  1.700000E+04  1.300000E+04  9.500000E+03  8.030000E+03  6.000000E+03
  3.900000E+03  3.740000E+03  3.000000E+03  2.580000E+03  2.290000E+03
  2.200000E+03  1.800000E+03  1.550000E+03  1.500000E+03  1.150000E+03
  9.500000E+02  6.830000E+02  6.700000E+02  5.500000E+02  3.050000E+02
  2.850000E+02  2.400000E+02  2.100000E+02  2.075000E+02  1.925000E+02
  1.860000E+02  1.220000E+02  1.190000E+02  1.150000E+02  1.080000E+02
  1.000000E+02  9.000000E+01  8.200000E+01  8.000000E+01  7.600000E+01
  7.200000E+01  6.750000E+01  6.500000E+01  6.100000E+01  5.900000E+01
  5.340000E+01  5.200000E+01  5.060000E+01  4.920000E+01  4.830000E+01
  4.700000E+01  4.520000E+01  4.400000E+01  4.240000E+01  4.100000E+01
  3.960000E+01  3.910000E+01  3.800000E+01  3.700000E+01  3.550000E+01
  3.460000E+01  3.375000E+01  3.325000E+01  3.175000E+01  3.125000E+01
  3.000000E+01  2.750000E+01  2.500000E+01  2.250000E+01  2.100000E+01
  2.000000E+01  1.900000E+01  1.850000E+01  1.700000E+01  1.600000E+01
  1.509990E+01  1.440000E+01  1.375000E+01  1.290000E+01  1.190000E+01
  1.150000E+01  1.000000E+01  9.099990E+00  8.099990E+00  7.150000E+00
  7.000000E+00  6.750000E+00  6.500000E+00  6.250000E+00  6.000000E+00
  5.400000E+00  5.000000E+00  4.750000E+00  4.000000E+00  3.730000E+00
  3.500000E+00  3.150000E+00  3.049990E+00  3.000000E+00  2.969990E+00
  2.870000E+00  2.770000E+00  2.669990E+00  2.570000E+00  2.469990E+00
  2.379990E+00  2.299990E+00  2.209990E+00  2.120000E+00  2.000000E+00
  1.940000E+00  1.860000E+00  1.770000E+00  1.679990E+00  1.589990E+00
  1.500000E+00  1.450000E+00  1.400000E+00  1.349990E+00  1.299990E+00
  1.250000E+00  1.224990E+00  1.200000E+00  1.174990E+00  1.150000E+00
  1.139990E+00  1.129990E+00  1.120000E+00  1.110000E+00  1.099990E+00
  1.089990E+00  1.080000E+00  1.070000E+00  1.059990E+00  1.049990E+00
  1.040000E+00  1.030000E+00  1.020000E+00  1.009990E+00  1.000000E+00
  9.750000E-01  9.500000E-01  9.250000E-01  9.000000E-01  8.500000E-01
  8.000000E-01  7.500000E-01  7.000000E-01  6.500000E-01  6.250000E-01
  6.000000E-01  5.500000E-01  5.000000E-01  4.500000E-01  4.000000E-01
  3.750000E-01  3.500000E-01  3.250000E-01  3.000000E-01  2.750000E-01
  2.500000E-01  2.250000E-01  2.000000E-01  1.750000E-01  1.500000E-01
```

```
  1.250000E-01  1.000000E-01  9.000000E-02  8.000000E-02  7.000000E-02
  6.000000E-02  5.000000E-02  4.000000E-02  3.000000E-02  2.530000E-02
  1.000000E-02  7.500000E-03  5.000000E-03  4.000000E-03  3.000000E-03
  2.500000E-03  2.000000E-03  1.500000E-03  1.200000E-03  1.000000E-03
  7.500000E-04  5.000000E-04  1.000000E-04  1.000000E-05
u-234        total                92234          1          0  0.000000E+00
  4.717915E-03  4.975886E-03 -1.289843E-04
  2.506892E-08  1.674526E-07  3.009241E-07  3.212515E-07  8.004773E-07
  9.602059E-06  2.767762E-05  8.282738E-05  2.095243E-04  1.240423E-04
  6.019243E-04  4.044642E-04  1.135071E-04  5.606487E-04  4.757248E-04
  1.462671E-04  6.761932E-05  6.289243E-05  1.095966E-04  8.561461E-05
  1.773753E-04  1.640000E-04  1.761175E-04  4.353814E-05  5.616768E-05
  3.283979E-05  1.005176E-04  1.825428E-04  1.823796E-04  2.225234E-05
```

(continues on next page)

(continued from previous page)

```
  1.648190E-04  5.830123E-05  4.848578E-05  1.014302E-04  5.276815E-05
  4.675121E-05  2.661435E-05  2.271738E-05  5.878201E-05  3.053242E-05
  1.441957E-05 -3.866848E-06 -8.105675E-06 -1.524720E-05 -1.112164E-05
 -1.873898E-06 -5.969614E-06 -1.989077E-06 -1.326105E-05 -9.321207E-06
 -2.404530E-06 -6.309741E-06 -1.968570E-05 -7.035957E-06 -9.446309E-06
 -4.643517E-06 -3.590477E-06 -1.204560E-06 -1.496028E-06 -1.231667E-06
 -9.897018E-08 -3.797082E-07 -1.884879E-07 -1.238474E-07 -2.742831E-08
 -9.015365E-08 -4.731163E-08 -5.688008E-09 -9.784721E-08 -4.339968E-08
 -3.857598E-08 -1.788772E-09 -9.179912E-09 -1.942945E-08 -7.808444E-10
 -1.033952E-09 -2.787285E-10 -7.649368E-11  3.174200E-12 -2.032114E-09
 -2.519205E-09 -8.466172E-12 -1.782872E-11 -4.164121E-10 -1.019102E-10
 -8.554906E-11 -8.181892E-12 -1.271741E-11 -2.199571E-10 -1.012644E-12
 -2.525998E-11  3.873520E-12  3.246969E-12 -1.846717E-12  2.073834E-12
 -2.864914E-12 -2.834941E-13 -1.647038E-12 -7.822336E-11 -3.875318E-12
 -2.367591E-11 -2.433239E-13  4.302250E-13  1.328992E-12 -2.683548E-14
 -3.569867E-15 -1.056719E-12  2.327642E-13  2.141648E-12 -6.464708E-15
  2.102727E-13  4.061489E-13  4.297794E-12 -1.140715E-11 -2.485342E-11
  3.520045E-12 -1.312734E-12 -6.466205E-13  4.677226E-14  5.384058E-13
 -7.029536E-14 -3.172012E-13 -1.050521E-12 -1.448066E-12  8.609614E-14
 -9.503147E-14  1.224622E-13 -5.610401E-14  1.206730E-13 -1.808347E-13
  5.110267E-13 -2.647888E-14 -5.274697E-13 -1.360891E-13 -9.483789E-15
  1.745369E-13 -2.324100E-13 -1.631119E-14  8.369275E-14 -2.896774E-12
 -4.645450E-14 -1.014518E-12 -1.253069E-12 -1.224193E-13 -3.247751E-13
 -2.170010E-14  5.679165E-15 -5.495932E-15 -5.738207E-15 -2.640171E-14
 -4.723927E-14 -2.692265E-14 -2.470002E-14 -2.212313E-14 -1.887611E-14
 -1.473034E-14 -1.451047E-14 -1.414891E-14  2.318902E-14  4.465008E-15
 -7.884706E-15 -1.256517E-14 -1.036473E-14 -9.936883E-15 -9.776219E-15
 -4.800508E-15 -4.391959E-15 -3.279705E-15  5.435813E-15 -6.187775E-15
 -1.940775E-15 -5.997969E-16 -3.333506E-16 -4.412221E-17 -4.046016E-16
 -3.825563E-17 -2.586768E-17 -1.933342E-17 -1.012773E-17 -5.796910E-18
 -1.903947E-18 -1.731421E-18 -2.378372E-19 -2.074179E-18 -7.863836E-19
 -1.446345E-18  1.001051E-18  5.654388E-18  7.607361E-18 -3.840432E-16
 -6.762651E-16 -7.569632E-16 -7.992030E-16 -1.579348E-15 -1.638912E-15
 -1.614449E-15 -1.524987E-15 -1.454677E-15 -2.958432E-16 -2.370000E-16
  2.145063E-16 -1.680721E-15 -3.103977E-15 -2.352000E-15  1.009233E-16
  1.583744E-17  4.777954E-18  5.125365E-19 -1.385105E-18 -7.333829E-19
  9.783341E-19  1.885422E-20  1.280582E-22 -1.217866E-22  6.410866E-23
```

```
 -2.958369E-24 -1.558733E-25 -5.544137E-26 -1.831995E-26 -5.448893E-27
 -1.415088E-27 -3.110222E-28 -5.522761E-29 -5.429459E-30 -1.728100E-30
 -2.514164E-32  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00
  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00
  0.000000E+00  0.000000E+00  0.000000E+00
u-234        scatter              92234           1           0  0.000000E+00
  8.032846E-04  8.241044E-04 -1.041006E-05
 -3.248099E-09 -2.849794E-09  3.761279E-09  4.894583E-09  8.374101E-09
 -3.099800E-09 -7.458623E-07 -7.683340E-06  6.521506E-06  7.401795E-06
  3.981803E-05  3.092353E-05  9.673315E-06  5.444300E-05  4.968639E-05
  2.118375E-05  1.138184E-05  1.134012E-05  2.083885E-05  1.703296E-05
  3.708780E-05  3.574228E-05  3.382708E-05  7.202870E-06  8.988859E-06
  4.995195E-06  1.446466E-05  2.424787E-05  2.765288E-05  3.825457E-06
  3.116989E-05  1.317151E-05  1.199450E-05  2.762489E-05  1.707364E-05
  1.844534E-05  1.263098E-05  1.257121E-05  4.478947E-05  4.142602E-05
  4.409844E-05  2.870006E-05  1.012683E-05  1.152729E-05  4.782801E-06
  1.259705E-06  1.793371E-06  2.483675E-07  1.496686E-06  1.706688E-07
  1.686857E-07  9.490054E-08 -7.954430E-07 -4.688914E-07 -2.309356E-07
 -2.471537E-07 -1.786610E-07 -2.395761E-08 -3.666843E-09  1.724470E-08
 -3.073805E-09 -2.746581E-09  1.909088E-09 -1.216879E-08  1.541106E-09
  3.070286E-09 -1.714544E-09  2.222325E-09  4.704197E-09 -5.869767E-10
 -2.023745E-09 -1.670404E-10 -3.456028E-10  4.846314E-10  1.168348E-10
  9.827623E-12 -5.473694E-11  1.402322E-11  2.383711E-11  2.450650E-10
  3.568164E-10 -7.159047E-12 -1.503380E-11  6.139393E-11  3.348865E-12
  1.628715E-10 -4.819499E-12 -9.202975E-12  8.831698E-12  3.135675E-13
 -1.961365E-11  3.846148E-12  2.808294E-12 -1.477353E-12  1.857857E-12
```

(continues on next page)

(continued from previous page)

```
 -2.155043E-12 -1.789186E-13 -4.133771E-13  3.033233E-12  4.904576E-14
  3.694654E-14 -6.397666E-14  4.128331E-13  1.060395E-12  1.458698E-15
  5.731954E-15 -7.811219E-13  2.242784E-13  1.751843E-12 -3.146954E-15
  1.844389E-13  3.293311E-13  3.467402E-12  2.026690E-12  5.768864E-12
  2.955440E-12 -8.597300E-13  5.334737E-14  1.100362E-13  4.331466E-13
 -4.817834E-14 -2.394196E-13 -7.884115E-13 -1.120566E-12  8.127751E-14
 -5.460604E-14  1.006648E-13 -3.436737E-14  9.646284E-14 -1.382066E-13
  4.648416E-13  5.996957E-15 -3.646852E-13  6.023955E-14 -4.109659E-15
  1.203473E-13 -1.351654E-13 -3.330587E-15  8.844682E-14 -1.428401E-13
  6.663312E-13 -1.568665E-14  1.282612E-14  1.681431E-14 -8.390950E-14
  1.250912E-14  5.585417E-15  3.200709E-16 -4.288993E-16 -4.661462E-15
 -1.315587E-14 -2.131794E-15 -1.135026E-16  6.150330E-16  1.873476E-16
  2.704163E-16 -8.424184E-17 -2.454189E-15  1.342258E-14  3.898874E-15
  1.177619E-16 -8.072667E-16  3.713938E-16  3.286023E-16 -2.195845E-17
  3.692910E-17 -8.551075E-18  2.126361E-16  4.077162E-15 -1.879803E-15
 -5.420138E-16  5.678537E-18  6.148490E-17  1.547221E-16 -1.861868E-16
 -1.463798E-17 -9.869435E-18 -7.532523E-18 -3.710468E-18 -2.008824E-18
 -3.941254E-19 -4.338395E-19  1.982443E-19 -6.647317E-19 -6.961168E-21
 -1.060476E-19  1.532491E-18  4.815554E-18  8.072217E-18  1.094195E-16
  6.851480E-18 -8.567528E-18 -1.375315E-17  2.677723E-17  9.881978E-18
  8.154495E-18  1.038738E-18 -1.150521E-17 -2.328087E-18 -1.094495E-18
  5.286105E-16 -2.343353E-16 -7.924307E-16 -5.631898E-16  3.676678E-17
  5.447929E-18  1.550351E-18  1.795543E-19 -3.911335E-19 -2.070213E-19
```

```
  2.666570E-19  5.006574E-21  6.432183E-23 -1.883732E-23  1.634739E-23
 -7.540091E-26 -3.531102E-27 -1.165437E-27 -3.541294E-28 -9.582782E-29
 -2.220252E-29 -4.245368E-30 -6.290953E-31 -5.169058E-32 -1.116416E-32
 -2.558539E-36  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00
  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00
  0.000000E+00  0.000000E+00  0.000000E+00

...

file verification information
 code system:    scale
 version:    6.2
 program:  sams
   creation date:  01_apr_2016
   library:  /home/c53/scale_freeze/Linux_x86_64-release/bin/scale
   migration code:  sams
   version:  6.2.0
 jobname:  c53
 machine name:  node23
 date of file creation:  01_apr_2016
time of file creation:  17:22:50.45
 filename:  tsunami-1d4.sdf
 chi sensitivities are constrained
```

Format of TSUNAMI-B sensitivity data file

The format of the TSUNAMI-B sensitivity data file produced by SAMS for cases with Monte Carlo transport solutions is given in Table 6.3.7. The occurrence of each entry in the data file is followed by an identification of the data contained on each line of the file and the FORTRAN edit descriptor denoting the format of each line. A brief description of each line is also presented.

A sample of the TSUNAMI-B data file for the LEU-COMP-THERM-009 case 10 sample problem is provided in Example 6.3.14. Here, only two profiles out of the 3389 computed are shown.

continues on next page

Table 6.3.7 – continued from previous page

Table 6.3.7: Format specification for TSUNAMI-B sensitivity data file.

ARRAY Format: READ ARRAY array parameters data type orientation data END ARRAY See Sections on ARRAY Data, Geometry, and alternative sample problem mockups

Repeat the sequence ARRAY PARAMETERS DATA TYPE ORIENTATION DATA for each array used in the problem.

ARRAY PARAMETERS

KEYWORD	DEFAULT	DEFINITION
ARA=	none[a]	no. defining the array
TYP=	Cuboidal[c]	array type (cuboidal or square, hexagonal or triangular, shexagonal, rhexagonal, dodecahedral)
NUX=	none[a]	no. of units in X direction
NUY=	none[a]	no. of units in Y direction
NUZ=	none[a]	no. of units in Z direction
GBL=	none[b]	global or overall array number
PRT=	YES	print any label
COM=	none	delim comment delim optional, maximum of 132 characters

ORIENTATION DATA FOR FILL

Enter unit numbers to define every position in the array. When entering data using the options in this table, the count and option fields must be adjacent with no imbedded blanks. The operand field may be separated from the option field by one or more blanks. Orientation data for FILL is terminated by entering END FILL.

COUNT FIELD	OPTION FIELD	OPERAND FIELD	COMMENTS
		j	stores j at the current position in the array
i	R, *, $	j	stores j in the next i positions in the array
	F	j	fills remainder of the array with unit no. j starting with the current array position
	A	j	sets the current position in the array to j
i	S		increments current position in the array by i (This allows skipping i positions. The value of i may be positive or negative.)
i	Q	j	repeats the previous j entries i times. The default value of i is 1
i	N	j	repeats the previous j entries i times, inverting the sequence each time. The default value of i is 1.
i	B	j	starting with the entry at -i from the current position, store entries in inverse order until position -(i+j) is reached. Default value of i=1.
i	I	j k	provides the end points (j, k) with i entries linearly interpolated between them (i.e., a total of i+2 points). At least one blank must separate j and k. When used for an integer array, the i option should only be used to generate integer steps (i.e., (k-j)/(i+1) should be a whole number).

ORIENTATION DATA FOR LOOP

Enter the unit number and nine numbers that define the position(s) of that unit. Data for each of these ten entries are repeated until every position in the array has been defined. Orientation data for LOOP are terminated by entering END LOOP.
ENTER DATA IN THE FORM:

DATA ENTRY	COMMENTS
LTYPE	The unit type. LTYPE must be greater than 0.
IX1	Starting position in the X direction. IX1 must be at least 1 and no larger than the value entered for NUX.
IX2	Ending position in the X direction. IX2 must be at least 1 and no larger than the value of NUX.
INCX	The number of units by which increments are made in the X direction.
IY1	The starting position in the Y direction. IY1 must be at least 1 and less than the value entered for NUY.
IY2	Ending position in the Y direction. IY2 must be at least 1 and no larger than the value of NUY.
INCY	The number of units by which increments are made in the positive Y direction.
IZ1	Starting position in the Z direction. IZ1 must be at least 1 and no larger than NUZ.
IZ2	Ending position in the Z direction. IZ2 must be at least 1 and no larger than NUZ.
INCZ	The number of units by which increments are made in the positive Z direction.

[a] In KENO V.a the default is 1

[b] In KENO V.a the default is the largest array number, *maxara*

continues on next page

Table 6.3.7 – continued from previous page

Occurrence	Data	Format	Description
	energy integrated sensitivity coefficient, standard deviation for energy integrated sensitivity coefficient, sum of absolute value of group-wise sensitivities, sum of the group-wise sensitivities with opposite sign as energy integrated value (osc), standard deviation for osc.	5es14.6	Energy-integrated sensitivity coefficients and their standard deviations for this profile.
	group-wise sensitivity coefficients	5es14.6 (repeats until all data is printed)	Energy-dependent sensitivity coefficients. Begins with highest energy group.
	standard deviation in group-wise sensitivity coefficients	5es14.6 (repeats until all data is printed)	Standard deviation for energy-dependent sensitivity coefficients. Begins with highest energy group.
Once at end of file.			Block of file verification information.

Example 6.3.14: Truncated sensitivity data file for LEU-COMP-THERM-009 sample problem.

```
sample 3 - tsunami-3d
       238   number of neutron groups
      3389   number of sensitivity profiles         291 are region integrated
  1.004674 +/-   0.000543 k-eff from the forward case
energy boundaries:
  2.000000E+07  1.733300E+07  1.568300E+07  1.455000E+07  1.384000E+07
  1.284000E+07  1.000000E+07  8.187300E+06  6.434000E+06  4.800000E+06
  4.304000E+06  3.000000E+06  2.479000E+06  2.354000E+06  1.850000E+06
  1.500000E+06  1.400000E+06  1.356000E+06  1.317000E+06  1.250000E+06
  1.200000E+06  1.100000E+06  1.010000E+06  9.200000E+05  9.000000E+05
  8.750000E+05  8.611000E+05  8.200000E+05  7.500000E+05  6.790000E+05
  6.700000E+05  6.000000E+05  5.730000E+05  5.500000E+05  4.995200E+05
  4.700000E+05  4.400000E+05  4.200000E+05  4.000000E+05  3.300000E+05
  2.700000E+05  2.000000E+05  1.500000E+05  1.283000E+05  1.000000E+05
  8.500000E+04  8.200000E+04  7.500000E+04  7.300000E+04  6.000000E+04
  5.200000E+04  5.000000E+04  4.500000E+04  3.000000E+04  2.500000E+04
  1.700000E+04  1.300000E+04  9.500000E+03  8.030000E+03  6.000000E+03
  3.900000E+03  3.740000E+03  3.000000E+03  2.580000E+03  2.290000E+03
  2.200000E+03  1.800000E+03  1.550000E+03  1.500000E+03  1.150000E+03
  9.500000E+02  6.830000E+02  6.700000E+02  5.500000E+02  3.050000E+02
  2.850000E+02  2.400000E+02  2.100000E+02  2.075000E+02  1.925000E+02
  1.860000E+02  1.220000E+02  1.190000E+02  1.150000E+02  1.080000E+02
  1.000000E+02  9.000000E+01  8.200000E+01  8.000000E+01  7.600000E+01
  7.200000E+01  6.750000E+01  6.500000E+01  6.100000E+01  5.900000E+01
  5.340000E+01  5.200000E+01  5.060000E+01  4.920000E+01  4.830000E+01
  4.700000E+01  4.520000E+01  4.400000E+01  4.240000E+01  4.100000E+01
  3.960000E+01  3.910000E+01  3.800000E+01  3.700000E+01  3.550000E+01
  3.460000E+01  3.375000E+01  3.325000E+01  3.175000E+01  3.125000E+01
  3.000000E+01  2.750000E+01  2.500000E+01  2.250000E+01  2.100000E+01
  2.000000E+01  1.900000E+01  1.850000E+01  1.700000E+01  1.600000E+01
  1.509990E+01  1.440000E+01  1.375000E+01  1.290000E+01  1.190000E+01
  1.150000E+01  1.000000E+01  9.099990E+00  8.099990E+00  7.150000E+00
  7.000000E+00  6.750000E+00  6.500000E+00  6.250000E+00  6.000000E+00
  5.400000E+00  5.000000E+00  4.750000E+00  4.000000E+00  3.730000E+00
  3.500000E+00  3.150000E+00  3.049990E+00  3.000000E+00  2.969990E+00
  2.870000E+00  2.770000E+00  2.669990E+00  2.570000E+00  2.469990E+00
  2.379990E+00  2.299990E+00  2.209990E+00  2.120000E+00  2.000000E+00
  1.940000E+00  1.860000E+00  1.770000E+00  1.679990E+00  1.589990E+00
  1.500000E+00  1.450000E+00  1.400000E+00  1.349990E+00  1.299990E+00
```

(continues on next page)

(continued from previous page)

```
  1.250000E+00  1.224990E+00  1.200000E+00  1.174990E+00  1.150000E+00
  1.139990E+00  1.129990E+00  1.120000E+00  1.110000E+00  1.099990E+00
  1.089990E+00  1.080000E+00  1.070000E+00  1.059990E+00  1.049990E+00
  1.040000E+00  1.030000E+00  1.020000E+00  1.009990E+00  1.000000E+00
  9.750000E-01  9.500000E-01  9.250000E-01  9.000000E-01  8.500000E-01
  8.000000E-01  7.500000E-01  7.000000E-01  6.500000E-01  6.250000E-01
  6.000000E-01  5.500000E-01  5.000000E-01  4.500000E-01  4.000000E-01
  3.750000E-01  3.500000E-01  3.250000E-01  3.000000E-01  2.750000E-01
  2.500000E-01  2.250000E-01  2.000000E-01  1.750000E-01  1.500000E-01
  1.250000E-01  1.000000E-01  9.000000E-02  8.000000E-02  7.000000E-02
  6.000000E-02  5.000000E-02  4.000000E-02  3.000000E-02  2.530000E-02
```

```
 1.000000E-02  7.500000E-03  5.000000E-03  4.000000E-03  3.000000E-03
 2.500000E-03  2.000000E-03  1.500000E-03  1.200000E-03  1.000000E-03
 7.500000E-04  5.000000E-04  1.000000E-04  1.000000E-05
h-1          total                 1001           1
     0      0
 0.000000E+00  0.000000E+00      0      0
 5.878199E-01  1.080812E-02  6.393673E-01 -2.577372E-02  6.213759E-03
 0.000000E+00  5.495367E-07  2.331587E-06  3.300773E-06  1.013678E-05
 1.664451E-04  6.565088E-04  2.936479E-03  1.086949E-02  6.494953E-03
 3.153059E-02  2.792794E-02  8.750866E-03  3.526685E-02  2.959999E-02
 9.882265E-03  4.334542E-03  3.162922E-03  5.604319E-03  5.096612E-03
 9.352559E-03  6.455556E-03  6.329793E-03  2.219300E-03  3.147784E-03
 1.846754E-03  5.699152E-03  1.105599E-02  1.187824E-02  1.369476E-03
 1.163906E-02  4.560338E-03  4.090504E-03  9.433514E-03  4.478346E-03
 3.184407E-03  1.346294E-03  2.053422E-03  1.053713E-02  1.140274E-02
 1.608992E-02  1.374175E-02  6.013477E-03  9.138809E-03  5.250737E-03
 1.183478E-03  2.853132E-03  8.610512E-04  5.268028E-03  3.930706E-03
 1.126351E-03  3.129538E-03  9.236847E-03  3.594958E-03  8.088362E-03
 6.009586E-03  6.688602E-03  3.495383E-03  5.765521E-03  8.388449E-03
 7.368939E-04  4.127158E-03  2.704750E-03  1.835688E-03  7.008772E-04
 3.511407E-03  2.453974E-03  6.089740E-04  4.814587E-03  2.868577E-03
 5.284937E-03  2.875470E-04  4.140647E-03  9.981738E-03  1.164813E-03
 2.291818E-03  2.495308E-03  2.124530E-04  9.511635E-04  5.255117E-04
 7.128428E-03  4.241791E-04  4.854808E-04  1.006414E-03  1.796904E-03
 1.660348E-03  6.851843E-04  3.984874E-04  7.034940E-04  5.358846E-04
 1.050901E-03  3.317304E-04  7.389811E-04  1.406010E-04  6.820000E-04
 1.101282E-03  4.862233E-04  6.513879E-04  3.097945E-04  2.935911E-04
 6.771809E-04  1.280746E-03  7.355201E-04  1.169403E-03  6.720975E-04
 2.362427E-04  6.381695E-04  3.898248E-04  6.170204E-04  5.175026E-04
 3.620459E-04  2.299800E-04  7.884422E-04  2.636381E-04  6.499650E-04
 1.500763E-03  1.643672E-03  2.051421E-03  8.798907E-04  8.456801E-04
 8.233886E-04  4.289658E-04  1.624594E-03  8.639144E-04  2.045707E-04
 1.167873E-03  4.272030E-04  1.133354E-03  1.883793E-03  6.082309E-04
-3.184982E-04  1.643061E-03  1.570067E-03  2.113940E-03  2.854535E-04
 5.201036E-04  4.741151E-04  5.402900E-04  5.840588E-04  1.450502E-03
 1.341676E-03  7.172654E-04  2.380048E-03  1.226849E-03  8.828276E-04
 1.603862E-03  4.655939E-04  2.244715E-04  1.612430E-04  5.069606E-04
 4.950527E-04  5.691839E-04  5.735686E-04  6.123026E-04  5.276913E-04
 4.411023E-04  5.995673E-04  4.464893E-04  8.972538E-04  4.808735E-04
 5.433926E-04  6.923829E-04  8.862597E-04  9.481183E-04  1.097237E-03
 6.662853E-04  7.941514E-04  9.267192E-04  9.465431E-04  1.350355E-03
 7.133665E-04  7.386729E-04  9.478894E-04  1.290858E-03  5.925235E-04
 7.198542E-04  8.246520E-04  1.024889E-03  1.260913E-03  1.611722E-03
 1.951751E-03  2.132400E-03  2.164822E-03  2.202685E-03  2.180256E-03
 2.151288E-03  1.960147E-03  1.642508E-03  1.331129E-03  2.219066E-03
 1.389834E-03  1.133856E-03  8.478845E-04  1.412585E-03  1.256246E-03
 1.073678E-03  1.132597E-03  8.625521E-04  4.983772E-04  5.043924E-04
 1.238305E-03  9.734085E-04  9.374273E-04  7.145087E-04  4.779918E-04
 4.910213E-04  5.642105E-04  7.554480E-04  8.415005E-04  1.107943E-03
 1.292846E-03  1.298630E-03  1.487261E-03  1.447523E-03  2.061059E-03
```

```
 2.543669E-03  5.434130E-04 -3.014440E-04 -8.338618E-05 -1.259993E-03
-1.049059E-03 -1.516055E-03 -4.246086E-03 -2.081466E-03 -9.899241E-03
-1.549410E-03 -1.532964E-03 -4.959768E-04 -4.645005E-04 -2.443465E-04
-1.921544E-04 -1.851809E-04 -9.943808E-05 -6.213754E-05 -6.412054E-05
-5.667409E-05 -6.599494E-05 -5.596935E-06
 0.000000E+00  2.166669E-07  4.563308E-07  5.797961E-07  1.052589E-06
 5.227409E-06  1.635907E-05  5.783163E-05  1.976880E-04  1.499190E-04
 7.129798E-04  6.959004E-04  2.559563E-04  9.831456E-04  9.627497E-04
 3.887061E-04  1.964393E-04  1.556215E-04  2.896371E-04  2.635342E-04
 4.911421E-04  4.089036E-04  4.619491E-04  1.526265E-04  1.964699E-04
 1.289384E-04  3.598922E-04  6.532525E-04  6.485064E-04  1.182854E-04
 6.882841E-04  3.167550E-04  2.880876E-04  6.021766E-04  3.712318E-04
 3.285352E-04  2.205750E-04  2.868595E-04  1.144578E-03  1.210874E-03
 1.630829E-03  1.478332E-03  8.691904E-04  1.300316E-03  8.875954E-04
 2.513224E-04  5.308395E-04  2.038769E-04  1.063601E-03  8.168670E-04
 2.744386E-04  6.274111E-04  1.948515E-03  9.865712E-04  1.823839E-03
 1.338510E-03  1.511362E-03  9.038528E-04  1.457087E-03  1.918360E-03
 2.801742E-04  1.096586E-03  7.956968E-04  6.608086E-04  2.723390E-04
 1.036666E-03  8.100112E-04  2.349632E-04  1.322213E-03  1.016259E-03
 1.612241E-03  1.628674E-04  1.053085E-03  2.383641E-03  3.969473E-04
 8.650431E-04  7.030079E-04  1.137526E-04  4.473213E-04  2.406486E-04
 1.745300E-03  1.777857E-04  2.259530E-04  3.595638E-04  4.174393E-04
 5.449888E-04  4.941223E-04  1.791767E-04  3.092047E-04  3.136610E-04
 3.682297E-04  2.325259E-04  3.616855E-04  2.189339E-04  5.192921E-04
 1.676002E-04  1.898789E-04  1.855957E-04  1.451389E-04  1.871217E-04
 2.472929E-04  1.716087E-04  2.387386E-04  2.055161E-04  2.282567E-04
 1.131949E-04  1.944377E-04  1.879942E-04  2.618990E-04  1.819375E-04
 1.802619E-04  1.265510E-04  2.860560E-04  1.303750E-04  2.620363E-04
 4.822369E-04  5.053777E-04  5.572759E-04  3.782100E-04  2.943088E-04
 3.092838E-04  1.883722E-04  4.454995E-04  3.535025E-04  3.330253E-04
 2.591977E-04  2.743464E-04  3.634571E-04  4.285595E-04  2.064984E-04
 6.655956E-04  4.948453E-04  5.874258E-04  5.804186E-04  1.536659E-04
 2.232270E-04  2.315359E-04  2.424440E-04  2.484870E-04  5.424511E-04
 4.148698E-04  2.984723E-04  8.109708E-04  3.782729E-04  3.508202E-04
 5.266942E-04  1.997956E-04  1.284097E-04  9.413755E-05  2.183365E-04
 2.207723E-04  2.293038E-04  2.357776E-04  2.480385E-04  2.314636E-04
 2.201457E-04  2.483637E-04  2.551705E-04  3.358805E-04  2.013221E-04
 2.605793E-04  2.935817E-04  3.074188E-04  3.212076E-04  3.372974E-04
 2.217707E-04  2.272350E-04  2.315085E-04  2.365833E-04  2.435013E-04
 1.491903E-04  1.504862E-04  1.526581E-04  1.494485E-04  7.853944E-05
 7.694254E-05  7.521152E-05  7.139824E-05  6.321032E-05  5.386979E-05
 4.227381E-05  3.244781E-05  2.600470E-05  2.441022E-05  2.685967E-05
 3.485016E-05  4.518938E-05  5.804115E-05  7.003531E-05  1.491623E-04
 1.711490E-04  1.797697E-04  1.924294E-04  3.367520E-04  3.626060E-04
 3.876486E-04  4.127060E-04  4.406396E-04  2.723181E-04  2.809896E-04
 5.027895E-04  5.409453E-04  5.750452E-04  5.933219E-04  3.346886E-04
 3.073575E-04  2.671577E-04  2.257729E-04  2.317883E-04  2.815263E-04
 3.724861E-04  4.884778E-04  6.219380E-04  8.232623E-04  1.116475E-03
 1.587840E-03  1.081433E-03  1.278081E-03  1.529288E-03  1.839648E-03
```

```
 2.160035E-03  2.473668E-03  2.749914E-03  1.747805E-03  2.894034E-03
 7.595401E-04  5.846859E-04  2.217471E-04  1.914071E-04  9.697134E-05
 8.595824E-05  7.688035E-05  4.872943E-05  3.432014E-05  3.488024E-05
 3.196222E-05  3.258262E-05  8.777834E-06
h-1          scatter               1001           1
     0      0
 0.000000E+00  0.000000E+00      0      0
 6.303991E-01  1.077414E-02  6.400794E-01 -4.840258E-03  4.481088E-03
 0.000000E+00  5.495671E-07  2.331707E-06  3.300919E-06  1.013723E-05
 1.664523E-04  6.565364E-04  2.936591E-03  1.086989E-02  6.495210E-03
 3.153197E-02  2.792903E-02  8.751186E-03  3.526826E-02  2.960132E-02
 9.882722E-03  4.334749E-03  3.163091E-03  5.604652E-03  5.096891E-03
 9.353111E-03  6.456018E-03  6.330305E-03  2.219428E-03  3.147952E-03
 1.846853E-03  5.699483E-03  1.105662E-02  1.187883E-02  1.369554E-03
 1.163969E-02  4.560593E-03  4.090726E-03  9.434017E-03  4.478649E-03
```

(continues on next page)

(continued from previous page)

```
 3.184690E-03  1.346479E-03  2.053654E-03  1.053816E-02  1.140387E-02
 1.609164E-02  1.374353E-02  6.014523E-03  9.140635E-03  5.252031E-03
 1.183774E-03  2.853883E-03  8.612880E-04  5.269829E-03  3.932118E-03
 1.126754E-03  3.130657E-03  9.241730E-03  3.597525E-03  8.094637E-03
 6.014777E-03  6.695679E-03  3.499683E-03  5.774016E-03  8.403651E-03
 7.385747E-04  4.136535E-03  2.711805E-03  1.841652E-03  7.029597E-04
 3.522480E-03  2.462885E-03  6.110088E-04  4.832324E-03  2.882767E-03
 5.312621E-03  2.893077E-04  4.159580E-03  1.005115E-02  1.174187E-03
 2.317001E-03  2.516301E-03  2.144128E-04  9.637796E-04  5.314553E-04
 7.209417E-03  4.294930E-04  4.929678E-04  1.020469E-03  1.814332E-03
 1.685521E-03  7.086673E-04  4.049238E-04  7.171356E-04  5.502240E-04
 1.068964E-03  3.422397E-04  7.575767E-04  1.504152E-04  7.128682E-04
 1.109028E-03  4.950626E-04  6.602400E-04  3.159006E-04  3.027021E-04
 6.905827E-04  1.289303E-03  7.485397E-04  1.180639E-03  6.848187E-04
 2.409294E-04  6.485939E-04  3.999771E-04  6.329166E-04  5.275019E-04
 3.719266E-04  2.359282E-04  8.071988E-04  2.701662E-04  6.670883E-04
 1.538525E-03  1.686220E-03  2.100862E-03  9.130201E-04  8.704552E-04
 8.502701E-04  4.430830E-04  1.668918E-03  8.980936E-04  2.380146E-04
 1.193211E-03  4.546899E-04  1.172791E-03  1.934487E-03  6.293361E-04
-2.253867E-04  1.710919E-03  1.658688E-03  2.209494E-03  3.032127E-04
 5.505222E-04  5.065812E-04  5.746219E-04  6.207397E-04  1.548970E-03
 1.416268E-03  7.683847E-04  2.563113E-03  1.305293E-03  9.570906E-04
 1.731522E-03  5.057083E-04  2.453513E-04  1.747015E-04  5.527029E-04
 5.433562E-04  6.206639E-04  6.277530E-04  6.703657E-04  5.820032E-04
 4.925180E-04  6.610006E-04  5.118483E-04  9.913894E-04  5.309230E-04
 6.149273E-04  7.776549E-04  9.787851E-04  1.048909E-03  1.207425E-03
 7.316338E-04  8.631854E-04  9.992484E-04  1.022619E-03  1.430072E-03
 7.546633E-04  7.810938E-04  9.917767E-04  1.334629E-03  6.098780E-04
 7.364101E-04  8.410845E-04  1.040464E-03  1.274770E-03  1.623622E-03
 1.961246E-03  2.139706E-03  2.170640E-03  2.208168E-03  2.186356E-03
 2.159295E-03  1.970835E-03  1.656340E-03  1.347787E-03  2.268713E-03
 1.446472E-03  1.195007E-03  9.137678E-04  1.554992E-03  1.416406E-03
 1.250617E-03  1.333341E-03  1.084074E-03  6.208760E-04  6.374314E-04
 1.520393E-03  1.303509E-03  1.300696E-03  1.122529E-03  6.883067E-04
 6.974777E-04  7.486881E-04  9.267605E-04  1.016119E-03  1.337179E-03
```

```
 1.601901E-03  1.714987E-03  2.065219E-03  2.269599E-03  3.267892E-03
 4.496744E-03  1.651763E-03  1.099488E-03  1.646667E-03  8.793735E-04
 1.613625E-03  1.724868E-03 -3.355957E-04 -3.661996E-05 -2.678600E-03
-4.340224E-04 -5.189590E-04 -1.363285E-04 -1.360189E-04 -9.593642E-05
-5.709878E-05 -6.221425E-05 -3.386675E-05 -2.253157E-05 -2.094053E-05
-1.934127E-05 -2.462690E-05 -2.170317E-06
 0.000000E+00  2.166573E-07  4.563122E-07  5.797737E-07  1.052550E-06
 5.227220E-06  1.635858E-05  5.783009E-05  1.976834E-04  1.499162E-04
 7.129665E-04  6.958902E-04  2.559530E-04  9.831330E-04  9.627389E-04
 3.887023E-04  1.964374E-04  1.556201E-04  2.896345E-04  2.635319E-04
 4.911378E-04  4.089002E-04  4.619455E-04  1.526254E-04  1.964684E-04
 1.289375E-04  3.598896E-04  6.532479E-04  6.485019E-04  1.182846E-04
 6.882796E-04  3.167530E-04  2.880858E-04  6.021730E-04  3.712296E-04
 3.285334E-04  2.205738E-04  2.868579E-04  1.144571E-03  1.210866E-03
 1.630818E-03  1.478321E-03  8.691841E-04  1.300305E-03  8.875877E-04
 2.513203E-04  5.308349E-04  2.038752E-04  1.063591E-03  8.168587E-04
 2.744358E-04  6.274046E-04  1.948488E-03  9.865571E-04  1.823805E-03
 1.338482E-03  1.511324E-03  9.038291E-04  1.457042E-03  1.918277E-03
 2.801630E-04  1.096534E-03  7.956573E-04  6.607744E-04  2.723250E-04
 1.036605E-03  8.099615E-04  2.349491E-04  1.322119E-03  1.016182E-03
 1.612093E-03  1.628536E-04  1.052985E-03  2.383268E-03  3.968910E-04
 8.649064E-04  7.028914E-04  1.137342E-04  4.472456E-04  2.406079E-04
 1.744864E-03  1.777447E-04  2.259000E-04  3.594763E-04  4.173333E-04
 5.448412E-04  4.939820E-04  1.791260E-04  3.091145E-04  3.135670E-04
 3.681154E-04  2.324528E-04  3.615673E-04  2.188618E-04  5.191086E-04
 1.675413E-04  1.898112E-04  1.855286E-04  1.450861E-04  1.870526E-04
 2.471996E-04  1.715434E-04  2.386457E-04  2.054351E-04  2.281650E-04
 1.131494E-04  1.943583E-04  1.879164E-04  2.617881E-04  1.818599E-04
```

(continues on next page)

(continued from previous page)

```
 1.801841E-04  1.264961E-04  2.859284E-04  1.303168E-04  2.619164E-04
 4.820038E-04  5.051208E-04  5.569755E-04  3.780011E-04  2.941432E-04
 3.091053E-04  1.882628E-04  4.452257E-04  3.532799E-04  3.328100E-04
 2.590273E-04  2.741619E-04  3.632040E-04  4.282485E-04  2.063481E-04
 6.650581E-04  4.944312E-04  5.869004E-04  5.798635E-04  1.535210E-04
 2.230120E-04  2.313091E-04  2.422020E-04  2.482344E-04  5.418594E-04
 4.144033E-04  2.981307E-04  8.099135E-04  3.777744E-04  3.503435E-04
 5.259281E-04  1.995055E-04  1.282225E-04  9.399701E-05  2.180021E-04
 2.204282E-04  2.289391E-04  2.353957E-04  2.476291E-04  2.310753E-04
 2.197705E-04  2.479305E-04  2.547174E-04  3.352607E-04  2.009506E-04
 2.600849E-04  2.930110E-04  3.068041E-04  3.205445E-04  3.365811E-04
 2.213007E-04  2.267429E-04  2.309969E-04  2.360546E-04  2.429477E-04
 1.488554E-04  1.501431E-04  1.523043E-04  1.490992E-04  7.836059E-05
 7.676734E-05  7.504028E-05  7.123577E-05  6.306674E-05  5.374799E-05
 4.217917E-05  3.237631E-05  2.594829E-05  2.435751E-05  2.680074E-05
 3.477155E-05  4.508479E-05  5.790404E-05  6.986748E-05  1.487892E-04
 1.707188E-04  1.793168E-04  1.919402E-04  3.358301E-04  3.615817E-04
 3.865266E-04  4.114530E-04  4.392739E-04  2.714958E-04  2.801094E-04
 5.011157E-04  5.390380E-04  5.729596E-04  5.910309E-04  3.334616E-04
 3.061873E-04  2.661426E-04  2.248651E-04  2.308670E-04  2.803224E-04
 3.708528E-04  4.862559E-04  6.188882E-04  8.189344E-04  1.110189E-03
```

```
  1.577516E-03  1.075698E-03  1.270736E-03  1.520277E-03  1.828479E-03
  2.146210E-03  2.457027E-03  2.729725E-03  1.737714E-03  2.854629E-03
  7.541633E-04  5.796042E-04  2.197686E-04  1.895209E-04  9.598149E-05
  8.502876E-05  7.599381E-05  4.814730E-05  3.389918E-05  3.443490E-05
  3.153711E-05  3.212018E-05  8.655023E-06

...

file verification information
 code system:    scale
 version:    6.2
 program:  sams
   creation date:  01_apr_2016
   library:  /home/c53/scale_freeze/Linux_x86_64-release/bin/scale
   migration code:  sams
   version:  6.2.0
 jobname:  c53
 machine name:  node23
 date of file creation:  01_apr_2016
 time of file creation:  17:34:23.02
 filename:  tsunami-3d3.sdf
 chi sensitivities are constrained
```

Format of HDF5-based sensitivity data file

An HDF5-based format for the sensitivity data files has been introduced. The format used here has been found to improve I/O speed for TSUNAMI-IP cases that read many SDF files. The HDF5-based SDF can be loaded in Fulcrum to produce the same plots that the text-based SDF can. The HDF5-based SDF may also be read with standard HDF5 tools such as h5dump, h5py, and HDFview.

The tao utility includes a feature that facilitates conversion between the HDF5-based SDF and the text-based SDF.

For example, the tao utility may be used to convert a batch of text-based SDF files in a directory into HDF5-based SDF files by:

```
tao convert *.sdf
```

Each HDF5-based SDF file has a file extension ".sdf.h5". The tao convert tool may also be used to convert from HDF5 to the text-based format via

```
tao convert -fromFormat=sdf.h5 -toFormat=sdf <filename>.sdf.h5
```

For tao convert, the `-fromFormat` and `-toFormat` options take the intended *file extension* as the arguments.

The HDF5-based SDF may be requested as output from TSUNAMI calculations by including the `sensitivity_format` keyword in the SAMS block of the calculation input. For example:

```
read sams
  sensitivity_format=hdf5
end sams
```

will produce the result senstivity file in HDF5 format, with extension ".sdf.h5". The default sensitivity output for TSUNAMI calculations is the text-based format described previously. This may be requested by removing the `sensitivity_format` keyword from the input file, or by

```
read sams
  sensitivity_format=txt
end sams
```

Brief Description of Data Layout

The HDF5 attributes are defined as

Table 6.3.8: HDF5-based SDF attributes

Name	Datatype	Comments: *MC* denotes "only in Monte Carlo-derived files"
`file_type`	string	The type of file - should be `SDF_ARCHIVE`
`file_version`	string	The version of the SDF HDF5 format used for this file
`id_chk_sum`	string	A simple checksum
`id_spec`	string	A link to the specification
`title`	string	The SDF title
`sdf_type`	enumeration	"MonteCarlo" or "Deterministic"
`reference_quantity`	string	The quantity of interest, e.g. k_{eff}
`reference_comment`	string	Any comment about the reference quantity
`reference_value`	double	The value of the reference quantity in the transport calculation
`reference_sigma`	double	*MC* Uncertainty in the reference value
`number_profiles`	int	The number of sensitivity profiles
`number_region_integrated`	int	The number of region-integrated sensitivity profiles
`number_neutron_groups`	int	The number of neutron groups for the sensitivity calculation
`constrained_chi`	bool	Whether the χ (fission neutron spectrum, MT=1018) sensitivities are constrained
`list_energies`	vector<double>	Energy group boundaries, with size = `number_neutron_groups` + 1

The datasets are stored as vectors, each of size `number_profiles`. The exceptions are `profile_values` and `profile_sigmas` – in order to optimize I/O time (especially in TSUNAMI-IP and TSURFER calculations that read many SDFs), both of these datasets are *hyperslabs* of dimension `number_profiles` ×

number_neutron_groups.

Table 6.3.9: SDF datasets

Name	Datatype	Comments: *MC* denotes "only present in Monte Carlo-derived files"
profile_values	double	The sensitivity coefficients
profile_sigmas	double	*MC* The uncertainties in the sensitivity coefficients
nuclide_id	int	The numerical IDs of the nuclides, in ZZAAA form
mt	int	The reaction ID (MT) values
comment	string	Any comment
region	int	Same conventions as text-based region values
region_uses	int	Number of times region has been used
material	int	Same conventions as text-based material values
unit	int	Same conventions as text-based unit values
volume	double	Volume of region
values_sum	double	Energy-integrated value of the sensitivity coefficients for corresponding profile
values_sigma	double	*MC* Uncertainty in energy-integrated value of the sensitivity coefficients for corresponding profile
values_abs_sum	double	Sum of *absolute values* of the sensitivity coefficients for corresponding profile
values_osc_sum	double	Sum of the sensitivity coefficients *with sign opposite* values_sigma for corresponding profile
values_osc_sigma	double	*MC* Uncertainty in values_osc_sum for corresponding profile

An example of the output of h5dump --header CE-TSUNAMI_test_k6_IFP_godiva.sdf.h5:

```
HDF5 "CE-TSUNAMI_test_k6_IFP_godiva.sdf.h5" {
GROUP "/" {
 ATTRIBUTE "constrained_chi" {
    DATATYPE  H5T_ENUM {
       H5T_STD_I8LE;
       "FALSE"            0;
       "TRUE"             1;
    }
    DATASPACE  SCALAR
 }
 ATTRIBUTE "file_type" {
    DATATYPE  H5T_STRING {
       STRSIZE 12;
       STRPAD H5T_STR_NULLTERM;
       CSET H5T_CSET_ASCII;
       CTYPE H5T_C_S1;
    }
    DATASPACE  SCALAR
 }
 ATTRIBUTE "file_version" {
    DATATYPE  H5T_STRING {
       STRSIZE 6;
       STRPAD H5T_STR_NULLTERM;
       CSET H5T_CSET_ASCII;
       CTYPE H5T_C_S1;
    }
```

(continues on next page)

(continued from previous page)

```
   DATASPACE  SCALAR
}
ATTRIBUTE "id_chk_sum" {
   DATATYPE  H5T_STRING {
      STRSIZE 12;
      STRPAD H5T_STR_NULLTERM;
      CSET H5T_CSET_ASCII;
      CTYPE H5T_C_S1;
   }
   DATASPACE  SCALAR
}
ATTRIBUTE "id_spec" {
   DATATYPE  H5T_STRING {
      STRSIZE 12;
      STRPAD H5T_STR_NULLTERM;
      CSET H5T_CSET_ASCII;
      CTYPE H5T_C_S1;
   }
   DATASPACE  SCALAR
}
ATTRIBUTE "list_energies" {
   DATATYPE  H5T_IEEE_F64LE
   DATASPACE  SIMPLE { ( 253 ) / ( 253 ) }
}
ATTRIBUTE "number_neutron_groups" {
   DATATYPE  H5T_STD_U64LE
   DATASPACE  SCALAR
}
ATTRIBUTE "number_profiles" {
   DATATYPE  H5T_STD_U64LE
   DATASPACE  SCALAR
}
ATTRIBUTE "number_region_integrated" {
   DATATYPE  H5T_STD_U64LE
   DATASPACE  SCALAR
}
ATTRIBUTE "reference_comment" {
   DATATYPE  H5T_STRING {
      STRSIZE 28;
      STRPAD H5T_STR_NULLTERM;
      CSET H5T_CSET_ASCII;
      CTYPE H5T_C_S1;
   }
   DATASPACE  SCALAR
}
ATTRIBUTE "reference_quantity" {
   DATATYPE  H5T_STRING {
      STRSIZE 5;
      STRPAD H5T_STR_NULLTERM;
      CSET H5T_CSET_ASCII;
      CTYPE H5T_C_S1;
   }
   DATASPACE  SCALAR
}
ATTRIBUTE "reference_sigma" {
   DATATYPE  H5T_IEEE_F64LE
   DATASPACE  SCALAR
}
ATTRIBUTE "reference_value" {
   DATATYPE  H5T_IEEE_F64LE
   DATASPACE  SCALAR
}
ATTRIBUTE "sdf_type" {
   DATATYPE  H5T_STRING {
      STRSIZE 11;
```

(continues on next page)

(continued from previous page)

```
      STRPAD H5T_STR_NULLTERM;
      CSET H5T_CSET_ASCII;
      CTYPE H5T_C_S1;
   }
   DATASPACE  SCALAR
}
ATTRIBUTE "title" {
   DATATYPE  H5T_STRING {
      STRSIZE 37;
      STRPAD H5T_STR_NULLTERM;
      CSET H5T_CSET_ASCII;
      CTYPE H5T_C_S1;
   }
   DATASPACE  SCALAR
}
DATASET "comment" {
   DATATYPE  H5T_STRING {
      STRSIZE H5T_VARIABLE;
      STRPAD H5T_STR_NULLTERM;
      CSET H5T_CSET_ASCII;
      CTYPE H5T_C_S1;
   }
   DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
}
DATASET "material" {
   DATATYPE  H5T_STD_I32LE
   DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
}
DATASET "mt" {
   DATATYPE  H5T_STD_I32LE
   DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
}
DATASET "nuclide_id" {
   DATATYPE  H5T_STD_I32LE
   DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
}
DATASET "profile_sigmas" {
   DATATYPE  H5T_IEEE_F64LE
   DATASPACE  SIMPLE { ( 54, 252 ) / ( 54, 252 ) }
}
DATASET "profile_values" {
   DATATYPE  H5T_IEEE_F64LE
   DATASPACE  SIMPLE { ( 54, 252 ) / ( 54, 252 ) }
}
DATASET "region" {
   DATATYPE  H5T_STD_I32LE
   DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
}
DATASET "region_uses" {
   DATATYPE  H5T_STD_I32LE
   DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
}
DATASET "unit" {
   DATATYPE  H5T_STD_I32LE
   DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
}
DATASET "values_abs_sum" {
   DATATYPE  H5T_IEEE_F64LE
   DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
}
DATASET "values_osc_sigma" {
   DATATYPE  H5T_IEEE_F64LE
   DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
}
DATASET "values_osc_sum" {
```

(continues on next page)

(continued from previous page)

```
    DATATYPE  H5T_IEEE_F64LE
    DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
 }
 DATASET "values_sigma" {
    DATATYPE  H5T_IEEE_F64LE
    DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
 }
 DATASET "values_sum" {
    DATATYPE  H5T_IEEE_F64LE
    DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
 }
 DATASET "volume" {
    DATATYPE  H5T_IEEE_F64LE
    DATASPACE  SIMPLE { ( 54 ) / ( 54 ) }
 }
}
}
```

Format of SENLIB.SEN data file

The SENLIB.SEN data file is generated by the SENLIB routines used in the TSUNAMI modules to compute implicit sensitivity effects. The data file contains the sensitivity of data computed in SENLIB and input to the resonance processing code BONAMIST. This file also contains the sensitivities of the extra cross sections input to BONAMIST to the number densities of each nuclide for each region of the BONAMIST input. In BONAMIST, the extra cross section contains data for the Dancoff factor. Each of the parameters for which sensitivity coefficients are computed has a unique identification number. The parameter identification numbers, corresponding parameters, and their locations in the resonance processing code input are shown in Table 6.3.10. The format of the records of the SENLIB.SEN data file is given in Table 6.3.11. Each sensitivity coefficient is identified by three records. These records are repeated until all sensitivities are listed on the data file.

Table 6.3.10: Parameter identifiers for SENLIB.SEN data file.

Identifier number	Parameter	Location of parameter in resonance processing code input
-1004	Extra cross section (cm^{-1}) by zone, account for Dancoff factor in BONAMIST	9* array in data block 2 of BONAMIST input

Table 6.3.11: Record format for SENLIB.SEN data file.

Record number	Data	Format	Description
1	Region for BONAMIST data and parameter	2i12	BONAMIST region and parameter for which sensitivity coefficient is computed.
2	Nuclide and reaction MT number to which sensitivity coefficient is computed	2i12	Nuclide and reaction MT number to which the sensitivity of the parameter is computed. For data currently computed by SENLIB, only the sensitivities of the parameters to the number densities of the nuclides are computed. Thus, the reaction MT number is always 1.

continues on next page

Table 6.3.11 – continued from previous page

3	Sensitivity coefficient	es14.6	Relative sensitivity of parameter identified in first record to nuclide and reaction identified in second record.

Format of BONAMIST.SEN data file

The BONAMIST.SEN data file is generated by the BONAMIST code, which is executed as part of the TSUNAMI-1D sequence. This data file contains the sensitivities of the groupwise cross sections that are resonance shielded in BONAMIST to the certain data input to BONAMIST. The data to which the sensitivity coefficients are computed are the number densities of the nuclides and the extra cross section by region, which contains data for the Dancoff factor. These data are listed in the data file according to the identifiers listed in Table 6.3.10. Additionally, the sensitivity of group cross sections to the number density of a particular nuclide is identified with the nuclide number from the AMPX cross-section data library.

The format of the records in the data file is identified in Table 6.3.12. Blocks of these records are printed for the non-zero sensitivities for all nuclides and reactions for which resolved resonances are processed.

Table 6.3.12: Record format for BONAMIST.SEN data file.

Record number	Data	Format	Description
1	Nuclide and MT number for cross sections for which sensitivity coefficients are computed	2i12	Nuclide identifier and MT number of reaction for which sensitivity data are computed.
2	Parameter or nuclide and the MT number to which the sensitivity data are computed	2i12	Parameter identifier for scattering cross sections and Dancoff factor or nuclide identifier and MT number to which the sensitivity coefficients are computed. If a parameter identifier is used, the MT number is set to 1. Currently only the sensitivities to number densities are computed in BONAMIST, so the only MT number printed is 1.
3	First and last group for which sensitivity coefficients are printed	2i12	First and last group for the sensitivity coefficients that follow.
4	Sensitivity coefficients, record repeats until all groups identified by record 3 are printed	5es14.6	Relative sensitivities of cross sections for nuclide and process identified by record 1 to parameter or nuclide identified in record 2 for groups corresponding to those identified by record 3.

COVERX format

The COVERX data file format for storing multigroup cross-section uncertainty information is presented in this section. The COVERX format was used with the FORSS code system at ORNL. The covariance data processing code PUFF-III outputs multigroup data in the COVERX format. This the only covariance format available for use with the sensitivity and uncertainty analysis codes in SCALE.

The overall structure of a COVERX file is given in Table 6.3.13. The occurrence of the particular record type, the name of the record type and when a given record type is present on the file are identified.

Table 6.3.13: COVERX file structure.

Occurrence	Record type	Present if
Once at beginning of file	File Identification	Always
	File Control	Always
	File Description	NHOLL > 0
	Neutron Group Boundaries	NNGRUP > 0
	Gamma Group Boundaries	NGGRUP > 0
	Material-Reaction Control	Always
Repeat for all material-reaction type pairs	Material-Reaction Type Cross Sections and Error Files	Always
Repeat for all matrices	Matrix Control	Always
	Block Control	Always
Repeat for all blocks	Matrix Data	Always

The details of each record type identified in Table 6.3.13 are presented in Table 6.3.14. Here, the name of the record type, the variables present on the record, the length of the record, the format of the record in FORTRAN context and a description of the content of the record are given.

continues on next page

Table 6.3.14 – continued from previous page

Table 6.3.14: Description of records on COVERX formatted file.

Record type	Variables	Length	Format		Description
File Identification	HNAME, (HUSE(I), I=1,2), IVERS	1+3*MULT	(' 0V COVERX ', A6, '*', 2A6, '*', I6)	HNAME	File name (A6)
				HUSE	User identification (A6)
				IVERS	File version number
				MULT	1 – A6 is a single precision word
					2 – A6 is double precision word
File Control	NGROUP, NNGRUP, NGGRUP, NTYPE, NMMP, NMTRIX, NHOLL	7	(' 1D ', 7I6)	NGROUP	Number of energy groups
				NNGRUP	Number of neutron groups
				NGGRUP	Number of gamma groups
				NTYPE	Type of Data
					1 – Covariance matrix, standard deviation
					2 – Relative covariance matrix, relative standard deviation
					3 – Correlation matrix, standard deviation
				NMMP	Number of MAT-MT pairs
				NMTRIX	Number of matrices
				NHOLL	Number of Hollerith words in description
File Description	(WORDS(J), J=1,NHOLL)	MULT*NHOLL	(' 2D ','*',11A6/(11A6))	WORDS(J)	Hollerith description of file
Neutron Group Boundaries	(GPBN(J), J=1,NNGRUP), ENMIN	NNGRUP+1	(' 3D ',5E12.4/(6E12.4))	GPBN(J)	Maximum energy bound of neutron group (J) (eV)
				ENMIN	Minimum energy of neutron in energy range
Gamma Group Boundaries	(GPBG(J), J=1,NGGRUP), EGMIN	NGGRUP+1	(' 4D ',5E12.4/(6E12.4))	GPBG(J)	Maximum energy bound of gamma group (J) (eV)
				EGMIN	Minimum energy of neutron in energy range

Record type	Variables	Length	Format		Description
Material-Reaction Control	(MATID(I), MTID(I), MWGT(I), I=1,NMMP)	3*NMMP	(' 5D ',11I6,(12I6))	MATID(I)	Material identification number
				MTID(I)	Reaction type identification number
				MWGT(I)	Cross section weighting option
					1 – Constant
					2 – 1/E
					3 – Thermal + 1/E + Fission
					4 – Arbitrary
					5 – Combined CTR CRBR
Material-Reaction Type Cross Sections and Error Files	(CRS(J), J=1,NGROUP), (ERROR(J), J=1,NGROUP)	2*NGROUP	(' 6D ',5E12.4/(6E12.4))	CRS	Cross section
				ERROR	Standard deviation
Matrix Control	MAT1, MT1, MAT2, MT2, NBLOK	5	(' 7D ',5I6)	MAT1	Material 1 identification number
				MT1	Reaction type 1 identification number
				MAT2	Material 2 identification number
				MT2	Reaction type 2 identification number
				NBLOK	Number of blocks into which matrix is divided
Block Control	(JBAND(J), IJJ(J), J=1,NGROUP), (LGRP(N), N=1,NBLOK)	2*NGROUP*NBLOK	(' 8D ',11I6/(12I6))	JBAND(J)	Bandwidth for group J
				IJJ(J)	Position of diagonal element for group J
				LGRP(N)	Number of groups in block(N)
Matrix Data	(COV(K), K=1,KMAX)	KMAX	(' 9D ',5E12.4/(6E12.4))	COV	NTYPE matrix data

continues on next page

Table 6.3.14 – continued from previous page

HTML colors

Colors available for use in HTML formatted output are shown in Table 6.3.15. This table was created from data available at http://www.w3schools.com/css/css_colornames.asp. Color names are not case sensitive.

Table 6.3.15: Colors available in HTML.

Color name	Color HEX	Color
AliceBlue	#F0F8FF	
AntiqueWhite	#FAEBD7	
Aqua	#00FFFF	
Aquamarine	#7FFFD4	
Azure	#F0FFFF	
Beige	#F5F5DC	
Bisque	#FFE4C4	
Black	#000000	
BlanchedAlmond	#FFEBCD	
Blue	#0000FF	
BlueViolet	#8A2BE2	
Brown	#A52A2A	
BurlyWood	#DEB887	
CadetBlue	#5F9EA0	
Chartreuse	#7FFF00	
Chocolate	#D2691E	
Coral	#FF7F50	
CornflowerBlue	#6495ED	
Cornsilk	#FFF8DC	
Crimson	#DC143C	
Cyan	#00FFFF	
DarkBlue	#00008B	
DarkCyan	#008B8B	
DarkGoldenRod	#B8860B	
DarkGray	#A9A9A9	
DarkGreen	#006400	
DarkKhaki	#BDB76B	
DarkMagenta	#8B008B	
DarkOliveGreen	#556B2F	
Darkorange	#FF8C00	
DarkOrchid	#9932CC	
DarkRed	#8B0000	

Table 6.3.16: Colors available in HTML (continued).

Color name	Color HEX	Color
DarkSalmon	#E9967A	
DarkSeaGreen	#8FBC8F	
DarkSlateBlue	#483D8B	
DarkSlateGray	#2F4F4F	
DarkTurquoise	#00CED1	
DarkViolet	#9400D3	
DeepPink	#FF1493	
DeepSkyBlue	#00BFFF	
DimGray	#696969	
DodgerBlue	#1E90FF	
Feldspar	#D19275	
FireBrick	#B22222	
FloralWhite	#FFFAF0	
ForestGreen	#228B22	
Fuchsia	#FF00FF	
Gainsboro	#DCDCDC	
GhostWhite	#F8F8FF	
Gold	#FFD700	
GoldenRod	#DAA520	
Gray	#808080	
Green	#008000	
GreenYellow	#ADFF2F	
HoneyDew	#F0FFF0	
HotPink	#FF69B4	
IndianRed	#CD5C5C	
Indigo	#4B0082	
Ivory	#FFFFF0	
Khaki	#F0E68C	
Lavender	#E6E6FA	
LavenderBlush	#FFF0F5	
LawnGreen	#7CFC00	
LemonChiffon	#FFFACD	
LightBlue	#ADD8E6	
LightCoral	#F08080	
LightCyan	#E0FFFF	
LightGoldenRodYellow	#FAFAD2	
LightGrey	#D3D3D3	
LightGreen	#90EE90	

Table 6.3.17: Colors available in HTML (continued).

Color name	Color HEX	Color
LightPink	#FFB6C1	
LightSalmon	#FFA07A	
LightSeaGreen	#20B2AA	
LightSkyBlue	#87CEFA	
LightSlateBlue	#8470FF	
LightSlateGray	#778899	
LightSteelBlue	#B0C4DE	
LightYellow	#FFFFE0	
Lime	#00FF00	
LimeGreen	#32CD32	
Linen	#FAF0E6	
Magenta	#FF00FF	
Maroon	#800000	
MediumAquaMarine	#66CDAA	
MediumBlue	#0000CD	
MediumOrchid	#BA55D3	
MediumPurple	#9370D8	
MediumSeaGreen	#3CB371	
MediumSlateBlue	#7B68EE	
MediumSpringGreen	#00FA9A	
MediumTurquoise	#48D1CC	
MediumVioletRed	#C71585	
MidnightBlue	#191970	
MintCream	#F5FFFA	
MistyRose	#FFE4E1	
Moccasin	#FFE4B5	
NavajoWhite	#FFDEAD	
Navy	#000080	
OldLace	#FDF5E6	
Olive	#808000	
OliveDrab	#6B8E23	
Orange	#FFA500	
OrangeRed	#FF4500	
Orchid	#DA70D6	
PaleGoldenRod	#EEE8AA	
PaleGreen	#98FB98	
PaleTurquoise	#AFEEEE	
PaleVioletRed	#D87093	

Table 6.3.18: Colors available in HTML (continued).

Color name	Color HEX	Color
PapayaWhip	#FFEFD5	
PeachPuff	#FFDAB9	
Peru	#CD853F	
Pink	#FFC0CB	
Plum	#DDA0DD	
PowderBlue	#B0E0E6	
Purple	#800080	
Red	#FF0000	
RosyBrown	#BC8F8F	
RoyalBlue	#4169E1	
SaddleBrown	#8B4513	
Salmon	#FA8072	
SandyBrown	#F4A460	
SeaGreen	#2E8B57	
SeaShell	#FFF5EE	
Sienna	#A0522D	
Silver	#C0C0C0	
SkyBlue	#87CEEB	
SlateBlue	#6A5ACD	
SlateGray	#708090	
Snow	#FFFAFA	
SpringGreen	#00FF7F	
SteelBlue	#4682B4	
Tan	#D2B48C	
Teal	#008080	
Thistle	#D8BFD8	
Tomato	#FF6347	
Turquoise	#40E0D0	
Violet	#EE82EE	
VioletRed	#D02090	
Wheat	#F5DEB3	
White	#FFFFFF	
WhiteSmoke	#F5F5F5	
Yellow	#FFFF00	
YellowGreen	#9ACD32	

6.4 SAMPLER: STATISTICAL UNCERTAINTY ANALYSIS WITH SCALE SEQUENCES

U. Mertyurek, W. A. Wieselquist, F. Bostelmann, M. L. Williams, F. Havlůj[1], *R. A. Lefebvre, W. Zwermann*[2], *D. Wiarda, M. T. Pigni, I. C. Gauld, M. A. Jessee, J. P. Lefebvre, K. J. Dugan*[3], *and B. T. Rearden*

ABSTRACT

Sampler is a "super-sequence" that performs general uncertainty analysis for SCALE sequences by statistically sampling the input data and analyzing the output distributions for specified responses. Among the input parameters that can be sampled are multigroup nuclear data, resonance self-shielding data (shielding factors and CENTRM pointwise cross sections), depletion data such as fission product yields and decay data, and model parameters such as nuclide concentrations, temperatures, and simple dimension specifications. Random perturbation factors for nuclear cross sections and depletion data are pre-computed with the XSUSA module Medusa by sampling covariance information and are stored in libraries read during the Sampler execution, while model parameters are sampled "on the fly". A wide variety of output response types for virtually all SCALE sequences can be specified for the uncertainty analysis, and correlations in uncertain parameters between multiple systems are also generated.

ACKNOWLEDGMENTS

Contributions from the Gesellschaft fur Anlagen- und Reaktorsicherheit (GRS) in Germany are gratefully acknowledged. The development of the SCALE Sampler module is based on GRS's suggestion that their XSUSA code could be used in conjunction with SCALE for stochastic uncertainty calculations. The original Sampler sequence was developed based on the XSUSA sampling sequence as well as collaboration and knowledge exchange with GRS staff members. The GRS module Medusa is used to generate perturbations of the MG cross sections, fission yields, and decay data.

The U.S. Nuclear Regulatory Commission Office of Nuclear Regulatory Research, the U.S. DOE Nuclear Fuel Storage and Transportation Planning Project, and the U.S. DOE Nuclear Criticality Safety Program supported the development of Sampler.

6.4.1 INTRODUCTION

The SCALE nuclear analysis code system provides a unified set of computational tools and data libraries to address a wide range of applications, including criticality safety, reactor physics, spent fuel characterization, burnup credit, national security, and neutron/photon radiation shielding [SAMPLER-RWJ+11]. In addition to determining the problem solutions, SCALE also provides tools to compute uncertainties in the results, arising from uncertainties in the data used for the calculations. Due to the diverse types of computational methods in SCALE, robust sensitivity/uncertainty (S/U) methods are necessary. Sampler implements stochastic sampling of uncertain parameters that can be applied to any type of SCALE calculation, propagating uncertainties throughout a computational sequence. Sampler treats uncertainties from two sources: 1) nuclear data and 2) input parameters. Sampler generates the uncertainty in any result generated by the computational sequence through stochastic means by repeating numerous passes through the computational sequence, each with a randomly perturbed sample of the requested uncertain quantities. The mean value and uncertainty in each parameter is reported along with the correlation in uncertain parameters where multiple systems are simultaneously sampled with correlated uncertainties.

Used in conjunction with nuclear data covariances available in SCALE, Sampler is a general, overarching sequence for obtaining uncertainties for many types of applications. SCALE includes covariances for

[1] ÚJV Řež, a. s., Czech Republic
[2] Gesellschaft fur Anlagen- und Reaktorsicherheit (GRS)
[3] Texas A&M University

multigroup neutron cross-section data, as well as for fission product yields and radioactive decay data, which allows uncertainty calculations to be performed for most multigroup (MG) computational sequences in SCALE. At the present time, nuclear data sampling cannot be applied to SCALE continuous energy (CE) Monte Carlo calculations (i.e., CE-KENO and CE-Monaco), although the fundamental approach is still valid.

Used in conjunction with uncertainties in input data, Sampler can determine the uncertainties and correlations in computed results due to uncertainties in dimensions, densities, distributions of material compositions, temperatures, or many other quantities that are defined in the user input for any SCALE computational sequence. This methodology was especially developed to produce uncertainties and correlations in criticality safety benchmark experiments, but it has a wide range of applications in numerous scenarios in nuclear safety analysis and design. The input sampling capabilities of Sampler also include a parametric capability to determine the response of a system to a systematic variation of an arbitrary number of input parameters.

6.4.1.1 Uncertainty analysis with stochastic versus perturbation methods

Two quite different approaches may be used for uncertainty analysis. One method uses first order perturbation theory expressions to compute sensitivity coefficients for a given response. This requires performing a forward transport calculation for the specified system and (sometimes) adjoint calculations for each response of interest. After the forward and adjoint transport solutions are obtained, sensitivity coefficients for all nuclear cross sections and material concentrations can be computed very efficiently with perturbation theory [SAMPLER-RWJ+11]. The sensitivities may be folded with covariance matrices to obtain response uncertainties due to nuclear data. The TSUNAMI modules and sequences in SCALE use perturbation theory for S/U analysis in this manner (see TSUNAMI-1D and TSUNAMI-3D).

For some types of applications, the adjoint-based perturbation methodology is not adequate or is inefficient. These include:

(a) Cases requiring codes with no adjoint functionality. SCALE has capability for critical eigenvalue adjoint solutions and generalized adjoint calculations using XSDRN, NEWT or KENO, but adjoint methods are not currently available for coupled neutronics-depletion calculations.

(b) Cases for which first order perturbation theory is not valid (i.e., problems with significant second order effects).

The Sampler module described in this section provides an alternative method for uncertainty analysis based on stochastic sampling (Monte Carlo) and does not require adjoint calculations. This approach samples joint probability density functions (PDFs) — such as given in the SCALE nuclear data covariance library — to produce a random sample for the nuclear cross sections used a transport calculation. If PDFs are available for other parameters such as depletion data or model parameters, etc., then these too can be sampled and included in the perturbed input vector. The perturbed data vector can be input to any SCALE sequence or functional module to obtain a single forward solution for all desired perturbed responses. The process is repeated for the desired number of samples-typically a few hundred; and the output distributions of results are analyzed to obtain standard deviations and correlation coefficients for all responses. The stochastic sampling method is not restricted to current SCALE modules; any new sequences or codes can be used for the forward calculations, without having to develop the capability for adjoint calculations.

Output distributions from the SCALE sampling also may be propagated to downstream codes for follow-on uncertainty analysis. For example, input for the TRITON lattice physics sequence can be sampled to produce a random set of output assembly-averaged, two-group cross section libraries. The two-group libraries can be input to a 3D core simulator that performs steady-state or transient calculations, and statistical analysis of the simulator output provides response uncertainties (possibly time-dependent) due to the SCALE input data

uncertainties. Response uncertainties computed with this approach are not limited to first order accuracy; i.e., they account for all non-linearities and discontinuities with the same accuracy as the original codes.

Thus there are several advantages to the statistical sampling method because it requires only forward calculations. The typical Sampler computational procedure perturbs the entire input data vector simultaneously, so that the total uncertainty in all responses, due to all data uncertainties, is obtained. This standard approach does not provide individual data sensitivity coefficients, unlike the perturbation theory method. In this sense, the statistical sampling method is complementary to the adjoint-based sensitivity method in the TSUNAMI modules. Computation of sensitivities using only forward calculations requires that each input parameter be varied individually, rather than collectively; and this may require a large number of simulations to obtain a full set of sensitivity coefficients.

6.4.2 METHODOLOGY

The main components of a Sampler calculation are the procedures for perturbing input data, obtaining the desired responses, and performing statistical analysis of the output distributions.

6.4.2.1 Definition of input data perturbations

The input data for a SCALE computation will generally be one of three types:

(a) *Nuclear data for transport calculations*. This includes multigroup (MG) and continuous energy (CE) cross sections, multiplicities, secondary particle distributions, and data used for resonance self-shielding of MG cross sections.

(b) *Nuclide transmutation data for depletion and burnup calculations*. This includes fission product yield data, decay constants, branching ratios to excited states, decay energies and distributions.

(c) *Modeling parameters for the system*. This includes information for defining nuclide number densities (e.g., density, weight fractions, enrichment, void fraction, etc.), temperature, and dimensions.

In principle Sampler can perform uncertainty analysis for all the above types of input data if uncertainties and correlations are known. The main restriction at this time is that CE cross sections for Monte Carlo calculations are not sampled (although the continuous data used for self-shielding *are* treated), so data perturbation applications are limited to MG calculations. Perturbations to input number densities and model dimensions are not impacted by this data limitation.

6.4.2.2 Nuclear data perturbations for multigroup calculations

Input MG nuclear data for SCALE sequences are obtained from an AMPX Master formatted library, which contains infinitely-dilute one-dimensional (1D) cross sections, two-dimensional (2D) scattering distributions, and Bondarenko self-shielding factors for various types of reactions. Only the 1D data and Bondarenko factors are varied in Sampler because no covariance data are available for the 2D scattering distributions; however, the 2D data are renormalized to be consistent the perturbed 1D scattering cross sections.

The Medusa module of the XSUSA program [SAMPLER-KHK94] is used to generate perturbation factors for the 1D cross sections on the MG library, assuming that the probability density functions are multivariate normal distributions with covariances given in the SCALE nuclear data covariance library. The library covariance data are given as infinitely-dilute, relative values; therefore a random sample for cross section $\sigma_{x,g}$ corresponds to $\frac{\Delta\sigma_{x,g}}{\sigma_{x,g}}$, where subscript x defines the nuclide/reaction type and g is the group number. The relative variations are transformed to multiplicative perturbation factors, defined by

$$Q_{x,g} = 1 + \frac{\Delta\sigma_{x,g}}{\sigma_{x,g}} \tag{6.4.1}$$

that can be applied to the reference data to obtain the altered infinitely-dilute values. A master sample file containing perturbation factors for 1000 samples (see note below) of the infinitely-dilute 1D data has been pre-computed and stored in the SCALE data directory. Each sample in the file contains perturbation factors for all groups and reactions in all materials. The master sample file is used for all cases, which avoids having to perform the data sampling during SCALE execution.

Because the 1D data in the MG library are infinitely-dilute (i.e., problem-independent), SCALE sequences include modules that compute resonance shielding corrections for the MG data. The self-shielding calculations generally require two additional types of input data: (a) Bondarenko self-shielding factors for the BONAMI module, which typically performs self-shielding calculations outside of the resolved resonance range; and (b) CE cross sections for the CENTRM/PMC modules, which compute pointwise (PW) flux spectra and process self-shielded cross sections for the resolved resonance range. Perturbations in the Bondarenko factors and CE cross sections used in self-shielding calculations must be consistent with perturbations made to the infinitely dilute 1D cross sections since all these data are based on the same fundamental ENDF/B information. It was shown in reference [SAMPLER-WIJ+13] that consistent perturbations can be obtained by using the same perturbation factors $Q_{x,g}$ in following expressions:

(a) infinitely-dilute MG cross sections $\sigma_{x,g}$:

$$\sigma'_{x,g} = Q_{x,g}\sigma_{x,g} \tag{6.4.2}$$

(b) Bondarenko factors f (σ_0, T), at background cross section σ_0 and temperature T:

$$f'_{x,g}(\sigma_0,\ T) = f_{x,g}\left(\sigma'_0, T\right); \quad \text{where } \sigma'_0 = \sigma_0/Q_{x,g} \tag{6.4.3}$$

(c) CE data σ_x(E):

$$\sigma'_x(E) = Q_{x,g}\sigma_x(E),\ \text{for } E \in g \tag{6.4.4}$$

In the above expressions, subscript x defines the nuclide/reaction type and g is the group number.

During Sampler execution the module ClarolPlus reads perturbation factors ($Q_{x,g}$) for a specified sample number from the master sample file, and evaluates Equations Eq. (6.4.2) and Eq. (6.4.3). ClarolPlus also writes a file containing perturbation factors only for the particular sample number used by the CrawdadPlus module, as described below.

Note: The tradeoff of size on disk of the pre-calculated samples distributed with SCALE versus the maximum number of perturbations required in practice has led to the current maximum of 1000 samples. Based on limited experience, correlation coefficients of near zero require the most samples to converge and typically about 1000 samples has been sufficient.

6.4.2.3 Depletion data perturbations

Multiplicative perturbation factors for fission product yields have been generated with XSUSA by sampling the covariances for the independent yield uncertainties. The yield uncertainties are taken from ENDF/B VII.1, which in general are given by fissionable nuclide and for up to three energies: 0.025 eV, 0.5 MeV, and 14 MeV. The ENDF/B yield uncertainties do not include correlations between fission products, which may arise due to constraints such as (a) the sum of all yields must always be two (i.e., the uncertainty in the yield sum is zero), and (b) the uncertainties in independent yields should be consistent with uncertainties given for cumulative yields. The constraints generally introduce positive and negative correlations into the yields

covariance matrix. A method developed by Pigni [SAMPLER-PFG15] was used to determine the correlations in ^{235}U yields. Correlations in yields from other fissionable nuclides are not available in SCALE at this time.

During Sampler execution the perturbation factors are read for a given data sample, compute a complete set of perturbed independent yields for all fissionable nuclides and energies, and renormalize the yields to ensure that they sum to two. An output file containing the perturbed yield data is written to an external file in the format read by ORIGEN. The perturbation factors are read once each time a sequence executed (i.e., for each data sample).

A set of 1,000 decay data perturbations has also been generated with XSUSA and stored in decay-only ORIGEN library files. Sampler automatically aliases the appropriate sample to the file "end7dec".

Note: In order for decay data perturbations to be performed, the "end7dec" decay library must be used **directly**. Typical TRITON and Polaris calculations **do not use "**end7dec" directly, due to using the unperturbed decay data embedded in a special ORIGEN reaction library aliased to "transition.def" as the basis for all coupled transport/depletion calculations. Experience has been that decay data contributes very little additional uncertainty compared to yield data and cross section data.

6.4.2.4 Model data perturbations

An approach presented by Areva NP GmbH utilizes statistical sampling on uncertain parameters to assess the uncertainty in individual system as well as correlations between multiple systems [SAMPLER-BHNS10]. In this approach, values for individual parameters in the input model are randomly modified within the reported uncertainty and distribution function and a series of perturbed values are obtained. Where sufficient samples are made, the distribution of the perturbed values is used to determine the uncertainty in the computed quantity due to uncertainties in the input parameters. In cases where the same uncertain parameters influence multiple experiments the simultaneous perturbation of the parameter for multiple cases will provide the correlation in uncertainties between the different configurations.

To obtain the uncertainty and correlation due to all uncertain parameters, all parameters are randomly perturbed for each calculation and the uncertainties and correlations are determined. Mathematically, the uncertainty in an individual output parameter k is determined as shown in Eq. Eq. (6.4.5).

$$\Delta k^{\exp}(i) = \hat{\mu}_i = \sqrt{\frac{1}{n-1}\sum_{a=1}^{n}\left(\left(k_{calc}^{MC}(i)\right)_a - \overline{k_{\text{calc}}^{MC}(i)}\right)^2}, \tag{6.4.5}$$

where $\Delta k^{\exp}(i)$ is the uncertainty (in terms of standard deviation) in system i due to uncertainties in the input parameters. $\left(k_{calc}^{MC}(i)\right)_a$ is the a^{th} Monte Carlo (MC) sample of system i, where all uncertain input parameters have been randomly varied within the specified distribution.

The covariance between two systems, i and j, is determined as shown in Eq. Eq. (6.4.6).

$$\hat{\Sigma}_{ij} = \sqrt{\frac{1}{n-1}\sum_{a=1}^{n}\left(\left(k_{\text{calc}}^{MC}(i)\right)_a - \overline{k_{\text{calc}}^{MC}(i)}\right)\left(\left(k_{\text{calc}}^{MC}(j)\right)_a - \overline{k_{\text{calc}}^{MC}(j)}\right)}. \tag{6.4.6}$$

The correlation coefficient between systems i and j can be determined from Eqs. Eq. (6.4.5) and Eq. (6.4.6) as shown in Eq. Eq. (6.4.7).

$$c_{ij} = \frac{\hat{\Sigma}_{ij}}{\hat{\mu}_i\hat{\mu}_j}. \tag{6.4.7}$$

The correlation coefficients determined with Eq. Eq. (6.4.7) are the values needed to perform the Generalized Least Linear Square (GLLS) analysis using TSURFER, which solves for a set of cross section data perturbations that would improve agreement between the computational simulations and experimental benchmark results. The correlation coefficients can be shown to be analogous to sensitivity/sensitivity coefficients (i.e., c:sub:k) calculated by SAMS [SAMPLER-HMAK20].

6.4.2.5 Importance Ranking R^2 Methodology

Sampler allows the analysis of the impact of cross section uncertainties on any output quantity of a reactor physics calculation. However, identification of top contributing nuclide reactions to this uncertainty, requires a more detailed analysis. Such rankings can be used for recommendations for additional measurements and evaluations of nuclear data can be made.

A ranking of important nuclide reactions to the uncertainty of response y can be obtained by the determination of the squared multiple correlation coefficient R^2 [SAMPLER-BWAW22]. All the cross sections of all nuclides that are relevant for the model of interest are divided into group A, the nuclide reaction of interest, and group B, all other nuclide reactions. Then R^2 is calculated as follows:

$$R^2_{y,A} = \bar{\rho}_{\bar{\sigma}_A,y} \cdot \bar{\bar{\rho}}_{\bar{\sigma}_A,\bar{\sigma}_A} \cdot \bar{\rho}^t_{\bar{\sigma}_A,y} \tag{6.4.8}$$

$\bar{\rho}_{\bar{\sigma}_A,y}$: vector of correlation coefficients between sampled input parameters of group A and response y

$\bar{\bar{\rho}}_{\bar{\sigma}_A,\bar{\sigma}_A}$: sample correlation matrix between input parameters of group A

R^2 values are between 0 and 1. It is important to note that $\bar{\bar{\rho}}$ is a matrix consisting of correlation coefficients between the input parameters. It is not to be confused with a correlation matrix derived from the original covariance matrix used for sampling the parameters. However, if the sample size approaches infinity, then the sample correlation matrix converges toward the correlation matrix that can be derived from the original covariance matrix and that is used for sampling the cross sections.

R^2 can be interpreted as the expected amount by which the total output variance would be reduced in case the true values of the input parameter group A would become known. The sample size for the determination of R^2 must be larger than the number of independently sampled input cross sections in group A.

Since R^2 is determined based on a statistical approach, they are provided with a statistical error estimate. Both values are accompanied with a 95% confidence interval. The R^2 values are additionally provided with a 95% significance level; only values above this level are considered statistically significant.

6.4.3 INPUT DESCRIPTION

This section describes the Sampler input file format.

6.4.3.1 Overall input structure

The order of the blocks is arbitrary, with the exception of dependent variables (see Sect. 6.4.3.6.3. Below is the layout of a typical sampler input.

```
=sampler

read parameters
  (control flags)
end parameters
read parametric
  (parametric studies definitions)
end parametric
```

(continues on next page)

(continued from previous page)

```
read case[casename]
  sequence=(sequence name)
    (sequence input)
  end sequence
  ... (more sequences) ...
  read variable[id1]
    (variable definition)
  end variable
  ... (more variables) ...

  read response[name1]
    (response definition)
  end response
  ... (more responses) ...
  end case
  ... (more cases) ...

read analysis[name1]
  (analysis definition)
end analysis
... (more analysis) ...

read save
  (file save definition)
end save
... (more saves) ...

end
```

Every Sampler input file has to contain the parameters block and at least one case block. Other input blocks are optional.

Cases and sequences

Within Sampler, multiple independent SCALE calculations, or *cases*, can be included. Since the same set of responses is extracted from each of the cases, these should have the same structure (i.e. produce the same kind of output files); the benefit of having multiple cases within one Sampler input deck is that it is possible to generate cross-correlations between cases as an output.

Every *case* contains one or more stacked *sequences*. The whole case is always run together.

Each case has an unique *identifier*. The identifier is a single word beginning with a letter followed by letters, numbers and underscores. Note that the dash "-" cannot be used in a case identifier.

Within the case block, the user can enter any number of sequences, which contain the actual user input. The format of each sequence block is:

```
sequence=(sequence name) (optional parm= setting)
 (sequence data)
end sequence
```

The sequence data is a SCALE input that is not processed by Sampler, except to substitute sampled values for variable placeholders (see Sect. 6.4.3.6.4 for more information). For `parm=` settings, no limit on column number is enforced.

Importing input data from external files

Instead of directly specifying the SCALE sequence input within the Sampler input file, the user can specify the path to a previously generated input file which can be imported for use within Sampler as:

```
read case[c1]
  import = "/home/usr/sampler_samples/sampler_1x_case.inp"
end case
```

In this case, absolute paths should be used (or a `shell` sequence before invoking Sampler, to copy the appropriate files into the temporary directory). This approach provides concise input files and is advantageous for quality assurance controlled input data.

6.4.3.2 Configuration parameters

In the parameters block the user can control the main workflow and output parameters for sampler. Valid keywords are shown in Table 6.4.1.

Important: Note that of the major perturbation modes, only "perturb_geometry=yes" is on by default. (Bondarenko factor and pointwise data perturbation for CENTRM controls how "perturb_xs=yes" is performed.)

Table 6.4.1: Parameter input for Sampler.

KEYWORD	DESCRIPTION	DEFAULT
`n_samples=N`	Number of samples (1–1000 for nuclear data perturbations, unlimited for input file perturbations)	none
`first_sample=N`	Number of the first sample	1
`perturb_geometry=(yes/no)`	Perform input file/model data/geometry perturbations	yes
`perturb_xs=(yes/no)`	Perform cross-section (XS) perturbation	no
`perturb_yields=(yes/no)`	Perform fission yield perturbation	no
`perturb_decay=(yes/no)`	Perform decay data perturbation	*no*
`perturb_bondarenko=(yes/no)`	When perturbing XS, perturb Bondarenko factors	yes
`perturb_pointwise=(yes/no)`	When perturbing XS, perturb pointwise data	yes
`perturb_kinetics=(yes/no)`	When perturbing XS, perturb decay constants of delayed neutron groups	no
`library=".."`	Name of the master XS library (in quotes), it is possible to use the filename or an alias (e.g. " v7.1-252n")	none
`perturbed_library=".."`	Name of the perturbed library (in quotes); this is the perturbed XS library used by the actual computational sequences	(same as `library`)
`mg_factors_library=".."`	Name of the multigroup XS perturbation factors library (in quotes); if not given, built-in library is used	(blank)

continues on next page

Table 6.4.1 – continued from previous page

KEYWORD	DESCRIPTION	DEFAULT
coverx_library=".."	Name of the covariance library (in quotes); this is the covariance library, that was used to generate the samples, and required for R^2 analysis	56groupcov7.1
run_cases=(yes/no)	Actually run inputs through SCALERTE or just generates them.	yes
force_run=(yes/no)	Enforce running SCALE even when the output files are present	no
plt=(yes/no)	Produce plot file histograms (PTP format) with response distributions that can be viewed with Fulcrum	yes
csv=(yes/no)	Produce CSV files with individual tables	yes
print_data=(yes/no)	Print per-sample values in the main output	no
print_chi2=(yes/no)	Print chi-square normality test in the main output	no

Notes on sample numbers:

The samples are selected from the perturbation factor libraries (except for geometry perturbation); it is up to the user to fit inside the range of samples available (i.e. n_samples+first_sample-1 must be less or equal to the number of samples). The built-in perturbation libraries based nuclear data covariances contain 1000 samples.

Note on perturbed library name:

The default behavior for Sampler is to set `perturbed_library` to the same name as `library`. Since Sampler creates a local file in the temporary directory, which is used by SCALE instead of the library in the lookup table, it in general results in the desired behavior. SCALE sequences only provide pre-defined resonance self-shielding options for known libraries, so where `perturbed_library` differs from the name of a standard SCALE library, the type of resonance self-shielding calculation desired must be specified (via the *PARM=* setting). Please review the documentation of the specified sequence for available options, such as *PARM=CENTRM*.

Warning: The library name must result in a valid filename. In some cases the use of "`xn252`" instead of "`v7.1-252`" is recommended because the dash might result in improper links to the perturbed library. This guidance applies only to the cases when `perturb_xs=yes` is used; otherwise, Sampler does not generate a perturbed library.

6.4.3.3 Sampler responses

For every case run within Sampler any number of responses can be extracted. A response can be a single number or a time-dependent series, which is assigned a name and optionally several parameters. The responses can be entered entered once and shared across all selected cases, i.e. every selected case returns the same set of responses or can be entered inside a case block and only returned for that case.

Sampler recognizes these kinds of responses:

- `opus_plt` – data from an OPUS-generated PLT file

- `triton` - TRITON homogenized cross-sections (xfile016)
- `stdcmp` – standard composition files
- `f71` – concentrations from the F71 ORIGEN dump
- `grep` – general expression from the text output file
- `variables` – the geometry perturbation sampled values

The general format of the response block is:

```
read response[(response id)]
  type = (response type)
  (response parameters)
  cases = (case list) end
end response
```

The *(response id)* is an arbitrary identifier (a single word) by which the the response is denoted in the results. The *(response type)* is one of the keywords `opus_plt`, `triton`, `stdcmp`, `f71`, variables and `grep`. Response parameters are different for each response type and are explained in Sect. 6.4.3.3.1 through Sect. 6.4.3.3.6 . The `cases=` specification applies only to the response defined at the global scope.

The response block can be placed either:

a) inside a case block. This response is collected only to this particular case.

In the following example, response `x` is only collected for the case `c1` and the response `y` is collected for the case `c2`.

```
read case[c1]
  sequence=...
    ...
  end sequence
  read response[x]
    ...
  end response
end case
read case[c2]
  sequence=...
    ...
  end sequence
  read response[y]
    ...
  end response
end case
```

b) at the global scope. The `cases=... end` has to be used to specify the cases for which this response is collected.

In this example, response `x` is collected for the both cases `c1` and `c2` and the response `y` is only collected the case `c1`. Since no `cases=` entry is provided, the response `z` is collected for all cases.

```
read case[c1]
  sequence=...
    ...
  end sequence
end case
read case[c2]
  sequence=...
    ...
```

(continues on next page)

(continued from previous page)

```
  end sequence
end case
read response[x]
  ...
  cases = c1 c2 end
end response
read response[y]
  ...
  cases = c1 end
end response
read response[z]
  ...
end response
```

OPUS PLT file responses

This response extracts any data from a PLT file generated by OPUS. The user specifies which PLT file should be used and which elements/nuclides should be used.

Parameter `ndataset` provides the number of the selected PLT file, i.e. `ndataset=1` will read data from the file ending with .00000000000000001.plt (which is the second generated PLT file in the given case).

Parameter `nuclides=...end` specifies the list of nuclides (or elements) which are read from the PLT file; nuclides can be specified as alphanumeric identifiers (U-235, ba137m) or six-digit ZAI identifiers (922350). In addition to that, any other PLT file response identifiers (i.e. the character strings in the first column of the plot table) may be used, which allows for example the usage of `total` and `subtotal` keywords.

Example

```
read response[fisrates]
  type = opus_plt
  ndataset = 1
  nuclides = u238 pu239 total end
end response
```

TRITON homogenized cross-section responses

This response extracts the homogenized cross-section data saved by TRITON on the `xfile016`.

Responses are retrieved for a selected homogenized mixture and all branches (which are then denoted by response name suffixes).

Using a `data= ... end` assignment specifies which data types are to be saved.

The available options for data entries are:

```
kinf sigma_total sigma_fission sigma_absorption
sigma_capture sigma_transport_out sigma_transport_in
sigma_transport sigma_elastic sigma_n2n nu_fission
kappa_fission nu chi flux diffusion
```

Example:

```
read response[xs]
  type = triton
  mixture = 1
  data = kinf sigma_absorption end
end response
```

Standard composition file responses

This response retrieves isotopic concentrations (in atoms/barn-cm) from the standard composition file. Parameter `nuclides=...end` specifies which nuclides should be retrieved. The parameter mixture specifies the number of the StdCmpMix file, so `mixture=10` would load concentrations from the file `StdCmpMix00010_*` (for all time steps).

Example:

```
read response[mix10]
  nuclides = u-235 pu-239 end
  mixture = 10
end response
```

ORIGEN concentration (F71) responses

This response retrieves the isotopic concentrations (in gram-atoms) from the ORIGEN concentration edit in the `ft71f001` file.

Parameter `nuclides=...end` specifies the list of nuclides (or elements) which are read from the F71 file; nuclides can be specified as alphanumeric identifiers (U-235, ba137m) or six-digit ZAI identifiers (922350).

Two options are available to choose the positions on the file from which data should be retrieved. Either `step_from=start` and `step_to=end` can be used to select a range of positions, or `mixture=N` can be used to choose either an ORIGEN case or a TRITON case mixture. This is convenient for TRITON cases where step numbers are usually not known in advance.

Example:

```
read response[concentrations]
  type=origen_nuclides
  nuclides = u-235 pu-239 pu-240 pu-241 end
  mixture = 10
end response
```

Generic regular expression (GREP) responses

In order to allow the user to collect other responses from the SCALE output, a generic regular expression (regexp) mechanism is provided by Sampler. For every response the user can enter one (or more) regular expressions, which are applied (using the "grep" system tool) to the main output file. "grep" is executed with the "-o" option, which returns only the matched portion of the line (and not the whole line). Usually it is necessary to use two expressions, one to find the line of interest and another to extract only the desired value. The POSIX character classes are supported in the grep used-the most commonly used are "[[:digit:]]" to match a single digit 0-9 and [[:space:]] to match a single space or tab. The "+" and "*" are used to match one or more and zero or more repeats, respectively. Note that as per standard regexp rules, "." matches any character and an escape is necessary, i.e. ".", in order to match a period.

Each regular expression is defined by the keyword `regexp="..."`. The result of the last regular expressions should be a single number (and is treated as such by Sampler).

The following example defines regular expression for extraction of k-effective from a CSAS5/6 output file:

```
read response[keff]
  type = grep
  regexp = "best estimate system k-eff[[:space:]]+[[:digit:]]+\.[[:digit:]]+"
  regexp = "[[:digit:]]+\.[[:digit:]]+"
end response
```

With the first `regexp` statement, a line containing "best estimate system k-eff" followed by a number is found, and then just the number part is extracted with the second `regexp` statement.

For ease of use, Sampler provides several regular expression shortcuts shown in Table 6.4.2.

Table 6.4.2: Regular expression shortcuts for SCALE sequences.

`regexp` = "`:kenova.keff:`"	keff from KENO5 / CSAS5 sequence
`regexp` = "`:kenova.ealf:`"	EALF from KENO5 / CSAS5 sequence
`regexp` = "`:kenova.nubar:`"	nu-bar from KENO5 / CSAS5 sequence
`regexp` = "`:kenovi.keff:`"	keff from KENO6 / CSAS6 sequence
`regexp` = "`:xsdrn.lambda:`"	lambda from XSDRN / CSAS1 sequence
`regexp` = "`:scale.number:`"	matches any number (e.g. "1", "1.0", "1.23e-7", "-0.3")

Thus, the previous example may be alternately rephrased as such:

```
read response[keff]
  type = grep
  regexp = ":kenova.keff:"
end response
```

In addition to this, the `grep` response also supports extraction of data with uncertainties. In order to get the response uncertainty, use the `eregexp=` keyword, which follows the same rules as `regexp=`. The same shortcuts as for `regexp=` may be used as well (for KENO V.a/VI multiplication coefficient). Therefore, to get KENO multiplication factor including the uncertainty, one might define the response like this:

```
read response[keff]
  type = grep
  regexp = ":kenova.keff:"
  eregexp = ":kenova.keff:"
end response
```

Sampled variable values

Using the `variables` response, the user can extract information from the sampled values of the geometry/material perturbation variables. The `data=` key contains the list of variable identifiers of interest.

This option is useful to generate the correlations between geometry/material perturbations and the responses of interest.

Example:

```
read variable[r1]
  ...
end variable
read response[r]
  type=variables
  data = r1 end
end response
```

Quick response definition overview

Table 6.4.3 summarizes the available options for the different response types. The nuclides specification can either be in terms of the standard alphanumeric identifier, e.g. "u235m" for ^{235m}U, or the IZZZAAA integer identifier, e.g. "1092235" for ^{235m}U.

Table 6.4.3: Response definition overview.

Key	**Description**
response type `opus_plt`	
Ndataset	Number of the PTP file
Nuclides	List of nuclides (alphanumeric/IZZZAAA, terminated by end)
response type `stdcmp`	
`Mixture`	Number of the StdCmpMix file
`Nuclides`	List of nuclides (alphanumeric/IZZZAAA, terminated by end)
response type `triton`	
Mixture	Homogenized mixture index
`Nuclides`	List of nuclides (alphanumeric/IZZZAAA, terminated by end)
`Data`	List of homogenized data types (terminated by end)
response type `f71`	
`Nuclides`	List of nuclides (alphanumeric/IZZZAAA, terminated by end)
`Mixture`	ORIGEN case number / TRITON mixture number
`step_from`	Lower bound of position range
`step_to`	Upper bound
response type `grep`	
`regexp`	Regular expression for the response value (quoted)
`eregexp`	Regular expression for the response uncertainty (quoted)
response type `variables`	
`data`	List of variable names (terminated by end)

6.4.3.4 Saving files

By default, Sampler saves from each run the input, output, message and terminal log files. In addition to that, if respective responses are requested, it saves the `ft71f001` as basename`.f71`, x``file016`` as basename`.x16` and the `StdCmpMix*` files.

The user might specify additional files to be saved into the sample subdirectory; this is achieved by defining one or more `save` blocks.

Each save block contains a `file="..."` parameter, which specifies the filename in the sample run temporary directory. Optionally, the user can specify `name="..."` to let Sampler rename the file to *basename.extension*, where *extension* is the value of the `name` parameter. If `name` is not specified, the file is not renamed and is just copied to the sample subdirectory. The `name` parameter cannot be used if wildcards are used in the `file` parameter.

The quotes for both `name` and `file` parameter values are mandatory.

Example 1:

```
read save
  name = "ft2"
  file = "ft71f002"
end save
```

Example 2:

```
read save
  file = "StdCmpMix*"
end save
```

6.4.3.5 Parametric studies

The Sampler infrastructure allows an efficient implementation of studies of parameter variation effects on various responses. This mode is activated by entering the `read parametric ... end parametric` block.

This block contains two arrays: `variables = ... end` and `n_samples = ... end`. The variables array lists the variables of the parametric study. The variables must have "`distribution=uniform`", and the minimum and maximum becomes the range for that variable in the parametric study. For each variable, the corresponding value in the `n_samples = ... end` array indicates the number of evenly spaced values to assume *in that dimension.* The total number of calculations is therefore the multiplication of all the `n_samples` values. Note that for a single sample with `n_samples=1`, only the minimum value is used. Below is as an example of the parametric block.

```
read parametric
  variables = density temperature end
  n_samples =      10           6 end
end parametric
```

The two variables are `density ``and ``temperature`, and there will be 10 evenly spaced density values (including the minimum and maximum) and 6 evenly spaced temperature values, for a total of $10 \times 6 = 60$ calculations. To perform the same number of samples in each dimension, the keyword `n_samples` in the parameters block may also be used.

Sampler generates a summary table of the parametric study, including values for which the minimum and maximum of each response occurs. Sampler also generates PTP plot files showing the dependency of each response on each variable.

6.4.3.6 Geometry and material perturbations

In addition to data perturbations, Sampler also allows the user to include geometry and material uncertainties in the calculation. This is achieved by defining `variable` blocks. Each variable may be linked to a particular value in the input and is associated either with a random variable distribution or with an arithmetic expression. The expression capability allows for dependent or derived parameters, such as ^{238}U content depending on enrichment or outer clad radius depending on the inner radius.

For each sample, Sampler creates a perturbed input by generating a set of variable values and substituting them into the input. For every variable, the user has to define the variable and specify its distribution (using one of the predefined random variable distributions described in Sect. 6.4.3.1.1) or its dependence on other variables (using an arithmetic expression). If desired, the user can also specify which part of the input will be replaced by the variable. This can be achieved either by specifying a SCALE Input Retrieval ENgine (SIREN) expression or by putting placeholders directly inside the input deck (see Appendix A for details on SIREN).

Variable definition

Variables are defined by a `read variable..end variable` block. The general format of the block is:

```
read variable[(variable id)]
  distribution = (distribution type)
  siren = "(siren expression)"
  (distribution-specific parameters)
  cases = (case list) end
end variable
```

The variable id is an arbitrary, single word consisting of letters, numbers and underscores (with number not being the first character). The variable id has to be unique and is case dependent. `distribution` is one of the distribution-type keywords (see Sect. 6.4.3.6.2) or `expression` for the dependent variable definition. The `cases=` specification applies only to the variables defined at the global scope (see below). The `siren=` specification is optional, see below.

The block can be placed either:

a) inside a case block. This variable applies only to this particular case.

In the following example, variable `x` applies to the case `c1` and the variable `y` applies to the case `c2`.

```
read case[c1]
  sequence=...
    ...
  end sequence
  read variable[x]
    ...
  end variable
end case
read case[c2]
  sequence=...
    ...
  end sequence
  read variable[y]
    ...
  end variable
end case
```

b) at the global scope. The `cases=... end` has to be used to specify the cases to which this variable applies.

In this example, variable `x` applies to the both cases `c1` and `c2` and the variable `y` applies only to the case `c1`.

```
read case[c1]
  sequence=...
    ...
  end sequence
end case
read case[c2]
  sequence=...
    ...
  end sequence
end case
read variable[x]
  ...
  cases = c1 c2 end
end variable
read variable[y]
  ...
  cases = c1 end
end variable
```

Distribution types

Sampler supports three random `distribution` types, selected using the distribution= keyword.

1) `uniform`: uniform distribution over a (closed) interval.

Additional parameters for a uniform distribution are shown in Table 6.4.4.

Table 6.4.4: Parameters for uniform distributions.

KEYWORD	DESCRIPTION	NOTE
minimum=	Lower bound value	required
maximum=	Upper bound value	required

Example:

```
read variable[rfuel]
  distribution=uniform
  minimum = 0.40
  value = 0.41
  maximum = 0.42
end variable
```

2) `normal`: normal (optionally truncated) distribution.

Additional parameters for normal distribution are shown in Table 6.4.5.

Table 6.4.5: Parameters for normal distributions.

KEYWORD	DESCRIPTION	NOTE
value=	Mean value	required
stddev=	Standard deviation	required
minimum=	Lower cutoff value	optional
maximum=	Upper cutoff value	optional

The user can specify both `minimum` and `maximum`, one of them, or neither. If a cutoff is not specified, the distribution is not truncated on that side.

Example:

```
read variable[c1_u235]
  distribution=normal
  value=95.0
  stddev=0.05
end variable
```

3) `beta`: beta distribution.

Additional parameters for the beta distribution are shown in Table 6.4.6.

Table 6.4.6: Parameters for beta distributions.

KEYWORD	DESCRIPTION	NOTE
minimum=	Lower cutoff value	required
maximum=	Upper cutoff value	required
beta_a=	First parameter for distribution	required
beta_b=	Second parameter for distribution	required

The Beta distribution is defined in the standard way given by Eq. Eq. (6.4.9). The parameters for the distribution determine where the peak is located in the interval [`minimum`, `maximum`] and the variance of the distribution; the parameters α and β are required to be integer values.

$$f(x;\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}, \tag{6.4.9}$$

Example:

```
read variable[rclad]
  distribution=beta
  value=0.47
  minimum=0.45
  maximum=0.50
  beta_a=2
  beta_b=6
end variable
```

Dependent variables (expressions)

Using `distribution=expression` allows the user to specify a variable using the values of other variables. Setting `expression`="(expression)" then specifies how to evaluate the variable. Sampler supports basic arithmetic operators and other variables can be used as well. However, Sampler currently provides no variable dependency resolution and therefore only variables that were defined (using the variable block) previously in the input deck can be referenced in an expression.

Example:

```
read variable[c1_u238]
  distribution = expression
  expression="100.0-c1_u235"
end variable
```

Using placeholders

Inside the `sequence`= blocks, a #{*variable id*} placeholder can be used. This will be replaced by a variable value when Sampler builds the particular input deck. Only a simple variable reference can be used; no expressions are allowed here.

Example:

```
sequence=csas5
...
uranium  1 den=18.76 1 300 92235 #{u235} 92238 #{u238} end
...
end sequence
read variable[u235]
...
end variable
read variable[u238]
...
end variable
```

In the input deck snippet the ^{235}U and ^{238}U content are inserted directly to the respective places (defined by variables u235 and u238).

Using placeholders is straightforward and simple; however, if for input deck quality assurance or other reasons it is not desirable to modify the input deck directly, SIREN expressions can be used.

Using SIREN expressions

SIREN is a package which provides an XPath-like interface to the SCALE input deck represented by a Document Object Model (DOM). In Sampler, the user can specify `siren`="*path*" to have the respective token(s) replaced by a variable value.

Please refer to Appendix A for more details on specifying the SIREN path expressions.

Example:

```
read variable[c1_u235]
  distribution=normal
  value=95.0
  stddev=0.05
  siren="/csas5/comps/stdcomp[decl='uranium']/wtpt_pair[id='92235']/wtpt"
end variable
```

The variable `c1_u235` value is inserted as the weight percent of ^{235}U in the basic standard composition declared as "uranium" in the CSAS5 sequence.

6.4.3.7 Response analysis

Sampler provides several statistical post processing options to analyze results of propagated uncertainties. Variations in a user selected response, mutual variations between two different responses with respect to perturbed data can provide insight to dependecy to input data or any correlation between responses.

The general format of the analysis block is:

```
read analysis[(analysis id)]
  type = (analysis type)
  targets = (list of target responses) end
  sources = (list of source responses or nuclides) end
end analysis
```

The analysis block mainly provides three input fields: target response, source response, and analysis type. Both the target and source responses can be a list of responses. The calculation of the selected analysis type is performed between the source and target response lists.

Sampler reconizes the following analysis types:

- `pearson_corr` - pearson correlation coefficient (similarity index) calculation between each response in the sources response list and the target response list
- `covariances` - covariance calculation between each response in the sources response list and the target response list
- `correlation_matrix` - pearson correlation coefficient calculation between responses in the sources response list (target responses are ignored for this type)
- `covariance_matrix` - covariance calculation between responses in the sources response list (target responses are ignored for this type)
- `r2` - sensitivity indices R^2 are calculated for responses in the target response list for the isotopes listed in the sources response list

Responses in target and sources lists are specified using the CASE:RESPONSE.SUB(INSTANCE) notation

where

- CASE : User-selected Sampler case name in the case block from which the response is collected.
- RESPONSE : Response name in the response block
- SUB : Sub-response listed in the specified response block, such as isotope names in origen or opus response types. If the response type has no sub-response (e.g., in the case of a grep response), then it is not used.
- INSTANCE : Index of occurrence in the collected response, such as the depletion step index for a burnup dependent response, or the line number in which the selected response occurs in the case of grep type responses. Note that the instance counting starts with 0 in case of grep responses. A missing index field is interpreted as if the response is requested for index=0. In addtion to integer indexes the following special keywords are also accepted
  - last : the last occurence
  - all : all occurences. This keyword populates the requested sub response for all occurence indices. For example, case:response.sub(all) is the same as case:response.sub(0) case:response.sub(1) ... case:response.sub(last)

The following input example calculates the similarity indices between ^{235}U concentration at the 15th time step in the MOX depletion model with ^{235}U, ^{239}Pu , ^{243}Am, and ^{109}Ag concentrations at the 14th depletion step from the UO2 model:

Pearson Correlation Example

```
read case[c1]
 import = "/home/usr/samples/moxdepletion.inp"
end case

read case[c2]
 import = "/home/usr/samples/uo2depletion.inp"
 read response [experiments]
   type = origen_nuclides
   nuclides = u-235 pu-239 am-243 ag-109 end
   mixture = 5
 end response
end case

read response [application]
  type = origen_nuclides
  nuclides = u-235 end
  mixture = 10
  cases = c1 end
end response

read analysis [iso_ck]
  type = pearson_corr
  targets = c1:application.u-235(15)   end
  sources = c2:experiments.u-235(last) c2:experiments.pu-239(14)  end
end analysis
```

Correlation Matrix Example

Correlation and covariance matrices only require the sources card since they are symmetric

```
read analysis [corr-covmatrix]
  type = correlation_matrix
  sources = c2:experiments.u-235(all)  end
end analysis
```

Squared Correlation Example

For the analysis of R^2, isotopes need to be entered as sources:

```
read analysis [sensindex]
  type = r2
  targets = c1:application.u-235(15) end
  sources = h-1 u-235 u-238  end
end analysis
```

6.4.3.8 Converting a standard SCALE input to a Sampler input

In this section, a short walkthrough is provided on how to convert a "normal" SCALE input into a Sampler input for cross section uncertainty propagation.

Beginning with a simple CSAS5 input deck:

```
=csas5
  sample problem 1  case 2c8 bare
  v7.1-252
  read comp
    uranium  1 den=18.76 1 300 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
  end comp

  read geometry
    unit 1
      cylinder 1 1 5.748 5.3825 -5.3825
      cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  end geometry
    read array
      gbl=1 ara=1 nux=2 nuy=2 nuz=2 fill f1  end fill
    end array
  end data
end
```

First, wrap the given input in sequence and case blocks and assign the case an arbitrary identifier (`c1`). It is also recommended to change the library alias to the actual filename.

```
=sampler
read case[c1]
  sequence=csas5
    sample problem 1  case 2c8 bare
    xn252
    read comp
      uranium  1 den=18.76 1 300 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
    end comp

    read geometry
      unit 1
        cylinder 1 1 5.748 5.3825 -5.3825
        cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
    end geometry
      read array
```

(continues on next page)

(continued from previous page)

```
        gbl=1 ara=1 nux=2 nuy=2 nuz=2 fill f1  end fill
      end array
    end data
  end sequence
end case
end
```

Second, add the parameters block to identify the base cross-section library used for building the perturbed ones. Note that this is the only SCALE module that uses the plural form of parameters and that the reference to the library matches exactly the one inside the CSAS5 input.

```
=sampler
read parameters
  library="xn252"
end parameters
read case[c1]
  sequence=csas5
    sample problem 1  case 2c8 bare
    xn252
    read comp
      uranium  1 den=18.76 1 300 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
    end comp
    read geometry
      unit 1
        cylinder 1 1 5.748 5.3825 -5.3825
        cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
    end geometry
      read array
        gbl=1 ara=1 nux=2 nuy=2 nuz=2 fill f1  end fill
      end array
    end data
  end sequence
end case
end
```

Finally, set up the perturbations (number of samples and what to perturb):

```
=sampler
read parameters
  library="xn252"
  n_samples = 40
  perturb_xs = yes
end parameters
read case[c1]
  sequence=csas5
    sample problem 1  case 2c8 bare
    xn252
    read comp
      uranium  1 den=18.76 1 300 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
    end comp
    read geometry
      unit 1
        cylinder 1 1 5.748 5.3825 -5.3825
        cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
    end geometry
      read array
        gbl=1 ara=1 nux=2 nuy=2 nuz=2 fill f1  end fill
      end array
    end data
  end sequence
end case
end
```

6.4.4 EXECUTION DETAILS

6.4.4.1 General workflow

The overall workflow for Sampler is as follows:

- for each sample, pick the perturbation factors and generate geometry perturbations
  - for each case, build SCALE input decks, which include:
    - calls to the perturbation modules, which generate the perturbed data libraries (based on the perturbation factors for this sample)
    - user sequence inputs
    - output data retrieval
  - insert each of the constructed input decks into the processing queue
- run all SCALE cases from the queue (serial or parallel, as available)
- perform data extraction (using the response mechanism) and statistical analysis
- print output and generate data files

The advantage of this workflow is that the individual SCALE runs are completely identified by the sample number (so they are reproducible) and they are independent. Each of the runs is executed within its own environment (with SCALE runtime as a subprocess), with its own decay data, fission yield and cross section library. This arrangement is very robust and, as there is no coupling between the runs, can be effectively parallelized.

6.4.4.2 File management

For every SCALE run, Sampler creates a subdirectory within its own temporary directory. Each subdirectory has a name in the form *(casename)* `_pert_` *(sample number)*. Within this directory all the useful data for the particular run are stored: the input file, the output file, the message file, the terminal log file (which is a joint capture of SCALE both standard output and standard error stream) along with the saved data files (ft71, xfile016, PTP files etc.)

By retaining the temporary directory, the user can then examine and possibly reuse saved files for the individual SCALE runs.

6.4.4.3 Parallel execution

Since the Sampler calculations usually consist of several hundred mutually completely independent calculations, it is desirable to run the subcases in parallel.

In SCALE 6.3, Sampler only supports threading for parallel calculations.

In order to run Sampler in parallel simply specify the `-I` command line arguments to ScaleRTE. .

6.4.4.4 Behavior when encountering errors

Any time a parameter within a SCALE input is perturbed, there is the possibility that the perturbation will cause unrealistic behavior (fuel pellet passing through cladding, etc.) that will cause SCALE to fail. The default behavior of Sampler is to finish all perturbed cases and check whether there are errors present for each case once all cases have been run.

6.4.5 EXAMPLE PROBLEMS AND OUTPUT DESCRIPTION

This section describes output files created by Sampler and provides several sample cases.

6.4.5.1 Output description

This section describes the contents of the main Sampler output file, as well as the other files generated by Sampler.

All of the CSV, PTP and SDF files are, for convenience, copied into a separate directory called *$ {OUTBASENAME}*.`samplerfiles`, where *$ {OUTBASENAME}* is the base name of the main SCALE output file, e.g. "my" in "my.out".

Main text output

The main text output summarizes the Sampler run progress and presents the most important results.

Sampler banner

The program verification information banner shows the program version and the main execution information (date and time, user name, computer name).

```
*******************************************************************************************************************************************
*******************************************************************************************************************************************
*******************************************************************************************************************************************
*****                                                                                                                                     *****
*****                                          program verification information                                                           *****
*****                                                                                                                                     *****
*****                                         code system:  SCALE   version: 6.2                                                          *****
*****                                                                                                                                     *****
*******************************************************************************************************************************************
*******************************************************************************************************************************************
*****                                                                                                                                     *****
*****                                                                                                                                     *****
*****                       program:  sampler                                                                                             *****
*****                                                                                                                                     *****
*****                       version:  6.2.0                                                                                               *****
*****                                                                                                                                     *****
*****                      username:  usr                                                                                                 *****
*****                                                                                                                                     *****
*****                      hostname:  node11.ornl.gov                                                                                     *****
*****                                                                                                                                     *****
*****                                                                                                                                     *****
*****             date of execution:  2013-04-04                                                                                          *****
*****                                                                                                                                     *****
*****             time of execution:  14:40:01                                                                                            *****
*****                                                                                                                                     *****
*******************************************************************************************************************************************
*******************************************************************************************************************************************
*******************************************************************************************************************************************
```

Input parameters echo

Input echo table summarizes the user selected parameters and options.

```
------------------------------------------------------------------------------------------------------------------------
↪--------------------
-                                                    Input parameters echo                                             ␣
↪                    -
------------------------------------------------------------------------------------------------------------------------
↪--------------------

   Number of cases                          : 1
   Number of samples                        : 500
   First sample index                       : 1
   Number of MAT-MT pairs                   : 0

   Perturb cross-sections                   : yes
   Perturb decay data                       : no
   Perturb fission yields                   : no

   Perturb pointwise XS                     : yes
   Perturb Bondarenko factors               : yes

   Master XS library                        : xn252
   Perturbed XS library                     : xn252
   Multigroup factors library               :
   Sensitivity factors library              : sensitivity_factors
   Covariance library                       : 44groupcov

   Print CSV tables                         : yes
   Print PTP histograms/histories           : yes
   Print per-sample data                    : no
   Print covariances                        : no
   Print correlations                       : no
   Print chi-square test                    : no
```

SCALE run overview

The run overview table displays the list of SCALE calculations processed by Sampler, i.e. for each case the baseline calculation (sample 0) and the requested number of samples.

```
*****************************************************************************************************************************************************
*                                                                  Sampling                                                                         *
*****************************************************************************************************************************************************

      case |              run         sample       running?                                                  sample directory
-----------+-------------------------------------------------------------------------------------------------------------------
 case00001 |   sample #1 / 41         #00000            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00000
 case00001 |   sample #2 / 41         #00001            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00001
 case00001 |   sample #3 / 41         #00002            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00002
 case00001 |   sample #4 / 41         #00003            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00003
 case00001 |   sample #5 / 41         #00004            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00004
 case00001 |   sample #6 / 41         #00005            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00005
 case00001 |   sample #7 / 41         #00006            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00006
 case00001 |   sample #8 / 41         #00007            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00007
 case00001 |   sample #9 / 41         #00008            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00008
 case00001 |  sample #10 / 41         #00009            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00009
 case00001 |  sample #11 / 41         #00010            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00010
 case00001 |  sample #12 / 41         #00011            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00011
 case00001 |  sample #13 / 41         #00012            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00012
 case00001 |  sample #14 / 41         #00013            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00013
 case00001 |  sample #15 / 41         #00014            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00014
 case00001 |  sample #16 / 41         #00015            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00015
 case00001 |  sample #17 / 41         #00016            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00016
 case00001 |  sample #18 / 41         #00017            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00017
 case00001 |  sample #19 / 41         #00018            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00018
 case00001 |  sample #20 / 41         #00019            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00019
 case00001 |  sample #21 / 41         #00020            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00020
 case00001 |  sample #22 / 41         #00021            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00021
 case00001 |  sample #23 / 41         #00022            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00022
 case00001 |  sample #24 / 41         #00023            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00023
 case00001 |  sample #25 / 41         #00024            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00024
 case00001 |  sample #26 / 41         #00025            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00025
 case00001 |  sample #27 / 41         #00026            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00026
 case00001 |  sample #28 / 41         #00027            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00027
 case00001 |  sample #29 / 41         #00028            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00028
 case00001 |  sample #30 / 41         #00029            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00029
 case00001 |  sample #31 / 41         #00030            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00030
 case00001 |  sample #32 / 41         #00031            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00031
 case00001 |  sample #33 / 41         #00032            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00032
 case00001 |  sample #34 / 41         #00033            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00033
 case00001 |  sample #35 / 41         #00034            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00034
 case00001 |  sample #36 / 41         #00035            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00035
 case00001 |  sample #37 / 41         #00036            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00036
 case00001 |  sample #38 / 41         #00037            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00037
 case00001 |  sample #39 / 41         #00038            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00038
 case00001 |  sample #40 / 41         #00039            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00039
 case00001 |  sample #41 / 41         #00040            yes     /home/usr/sampler_samples/sampler_3_tmp/case00001_pert_00040
```

(continues on next page)

(continued from previous page)

```
--- Master process needs to run 41 SCALE runs.
```

The table shows case name, sample index, sample number (i.e. the number in the perturbation factor library), whether the case has to be executed (if not it means that the results were already available in the samplerfiles directory) and the full path to the run subdirectory.

Response tables

According to the print flags set by the user in the parameters block, Sampler prints the following tables:

- values of all responses for all samples (printed if `print_data=yes`)
- average values and standard deviation over the samples population (always printed)
- comparison of average and baseline value (always printed)
- chi-square normality test (printed if `print_chi2=yes`)
- covariance matrices (printed if `print_cov=yes`)
- correlation matrices (printed if `print_corr=yes`)
- case- and response- specific tables

The case-specific tables contain only responses for a given case (so it is possible to explore correlations only within a given case). The response-specific tables are, on the other hand, contain responses across all cases, so they are useful for case cross-correlation analysis.

All of the tables are, regardless of the print flags, saved in the CSV files (see the following section).

CSV tables

Every table produced by Sampler is (regardless of whether it has been selected for the main text output) saved also in the CSV (comma separated values) format, which makes it convenient to process Sampler results with a spreadsheet program, plotting package, or any scripting workflow.

These types of tables are created:

- values for every sample for time-independent responses

  (`response_table.static.val.all.csv`)
- values for every sample for time-independent responses for each case

  (`response_table.static.val.case-*.csv`)
- values for every sample for time-independent responses for each response

  (`response_table.static.val.response-*.csv`)
- values for every sample for time-dependent responses

  (`response_table.*.csv`)
- average values for time-dependent responses

  (`response_table.*.avg.csv`)
- standard deviations for time-dependent responses

  (`response_table.*.stddev.csv`)
- analysis results for the analysis named myAnalysis

  (`response_table.analysis.myAnalysis.csv`)

Sampling histograms and running averages

In order to provide information on sampling convergence, Sampler provides two plots for each response at every time step:

- histogram plot – distribution of the response values in directory *${OUTBASENAME}*.samplerfiles/histogram
- running average plot – average and standard deviation for first N samples of the population in directory *${OUTBASENAME}*.samplerfiles/running_averages

Both plots are in the PTP format and can be plotted by Fulcrum, as shown in Fig. 6.4.1 and Fig. 6.4.2.

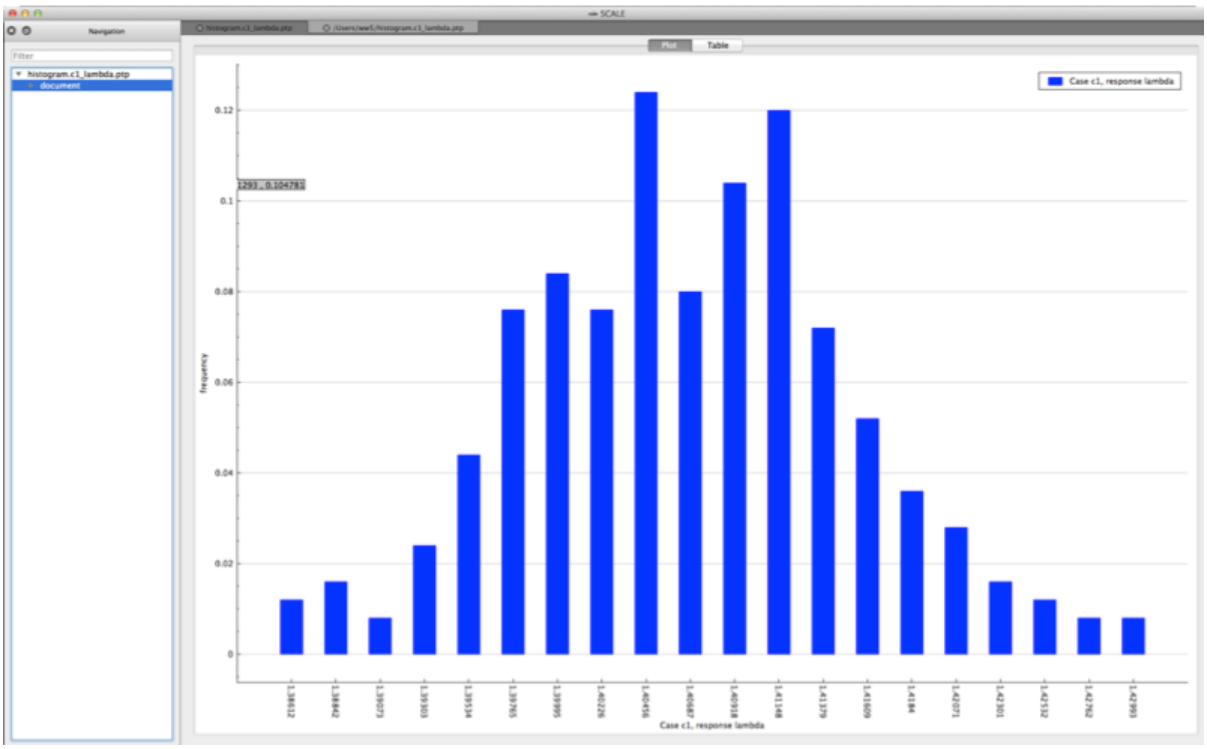


Fig. 6.4.1: Example histogram viewed in Fulcrum.

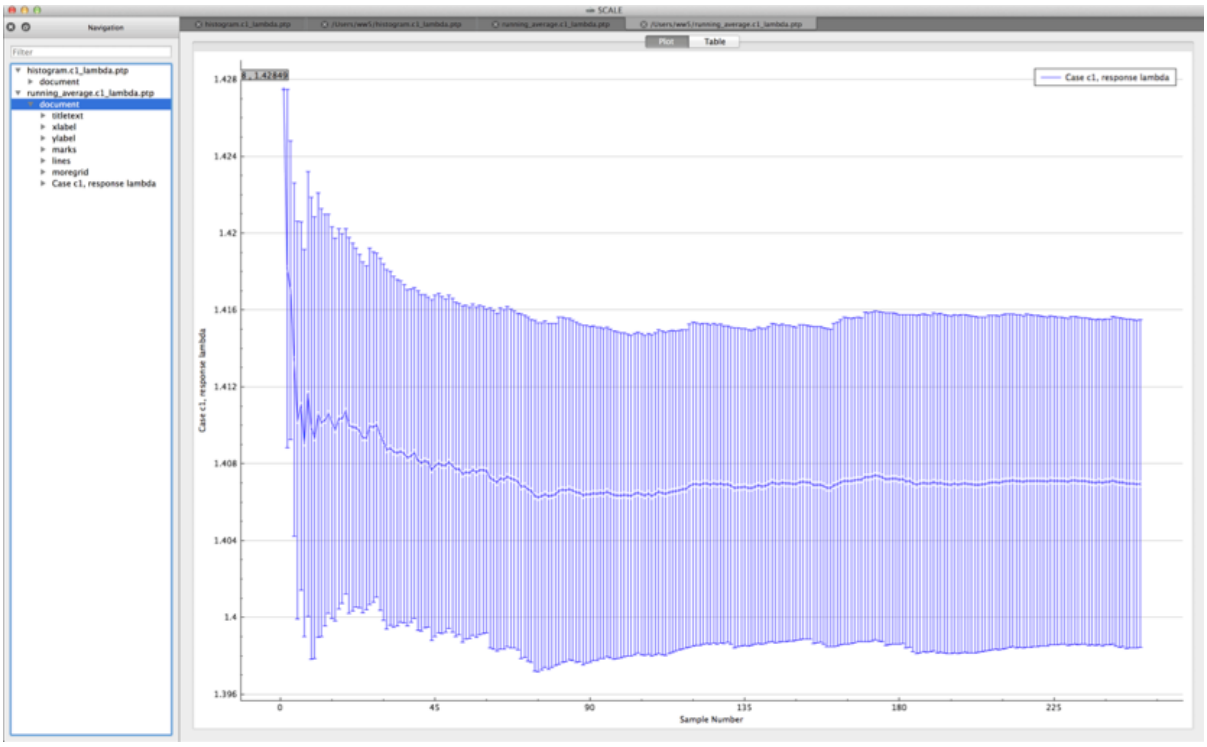


Fig. 6.4.2: Example running average viewed in Fulcrum.

Response histories

For time-dependent responses, Sampler produces two plots with time-dependent summary data.

- standard deviation plot – time-dependent average response with 1-sigma uncertainty bars in directory *$ {OUTBASENAME}*.samplerfiles/histories/history.avg.*
- min/max plot – time-dependent average response with min/max error bars in directory *$ {OUTBASE-NAME}*.samplerfiles/ histories/history.min_max.*

An example of the standard deviation plot is shown Fig. 6.4.3 and the min/max plot in Fig. 6.4.4.

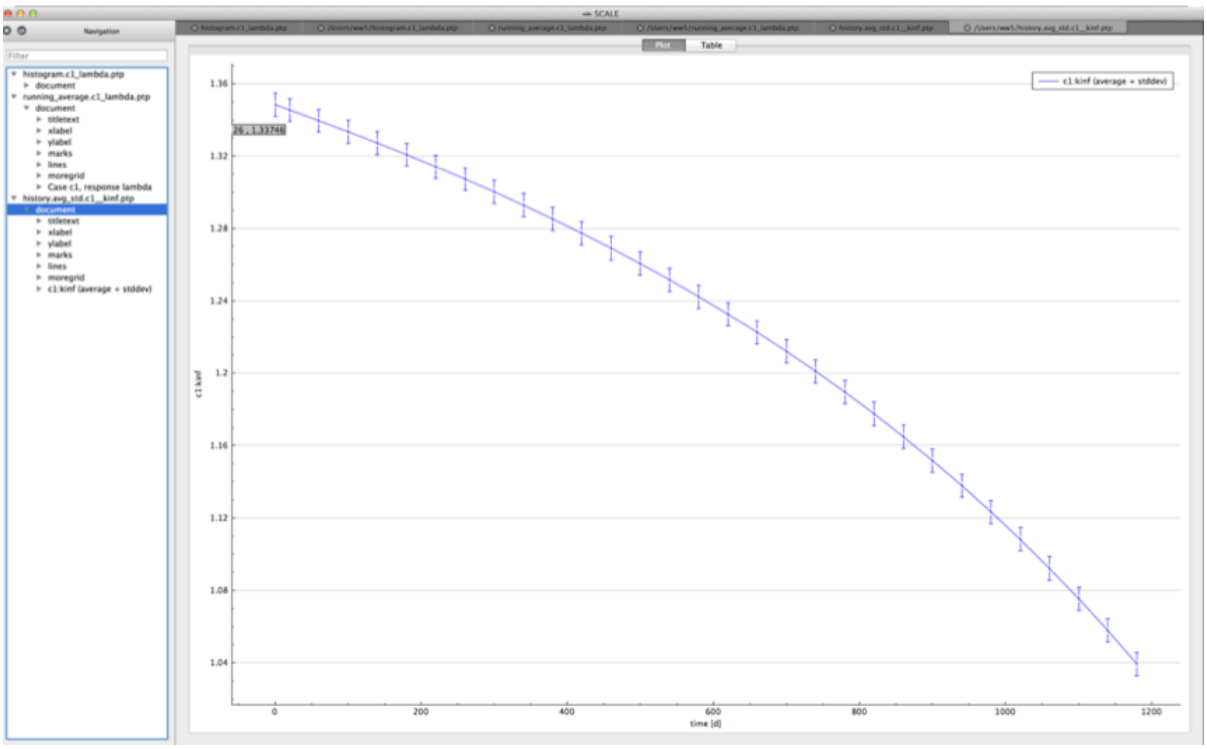


Fig. 6.4.3: Time-dependent average plus standard deviation plot.

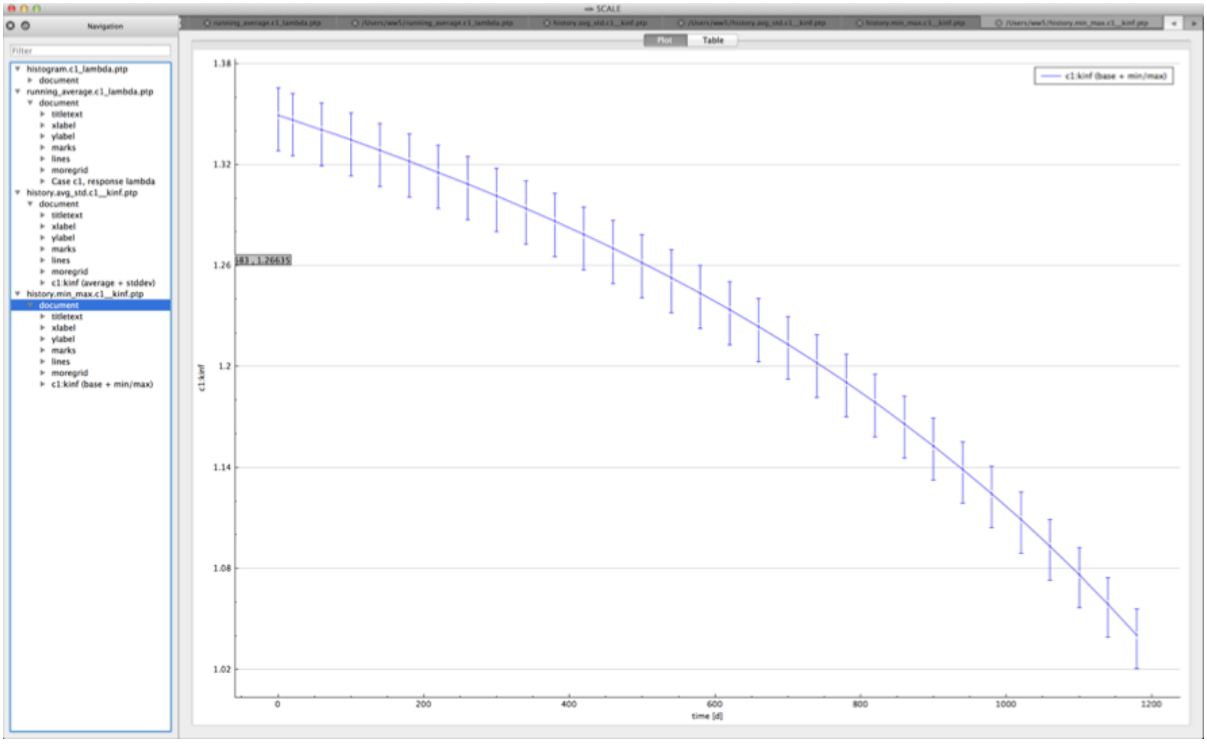


Fig. 6.4.4: Time-dependent average plus min/max plot.

6.4.5.2 Sample problems

The following sample problems demonstrate various computational and output capabilities of Sampler in various situations, for both uncertainty and parametric calculations.

Input files for those sample problems can be found in the `samples/input` directory of the SCALE installation. The naming convention for the inputs is sampler_*N*.inp, where *N* is the sample problem number. The number of samples (n_samples) shown here may vary from the number included in the sample inputs.

Sample problem 1

This simple, single-case problem, evaluates uncertainty in eigenvalue for a T-XSDRN calculation of a MOX pincell. Only the cross-sections are perturbed.

```
=sampler

read parameters
  n_samples=250
  library="xn252"
  perturb_xs = yes
end parameters

read case[c1]
  sequence=t-xsdrn           parm=2region
    pin-cell model with MOX
    xn252
    read comp
      uo2   1 0.95 900 92235 4.5 92238 95.5   end
      zirc2 2 1 600   end
      h2o   3 den=0.75 0.9991 540   end
    end comp
    read cell
      latticecell squarepitch pitch=1.3127 3 fuelr=0.42 1 cladd=0.9500 2 end
    end cell
    read model
    pin-cell model with MOX
    read parm
      sn=16
    end parm
    read materials
      mix=1 com='fuel' end
      mix=2 com='clad' end
      mix=3 com='moderator' end
    end materials
    read geom
      geom=cylinder
      rightBC=white
      zoneIDs 1 2 3 end zoneids
      zoneDimensions 0.42 0.475 0.7406117 end zoneDimensions
      zoneIntervals 3r10 end zoneIntervals
    end geom
    end model
  end sequence
end case

read response[lambda]
  type = grep
  regexp = ":xsdrn.lambda:"
end response

end
```

The distribution of lambda (k-eff) from sample problem 1 is shown in Fig. 6.4.5.

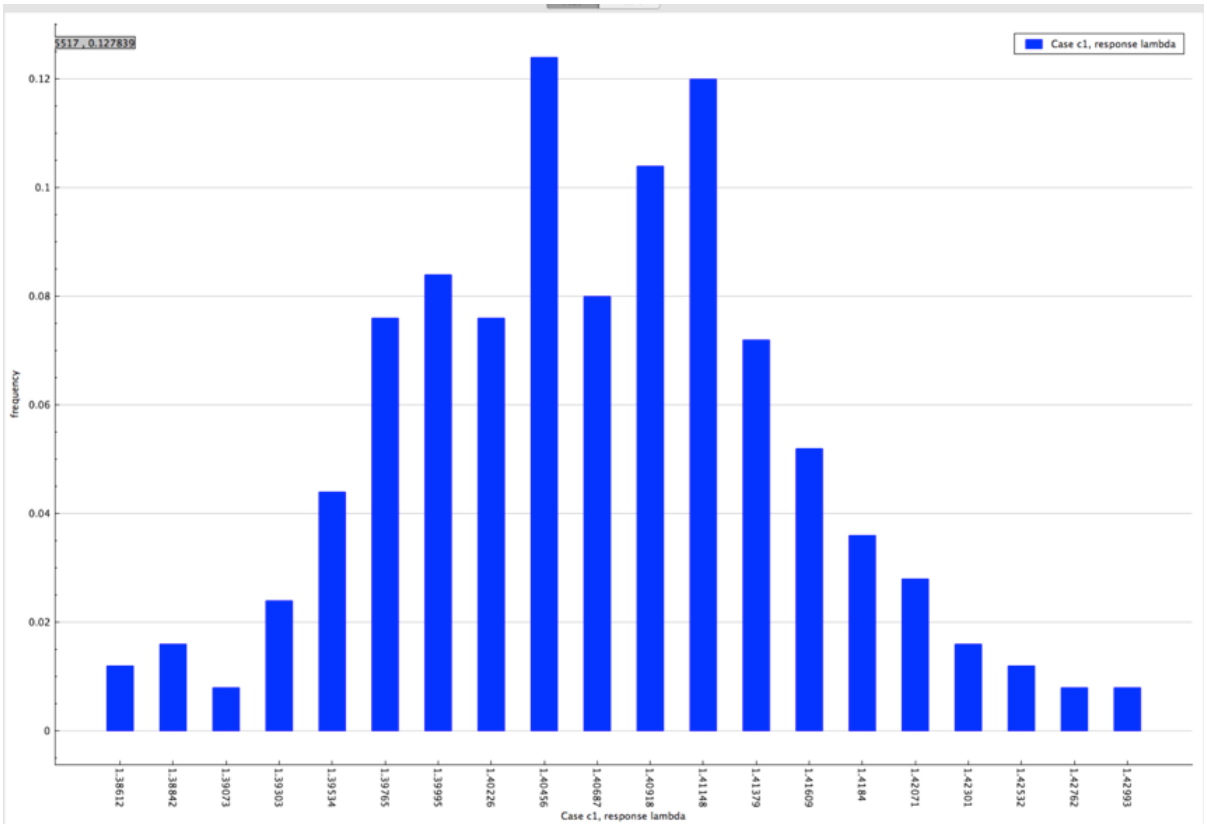


Fig. 6.4.5: Distribution of lambda (k-eff) obtained from sample problem 1.

Sample problem 2

This problem demonstrates a two-dimensional parametric study (using inline placeholders) for two pincell systems.

```
=sampler

read parametric
  variables = temp rho end
  n_samples = 3 5 end
end parametric

read case[c1]
  sequence=t-xsdrn           parm=2region
    pin-cell model with MOX
    xn252
    read comp
      uo2   1 0.95 #{temp} 92235 4.5 92238 95.5   end
      zirc2 2 1 600   end
      h2o   3 den=#{rho} 0.9991 540   end
    end comp
    read cell
      latticecell squarepitch pitch=1.8127 3 fuelr=0.45 1 cladd=0.9500 2 end
    end cell
    read model
    pin-cell model with MOX
    read parm
      sn=16
    end parm
```

(continues on next page)

(continued from previous page)

```
    read materials
      mix=1 com='fuel' end
      mix=2 com='clad' end
      mix=3 com='moderator' end
    end materials
    read geom
      geom=cylinder
      rightBC=white
      zoneIDs 1 2 3 end zoneids
      zoneDimensions 0.45 0.475 1.0006117 end zoneDimensions
      zoneIntervals 3r10 end zoneIntervals
    end geom
    end model
  end sequence
end case

read case[c2]
  sequence=t-xsdrn           parm=2region
    pin-cell model with MOX
    xn252
    read comp
      uo2   1 0.95 #{temp} 92235 4.5 92238 95.5   end
      zirc2 2 1 600   end
      h2o   3 den=#{rho} 0.9991 540   end
    end comp
    read cell
      latticecell squarepitch pitch=1.9127 3 fuelr=0.45 1 cladd=0.9500 2 end
    end cell
    read model
    pin-cell model with MOX
    read parm
      sn=16
    end parm
    read materials
      mix=1 com='fuel' end
      mix=2 com='clad' end
      mix=3 com='moderator' end
    end materials
    read geom
      geom=cylinder
      rightBC=white
      zoneIDs 1 2 3 end zoneids
      zoneDimensions 0.45 0.475 1.0406117 end zoneDimensions
      zoneIntervals 3r10 end zoneIntervals
    end geom
    end model
  end sequence
end case

read variable[rho]
  distribution=uniform
  minimum = 0.5
  value = 0.65
  maximum = 0.8
  cases = c1 c2 end
end variable

read variable[temp]
  distribution=uniform
  minimum = 700
  value = 900
  maximum = 1100
  cases = c1 c2 end
end variable
```

(continues on next page)

(continued from previous page)

```
read response[lambda]
  type = grep
  regexp = ":xsdrn.lambda:"
end response

end
```

The dependency of eigenvalue on coolant density for the two systems is printed to the output file, as shown in Fig. 6.4.6, where the first table shows the temperature and density combinations in the first two columns and the corresponding eigenvalue from case 1 and case 2 in the next two columns. Two summary tables are printed, identifying the conditions of the maximum and minimum for each response, in each case.

```
================================================================================================
=                              Response Tables (parametric study)
================================================================================================

 index \ variable |          temp          rho Case c1, response lambda Case c2, response lambda
------------------+-----------------------------------------------------------------------------
                1 |  7.00000e+02 5.00000e-01              1.47588e+00              1.48621e+00
                2 |  7.00000e+02 5.75000e-01              1.48958e+00              1.49665e+00
                3 |  7.00000e+02 6.50000e-01              1.49781e+00              1.50179e+00
                4 |  7.00000e+02 7.25000e-01              1.50193e+00              1.50300e+00
                5 |  7.00000e+02 8.00000e-01              1.50290e+00              1.50123e+00
                6 |  9.00000e+02 5.00000e-01              1.47007e+00              1.48073e+00
                7 |  9.00000e+02 5.75000e-01              1.48422e+00              1.49162e+00
                8 |  9.00000e+02 6.50000e-01              1.49284e+00              1.49714e+00
                9 |  9.00000e+02 7.25000e-01              1.49730e+00              1.49867e+00
               10 |  9.00000e+02 8.00000e-01              1.49856e+00              1.49718e+00
               11 |  1.10000e+03 5.00000e-01              1.46398e+00              1.47499e+00
               12 |  1.10000e+03 5.75000e-01              1.47860e+00              1.48634e+00
               13 |  1.10000e+03 6.50000e-01              1.48763e+00              1.49226e+00
               14 |  1.10000e+03 7.25000e-01              1.49244e+00              1.49414e+00
               15 |  1.10000e+03 8.00000e-01              1.49402e+00              1.49295e+00


************************************************************************************************
*                                  Parametric study minimum
************************************************************************************************

                  response |          min        temp         rho
---------------------------+-----------------------------------
 Case c1, response lambda |  1.46398e+00 1.10000e+03 5.00000e-01
 Case c2, response lambda |  1.47499e+00 1.10000e+03 5.00000e-01



************************************************************************************************
*                                  Parametric study maximum
************************************************************************************************

                  response |          max        temp         rho
---------------------------+-----------------------------------
 Case c1, response lambda |  1.50290e+00 7.00000e+02 8.00000e-01
 Case c2, response lambda |  1.50300e+00 7.00000e+02 7.25000e-01
```

Fig. 6.4.6: Dependency of lambda (k-eff) on coolant density and fuel temperature for sample problem 2.

Sample problem 3

This sample problem demonstrates enrichment variation using SIREN expressions and dependent variables.

```
=sampler

read parameters
  n_samples=50
  perturb_geometry=yes
end parameters

read case[sphere]
  sequence=csas5     parm=bonami
  sample problem 14  u metal cylinder in an annulus
  xn252
  read comp
    uranium  1 den=18.69 1 300  92235 94.4 92238 5.6 end
  end comp
  read geom
    global unit 1
      cylinder  1 1 8.89 10.109 0.0 orig 5.0799 0.0
      cylinder  0 1 13.97 10.109 0.0
      cylinder  1 1 19.05 10.109 0.0
  end geom
  end data
  end sequence
  read variable[u235]
    distribution=uniform
    minimum=91.0
    value=94.4
    maximum=95.0
    siren="/csas5/comps/stdcomp/wtpt_pair[id='92235']/wtpt"
  end variable
  read variable[u238]
    distribution=expression
    expression="100.0-u235"
    siren="/csas5/comps/stdcomp/wtpt_pair[id='92238']/wtpt"
  end variable
end case

read response[keff]
  type=grep
  regexp=":kenovi.keff:"
end response
```

The distribution of the multiplication factor with the specified enrichment distribution is shown in Fig. 6.4.7.

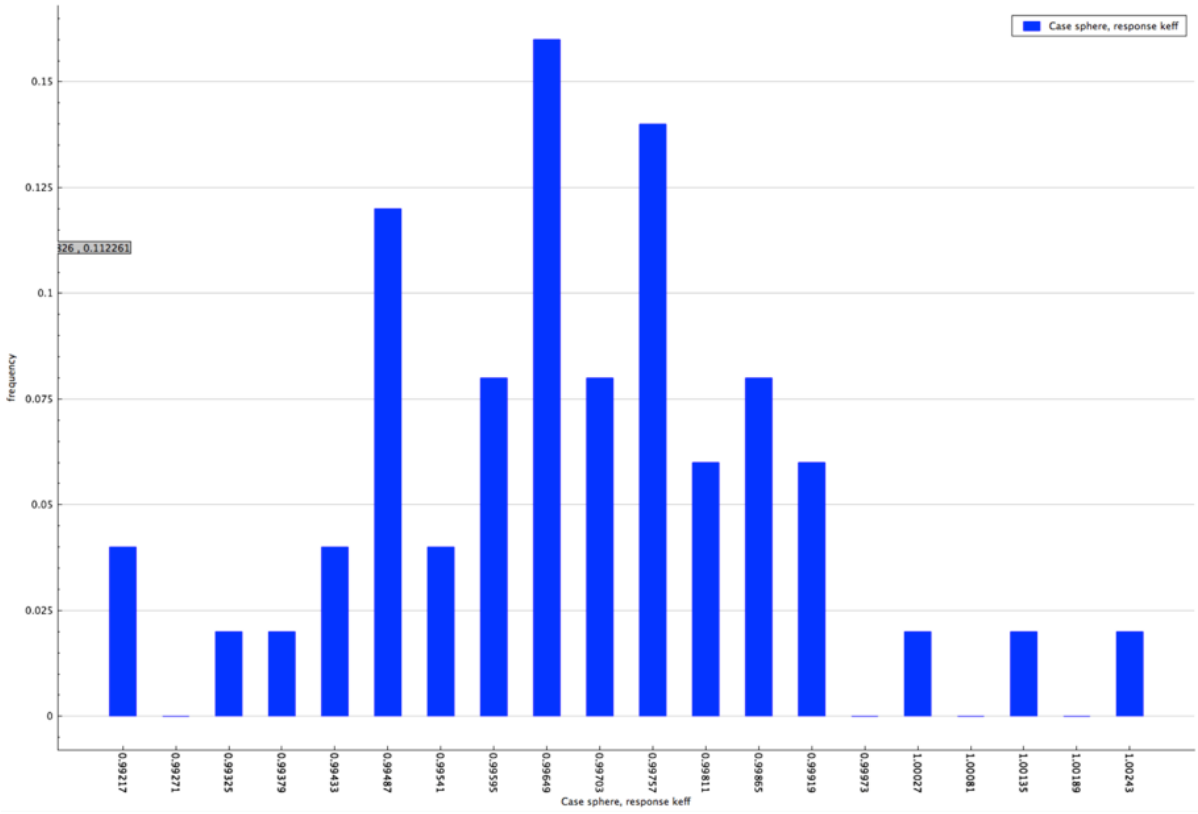


Fig. 6.4.7: Distribution of multiplication factor with sampled enrichment distribution for sample problem 3.

Sample problem 4

Sample problem 4 demonstrates sampling with covariance data for neutron cross sections and fission product yields. Note that decay sampling does not work with TRITON at this time due to not using the perturbed ORIGEN decay libraries *directly*. Additionally, it demonstrates how to extract reaction rates from a TRITON case, combining an additional OPUS run with the `opus_plt` response.

```
=sampler

read parameters
  n_samples=100
  library="xn252"
  perturb_xs = yes
  perturb_decay = no
  perturb_yields = yes
end parameters

read case[c1]
  sequence=t-depl  parm=(bonami,addnux=0)
    pincell model
    xn252
    read composition
     uo2         10 0.95 900 92235 3.6 92238 96.4   end
     zirc2       20 1 600   end
     h2o         30 den=0.75 0.9991 540   end
    end composition
    read celldata
      latticecell squarepitch pitch=1.2600 30 fuelr=0.4095 10 cladr=0.4750 20 end
```

(continues on next page)

(continued from previous page)

```
    end celldata
    read depletion
      10
    end depletion
    read burndata
      power=25 burn=1200 nlib=30 end
    end burndata
    read model
    read materials
      mix=10 com="4.5 enriched fuel" end
      mix=20 com="cladding" end
      mix=30 com="water" end
    end materials
    read geom
      global unit 1
        cylinder 10 0.4095
        cylinder 20 0.4750
        cuboid   30 4p0.63
        media 10 1 10
        media 20 1 20 -10
        media 30 1 30 -20
      boundary 30 3 3
    end geom
    read collapse
     150r1 88r2
    end collapse
    read homog
      500 mini 10 20 30 end
    end homog
    read bounds
      all=refl
    end bounds
    end model
  end sequence
  sequence=opus
    typarams=nuclides
    units=fissions
    symnuc=u238 pu239 end
    library="ft33f001.cmbined"
    case = 10
  end sequence
  sequence=opus
    typarams=nuclides
    units=captures
    symnuc=u238 pu239 end
    library="ft33f001.cmbined"
    case = 10
  end sequence
end case

read response[hmgxs]
  type = triton
  mixture = 1
  data = kinf end
end response

read response[concentrations]
  type=origen_nuclides
  nuclides = u-235 pu-239 pu-240 pu-241 nd-148 sm-149 sm-150 sm-151 cs-133 cs-134 cs-137 end
  mixture = 10
end response

read response[fisrates]
  type = opus_plt
  ndataset = 1
```

(continues on next page)

(continued from previous page)

```
  nuclides = u238 pu239 end
end response

read response[caprates]
  type = opus_plt
  ndataset = 2
  nuclides = u238 pu239 end
end response

end
```

Plots of various isotopic uncertainties as a function of time are shown in Fig. 6.4.8 for ^{235}U, Fig. 6.4.9 for ^{239}Pu, Fig. 6.4.10 for ^{240}Pu, Fig. 6.4.11 for ^{241}Pu, Fig. 6.4.12 for ^{148}Nd, Fig. 6.4.13 for ^{150}Sm, and Fig. 6.4.14 for ^{151}Sm in terms of the unperturbed with error bars for the *minimum* and *maximum*, that is over the 100 samples performed, the minimum value at that time and the maximum value at that time. Note that most isotopes accumulate uncertainty over time, but this is not always the case, e.g. for ^{151}Sm. Two common burnup indicators (measured in spent fuel to infer/confirm burnup) ^{148}Nd and ^{150}Sm show linear increase with time (as expected for a burnup indicator).

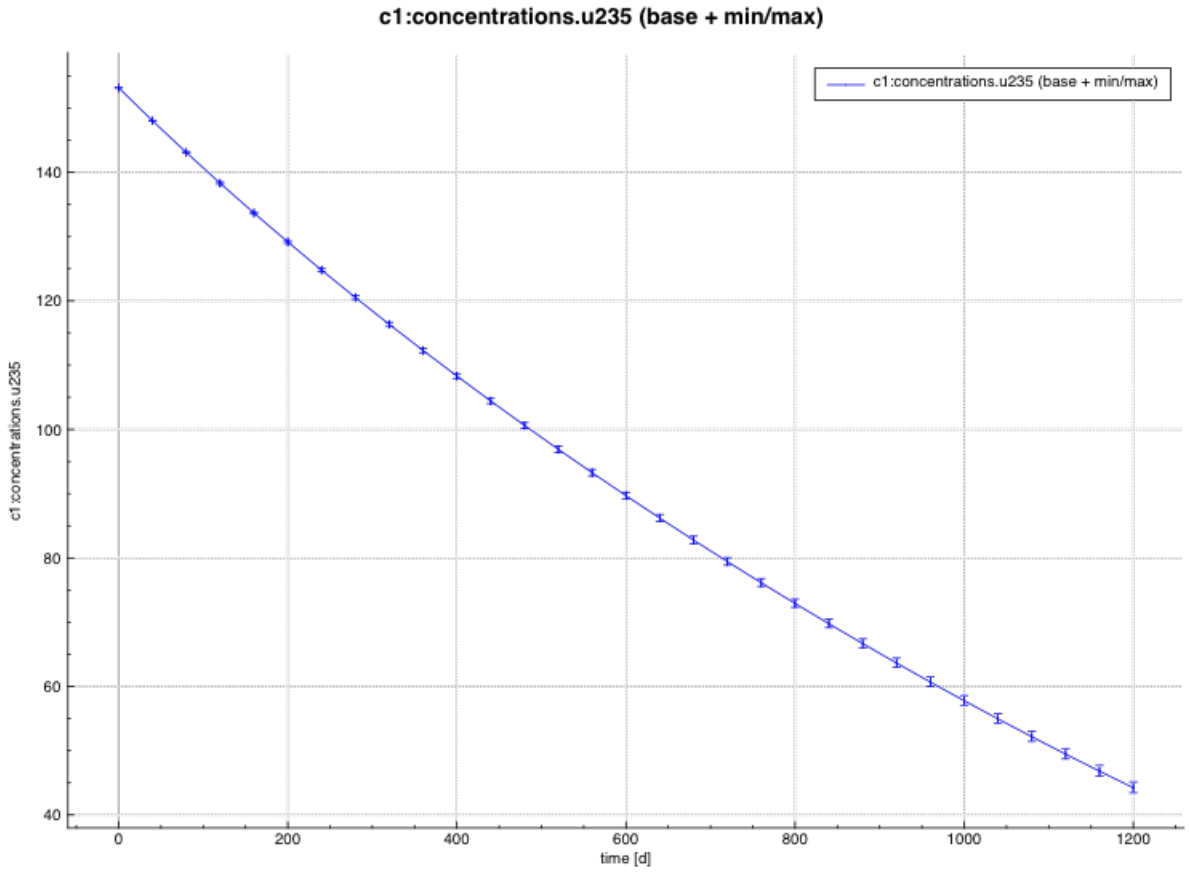


Fig. 6.4.8: Uncertainty in ^{235}U concentration for problem 4.

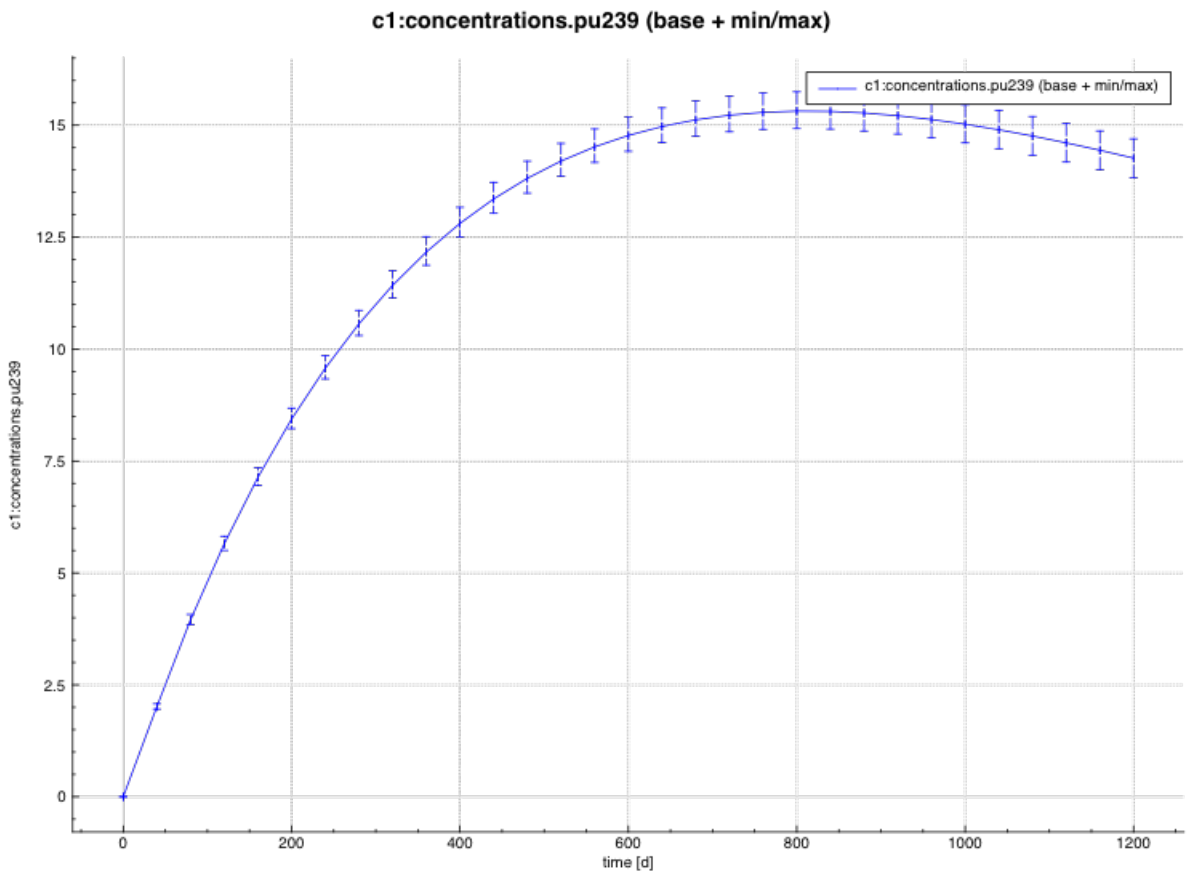


Fig. 6.4.9: Uncertainty in ^{239}Pu concentration for problem 4.

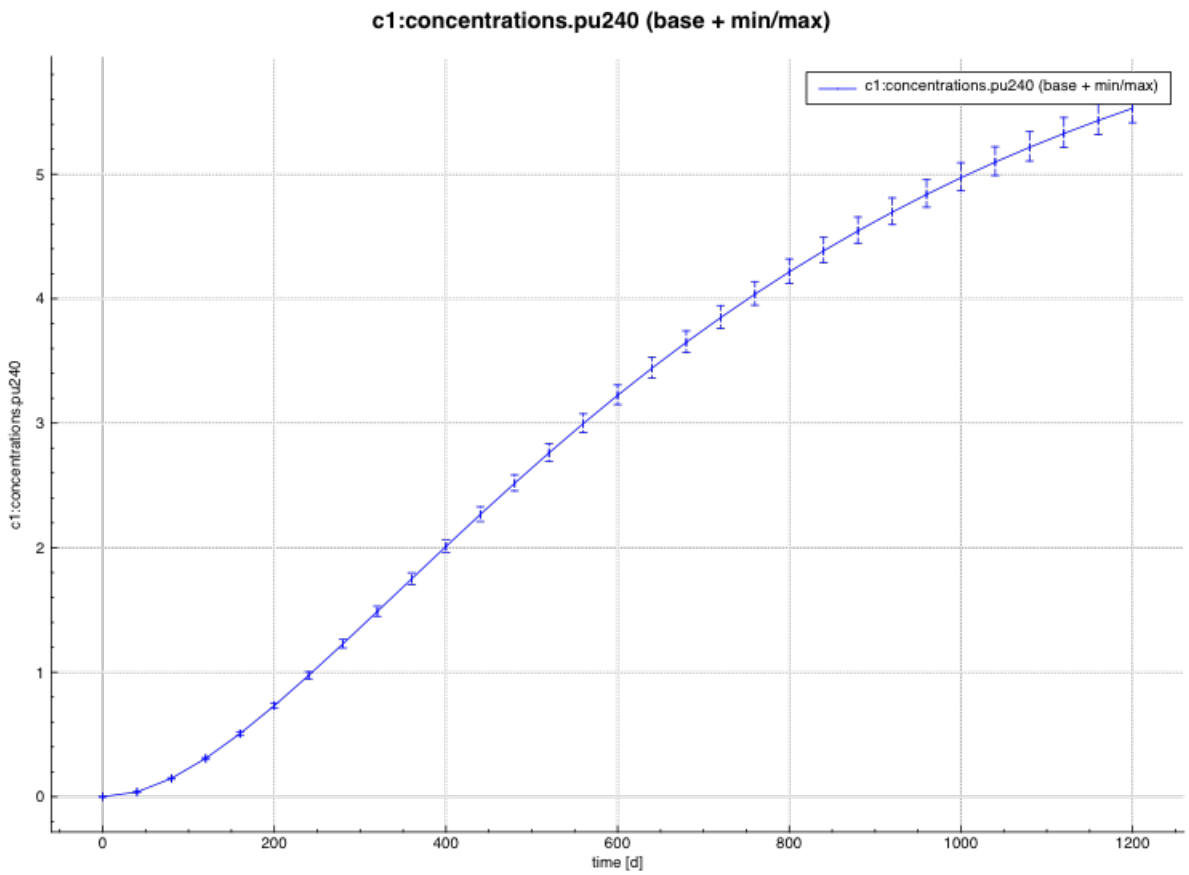


Fig. 6.4.10: Uncertainty in ^{240}Pu concentration for problem 4.

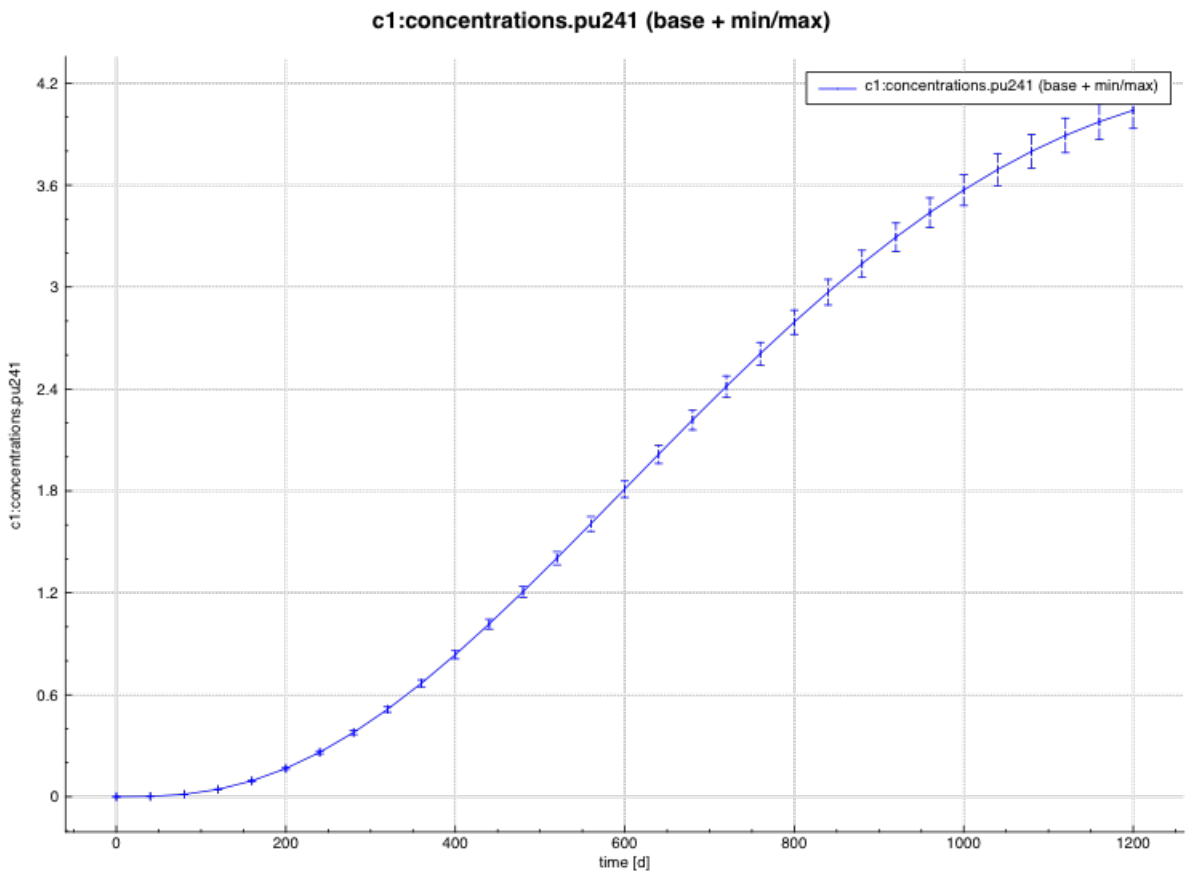


Fig. 6.4.11: Uncertainty in ^{241}Pu concentration for problem 4.

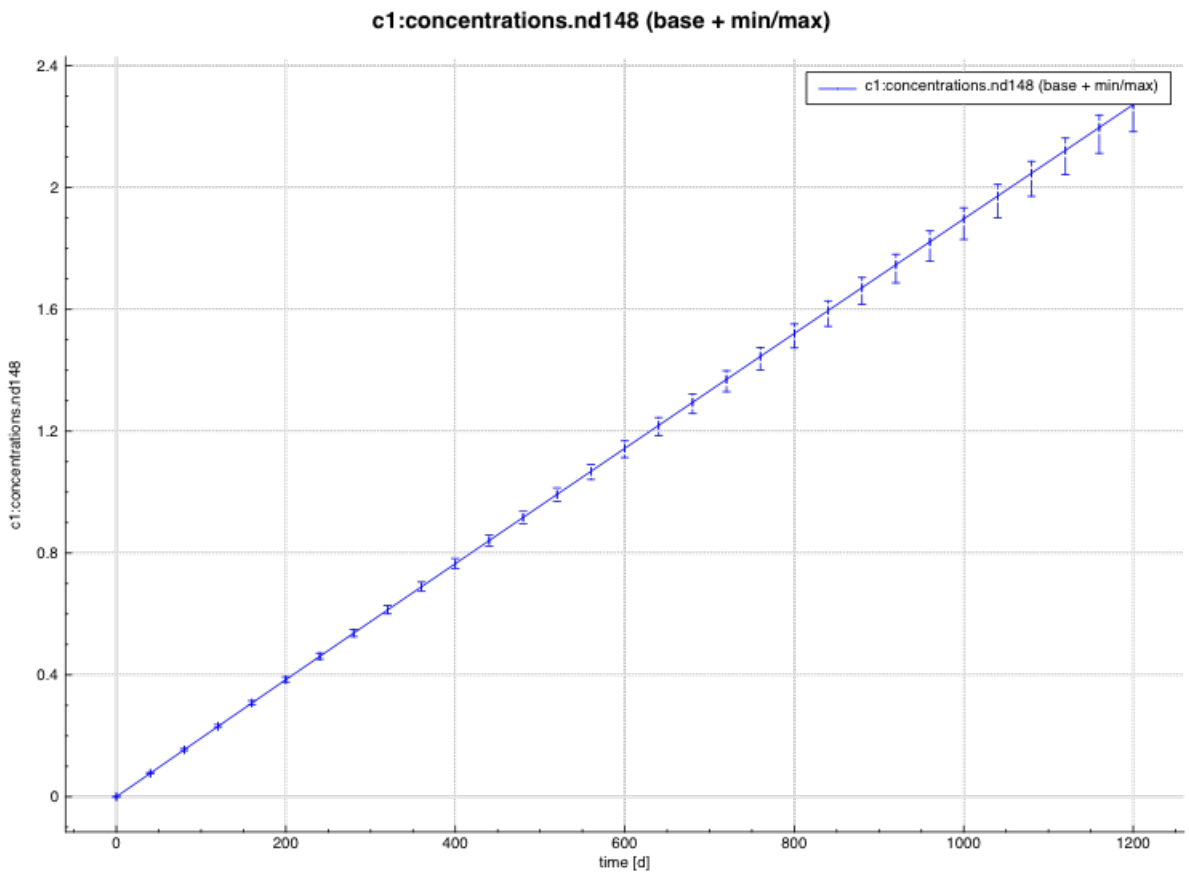


Fig. 6.4.12: Uncertainty in ^{148}Nd concentration for problem 4.

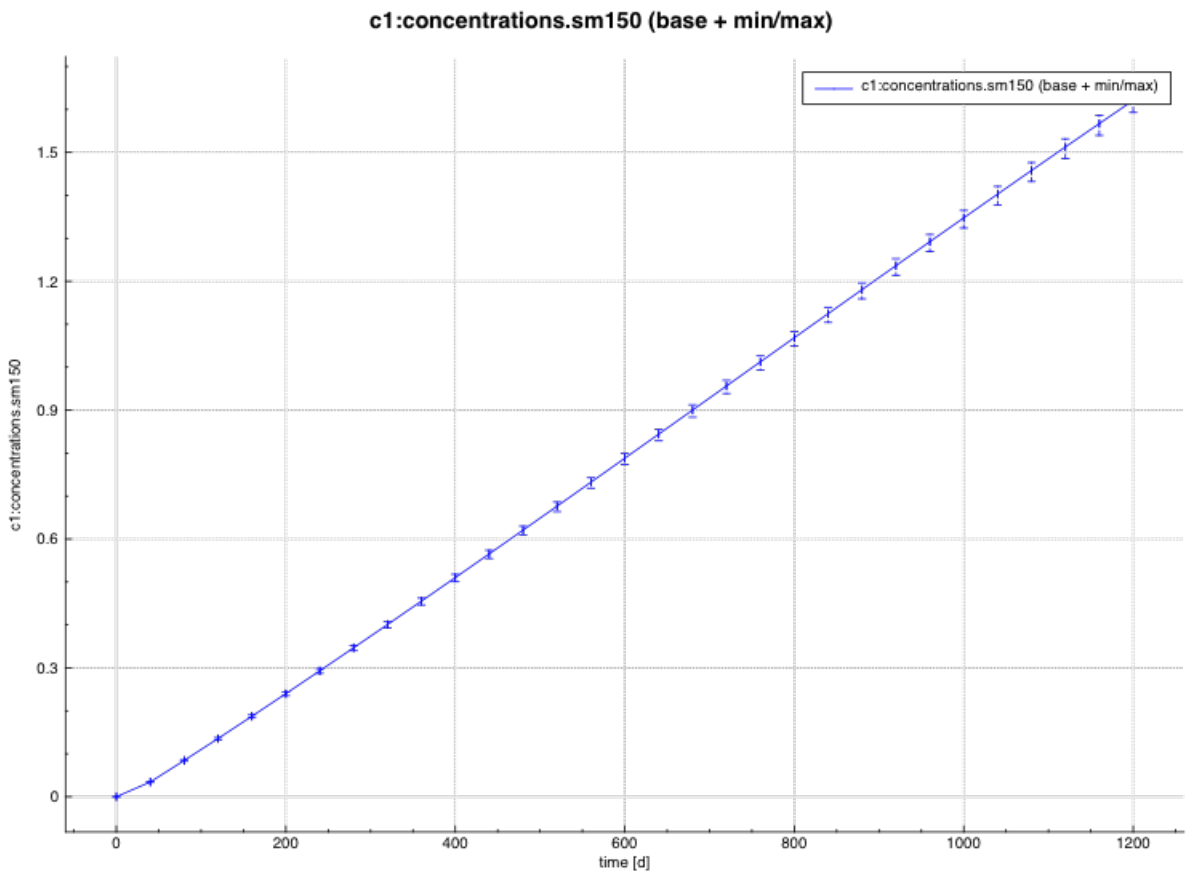


Fig. 6.4.13: Uncertainty in ^{150}Sm concentration for problem 4.

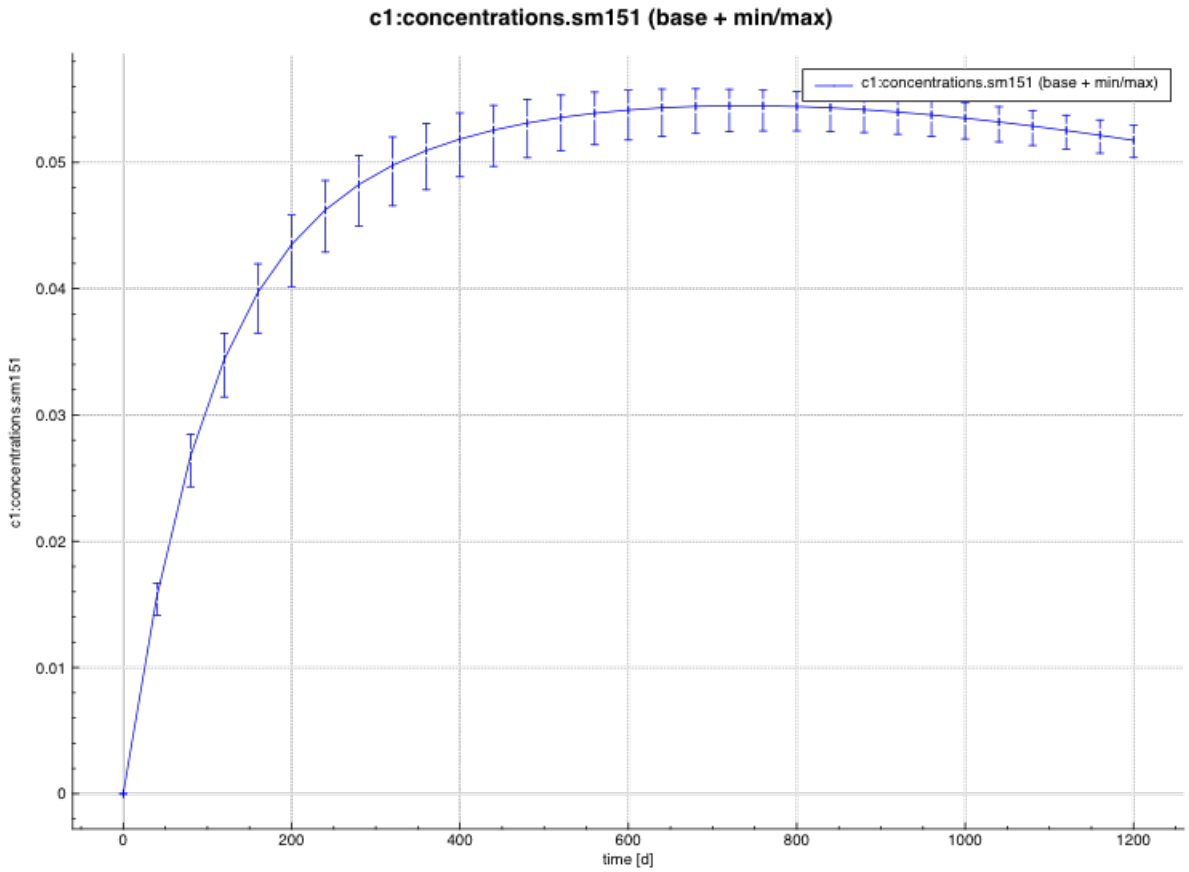


Fig. 6.4.14: Uncertainty in ^{151}Sm concentration for problem 4.

Sample problem 5

This problem provides an example of two correlated depletion cases. The second "c2" case has a larger pitch and lower enrichment fuel. All data libraries are perturbed.

```
=sampler

read parameters
  n_samples = 100
  library = "xn252"
  perturb_xs = yes
  perturb_yields = yes
  perturb_decay = yes
end parameters

read case[c1]
  sequence=t-depl  parm=(bonami,addnux=0)
    pincell model
    xn252
    read composition
     uo2          10 0.95 900 92235 3.6 92238 96.4   end
     zirc2        20 1 600   end
     h2o          30 den=0.75 0.9991 540   end
    end composition
    read celldata
      latticecell squarepitch pitch=1.2600 30 fuelr=0.4095 10 cladr=0.4750 20 end
    end celldata
```

(continues on next page)

(continued from previous page)

```
    read depletion
      10
    end depletion
    read burndata
      power=25 burn=1200 nlib=30 end
    end burndata
    read model
    read materials
      mix=10 com="4.5 enriched fuel" end
      mix=20 com="cladding" end
      mix=30 com="water" end
    end materials
    read geom
      global unit 1
        cylinder 10 0.4095
        cylinder 20 0.4750
        cuboid   30 4p0.63
        media 10 1 10
        media 20 1 20 -10
        media 30 1 30 -20
      boundary 30 3 3
    end geom
    read collapse
     150r1 88r2
    end collapse
    read homog
      500 mini 10 20 30 end
    end homog
    read bounds
      all=refl
    end bounds
    end model
  end sequence
end case

read case[c2]
  sequence=t-depl  parm=(bonami,addnux=0)
    pincell model
    xn252
    read composition
     uo2         10 0.95 900 92235 3.6 92238 96.4   end
     zirc2       20 1 600   end
     h2o         30 den=0.75 0.9991 540   end
    end composition
    read celldata
      latticecell squarepitch pitch=1.6600 30 fuelr=0.4095 10 cladr=0.4750 20 end
    end celldata
    read depletion
      10
    end depletion
    read burndata
      power=25 burn=1200 nlib=30 end
    end burndata
    read model
    read materials
      mix=10 com="3.6 enriched fuel" end
      mix=20 com="cladding" end
      mix=30 com="water" end
    end materials
    read geom
      global unit 1
        cylinder 10 0.4095
        cylinder 20 0.4750
        cuboid   30 4p0.83
        media 10 1 10
```

(continues on next page)

(continued from previous page)

```
        media 20 1 20 -10
        media 30 1 30 -20
      boundary 30 3 3
    end geom
    read collapse
     150r1 88r2
    end collapse
    read homog
      500 mini 10 20 30 end
    end homog
    read bounds
      all=refl
    end bounds
    end model
  end sequence
end case

read response[concentrations]
  type=origen_nuclides
  nuclides = u-235 pu-239 pu-240 pu-241 nd-148 sm-149 sm-150 sm-151 cs-133 cs-134 cs-137 end
  mixture = 10
end response

end
```

The correlation matrix for concentrations at 1200 days of irradiation is shown in Fig. 6.4.15, post-processed with Excel to show high and low correlation coefficients. The bold and underline entries indicate that the each extracted isotope shows a correlation coefficient of 1.0 between the two cases. This means that the two cases are the same in terms of their isotopic response to data perturbations.

	c1:u235	c1:pu239	c1:pu240	c1:pu241	c1:nd148	c1:sm149	c1:sm150	c1:sm151	c1:cs133	c1:cs134	c1:cs137	c2:u235	c2:pu239	c2:pu240	c2:pu241	c2:nd148	c2:sm149	c2:sm150	c2:sm151	c2:cs133	c2:cs134	c2:cs137
c1:u235	1.0	-1.0	-1.0	-1.0	0.1	0.9	0.9	0.1	0.1	0.1	0.0	1.0	-1.0	-1.0	-1.0	0.1	0.9	0.9	0.1	0.1	0.1	0.0
c1:pu239	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0
c1:pu240	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0
c1:pu241	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0
c1:nd148	0.1	-0.1	-0.1	-0.1	1.0	0.1	0.1	0.0	0.0	0.0	-0.2	0.1	-0.1	-0.1	-0.1	1.0	0.1	0.1	0.0	-0.1	-0.1	-0.2
c1:sm149	0.9	-0.9	-0.9	-0.9	0.1	1.0	1.0	-0.1	0.0	0.0	0.0	0.9	-0.9	-0.9	-0.9	0.1	1.0	1.0	-0.1	0.1	0.1	0.0
c1:sm150	0.9	-0.9	-0.9	-0.9	0.1	1.0	1.0	-0.1	0.0	0.0	0.0	0.9	-0.9	-0.9	-0.9	0.1	1.0	1.0	-0.1	0.1	0.1	0.0
c1:sm151	0.1	-0.1	-0.1	-0.1	0.0	-0.1	-0.1	1.0	-0.1	-0.1	-0.1	0.1	-0.1	-0.1	-0.1	0.0	-0.1	-0.1	1.0	-0.1	-0.1	-0.1
c1:cs133	0.1	-0.1	-0.1	-0.1	0.0	0.0	0.0	-0.1	1.0	1.0	0.0	0.1	-0.1	-0.1	-0.1	0.0	0.0	0.0	-0.1	1.0	1.0	0.0
c1:cs134	0.1	-0.1	-0.1	-0.1	0.0	0.0	0.0	-0.1	1.0	1.0	0.0	0.1	-0.1	-0.1	-0.1	0.0	0.0	0.0	-0.1	1.0	1.0	0.0
c1:cs137	0.0	0.0	0.0	0.0	-0.2	0.0	0.0	-0.1	0.0	0.0	1.0	0.0	0.0	0.0	0.0	-0.2	0.0	0.0	-0.1	0.0	0.0	1.0
c2:u235	1.0	-1.0	-1.0	-1.0	0.1	0.9	0.9	0.1	0.1	0.1	0.0	1.0	-1.0	-1.0	-1.0	0.1	0.9	0.9	0.1	0.1	0.1	0.0
c2:pu239	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0
c2:pu240	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0
c2:pu241	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0	-1.0	1.0	1.0	1.0	-0.1	-0.9	-0.9	-0.1	-0.1	-0.1	0.0
c2:nd148	0.1	-0.1	-0.1	-0.1	1.0	0.1	0.1	0.0	0.0	0.0	-0.2	0.1	-0.1	-0.1	-0.1	1.0	0.1	0.1	0.0	-0.1	-0.1	-0.2
c2:sm149	0.9	-0.9	-0.9	-0.9	0.1	1.0	1.0	-0.1	0.0	0.0	0.0	0.9	-0.9	-0.9	-0.9	0.1	1.0	1.0	-0.1	0.1	0.1	0.0
c2:sm150	0.9	-0.9	-0.9	-0.9	0.1	1.0	1.0	-0.1	0.0	0.0	0.0	0.9	-0.9	-0.9	-0.9	0.1	1.0	1.0	-0.1	0.1	0.1	0.0
c2:sm151	0.1	-0.1	-0.1	-0.1	0.0	-0.1	-0.1	1.0	-0.1	-0.1	-0.1	0.1	-0.1	-0.1	-0.1	0.0	-0.1	-0.1	1.0	-0.1	-0.1	-0.1
c2:cs133	0.1	-0.1	-0.1	-0.1	-0.1	0.1	0.1	-0.1	1.0	1.0	0.0	0.1	-0.1	-0.1	-0.1	-0.1	0.1	0.1	-0.1	1.0	1.0	0.0
c2:cs134	0.1	-0.1	-0.1	-0.1	-0.1	0.1	0.1	-0.1	1.0	1.0	0.0	0.1	-0.1	-0.1	-0.1	-0.1	0.1	0.1	-0.1	1.0	1.0	0.0
c2:cs137	0.0	0.0	0.0	0.0	-0.2	0.0	0.0	-0.1	0.0	0.0	1.0	0.0	0.0	0.0	0.0	-0.2	0.0	0.0	-0.1	0.0	0.0	1.0

Fig. 6.4.15: Correlation matrix for concentrations in sample problem 5.

Sample problem 6

Sample problem 6 provides an example of fission yield and decay data sampling. First, COUPLE is run to assemble an ORIGEN library (on file ft33f001) from the perturbed yields and decay data libraries. Burst fission with long decay is then performed by ORIGEN and OPUS is used to extract the fission product decay heat.

```
=sampler

read parameters
  n_samples = 500
  perturb_yields = yes
```

(continues on next page)

(continued from previous page)

```
  perturb_decay = yes
end parameters

read case[heat]
sequence=couple

0$$ a3 80 21 0 33 e
1$$ a4 1 a18 238 e 1t
9**
 13z 1.25 0.3 223z
2t
done
end sequence
sequence=origens
3$$ 1 1 e t
ft33f001
56$$ 40 1 1 0 -1 a9 0 1 2 1 1 e
t
u235 thermal fission energy release
59**  3.08617e12 f0 e
60** 1e-10 1 1.5 2 4 6 8 10 15 20 40 60 80 100
     150 200 400 600 800 1000 1500 2000 4000 6000 8000 1e4
     1.5e4 2e4 4e4 6e4 8e4 1e5
     1e6 1e7 1e8 1e9 1e10 1e11 1e12 1e13
61** f1e-14
67$$ a47 1 0 1 e
68$$ a2 1 e
73$$ 922350
74** 1
75$$ 2
80$$ 0 f1
t
ft71f001
end sequence
sequence=opus
  library="ft33f001"
  units=watts
  libtype=fiss
  minposition=2
end sequence
end case

read response[heat]
  type=opus_plt
  ndataset=0
  nuclides=total end
end response

end
```

The total decay heat with the uncertainty is shown in Table 6.4.7 and Fig. 6.4.16.

Table 6.4.7: Decay heat and uncertainties from sample problem 6.

GEOMETRY (region)

Format: READ GEOM enter geometry region data here END GEOM. See Appendix B.
GEOMETRY REGION DATA consist of SIMPLE GEOMETRY REGION DATA and EXTENDED GEOMETRY REGION DATA.
* *
ENTER DATA IN THE FOLLOWING FORM:
GLOBAL Enter once to specify this unit as the global unit.
UNIT n
COM=delim comment delim This optional comment can be up to 132 characters. It must begin and end with a delimiter (i.e. ", ', *, ^, or !).
Geometry Keyword Label Input Parameters Optional origin data (ORIGIN) Optional chord data (CHORD) Optional rotation data (ROTATE degrees)
Media Keyword Input Parameters
Boundary Keyword Input Parameters Each unit must have one boundary record.
Enter as many geometry and content descriptions as necessary to describe the unit and as many units as necessary to describe the system.

GEOMETRY REGION INPUT DATA REQUIREMENTS

TYPE OF DATA	KEYWORD	INPUT PARAMETERS	NOTES
GEOMETRY	CONE	Lbl Rt Zt Rb Zb	
	CUBOID	Lbl +X –X +Y –Y +Z –Z	
	CYLINDER	Lbl R Zt Zb	
	DODECAHEDRON	Lbl R	
	ECYLINDER	Lbl Rx Ry Zt Zb	
	ELLIPSOID	Lbl Rx Ry Rz	
	HEXPRISM	Lbl R Zt Zb	
	HOPPER	Lbl Lxt Lyt Zt Lxb Lyb Zb	
	PARALLELPIPED	Lbl Xdist Ydist Zdist Psi Theta Phi	
	PENTAGON	Lbl R Zt Zb	
	PLANE	Lbl XPL=a YPL=b ZPL=c CON=d	For geometry keywords, PLANE and QUADRATIC data are entered after the = following the respective subordinate keyword. Only nonzero data are required.
	QUADRATIC	Lbl AQU=a BQU=b CQU=c DQU=d EQU=e FQU=f GQU=g HQU=h IQU=i JQU=j	
	RHEXPRISM	Lbl R Zt Zb	
	RHOMBOID	Lbl Dx Psi	
	RING	Lbl Ri Ro Zt Zb	
	SPHERE	Lbl R	
	WEDGE	Lbl Xbase Xpt Ypt Zlng	
	XCYLINDER	Lbl R Yc Xb	
	XPPLANE	Lbl Xt Xb	
	YCYLINDER	Lbl R Yc Yb	
	YPPLANE	Lbl Yt Yb	
	ZCYLINDER	Lbl R Zc Zb	
	ZPPLANE	Lbl Zt Zb	
MODIFICATION*	CHORD	=X= –X= +Y= –Y= +Z= –Z=	Only nonzero data are required.
	ORIGIN	X= Y= Z=	Only nonzero data are required.
	ROTATE	A1= A2= A3=	The body is rotated about the Origin using the Euler angle X-convention. Only nonzero data are required; input is in degrees.
MEDIA**	MEDIA	Mixture ID Bias ID i1 i2 . . .	
	HOLE	Hole unit number	
	ARRAY	Array ID i1 i2 . . . PLACE nx ny nz xx yy zz	Place positions the array by placing the origin of the unit in array position nx ny nz at xx yy zz. The region boundary containing the array must be enclosed in or on the array boundary.
BOUNDARY	BOUNDARY	i1 i2 ...	Defines the overall volume of the unit.

*Modification data can be entered in any order. Any combination of modification data can follow any geometry record.
**The Region definition vector, i1 i2 . . ., should contain references to all geometry records which share a boundary with t he region being defined.

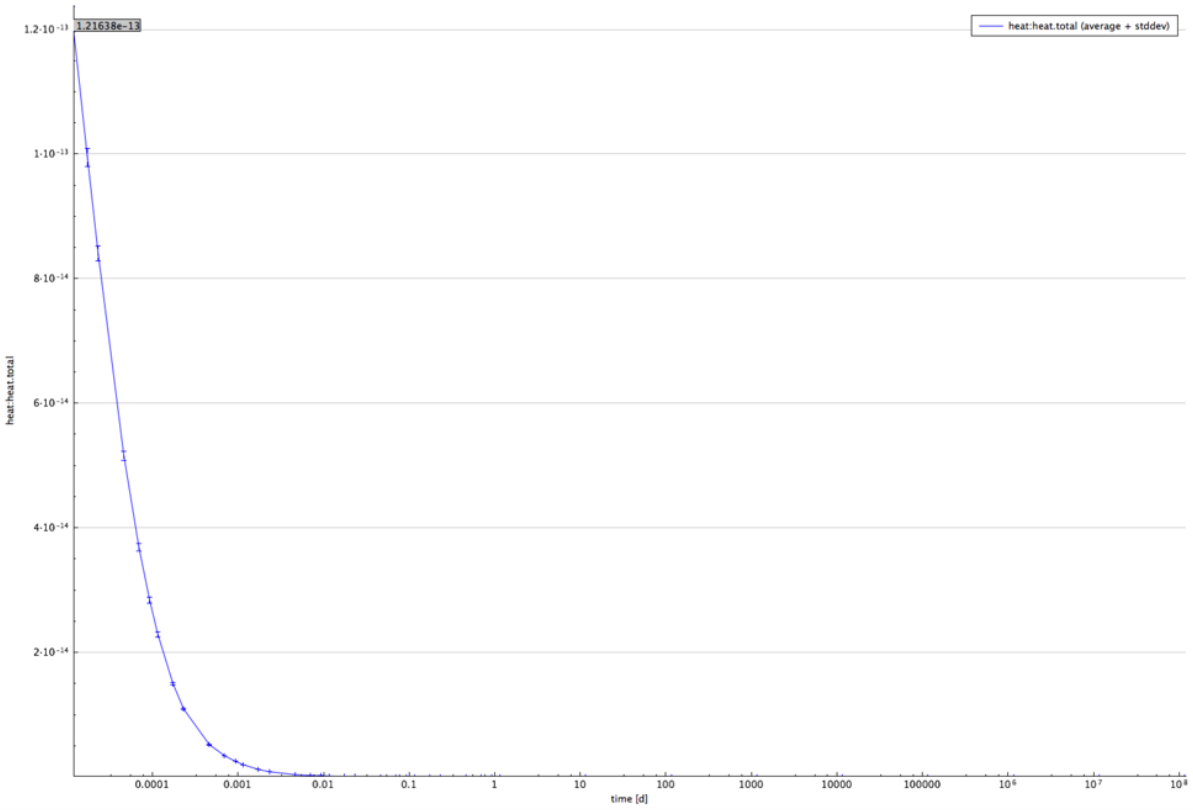


Fig. 6.4.16: Decay heat and uncertainties from sample problem 6.

Sample problem 7

Sample problem 7 illustrates an example of a shielding parametric study with MAVRIC. It also demonstrates usage of the grep response for both response value and uncertainty.

```
=sampler

read parameters
  n_samples = 50
end parameters

read parametric
  variables = thick end
end parametric

read case[graphite]
sequence=mavric
Monaco/MAVRIC Training - Exercise 3. Graphite Shielding Measurements Revisited
v7-27n19g

'------------------------------------------------------------------------------
' Composition Block - standard SCALE input
'------------------------------------------------------------------------------
read composition
    para(h2o)    3         1.0 293.0 end
    carbon       4 den=1.7  1.0 300.0 end
end composition

'------------------------------------------------------------------------------
' Geometry Block - SCALE standard geometry package (SGGP)
'------------------------------------------------------------------------------
read geometry
    global unit 1
        cuboid  1   25.0 -25.0  25.0 -25.0   25.0 -25.0
        cone    2  10.35948  25.01   0.0 0.0   rotate  a1=-90 a2=-90 a3=0
        cuboid  3   #{x90}  70.0  40.0 -40.0   40.0 -40.0
        cuboid 99  120.0 -30.0  50.0 -50.0   50.0 -50.0
        media 3 1  1 -2
        media 0 1  2
        media 4 1  3
        media 0 1  99 -1  -2 -3
    boundary 99
end geometry

'------------------------------------------------------------------------------
' Definitions Block
'------------------------------------------------------------------------------
read definitions
    location 1
        position 110 0 0
    end location
    response 5
        title="ANSI standard (1977) neutron flux-to-dose-rate factors"
        specialDose=9029
    end response
    distribution 1
        title="Cf-252 neutrons, Watt spectrum a=1.025 MeV and b=2.926/MeV"
        special="wattSpectrum"
        parameters 1.025 2.926 end
    end distribution
    gridGeometry 7
        title="large meshes in paraffin, 5 cm mesh for shield thicknesses"
        xLinear 5 -25 25
        xLinear 12 30 #{x90}
        xplanes 100 110 120 -30 end
        yplanes -50 -40 40 50 end
```

(continues on next page)

(continued from previous page)

```
        yLinear 7 -35 35
        zplanes -50 -40 40 50 end
        zLinear 7 -35 35
    end gridGeometry
end definitions

'------------------------------------------------------------------------------
' Sources Block
'   Cf-252 neutrons, Watt fission spectrum model
'   with a=1.025 MeV and  b=2.926/MeV
'------------------------------------------------------------------------------
read sources
    src 1
        title="Cf-252 neutrons, Watt fission spectrum model"
        strength=4.05E+07
        cuboid  0.01 0.01  0 0  0 0
        neutrons
        eDistributionID=1
    end src
end sources

'------------------------------------------------------------------------------
' Tallies Block
'------------------------------------------------------------------------------
read tallies
    pointDetector 1
        title="center of detector"
        neutron
        locationID=1
        responseID=5
    end pointDetector
    meshTally 1
        title="example mesh tally"
        neutron
        gridGeometryID=7
        responseID=5
        noGroupFluxes
    end meshTally
end tallies

'------------------------------------------------------------------------------
' Parameters Block
'------------------------------------------------------------------------------
read parameters
    randomSeed=00003ecd7b4e3e8b
    library="v7-200n47g"
    perBatch=10000 batches=10
    fissionMult=0   noPhotons
end parameters

'------------------------------------------------------------------------------
' Importance Map Block
'------------------------------------------------------------------------------
read importanceMap
    adjointSource 1
        locationID=1
        responseID=5
    end adjointSource
    gridGeometryID=7
    mmTolerance=0.01
end importanceMap

end data
end sequence
```

(continues on next page)

(continued from previous page)

```
read variable[thick]
  value = 20
  minimum = 10
  maximum = 30
  distribution = uniform
end variable

read variable[x90]
  distribution = expression
  expression = "thick+70"
end variable

end case

read response[flux]
  type = grep
  regexp = "total flux[[:space:]]+:scale.number:"
  regexp = ":scale.number:"
end response

end
```

Flux dependence on the shielding thickness is shown in Fig. 6.4.17.

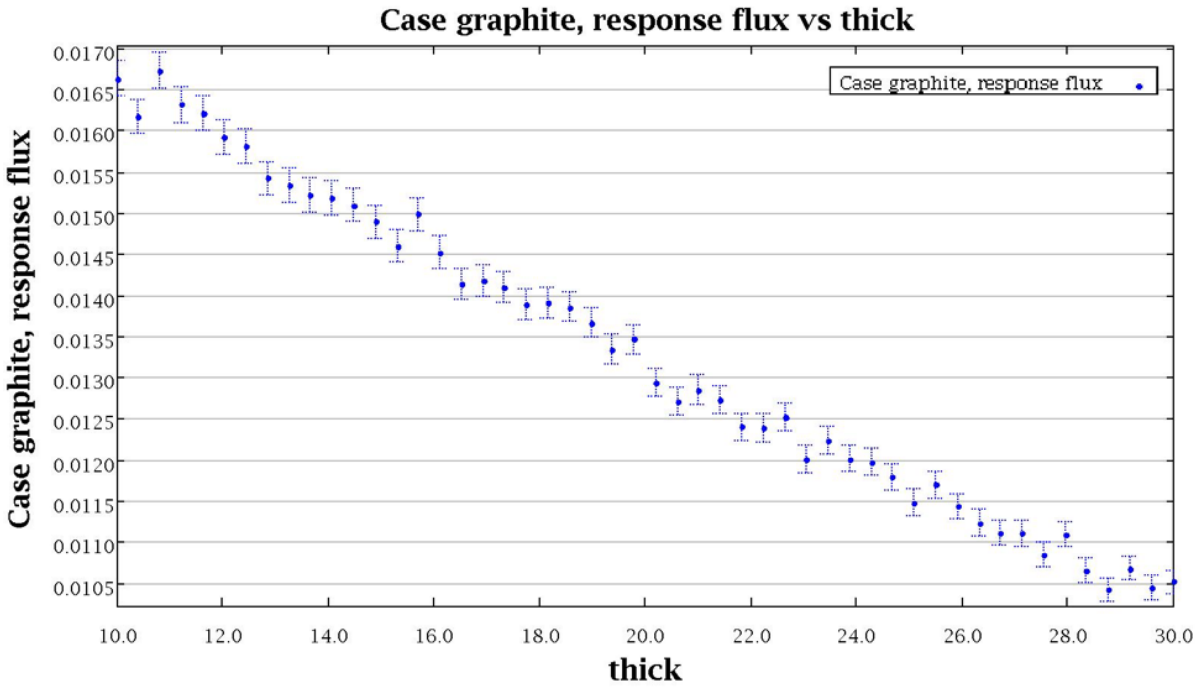


Fig. 6.4.17: Flux as a function of shielding thickness for sample problem 7.

6.4.6 SCALE INPUT RETRIEVAL ENGINE (SIREN)

SIREN (SCALE Input Retrieval ENgine) provides a hierarchical representation of SCALE input and enables access to individual input components or sets of related input parameters. SIREN sorts the contents of supported SCALE input sequences into a tree structure so that every component in a SCALE input can be accessed using an XPath designation. The SIREN package is used by the SCALE super-sequence Sampler to select individual components within an input file, perturb the parameter in memory, and write the perturbed input to a new file. In this way, components within an input file can be perturbed without modification to the original input file, which can be maintained as a read-only file under configuration control for quality assurance purposes.

To explore the use of SIREN expressions, two utilities are available in the SCALE bin directory. The utilities are the executables "InputViewer" and "InputSelector." The InputViewer utility accepts a SCALE input file and outputs the XPath designation for each component of that input file. The InputSelector utility accepts a SCALE input file and a SIREN select statement and then outputs the part of the input file specified by the SIREN select statement. The remainder of this section provides practical examples in the use of SIREN select statements to designate specific components of a SCALE input.

A simple CSAS6 input file, which models a sphere of HEU, is used to show how to use SIREN select statements to access specific portions of a SCALE input. The contents of the example CSAS6 input file are shown below.

```
=csas6
sample problem 6  bare 93.2% U sphere, eighth sphere w/ mirror albedo
v7-238
read comp
  uranium  1 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp
read geometry
  global unit 1
    sphere 10  8.741  chord +x=0.0  chord  +y=0.0  chord +z=0.0
    cuboid   20  8.741 0.0 8.741 0.0 8.741 0.0
    media  1 1 10      vol=2797.5121
    media  0 1 20 -10  vol=2545.3424
    boundary 20
end geometry
read bounds
  -fc=mirror
end bounds
end data
end
```

The InputViewer utility is used to view the contents of the example input file with their respective XPath designations. The utility accepts the path to the input file as an argument so that the usage is:

```
$ InputViewer inputfile
```

A portion of the output from InputViewer is shown below; this portion corresponds to the material definition in the example input file. Notice that each component of the composition has a specific XPath designation. For instance, the "wtptPair" has several pieces of information (ids for each component and the weight percent for that component). To gain access to a specific component, the InputSelector utility is introduced below.

```
/csas6/comps
/csas6/comps/decl(read comp)
/csas6/comps/stdcomp
/csas6/comps/stdcomp/name(uranium)
/csas6/comps/stdcomp/mixture(1)
```

(continues on next page)

(continued from previous page)

```
/csas6/comps/stdcomp/roth
/csas6/comps/stdcomp/roth/decl(den)
/csas6/comps/stdcomp/roth/assign(=)
/csas6/comps/stdcomp/roth/value(18.76)
/csas6/comps/stdcomp/vf(1)
/csas6/comps/stdcomp/temp(293)
/csas6/comps/stdcomp/wtpt_pair
/csas6/comps/stdcomp/wtpt_pair/id(92235)
/csas6/comps/stdcomp/wtpt_pair/wtpt(93.2)
/csas6/comps/stdcomp/wtpt_pair
/csas6/comps/stdcomp/wtpt_pair/id(92238)
/csas6/comps/stdcomp/wtpt_pair/wtpt(5.6)
/csas6/comps/stdcomp/wtpt_pair
/csas6/comps/stdcomp/wtpt_pair/id(92234)
/csas6/comps/stdcomp/wtpt_pair/wtpt(1.0)
/csas6/comps/stdcomp/wtpt_pair
/csas6/comps/stdcomp/wtpt_pair/id(92236)
/csas6/comps/stdcomp/wtpt_pair/wtpt(0.2)
/csas6/comps/stdcomp/term(end)
/csas6/comps/term(end comp)
```

To explore how SIREN select statements are used, the InputSelector utility is used for a few cases with the example CSAS6 input file. Suppose the ^{235}U weight percent needed to be accessed for perturbation; the SIREN expression to access the ^{235}U weight percent is as follows:

Long form:

```
'/csas6/comps/stdcomp/wtpt_pair[id="92235"]/wtpt'
```

Short form:

```
'///wtpt_pair[id="92235"]/wtpt'
```

This statement tells SIREN to look for the weight percent with an ID of 92235 in the weight percent pair. A short form of the select statement (designated by the triple slash) can be used when there are unique qualities about a component; for instance, there is only a single material with an ID of 92235, so the weight percent pair must belong to that material. There are many combinations of select statements that can point to a specific input component, and the developers suggest using the InputSelector utility to familiarize oneself with this capability. The usage of InputSelector is:

```
$ InputSelector inputfile quoted_Select_Statement
```

Below is the screen output from the InputSelector with the example CSAS6 input file and the select statement for the ^{235}U weight percent.

```
---- 1 nodes selected with statement '///wtpt_pair[id='92235']/wtpt' ----
1) /csas6/comps/stdcomp/wtpt_pair/wtpt
93.2
```

The output from InputSelector shows that the select statement points to a single component in the SCALE input, which is indeed the value for the ^{235}U weight percent. There are cases where a select statement points to a portion of the input that has multiple components and the select statement may need to be refined to point to a specific component. For instance, the cuboid defined in the geometry section of the input has multiple dimensions (+x, -x, +y, -y, etc.). By using the select statement below, the dimensions of the cuboid are shown

```
'//cuboid/[id="20"]/dimensions'
```

The output from using InputSelector with this select statement is shown below.

```
---- 1 nodes selected with statement '//cuboid/[id="20"]/dimensions' ----
1) /csas6/geometry/global_unit/cuboid/dimensions

8.741 0.0 8.741 0.0 8.741 0.0
```

The output from InputSelector shows six values, which correspond to the dimensions of the cuboid (+x, x, +y, -y, +z, -z faces). To select a single dimension, the select statement should include which component is desired. For instance, to select the height of the cuboid the select statement becomes:

```
'//cuboid/[id="20"]/dimensions/plus_z'
```

In the same way, the radius of the sphere can be selected using the SIREN expression that identifies the radius of a sphere with an ID of 10:

```
'//sphere/[id="10"]/dimensions/r'
```

The output from InputSelector is shown below.

```
---- 1 nodes selected with statement '//sphere/[id="10"]/dimensions/r' ----
1) /csas6/geometry/global_unit/sphere/dimensions/r

8.741
```

There are scenarios where there is no unique or obvious identifier for a component. As an example, the media volume can utilize an array index to retrieve the volume value.

```
'//media/volume/value'
```

The output from InputSelector illustrates both media's volume value are selected below.

```
---- 2 nodes selected with statement '//media/volume/value' ----
1) /csas6/geometry/global_unit/media/volume/value
2797.5121
2) /csas6/geometry/global_unit/media/volume/value
2545.3424
```

The printed index (1) can be utilized to narrow the selection.

```
‘//media[1]/volume/value’
```

Here the first media is selected and subsequently, the media's volume value, removing the selection of the second media entirely. The output from InputSelector is shown below.

```
---- 1 nodes selected with statement '//media[1]/volume/value' ----
1) /csas6/geometry/global_unit/media/volume/value
2797.5121
```

For the sake of completeness, the following example illustrates selection of the media's volume value via the mixture's value comparator.

```
'//media[mixture="1"]/volume/value'
```

This illustrates the first media's unique mixture component can be used to select the volume for the first media. The output from InputSelector is shown below.

```
---- 1 nodes selected with statement '//media[mixture="1"]/volume/value' ----
1) /csas6/geometry/global_unit/media/volume/value

2797.5121
```

The above examples were intended to provide users with an idea of how to use SIREN select statements to access specific components within a SCALE input. The select statements are primarily used in cases where the input is to be perturbed by the Sampler super-sequence, but modification to the original input is not desirable. The recommended way to ensure that the select statements are correct is to use the InputSelector utility to check that a select statement points to the correct input component.

For viewing the input paths, an alternative to the InputViewer command line utility is available within the Fulcrum graphical user interface (GUI). By clicking on the Run Button drop-down and selecting "Input Listing" as shown in Fig. 6.4.18, the full listing of the current input file is displayed in the "Messages" box. Also, notice the path is displayed in the bottom left corner for the input component located at the cursor.

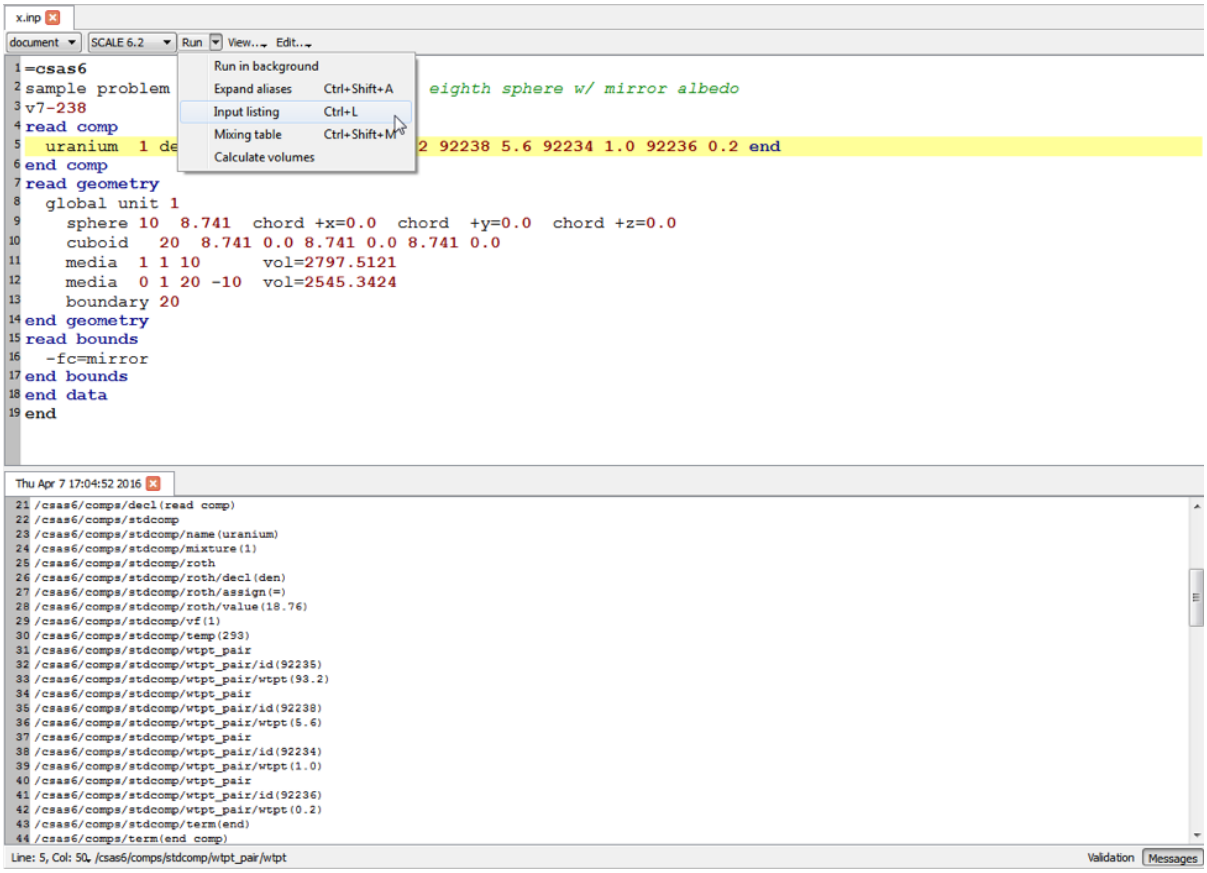


Fig. 6.4.18: Input Listing in Fulcrum.

A table of SIREN expressions is as follows.

Table 6.4.8: Expressions for SIREN.

Expressions	Description	Examples
component	Component match.	csas6
/	Component separator. When the expression begins with this, it selects the root of the document.	csas6/xslib /
//	This expression is the 'Any Child' select.	csas6//uranium //
[index]	Array index selector. When an array of objects has been selected, this expression selects the object at the given index.	csas6/ geometry/global_unit/media**[1]**
[sindex:eindex]	Subarray index selector. When an array of objects has been selected, this expression selects a subarray.	csas6/ comps/stdcomp/uranium/wtpt_pair**[2:3]**
[sindex:eindex:stride]	Subarray index selector. When an array of objects has been selected, this expression selects a subarray and subsequently components stride apart.	csas6/comps/stdcomp/uranium/wtptPair**[1:4:2]**
*	Component wild card. This can only be used in place of a component or to wild card the remainder of a component.	csas6/read comp/* csas6/read comp/ur*

6.4.7 EXPRESSION OPERATORS AND FUNCTIONS FOR SAMPLER

The expression engine utilized by Sampler supports all basic mathematical and logical operators, as well as a number of functions. Logical expressions and functions evaluate to the number 1 for true, and the number 0 for false.

Table 6.4.9: Expression operators and functions for Sampler.

Operator	Description	Example
+	Addition	3+3, x+y
+	Unary Plus	+3, +(-3), +x
-	Subtraction	3-3, x-y
-	Unary Minus	-3, -(3+3), -x
*	Multiplication	3*3, x*y
/	Division	3/3, x/y
%	Modulo	3%3, x%y
^	Power	3^3, x^y
==	Equal to	3==3, x==y
!=	Not equal to	3!=3, x!=y
<	Less than	3<3, x<y
>	Greater than	3>3, x>y
<=	Less than or equal to	3<=3, x<=y
>=	Greater than or equal to	3>=3, x>=y

continues on next page

Table 6.4.9 – continued from previous page

Operator	Description	Example
and	Left is true and right is true	1 and 1 , x==y and y==z
or	Left is true or right is true	1 or 1, x==y or y==z
nand	Not and	1 nand 1, 1 nand 0, x nand y
xor	Exclusive or	1 xor 1, 1 xor 0, x xor y
nor	Not or	1 nor 1, 1 nor 0, x nor y
abs(x)	Absolute value	abs(1),abs(-1),abs(x) ,abs(y)
min(x,. . . n)	Minimum of values	min(1),min(1,2,3,4,5, x,y),etc
max(x,. . . n)	Maximum of values	max(1),max(1,2,3,4,5, x,y),etc
avg(x,. . . n)	Average of values	avg(1),avg(1,2,3,4,5, x,y),etc
Sum(x,. . . n)	Summation of values	sum(1),sum(1,2,3,4,5, x,y),etc
mul(x,. . . n)	Multiply values	mul(1),mul(1,2,3,4,5, x,y),etc
floor(x)	Floor of the value	floor(1)=1, floor(-1.0)=-1.0, floor(-1.1)=-2.0
ceil(x)	Ceiling of the value	ceil(1)=1,ceil(-1.1)= -1
round(x)	Round the value to nearest integer	round(1.1)=1,round(1.5)=2,round(1.9)=2
roundn(x,n)	Round the value to the nth decimal position	roundn(1.15,1)= 1.2, roundn(1.519,2)= 1.52
cos(x)	Cosine of the value	cos(0.0)=1
sin(x)	Sine of the value	sin(0.0)=0
tan(x)	Tangent of the value	tan(0.0)=0
sec(x)	Secant of the value	sec(0.0)=1
csc(x)	Cosecant of the value	csc(1.0)= 1.1884
cot(x)	Cotangent of the value	cot(1.0)= 0.642093
sinh(x)	Hyperbolic sine of the value	sinh(1.0)= 1.1752
cosh(x)	Hyperbolic cosine of the value	cosh(1.0)= 1.54308
tanh(x)	Hyperbolic tangent of the value	tanh(1.0)= 0.761594
log(x)	Natural log of the value	log(2.7182818)=1
log10(x)	Common log of the value	log10(10.0)=1
exp(x)	Exponential of the value	exp(1)=2.7182818
clamp(l,x,u)	Clamp the value x between lowerbound l and upperbound u	clamp(-1,1,+1)=1,clamp(-1,-5,+1)=-1
inrange(l,x,u)	Determine if value x is greater than or equal to lowerbound l, and less than or equal to upperbound u	inrange(-1,1,+1)=1,inrange(-1,-5,+1)=0
sgn(x)	Determine the sign of x.	sgn(+1)=1,sgn(-1)=-1, sgn(0)=0,sgn(5)=1
sqrt(x)	Square Root of the value	sqrt(1)=1,sqrt(4)=2
root(x,y)	Root of x given y	root(4,2)=2,root(8,3)=2
true	Constant true value	true
false	Constant false value	False
if(condition,true,false)	If function that evaluates the condition. If true, the true value is substituted, else the false value is substituted	if(x > y, x, y) = x

6.4.8 GUIDELINES FOR RUNNING SAMPLER IN PARALLEL

Sampler calculations consist of three basic steps:

1. generate N perturbed input files and libraries,
2. run N independent SCALE jobs, and
3. analyze N results sets.

At this time, the preferred way of performing Sampler calculations is to use *run_cases=no* (in the *parameters* block) and perform step 2 above outside of Sampler, i.e. using scripting or your system's queuing system to launch the *N* SCALE jobs. The "${OUTBASENAME}.samplerfiles" directory that Sampler creates in step 1 will contain all the perturbed input files. Each of these input files should be run as a separate SCALE job. When all jobs are complete, Sampler should be run again (still with *run_cases=no*). Sampler will see that all jobs have completed, and perform the analysis in step 3. The Sampler sample problem 8 is a quick-running calculation that can be used to test and, if necessary adjust, the manual 3-step scheme described above on your system.

However, for small jobs that fit on a single node, threading can be used. To launch a Sampler calculation with 5 parallel threads, one would issue the following from the command line:

```
scalerte -r -I 5 -m sample_8.inp
```

This will perform all 3 steps of the Sampler process.

6.5 VADER: TRENDING ANALYSIS FOR CODE/DATA VALIDATION

S. Hart, J. Clarity

VADER (Validation Analysis Data Evaluation Resource) is a new module in SCALE 6.3 derived from the legacy USLSTATS program. VADER is a tool that allows the determination of bias and bias uncertainty for criticality safety computational methods. The older USLSTATS program, written in Java, existed outside of SCALE and provided tools to calculate only the confidence band with administrative margin (sometimes called USL-1) and the single-sided uniform width closed interval (USL-2). For normality testing it offered only a crude chi-squared normality test that had no user-configurable options and presented a simple pass/no-pass functionality.

VADER has been completely written in C++ using modern coding practices and standards. It has been moved to be part of the SCALE code package and is run through the normal SCALE executables. To support the VADER methods, large updates were made to the statistical capabilities of SCALE. Interfaces to known statistical distributions were added and many of the statistical calculations required for VADER were made available to all SCALE sequences. In addition to being able to process legacy USLSTATS inputs, VADER adds support for the single-sided tolerance band method (from NUREG/CR-6698) as well as two non-trending methods: the "historical" nonparametric and parametric methods, also as described in NUREG/CR-6698. VADER also includes several new tests for normality in addition to adding user control options on the original chi-squared normality test. A new t-test for the statistical significance of the trend is included.

VADER has also been integrated into the Fulcrum GUI used by SCALE. This allows VADER to benefit from the auto-complete, templating, and input validation features offered by Fulcrum. In addition, Fulcrum can plot the results given by VADER, and the plot files have been expanded to include the new methods available in VADER.

6.5.1 THEORY

The methods and theory for VADER are largely taken from NUREG/CR-6698. That document contains equations and descriptions of the methods used in VADER. It also includes working examples, some of which data was used to verify the results and create example problems for VADER.

6.5.2 INPUT

The VADER input is based on the SON input syntax common in other modernized SCALE codes such as ORIGEN and ORIGAMI. The SON input format consists of key/value pairs where the value can be either a single item, an array of items, or other SON blocks.

The VADER input generally just has one level of depth. There are no SON blocks inside of SON blocks. All input is mostly arrays at the top level of the tree. A quick look at all the top level items is given in Table 6.5.1. Each top level input will then be described in detail in the next section.

Table 6.5.1: Summary of top level options for VADER

Keyword	Required	Description
data	Yes	Data entered in triplets in the form `X, Y, std. dev`.
title	No	Title to give the case.
parameters	No	Global parameters that apply to each test or trend.
trend_values	No	User desired trend values to run the tests on.
methods	Yes	Methods that the user wishes to apply to the data.
tests	No	Statistical checks that the user wishes to apply to the data.

6.5.3 DETAILED INPUT

6.5.3.1 `data`

The `data` input array is a flat array consisting of data triplets. There is no punctuation between the triplets. The data is entered in the order $X_1\ Y_1\ \sigma_1\ X_2\ Y_2\ \sigma_2\ ...\ X_N\ Y_N\ \sigma_N$.

Where

X	The trended value, usually something like enrichment or moderator density.
Y	The y value, usually something like C/E or eigenvalue.
σ	The standard deviation of Y.

VADER will emit an error if the number of entries in this array is not divisible by three.

Example 6.5.1: Snippet of `data` input

```
data=[2.4 0.9976 0.00122 2.6 0.9978 0.00022
      ...
      1.9 1.0011 0.00213 1.99 1.013 0.00012]
```

6.5.3.2 `title`

The `title` entry allows the user to specify a title for the case. The title is cosmetic and doesn't affect any of the results. The title is echoed to the output in several places and is included on some of the graphs generated by VADER.

Example 6.5.2: `title`

```
title="My Very First VADER Case"
```

6.5.3.3 `parameters`

The `parameters` section can exist outside of any test or method. If a parameter is listed in this global section, then it will apply to any method or test that has that parameter available instead of the default value. Values in this global `parameters` section can be overridden in the specific method or test block for which the user wishes to override the global parameter. For a list of all parameters, their default values, and which tests and methods they are applied to, see Sect. 6.5.6.

Example 6.5.3: `parameters` array

```
parameters {extrapolate=yes admin_margin=0.03}
```

6.5.3.4 `trend_values`

This array allows the user to insert values at which they wish the trend Y (typically the upper subcritical limit) values to be calculated. Extrapolation outside of the data range is only available if the method supports it and extrapolation is enabled.

Example 6.5.4: `trend_values` array

```
trend_values=[-0.1 0.2 2.4 10.0]
```

6.5.3.5 `methods`

This is where the user specifies the methods to enable and, optionally, gives method-specific parameters. Each method is followed by an array of the parameters the user wishes to set. Any omitted parameters are initialized with the default value (see Table 6.5.2). If the user wishes to enable a method with all default values, then an empty array can be given.

Example 6.5.5: *methods* block with some default methods and some specific parameters

```
methods {
  USL1 {} % uses default parameters
  CR6698 {weighted=yes admin_margin=0.03 chi_p=0.97}
  Parametric {confidence=0.9}
}
```

6.5.3.6 `tests`

This block is similar to the *methods* block, but it allows the user to specify which tests they wish to enable. These tests are run on the data and generally either pass or fail and give a few statistical values as their result. Most of the tests implemented for VADER for now are normality tests.

Example 6.5.6: *tests* block

```
tests {
  chisq {} % default parameters
  t-test {confidence=0.98}
}
```

6.5.4 AVAILABLE METHODS

The sections below indicate the input name to be used to activate the available method. Details on the methods can currently be found in NUREG/CR-6698.

6.5.4.1 `USL1` - Confidence Band with Administrative Margin

6.5.4.2 `USL2` - Single-Sided Uniform Width Closed Interval Approach

6.5.4.3 `CR6698` - Single-Sided Lower Tolerance Band

6.5.4.4 `NonParametric` - Historical Non-Parametric Method

6.5.4.5 `Parametric` - Historical Parametric Method

6.5.5 AVAILABLE TESTS

The sections below indicate the input name to be used to activate the available test. Details on the tests can currently be found in NUREG/CR-6698.

6.5.5.1 `chisq` - χ^2 Normality Test

6.5.5.2 `t-test` - Trend Significance t-test

6.5.5.3 `anderson_darling` - Anderson-Darling Normality Test

6.5.6 AVAILABLE PARAMETERS

Table 6.5.2 shows all of the available parameters, their default value, and the methods and tests to which they apply. Any parameter can be put into the global `parameters` block. Only parameters that apply to a specific method or test can be put into that method or tests `parameters` block. Following the table is a more detailed description of each parameter.

Table 6.5.2: Available Parameters for VADER

Keyword	Default Value	Available In
tsunami	Off	USL1 USL2
extrapolate	Off	USL1
fit_confidence	0.95	USL1 USL2
admin_margin	0.05	USL1 USL2 CR6698 NonParametric Parametric
xrange	All X Values	USL1 USL2 CR6698 NonParametric Parametric
proportion_confidence	0.95	USL2
proportion	0.95	USL2
f_p	0.95	CR6698
normal_p	0.95	CR6698
chi_p	0.95	CR6698
weighted	No	CR6698 NonParametric Parametric
confidence	0.95	NonParametric Parametric
proportion_of_pop	0.95	NonParametric Parametric
use_npm	No	NonParametric

6.5.6.1 `tsunami`

This enables a TSUNAMI-mode calculation for VADER. It requires that all X values be between -1.0 and 1.0.

6.5.6.2 `extrapolate`

This allows extrapolation past the data values entered by the user. This is generally only applicable when the user has input values in the trend_values block.

6.5.6.3 `fit_confidence`

Confidence in the statistical result. Sometimes referred to as $1 - \gamma$. In USL1 this is used to calculated the t-statistic quantile.

6.5.6.4 `admin_margin`

Administrative margin subtracted from the result. This can be used to add an extra layer of confidence that the value reported is below criticality.

6.5.6.5 `xrange`

Allows the user to tailor X values for each method or test or change the values for all tests without having to change the *data* block. Data is entered as an array of two values. By default all of the X values are used.

Example 6.5.7: xrange

```
xrange=[-0.5 2.5]
```

6.5.6.6 proportion_confidence

In USL2 this controls the confidence on the proportion that is in the confidence band.

6.5.6.7 proportion

In USL2 this controls the proportion of the data points that will be inside the confidence band.

6.5.6.8 f_p

The percentile used for the F-distribution in the single-sided lower tolerance band method.

6.5.6.9 normal_p

The percentile used for the normal distribution in the single-sided lower tolerance band method.

6.5.6.10 chi_p

The percentile used for the χ^2 distribution in the single-sided lower tolerance band method.

6.5.6.11 weighted

Whether to use the weighted version of the mean and standard deviation equations when calculating the statistics for this method.

6.5.6.12 confidence

Statistical confidence that the proportion calculated to be above the cutoff is correct.

6.5.6.13 proportion_of_pop

Proportion of the population that will be above the cutoff value in the non-parametric method.

6.5.6.14 use_npm

Use (add) non-parametric margin to the non-parametric method.

6.6 SAMS: SENSITIVITY ANALYSIS MODULE FOR SCALE

K. B. Bekar, J. D. McDonnell, B. T. Rearden, W. A. Wieselquist

ABSTRACT

The **S**ensitivity **A**nalysis **M**odule for **S**CALE (SAMS) calculates sensitivity coefficients that predict the expected changes of the calculated value of system responses such as $k_{\text{eff}}$and reaction rate ratios due to perturbations in constituent cross-section data. Additionally, the uncertainty in each response due to cross section covariance data can be predicted. For multigroup calculations, SAMS performs sensitivity calculations using linear perturbation theory and requires the calculation of the forward and adjoint flux moments. Multigroup sensitivity coefficients computed by SAMS also include the implicit effect due to resonance self-shielding calculations. In multigroup mode, SAMS works in conjunction with XSDRNPM, NEWT, KENO V.a, or KENO-VI, which are capable of calculating the desired responses as well as associated fluxes, flux moments and/or mesh fluxes. The implicit calculations are performed with data generated by a

combination of the BONAMIST and SENLIB codes specially developed to produce resonance self-shielded cross sections and their sensitivities to input data. SAMS is also used to edit the sensitivity coefficients generated by KENO continuous energy sensitivity calculations.

SAMS automatically selects all of the sensitivity parameters that can be calculated or edited for each nuclide in each region of the system based on available cross-section data. Sensitivity parameters for a given nuclide may be generated for a number of parameters, including total, scatter, capture, and fission cross sections, as well as χ. The uncertainty information is produced for all processes available on the cross section covariance data file.

SAMS has been designed for automated operation with the TSUNAMI-1D, TSUNAMI-3D-K5 and TSUNAMI-3D-K6 SCALE control modules and produces data suitable for use with the TSUNAMI-IP, TSURFER, and TSAR modules.

Version Information

Version 6.2 (2016)

Code Responsible(s): B. T. Rearden, L. M. Petrie, M. A. Jessee, M. L Williams

The following section acknowledgements appeared in the SCALE 6.2 manual.

The author wishes to acknowledge B. L. Broadhead of the Oak Ridge National Laboratory and R. L. Childs, formerly of the Oak Ridge National Laboratory, for their assistance with this work. The support and encouragement of C. V. Parks of the Oak Ridge National Laboratory and C. M. Hopper, formerly of the Oak Ridge National Laboratory is also appreciated. The support of U.S. Department of Energy Nuclear Criticality Safety Program is greatly appreciated.

6.6.1 INTRODUCTION

SAMS is the **S**ensitivity **A**nalysis **M**odule for **S**CALE, which has been designed for automated operation with the TSUNAMI-1D, TSUNAMI-3D-K5 and TSUNAMI-3D-K6 SCALE control modules. In multigroup mode, SAMS utilizes data from forward and adjoint neutron transport analyses as well as the sensitivity of resonance self-shielded cross-section data to input data to compute sensitivity coefficients that represent the percentage effect on a system response, such as multiplication factor (k_{eff}) or reaction rate ratio, to a percentage change in the nuclear reaction probabilities or cross sections. Sensitivity coefficients computed with SAMS can be presented as an energy-integrated value, which represents a uniform change in a given cross section across all energies, or as a "profile" in which a value is calculated for each discrete energy group. When combined with the uncertainties in the cross-section data, the cross-section-covariance data, the uncertainty in k_{eff} due to uncertainties in the cross-section data can be assessed. The sensitivity data produced by SAMS are appropriate for use with the TSUNAMI-IP, TSURFER and TSAR functional modules of SCALE and may be plotted in the Fulcrum SCALE user interface.

The methodology for calculating sensitivity and uncertainty parameters in SAMS is an extension of that used in the Fantastic Oak Ridge Sensitivity System (FORSS) [SAMSLWM+81, SAMSWML+76, SAMSWOM+78]. The FORSS system was capable of calculating the sensitivity of the system k_{eff} to changes in group-wise cross-section data for any given isotope for a number of reaction types. This data could also be used in conjunction with cross-section covariance data to determine the uncertainty in the calculated k_{eff} due to uncertainties in the nuclear data library. The FORSS methodology requires the calculation of the forward and adjoint angular and scalar fluxes and flux moments. Once these flux solutions are determined, the relative change in the system response due to a change in a component cross section can be determined. These sensitivities are calculated for each material region, nuclide, reaction type, and energy group in the system model.

In multigroup transport solution mode, the generation of sensitivity coefficients with SAMS requires forward and adjoint criticality calculations to compute the system responses, required fluxes as well as the volume of each flux region. Additionally, the neutron cross-section data used in the transport solution and the sensitivity of the resonance self-shielded cross-section data to the input material number densities must be available. SAMS currently functions with neutron transport solutions from XSDRNPM, NEWT, KENO V.a and KENO-VI. The sensitivities of the resonance self-shielded cross-section data to the input material number densities is computed by routines from SENLIB and BONAMIST. SAMS reads the required data files and produces sensitivity coefficients for many reactions for all nuclides for all regions in the problem. SAMS offers several output options to provide convenient transfer of the sensitivity data for plotting and additional analysis with other modules, such as TSUNAMI-IP, TSURFER, TSAR and Fulcrum.

As is described in the SCALE TSUNAMI-3D manual section, SAMS is also used in continuous energy mode transport solution sensitivity calculations. In these calculations, sensitivities are generated by TSUNAMI-3D and SAMS is used only to provide edits of sensitivity and uncertainty results. Sensitivity coefficients generated in continuous energy mode are stored in a multigroup sensitivity data file (SDF) format that is consistent with the various SCALE tools that use SDFs. The text provided in the rest of the SAMS manual chapter is oriented toward SAMS use with multigroup transport solutions.

6.6.2 THEORY

The methodology used by SAMS to produce sensitivity coefficients is presented in this section. The explicit portion of the sensitivity coefficient represents the sensitivity of the system responses to the problem-dependent resonance self-shielded multigroup cross-section data used in the analysis. The implicit portion of the complete sensitivity coefficient is the sensitivity of the resonance self-shielded multigroup cross-section data to the data input to the resonance self-shielding calculation. Summing the implicit and explicit contributions from a particular cross-section data component produces the complete sensitivity coefficient. Uncertainty analysis involves propagating the uncertainty information from the cross-section-covariance data file to the computed value of k_{eff} via the sensitivity coefficients. Problem characterization data and system average quantities are computed, such as the energy of average lethargy causing fission. Each of these quantities is computed by SAMS and is explained in this section. When executed as part of one of the TSUNAMI sequences, the SAMS input is generated automatically by the control module for that sequence.

6.6.2.1 Explicit sensitivity coefficient generation

This methodology used to generate the explicit portion of the sensitivity coefficients is identical to that used in the FORSS [SAMSWML+76] code system for fast reactor applications, with the addition of the sensitivity of k_{eff} to the fission energy spectrum (χ). The sensitivity coefficients produced with these techniques give the sensitivity of the computed k_{eff} to a particular component of the group-wise cross-section data.

The explicit sensitivity coefficients are calculated using the well-established adjoint-based perturbation theory approach [SAMSGan67, SAMSObl76, SAMSSJ72, SAMSUsa64]. The full derivation of the general procedure is not given here; however, the specific theory for the generation of k_{eff} sensitivities is presented below. For the full derivation of the general sensitivity equations the reader is referred to [SAMSWML+76].

The Boltzmann transport equation can be written in the form

$$[A - \lambda B]\phi = 0 \tag{6.6.1}$$

where

ϕ = neutron flux,

λ = represents the eigenvalues where the largest eigenvalue is $1/k_{\text{eff}}$,

A = operator that represents all of the transport equation except for the fission term,

B = operator that represents the fission term of the transport equation.

Defining perturbed transport operators and the perturbed eigenvalues as

$$\begin{aligned} A' &= A + \delta A \\ B' &= B + \delta B, \text{ and} \\ \lambda' &= \lambda + \delta\lambda \end{aligned} \tag{6.6.2}$$

where δA and δB represent small linear perturbations in their corresponding transport operators and $\delta\lambda$ represents the resulting change in the eigenvalues. The perturbed transport equation can be written in the form

$$[A' - \lambda' B']\phi' = 0 \tag{6.6.3}$$

The equation adjoint to Eq. (6.6.1) is

$$[A^\dagger - \lambda B^\dagger]\phi^\dagger = 0, \tag{6.6.4}$$

where $\phi^\dagger$ is the adjoint flux and has a special physical significance as the "importance" of the particles within the system, and $A^\dagger$ and $B^\dagger$ are the adjoint operators corresponding to A and B.

Multiplying Eq. (6.6.3) by $\phi^\dagger$, and integrating over all phase space yields

$$\langle \phi^\dagger (A - \lambda B + \delta A - \lambda\delta B - B\delta\lambda - \delta\lambda\delta B)\phi' \rangle = 0. \tag{6.6.5}$$

Using the property of adjointness (i.e., $\langle \phi^\dagger (A - \lambda B)\phi' \rangle = \langle \phi' (A^\dagger - \lambda^\dagger B^\dagger)\, \phi^\dagger \rangle$) and Eq. (6.6.4) to reduce the number of terms yields

$$\left\langle \phi^\dagger (A - \lambda B + \delta A - \lambda\delta B - B\delta\lambda - \delta\lambda\delta B)\phi' \right\rangle = 0 \tag{6.6.6}$$

Eq. (6.6.6) is further simplified by ignoring the second-order perturbation term ($\delta\lambda\delta$B) and substituting ϕ' with ϕ, indicating that the perturbations in the transport operators do not cause significant perturbations in the flux solution. The eigenvalue perturbation becomes

$$\frac{\delta\lambda}{\lambda} = \frac{\langle \phi^\dagger (\delta A - \lambda\delta B)\phi \rangle}{\langle \phi^\dagger (\lambda B)\phi \rangle}. \tag{6.6.7}$$

Substituting the perturbation terms with partial derivatives with respect to a macroscopic cross section, Σ, of the transport operator at some point in phase space $\vec{r}$, the relative sensitivity of λ, becomes

$$\frac{\delta\lambda}{\lambda} = \frac{\left\langle \phi^\dagger(\bar{\xi}) \left(\frac{\partial A[\Sigma(\bar{\xi})]}{\partial\Sigma(\vec{r})} - \lambda \frac{\partial B[\Sigma(\bar{\xi})]}{\partial\Sigma(\vec{r})} \right) \phi(\bar{\xi}) \right\rangle}{\left\langle \phi^\dagger(\bar{\xi}) \lambda B[\Sigma(\bar{\xi})]\phi(\bar{\xi}) \right\rangle} \tag{6.6.8}$$

where ξ is the phase space vector and the brackets indicate integration over space, direction and energy variables.

Note that since $\lambda = 1/k$, then $\partial\lambda/\lambda = -\partial k/k$, where $k = k_{eff}$, the sensitivity of k due to a small perturbation in a macroscopic cross section, Σ, of the transport operator at some point in phase space $\vec{r}$ can be expressed as

$$S_{k,\Sigma(r)} \equiv \frac{\Sigma(\vec{r})}{k} \frac{\partial k}{\partial\Sigma(\vec{r})} = -\frac{\Sigma(\vec{r})}{k} \frac{\left\langle \phi^\dagger(\bar{\xi}) \left(\frac{\partial A[\Sigma(\bar{\xi})]}{\partial\Sigma(\vec{r})} - \frac{1}{k} \frac{\partial B[\Sigma(\bar{\xi})]}{\partial\Sigma(\vec{r})} \right) \phi(\bar{\xi}) \right\rangle}{\left\langle \phi^\dagger(\bar{\xi}) \frac{1}{k^2} B[\Sigma(\bar{\xi})]\phi(\bar{\xi}) \right\rangle} \tag{6.6.9}$$

The k sensitivity for individual cross sections can be obtained from Eq. (6.6.9) using the discrete ordinates form of the transport equation. In doing so, the phase space vector, $\bar{\xi}$, has been replaced by indices representing discretization in space, energy and angle. It has been demonstrated in Ref. [SAMSWML+76] that sensitivity coefficients for reaction x, isotope i, energy group g, and computational region z can be represented as

$$S_{k,\Sigma^i_{x,g,z}} = \frac{T^i_{1,x,g,z} + T^i_{2,g,z} + T^i_{3,x,g,z}}{D}. \tag{6.6.10}$$

where the denominator, D, is expressed as

$$D = \frac{1}{k} \sum_{i=1}^{I} \sum_{z=1}^{R} V_z \sum_{g=1}^{G} \left(\bar{\nu}^i_{g,z} \Sigma^i_{f,g,z} \phi_{g,z} \right) \sum_{g'=1}^{G} \left(\chi^i_{g',z} \phi^\dagger_{g',z} \right) \tag{6.6.11}$$

where

$\chi^i_{g',z}$ = average fraction of fission neutrons emitted into energy group from fission of isotope i in region z,

$\bar{\nu}^i_{g,z}$ = average number of fission neutrons emitted from fission of isotope i in region z in energy group g,

$\Sigma^i_{f,g,z}$ = macroscopic cross section for fission of isotope i in region z and energy group g,

I = number of isotopes in the system model,

R = number of computational regions in the system model,

G = number of neutron energy groups in the system model.

Energy-integrated coefficients are obtained by summing the group-wise coefficients over all energy groups. The T terms of Eq. (6.6.10) represent the transport processes for neutron loss, fission production and scattering to the group of interest in T_1, T_2 and T_3, respectively. The first term is expressed as

$$T^i_{1,x,g,z} = -\Sigma^i_{x,g,z} V_z \sum_{j=0}^{NMOM} (2\ell + 1)\, \tilde{\phi}^{\dagger j}_{g,z} \tilde{\phi}^j_{g,z} \tag{6.6.12}$$

where

$\Sigma^i_{x,g,z}$ = macroscopic cross section for some reaction x, of isotope i, energy group g, in region z,

ℓ = Legendre order that corresponds to the j^{th} real valued flux moment,

$\tilde{\phi}^{\dagger j}_{g,z}$ = j^{th} component real valued adjoint flux moment for energy group g, and region z,

$NMOM$ = the total number of real valued flux moments corresponding to the desired Legendre order of expansion.

V_z = the volume of region z

The second and third terms can be expressed as

$$T^i_{2,g,z} = \frac{1}{k} V_z \bar{\nu}^i_{g,z} \Sigma^i_{f,g,z} \phi_{g,z} \sum_{g'=1}^{G} \phi^\dagger_{g',z} \chi_{g',z} \tag{6.6.13}$$

$$T^i_{3,x,g,z} = \sum_{j=0}^{NMOM} V_z \sum_{g'=1}^{G} \tilde{\phi}^{\dagger j}_{g',z} \tilde{\phi}^j_{g,z} \Sigma^{\ell,i}_{x,g' \to g,z} \tag{6.6.14}$$

where

$\Sigma^{\ell,i}_{x,g\to g',z}$ = ℓ^{th} moment of the transfer cross section for reaction x of isotope i, from energy group to energy group g in region z.

For specific reactions, not all of the T terms defined above are needed to calculate the sensitivity coefficient. The application of Eq. (6.6.10) for each type of reaction is outlined below. The computational form of each equation is expressed with the volume integrated product of the forward and adjoint flux moments as

$$P^{\ell}_{g,g',z} = V_z \sum_j \tilde{\phi}^{\dagger j}_{g',z} \tilde{\phi}^{j}_{g,z} \tag{6.6.15}$$

For KENO calculations without mesh fluxes, the flux product is computed with Eq. (6.6.12) , where z simply represents each material region. For KENO calculations using the mesh flux generation option, the flux product for each material region is computed as

$$P^{\ell}_{g,g',z} = \sum_j \sum_m \tilde{\phi}^{\dagger j}_{g',z_m} \tilde{\phi}^{j}_{g,z_m} V_{z_m} \tag{6.6.16}$$

where m represents flux meshes that occur in region z, and the subscript z_m denotes fluxes computed in mesh m of region z. Also, V_{z_m} is the volume of mesh m in region z.

Similarly, for transport calculations using XSDRNPM and NEWT, the flux product is computed over the computational mesh using Eq. (6.6.16) where m represents the computational mesh used in the transport calculation defined for each material region, z.

Once the flux products are computed for each material region or zone, the sensitivity coefficients for each reaction type can be computed as follows.

1. Capture Reaction Sensitivity (non-fission, non-scattering)

   Only the $T^{i}_{1,x,g,z}$ term is used for this class of reactions where $\Sigma^{i}_{x,g,z}$ is the absorption cross section of interest ((n, γ), (n, α), (n, p), etc.) and can be expressed as

$$S^{i}_{x,g,z} = \frac{-\Sigma^{i}_{x,g,z} \sum\limits_{\ell=0}^{ISCT} (2\ell+1) P^{\ell}_{g,g,z}}{D} \tag{6.6.17}$$

   where $ISCT$ = the highest Legendre order of scattering used in the sensitivity calculations.

2. Fission Reaction Sensitivity

   The fission reaction requires $T^{i}_{1,x,g,z}$ and $T^{i}_{2,g,z}$, where $\Sigma^{i}_{x,g,z}$ in the definition of $T^{i}_{1,x,g,z}$ is the fission cross section and can be expressed as

$$\begin{aligned} S^{i}_{f,g,z} = \frac{1}{D} \Bigg[& \left(\tfrac{1}{k} \bar{\nu}^{i}_{g,z} \Sigma^{i}_{f,g,z} \chi^{i}_{g,z} - \Sigma^{i}_{f,g,z} \right) P^{0}_{g,g,z} + \sum\nolimits_{\substack{g'=1 \\ g\neq g'}}^{G} \tfrac{1}{k} \bar{\nu}^{i}_{g,z} \Sigma^{i}_{f,g,z} \chi^{i}_{g',z} P^{0}_{g,g',z} \\ & - \sum\nolimits^{i}_{f,g,z} \sum\nolimits_{=1}^{ISCT} (2+1) P_{g,g,z} \Bigg] \end{aligned} \tag{6.6.18}$$

3. $\bar{\nu}$ Sensitivity

   The $\bar{\nu}$ reaction only requires $T^{i}_{2,g,z}$ and can be expressed as

$$S^{i}_{\bar{\nu},g,z} = \frac{\frac{1}{k} \sum\limits_{g'=1}^{G} \bar{\nu}^{i}_{g,z} \Sigma^{i}_{f,g,z} \chi^{i}_{g',z} P^{0}_{g,g',z}}{D}. \tag{6.6.19}$$

4. χ Sensitivity

The computation of the unconstrained χ reaction sensitivity only requires $T^i_{2,g,z}$, with the χ and $\nu\Sigma_f$ terms interchanged, and is expressed as

$$S^i_{\chi,g,z} = \frac{\frac{1}{k}\sum_{g'=1}^{G} \bar{\nu}^i_{g',z}\Sigma^i_{f,g',z}\chi^i_{g,z}P^0_{g',g,z}}{D} \tag{6.6.20}$$

In Eq. (6.6.20), the sensitivities coefficients sum to 1.0 when added over all energy groups and nuclides. However, since the fission spectrum probability distribution for any nuclide must, by definition, sum to 1.0 over all energy groups, the sensitivity of k_{eff} to the fission spectrum should sum to 0.0, as any change in fission spectrum in any group must be compensated by changes in other groups to maintain the constraint that all values sum to 1.0. This methodology was first developed for the SAGEP code [SAMSHTK84] and is implemented in SAMS as

$$\tilde{S}^i_{\chi,g,z} = S^i_{\chi,g,z} - \chi^i_{g,z}\sum_{g'=1}^{G} S^i_{\chi,g',z} \tag{6.6.21}$$

The constrained χ calculation from Eq. (6.6.21) is the default for SAMS.

5. Scattering Reaction Sensitivity

All scattering reactions (elastic, inelastic, and (n, 2n) reactions) require $T^i_{1,x,g,z}$ and $T^i_{3,x,g,z}$ where $\Sigma^i_{x,g,z}$ in the definition of $T^i_{1,x,g,z}$ is the scattering cross section and $\Sigma^{\ell,i}_{x,g'\to g,k}$ in the definition of $T^i_{3,x,g,z}$ is a component of the group-to-group scattering matrix for the ℓ^{th} scattering moment of reaction x.

$$\begin{aligned} S^i_{x,g,z} = \frac{1}{D}\Bigg(&\sum_{\ell=0}^{ISCT}\left(\Sigma^{\ell,i}_{x,g\to g,z} - (2\ell+1)\Sigma^i_{x,g,z}\right)P^\ell_{g,g,z} \\ &+ \sum_{\substack{g'=1\\ g'\neq g}}^{G}\Sigma^{\ell,i}_{x,g\to g',z}P^\ell_{g,g',z}\Bigg) \end{aligned} \tag{6.6.22}$$

6. Total Reaction Sensitivity

The total reaction requires $T^i_{1,x,g,z}$, $T^i_{2,g,z}$, and $T^i_{3,x,g,z}$. Here, $\Sigma^i_{x,g,z}$ in the definition of $T^i_{1,x,g,z}$ is the total cross section and $\Sigma^{\ell,i}_{x,g'\to g,k}$ in the definition of $T^i_{3,x,g,z}$ is a component of the group-to-group scattering matrix for the ℓ^{th} scattering moment. For non-fissionable isotopes, $T^i_{2,g,z}$ will be zero. The total reaction sensitivity coefficient can be expressed as

$$\begin{aligned} S^i_{t,g,z} = \frac{1}{D}\Bigg[&\left(\sum_{s,g\to g,z}^{0,i} + \frac{1}{k}\bar{\nu}^i_{g,z}\Sigma^i_{f,g,z}\chi^i_{g,z} - \Sigma^i_{t,g,z}\right)P^0_{g,g,z} \\ &+ \sum_{\substack{g'=1\\ g\neq g'}}^{G}\left(\sum_{s,g\to g',z}^{0,i} + \frac{1}{k}\bar{\nu}^i_{g,z}\Sigma^i_{f,g,z}\chi^i_{g',z}\right)P^0_{g,g',z} \\ &+ \sum_{=1}^{ISCT}\left(\sum_{s,g\to g,z}^{,i} -(2+1)\sum_{t,g,z}^{i}\right)P^l_{g,g,z} + \sum_{\substack{g'=1\\ g'\neq g}}^{G}\sum_{s,g\to g',z}P^l_{g,g',z} \end{aligned} \tag{6.6.23}$$

6.6.2.2 Implicit sensitivity coefficient generation

The methodology to calculate the sensitivity coefficients, as presented in the previous section, was developed for fast reactor applications in which the effect of resonance self-shielding in the multigroup cross-section data is minimal. To provide an accurate estimation of the sensitivity coefficients for systems in which resonance self-shielding is important, the sensitivity coefficients as computed in Eq. (6.6.10) require additional terms to account for the first-order implicit effect of perturbations in the material number densities or nuclear data upon the shielded group-wise macroscopic cross-section data [SAMSWBP01].

The sensitivity of the cross-section data to the input data in turn affects the k_{eff} sensitivities. The implicit portion of the sensitivity coefficient, the sensitivity of the group-wise data to the input quantities, is defined as

$$S_{\Sigma_{x,g},\omega_i} = \frac{\omega_i}{\Sigma_{x,g}} \frac{\partial \Sigma_{x,g}}{\partial \omega_i} \tag{6.6.24}$$

where ω_i is some input quantity. The ω_i term could represent the number density of a particular material, a certain nuclear data component or a physical dimension of a system. For the sensitivity coefficients produced by SAMS, which are the sensitivities of k_{eff} to the group-wise cross-section data, the effect on k_{eff} of perturbing one cross section that affects the resonance-shielded values of other cross sections must be computed. If ω_i is a certain cross-section data component for process y of nuclide j in energy group h expressed as $\Sigma^j_{y,h}$, which is sensitive to perturbations in process x in energy group g for nuclide i expressed as $\Sigma^i_{x,g}$, the complete sensitivity of k_{eff} due to perturbations of $\Sigma^i_{x,g}$ can be defined using the chain rule for derivatives as

$$\begin{aligned} \left(S_{k,\Sigma^i_{x,8}}\right)_{\text{complete}} &= \frac{\Sigma^i_{x,g}}{k}\frac{dk}{d\Sigma^i_{x,g}} = \frac{\Sigma^i_{x,g}}{k}\frac{\partial k}{\partial \Sigma^i_{x,g}} + \sum_j \sum_h \frac{\Sigma^j_{y,h}}{k}\frac{\partial k}{\partial \Sigma^j_{y,h}} \times \frac{\Sigma^i_{x,g}}{\Sigma^j_{y,h}}\frac{\partial \Sigma^j_{y,h}}{\partial \Sigma^i_{x,g}} \\ &= S_{k,\Sigma^i_{x,g}} + \sum_j \sum_h S_{k,\Sigma^j_{yh}} S_{\Sigma^j_{yh},\Sigma^i_{x,g}} \end{aligned}, \tag{6.6.25}$$

where the sensitivity coefficients with respect to k_{eff} are the explicit components as computed in Eq. (6.6.10), with the region subscript, z, omitted, and j and h are varied to include all processes that are influenced by the value of $\Sigma^i_{x,g}$.

Subsequent to the computation of the explicit portion of the sensitivity coefficients, data from SENLIB and BONAMIST are used to compute the implicit portion of the sensitivity coefficients. This implementation of the implicit sensitivity coefficients is more general than that presented in the example calculation of [SAMSWBP01], as it allows for the assessment of the implicit components for all reactions due to interaction with all nuclides. Because the sensitivity of a response to a material number density is equivalent to the sensitivity of the same response to the corresponding total macroscopic cross section, the computation of the implicit sensitivity coefficients can be based on the sensitivity to the input material number densities. The implicit effect of the number densities on k_{eff} must be accounted for from several sources. One source is the effect of the number densities input to BONAMIST on the shielded cross sections in the unresolved resonance region. For this case, the implicit sensitivity of k_{eff} to the total cross section of nuclide i is

$$\begin{aligned} &\left(S_{k,\Sigma^i_{T,g}}\right)_{\text{implicit}} = \sum_j \sum_y \sum_h \frac{\Sigma^j_{y,h}}{k}\frac{\partial}{\partial \Sigma^j_{y,h}} \times \frac{\Sigma^i_T}{\Sigma^j_{y,h}}\frac{\partial \Sigma^j_{y,h}}{\partial \Sigma^i_T} \times \frac{\Sigma^i_{T,g}}{\Sigma^i_T}\frac{\partial \Sigma^i_T}{\partial \Sigma^i_{T,g}} \\ &= \sum_j \sum_y \sum_h S_{k,\Sigma^j_{yh}} S_{\Sigma^j_{y,h},\Sigma^i_T} S_{\Sigma^i_T,\Sigma^i_{T,g}} = \sum_j \sum_y \sum_h S_{k,\Sigma^j_{yh}} S_{\Sigma^j_{yh},N^i} S_{\Sigma^i_T,\Sigma^i_{T,g}} \end{aligned} \tag{6.6.26}$$

where j and y are varied to include all processes that are sensitive to N^i, the number density of the i^{th} nuclide. Additionally, the energy group for the implicit sensitivity, g, is varied over all energies. The sensitivity of the total macroscopic cross section to the group-wise macroscopic total cross section, $S_{\Sigma^i_T,\Sigma^i_{T,g}}$, is simply 1.0.

For data computed by SENLIB and input to BONAMIST, an additional term is necessary to account for the sensitivity of the SENLIB parameter, denoted ω_m. The chain rule for derivatives can again be used to propagate this sensitivity to a k_{eff} sensitivity. The implicit sensitivity of k_{eff} to the input number densities, in this case, is

$$\begin{aligned}\left(S_{k,\Sigma^i_{T,8}}\right)_{\text{implicit}} &= \sum_m \sum_j \sum_y \sum_h \frac{\Sigma^j_{y,h}}{k}\frac{\partial k}{\partial \Sigma^j_{y,h}} \times \frac{\omega_m}{\Sigma^j_{y,h}}\frac{\partial \mathcal{S}^j_{y,h}}{\partial \omega_m} \times \frac{\Sigma^i_T}{\omega_m}\frac{\partial \omega_m}{\partial \Sigma^i_T} \times \frac{\Sigma^i_{T,g}}{\Sigma^i_T}\frac{\partial \Sigma^i_T}{\partial \Sigma^i_{T,g}} \\ &= \sum_m \sum_j \sum_y \sum_h S_{k,\Sigma^j_{yh}} S_{\Sigma^j_{yh},\omega_m} S_{\omega_m,\Sigma^i_T} S_{\Sigma^i_T,\Sigma^i_{T,g}} \\ &= \sum_m \sum_j \sum_y \sum_h S_{k,\Sigma^j_{y,h}} S_{\Sigma^j_{y,h},\omega_m} S_{\omega_m,N^i} S_{\Sigma^i_T,\Sigma^i_{T,8}}\end{aligned} \tag{6.6.27}$$

where m is varied to include all SENLIB computed parameters that use the material number densities in their calculation and are input to BONAMIST.

The calculation of the implicit sensitivity of a total cross-section component in the unresolved resonance region requires the sum of the implicit quantities computed in Eq. (6.6.26) and Eq. (6.6.27) with ω_m varied to include the Dancoff factor for each zone used in the BONAMIST calculation.

To compute the implicit portion of sensitivity coefficients for reactions, x, other than total, an additional term must be employed. With the implicit sensitivity of k_{eff} to the total cross section computed, the chain rule for derivatives is again applied to propagate the sensitivity of k_{eff} to the total cross section to the sensitivity of k_{eff} to some other process. This is accomplished using the sensitivity of the total cross section to the particular processes, computed from the unshielded cross-section data as

$$\left(S_{k,\Sigma^i_{x,g}}\right)_{implicit} = \left(\frac{\Sigma^i_{T,g}}{k}\frac{\partial k}{\partial \Sigma^i_{T,g}}\right)_{implicit} \times \left(\frac{\Sigma^i_{x,g}}{\Sigma^i_{T,g}}\frac{\partial \Sigma^i_{T,g}}{\partial \Sigma^i_{x,g}}\right) \tag{6.6.28}$$

The second term on the right hand side of Eq. (6.6.28) is computed analytically using data from the short master cross-section data library.

Complete sensitivity coefficient

With the implicit sensitivities properly computed, the complete sensitivity coefficient by group can be computed as the sum of the explicit and implicit terms as

$$\left(S_{k,\Sigma^i_{x,g}}\right)_{complete} = \left(S_{k,\Sigma^i_{x,g}}\right)_{explicit} + \left(S_{k,\Sigma^i_{x,g}}\right)_{implicit} \tag{6.6.29}$$

When a Monte Carlo transport solution is used to produce sensitivity coefficients, uncertainties in the forward and adjoint flux solutions and the value of k_{eff} are propagated to the final sensitivity results using standard error propagation techniques [SAMSBev69]. The forward and adjoint fluxes are treated as uncorrelated to each other. Also, the group-wise values of each flux solution are treated as uncorrelated. The flux moments within each group are treated as fully correlated. Although this method provides an adequate assessment of the statistical uncertainty in the sensitivity coefficients, a more robust technique may be implemented in subsequent revisions.

6.6.2.3 Summary of sensitivity coefficients calculated by SAMS

Sensitivity coefficients are calculated for the sensitivity of k_{eff} to the reactions listed in Table 6.6.1, if appropriate cross-section data is available. The identifier used in the SAMS output for each of these sensitivity types is also given. The MT of 0 assigned to scattering is arbitrary, as a sum of scattering reaction does not exist in the AMPX format.

Table 6.6.1: Sensitivity types computed by SAMS.

MT	Reaction	SAMS identifier
0	Sum of scattering	scatter
1	Total	total
2	Elastic scattering	elastic
4	Inelastic scattering	n,n'
16	n,2n	n,2n
18	Fission	fission
101	Neutron disappearance	capture
102	n,γ	n,gamma
103	n,p	n,p
104	n,d	n,d
105	n,t	n,t
106	n,^{3}He	n,he-3
107	n,α	n,alpha
452	$\bar{\nu}$	nubar
1018	χ	chi

6.6.2.4 Uncertainty analysis

Given uncertainty information for the cross sections for all nuclides and reaction processes that are important to the system of interest, it is possible to estimate the uncertainty in the calculated system multiplication factor due to these data uncertainties.

The nuclear data parameters are represented by the vector α, the elements of which are $(\alpha^i_{x,g})$, where i is varied over all isotopes, x is varied over all reactions for each isotope and g is varied over all energy groups. If M is the number of nuclide-reaction pairs $\times$ the number of energy groups (i.e., the number of elements in α), the symmetric $M \times M$ matrix containing the relative variances (diagonal elements) and relative covariances (off-diagonal elements) in the nuclear data is $C_{\alpha\alpha}$. The elements of $C_{\alpha\alpha}$ are

$$\left(C_{\alpha^i_{x,g}\alpha^j_{y,g'}}\right) = \frac{COV(\alpha^i_{x,g}, \alpha^j_{y,g'})}{\alpha^i_{x,g}\alpha^j_{y,g'}}, \tag{6.6.30}$$

where i and j are varied over all isotopes, x and y are varied over all reactions for each isotope and g and g' are varied over all energy groups. Additionally,

$$COV\left(\alpha^i_{x,g}, \alpha^j_{y,g'}\right) = \left\langle \delta\alpha^i_{x,g}\delta\alpha^j_{y,g'} \right\rangle, \tag{6.6.31}$$

where $\delta\alpha^i_{x,g}$ and $\delta\alpha^j_{y,g'}$ represent the difference between the values and expectation values of the nuclear data parameters and $\langle\rangle$ represents integration over the ranges of $\alpha^i_{x,g}$ and $\alpha^j_{y,g'}$ weighted with a probability density function. A rigorous definition of the cross-section-covariance data is given in [SAMSDun00]. SAMS simply reads the covariance data from a standard COVERX data file.

The vector of length M containing sensitivities of the calculated k_{eff} to the α parameters is represented by $\mathbf{S_k}$, where each element is

$$\left(S_{k,\alpha^i_{x,g}}\right) = \frac{\alpha^i_{x,g}}{k}\frac{\partial k}{\partial \alpha^i_{x,g}}, \tag{6.6.32}$$

For the purposes of SAMS uncertainty calculations, the $\alpha^i_{x,g}$ parameters are simply the group-wise cross section data. If a particular material is present in more than one material region, the sensitivity coefficients for all regions are summed prior to creating the $\mathbf{S_k}$ vector.

The variance for the system k_{eff} value is given as

$$\sigma_k^2 = \mathbf{S_k} \mathbf{C}_{\alpha\alpha} \mathbf{S}_\mathbf{k}^T, \tag{6.6.33}$$

where T indicates a transpose.

The covariance in k_{eff} due to the energy correlations of two particular processes can be assessed by examining a subset of the elements of $\mathrm{C}_{\alpha\alpha}$, where *i, j, x* and *y* are held constant. If G is the number of energy groups, the covariance data for a particular process is represented as the $G \times G$ matrix $\mathbf{C}_{\alpha^i_x \alpha^\mathbf{j}_\mathbf{y}}$ and the group-wise sensitivity vectors, of length G, for the processes are represented as $\mathbf{S}_{\mathbf{k},\alpha^\mathbf{i}_\mathbf{x}}$ and $\mathbf{S}_{\mathbf{k},\alpha^\mathbf{j}_\mathbf{y}}$. The relative covariance in k_{eff} due to the particular process or processes is given as

$$\sigma^2_{k^{i,j}_{x,y}} = \mathbf{S}_{\mathbf{k},\alpha^\mathbf{i}_\mathbf{x}} \mathbf{C}_{\alpha^i_x \alpha^\mathbf{j}_\mathbf{y}} \mathbf{S}^T_{\mathbf{k},\alpha^\mathbf{j}_\mathbf{y}} \ . \tag{6.6.34}$$

In actuality, the COVERX data file represents the covariance data in the form of multiple $\mathbf{C}_{\alpha^i_x \alpha^\mathbf{j}_\mathbf{y}}$ matrices. Thus, although commonly used for its mathematical convenience, $\mathbf{C}_{\alpha\alpha}$ does not exist as a continuous matrix. In the COVERX format, if $\mathbf{C}_{\alpha^i_x \alpha^\mathbf{j}_\mathbf{y}}$ is present on the data file with $i \neq j$ and/or $x \neq y$, then the transpose matrix $\mathbf{C}_{\alpha^\mathbf{j}_\mathbf{y} \alpha^i_x}$ is not present. Thus, using each matrix on the COVERX file only once, an upper (or lower) triangular $\mathbf{C}_{\alpha\alpha}$ matrix could be constructed, but not a full matrix.

In SAMS, the value of σ_k^2 is calculated by first determining the values of the variances or covariance as in Eq. (6.6.35) for all processes in the system under consideration, excluding the total reaction. The total reaction is excluded because it is the sum of the other processes and its inclusion would increase the variance from its actual value. The value of is then computed as the sum of the variances (diagonal elements of $\mathbf{C}_{\alpha\alpha}$ plus twice the sum of the covariances (off-diagonal elements of $\mathbf{C}_{\alpha\alpha}$. The standard deviation of k_{eff} is simply the square root of σ_k^2.

6.6.2.5 Problem characterization

SAMS computes a number of characteristic parameters of fissile systems for the convenience of the user for multigroup sensitivity calculations. These characterization parameters are not calculated in continuous-energy mode. These include the median fission group, average fission group, average energy causing fission, and energy of average lethargy causing fission. Corresponding parameters are also computed for capture and scattering.

The median fission group is computed as the first group M that satisfies the inequality

$$\frac{\sum_{z=1}^{Z} \sum_{g=1}^{M} \phi_{g,z} \Sigma_{f,g,z}}{\sum_{z=1}^{Z} \sum_{g=1}^{G} \phi_{g,z} \Sigma_{f,g,z}} \geq \frac{1}{2} \ , \tag{6.6.35}$$

where

z = index for material regions or zones,

Z = total number of regions or zones in the system model,

G = total number of energy groups.

The average fission group (AFG) is computed as

$$AFG = \frac{\sum_{z=1}^{Z}\sum_{g=1}^{G} g\phi_{g,z}\Sigma_{f,g,z}}{\sum_{z=1}^{Z}\sum_{g=1}^{G} \phi_{g,z}\Sigma_{f,g,z}} . \tag{6.6.36}$$

The average energy causing fission (AECF) is computed as

$$AECF = \frac{\sum_{z=1}^{Z}\sum_{g=1}^{G} \bar{E}_g\phi_{g,z}\Sigma_{f,g,z}}{\sum_{z=1}^{Z}\sum_{g=1}^{G} \phi_{g,z}\Sigma_{f,g,z}} , \tag{6.6.37}$$

where

$\bar{E}_g$ = the average energy of group g, computed as $(E_g + E_{g+1})/2$.

Lethargy is defined as $u = \ln(E_0/E^*)$, where E_0 is the maximum energy considered, which is assumed for the lethargy calculation to be 10 MeV. The average lethargy causing fission is computed as

$$\bar{u} = \frac{\sum_{z=1}^{Z}\sum_{g=1}^{G} \bar{u}_g\phi_{g,z}\Sigma_{f,g,z}}{\sum_{z=1}^{Z}\sum_{g=1}^{G} \phi_{g,z}\Sigma_{f,g,z}} , \tag{6.6.38}$$

where

$\bar{u}_g$ = the average lethargy of group g, computed as $(u_g+u_{g+1})/2$.

The energy of the average lethargy causing fission (EALF) is

$$E_{\bar{u}} = \frac{E_0}{e^{\bar{u}}} = \frac{E_0}{exp\left(\frac{\sum_{z=1}^{Z}\sum_{g=1}^{G} \bar{u}_g\phi_{g,z}\Sigma_{f,g,z}}{\sum_{z=1}^{Z}\sum_{g=1}^{G} \phi_{g,z}\Sigma_{f,g,z}}\right)} . \tag{6.6.39}$$

The above definitions are applicable to determining corresponding parameters for capture and scattering reactions where the capture and 1-D scattering cross-section data are substituted for the fission cross section data.

Generalized perturbation theory

General Responses for Critical Systems

The previous sections have presented perturbation theory expressions for the explicit and implicit sensitivity coefficients relating changes in cross section data to changes in the fundamental eigenvalue of the neutron transport equation in Eq. (6.6.1). Generalized perturbation theory (GPT) can be used to extend the deterministic S/U methods in TSUNAMI-1D to address more general responses that depend on the solution of the neutron transport equation. There is currently no TSUNAMI-3D GPT capability based on MG KENO Monte Carlo calculations, but a CE capability exists, as described in the TSUNAMI-3D section of the SCALE manual. Here the basic expressions used in GPT computations are described. More details and background material on GPT can be found in [SAMSWil86].

The neutron transport equation is homogeneous because the fundamental eigenvalue effectively forces the multiplying medium to satisfy a pseudo-critical condition by scaling the fission production term to exactly balance the neutron losses. Since the solution of a homogeneous equation can only be found within an arbitrary normalization factor, the responses of interest in a critical system are generally ratios of linear functionals of the neutron flux, such as

$$R = \frac{\langle H_N \phi \rangle}{\langle H_D \phi \rangle}, \tag{6.6.40}$$

where H_N and H_D are response functions defining the response of interest. The value for a ratio response is independent of the flux normalization since it cancels from the numerator and denominator; thus the response is uniquely defined. The following are some examples of ratio responses:

(a) reaction rate ratios such as the conversion factor:

$$\begin{aligned} R &= \frac{\left\langle \Sigma_c^{(fertile)} \phi \right\rangle}{\left\langle \Sigma_a^{(fissile)} \phi \right\rangle}, \\ \Rightarrow H_N &= \Sigma_c^{(fertile)} \\ \Rightarrow H_D &= \Sigma_a^{(fissile)} \end{aligned} \tag{6.6.41}$$

(b) relative power density in fuel region V_z (e.g., fuel pin):

$$\begin{aligned} R &= \frac{\left\langle \frac{h(V_z)}{V_z} \Sigma_f \phi \right\rangle}{\left\langle \frac{1}{V_{tot}} \Sigma_f \phi \right\rangle}, \\ \Rightarrow H_N &= \frac{h(V_z)}{V_z} \Sigma_f \\ \Rightarrow H_D &= \frac{1}{V_{tot}} \Sigma_f, \\ h(V_z) &= 1, r \in V_z \\ h(V_z) &= 0, r \notin V_z \end{aligned} \tag{6.6.42}$$

(c) fast-to-thermal flux ratio:

$$\begin{aligned} R &= \frac{\langle [1 - h(E < E_{th})] \phi \rangle}{\langle (h(E < E_{th})) \phi \rangle}, \\ \Rightarrow H_N &= [1 - h(E > E_{th})] \\ \Rightarrow H_D &= h(E < E_{th}) \\ \Rightarrow E_{th} &\equiv \text{ thermal energy boundary.} \\ h(E < E_{th}) &= 1, \quad E < E_{th} \\ h(E < E_{th}) &= 0, E > E_{th} \end{aligned} \tag{6.6.43}$$

(d) collapsed/homogenized thermal absorption cross section:

$$\begin{aligned} R &= \frac{\langle h(E < E_{th}) \Sigma_a \phi \rangle}{\langle h(E < E_{th}) \phi \rangle}, \\ \Rightarrow H_N &= \Sigma_a(E) h(E < E_{th}) \\ \Rightarrow H_N &= h(E < E_{th}) \end{aligned} \tag{6.6.44}$$

Generalized Perturbation Theory for Response Ratios.

GPT is used to relate variations in input data to changes in the calculated ratio-response. Let α be an arbitrary input parameter (number density, multigroup cross section, nubar, chi, etc.) that affects the response ratio because the response functions H_N and/or H_D depend on α, or because the parameter appears in the transport equation, so that the flux is a function of α; i.e.,

$$R = R\left[H_N(\alpha), H_D(\alpha), \phi(\alpha)\right] \tag{6.6.45}$$

The response perturbation due to a sufficiently small variation in α can be approximated by the first-order term in a functional Taylor series expansion,

$$\delta R \;\cong\; \left\langle \left(\frac{\partial R}{\partial H_N}\frac{\partial H_N}{\partial \alpha} \;+\; \frac{\partial R}{\partial H_D}\frac{\partial H_D}{\partial \alpha} \;+\; \frac{\partial R}{\partial \phi}\frac{d\phi}{d\alpha}\right)\; \delta\alpha\left(\vec{\xi}\right)\right\rangle . \tag{6.6.46}$$

In GPT it is common to express the response perturbation as the sum of a "direct" and an "indirect" component. Eq. (6.6.46) is rearranged slightly to obtain:

$$\delta R = \quad \delta R^{(direct)} \;+\; \delta R^{(indirect)}, \tag{6.6.47}$$

where

$$\delta R^{(direct)} \;=\; \left\langle \left(\frac{\partial R}{\partial H_N}\frac{\partial H_N}{\partial \alpha} \;+\; \frac{\partial R}{\partial H_D}\frac{\partial H_D}{\partial \alpha} \right)\, \delta\alpha \right\rangle \;, \tag{6.6.48}$$

$$\delta R^{(indirect)} \;=\; \left\langle \left(\frac{\partial R}{\partial \phi}\frac{d\phi}{d\alpha}\right)\; \delta\alpha \right\rangle . \tag{6.6.49}$$

The direct component accounts for response changes due to perturbations in the response functions H_N and/or H_D that depend explicitly (or implicitly through self-shielding) on data parameter α. The second term, called the indirect component, accounts for the response change due to a flux perturbation in the numerator or denominator of Eq. (6.6.40), which is caused by varying α in the transport equation.

If the response functions depend explicitly on α, then the functional derivatives appearing in the direct effect component can be evaluated to give

$$\delta R^{(direct)} \equiv \frac{R}{\langle H_N\, \phi\rangle} \left\langle \frac{\partial H_N}{\partial \alpha}\, \delta\alpha \right\rangle \;-\; \frac{R}{\langle H_D\, \phi\rangle} \left\langle \frac{\partial H_D}{\partial \alpha}\, \delta\alpha \right\rangle . \tag{6.6.50}$$

In order to evaluate the indirect effect it is necessary to determine the impact of the data perturbation on the neutron balance equation. The operators A and B, as well as the flux ϕ and eigenvalue λ may all be functions of α. Taking the first-order expansion for each of these functions and neglecting higher order terms gives

$$\left[A - \lambda B\right] \left(\frac{d\phi}{d\alpha}\, \delta\alpha \right) \;=\; -\left[\left(\frac{\partial A}{\partial \alpha} - \lambda \frac{\partial B}{\partial \alpha} \right) \delta\alpha \right] \phi \;+\; \delta\lambda\; B\phi \,. \tag{6.6.51}$$

An equation called the generalized adjoint equation is introduced as follows

$$\left[A^{\dagger} - \lambda B^{\dagger}\right]\, \Gamma^{*} \;=\; \frac{1}{R}\, \frac{\partial R}{\partial \phi} \;=\; Q^{*} \,, \tag{6.6.52}$$

where Γ^{*} is known as the generalized adjoint, or generalized importance, function. The adjoint source term in Eq. (6.6.52) is equal to the functional derivative of the response ratio with respect to the neutron flux, which corresponds to

$$\frac{1}{R}\frac{\partial R}{\partial \phi} = \frac{H_N}{\langle H_N \phi\rangle} - \frac{H_D}{\langle H_D \phi\rangle} \tag{6.6.53}$$

Taking the inner product of Γ^* with Eq. (6.6.51), and the inner product of $\left(\frac{d\phi}{d\alpha}\delta\alpha\right)$ with eq:*eq6-3-53*, and then applying the property of adjoint operators and rearranging, gives:

$$\left\{\left\langle \left(\frac{\partial R}{\partial \phi}\frac{d\phi}{d\alpha}\right)\ \delta\alpha\right\rangle = -\left\langle \Gamma^*\ \left(\frac{\partial A}{\partial \alpha} - \lambda\frac{\partial B}{\partial \alpha}\right)\ \phi\ \delta\alpha\right\rangle\ +\ \delta\lambda\,\langle\Gamma^* B\phi\rangle\ . \tag{6.6.54}$$

The left side of Eq. (6.6.54) is identical to the indirect component defined in Eq. (6.6.49). As shown in the next section, the generalized adjoint function is calculated such that the term $\langle\Gamma^* B\phi\rangle$ is equal to zero; thus

$$\delta R^{(\text{indirect})} = -\left\langle \Gamma^*\left(\frac{\partial A}{\partial \alpha} - \lambda\frac{\partial B}{\partial \alpha}\right)\phi\delta\alpha\right\rangle. \tag{6.6.55}$$

Combining Eq. (6.6.50) and Eq. (6.6.55) gives the response perturbation due to both direct and indirect components:

$$\delta R \equiv \left\{\frac{R}{\langle H_N\phi\rangle}\left\langle\frac{\partial H_N}{\partial\alpha}\delta\alpha\right\rangle - \frac{R}{\langle H_D\phi\rangle}\left\langle\frac{\partial H_D}{\partial\alpha}\delta\alpha\right\rangle\right\} - \left\langle\Gamma^*\left(\frac{\partial A}{\partial\alpha} - \lambda\frac{\partial B}{\partial\alpha}\right)\phi\delta\alpha\right\rangle \tag{6.6.56}$$

Calculation of the Generalized Adjoint Function

The forward flux ϕ and adjoint function $\phi^*$are fundamental eigenfunctions of the homogeneous equations in Eq. (6.6.1) and Eq. (6.6.3), respectively. In contrast, the generalized adjoint function Γ^* obeys an inhomogeneous equation. Furthermore, the operator $\left(A^\dagger - \lambda B^\dagger\right)$ in Eq. (6.6.52) is singular, since by definition λ is an eigenvalue. Taking the inner production of ϕ with Eq. (6.6.52) and applying the property of adjoint operators gives

$$\langle\phi Q^*\rangle = \left\langle\phi\frac{1}{R}\frac{\partial R}{\partial\phi}\right\rangle = 0. \tag{6.6.57}$$

The above relation is necessary for the singular inhomogeneous equation to have a solution; i.e., the adjoint source must be orthogonal to the forward flux. It is easily shown that the terms in Eq. (6.6.53) satisfy this condition. However a solution to Eq. (6.6.52) is not unique—the general solution for Γ^* is the sum of a particular and a homogeneous solution. If Γ^*_{p} is any particular solution, then the function $\left(\Gamma^*_{\text{p}} + a\,\phi^\dagger\right)$ is also a solution, where "a" is any constant. This can be proved by direct substitution into Eq. (6.6.52). For GPT calculations it is convenient to define an auxiliary condition that "normalizes" the generalized fission source $B^\dagger\,\Gamma^*$ to contain no fundamental mode; i.e.,

$$\left\langle\phi B^\dagger\,\Gamma^*\right\rangle = \langle\Gamma^* B\phi\,\rangle\ \ =\ \ 0\,. \tag{6.6.58}$$

The above relation is satisfied by defining the constant to be $\text{a}\ =\ \frac{\left\langle\phi\ B^\dagger\,\Gamma^*_{\text{p}}\right\rangle}{\left\langle\phi\ B^\dagger\,\phi^\dagger\right\rangle}$, so that:

$$\Gamma^* \rightarrow\ \ \Gamma^*_{\text{p}}\ -\ \ \frac{\left\langle\phi\,B^\dagger\,\Gamma^*_{\text{p}}\right\rangle}{\langle\phi B^\dagger\,\phi^\dagger\,\rangle}\ \phi^\dagger\,. \tag{6.6.59}$$

The normalization in Eq. (6.6.58) is done for two reasons. First, the 2nd term on the right side of Eq. (6.6.54) vanishes, so that the indirect effect is stationary with respect to eigenvalue perturbations introduced by $\delta\alpha$. The second reason is due to numerical difficulties in converging the generalized fission source. During outer iterations the fundamental mode component multiplies like a resonance frequency compared to other harmonics. This can lead to loss of numerical significance in the general solution. Theoretically, if the initial outer iteration has no fundamental mode in $B^\dagger\,\Gamma^*$, then subsequent outer iterations also will not. However in practice numerical approximations and incomplete convergence of inner iterations tend to re-introduce the fundamental mode. Therefore the operation in Eq. (6.6.59) is applied after each outer iteration to remove fundamental mode "contamination."

Analogous to the eigenvalue sensitivities, the response sensitivity coefficient at phase space coordinate $\vec{\xi}$ is defined as the relative change in R due to a relative change in $\alpha(\vec{\xi})$:

$$S_{R,\alpha}(\vec{\xi}) \equiv \frac{\delta R/R}{\delta\alpha/\alpha}, \tag{6.6.60}$$

The total response perturbation is obtained by summing over the response perturbations due to arbitrary α-variations throughout phase space

$$\frac{\delta R}{R} \cong \left\langle S_{R,\alpha}(\vec{\xi})\frac{\delta\alpha}{\alpha}(\vec{\xi})\right\rangle \tag{6.6.61}$$

Comparing Eq. (6.6.46) and Eq. (6.6.56) shows that the sensitivity coefficient is equal to

$$\begin{aligned} &S_{R,\alpha}(\vec{\xi}) = \\ &\underbrace{\left\{\frac{1}{\langle H_N\,\phi\rangle}\frac{\partial H_N}{\partial\alpha} - \frac{1}{\langle H_D\,\phi\rangle}\frac{\partial H_D}{\partial\alpha}\right\}}_{S^{(direct)}} - \underbrace{\frac{1}{R}\;\Gamma^*\left(\frac{\partial A}{\partial\alpha} - \lambda\frac{\partial B}{\partial\alpha}\right)\phi}_{S^{(indirect)}}. \end{aligned} \tag{6.6.62}$$

As in the case for eigenvalue sensitivity theory, the response perturbations introduced by $\delta\alpha$ may include explicit as well as implicit effects caused by changes in self-shielded cross sections $\tilde{\alpha}$ that are impacted by the value of parameter α. This can be treated by extending the partial derivative operator with respect to α to be a total derivative, so that

$$\frac{\partial}{\partial\alpha} \;\rightarrow\; \frac{d}{d\alpha} \;=\; \frac{\partial}{\partial\alpha} + \sum_{\tilde{\alpha}}\left(\frac{\partial\tilde{\alpha}}{\partial\alpha}\right)\frac{\partial}{\partial\tilde{\alpha}}. \tag{6.6.63}$$

When the above substitution is made into Eq. (6.6.62), the complete sensitivity coefficient is obtained

$$\begin{aligned} S^{(\text{complete})}_{R,\alpha} = &\left\{\frac{1}{\langle H_N\phi\rangle}\frac{\partial H_N}{\partial\alpha} - \frac{1}{\langle H_D\phi\rangle}\frac{\partial H_D}{\partial\alpha}\right\} + \left\{\sum_\alpha\left(\frac{1}{\langle H_N\phi\rangle}\frac{\partial H_N}{\partial\alpha} - \frac{1}{\langle H_D\phi\rangle}\frac{\partial H_D}{\partial\alpha}\right)\left(\frac{\partial\alpha}{\partial\alpha}\right)\right\} \\ &+ \left\{\frac{1}{R}\Gamma^*\left(\frac{\partial A}{\partial\alpha} - \lambda\frac{\partial B}{\partial\alpha}\right)\phi\right\} + \left\{\sum_\alpha\left(\left\{\frac{1}{R}\Gamma^*\left(\frac{\partial A}{\partial\alpha} - \lambda\frac{\partial B}{\partial\alpha}\right)\phi\right\}\right)\left(\frac{\partial\alpha}{\partial\alpha}\right)\right\} \end{aligned} \tag{6.6.64}$$

It can be seen that for GPT there are four distinct contributions to the complete sensitivity coefficient:

$$S^{(\text{complete})}_{R,\alpha} = S^{(\text{direct})}_{\text{explicit}} + S^{(\text{direct})}_{\text{implicit}} + S^{(\text{indirect})}_{\text{explicit}} + S^{(\text{indirect})}_{\text{implicit}} \tag{6.6.65}$$

The SAMS code evaluates sensitivity coefficients for GPT as well as eigenvalue perturbation theory. Direct sensitivity coefficients for GPT are easily determined since they depend only on the response functions and do not require an adjoint calculation. On the other hand, the expressions for indirect explicit sensitivities are very similar to those presented in Sect. 6.6.2.1 for eigenvalue responses. The detailed expressions in Eq. (6.6.10) through Eq. (6.6.23) for eigenvalue sensitivity coefficients with respect to each type of nuclear data are also valid for general responses, by replacing the fundamental mode adjoint function $\phi^\dagger$ with the generalized adjoint Γ^*, and setting the value of the denominator D equal to the response R. The treatment of implicit effects described in Section Sect. 6.6.2.2 for eigenvalue responses is also the same for general responses.

6.6.3 SAMS INPUT DESCRIPTION

The input to SAMS consists of a SCALE Analytical Sequence Specification Record, a single block of data that contains keyword input that controls the code execution, and a SCALE input termination END record. Very little input is required to obtain sensitivity and uncertainty results, given the availability of appropriate data files, as the default values are suitable for a basic analysis. The input data for SAMS are entered using the SCALE free-form reading routines. The input is not case sensitive, so either upper- or lowercase letters may be used. A maximum of 252 columns per line may be used for input.

6.6.3.1 Analytical sequence specification record

The analytical sequence specification begins in column 1 of the first line of the input file and must contain the following line.

=SAMS5 (for XSDRN or KENO V.a calculations)

=SAMS6 (for KENO-VI calculations)

This signals the SCALE driver to execute the appropriate SAMS functional module. Input data following the Analytical Sequence Specification Record and preceding the END record are passed to SAMS as input.

6.6.3.2 SAMS input data

SAMS input data consists of a single SCALE input data block, the INITIAL block. The input must begin with the keywords **READ INITIAL** and end with the keywords **END INITIAL**. The keyword entries summarized in Table 6.6.2 and Table 6.6.3 are allowed between the **READ INITIAL** and **END INITIAL** keywords. Keywords ending with '=' must be followed by the value to be assigned to the corresponding variable. Default values listed in Table 6.6.2 and Table 6.6.3 may be set or altered if SAMS is executed as part of a sequence, such as TSUNAMI-3D-K5 or TSUNAMI-3D-K6; in these cases, the sequence manual for that sequence should also be consulted for default variable values.

Table 6.6.2: SAMS input keywords

Keyword	Default value	Description
binsen	F	Produces SENPRO formatted binary sensitivity data file on unit 40
coverx=	56group-cov7.1	Name of covariance data file to use for uncertainty analysis
kenova	Kenova	Indicates whether KENO V.a or KENO-VI data files are present on logical unit number kunit and kunita
kenovi	kenova	Indicates whether KENO V.a or KENO-VI data files are present on logical unit number kunit and kunita
kunit=	35	Logical unit number of the binary data file from the forward calculation
kunita=	36	Logical unit number of the binary data file from the adjoint case
largeimp=	100.0	Value for the absolute value of implicit sensitivities, which if exceeded, will be reset to 0.0 and print a warning message.
makeimp	F	Flag to cause implicit sensitivity coefficients to be generated.
newt	Kenova	Indicates that NEWT data files are present on logical unit number kunit and kunita

continues on next page

Table 6.6.2 – continued from previous page

nocovar	T	Flag to cause uncertainty edit to be turned off (sets print_covar to F)
nohtml	F	Flag to cause HTML output to not be produced.
nomix	F	Flag to cause the sensitivities by mixture to be turned off
pmeshvol	F	Flag to print mesh volumes
pn=	3	Legendre order for moment calculations
pltp	F	Flag to cause a Javapeño 3D file containing $\phi\phi^*$ matrices to be written inside the .htmd directory of the output.
prtgeom	F	Flag to cause the sensitivities to be output for each geometry region
prtimp	F	Prints explicit sensitivities coefficients, implicit sensitivity coefficients and complete sensitivity coefficients
prtp	F	Flag to cause $\phi\phi^*$ matrices to be written to the output file.
prtvols	F	Flag to cause the volumes of the regions to be printed by SAMS
useang	F	Flag to cause the angular flux calculated in KENO to be used to calculate flux moments for sensitivity calculations. If angular fluxes were not computed by KENO, *useang* is set to false internally.
usemom	T	Flag to cause the flux moments calculated with KENO to be used to in sensitivity coefficient generation. If flux moments were not computed by KENO, *usemom* is set to false internally.
xsdrn	kenova	Indicates that XSDRN data files are present on kunit and kunita logical unit numbers
xunit=	4	Logical unit number of the AMPX working formatted cross-section library
xunitm=	42	Logical unit number for master sensitivity library
shortx=	11	Logical unit number of the AMPX short master formatted cross-section library.
unconstrained-chi	F	Flag to generate pre-SCALE 6 unconstrained chi (fission spectrum) sensitivities
sensitivity_format=	txt	Specifies desired format for the resulting sensitivity data file (SDF). May be 'txt' for text-based format or 'hdf5' for HDF5-based format.

Due to differences in the continuous-energy and multigroup sensitivity coefficient calculation methods, SAMS input files that use continuous-energy TSUNAMI do not use most of the input parameters that are described in Table 6.6.2. In fact, continuous-energy TSUNAMI calculations will only use the nomix, unconstrainedchi, and sensitivity_format input parameters; all other parameters that are input will be ignored.

Table 6.6.3: SAMS input keywords for default covariance data.

Keyword	**Default value**	**Description**
use_dcov	F	Use default covariance data

continues on next page

Table 6.6.3 – continued from previous page

use_icov	F	Use user-input covariance data
cov_fix	F	Correct covariance data if the uncertainty is large >1000% or zero
large_cov	10.0	Relative Standard deviation to apply cov_fix
return_work_cov	F	Create a new covariance data file with only the cross section covariance data used in the analysis
udcov=	0.05	User-defined default value of standard deviation for all energy groups
udcov_corr=	1.0	User-defined default correlation value
udcov_corr_type=	zone	User-defined default correlation type; allowed values are long, zone, and short
udcov_therm=	0.0	User-defined default value of standard deviation for thermal energy groups
udcov_inter=	0.0	User-defined default value of standard deviation for intermediate energy groups
udcov_fast=	0.0	User-defined default value of standard deviation for fast energy groups

Additionally, user-defined covariance data can be specified for individual nuclides and reactions using the *COVARIANCE* data block. This data begins with the keywords **READ COVARIANCE** and ends with the keywords **END COVARIANCE**. Any of the optional *COVARIANCE* input data may be entered in free form format between the **READ COVARIANCE** and **END COVARIANCE** keywords. The specifications for the *COVARIANCE* data block are described in the "User Input Covariance Data" section of the TSUNAMI Utility Modules chapter of the SCALE manual.

As the SAMS module generates HTML output, the optional *HTML* data block will provide user control over some formats of the output. This data begins with the keywords **READ HTML** and ends with the keywords **END HTML**. Any of the optional *HTML* input data may be entered in free form format between the **READ HTML** and **END HTML** keywords. The specifications for the *HTML* data block are described in the *HTML Data* section of the TSUNAMI Utility Modules chapter of the SCALE manual.

6.6.3.3 Data files required by SAMS

SAMS requires a number of data files to be available in the directory in which in the program executes to calculate MG sensitivities. These files are summarized in Table 6.6.4, but are not needed for CE sensitivity calculations.

Table 6.6.4: Data files required by SAMS.

Filename	When required	Description
ft*kunit*f001	Always	Data from forward transport solution
ft*kunita*f001	Always	Data from adjoint transport solution
ft*xunit*f001	Always	AMPX working formatted cross-section data library containing resonance self-shielded cross-section data used in transport calculations

continues on next page

Table 6.6.4 – continued from previous page

ft*xunitm*f001	Always	Master sensitivity cross-section data library; this library contains resonance self-shielded data for more reactions than are available on the working formatted library
ft*xunitsx*f001	Implicit Sensitivities	Short master cross-section data library containing infinitely dilute cross-section data
senlib.sen	Implicit Sensitivities	Sensitivity data file produced by SENLIB; contains sensitivity of values input to resonance self-shielding codes to input material number densities
bonamist.sen	Implicit Sensitivities	Sensitivity data file produced by BONAMIST; contains the sensitivity of shielded group-wise cross sections to data input to BONAMIST
coverx	Uncertainty Analysis	COVERX formatted cross-section-covariance data file

6.6.3.4 Example input

An example input file is given in Example 6.6.1. In this example, XSDRNPM was used as the transport solver (*xsdrn*), the data file from the forward transport solution is read from file ft33f001 (*kunit=33*), and the data file from the adjoint transport solution is read from file ft32f001 (*kunita=32*). SAMS will generate implicit sensitivity coefficients (*makeimp*) and print them in detail (*prtimp*). SAMS will compute and print sensitivity coefficients by region (*prtgeom*) and print the fluxes (*prtflx*).

Example 6.6.1: Example SAMS input.

```
=sams5
read initial
xsdrn
kunit=33 kunita=32
makeimp prtimp prtgeom
end initial
end
```

6.6.4 SAMPLE PROBLEMS AND OUTPUT DESCRIPTION

6.6.4.1 Example problem using XSDRNPM

This sample problem is the Flattop-25 metal system from the Cross-Section Evaluation Working Group benchmark specifications [SAMSNationalNDCenter74]. The system consists of a 6.116-cm sphere of 93%-enriched uranium with a natural uranium reflector. The outer radius of the reflector is 24.13 cm.

Prior to running SAMS, the data files listed in Table 6.6.4 were generated for this system. It is recommended that users perform sensitivity analyses on 1-D systems using the TSUNAMI-1D control module, which executes the appropriate codes to generate the required data files in an automated manner from a single input. SAMS is executed automatically as a part of the TSUNAMI-1D control sequence. However, given appropriate data files SAMS can be run as stand-alone program.

Input and text output

The SAMS input is shown in Example 6.6.2 and the output is shown in Example 6.6.3. Each section of the output is described below. Note that output shown in this section is intended only to represent the format of output that the user will encounter. The actual computed results may vary.

1. Parameter Table and Summary of Transport Calculation

In the output listing, the SAMS Parameter Table gives the values of the SAMS input data. Next, the title from the XSDRNPM input file is given and a brief summary of the transport solution is given.

2. Energy-, Region-, and Mixture Integrated Sensitivity Coefficients

Next the energy-, region-, and mixture integrated sensitivity coefficients are given for each reaction of each nuclide. The nuclide symbol and reaction name are given. The data given in the Sensitivity column represents the expected change in k_{eff} of this system due to a uniform fractional change in the cross section of the given nuclide and reaction in all energy groups for all regions and mixtures that contain this nuclide. For example, a uniform fractional change of 0.01 (or 1%) in the fission cross section of ^{235}U would produce a fractional change in k_{eff} of $5.7367 \cdot 10^{-1} \times 0.01$ or $5.7367 \cdot 10^{-3}$ (0.57367% $\Delta k/k$). Because the keyword PRTIMP was entered in the SAMS input, the explicit, implicit and complete sensitivity coefficients are listed in separate columns. If PRTIMP were not entered, only the complete sensitivity would be given in the Sensitivity column.

3. Energy- and Region-Integrated Sensitivity Coefficients

Next the energy- and region-integrated sensitivity coefficients are given for each reaction of each nuclide. The mixture number, nuclide symbol and reaction name are given. The data given in the Sensitivity column represents the expected change in k_{eff} of this system due to a uniform fractional change in the cross section of the given nuclide and reaction in all energy groups for all regions that contain the particular mixture. For example, a uniform fractional change of 0.01 (or 1%) in the fission cross section of ^{235}U in mixture 1 would produce a fractional change in k_{eff} of $5.6594 \cdot 10^{-1} \times 0.01$ or $5.6594 \cdot 10^{-3}$ (0.56594% $\Delta k/k$). Because the keyword PRTIMP was entered in the SAMS input, the explicit, implicit and complete sensitivity coefficients are listed in separate columns. If PRTIMP were not entered, only the complete sensitivity would be given in the Sensitivity column.

4. Total Sensitivity Coefficients by Nuclide

The next section of the output summarizes the total sensitivity by nuclide and mixture. The sensitivity values given here are the same as the values given in the previous section for the sensitivity of k_{eff} to the total cross section. Also included in this edit are the atom densities for each nuclide in each mixture.

5. Total Sensitivity Coefficients by Mixture

The next output edit gives the total sensitivity of k_{eff} to the mixture. Here, the sensitivity of k_{eff} to the total cross section is summed over each nuclide in a given mixture.

6. Problem Characterization (MG Calculations Only)

The next output edit is for the problem characterization data. This section includes the median group of neutrons causing fission, capture and scattering; average group for fission, capture and scattering; the average energy causing fission, capture and scattering; and the energy of the average lethargy causing fission, capture and scattering.

7. Sensitivity Coefficients by Material Zone

Because the keyword PRTGEOM was entered in the input, the next output edit gives the energy-integrated sensitivity coefficients for each material zone defined in the XSDRNPM criticality model. First, the zone number, material number and volume of the zone are given. Next, the sensitivity coefficients are given in the same format as for the region-integrated data. The edit is repeated for each material zone in the XSDRNPM criticality model.

8. Uncertainty Information

The next output edit contains the uncertainty information. First, the percent relative standard deviation in k_{eff} ($\Delta k_{\text{eff}}/k_{\text{eff}} \times 100\%$) due to cross-section-covariance data is given. Next, a listing of the uncertainty terms as produced by Eq. (6.6.58) are given. The nuclide-reaction to nuclide-reaction covariance matrix responsible for the uncertainty contribution is given followed by the contribution to the uncertainty in terms of % Δk/k. The total uncertainty can be computed from individual values by adding the square of the values with positive signs and subtracting the square of the values with negative signs, then taking the square root. The negative values are the result of anti-correlations in the cross-section-covariance data.

9. Execution Complete

The final edit produced simply states that the execution is complete and gives the elapsed time.

Example 6.6.2: SAMS input for Flattop-25 sample problem.

```
=sams5
read initial
xsdrn kunit= 31 kunita= 32 xunit=   4 xunitm=  42 makeimp prtgeom prtimp largeimp=   1.0000E+03
end initial
end
```

Example 6.6.3: SAMS output for Flattop-25 sample problem.

```
SAMS:  Sensitivity Analysis Module for SCALE

         SAMS Parameter Table
         ====================

                                      transport solution from:   XSDRNPM
                              forward transport solution on unit:   31
                              adjoint transport solution on unit:   32
                            working cross-section library on unit:   4
                 master sensitivity cross-section library on unit:   42
                       short master cross-section library on unit:   11
                           covariance data library read from file:   56groupcov7.1
   order of flux moments requested for sensitivity calculations:   5
                                    make implicit sensitivities:   true
                                  print implicit sensitivities:   true
                                print sensitivities by mixture:   true
                      print sensitivities for each region/zone:   true
                                    print uncertainty data:   true
                                        print mesh volumes:   false
                               output binary SENPRO data file:   false
                               implicit sensitivity threshold:   1000.0000
                                                 HTML output:   true
                             Chi sensitivities are constrained:   true
                              Fixup large/zero covariance data:   false
                                   Use Default Covariance Data:   false
                                 Use User Input Covariance Data:   false

         HTML Format Options
         ===================

                                        Background color:   PapayaWhip
                                    Major Headings color:   Maroon
                                      Sub-headings color:   Navy
                                        Plain text color:   Black
                                         Hyperlink color:   Navy
                                          Hypelink style:   none
                                     Visited link color:    Navy
```

(continues on next page)

(continued from previous page)

```
                                 Default Covariance color:   Blue
                               User Input Covariance color:   Red
              Corrected with default covariance data color:   RoyalBlue
            Corrected with user input covariance data color:   Green
```

```
                    tsunami-1d flattop
                    ==================

                Summary of Transport Solution
                -----------------------------


          number of neutron groups:   238
            number of gamma groups:   0
                 number of regions:   2
            maximum mixture number:   2
    number of mixing table entries:   5
quadrature order of angular fluxes:   32
     maximum order of flux moments:   5
       Forward Calculation: k-eff =    1.00533677
       Adjoint Calculation: k-eff =    1.00528886
```

```
-------------------------------------------------------------------------------
   Energy, Region and Mixture Integrated Sensitivity Coefficients for this Problem
-------------------------------------------------------------------------------


   Nuclide       Reaction        Explicit      Implicit     Sensitivity
 ------------ ---------------   -----------   -----------   -----------
  u-234        total             4.5108E-03    1.4614E-07    4.5110E-03
  u-234        scatter           6.6431E-04    1.2145E-07    6.6443E-04
  u-234        elastic           4.0086E-04    1.1594E-07    4.0097E-04
  u-234        n,n'              2.6201E-04    5.5685E-09    2.6202E-04
  u-234        n,2n              1.4354E-06   -5.7151E-11    1.4354E-06
  u-234        fission           4.3022E-03    1.6639E-08    4.3022E-03
  u-234        capture          -4.5566E-04    8.0468E-09   -4.5565E-04
  u-234        n,gamma          -4.5566E-04    8.0468E-09   -4.5565E-04
  u-234        nubar             6.5581E-03    0.0000E+00    6.5581E-03
  u-234        chi               1.7178E-10    0.0000E+00    1.7178E-10
  u-235        total             5.9204E-01   -4.7099E-06    5.9204E-01
  u-235        scatter           6.6931E-02   -3.6751E-06    6.6927E-02
  u-235        elastic           3.2362E-02   -3.6651E-06    3.2358E-02
  u-235        n,n'              3.3483E-02   -8.6100E-09    3.3483E-02
  u-235        n,2n              1.0842E-03   -1.4191E-09    1.0842E-03
  u-235        fission           5.7367E-01   -7.7390E-07    5.7367E-01
  u-235        capture          -4.8559E-02   -2.6087E-07   -4.8559E-02
  u-235        n,gamma          -4.8559E-02   -2.6087E-07   -4.8559E-02
  u-235        nubar             9.1500E-01    0.0000E+00    9.1500E-01
  u-235        chi              -2.3075E-08    0.0000E+00   -2.3075E-08
  u-238        total             2.1733E-01    6.2352E-05    2.1740E-01
  u-238        scatter           2.1045E-01    6.0318E-05    2.1051E-01
  u-238        elastic           1.4370E-01    5.9909E-05    1.4376E-01
  u-238        n,n'              6.5971E-02    3.8252E-07    6.5971E-02
  u-238        n,2n              7.6541E-04    2.6330E-08    7.6544E-04
  u-238        fission           5.5624E-02    4.9512E-08    5.5624E-02
  u-238        capture          -4.8736E-02    1.9847E-06   -4.8734E-02
  u-238        n,gamma          -4.8736E-02    1.9847E-06   -4.8734E-02
  u-238        nubar             7.8442E-02    0.0000E+00    7.8442E-02
  u-238        chi               7.5020E-10    0.0000E+00    7.5020E-10
```

```
-------------------------------------------------------------------------
   Energy and Region Integrated Sensitivity Coefficients for this Problem
-------------------------------------------------------------------------
```

(continues on next page)

(continued from previous page)

Mixture	Nuclide	Reaction	Explicit	Implicit	Sensitivity
1	u-234	total	4.5108E-03	1.4614E-07	4.5110E-03
1	u-234	scatter	6.6431E-04	1.2145E-07	6.6443E-04
1	u-234	elastic	4.0086E-04	1.1594E-07	4.0097E-04
1	u-234	n,n'	2.6201E-04	5.5685E-09	2.6202E-04
1	u-234	n,2n	1.4354E-06	-5.7151E-11	1.4354E-06
1	u-234	fission	4.3022E-03	1.6639E-08	4.3022E-03
1	u-234	capture	-4.5566E-04	8.0468E-09	-4.5565E-04
1	u-234	n,gamma	-4.5566E-04	8.0468E-09	-4.5565E-04
1	u-234	nubar	6.5581E-03	0.0000E+00	6.5581E-03
1	u-234	chi	1.7178E-10	0.0000E+00	1.7178E-10
1	u-235	total	5.8379E-01	2.3611E-06	5.8379E-01
1	u-235	scatter	6.5770E-02	1.8838E-06	6.5772E-02
1	u-235	elastic	3.1559E-02	1.8623E-06	3.1561E-02
1	u-235	n,n'	3.3131E-02	2.1512E-08	3.3131E-02
1	u-235	n,2n	1.0797E-03	-1.7622E-11	1.0797E-03
1	u-235	fission	5.6593E-01	3.5945E-07	5.6594E-01
1	u-235	capture	-4.7915E-02	1.1785E-07	-4.7914E-02
1	u-235	n,gamma	-4.7915E-02	1.1785E-07	-4.7914E-02
1	u-235	nubar	9.0395E-01	0.0000E+00	9.0395E-01
1	u-235	chi	-2.3539E-08	0.0000E+00	-2.3539E-08
1	u-238	total	8.0059E-03	3.2531E-09	8.0059E-03
1	u-238	scatter	4.6599E-03	3.1531E-09	4.6599E-03
1	u-238	elastic	2.3732E-03	3.0891E-09	2.3732E-03
1	u-238	n,n'	2.2134E-03	5.9442E-11	2.2134E-03
1	u-238	n,2n	7.2528E-05	4.5161E-12	7.2528E-05
1	u-238	fission	5.0328E-03	9.2930E-12	5.0328E-03
1	u-238	capture	-1.6868E-03	9.0770E-11	-1.6868E-03
1	u-238	n,gamma	-1.6868E-03	9.0770E-11	-1.6868E-03
1	u-238	nubar	7.6590E-03	0.0000E+00	7.6590E-03
1	u-238	chi	-4.8497E-10	0.0000E+00	-4.8497E-10
2	u-235	total	8.2517E-03	-7.0711E-06	8.2446E-03
2	u-235	scatter	1.1604E-03	-5.5590E-06	1.1548E-03
2	u-235	elastic	8.0337E-04	-5.5275E-06	7.9785E-04
2	u-235	n,n'	3.5255E-04	-3.0122E-08	3.5252E-04
2	u-235	n,2n	4.4336E-06	-1.4015E-09	4.4322E-06
2	u-235	fission	7.7357E-03	-1.1333E-06	7.7346E-03
2	u-235	capture	-6.4439E-04	-3.7872E-07	-6.4477E-04
2	u-235	n,gamma	-6.4439E-04	-3.7872E-07	-6.4477E-04
2	u-235	nubar	1.1046E-02	0.0000E+00	1.1046E-02
2	u-235	chi	4.6475E-10	0.0000E+00	4.6475E-10
2	u-238	total	2.0933E-01	6.2348E-05	2.0939E-01
2	u-238	scatter	2.0579E-01	6.0314E-05	2.0585E-01
2	u-238	elastic	1.4133E-01	5.9906E-05	1.4139E-01
2	u-238	n,n'	6.3758E-02	3.8246E-07	6.3758E-02
2	u-238	n,2n	6.9288E-04	2.6325E-08	6.9291E-04
2	u-238	fission	5.0591E-02	4.9503E-08	5.0591E-02
2	u-238	capture	-4.7050E-02	1.9846E-06	-4.7048E-02
2	u-238	n,gamma	-4.7050E-02	1.9846E-06	-4.7048E-02
2	u-238	nubar	7.0783E-02	0.0000E+00	7.0783E-02
2	u-238	chi	1.2352E-09	0.0000E+00	1.2352E-09

Total Sensitivity Coefficients by Nuclide

Mixture	Nuclide	Atom Density	Sensitivity
1	u-234	4.9000E-04	4.5110E-03
1	u-235	4.4490E-02	5.8379E-01
2	u-235	3.4000E-04	8.2446E-03
1	u-238	2.7000E-03	8.0059E-03

(continues on next page)

(continued from previous page)

```
     2    u-238          4.7740E-02    2.0939E-01

 ------------------------------------------------
    Total Sensitivity Coefficients by Mixture
 ------------------------------------------------

 Mixture   Sensitivity
 -------   -----------
     1     5.9631E-01
     2     2.1763E-01

 ------------------------------
    Problem Characterization
 ------------------------------

 median fission group is 25    9.000E+05 to 8.750E+05(eV)
                           average fission group   2.682E+01
              average energy(eV) causing fission   1.516E+06
  energy(eV) of average lethargy causing fission   7.402E+05

 median capture group is 42    2.000E+05 to 1.500E+05(eV)
                           average capture group   4.216E+01
              average energy(eV) causing capture   3.014E+05
  energy(eV) of average lethargy causing capture   1.498E+05

 median scatter group is 41    2.700E+05 to 2.000E+05(eV)
                           average scatter group   3.915E+01
              average energy(eV) causing scatter   4.743E+05
  energy(eV) of average lethargy causing scatter   2.231E+05
```

```
------------------------------------------------
   Sensitivity Coefficients by Material Zone
------------------------------------------------


Zone 1 Material 1 Volume of zone =    9.58277E+02 cm^3

Mixture    Nuclide       Reaction        Explicit      Implicit     Sensitivity
------- ------------- ---------------   -----------   -----------   -----------
    1     u-234        total             4.5108E-03    1.4614E-07    4.5110E-03
    1     u-234        scatter           6.6431E-04    1.2145E-07    6.6443E-04
    1     u-234        elastic           4.0086E-04    1.1594E-07    4.0097E-04
    1     u-234        n,n'              2.6201E-04    5.5685E-09    2.6202E-04
    1     u-234        n,2n              1.4354E-06   -5.7151E-11    1.4354E-06
    1     u-234        fission           4.3022E-03    1.6639E-08    4.3022E-03
    1     u-234        capture          -4.5566E-04    8.0468E-09   -4.5565E-04
    1     u-234        n,gamma          -4.5566E-04    8.0468E-09   -4.5565E-04
    1     u-234        nubar             6.5581E-03    0.0000E+00    6.5581E-03
    1     u-234        chi               1.7178E-10    0.0000E+00    1.7178E-10
    1     u-235        total             5.8379E-01    2.3611E-06    5.8379E-01
    1     u-235        scatter           6.5770E-02    1.8838E-06    6.5772E-02
    1     u-235        elastic           3.1559E-02    1.8623E-06    3.1561E-02
    1     u-235        n,n'              3.3131E-02    2.1512E-08    3.3131E-02
    1     u-235        n,2n              1.0797E-03   -1.7622E-11    1.0797E-03
    1     u-235        fission           5.6593E-01    3.5945E-07    5.6594E-01
    1     u-235        capture          -4.7915E-02    1.1785E-07   -4.7914E-02
    1     u-235        n,gamma          -4.7915E-02    1.1785E-07   -4.7914E-02
    1     u-235        nubar             9.0395E-01    0.0000E+00    9.0395E-01
    1     u-235        chi              -2.3539E-08    0.0000E+00   -2.3539E-08
    1     u-238        total             8.0059E-03    3.2531E-09    8.0059E-03
    1     u-238        scatter           4.6599E-03    3.1531E-09    4.6599E-03
    1     u-238        elastic           2.3732E-03    3.0891E-09    2.3732E-03
    1     u-238        n,n'              2.2134E-03    5.9442E-11    2.2134E-03
    1     u-238        n,2n              7.2528E-05    4.5161E-12    7.2528E-05
```

(continues on next page)

(continued from previous page)

```
    1     u-238        fission          5.0328E-03    9.2930E-12    5.0328E-03
    1     u-238        capture         -1.6868E-03    9.0770E-11   -1.6868E-03
    1     u-238        n,gamma         -1.6868E-03    9.0770E-11   -1.6868E-03
    1     u-238        nubar            7.6590E-03    0.0000E+00    7.6590E-03
    1     u-238        chi             -4.8497E-10    0.0000E+00   -4.8497E-10
```

```
Zone 2 Material 2 Volume of zone =    5.78936E+04 cm^3

 Mixture    Nuclide       Reaction        Explicit      Implicit     Sensitivity
 ------- ------------- ---------------   -----------   -----------   -----------
     2     u-235        total             8.2517E-03   -7.0711E-06    8.2446E-03
     2     u-235        scatter           1.1604E-03   -5.5590E-06    1.1548E-03
     2     u-235        elastic           8.0337E-04   -5.5275E-06    7.9785E-04
     2     u-235        n,n'              3.5255E-04   -3.0122E-08    3.5252E-04
     2     u-235        n,2n              4.4336E-06   -1.4015E-09    4.4322E-06
     2     u-235        fission           7.7357E-03   -1.1333E-06    7.7346E-03
     2     u-235        capture          -6.4439E-04   -3.7872E-07   -6.4477E-04
     2     u-235        n,gamma          -6.4439E-04   -3.7872E-07   -6.4477E-04
     2     u-235        nubar             1.1046E-02    0.0000E+00    1.1046E-02
     2     u-235        chi               4.6475E-10    0.0000E+00    4.6475E-10
     2     u-238        total             2.0933E-01    6.2348E-05    2.0939E-01
     2     u-238        scatter           2.0579E-01    6.0314E-05    2.0585E-01
     2     u-238        elastic           1.4133E-01    5.9906E-05    1.4139E-01
     2     u-238        n,n'              6.3758E-02    3.8246E-07    6.3758E-02
     2     u-238        n,2n              6.9288E-04    2.6325E-08    6.9291E-04
     2     u-238        fission           5.0591E-02    4.9503E-08    5.0591E-02
     2     u-238        capture          -4.7050E-02    1.9846E-06   -4.7048E-02
     2     u-238        n,gamma          -4.7050E-02    1.9846E-06   -4.7048E-02
     2     u-238        nubar             7.0783E-02    0.0000E+00    7.0783E-02
     2     u-238        chi               1.2352E-09    0.0000E+00    1.2352E-09


 Generating working covariance matrix ...


 Working covariance matrix created for future processing.
```

```
-----------------------------
   Uncertainty Information
-----------------------------


 the relative standard deviation of k-eff (% delta-k/k) due to cross-section covariance data is:

   1.2743 % delta-k/k

 contributions to uncertainty in k-eff (% delta-k/k) by individual energy covariance matrices:

                              covariance matrix
            nuclide-reaction        with        nuclide-reaction            % delta-k/k due to this matrix
     ------------------------------      ------------------------------  ----------------------------------
              u-235 n,gamma                       u-235 n,gamma                1.1253E+00
              u-238 elastic                       u-238 n,n'                  -7.6667E-01
              u-238 n,n'                          u-238 n,n'                   7.2823E-01
              u-238 elastic                       u-238 elastic                4.5961E-01
              u-235 n,n'                          u-235 n,n'                   2.7423E-01
              u-235 fission                       u-235 fission                2.3563E-01
              u-235 elastic                       u-235 n,n'                  -2.1997E-01
              u-235 elastic                       u-235 n,gamma                1.8368E-01
              u-235 chi                           u-235 chi                    1.8056E-01
              u-235 nubar                         u-235 nubar                  1.3559E-01
              u-234 fission                       u-234 fission                1.1526E-01
              u-235 elastic                       u-235 elastic                1.0107E-01
              u-238 nubar                         u-238 nubar                  9.1147E-02
```

(continues on next page)

(continued from previous page)

```
              u-238 n,gamma                     u-238 n,gamma                6.1302E-02
              u-235 elastic                     u-235 fission               -4.0658E-02
              u-238 chi                         u-238 chi                    3.7177E-02
              u-238 fission                     u-238 fission                2.9009E-02
              u-238 elastic                     u-238 n,gamma                1.9436E-02
              u-238 n,2n                        u-238 n,2n                   1.2044E-02
              u-235 n,2n                        u-235 n,2n                   9.4214E-03
              u-238 elastic                     u-238 fission               -7.8074E-03
              u-238 elastic                     u-238 n,2n                  -5.2898E-03
              u-234 n,gamma                     u-234 n,gamma                4.4184E-03
              u-234 n,n'                        u-234 n,n'                   3.9180E-03
              u-235 elastic                     u-235 n,2n                  -2.4536E-03
              u-234 elastic                     u-234 elastic                1.8845E-03
              u-235 fission                     u-235 n,gamma                1.6297E-04
              u-238 fission                     u-238 n,gamma                9.5794E-05
              u-234 n,2n                        u-234 n,2n                   4.1081E-05

Note: relative standard deviation in k-eff can be computed from
individual values by adding the square of the values with positive signs and
subtracting the square of the values with negative signs, then taking the square root


-----------------------------------
 SAMS execution complete
 Elapsed time: 0.02717 minutes
-----------------------------------
```

Sensitivity data file

The energy-dependent sensitivity data computed by SAMS is written to a sensitivity data file (SDF), which SCALE returns to the same directory as the output file with the extension ".sdf". The SDF is generated directly in KENO in CE TSUNAMI-3D calculations. When XSDRNPM is used for the transport solution, the data file is presented in the TSUNAMI-A format described in Appendix A. This data file is suitable for visualization with the Fulcrum SCALE user interface or for system comparison with the TSUNAMI-IP code, which is described in section 6.5. A sample section of the TSUNAMI-A formatted sensitivity data file is show in Example 6.6.4.

Example 6.6.4: Sample of sensitivity data file for Flattop-25 sample problem.

```
tsunami-1d flattop
       238 number of neutron groups
       130   number of sensitivity profiles          30 are region integrated
  1.005337    k-eff from the forward case
energy boundaries:
  2.000000E+07  1.733300E+07  1.568300E+07  1.455000E+07  1.384000E+07
  1.284000E+07  1.000000E+07  8.187300E+06  6.434000E+06  4.800000E+06
  4.304000E+06  3.000000E+06  2.479000E+06  2.354000E+06  1.850000E+06
  1.500000E+06  1.400000E+06  1.356000E+06  1.317000E+06  1.250000E+06
  1.200000E+06  1.100000E+06  1.010000E+06  9.200000E+05  9.000000E+05
  8.750000E+05  8.611000E+05  8.200000E+05  7.500000E+05  6.790000E+05
  6.700000E+05  6.000000E+05  5.730000E+05  5.500000E+05  4.995200E+05
  4.700000E+05  4.400000E+05  4.200000E+05  4.000000E+05  3.300000E+05
  2.700000E+05  2.000000E+05  1.500000E+05  1.283000E+05  1.000000E+05
  8.500000E+04  8.200000E+04  7.500000E+04  7.300000E+04  6.000000E+04
  5.200000E+04  5.000000E+04  4.500000E+04  3.000000E+04  2.500000E+04
  1.700000E+04  1.300000E+04  9.500000E+03  8.030000E+03  6.000000E+03
  3.900000E+03  3.740000E+03  3.000000E+03  2.580000E+03  2.290000E+03
  2.200000E+03  1.800000E+03  1.550000E+03  1.500000E+03  1.150000E+03
  9.500000E+02  6.830000E+02  6.700000E+02  5.500000E+02  3.050000E+02
  2.850000E+02  2.400000E+02  2.100000E+02  2.075000E+02  1.925000E+02
  1.860000E+02  1.220000E+02  1.190000E+02  1.150000E+02  1.080000E+02
```

(continues on next page)

(continued from previous page)

```
  1.000000E+02  9.000000E+01  8.200000E+01  8.000000E+01  7.600000E+01
  7.200000E+01  6.750000E+01  6.500000E+01  6.100000E+01  5.900000E+01
  5.340000E+01  5.200000E+01  5.060000E+01  4.920000E+01  4.830000E+01
  4.700000E+01  4.520000E+01  4.400000E+01  4.240000E+01  4.100000E+01
  3.960000E+01  3.910000E+01  3.800000E+01  3.700000E+01  3.550000E+01
  3.460000E+01  3.375000E+01  3.325000E+01  3.175000E+01  3.125000E+01
  3.000000E+01  2.750000E+01  2.500000E+01  2.250000E+01  2.100000E+01
  2.000000E+01  1.900000E+01  1.850000E+01  1.700000E+01  1.600000E+01
  1.509990E+01  1.440000E+01  1.375000E+01  1.290000E+01  1.190000E+01
  1.150000E+01  1.000000E+01  9.099990E+00  8.099990E+00  7.150000E+00
  7.000000E+00  6.750000E+00  6.500000E+00  6.250000E+00  6.000000E+00
  5.400000E+00  5.000000E+00  4.750000E+00  4.000000E+00  3.730000E+00
  3.500000E+00  3.150000E+00  3.049990E+00  3.000000E+00  2.969990E+00
  2.870000E+00  2.770000E+00  2.669990E+00  2.570000E+00  2.469990E+00
  2.379990E+00  2.299990E+00  2.209990E+00  2.120000E+00  2.000000E+00
  1.940000E+00  1.860000E+00  1.770000E+00  1.679990E+00  1.589990E+00
  1.500000E+00  1.450000E+00  1.400000E+00  1.349990E+00  1.299990E+00
  1.250000E+00  1.224990E+00  1.200000E+00  1.174990E+00  1.150000E+00
  1.139990E+00  1.129990E+00  1.120000E+00  1.110000E+00  1.099990E+00
  1.089990E+00  1.080000E+00  1.070000E+00  1.059990E+00  1.049990E+00
  1.040000E+00  1.030000E+00  1.020000E+00  1.009990E+00  1.000000E+00
  9.750000E-01  9.500000E-01  9.250000E-01  9.000000E-01  8.500000E-01
  8.000000E-01  7.500000E-01  7.000000E-01  6.500000E-01  6.250000E-01
  6.000000E-01  5.500000E-01  5.000000E-01  4.500000E-01  4.000000E-01
  3.750000E-01  3.500000E-01  3.250000E-01  3.000000E-01  2.750000E-01
  2.500000E-01  2.250000E-01  2.000000E-01  1.750000E-01  1.500000E-01
  1.250000E-01  1.000000E-01  9.000000E-02  8.000000E-02  7.000000E-02
  6.000000E-02  5.000000E-02  4.000000E-02  3.000000E-02  2.530000E-02
  1.000000E-02  7.500000E-03  5.000000E-03  4.000000E-03  3.000000E-03
  2.500000E-03  2.000000E-03  1.500000E-03  1.200000E-03  1.000000E-03
  7.500000E-04  5.000000E-04  1.000000E-04  1.000000E-05
u-234        total            92234      1      0  0.000000E+00
```

```
 4.510986E-03  4.757363E-03 -1.231885E-04
 5.814569E-10  9.742489E-08  2.063851E-07  2.319431E-07  6.176152E-07
 9.231530E-06  2.764173E-05  8.188761E-05  1.996819E-04  1.188430E-04
 5.678897E-04  3.723359E-04  1.044565E-04  5.127857E-04  4.478409E-04
 1.387182E-04  6.498478E-05  5.994327E-05  1.059632E-04  8.117370E-05
 1.721682E-04  1.626100E-04  1.709000E-04  4.117717E-05  5.369000E-05
 3.138744E-05  9.733480E-05  1.855348E-04  1.821296E-04  2.169121E-05
 1.616294E-04  5.783673E-05  4.866144E-05  1.037796E-04  4.962128E-05
 4.391558E-05  2.489447E-05  2.192546E-05  5.810353E-05  3.308191E-05
 1.757015E-05 -9.932299E-07 -5.762635E-06 -1.347860E-05 -1.218557E-05
-2.017047E-06 -6.224814E-06 -1.817214E-06 -1.336107E-05 -8.857124E-06
-2.534309E-06 -6.268505E-06 -1.942463E-05 -6.952287E-06 -9.270964E-06
-5.016695E-06 -3.539706E-06 -1.224633E-06 -1.550815E-06 -1.328389E-06
-1.087067E-07 -4.089939E-07 -1.942041E-07 -1.483071E-07 -3.014540E-08
-1.437365E-07 -9.007177E-08 -1.296454E-08 -1.098961E-07 -4.759211E-08
-3.958266E-08 -2.063948E-09 -1.005130E-08 -2.381676E-08 -1.047572E-09
-1.118473E-09  1.241990E-10 -5.366082E-11 -5.346411E-11 -2.923313E-09
-3.338868E-09 -1.493598E-11 -2.279208E-11 -7.199807E-10 -8.987974E-11
-2.189838E-10 -2.175781E-11 -9.396965E-12 -1.682207E-10  1.138659E-11
-4.801163E-11  4.277212E-14  6.418566E-11 -3.590361E-12  1.551486E-11
-7.022784E-12  4.754449E-13  1.262907E-12 -1.061301E-10 -3.115855E-12
-3.998677E-11  5.218320E-14  2.886770E-12  9.972963E-12  3.602206E-13
-3.544641E-14 -4.565340E-13 -3.906762E-13 -4.094176E-12  1.509958E-12
 9.896480E-13 -8.890050E-15 -9.980198E-12 -1.834534E-11 -6.660895E-11
-8.361293E-12 -2.314084E-12  7.910957E-12 -3.747421E-12 -4.674509E-13
-2.783726E-13  5.640714E-13 -1.263619E-12  1.876653E-12 -3.458782E-13
 3.964266E-13 -8.096029E-14 -3.014990E-13  3.357317E-13  7.684112E-13
 4.710800E-14  2.347336E-13 -8.513845E-13 -2.995314E-13 -5.691168E-15
-1.575343E-14  1.975820E-15  3.589577E-14 -3.283024E-14 -4.554053E-12
-1.031342E-11 -3.311645E-13 -1.711711E-12 -2.871135E-13 -2.395040E-14
-4.988314E-14 -8.021950E-15 -6.806139E-15 -6.283709E-15 -1.928744E-14
```

(continues on next page)

(continued from previous page)

```
-2.273676E-14 -2.624242E-14 -2.940419E-14 -2.932818E-14 -2.487096E-14
-2.055185E-14 -2.090153E-14 -1.568286E-14  4.044556E-15 -1.100004E-15
-1.006007E-14 -1.512058E-14 -1.528490E-14 -1.476455E-14 -1.377876E-14
-6.998115E-15 -6.414986E-15 -5.551053E-15 -4.272736E-15 -3.051582E-15
-1.120191E-15 -8.480302E-16 -6.478704E-16 -5.176351E-16 -6.748717E-17
-6.293938E-17 -6.058076E-17 -5.969218E-17 -6.119141E-17 -6.332008E-17
-6.634259E-17 -7.015929E-17 -7.458915E-17 -7.908430E-17 -8.370476E-17
-8.862749E-17 -9.090348E-17 -9.294229E-17 -1.020032E-16 -8.841811E-16
-9.690290E-16 -1.031417E-15 -1.062247E-15 -2.241879E-15 -2.303293E-15
-2.306673E-15 -2.165671E-15 -2.024708E-15 -6.316471E-16 -5.119917E-16
-1.643515E-15 -1.461638E-15 -1.228501E-15 -9.115838E-16 -9.803894E-17
-5.921025E-17 -3.726506E-17 -2.782963E-17 -2.087060E-17 -3.530473E-18
-8.656642E-19 -2.791854E-19 -8.745909E-20 -2.513338E-20 -6.505315E-21
-1.451612E-21 -7.438999E-23 -2.611310E-23 -8.538001E-24 -2.522094E-24
-6.532481E-25 -1.437914E-25 -2.575603E-26 -2.525173E-27 -8.675912E-28
-1.414196E-29 -2.457059E-30 -1.730775E-31 -5.537974E-32 -3.241248E-38
 0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00
 0.000000E+00  0.000000E+00  0.000000E+00
u-234        scatter          92234      0      0  0.000000E+00
 6.644297E-04  6.789372E-04 -7.253796E-06
 1.624068E-11  1.498993E-09  2.210061E-09  3.019234E-09  1.485854E-08
 4.147500E-07  4.994740E-07 -2.057688E-06  8.260277E-06  6.384048E-06
 3.416607E-05  2.314843E-05  6.702604E-06  3.465375E-05  3.799558E-05
 1.587156E-05  8.343846E-06  8.102940E-06  1.545759E-05  1.265593E-05
 2.843251E-05  2.820164E-05  2.933217E-05  6.597646E-06  8.259760E-06
 4.585853E-06  1.321025E-05  2.173617E-05  2.408941E-05  3.326117E-06
 2.761796E-05  1.187137E-05  1.046844E-05  2.436908E-05  1.536275E-05
 1.655217E-05  1.118884E-05  1.113633E-05  3.968209E-05  3.674090E-05
 3.947471E-05  2.495301E-05  8.168438E-06  8.267026E-06  2.511112E-06
 8.989297E-07  9.842643E-07  2.826410E-07  5.933744E-07  1.081451E-07
-1.158760E-07 -3.228441E-07 -1.829070E-06 -8.369636E-07 -6.930253E-07
-6.464461E-07 -3.126912E-07 -7.491714E-08 -1.256803E-07 -8.790425E-08
-6.682722E-09 -3.177474E-08 -1.358181E-08 -2.448430E-08 -7.405575E-10
-2.381531E-08 -1.677138E-08  1.087897E-09 -1.518392E-08 -8.140176E-09
-5.731765E-09 -2.551186E-10 -8.359771E-10 -1.978769E-09 -4.494942E-11
 1.914997E-10  2.502574E-10  3.854115E-11 -1.587687E-11 -3.741346E-10
```

```
-1.176212E-10 -1.434239E-11 -2.200907E-11 -5.662974E-12  1.412261E-11
 1.968906E-10 -1.946603E-11 -8.655245E-12 -2.396444E-12  1.215475E-11
-4.719829E-11  1.913950E-13  6.404752E-11 -3.486064E-12  1.559158E-11
-6.699422E-12  5.604126E-13  2.181300E-12 -2.223242E-12  4.962365E-14
-3.415680E-13  3.069277E-13  2.900106E-12  9.854989E-12  4.147955E-13
-2.949746E-14 -4.034699E-13 -3.260798E-13 -3.978481E-12  1.474672E-12
 9.657468E-13  7.884317E-15 -7.658717E-12  4.440999E-13 -6.924095E-14
-5.525669E-12 -2.119092E-12  3.596589E-12 -3.585189E-12 -4.470271E-13
-2.687707E-13  5.603641E-13 -1.206110E-12  1.865397E-12 -3.252544E-13
 4.068594E-13 -6.950363E-14 -2.712372E-13  3.388504E-13  7.461941E-13
 1.470861E-13  2.514678E-13 -7.298236E-13 -6.930753E-14 -1.983845E-15
-8.764347E-15  4.590198E-15  2.890316E-14  9.989641E-15 -2.343592E-13
-2.819649E-14  9.121820E-16  1.567449E-16  3.853913E-16 -9.749661E-16
 7.412544E-16  6.365694E-16 -2.435351E-16 -2.771577E-16 -1.120208E-15
-1.646584E-15  3.543834E-16  6.820250E-17 -5.691048E-16 -7.483003E-16
-9.871186E-16 -1.800469E-15 -2.429645E-15  5.703603E-15  1.945347E-15
-1.574540E-17 -5.370011E-16 -5.154802E-16 -5.423486E-16 -5.361593E-16
-2.856485E-16 -2.903410E-16 -2.902574E-16 -1.815125E-16 -4.625630E-17
-2.416642E-17 -2.314575E-17 -2.404208E-17 -1.934835E-17 -4.763757E-18
-4.636201E-18 -4.626402E-18 -4.616198E-18 -4.937270E-18 -5.223761E-18
-5.573777E-18 -5.975374E-18 -6.465877E-18 -6.933935E-18 -7.378606E-18
-7.813760E-18 -7.974083E-18 -7.988734E-18 -8.346500E-18 -2.236693E-17
-2.469848E-17 -2.698569E-17 -2.763953E-17 -5.769021E-17 -5.864681E-17
-5.662232E-17 -5.234773E-17 -4.781621E-17 -2.247383E-17 -2.022061E-17
-3.282721E-17 -2.644715E-17 -1.936636E-17 -8.471592E-18 -3.742548E-18
-1.898564E-18 -8.735266E-19 -4.796991E-19 -3.169119E-19 -1.568985E-19
-4.404778E-20 -1.393036E-20 -4.185272E-21 -1.127225E-21 -2.686736E-22
```

(continues on next page)

(continued from previous page)

```
-5.494400E-23 -2.394512E-24 -7.783942E-25 -2.344133E-25 -6.314507E-26
-1.460224E-26 -2.787798E-27 -4.105965E-28 -3.404075E-29 -2.367579E-30
-6.950740E-32 -2.676098E-33 -8.730089E-35 -3.622198E-35  0.000000E+00
 0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00
 0.000000E+00  0.000000E+00  0.000000E+00
```

Sample plots of the energy-dependent sensitivity profiles from the SDF are shown below. Fig. 6.6.1 and Fig. 6.6.2 depict the region-dependent sensitivity profiles for ^{235}U fission and ^{238}U capture, respectively, in zones 1 and 2 of the Flattop-25 sample problem, which are the core and the reflector regions, respectively.

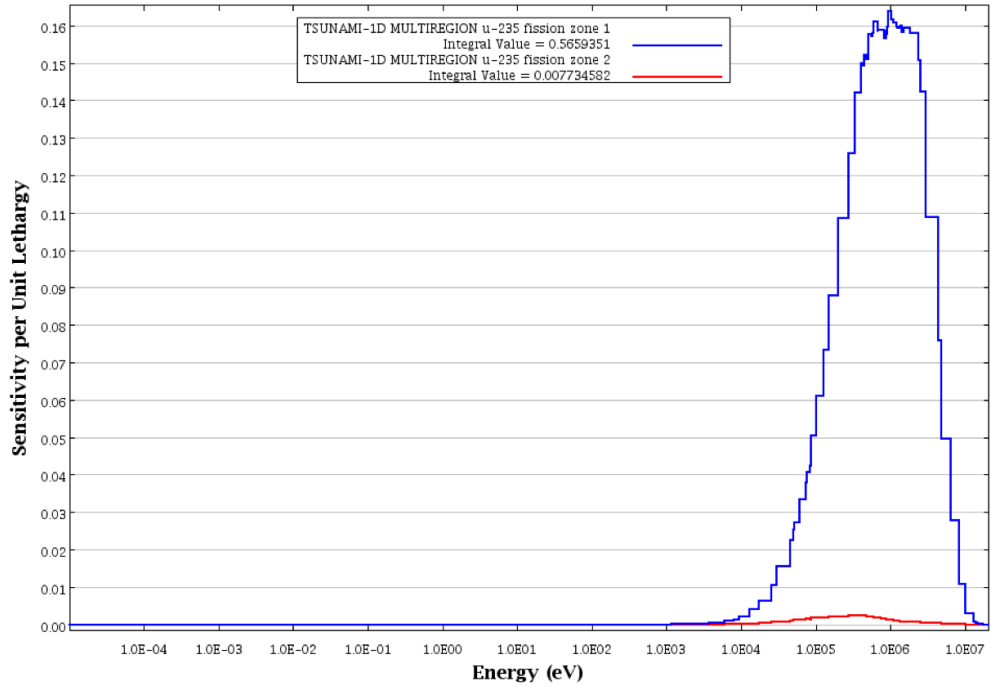


Fig. 6.6.1: Region-dependent fission sensitivity profiles for Flattop-25 sample problem.

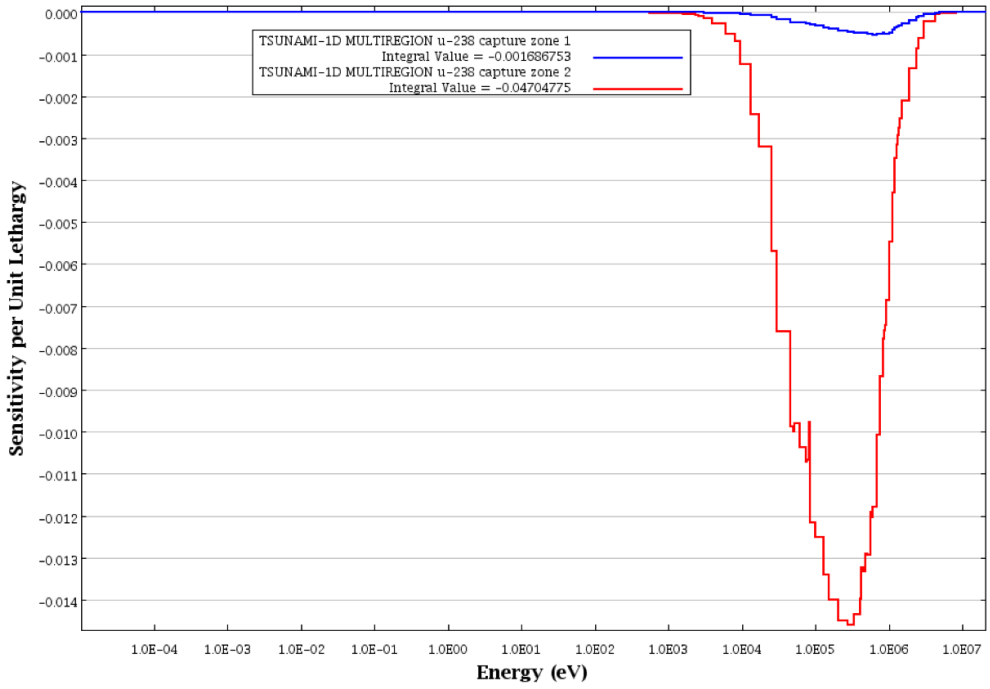


Fig. 6.6.2: Region-dependent capture sensitivity profiles for Flattop-25 sample problem.

HTML formatted output

SAMS will, by default, present the sensitivity results in HTML formatted output for interactive viewing in a web browser. SCALE will return an HTML file to the users' directory with the same name as the output file, but with the extension ".html". Opening this file in a web browser will allow the user to begin viewing the output. Other files necessary for the HTML output are stored in the ".htmd" directory associated with the output file and some utilities for data visualization are stored in the applet_resources directory, which is created in the same directory as the output file.

When the .html file is opened, an index page like the one in Fig. 6.6.3 will be shown. In this example, only one SCALE module with HTML output, SAMS, was executed from the input file. If more than one set of HTML output was available, additional modules would be listed on this page. To begin viewing the HTML output, click on the word *SAMS* in the middle of the page. Once SAMS is selected, the program verification information page is displayed, as shown in Fig. 6.6.4. The output data is sorted into four sections, three of which are shown on the left side of the window in this example. *General Information* consists of the *Program Verification Information* and the *Elapsed Time* for program execution. *Input Data* lists data input to the calculation including user input and data read from the required forward and adjoint transport calculations. *Results* contains the computed results, and an additional *Messages* section, not shown, would contain any error or warning messages generated during the SAMS calculation.

Clicking on the *Input Data* menu item reveals the menu of pages containing input data. Clicking on the *Input Parameters* item reveals the page shown in Fig. 6.6.5. Clicking on the *Summary of Transport Calculations* item reveals the page shown in Fig. 6.6.6.

Clicking on the *Results* menu item reveals the results available for viewing. Clicking on *Energy, Region and Mixture Integrated Sensitivity Coefficients for this Problem* under the results menu reveals the energy-, region- and mixture integrated sensitivity coefficients for this problem, as shown in Fig. 6.6.7. As with most of the tables in the HTML output, the table of sensitivity coefficients on this page can be sorted in ascending and

descending order by clicking on the heading of the column for which the sorting is desired. When clicking twice on the sensitivity column, the sensitivity results are sorted in descending order as shown in Fig. 6.6.8.

Skipping several output edits for brevity and selecting the *Sensitivity Data Plot* item will open a page with an applet version of Javapeño with the sensitivity data file preloaded as shown in Fig. 6.6.9. To create a plot of sensitivity data, double-click on the desired information on the right side of the window. Multiple items can be simultaneously selected in the list using standard platform specific modifier keys such as shift or control. Right-clicking on an item in the list will present a menu with various options for plotting. For more information on the use of Javapeño, please select Help from the Javapeño help menu.

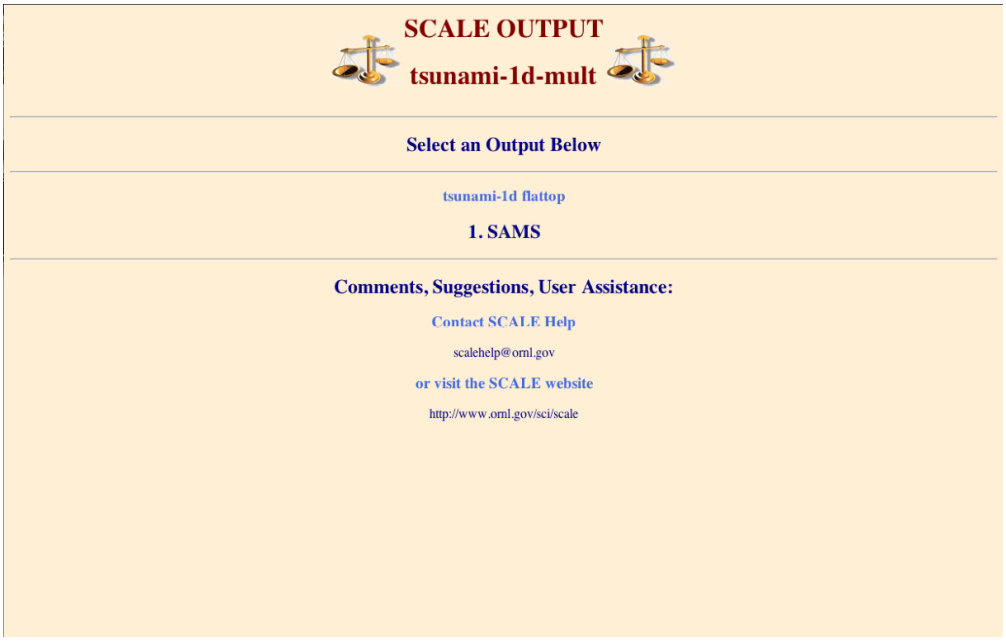


Fig. 6.6.3: Index page of SAMS HTML output.

Fig. 6.6.4: Program verification page of SAMS HTML output.

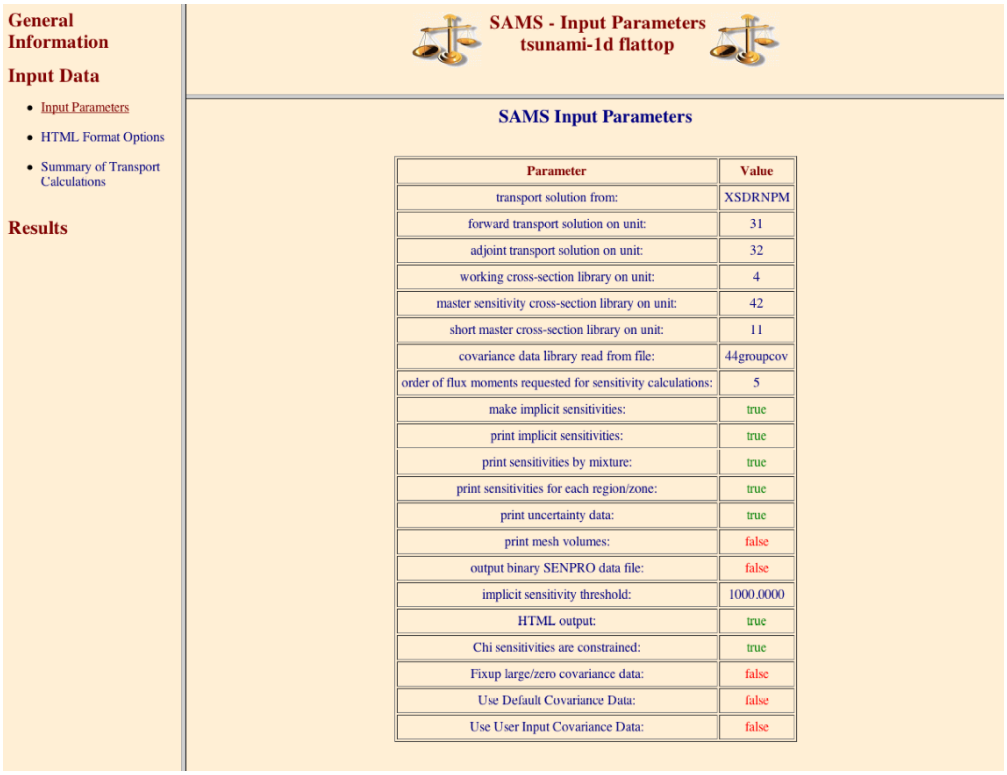


Fig. 6.6.5: Input parameters from SAMS HTML output.

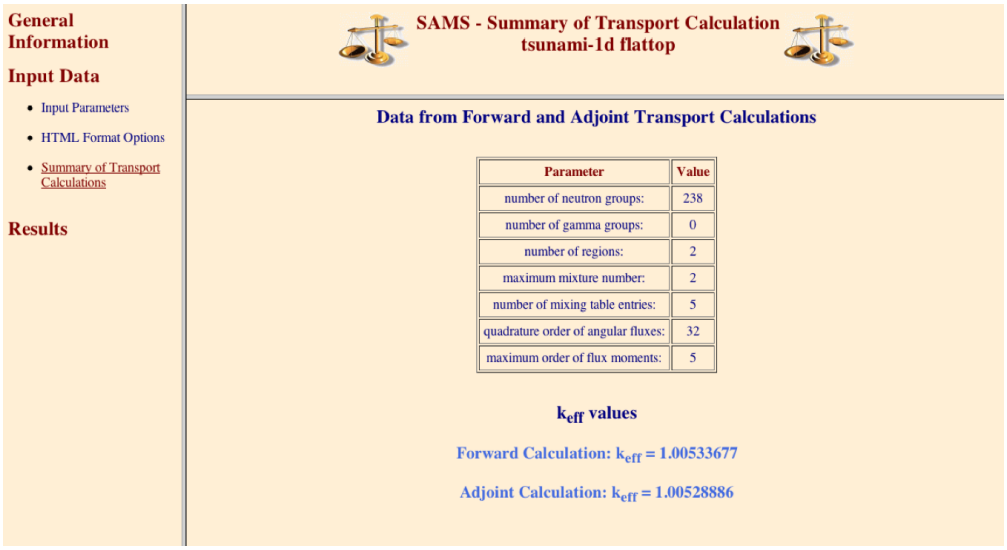
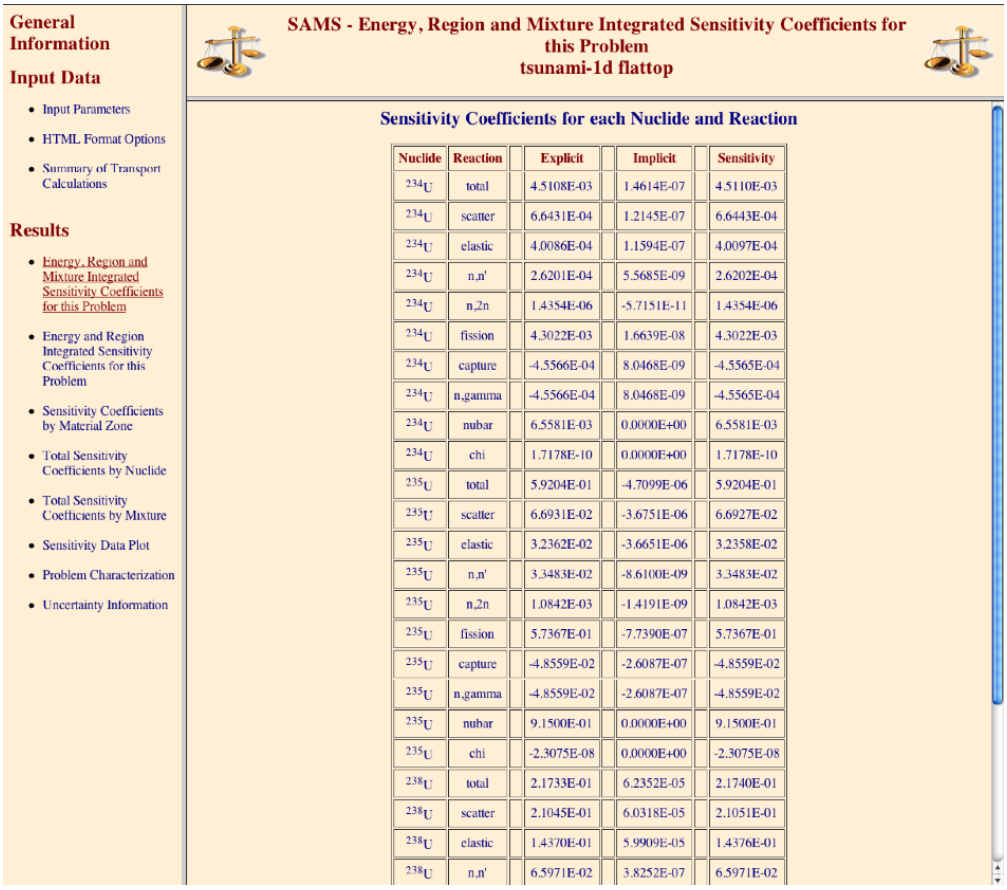
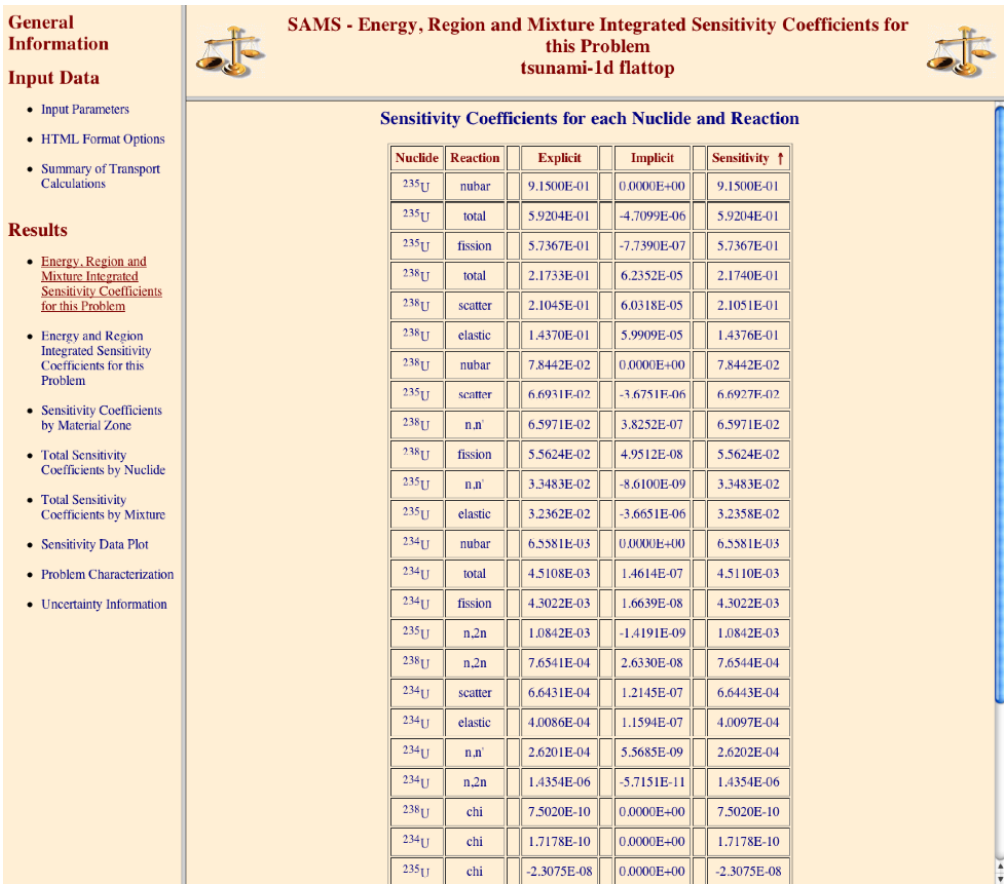


Fig. 6.6.6: Summary of transport calculations from SAMS HTML output.



Nuclide	Reaction	Explicit	Implicit	Sensitivity
^{234}U	total	4.5108E-03	1.4614E-07	4.5110E-03
^{234}U	scatter	6.6431E-04	1.2145E-07	6.6443E-04
^{234}U	elastic	4.0086E-04	1.1594E-07	4.0097E-04
^{234}U	n,n'	2.6201E-04	5.5685E-09	2.6202E-04
^{234}U	n,2n	1.4354E-06	-5.7151E-11	1.4354E-06
^{234}U	fission	4.3022E-03	1.6639E-08	4.3022E-03
^{234}U	capture	-4.5566E-04	8.0468E-09	-4.5565E-04
^{234}U	n,gamma	-4.5566E-04	8.0468E-09	-4.5565E-04
^{234}U	nubar	6.5581E-03	0.0000E+00	6.5581E-03
^{234}U	chi	1.7178E-10	0.0000E+00	1.7178E-10
^{235}U	total	5.9204E-01	-4.7099E-06	5.9204E-01
^{235}U	scatter	6.6931E-02	-3.6751E-06	6.6927E-02
^{235}U	elastic	3.2362E-02	-3.6651E-06	3.2358E-02
^{235}U	n,n'	3.3483E-02	-8.6100E-09	3.3483E-02
^{235}U	n,2n	1.0842E-03	-1.4191E-09	1.0842E-03
^{235}U	fission	5.7367E-01	-7.7390E-07	5.7367E-01
^{235}U	capture	-4.8559E-02	-2.6087E-07	-4.8559E-02
^{235}U	n,gamma	-4.8559E-02	-2.6087E-07	-4.8559E-02
^{235}U	nubar	9.1500E-01	0.0000E+00	9.1500E-01
^{235}U	chi	-2.3075E-08	0.0000E+00	-2.3075E-08
^{238}U	total	2.1733E-01	6.2352E-05	2.1740E-01
^{238}U	scatter	2.1045E-01	6.0318E-05	2.1051E-01
^{238}U	elastic	1.4370E-01	5.9909E-05	1.4376E-01
^{238}U	n,n'	6.5971E-02	3.8252E-07	6.5971E-02

Fig. 6.6.7: Energy- region- and mixture-integrated sensitivity coefficients from SAMS HTML output.



Nuclide	Reaction	Explicit	Implicit	Sensitivity ↑
^{235}U	nubar	9.1500E-01	0.0000E+00	9.1500E-01
^{235}U	total	5.9204E-01	-4.7099E-06	5.9204E-01
^{235}U	fission	5.7367E-01	-7.7390E-07	5.7367E-01
^{238}U	total	2.1733E-01	6.2352E-05	2.1740E-01
^{238}U	scatter	2.1045E-01	6.0318E-05	2.1051E-01
^{238}U	elastic	1.4370E-01	5.9909E-05	1.4376E-01
^{238}U	nubar	7.8442E-02	0.0000E+00	7.8442E-02
^{235}U	scatter	6.6931E-02	-3.6751E-06	6.6927E-02
^{238}U	n,n'	6.5971E-02	3.8252E-07	6.5971E-02
^{238}U	fission	5.5624E-02	4.9512E-08	5.5624E-02
^{235}U	n,n'	3.3483E-02	-8.6100E-09	3.3483E-02
^{235}U	elastic	3.2362E-02	-3.6651E-06	3.2358E-02
^{234}U	nubar	6.5581E-03	0.0000E+00	6.5581E-03
^{234}U	total	4.5108E-03	1.4614E-07	4.5110E-03
^{234}U	fission	4.3022E-03	1.6639E-08	4.3022E-03
^{235}U	n,2n	1.0842E-03	-1.4191E-09	1.0842E-03
^{238}U	n,2n	7.6541E-04	2.6330E-08	7.6544E-04
^{234}U	scatter	6.6431E-04	1.2145E-07	6.6443E-04
^{234}U	elastic	4.0086E-04	1.1594E-07	4.0097E-04
^{234}U	n,n'	2.6201E-04	5.5685E-09	2.6202E-04
^{234}U	n,2n	1.4354E-06	-5.7151E-11	1.4354E-06
^{238}U	chi	7.5020E-10	0.0000E+00	7.5020E-10
^{234}U	chi	1.7178E-10	0.0000E+00	1.7178E-10
^{235}U	chi	-2.3075E-08	0.0000E+00	-2.3075E-08

Fig. 6.6.8: Energy- region- and mixture-integrated sensitivity coefficients from SAMS HTML output sorted by sensitivity.

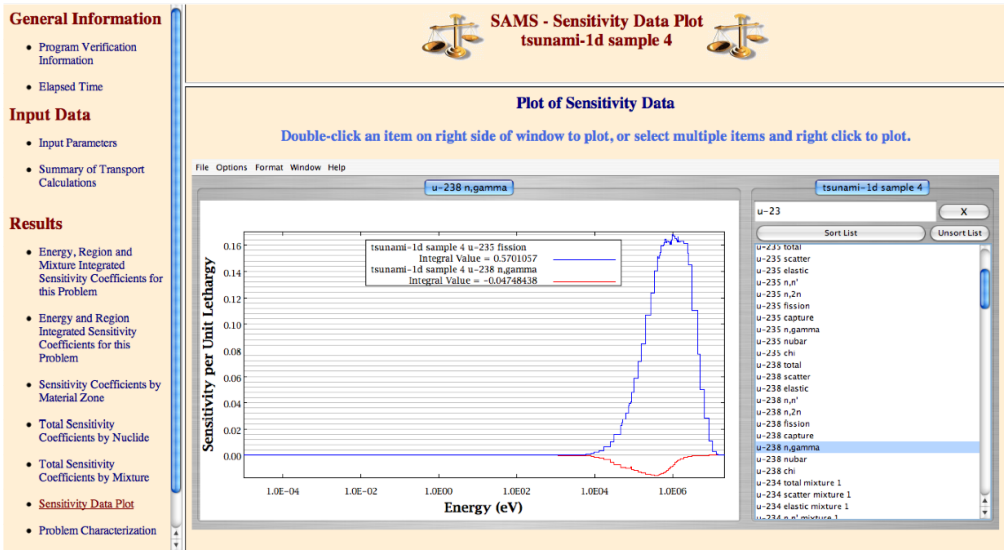


Fig. 6.6.9: Sensitivity data plot from SAMS HTML output.

6.6.4.2 Example problem using KENO V.a

This sample problem is critical experiment number 10 from evaluation LEU-COMP-THERM-009 of the ICSBEP Handbook [SAMSCom99]. The critical assembly consists of three 8 × 15 fuel-rod arrays separated by 6.46 mm thick copper plates. The U(4.31)O_2 fuel rods are 92.075-cm long. A KENO3D rendering of the KENO V.a geometry model with the water moderator removed is shown in Fig. 6.6.10.

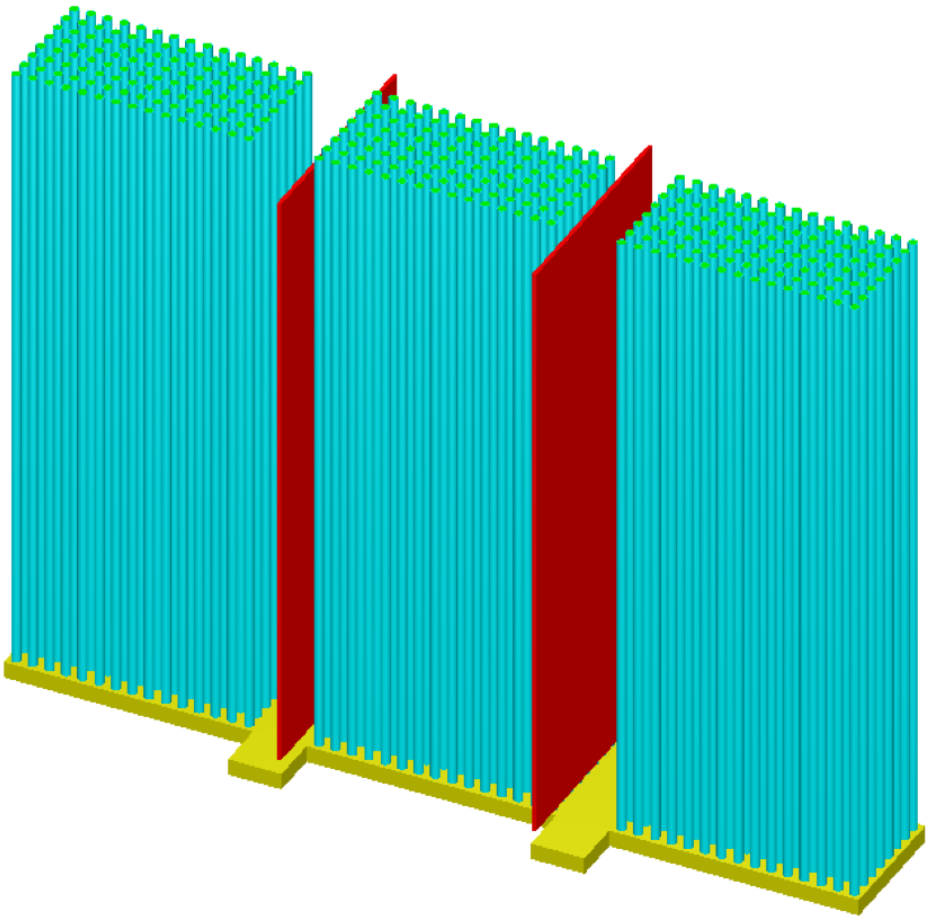


Fig. 6.6.10: KENO3D rendering of KENO V.a model of LEU-COMP-THERM-009 case 10.

Prior to executing SAMS, the data files listed in Table 6.6.4 were generated for this system. It is recommended that users perform sensitivity analyses on 3-D systems using the TSUNAMI-3D control module, which executes the appropriate codes to generate the required data files in an automated manner from a single input. SAMS is executed as a part of the TSUNAMI-3D control sequence. However, given appropriate data files SAMS can be run as a stand-alone program.

Input data and text output

The SAMS input is shown in Example 6.6.5 and the output is shown in Example 6.6.6. Each section of the output is described below. Note that output shown in this section is intended only to represent the format of output that the user will encounter. The actual computed results may vary.

1. Parameter Table and Summary of Transport Calculation

In the output listing, the SAMS Parameter Table gives the values of the SAMS input data. Next, the title from the KENO V.a input file is given and a brief summary of the transport solution is given.

2. Energy-, Region-, and Mixture Integrated Sensitivity Coefficients

Next the energy-, region-, and mixture integrated sensitivity coefficients are given for each reaction of each nuclide. The nuclide symbol and reaction name are given. The data given in the Sensitivity column represents the expected change in k_{eff} of this system due to a uniform fractional change in the cross section of the given nuclide and reaction in all energy groups for all regions and mixtures that contain this nuclide. The standard deviations represent stochastic uncertainties from the Monte Carlo calculation. For example, a uniform fractional change of 0.01 (or 1%) in the fission cross section of ^{235}U would produce a fractional change in k_{eff} of $3.2433 \cdot 10^{-1} \times 0.01$ or $3.2433 \cdot 10^{-3}$ (0.32433% $\Delta k/k$). Because the keyword PRTIMP was not entered in the SAMS input, only the complete sensitivity would be given in the Sensitivity column. If PRTIMP was entered, the explicit, implicit and complete sensitivity coefficients are listed in separate columns. Region-integrated sensitivity profiles for ^{1}H elastic scattering, ^{238}U capture and ^{235}U fission are shown in Fig. 6.6.11.

3. Energy- and Region-Integrated Sensitivity Coefficients

Next the energy- and region-integrated sensitivity coefficients are given for each reaction of each nuclide. The mixture number, nuclide symbol and reaction name are given. The data given in the Sensitivity column represents the expected change in k_{eff} of this system due to a uniform fractional change in the cross section of the given nuclide and reaction in all energy groups for all regions that contain the particular mixture. The standard deviations represent stochastic uncertainties from the Monte Carlo calculation. Because the keyword PRTIMP was not entered in the SAMS input, only the complete sensitivity would be given in the Sensitivity column. If PRTIMP was entered the explicit, implicit and complete sensitivity coefficients are listed in separate columns.

4. Total Sensitivity Coefficients by Nuclide

The next section of the output summarizes the total sensitivity by nuclide. The sensitivity values given here are the same as the values given in the previous section for the sensitivity of k_{eff} to the total cross section. Also included in this edit are the atom densities for each nuclide in each mixture.

5. Total Sensitivity Coefficients by Mixture

The next output edit gives the total sensitivity of k_{eff} to the mixture. Here, the sensitivity of k_{eff} to the total cross section is summed over each nuclide in a given mixture.

6. Problem Characterization (MG Calculations Only)

The next output edit is for the problem characterization data. This section includes the median neutron energy group causing fission, capture and scattering, average group for fission, capture and scattering, the average energy causing fission, capture and scattering, and the energy of the average lethargy causing fission, capture and scattering.

7. Sensitivity Coefficients by Region

If the keyword, PRTGEOM was entered in the SAMS input, the next output edit would give the energy-integrated sensitivity coefficients for each region of each unit defined in the KENO V.a geometry input. Note that for brevity, this output edit was not requested.

8. Uncertainty Information

The next output edit contains the uncertainty information. First, the percent relative standard deviation in k_{eff} $\left(\Delta k_{eff}/k_{eff} \times 100\%\right)$ due to cross-section-covariance data is given. Next a listing of the uncertainty terms as produced by Eq. (6.6.38) are given. The nuclide-reaction to nuclide-reaction covariance matrix responsible for the uncertainty contribution is given followed by the contribution to the uncertainty in terms of % $\Delta k/k$. The total uncertainty can be computed from individual values by adding the square of the values with positive signs and subtracting the square of the values with negative signs, then taking the square root. The negative values are the result of anti-correlations in the cross-section-covariance data.

9. Execution Complete

The final edit produced simply states that the execution is complete and gives the elapsed time.

Example 6.6.5: SAMS input LEU-COMP-THERM-009 case 10 sample problem.

```
=sams5
read initial xunit=   4 xunitm=  42
makeimp prtgeom end initial
end
```

Example 6.6.6: Parameter table.

```
------------------------------------------------------------
  SAMS:  Sensitivity Analysis Module for SCALE

------------------------------------------------------------

  SAMS Parameter Table

                                        transport solution from:   KENO V.a
                    use angular fluxes to compute flux moments:   false
                             forward transport solution on unit:   35
                             adjoint transport solution on unit:   36
                          working cross-section library on unit:   4
               master sensitivity cross-section library on unit:   42
                    short master cross-section library on unit:   11
                        covariance data library read from file:   56groupcov7.1
   order of flux moments requested for sensitivity calculations:   3
                                    make implicit sensitivities:   true
                                  print implicit sensitivities:   false
                               print sensitivities by mixture:   true
                     print sensitivities for each region/zone:   false
                                       print uncertainty data:   true
                                           print mesh volumes:   false
                                output binary SENPRO data file:   false
                                implicit sensitivity threshold:   100.0000
                                                  HTML output:   true
                             Chi sensitivities are constrained:   true
                              Fixup large/zero covariance data:   false
                                   Use Default Covariance Data:   false
                                 Use User Input Covariance Data:   false
```

```
HTML Format Options

                                Background color:   PapayaWhip
                             Major Headings color:   Maroon
                               Sub-headings color:   Navy
                                  Plain text color:   Black
                                   Hyperlink color:   Navy
                                    Hypelink style:   none
                                Visitied link color:   Navy
```

(continues on next page)

(continued from previous page)

```
                      Default Covariance color:   Blue
                   User Input Covariance color:   Red
   Corrected with default covariance data color:   RoyalBlue
 Corrected with user input covariance data color:   Green
```

```
f410p three 15x8 clusters, 2.54  cm pitch, 8.15   cm separation

              Summary of Transport Solution


               number of neutron groups:   238
                 number of gamma groups:   0
                      number of regions:   17
                 maximum mixture number:   7
         number of mixing table entries:   70
     quadrature order of angular fluxes:   0
          maximum order of flux moments:   3
                   mesh fluxes generated:   true
                differential sensitivity:   false
                               Forward Calculation
               histories per generation:   10000
                  number of generations:   1885
             k-eff = 0.99783 +/- 0.00019   which occurs for 101 generations skipped

                               Adjoint Calculation
               histories per generation:   30000
                  number of generations:   10200
                k-eff = 0.9988 +/- 0.0013   which occurs for 304 generations skipped
```

```
-------------------------------------------------------------------------------
   Energy, Region and Mixture Integrated Sensitivity Coefficients for this Problem
-------------------------------------------------------------------------------

   Nuclide       Reaction       Sensitivity      Std. Dev. % Std. Dev.
------------- ---------------   -----------     ---------- -----------
 h-1          total              1.6868E-01 +\- 7.6109E-04 (   0.45%)
 h-1          scatter            3.6624E-01 +\- 7.5366E-04 (   0.21%)
 h-1          elastic            3.6625E-01 +\- 7.5366E-04 (   0.21%)
 h-1          capture           -1.9756E-01 +\- 1.3139E-05 (   0.01%)
 h-1          n,gamma           -1.9756E-01 +\- 1.3139E-05 (   0.01%)
 c            total              3.0487E-04 +\- 4.0117E-06 (   1.32%)
 c            scatter            3.0745E-04 +\- 3.9729E-06 (   1.29%)
 c            elastic            3.0569E-04 +\- 3.9725E-06 (   1.30%)
 c            n,n'               1.7567E-06 +\- 2.2595E-08 (   1.29%)
 c            capture           -2.5788E-06 +\- 5.6099E-09 (   0.22%)
 c            n,gamma           -2.4558E-06 +\- 3.2370E-09 (   0.13%)
 c            n,p               -2.8483E-11 +\- 0.0000E+00 (   0.00%)
 c            n,d               -7.1686E-11 +\- 0.0000E+00 (   0.00%)
 c            n,alpha           -1.2290E-07 +\- 4.5772E-09 (   3.72%)
 o-16         total              7.4111E-02 +\- 5.3610E-05 (   0.07%)
 o-16         scatter            7.5947E-02 +\- 5.3599E-05 (   0.07%)
 o-16         elastic            7.5552E-02 +\- 5.3596E-05 (   0.07%)
 o-16         n,n'               3.8299E-04 +\- 2.5943E-07 (   0.07%)
 o-16         n,2n               3.7303E-11 +\- 1.5211E-17 (   0.00%)
 o-16         capture           -1.8367E-03 +\- 2.2526E-07 (   0.01%)
 o-16         n,gamma           -6.6536E-05 +\- 3.9010E-09 (   0.01%)
 o-16         n,p               -8.9997E-06 +\- 8.7659E-09 (   0.10%)
 o-16         n,d               -9.2619E-07 +\- 1.7089E-09 (   0.18%)
 o-16         n,t               -3.3277E-12 +\- 2.7563E-18 (   0.00%)
 o-16         n,alpha           -1.7602E-03 +\- 2.2246E-07 (   0.01%)
 na-23        total              2.3175E-08 +\- 1.4447E-09 (   6.23%)
 na-23        scatter            5.7809E-08 +\- 1.4371E-09 (   2.49%)
 na-23        elastic            5.2606E-08 +\- 1.4371E-09 (   2.73%)
 na-23        n,n'               5.2023E-09 +\- 7.8251E-15 (   0.00%)
```

(continues on next page)

(continued from previous page)

```
na-23       n,2n             4.5318E-13 +\- 0.0000E+00 (   0.00%)
na-23       capture         -3.4634E-08 +\- 2.0188E-11 (   0.06%)
na-23       n,gamma         -3.4409E-08 +\- 2.0188E-11 (   0.06%)
na-23       n,p             -1.5160E-10 +\- 3.2974E-18 (   0.00%)
na-23       n,alpha         -7.2671E-11 +\- 5.0907E-18 (   0.00%)
mg-24       total            4.5199E-05 +\- 4.5339E-07 (   1.00%)
mg-24       scatter          5.5375E-05 +\- 4.5356E-07 (   0.82%)
mg-24       elastic          4.3580E-05 +\- 4.5024E-07 (   1.03%)
mg-24       n,n'             1.1783E-05 +\- 3.3434E-08 (   0.28%)
mg-24       n,2n             5.8604E-13 +\- 1.0107E-22 (   0.00%)
mg-24       capture         -1.0176E-05 +\- 3.6669E-09 (   0.04%)
mg-24       n,gamma         -9.1064E-06 +\- 3.1240E-09 (   0.03%)
mg-24       n,p             -2.5472E-07 +\- 5.7541E-10 (   0.23%)
mg-24       n,alpha         -8.1482E-07 +\- 1.3622E-09 (   0.17%)
mg-25       total            1.2162E-06 +\- 3.2733E-08 (   2.69%)
mg-25       scatter          5.8494E-06 +\- 3.1369E-08 (   0.54%)
mg-25       elastic          4.2267E-06 +\- 3.0490E-08 (   0.72%)
mg-25       n,n'             1.5857E-06 +\- 3.7904E-09 (   0.24%)
mg-25       n,2n             3.6950E-08 +\- 9.6404E-11 (   0.26%)
mg-25       capture         -4.6331E-06 +\- 1.5204E-09 (   0.03%)
mg-25       n,gamma         -4.2713E-06 +\- 1.4848E-09 (   0.03%)
mg-25       n,p             -3.2385E-08 +\- 3.3458E-11 (   0.10%)
mg-25       n,alpha         -3.2950E-07 +\- 2.8796E-10 (   0.09%)
mg-26       total            5.3028E-06 +\- 3.2698E-08 (   0.62%)
mg-26       scatter          6.2567E-06 +\- 3.2364E-08 (   0.52%)
mg-26       elastic          4.8131E-06 +\- 3.1809E-08 (   0.66%)
mg-26       n,n'             1.4401E-06 +\- 3.4023E-09 (   0.24%)
mg-26       n,2n             3.6129E-09 +\- 2.0677E-18 (   0.00%)
mg-26       capture         -9.5390E-07 +\- 3.2481E-10 (   0.03%)
mg-26       n,gamma         -9.4936E-07 +\- 3.2481E-10 (   0.03%)
mg-26       n,p             -5.2412E-10 +\- 1.6179E-19 (   0.00%)
mg-26       n,alpha         -4.0217E-09 +\- 5.7744E-19 (   0.00%)
al-27       total           -6.7582E-05 +\- 2.3201E-05 (  34.33%)
al-27       scatter          4.6080E-03 +\- 2.2663E-05 (   0.49%)
```

```
al-27       elastic          3.2847E-03 +\- 2.1856E-05 (   0.67%)
al-27       n,n'             1.3189E-03 +\- 3.0167E-06 (   0.23%)
al-27       n,2n             5.9944E-08 +\- 1.4100E-09 (   2.35%)
al-27       capture         -4.6756E-03 +\- 1.6107E-06 (   0.03%)
al-27       n,gamma         -4.5949E-03 +\- 1.6079E-06 (   0.03%)
al-27       n,p             -6.7911E-05 +\- 6.9477E-08 (   0.10%)
al-27       n,d             -1.9733E-07 +\- 1.4218E-09 (   0.72%)
al-27       n,t             -6.6132E-09 +\- 5.0053E-16 (   0.00%)
al-27       n,alpha         -1.2531E-05 +\- 3.2986E-08 (   0.26%)
si-28       total            4.6433E-06 +\- 1.2249E-07 (   2.64%)
si-28       scatter          2.3947E-05 +\- 1.2025E-07 (   0.50%)
si-28       elastic          1.9104E-05 +\- 1.1760E-07 (   0.62%)
si-28       n,n'             4.8407E-06 +\- 1.3004E-08 (   0.27%)
si-28       capture         -1.9303E-05 +\- 6.5597E-09 (   0.03%)
si-28       n,gamma         -1.8340E-05 +\- 6.3655E-09 (   0.03%)
si-28       n,p             -6.6369E-07 +\- 1.0533E-09 (   0.16%)
si-28       n,d             -8.9840E-10 +\- 1.1828E-18 (   0.00%)
si-28       n,alpha         -2.9909E-07 +\- 5.3824E-10 (   0.18%)
si-29       total            5.5254E-07 +\- 5.6554E-09 (   1.02%)
si-29       scatter          1.2730E-06 +\- 4.0360E-09 (   0.32%)
si-29       elastic          8.2855E-07 +\- 3.6454E-09 (   0.44%)
si-29       n,n'             4.4317E-07 +\- 9.9759E-10 (   0.23%)
si-29       n,2n             1.2964E-09 +\- 8.4820E-19 (   0.00%)
si-29       capture         -7.2047E-07 +\- 2.3029E-10 (   0.03%)
si-29       n,gamma         -6.7259E-07 +\- 2.2493E-10 (   0.03%)
si-29       n,p             -1.7997E-08 +\- 1.7962E-11 (   0.10%)
si-29       n,alpha         -2.9884E-08 +\- 2.2339E-11 (   0.07%)
si-30       total            1.4343E-07 +\- 4.4925E-09 (   3.13%)
si-30       scatter          7.5256E-07 +\- 3.2216E-09 (   0.43%)
```

(continues on next page)

(continued from previous page)

```
si-30       elastic          5.6016E-07 +\- 3.1391E-09 (   0.56%)
si-30       n,n'             1.9152E-07 +\- 4.2667E-10 (   0.22%)
si-30       n,2n             8.8318E-10 +\- 5.2234E-19 (   0.00%)
si-30       capture         -6.0913E-07 +\- 8.1712E-10 (   0.13%)
si-30       n,gamma         -6.0862E-07 +\- 8.1712E-10 (   0.13%)
si-30       n,p             -9.3872E-11 +\- 1.2377E-19 (   0.00%)
si-30       n,alpha         -4.1605E-10 +\- 3.2227E-19 (   0.00%)
s-32        total           -9.4450E-07 +\- 1.5023E-08 (   1.59%)
s-32        scatter          1.1163E-06 +\- 1.1617E-08 (   1.04%)
s-32        elastic          1.0496E-06 +\- 1.1393E-08 (   1.09%)
s-32        n,n'             6.6511E-08 +\- 8.6196E-10 (   1.30%)
s-32        n,2n             1.0953E-14 +\- 0.0000E+00 (   0.00%)
s-32        capture         -2.0608E-06 +\- 4.4706E-09 (   0.22%)
s-32        n,gamma         -1.9066E-06 +\- 4.2718E-09 (   0.22%)
s-32        n,p             -4.8964E-08 +\- 5.3423E-10 (   1.09%)
s-32        n,alpha         -1.0530E-07 +\- 5.3326E-10 (   0.51%)
s-33        total           -1.2276E-08 +\- 2.3332E-14 (   0.00%)
s-33        scatter          8.0190E-09 +\- 2.1686E-14 (   0.00%)
s-33        elastic          7.2275E-09 +\- 2.1662E-14 (   0.00%)
```

```
s-33        n,n'             7.8504E-10 +\- 1.7509E-16 (   0.00%)
s-33        n,2n             5.0222E-12 +\- 2.0474E-20 (   0.00%)
s-33        capture         -2.0295E-08 +\- 1.6612E-15 (   0.00%)
s-33        n,gamma         -9.9685E-09 +\- 3.3428E-17 (   0.00%)
s-33        n,p             -4.7345E-10 +\- 1.0045E-16 (   0.00%)
s-33        n,alpha         -9.8528E-09 +\- 1.6143E-15 (   0.00%)
s-34        total            1.2906E-08 +\- 3.9241E-10 (   3.04%)
s-34        scatter          4.8732E-08 +\- 1.6902E-10 (   0.35%)
s-34        elastic          4.4628E-08 +\- 1.1281E-10 (   0.25%)
s-34        n,n'             4.0885E-09 +\- 7.8533E-16 (   0.00%)
s-34        n,2n             6.7127E-12 +\- 1.8198E-20 (   0.00%)
s-34        capture         -3.5826E-08 +\- 6.1670E-11 (   0.17%)
s-34        n,gamma         -3.5723E-08 +\- 6.1670E-11 (   0.17%)
s-34        n,p             -1.9406E-11 +\- 1.4282E-18 (   0.00%)
s-34        n,alpha         -8.3072E-11 +\- 9.4595E-18 (   0.00%)
s-36        total            1.3056E-10 +\- 5.1210E-16 (   0.00%)
s-36        scatter          2.4907E-10 +\- 5.1183E-16 (   0.00%)
s-36        elastic          2.3590E-10 +\- 5.1180E-16 (   0.00%)
s-36        n,n'             1.2962E-11 +\- 1.1331E-18 (   0.00%)
s-36        n,2n             2.0465E-13 +\- 8.5709E-22 (   0.00%)
s-36        capture         -1.1851E-10 +\- 3.2691E-19 (   0.00%)
s-36        n,gamma         -1.1850E-10 +\- 3.2691E-19 (   0.00%)
s-36        n,p             -7.0565E-17 +\- 1.4329E-24 (   0.00%)
s-36        n,alpha         -5.9656E-15 +\- 2.0535E-22 (   0.00%)
ca-40       total           -1.7949E-06 +\- 1.1894E-07 (   6.63%)
ca-40       scatter          6.7330E-06 +\- 9.7909E-08 (   1.45%)
ca-40       elastic          6.5226E-06 +\- 9.7589E-08 (   1.50%)
ca-40       n,n'             2.0785E-07 +\- 3.3003E-09 (   1.59%)
ca-40       n,2n             4.5607E-14 +\- 0.0000E+00 (   0.00%)
ca-40       capture         -8.5279E-06 +\- 1.9716E-08 (   0.23%)
ca-40       n,gamma         -7.9727E-06 +\- 1.8069E-08 (   0.23%)
ca-40       n,p             -3.7411E-07 +\- 5.5225E-09 (   1.48%)
ca-40       n,d             -5.9766E-11 +\- 6.5478E-19 (   0.00%)
ca-40       n,t             -4.2787E-15 +\- 0.0000E+00 (   0.00%)
ca-40       n,he-3          -5.7096E-14 +\- 0.0000E+00 (   0.00%)
ca-40       n,alpha         -1.8104E-07 +\- 2.0743E-09 (   1.15%)
ca-42       total           -3.3798E-08 +\- 4.2370E-10 (   1.25%)
ca-42       scatter          5.5595E-08 +\- 1.8840E-10 (   0.34%)
ca-42       elastic          4.4749E-08 +\- 7.7226E-14 (   0.00%)
ca-42       n,n'             1.0831E-08 +\- 3.5480E-15 (   0.00%)
ca-42       n,2n             4.2807E-12 +\- 0.0000E+00 (   0.00%)
ca-42       capture         -8.9393E-08 +\- 1.9154E-10 (   0.21%)
ca-42       n,gamma         -8.9285E-08 +\- 1.9154E-10 (   0.21%)
ca-42       n,p             -6.3001E-11 +\- 6.1936E-18 (   0.00%)
```

(continues on next page)

(continued from previous page)

```
ca-42       n,d             -3.9571E-14 +\- 0.0000E+00 (   0.00%)
ca-42       n,t             -1.1401E-16 +\- 0.0000E+00 (   0.00%)
ca-42       n,he-3          -2.8628E-18 +\- 0.0000E+00 (   0.00%)
ca-42       n,alpha         -4.4886E-11 +\- 4.9613E-18 (   0.00%)
```

```
ca-43       total           -2.9992E-07 +\- 8.5309E-10 (   0.28%)
ca-43       scatter          1.7308E-08 +\- 9.3864E-15 (   0.00%)
ca-43       elastic          1.4035E-08 +\- 8.9886E-15 (   0.00%)
ca-43       n,n'             3.2593E-09 +\- 1.0415E-15 (   0.00%)
ca-43       n,2n             1.3992E-11 +\- 1.0619E-22 (   0.00%)
ca-43       capture         -3.1723E-07 +\- 7.1329E-10 (   0.22%)
ca-43       n,gamma         -3.1720E-07 +\- 7.1328E-10 (   0.22%)
ca-43       n,p             -1.2736E-11 +\- 1.4953E-18 (   0.00%)
ca-43       n,d             -3.9211E-15 +\- 0.0000E+00 (   0.00%)
ca-43       n,t             -2.0903E-16 +\- 0.0000E+00 (   0.00%)
ca-43       n,he-3          -8.8381E-21 +\- 0.0000E+00 (   0.00%)
ca-43       n,alpha         -1.8078E-11 +\- 1.6658E-18 (   0.00%)
ca-44       total           -1.9121E-07 +\- 2.9350E-09 (   1.53%)
ca-44       scatter          1.8201E-07 +\- 9.8994E-10 (   0.54%)
ca-44       elastic          1.4003E-07 +\- 6.5307E-10 (   0.47%)
ca-44       n,n'             4.1934E-08 +\- 3.9319E-10 (   0.94%)
ca-44       n,2n             3.7701E-11 +\- 0.0000E+00 (   0.00%)
ca-44       capture         -3.7322E-07 +\- 8.4317E-10 (   0.23%)
ca-44       n,gamma         -3.7321E-07 +\- 8.4317E-10 (   0.23%)
ca-44       n,p             -4.3462E-12 +\- 1.1952E-19 (   0.00%)
ca-44       n,d             -1.9015E-14 +\- 0.0000E+00 (   0.00%)
ca-44       n,t             -3.6521E-17 +\- 0.0000E+00 (   0.00%)
ca-44       n,he-3          -5.7133E-22 +\- 0.0000E+00 (   0.00%)
ca-44       n,alpha         -2.1833E-12 +\- 5.2524E-20 (   0.00%)
ca-46       total           -9.1789E-11 +\- 2.9314E-16 (   0.00%)
ca-46       scatter          5.0404E-10 +\- 2.9308E-16 (   0.00%)
ca-46       elastic          4.3313E-10 +\- 2.8501E-16 (   0.00%)
ca-46       n,n'             7.0696E-11 +\- 2.2304E-17 (   0.00%)
ca-46       n,2n             2.0718E-13 +\- 0.0000E+00 (   0.00%)
ca-46       capture         -5.9583E-10 +\- 5.9929E-20 (   0.00%)
ca-46       n,gamma         -5.9583E-10 +\- 5.9929E-20 (   0.00%)
ca-46       n,p             -3.2267E-16 +\- 3.6714E-26 (   0.00%)
ca-46       n,d             -6.8722E-18 +\- 0.0000E+00 (   0.00%)
ca-46       n,t             -3.3633E-21 +\- 0.0000E+00 (   0.00%)
ca-46       n,alpha         -2.2633E-17 +\- 0.0000E+00 (   0.00%)
ca-48       total           -2.4743E-08 +\- 1.8724E-10 (   0.76%)
ca-48       scatter          1.6341E-08 +\- 8.2057E-15 (   0.00%)
ca-48       elastic          1.5512E-08 +\- 8.2022E-15 (   0.00%)
ca-48       n,n'             8.1356E-10 +\- 6.9284E-17 (   0.00%)
ca-48       n,2n             1.5225E-11 +\- 0.0000E+00 (   0.00%)
ca-48       capture         -4.1084E-08 +\- 7.9141E-11 (   0.19%)
ca-48       n,gamma         -4.1084E-08 +\- 7.9141E-11 (   0.19%)
ca-48       n,p             -1.0552E-16 +\- 0.0000E+00 (   0.00%)
ca-48       n,d             -2.1494E-17 +\- 0.0000E+00 (   0.00%)
ca-48       n,t             -5.0004E-22 +\- 0.0000E+00 (   0.00%)
ca-48       n,alpha         -1.9945E-17 +\- 0.0000E+00 (   0.00%)
ti-46       total           -1.8856E-07 +\- 8.6800E-10 (   0.46%)
```

```
ti-46       scatter          2.1933E-07 +\- 4.7014E-10 (   0.21%)
ti-46       elastic          1.2896E-07 +\- 3.2795E-10 (   0.25%)
ti-46       n,n'             9.0312E-08 +\- 1.9591E-10 (   0.22%)
ti-46       n,2n             4.8498E-12 +\- 2.8827E-21 (   0.00%)
ti-46       capture         -4.0789E-07 +\- 1.3319E-10 (   0.03%)
ti-46       n,gamma         -3.9961E-07 +\- 1.3319E-10 (   0.03%)
ti-46       n,p             -7.7485E-09 +\- 3.8514E-18 (   0.00%)
ti-46       n,d             -3.1806E-13 +\- 2.3627E-22 (   0.00%)
ti-46       n,t             -9.9763E-17 +\- 3.2655E-26 (   0.00%)
ti-46       n,he-3          -4.2439E-16 +\- 1.9547E-25 (   0.00%)
ti-46       n,alpha         -5.2794E-10 +\- 2.5965E-19 (   0.00%)
```

(continues on next page)

(continued from previous page)

```
ti-47       total           -8.7007E-07 +\- 2.4463E-09 (   0.28%)
ti-47       scatter          2.4268E-07 +\- 2.2853E-09 (   0.94%)
ti-47       elastic          1.3531E-07 +\- 2.2669E-09 (   1.68%)
ti-47       n,n'             1.0717E-07 +\- 1.8872E-10 (   0.18%)
ti-47       n,2n             1.9211E-10 +\- 4.0015E-20 (   0.00%)
ti-47       capture         -1.1128E-06 +\- 3.6552E-10 (   0.03%)
ti-47       n,gamma         -1.1003E-06 +\- 3.6552E-10 (   0.03%)
ti-47       n,p             -1.0646E-08 +\- 1.4197E-16 (   0.00%)
ti-47       n,d             -2.0197E-13 +\- 1.5135E-22 (   0.00%)
ti-47       n,t             -4.3601E-15 +\- 3.0922E-24 (   0.00%)
ti-47       n,he-3          -2.3300E-17 +\- 7.4079E-27 (   0.00%)
ti-47       n,alpha         -1.8482E-09 +\- 1.3222E-17 (   0.00%)
ti-48       total           -4.5957E-05 +\- 3.3263E-08 (   0.07%)
ti-48       scatter          3.0030E-06 +\- 2.3419E-08 (   0.78%)
ti-48       elastic          2.1566E-06 +\- 2.3249E-08 (   1.08%)
ti-48       n,n'             8.4526E-07 +\- 1.8984E-09 (   0.22%)
ti-48       n,2n             1.0593E-09 +\- 5.3938E-19 (   0.00%)
ti-48       capture         -4.8960E-05 +\- 1.7297E-08 (   0.04%)
ti-48       n,gamma         -4.8958E-05 +\- 1.7297E-08 (   0.04%)
ti-48       n,p             -1.6058E-09 +\- 9.4652E-19 (   0.00%)
ti-48       n,d             -1.6876E-12 +\- 1.6673E-21 (   0.00%)
ti-48       n,t             -8.0952E-16 +\- 2.7332E-25 (   0.00%)
ti-48       n,he-3          -1.9723E-17 +\- 6.7861E-27 (   0.00%)
ti-48       n,alpha         -1.5654E-10 +\- 4.0991E-20 (   0.00%)
ti-49       total           -6.8733E-07 +\- 6.2660E-10 (   0.09%)
ti-49       scatter          1.6784E-07 +\- 2.2439E-09 (   1.34%)
ti-49       elastic          1.0265E-07 +\- 2.2349E-09 (   2.18%)
ti-49       n,n'             6.4463E-08 +\- 1.2032E-10 (   0.19%)
ti-49       n,2n             7.2472E-10 +\- 7.6807E-20 (   0.00%)
ti-49       capture         -8.5517E-07 +\- 2.9502E-10 (   0.03%)
ti-49       n,gamma         -8.5476E-07 +\- 2.9502E-10 (   0.03%)
ti-49       n,p             -3.3313E-10 +\- 3.2208E-19 (   0.00%)
ti-49       n,d             -7.4647E-14 +\- 6.7611E-23 (   0.00%)
ti-49       n,t             -2.1162E-15 +\- 9.3763E-25 (   0.00%)
ti-49       n,he-3          -9.3750E-20 +\- 0.0000E+00 (   0.00%)
ti-49       n,alpha         -7.1068E-11 +\- 4.3628E-20 (   0.00%)
ti-50       total            5.1758E-08 +\- 4.1741E-10 (   0.81%)
```

```
ti-50       scatter          1.3378E-07 +\- 1.9577E-10 (   0.15%)
ti-50       elastic          8.8832E-08 +\- 1.1755E-10 (   0.13%)
ti-50       n,n'             4.4731E-08 +\- 8.8290E-11 (   0.20%)
ti-50       n,2n             2.1770E-10 +\- 7.0963E-20 (   0.00%)
ti-50       capture         -8.2023E-08 +\- 2.5593E-11 (   0.03%)
ti-50       n,gamma         -8.2018E-08 +\- 2.5593E-11 (   0.03%)
ti-50       n,p             -2.9356E-12 +\- 9.4805E-22 (   0.00%)
ti-50       n,d             -1.7101E-14 +\- 1.1908E-23 (   0.00%)
ti-50       n,t             -1.0953E-17 +\- 2.8155E-27 (   0.00%)
ti-50       n,he-3          -4.0903E-23 +\- 0.0000E+00 (   0.00%)
ti-50       n,alpha         -1.7272E-12 +\- 6.5310E-22 (   0.00%)
cr-50       total           -1.3040E-05 +\- 9.1891E-09 (   0.07%)
cr-50       scatter          1.3376E-06 +\- 7.2895E-09 (   0.54%)
cr-50       elastic          1.2307E-06 +\- 7.2762E-09 (   0.59%)
cr-50       n,n'             1.0681E-07 +\- 2.3841E-10 (   0.22%)
cr-50       n,2n             3.1922E-12 +\- 2.3793E-21 (   0.00%)
cr-50       capture         -1.4377E-05 +\- 5.0788E-09 (   0.04%)
cr-50       n,gamma         -1.4333E-05 +\- 5.0787E-09 (   0.04%)
cr-50       n,p             -4.1672E-08 +\- 3.2114E-11 (   0.08%)
cr-50       n,d             -7.5849E-12 +\- 5.4355E-19 (   0.00%)
cr-50       n,alpha         -2.7989E-09 +\- 1.7847E-15 (   0.00%)
cr-52       total           -8.7501E-06 +\- 2.6630E-08 (   0.30%)
cr-52       scatter          4.4850E-06 +\- 2.1763E-08 (   0.49%)
cr-52       elastic          2.3489E-06 +\- 2.0602E-08 (   0.88%)
cr-52       n,n'             2.1346E-06 +\- 4.5774E-09 (   0.21%)
cr-52       n,2n             1.3180E-09 +\- 9.1937E-19 (   0.00%)
```

(continues on next page)

(continued from previous page)

```
cr-52       capture          -1.3235E-05 +\- 4.5982E-09 (   0.03%)
cr-52       n,gamma          -1.3217E-05 +\- 4.5982E-09 (   0.03%)
cr-52       n,p              -1.5832E-08 +\- 1.8313E-11 (   0.12%)
cr-52       n,alpha          -1.9304E-09 +\- 3.8775E-17 (   0.00%)
cr-53       total            -3.4245E-05 +\- 2.1670E-08 (   0.06%)
cr-53       scatter           1.3865E-06 +\- 1.4475E-08 (   1.04%)
cr-53       elastic           1.0474E-06 +\- 1.4432E-08 (   1.38%)
cr-53       n,n'              3.3527E-07 +\- 7.1251E-10 (   0.21%)
cr-53       n,2n              3.8152E-09 +\- 3.6614E-17 (   0.00%)
cr-53       capture          -3.5632E-05 +\- 1.2596E-08 (   0.04%)
cr-53       n,gamma          -3.5630E-05 +\- 1.2596E-08 (   0.04%)
cr-53       n,p              -6.9509E-10 +\- 5.2410E-17 (   0.00%)
cr-53       n,alpha          -1.5386E-09 +\- 1.4981E-16 (   0.00%)
cr-54       total            -1.6695E-08 +\- 4.1104E-10 (   2.46%)
cr-54       scatter           1.6318E-07 +\- 2.3409E-10 (   0.14%)
cr-54       elastic           9.3303E-08 +\- 7.9919E-11 (   0.09%)
cr-54       n,n'              6.9373E-08 +\- 1.3815E-10 (   0.20%)
cr-54       n,2n              5.0486E-10 +\- 8.3567E-20 (   0.00%)
cr-54       capture          -1.7988E-07 +\- 5.9710E-11 (   0.03%)
cr-54       n,gamma          -1.7985E-07 +\- 5.9710E-11 (   0.03%)
cr-54       n,p              -4.2582E-12 +\- 1.1391E-21 (   0.00%)
cr-54       n,alpha          -2.1413E-11 +\- 1.0788E-20 (   0.00%)
```

```
mn-55       total            -9.6360E-05 +\- 1.4079E-07 (   0.15%)
mn-55       scatter           6.8704E-06 +\- 1.3262E-07 (   1.93%)
mn-55       elastic           5.7292E-06 +\- 1.3254E-07 (   2.31%)
mn-55       n,n'              1.1366E-06 +\- 2.5436E-09 (   0.22%)
mn-55       n,2n              4.4184E-09 +\- 1.0180E-18 (   0.00%)
mn-55       capture          -1.0323E-04 +\- 3.5222E-08 (   0.03%)
mn-55       n,gamma          -1.0323E-04 +\- 3.5222E-08 (   0.03%)
mn-55       n,p              -3.7300E-09 +\- 9.1810E-18 (   0.00%)
mn-55       n,d              -3.5791E-11 +\- 6.7992E-21 (   0.00%)
mn-55       n,t              -1.3299E-12 +\- 1.2646E-21 (   0.00%)
mn-55       n,he-3           -4.6031E-14 +\- 4.8445E-23 (   0.00%)
mn-55       n,alpha          -8.3737E-10 +\- 6.0947E-19 (   0.00%)
fe-54       total            -3.9443E-06 +\- 8.5217E-09 (   0.22%)
fe-54       scatter           7.5526E-07 +\- 8.1028E-09 (   1.07%)
fe-54       elastic           5.5856E-07 +\- 8.0723E-09 (   1.45%)
fe-54       n,n'              1.9618E-07 +\- 4.3649E-10 (   0.22%)
fe-54       n,2n              3.8024E-12 +\- 1.3165E-21 (   0.00%)
fe-54       capture          -4.6996E-06 +\- 1.5839E-09 (   0.03%)
fe-54       n,gamma          -4.5471E-06 +\- 1.5807E-09 (   0.03%)
fe-54       n,p              -1.5084E-07 +\- 9.9241E-11 (   0.07%)
fe-54       n,d              -2.5872E-11 +\- 1.0217E-20 (   0.00%)
fe-54       n,alpha          -1.5885E-09 +\- 6.8285E-19 (   0.00%)
fe-56       total            -6.9531E-05 +\- 1.3507E-07 (   0.19%)
fe-56       scatter           1.1558E-05 +\- 1.1066E-07 (   0.96%)
fe-56       elastic           7.5772E-06 +\- 1.0976E-07 (   1.45%)
fe-56       n,n'              3.9750E-06 +\- 9.5385E-09 (   0.24%)
fe-56       n,2n              5.5701E-09 +\- 3.2182E-18 (   0.00%)
fe-56       capture          -8.1090E-05 +\- 2.8413E-08 (   0.04%)
fe-56       n,gamma          -8.1049E-05 +\- 2.8413E-08 (   0.04%)
fe-56       n,p              -2.8313E-08 +\- 4.3821E-11 (   0.15%)
fe-56       n,d              -6.3279E-11 +\- 5.4153E-20 (   0.00%)
fe-56       n,t              -1.5517E-13 +\- 1.2163E-22 (   0.00%)
fe-56       n,he-3           -1.7491E-14 +\- 1.2250E-23 (   0.00%)
fe-56       n,alpha          -1.1979E-08 +\- 1.2530E-11 (   0.10%)
fe-57       total            -1.4596E-06 +\- 2.4578E-09 (   0.17%)
fe-57       scatter           2.9573E-07 +\- 2.2278E-09 (   0.75%)
fe-57       elastic           1.6174E-07 +\- 2.1139E-09 (   1.31%)
fe-57       n,n'              1.3200E-07 +\- 2.4140E-10 (   0.18%)
fe-57       n,2n              1.9892E-09 +\- 3.7761E-19 (   0.00%)
fe-57       capture          -1.7553E-06 +\- 6.0368E-10 (   0.03%)
fe-57       n,gamma          -1.7542E-06 +\- 6.0368E-10 (   0.03%)
```

(continues on next page)

(continued from previous page)

```
fe-57       n,p             -4.5820E-10 +\- 1.9581E-19 (   0.00%)
fe-57       n,alpha         -6.5747E-10 +\- 3.5652E-19 (   0.00%)
fe-58       total           -8.6319E-08 +\- 9.2863E-11 (   0.11%)
fe-58       scatter          3.2999E-08 +\- 2.1157E-13 (   0.00%)
fe-58       elastic          1.9560E-08 +\- 2.1157E-13 (   0.00%)
fe-58       n,n'             1.3357E-08 +\- 1.7645E-16 (   0.00%)
fe-58       n,2n             8.1863E-11 +\- 2.4070E-20 (   0.00%)
```

```
fe-58       capture         -1.1932E-07 +\- 3.5967E-11 (   0.03%)
fe-58       n,gamma         -1.1931E-07 +\- 3.5967E-11 (   0.03%)
fe-58       n,p             -8.9843E-13 +\- 3.3131E-22 (   0.00%)
fe-58       n,alpha         -5.5353E-12 +\- 1.7256E-21 (   0.00%)
cu-63       total           -3.2291E-03 +\- 6.7644E-06 (   0.21%)
cu-63       scatter          5.2401E-04 +\- 4.5049E-06 (   0.86%)
cu-63       elastic          4.4000E-04 +\- 4.1846E-06 (   0.95%)
cu-63       n,n'             8.2502E-05 +\- 1.1401E-06 (   1.38%)
cu-63       n,2n             2.1792E-07 +\- 7.0126E-09 (   3.22%)
cu-63       capture         -3.7531E-03 +\- 3.1714E-06 (   0.08%)
cu-63       n,gamma         -3.7267E-03 +\- 3.1712E-06 (   0.09%)
cu-63       n,p             -2.5132E-05 +\- 3.5516E-08 (   0.14%)
cu-63       n,d             -5.3278E-07 +\- 1.5798E-09 (   0.30%)
cu-63       n,he-3          -3.1134E-12 +\- 3.9260E-19 (   0.00%)
cu-63       n,alpha         -6.8776E-07 +\- 2.7471E-09 (   0.40%)
cu-65       total           -6.2532E-04 +\- 4.7383E-06 (   0.76%)
cu-65       scatter          1.8352E-04 +\- 4.2466E-06 (   2.31%)
cu-65       elastic          1.4901E-04 +\- 4.1861E-06 (   2.81%)
cu-65       n,n'             3.4114E-05 +\- 4.9055E-07 (   1.44%)
cu-65       n,2n             3.8621E-07 +\- 7.8368E-09 (   2.03%)
cu-65       capture         -8.0884E-04 +\- 6.8255E-07 (   0.08%)
cu-65       n,gamma         -8.0848E-04 +\- 6.8255E-07 (   0.08%)
cu-65       n,p             -3.1527E-07 +\- 7.8886E-10 (   0.25%)
cu-65       n,d             -4.1498E-08 +\- 1.8559E-10 (   0.45%)
cu-65       n,t             -1.0755E-11 +\- 2.7949E-18 (   0.00%)
cu-65       n,he-3          -3.9455E-13 +\- 1.3313E-20 (   0.00%)
cu-65       n,alpha         -9.0375E-09 +\- 1.3569E-15 (   0.00%)
u-234       total           -9.0979E-04 +\- 3.1204E-07 (   0.03%)
u-234       scatter          6.2441E-06 +\- 1.2041E-08 (   0.19%)
u-234       elastic          4.0083E-06 +\- 1.1854E-08 (   0.30%)
u-234       n,n'             2.2050E-06 +\- 1.4630E-09 (   0.07%)
u-234       n,2n             3.0794E-08 +\- 1.7472E-11 (   0.06%)
u-234       fission          2.2909E-05 +\- 1.7241E-09 (   0.01%)
u-234       capture         -9.3894E-04 +\- 3.0291E-07 (   0.03%)
u-234       n,gamma         -9.3894E-04 +\- 3.0291E-07 (   0.03%)
u-234       nubar            3.6021E-05 +\- 2.6348E-09 (   0.01%)
u-234       chi              8.7265E-13 +\- 1.5382E-09 (-999.99%)
u-235       total            2.0616E-01 +\- 7.0501E-05 (   0.03%)
u-235       scatter          8.8279E-04 +\- 1.0661E-06 (   0.12%)
u-235       elastic          3.1799E-04 +\- 9.6547E-07 (   0.30%)
u-235       n,n'             5.3198E-04 +\- 3.3888E-07 (   0.06%)
u-235       n,2n             3.2795E-05 +\- 1.0524E-08 (   0.03%)
u-235       fission          3.2433E-01 +\- 6.2460E-05 (   0.02%)
u-235       capture         -1.1906E-01 +\- 1.1810E-05 (   0.01%)
u-235       n,gamma         -1.1906E-01 +\- 1.1810E-05 (   0.01%)
u-235       nubar            9.6382E-01 +\- 8.0129E-05 (   0.01%)
u-235       chi             -6.6884E-08 +\- 4.6474E-05 (-999.99%)
u-236       total           -2.1805E-04 +\- 1.2725E-07 (   0.06%)
```

```
u-236       scatter          5.6471E-06 +\- 9.6565E-09 (   0.17%)
u-236       elastic          2.3929E-06 +\- 9.2543E-09 (   0.39%)
u-236       n,n'             3.1626E-06 +\- 2.0863E-09 (   0.07%)
u-236       n,2n             9.1002E-08 +\- 4.1257E-11 (   0.05%)
u-236       fission          1.2829E-05 +\- 1.7760E-09 (   0.01%)
u-236       capture         -2.3653E-04 +\- 1.2003E-07 (   0.05%)
u-236       n,gamma         -2.3653E-04 +\- 1.2003E-07 (   0.05%)
```

(continues on next page)

(continued from previous page)

```
u-236        nubar              2.1111E-05 +\- 1.6677E-09 (   0.01%)
u-236        chi               -4.2095E-13 +\- 1.0077E-09 (-999.99%)
u-238        total             -5.9609E-02 +\- 2.6857E-05 (   0.05%)
u-238        scatter            2.9406E-02 +\- 2.0373E-05 (   0.07%)
u-238        elastic            1.3217E-02 +\- 1.6047E-05 (   0.12%)
u-238        n,n'               1.5251E-02 +\- 9.7098E-06 (   0.06%)
u-238        n,2n               9.2840E-04 +\- 3.7033E-07 (   0.04%)
u-238        fission            2.4727E-02 +\- 2.4196E-06 (   0.01%)
u-238        capture           -1.1374E-01 +\- 9.9579E-06 (   0.01%)
u-238        n,gamma           -1.1374E-01 +\- 9.9579E-06 (   0.01%)
u-238        nubar              3.6122E-02 +\- 3.8010E-06 (   0.01%)
u-238        chi               -2.9925E-09 +\- 1.7124E-06 (-999.99%)
```

```
------------------------------------------------------------------------
   Energy and Region Integrated Sensitivity Coefficients for this Problem
------------------------------------------------------------------------

Mixture    Nuclide       Reaction       Sensitivity      Std. Dev. % Std. Dev.
------- ------------- ---------------   -----------     ---------- -----------
  1    o-16         total              1.1444E-02 +/- 3.3431E-05 (   0.29%)
  1    o-16         scatter            1.2013E-02 +/- 3.3421E-05 (   0.28%)
  1    o-16         elastic            1.1922E-02 +/- 3.3419E-05 (   0.28%)
  1    o-16         n,n'               8.7378E-05 +/- 1.9449E-07 (   0.22%)
  1    o-16         n,2n               1.0127E-11 +/- 1.6420E-20 (   0.00%)
  1    o-16         capture           -5.6947E-04 +/- 1.5760E-07 (   0.03%)
  1    o-16         n,gamma           -1.0052E-05 +/- 1.0479E-09 (   0.01%)
  1    o-16         n,p               -2.8703E-06 +/- 6.5195E-09 (   0.23%)
  1    o-16         n,d               -2.9590E-07 +/- 1.2700E-09 (   0.43%)
  1    o-16         n,t               -1.0698E-12 +/- 2.9755E-21 (   0.00%)
  1    o-16         n,alpha           -5.5625E-04 +/- 1.5540E-07 (   0.03%)
  1    u-234        total             -9.0979E-04 +/- 3.1204E-07 (   0.03%)
  1    u-234        scatter            6.2441E-06 +/- 1.2041E-08 (   0.19%)
  1    u-234        elastic            4.0083E-06 +/- 1.1854E-08 (   0.30%)
  1    u-234        n,n'               2.2050E-06 +/- 1.4630E-09 (   0.07%)
  1    u-234        n,2n               3.0794E-08 +/- 1.7472E-11 (   0.06%)
  1    u-234        fission            2.2909E-05 +/- 1.7241E-09 (   0.01%)
  1    u-234        capture           -9.3894E-04 +/- 3.0291E-07 (   0.03%)
  1    u-234        n,gamma           -9.3894E-04 +/- 3.0291E-07 (   0.03%)
  1    u-234        nubar              3.6021E-05 +/- 2.6348E-09 (   0.01%)
  1    u-234        chi                8.7265E-13 +/- 1.5382E-09 (-999.99%)
  1    u-235        total              2.0616E-01 +/- 7.0501E-05 (   0.03%)
```

```
1    u-235        scatter            8.8279E-04 +/- 1.0661E-06 (   0.12%)
1    u-235        elastic            3.1799E-04 +/- 9.6547E-07 (   0.30%)
1    u-235        n,n'               5.3198E-04 +/- 3.3888E-07 (   0.06%)
1    u-235        n,2n               3.2795E-05 +/- 1.0524E-08 (   0.03%)
1    u-235        fission            3.2433E-01 +/- 6.2460E-05 (   0.02%)
1    u-235        capture           -1.1906E-01 +/- 1.1810E-05 (   0.01%)
1    u-235        n,gamma           -1.1906E-01 +/- 1.1810E-05 (   0.01%)
1    u-235        nubar              9.6382E-01 +/- 8.0129E-05 (   0.01%)
1    u-235        chi               -6.6884E-08 +/- 4.6474E-05 (-999.99%)
1    u-236        total             -2.1805E-04 +/- 1.2725E-07 (   0.06%)
1    u-236        scatter            5.6471E-06 +/- 9.6565E-09 (   0.17%)
1    u-236        elastic            2.3929E-06 +/- 9.2543E-09 (   0.39%)
1    u-236        n,n'               3.1626E-06 +/- 2.0863E-09 (   0.07%)
1    u-236        n,2n               9.1002E-08 +/- 4.1257E-11 (   0.05%)
1    u-236        fission            1.2829E-05 +/- 1.7760E-09 (   0.01%)
1    u-236        capture           -2.3653E-04 +/- 1.2003E-07 (   0.05%)
1    u-236        n,gamma           -2.3653E-04 +/- 1.2003E-07 (   0.05%)
1    u-236        nubar              2.1111E-05 +/- 1.6677E-09 (   0.01%)
1    u-236        chi               -4.2095E-13 +/- 1.0077E-09 (-999.99%)
1    u-238        total             -5.9609E-02 +/- 2.6857E-05 (   0.05%)
1    u-238        scatter            2.9406E-02 +/- 2.0373E-05 (   0.07%)
1    u-238        elastic            1.3217E-02 +/- 1.6047E-05 (   0.12%)
```

(continues on next page)

(continued from previous page)

```
1    u-238       n,n'             1.5251E-02 +/- 9.7098E-06 (   0.06%)
1    u-238       n,2n             9.2840E-04 +/- 3.7033E-07 (   0.04%)
1    u-238       fission          2.4727E-02 +/- 2.4196E-06 (   0.01%)
1    u-238       capture         -1.1374E-01 +/- 9.9579E-06 (   0.01%)
1    u-238       n,gamma         -1.1374E-01 +/- 9.9579E-06 (   0.01%)
1    u-238       nubar            3.6122E-02 +/- 3.8010E-06 (   0.01%)
1    u-238       chi             -2.9925E-09 +/- 1.7124E-06 (-999.99%)
2    h-1         total            2.0159E-01 +/- 4.6480E-04 (   0.23%)
2    h-1         scatter          3.4592E-01 +/- 4.6062E-04 (   0.13%)
2    h-1         elastic          3.4592E-01 +/- 4.6062E-04 (   0.13%)
2    h-1         capture         -1.4433E-01 +/- 1.0386E-05 (   0.01%)
2    h-1         n,gamma         -1.4433E-01 +/- 1.0386E-05 (   0.01%)
2    o-16        total            3.6005E-02 +/- 2.9143E-05 (   0.08%)
2    o-16        scatter          3.7111E-02 +/- 2.9137E-05 (   0.08%)
2    o-16        elastic          3.6871E-02 +/- 2.9136E-05 (   0.08%)
2    o-16        n,n'             2.3245E-04 +/- 1.5061E-07 (   0.06%)
2    o-16        n,2n             2.3322E-11 +/- 1.2879E-17 (   0.00%)
2    o-16        capture         -1.1055E-03 +/- 1.4843E-07 (   0.01%)
2    o-16        n,gamma         -4.1305E-05 +/- 2.9719E-09 (   0.01%)
2    o-16        n,p             -5.3552E-06 +/- 5.0619E-09 (   0.09%)
2    o-16        n,d             -5.5135E-07 +/- 9.9279E-10 (   0.18%)
2    o-16        n,t             -2.0353E-12 +/- 2.3337E-18 (   0.00%)
2    o-16        n,alpha         -1.0583E-03 +/- 1.4700E-07 (   0.01%)
3    mg-24       total            4.5166E-05 +/- 4.5339E-07 (   1.00%)
3    mg-24       scatter          5.5339E-05 +/- 4.5356E-07 (   0.82%)
3    mg-24       elastic          4.3547E-05 +/- 4.5024E-07 (   1.03%)
```

```
3    mg-24       n,n'             1.1781E-05 +/- 3.3434E-08 (   0.28%)
3    mg-24       n,2n             5.8600E-13 +/- 1.0107E-22 (   0.00%)
3    mg-24       capture         -1.0173E-05 +/- 3.6669E-09 (   0.04%)
3    mg-24       n,gamma         -9.1039E-06 +/- 3.1240E-09 (   0.03%)
3    mg-24       n,p             -2.5459E-07 +/- 5.7541E-10 (   0.23%)
3    mg-24       n,alpha         -8.1441E-07 +/- 1.3622E-09 (   0.17%)
3    mg-25       total            1.2140E-06 +/- 3.2733E-08 (   2.70%)
3    mg-25       scatter          5.8458E-06 +/- 3.1369E-08 (   0.54%)
3    mg-25       elastic          4.2233E-06 +/- 3.0490E-08 (   0.72%)
3    mg-25       n,n'             1.5854E-06 +/- 3.7904E-09 (   0.24%)
3    mg-25       n,2n             3.6936E-08 +/- 9.6404E-11 (   0.26%)
3    mg-25       capture         -4.6318E-06 +/- 1.5204E-09 (   0.03%)
3    mg-25       n,gamma         -4.2701E-06 +/- 1.4848E-09 (   0.03%)
3    mg-25       n,p             -3.2369E-08 +/- 3.3458E-11 (   0.10%)
3    mg-25       n,alpha         -3.2935E-07 +/- 2.8796E-10 (   0.09%)
3    mg-26       total            5.2992E-06 +/- 3.2698E-08 (   0.62%)
3    mg-26       scatter          6.2529E-06 +/- 3.2364E-08 (   0.52%)
3    mg-26       elastic          4.8094E-06 +/- 3.1809E-08 (   0.66%)
3    mg-26       n,n'             1.4399E-06 +/- 3.4023E-09 (   0.24%)
3    mg-26       n,2n             3.6117E-09 +/- 2.0668E-18 (   0.00%)
3    mg-26       capture         -9.5364E-07 +/- 3.2481E-10 (   0.03%)
3    mg-26       n,gamma         -9.4910E-07 +/- 3.2481E-10 (   0.03%)
3    mg-26       n,p             -5.2381E-10 +/- 1.5687E-19 (   0.00%)
3    mg-26       n,alpha         -4.0194E-09 +/- 5.3317E-19 (   0.00%)
3    al-27       total           -6.7582E-05 +/- 2.3201E-05 (  34.33%)
3    al-27       scatter          4.6080E-03 +/- 2.2663E-05 (   0.49%)
3    al-27       elastic          3.2847E-03 +/- 2.1856E-05 (   0.67%)
3    al-27       n,n'             1.3189E-03 +/- 3.0167E-06 (   0.23%)
3    al-27       n,2n             5.9944E-08 +/- 1.4100E-09 (   2.35%)
3    al-27       capture         -4.6756E-03 +/- 1.6107E-06 (   0.03%)
3    al-27       n,gamma         -4.5949E-03 +/- 1.6079E-06 (   0.03%)
3    al-27       n,p             -6.7911E-05 +/- 6.9477E-08 (   0.10%)
3    al-27       n,d             -1.9733E-07 +/- 1.4218E-09 (   0.72%)
3    al-27       n,t             -6.6132E-09 +/- 5.0053E-16 (   0.00%)
3    al-27       n,alpha         -1.2531E-05 +/- 3.2986E-08 (   0.26%)
3    si-28       total            4.3107E-06 +/- 1.2244E-07 (   2.84%)
3    si-28       scatter          2.3404E-05 +/- 1.2022E-07 (   0.51%)
```

(continues on next page)

(continued from previous page)

```
3    si-28       elastic          1.8601E-05 +/- 1.1757E-07 (   0.63%)
3    si-28       n,n'             4.8005E-06 +/- 1.3001E-08 (   0.27%)
3    si-28       capture         -1.9094E-05 +/- 6.5542E-09 (   0.03%)
3    si-28       n,gamma         -1.8139E-05 +/- 6.3599E-09 (   0.04%)
3    si-28       n,p             -6.5750E-07 +/- 1.0533E-09 (   0.16%)
3    si-28       n,d             -8.8870E-10 +/- 3.5446E-19 (   0.00%)
3    si-28       n,alpha         -2.9630E-07 +/- 5.3824E-10 (   0.18%)
3    si-29       total            5.3687E-07 +/- 5.6554E-09 (   1.05%)
3    si-29       scatter          1.2496E-06 +/- 4.0360E-09 (   0.32%)
3    si-29       elastic          8.0857E-07 +/- 3.6454E-09 (   0.45%)
3    si-29       n,n'             4.3972E-07 +/- 9.9759E-10 (   0.23%)
```

```
3    si-29       n,2n             1.2847E-09 +/- 2.7509E-19 (   0.00%)
3    si-29       capture         -7.1270E-07 +/- 2.3029E-10 (   0.03%)
3    si-29       n,gamma         -6.6526E-07 +/- 2.2493E-10 (   0.03%)
3    si-29       n,p             -1.7830E-08 +/- 1.7962E-11 (   0.10%)
3    si-29       n,alpha         -2.9617E-08 +/- 2.2339E-11 (   0.08%)
3    si-30       total            1.3408E-07 +/- 4.4925E-09 (   3.35%)
3    si-30       scatter          7.3741E-07 +/- 3.2216E-09 (   0.44%)
3    si-30       elastic          5.4630E-07 +/- 3.1391E-09 (   0.57%)
3    si-30       n,n'             1.9023E-07 +/- 4.2667E-10 (   0.22%)
3    si-30       n,2n             8.7604E-10 +/- 2.6538E-19 (   0.00%)
3    si-30       capture         -6.0334E-07 +/- 8.1712E-10 (   0.14%)
3    si-30       n,gamma         -6.0283E-07 +/- 8.1712E-10 (   0.14%)
3    si-30       n,p             -9.2891E-11 +/- 2.3442E-20 (   0.00%)
3    si-30       n,alpha         -4.1193E-10 +/- 7.5633E-20 (   0.00%)
3    ti-46       total           -1.8856E-07 +/- 8.6800E-10 (   0.46%)
3    ti-46       scatter          2.1933E-07 +/- 4.7014E-10 (   0.21%)
3    ti-46       elastic          1.2896E-07 +/- 3.2795E-10 (   0.25%)
3    ti-46       n,n'             9.0312E-08 +/- 1.9591E-10 (   0.22%)
3    ti-46       n,2n             4.8498E-12 +/- 2.8827E-21 (   0.00%)
3    ti-46       capture         -4.0789E-07 +/- 1.3319E-10 (   0.03%)
3    ti-46       n,gamma         -3.9961E-07 +/- 1.3319E-10 (   0.03%)
3    ti-46       n,p             -7.7485E-09 +/- 3.8514E-18 (   0.00%)
3    ti-46       n,d             -3.1806E-13 +/- 2.3627E-22 (   0.00%)
3    ti-46       n,t             -9.9763E-17 +/- 3.2655E-26 (   0.00%)
3    ti-46       n,he-3          -4.2439E-16 +/- 1.9547E-25 (   0.00%)
3    ti-46       n,alpha         -5.2794E-10 +/- 2.5965E-19 (   0.00%)
3    ti-47       total           -8.7007E-07 +/- 2.4463E-09 (   0.28%)
3    ti-47       scatter          2.4268E-07 +/- 2.2853E-09 (   0.94%)
3    ti-47       elastic          1.3531E-07 +/- 2.2669E-09 (   1.68%)
3    ti-47       n,n'             1.0717E-07 +/- 1.8872E-10 (   0.18%)
3    ti-47       n,2n             1.9211E-10 +/- 4.0015E-20 (   0.00%)
3    ti-47       capture         -1.1128E-06 +/- 3.6552E-10 (   0.03%)
3    ti-47       n,gamma         -1.1003E-06 +/- 3.6552E-10 (   0.03%)
3    ti-47       n,p             -1.0646E-08 +/- 1.4197E-16 (   0.00%)
3    ti-47       n,d             -2.0197E-13 +/- 1.5135E-22 (   0.00%)
3    ti-47       n,t             -4.3601E-15 +/- 3.0922E-24 (   0.00%)
3    ti-47       n,he-3          -2.3300E-17 +/- 7.4079E-27 (   0.00%)
3    ti-47       n,alpha         -1.8482E-09 +/- 1.3222E-17 (   0.00%)
3    ti-48       total           -4.5957E-05 +/- 3.3263E-08 (   0.07%)
3    ti-48       scatter          3.0030E-06 +/- 2.3419E-08 (   0.78%)
3    ti-48       elastic          2.1566E-06 +/- 2.3249E-08 (   1.08%)
3    ti-48       n,n'             8.4526E-07 +/- 1.8984E-09 (   0.22%)
3    ti-48       n,2n             1.0593E-09 +/- 5.3938E-19 (   0.00%)
3    ti-48       capture         -4.8960E-05 +/- 1.7297E-08 (   0.04%)
3    ti-48       n,gamma         -4.8958E-05 +/- 1.7297E-08 (   0.04%)
3    ti-48       n,p             -1.6058E-09 +/- 9.4652E-19 (   0.00%)
3    ti-48       n,d             -1.6876E-12 +/- 1.6673E-21 (   0.00%)
3    ti-48       n,t             -8.0952E-16 +/- 2.7332E-25 (   0.00%)
```

```
3    ti-48       n,he-3          -1.9723E-17 +/- 6.7861E-27 (   0.00%)
3    ti-48       n,alpha         -1.5654E-10 +/- 4.0991E-20 (   0.00%)
3    ti-49       total           -6.8733E-07 +/- 6.2660E-10 (   0.09%)
```

(continues on next page)

(continued from previous page)

```
3    ti-49       scatter          1.6784E-07 +/- 2.2439E-09 (   1.34%)
3    ti-49       elastic          1.0265E-07 +/- 2.2349E-09 (   2.18%)
3    ti-49       n,n'             6.4463E-08 +/- 1.2032E-10 (   0.19%)
3    ti-49       n,2n             7.2472E-10 +/- 7.6807E-20 (   0.00%)
3    ti-49       capture         -8.5517E-07 +/- 2.9502E-10 (   0.03%)
3    ti-49       n,gamma         -8.5476E-07 +/- 2.9502E-10 (   0.03%)
3    ti-49       n,p             -3.3313E-10 +/- 3.2208E-19 (   0.00%)
3    ti-49       n,d             -7.4647E-14 +/- 6.7611E-23 (   0.00%)
3    ti-49       n,t             -2.1162E-15 +/- 9.3763E-25 (   0.00%)
3    ti-49       n,he-3          -9.3750E-20 +/- 0.0000E+00 (   0.00%)
3    ti-49       n,alpha         -7.1068E-11 +/- 4.3628E-20 (   0.00%)
3    ti-50       total            5.1758E-08 +/- 4.1741E-10 (   0.81%)
3    ti-50       scatter          1.3378E-07 +/- 1.9577E-10 (   0.15%)
3    ti-50       elastic          8.8832E-08 +/- 1.1755E-10 (   0.13%)
3    ti-50       n,n'             4.4731E-08 +/- 8.8290E-11 (   0.20%)
3    ti-50       n,2n             2.1770E-10 +/- 7.0963E-20 (   0.00%)
3    ti-50       capture         -8.2023E-08 +/- 2.5593E-11 (   0.03%)
3    ti-50       n,gamma         -8.2018E-08 +/- 2.5593E-11 (   0.03%)
3    ti-50       n,p             -2.9356E-12 +/- 9.4805E-22 (   0.00%)
3    ti-50       n,d             -1.7101E-14 +/- 1.1908E-23 (   0.00%)
3    ti-50       n,t             -1.0953E-17 +/- 2.8155E-27 (   0.00%)
3    ti-50       n,he-3          -4.0903E-23 +/- 0.0000E+00 (   0.00%)
3    ti-50       n,alpha         -1.7272E-12 +/- 6.5310E-22 (   0.00%)
3    cr-50       total           -1.3040E-05 +/- 9.1891E-09 (   0.07%)
3    cr-50       scatter          1.3376E-06 +/- 7.2895E-09 (   0.54%)
3    cr-50       elastic          1.2307E-06 +/- 7.2762E-09 (   0.59%)
3    cr-50       n,n'             1.0681E-07 +/- 2.3841E-10 (   0.22%)
3    cr-50       n,2n             3.1922E-12 +/- 2.3793E-21 (   0.00%)
3    cr-50       capture         -1.4377E-05 +/- 5.0788E-09 (   0.04%)
3    cr-50       n,gamma         -1.4333E-05 +/- 5.0787E-09 (   0.04%)
3    cr-50       n,p             -4.1672E-08 +/- 3.2114E-11 (   0.08%)
3    cr-50       n,d             -7.5849E-12 +/- 5.4355E-19 (   0.00%)
3    cr-50       n,alpha         -2.7989E-09 +/- 1.7847E-15 (   0.00%)
3    cr-52       total           -8.7501E-06 +/- 2.6630E-08 (   0.30%)
3    cr-52       scatter          4.4850E-06 +/- 2.1763E-08 (   0.49%)
3    cr-52       elastic          2.3489E-06 +/- 2.0602E-08 (   0.88%)
3    cr-52       n,n'             2.1346E-06 +/- 4.5774E-09 (   0.21%)
3    cr-52       n,2n             1.3180E-09 +/- 9.1937E-19 (   0.00%)
3    cr-52       capture         -1.3235E-05 +/- 4.5982E-09 (   0.03%)
3    cr-52       n,gamma         -1.3217E-05 +/- 4.5982E-09 (   0.03%)
3    cr-52       n,p             -1.5832E-08 +/- 1.8313E-11 (   0.12%)
3    cr-52       n,alpha         -1.9304E-09 +/- 3.8775E-17 (   0.00%)
3    cr-53       total           -3.4245E-05 +/- 2.1670E-08 (   0.06%)
3    cr-53       scatter          1.3865E-06 +/- 1.4475E-08 (   1.04%)
3    cr-53       elastic          1.0474E-06 +/- 1.4432E-08 (   1.38%)
```

```
3    cr-53       n,n'             3.3527E-07 +/- 7.1251E-10 (   0.21%)
3    cr-53       n,2n             3.8152E-09 +/- 3.6614E-17 (   0.00%)
3    cr-53       capture         -3.5632E-05 +/- 1.2596E-08 (   0.04%)
3    cr-53       n,gamma         -3.5630E-05 +/- 1.2596E-08 (   0.04%)
3    cr-53       n,p             -6.9509E-10 +/- 5.2410E-17 (   0.00%)
3    cr-53       n,alpha         -1.5386E-09 +/- 1.4981E-16 (   0.00%)
3    cr-54       total           -1.6695E-08 +/- 4.1104E-10 (   2.46%)
3    cr-54       scatter          1.6318E-07 +/- 2.3409E-10 (   0.14%)
3    cr-54       elastic          9.3303E-08 +/- 7.9919E-11 (   0.09%)
3    cr-54       n,n'             6.9373E-08 +/- 1.3815E-10 (   0.20%)
3    cr-54       n,2n             5.0486E-10 +/- 8.3567E-20 (   0.00%)
3    cr-54       capture         -1.7988E-07 +/- 5.9710E-11 (   0.03%)
3    cr-54       n,gamma         -1.7985E-07 +/- 5.9710E-11 (   0.03%)
3    cr-54       n,p             -4.2582E-12 +/- 1.1391E-21 (   0.00%)
3    cr-54       n,alpha         -2.1413E-11 +/- 1.0788E-20 (   0.00%)
3    mn-55       total           -9.6360E-05 +/- 1.4079E-07 (   0.15%)
3    mn-55       scatter          6.8704E-06 +/- 1.3262E-07 (   1.93%)
3    mn-55       elastic          5.7292E-06 +/- 1.3254E-07 (   2.31%)
```

(continues on next page)

(continued from previous page)

```
3    mn-55       n,n'             1.1366E-06 +/- 2.5436E-09 (   0.22%)
3    mn-55       n,2n             4.4184E-09 +/- 1.0180E-18 (   0.00%)
3    mn-55       capture         -1.0323E-04 +/- 3.5222E-08 (   0.03%)
3    mn-55       n,gamma         -1.0323E-04 +/- 3.5222E-08 (   0.03%)
3    mn-55       n,p             -3.7300E-09 +/- 9.1810E-18 (   0.00%)
3    mn-55       n,d             -3.5791E-11 +/- 6.7992E-21 (   0.00%)
3    mn-55       n,t             -1.3299E-12 +/- 1.2646E-21 (   0.00%)
3    mn-55       n,he-3          -4.6031E-14 +/- 4.8445E-23 (   0.00%)
3    mn-55       n,alpha         -8.3737E-10 +/- 6.0947E-19 (   0.00%)
3    fe-54       total           -3.9407E-06 +/- 8.5216E-09 (   0.22%)
3    fe-54       scatter          7.5145E-07 +/- 8.1027E-09 (   1.08%)
3    fe-54       elastic          5.5492E-07 +/- 8.0723E-09 (   1.45%)
3    fe-54       n,n'             1.9601E-07 +/- 4.3649E-10 (   0.22%)
3    fe-54       n,2n             3.7980E-12 +/- 1.3165E-21 (   0.00%)
3    fe-54       capture         -4.6921E-06 +/- 1.5839E-09 (   0.03%)
3    fe-54       n,gamma         -4.5400E-06 +/- 1.5807E-09 (   0.03%)
3    fe-54       n,p             -1.5046E-07 +/- 9.9241E-11 (   0.07%)
3    fe-54       n,d             -2.5789E-11 +/- 3.7341E-21 (   0.00%)
3    fe-54       n,alpha         -1.5839E-09 +/- 6.7397E-19 (   0.00%)
3    fe-56       total           -6.9426E-05 +/- 1.3507E-07 (   0.19%)
3    fe-56       scatter          1.1537E-05 +/- 1.1066E-07 (   0.96%)
3    fe-56       elastic          7.5606E-06 +/- 1.0976E-07 (   1.45%)
3    fe-56       n,n'             3.9702E-06 +/- 9.5385E-09 (   0.24%)
3    fe-56       n,2n             5.5596E-09 +/- 3.1334E-18 (   0.00%)
3    fe-56       capture         -8.0963E-05 +/- 2.8413E-08 (   0.04%)
3    fe-56       n,gamma         -8.0923E-05 +/- 2.8413E-08 (   0.04%)
3    fe-56       n,p             -2.8231E-08 +/- 4.3821E-11 (   0.16%)
3    fe-56       n,d             -6.3061E-11 +/- 4.9191E-20 (   0.00%)
3    fe-56       n,t             -1.5472E-13 +/- 1.2160E-22 (   0.00%)
3    fe-56       n,he-3          -1.7440E-14 +/- 1.2169E-23 (   0.00%)
```

```
3    fe-56       n,alpha         -1.1945E-08 +/- 1.2530E-11 (   0.10%)
3    fe-57       total           -1.4580E-06 +/- 2.4578E-09 (   0.17%)
3    fe-57       scatter          2.9453E-07 +/- 2.2278E-09 (   0.76%)
3    fe-57       elastic          1.6082E-07 +/- 2.1139E-09 (   1.31%)
3    fe-57       n,n'             1.3172E-07 +/- 2.4140E-10 (   0.18%)
3    fe-57       n,2n             1.9852E-09 +/- 1.4642E-19 (   0.00%)
3    fe-57       capture         -1.7526E-06 +/- 6.0368E-10 (   0.03%)
3    fe-57       n,gamma         -1.7515E-06 +/- 6.0368E-10 (   0.03%)
3    fe-57       n,p             -4.5690E-10 +/- 1.9448E-19 (   0.00%)
3    fe-57       n,alpha         -6.5570E-10 +/- 3.5601E-19 (   0.00%)
3    fe-58       total           -8.6249E-08 +/- 9.2863E-11 (   0.11%)
3    fe-58       scatter          3.2882E-08 +/- 2.1096E-13 (   0.00%)
3    fe-58       elastic          1.9457E-08 +/- 2.1096E-13 (   0.00%)
3    fe-58       n,n'             1.3343E-08 +/- 1.7621E-16 (   0.00%)
3    fe-58       n,2n             8.1702E-11 +/- 1.5798E-20 (   0.00%)
3    fe-58       capture         -1.1913E-07 +/- 3.5967E-11 (   0.03%)
3    fe-58       n,gamma         -1.1912E-07 +/- 3.5967E-11 (   0.03%)
3    fe-58       n,p             -8.9556E-13 +/- 2.0142E-22 (   0.00%)
3    fe-58       n,alpha         -5.5186E-12 +/- 1.4040E-21 (   0.00%)
3    cu-63       total           -9.5973E-05 +/- 7.0063E-08 (   0.07%)
3    cu-63       scatter          8.9795E-06 +/- 3.9903E-08 (   0.44%)
3    cu-63       elastic          5.1356E-06 +/- 3.8382E-08 (   0.75%)
3    cu-63       n,n'             3.8154E-06 +/- 7.8316E-09 (   0.21%)
3    cu-63       n,2n             4.8437E-09 +/- 1.9022E-18 (   0.00%)
3    cu-63       capture         -1.0495E-04 +/- 3.5784E-08 (   0.03%)
3    cu-63       n,gamma         -1.0450E-04 +/- 3.5783E-08 (   0.03%)
3    cu-63       n,p             -4.3880E-07 +/- 2.3679E-10 (   0.05%)
3    cu-63       n,d             -7.9573E-09 +/- 9.1280E-17 (   0.00%)
3    cu-63       n,he-3          -3.8432E-14 +/- 1.6090E-23 (   0.00%)
3    cu-63       n,alpha         -1.0048E-08 +/- 7.9401E-17 (   0.00%)
3    cu-65       total           -1.7689E-05 +/- 4.4209E-08 (   0.25%)
3    cu-65       scatter          4.5920E-06 +/- 3.3439E-08 (   0.73%)
3    cu-65       elastic          2.9304E-06 +/- 3.3117E-08 (   1.13%)
```

(continues on next page)

(continued from previous page)

3	cu-65	n,n'	1.6531E-06 +/- 3.3430E-09	(0.20%)
3	cu-65	n,2n	8.2988E-09 +/- 2.5459E-11	(0.31%)
3	cu-65	capture	-2.2281E-05 +/- 7.6516E-09	(0.03%)
3	cu-65	n,gamma	-2.2276E-05 +/- 7.6516E-09	(0.03%)
3	cu-65	n,p	-4.8468E-09 +/- 2.3482E-16	(0.00%)
3	cu-65	n,d	-5.9497E-10 +/- 1.8412E-17	(0.00%)
3	cu-65	n,t	-1.3215E-13 +/- 7.8888E-23	(0.00%)
3	cu-65	n,he-3	-4.7968E-15 +/- 6.6710E-24	(0.00%)
3	cu-65	n,alpha	-1.2252E-10 +/- 1.7122E-18	(0.00%)
4	h-1	total	-3.3487E-05 +/- 3.3383E-05	(99.69%)
4	h-1	scatter	1.1785E-04 +/- 3.3082E-05	(28.07%)
4	h-1	elastic	1.1785E-04 +/- 3.3082E-05	(28.07%)
4	h-1	capture	-1.5134E-04 +/- 3.4431E-07	(0.23%)
4	h-1	n,gamma	-1.5134E-04 +/- 3.4431E-07	(0.23%)
4	c	total	1.4324E-04 +/- 3.4393E-06	(2.40%)
4	c	scatter	1.4444E-04 +/- 3.4379E-06	(2.38%)
4	c	elastic	1.4356E-04 +/- 3.4374E-06	(2.39%)
4	c	n,n'	8.7941E-07 +/- 2.1229E-08	(2.41%)
4	c	capture	-1.2004E-06 +/- 5.0249E-09	(0.42%)
4	c	n,gamma	-1.1534E-06 +/- 2.6118E-09	(0.23%)
4	c	n,p	-1.2812E-11 +/- 0.0000E+00	(0.00%)
4	c	n,d	-3.3016E-11 +/- 0.0000E+00	(0.00%)
4	c	n,alpha	-4.6915E-08 +/- 4.2884E-09	(9.14%)
4	o-16	total	3.9480E-05 +/- 8.0523E-07	(2.04%)
4	o-16	scatter	3.9628E-05 +/- 8.0503E-07	(2.03%)
4	o-16	elastic	3.9526E-05 +/- 8.0498E-07	(2.04%)
4	o-16	n,n'	1.0040E-07 +/- 3.9571E-09	(3.94%)
4	o-16	n,2n	5.9840E-15 +/- 0.0000E+00	(0.00%)
4	o-16	capture	-1.4730E-07 +/- 2.8085E-09	(1.91%)
4	o-16	n,gamma	-1.8551E-08 +/- 3.2427E-11	(0.17%)
4	o-16	n,p	-4.6597E-10 +/- 2.6874E-13	(0.06%)
4	o-16	n,d	-4.2225E-11 +/- 1.3904E-14	(0.03%)
4	o-16	n,t	-3.5750E-16 +/- 0.0000E+00	(0.00%)
4	o-16	n,alpha	-1.2824E-07 +/- 2.8083E-09	(2.19%)
4	si-28	total	1.5659E-07 +/- 3.1168E-09	(1.99%)
4	si-28	scatter	2.7569E-07 +/- 2.1958E-09	(0.80%)
4	si-28	elastic	2.4930E-07 +/- 2.0614E-09	(0.83%)
4	si-28	n,n'	2.6386E-08 +/- 2.5243E-10	(0.96%)
4	si-28	capture	-1.1911E-07 +/- 2.5877E-10	(0.22%)
4	si-28	n,gamma	-1.1814E-07 +/- 2.5877E-10	(0.22%)
4	si-28	n,p	-6.7032E-10 +/- 6.8669E-17	(0.00%)
4	si-28	n,d	-7.0401E-13 +/- 0.0000E+00	(0.00%)
4	si-28	n,alpha	-2.9677E-10 +/- 2.4154E-17	(0.00%)
4	si-29	total	8.3353E-09 +/- 7.5001E-15	(0.00%)
4	si-29	scatter	1.2660E-08 +/- 7.4989E-15	(0.00%)
4	si-29	elastic	1.0360E-08 +/- 7.1408E-15	(0.00%)
4	si-29	n,n'	2.2970E-09 +/- 7.6907E-16	(0.00%)
4	si-29	n,2n	3.6953E-12 +/- 0.0000E+00	(0.00%)
4	si-29	capture	-4.3250E-09 +/- 5.9139E-18	(0.00%)
4	si-29	n,gamma	-4.2731E-09 +/- 1.0615E-18	(0.00%)
4	si-29	n,p	-1.8703E-11 +/- 2.1885E-18	(0.00%)
4	si-29	n,alpha	-3.3206E-11 +/- 3.9362E-18	(0.00%)
4	si-30	total	5.0684E-09 +/- 6.4263E-15	(0.00%)
4	si-30	scatter	8.0334E-09 +/- 6.4252E-15	(0.00%)
4	si-30	elastic	7.1395E-09 +/- 6.3753E-15	(0.00%)
4	si-30	n,n'	8.9160E-10 +/- 2.1621E-16	(0.00%)
4	si-30	n,2n	2.2872E-12 +/- 0.0000E+00	(0.00%)
4	si-30	capture	-2.9651E-09 +/- 1.4537E-18	(0.00%)
4	si-30	n,gamma	-2.9646E-09 +/- 1.4536E-18	(0.00%)
4	si-30	n,p	-7.7936E-14 +/- 3.6748E-26	(0.00%)
4	si-30	n,alpha	-3.6427E-13 +/- 5.9486E-21	(0.00%)
4	s-32	total	-9.3027E-07 +/- 1.5023E-08	(1.61%)
4	s-32	scatter	1.0965E-06 +/- 1.1617E-08	(1.06%)

```
4    s-32        elastic          1.0303E-06 +/- 1.1393E-08 (   1.11%)
4    s-32        n,n'             6.5967E-08 +/- 8.6196E-10 (   1.31%)
4    s-32        n,2n             1.0859E-14 +/- 0.0000E+00 (   0.00%)
4    s-32        capture         -2.0268E-06 +/- 4.4706E-09 (   0.22%)
4    s-32        n,gamma         -1.8835E-06 +/- 4.2718E-09 (   0.23%)
4    s-32        n,p             -4.4415E-08 +/- 5.3423E-10 (   1.20%)
4    s-32        n,alpha         -9.8944E-08 +/- 5.3326E-10 (   0.54%)
4    s-33        total           -1.1925E-08 +/- 1.0473E-14 (   0.00%)
4    s-33        scatter          7.9053E-09 +/- 9.6281E-15 (   0.00%)
4    s-33        elastic          7.1213E-09 +/- 9.5751E-15 (   0.00%)
4    s-33        n,n'             7.7770E-10 +/- 1.7507E-16 (   0.00%)
4    s-33        n,2n             4.8458E-12 +/- 0.0000E+00 (   0.00%)
4    s-33        capture         -1.9830E-08 +/- 8.4932E-16 (   0.00%)
4    s-33        n,gamma         -9.8480E-09 +/- 1.5761E-18 (   0.00%)
4    s-33        n,p             -4.3988E-10 +/- 9.9510E-17 (   0.00%)
4    s-33        n,alpha         -9.5421E-09 +/- 8.0138E-16 (   0.00%)
4    s-34        total            1.2596E-08 +/- 3.9241E-10 (   3.12%)
4    s-34        scatter          4.7978E-08 +/- 1.6902E-10 (   0.35%)
4    s-34        elastic          4.3911E-08 +/- 1.1281E-10 (   0.26%)
4    s-34        n,n'             4.0515E-09 +/- 7.8532E-16 (   0.00%)
4    s-34        n,2n             6.4633E-12 +/- 0.0000E+00 (   0.00%)
4    s-34        capture         -3.5381E-08 +/- 6.1670E-11 (   0.17%)
4    s-34        n,gamma         -3.5292E-08 +/- 6.1670E-11 (   0.17%)
4    s-34        n,p             -1.6841E-11 +/- 1.4263E-18 (   0.00%)
4    s-34        n,alpha         -7.2939E-11 +/- 9.4585E-18 (   0.00%)
4    s-36        total            1.2812E-10 +/- 1.6526E-16 (   0.00%)
4    s-36        scatter          2.4510E-10 +/- 1.6524E-16 (   0.00%)
4    s-36        elastic          2.3204E-10 +/- 1.6517E-16 (   0.00%)
4    s-36        n,n'             1.2864E-11 +/- 1.1331E-18 (   0.00%)
4    s-36        n,2n             1.9756E-13 +/- 0.0000E+00 (   0.00%)
4    s-36        capture         -1.1697E-10 +/- 2.0921E-20 (   0.00%)
4    s-36        n,gamma         -1.1697E-10 +/- 2.0919E-20 (   0.00%)
4    s-36        n,p             -5.8200E-17 +/- 0.0000E+00 (   0.00%)
4    s-36        n,alpha         -5.1249E-15 +/- 2.0171E-22 (   0.00%)
4    ca-40       total           -1.7949E-06 +/- 1.1894E-07 (   6.63%)
4    ca-40       scatter          6.7330E-06 +/- 9.7909E-08 (   1.45%)
4    ca-40       elastic          6.5226E-06 +/- 9.7589E-08 (   1.50%)
4    ca-40       n,n'             2.0785E-07 +/- 3.3003E-09 (   1.59%)
4    ca-40       n,2n             4.5607E-14 +/- 0.0000E+00 (   0.00%)
4    ca-40       capture         -8.5279E-06 +/- 1.9716E-08 (   0.23%)
4    ca-40       n,gamma         -7.9727E-06 +/- 1.8069E-08 (   0.23%)
4    ca-40       n,p             -3.7411E-07 +/- 5.5225E-09 (   1.48%)
4    ca-40       n,d             -5.9766E-11 +/- 6.5478E-19 (   0.00%)
4    ca-40       n,t             -4.2787E-15 +/- 0.0000E+00 (   0.00%)
4    ca-40       n,he-3          -5.7096E-14 +/- 0.0000E+00 (   0.00%)
4    ca-40       n,alpha         -1.8104E-07 +/- 2.0743E-09 (   1.15%)
4    ca-42       total           -3.3798E-08 +/- 4.2370E-10 (   1.25%)
4    ca-42       scatter          5.5595E-08 +/- 1.8840E-10 (   0.34%)
```

```
4    ca-42       elastic          4.4749E-08 +/- 7.7226E-14 (   0.00%)
4    ca-42       n,n'             1.0831E-08 +/- 3.5480E-15 (   0.00%)
4    ca-42       n,2n             4.2807E-12 +/- 0.0000E+00 (   0.00%)
4    ca-42       capture         -8.9393E-08 +/- 1.9154E-10 (   0.21%)
4    ca-42       n,gamma         -8.9285E-08 +/- 1.9154E-10 (   0.21%)
4    ca-42       n,p             -6.3001E-11 +/- 6.1936E-18 (   0.00%)
4    ca-42       n,d             -3.9571E-14 +/- 0.0000E+00 (   0.00%)
4    ca-42       n,t             -1.1401E-16 +/- 0.0000E+00 (   0.00%)
4    ca-42       n,he-3          -2.8628E-18 +/- 0.0000E+00 (   0.00%)
4    ca-42       n,alpha         -4.4886E-11 +/- 4.9613E-18 (   0.00%)
4    ca-43       total           -2.9992E-07 +/- 8.5309E-10 (   0.28%)
4    ca-43       scatter          1.7308E-08 +/- 9.3864E-15 (   0.00%)
4    ca-43       elastic          1.4035E-08 +/- 8.9886E-15 (   0.00%)
4    ca-43       n,n'             3.2593E-09 +/- 1.0415E-15 (   0.00%)
4    ca-43       n,2n             1.3992E-11 +/- 1.0619E-22 (   0.00%)
4    ca-43       capture         -3.1723E-07 +/- 7.1329E-10 (   0.22%)
```

(continues on next page)

(continued from previous page)

```
4    ca-43       n,gamma          -3.1720E-07 +/- 7.1328E-10 (   0.22%)
4    ca-43       n,p              -1.2736E-11 +/- 1.4953E-18 (   0.00%)
4    ca-43       n,d              -3.9211E-15 +/- 0.0000E+00 (   0.00%)
4    ca-43       n,t              -2.0903E-16 +/- 0.0000E+00 (   0.00%)
4    ca-43       n,he-3           -8.8381E-21 +/- 0.0000E+00 (   0.00%)
4    ca-43       n,alpha          -1.8078E-11 +/- 1.6658E-18 (   0.00%)
4    ca-44       total            -1.9121E-07 +/- 2.9350E-09 (   1.53%)
4    ca-44       scatter           1.8201E-07 +/- 9.8994E-10 (   0.54%)
4    ca-44       elastic           1.4003E-07 +/- 6.5307E-10 (   0.47%)
4    ca-44       n,n'              4.1934E-08 +/- 3.9319E-10 (   0.94%)
4    ca-44       n,2n              3.7701E-11 +/- 0.0000E+00 (   0.00%)
4    ca-44       capture          -3.7322E-07 +/- 8.4317E-10 (   0.23%)
4    ca-44       n,gamma          -3.7321E-07 +/- 8.4317E-10 (   0.23%)
4    ca-44       n,p              -4.3462E-12 +/- 1.1952E-19 (   0.00%)
4    ca-44       n,d              -1.9015E-14 +/- 0.0000E+00 (   0.00%)
4    ca-44       n,t              -3.6521E-17 +/- 0.0000E+00 (   0.00%)
4    ca-44       n,he-3           -5.7133E-22 +/- 0.0000E+00 (   0.00%)
4    ca-44       n,alpha          -2.1833E-12 +/- 5.2524E-20 (   0.00%)
4    ca-46       total            -9.1789E-11 +/- 2.9314E-16 (   0.00%)
4    ca-46       scatter           5.0404E-10 +/- 2.9308E-16 (   0.00%)
4    ca-46       elastic           4.3313E-10 +/- 2.8501E-16 (   0.00%)
4    ca-46       n,n'              7.0696E-11 +/- 2.2304E-17 (   0.00%)
4    ca-46       n,2n              2.0718E-13 +/- 0.0000E+00 (   0.00%)
4    ca-46       capture          -5.9583E-10 +/- 5.9929E-20 (   0.00%)
4    ca-46       n,gamma          -5.9583E-10 +/- 5.9929E-20 (   0.00%)
4    ca-46       n,p              -3.2267E-16 +/- 3.6714E-26 (   0.00%)
4    ca-46       n,d              -6.8722E-18 +/- 0.0000E+00 (   0.00%)
4    ca-46       n,t              -3.3633E-21 +/- 0.0000E+00 (   0.00%)
4    ca-46       n,alpha          -2.2633E-17 +/- 0.0000E+00 (   0.00%)
4    ca-48       total            -2.4743E-08 +/- 1.8724E-10 (   0.76%)
4    ca-48       scatter           1.6341E-08 +/- 8.2057E-15 (   0.00%)
4    ca-48       elastic           1.5512E-08 +/- 8.2022E-15 (   0.00%)
```

```
4    ca-48       n,n'              8.1356E-10 +/- 6.9284E-17 (   0.00%)
4    ca-48       n,2n              1.5225E-11 +/- 0.0000E+00 (   0.00%)
4    ca-48       capture          -4.1084E-08 +/- 7.9141E-11 (   0.19%)
4    ca-48       n,gamma          -4.1084E-08 +/- 7.9141E-11 (   0.19%)
4    ca-48       n,p              -1.0552E-16 +/- 0.0000E+00 (   0.00%)
4    ca-48       n,d              -2.1494E-17 +/- 0.0000E+00 (   0.00%)
4    ca-48       n,t              -5.0004E-22 +/- 0.0000E+00 (   0.00%)
4    ca-48       n,alpha          -1.9945E-17 +/- 0.0000E+00 (   0.00%)
5    h-1         total            -4.6343E-05 +/- 2.5632E-05 (  55.31%)
5    h-1         scatter           1.4599E-04 +/- 2.5374E-05 (  17.38%)
5    h-1         elastic           1.4599E-04 +/- 2.5374E-05 (  17.38%)
5    h-1         capture          -1.9233E-04 +/- 2.9929E-07 (   0.16%)
5    h-1         n,gamma          -1.9233E-04 +/- 2.9929E-07 (   0.16%)
5    c           total             1.5186E-04 +/- 2.0605E-06 (   1.36%)
5    c           scatter           1.5313E-04 +/- 1.9864E-06 (   1.30%)
5    c           elastic           1.5229E-04 +/- 1.9863E-06 (   1.30%)
5    c           n,n'              8.3833E-07 +/- 7.6391E-09 (   0.91%)
5    c           capture          -1.2702E-06 +/- 2.4828E-09 (   0.20%)
5    c           n,gamma          -1.2318E-06 +/- 1.9115E-09 (   0.16%)
5    c           n,p              -9.4682E-12 +/- 0.0000E+00 (   0.00%)
5    c           n,d              -2.8081E-11 +/- 0.0000E+00 (   0.00%)
5    c           n,alpha          -3.8362E-08 +/- 1.5828E-09 (   4.13%)
5    o-16        total             5.7694E-05 +/- 6.3654E-07 (   1.10%)
5    o-16        scatter           5.7871E-05 +/- 6.3645E-07 (   1.10%)
5    o-16        elastic           5.7729E-05 +/- 6.3644E-07 (   1.10%)
5    o-16        n,n'              1.3911E-07 +/- 2.0477E-09 (   1.47%)
5    o-16        n,2n              1.1747E-14 +/- 0.0000E+00 (   0.00%)
5    o-16        capture          -1.7668E-07 +/- 1.5046E-09 (   0.85%)
5    o-16        n,gamma          -2.7736E-08 +/- 3.4230E-11 (   0.12%)
5    o-16        n,p              -6.3458E-10 +/- 1.2415E-13 (   0.02%)
5    o-16        n,d              -5.2519E-11 +/- 6.4231E-15 (   0.01%)
```

(continues on next page)

(continued from previous page)

```
5    o-16        n,t             -4.4058E-16 +/- 0.0000E+00 (   0.00%)
5    o-16        n,alpha         -1.4826E-07 +/- 1.4006E-09 (   0.94%)
6    c           total            9.7668E-06 +/- 1.3853E-07 (   1.42%)
6    c           scatter          9.8750E-06 +/- 1.3850E-07 (   1.40%)
6    c           elastic          9.8360E-06 +/- 1.3846E-07 (   1.41%)
6    c           n,n'             3.8987E-08 +/- 1.2258E-09 (   3.14%)
6    c           capture         -1.0818E-07 +/- 2.4021E-10 (   0.22%)
6    c           n,gamma         -7.0538E-08 +/- 5.6578E-11 (   0.08%)
6    c           n,p             -6.2023E-12 +/- 0.0000E+00 (   0.00%)
6    c           n,d             -1.0588E-11 +/- 0.0000E+00 (   0.00%)
6    c           n,alpha         -3.7621E-08 +/- 2.3320E-10 (   0.62%)
6    o-16        total            6.9207E-07 +/- 7.5312E-09 (   1.09%)
6    o-16        scatter          7.0981E-07 +/- 7.5282E-09 (   1.06%)
6    o-16        elastic          7.0868E-07 +/- 7.5282E-09 (   1.06%)
6    o-16        n,n'             1.0108E-09 +/- 2.8433E-16 (   0.00%)
6    o-16        n,2n             6.5082E-17 +/- 0.0000E+00 (   0.00%)
6    o-16        capture         -1.7739E-08 +/- 2.1801E-11 (   0.12%)
```

```
6    o-16        n,gamma         -2.6146E-10 +/- 2.2204E-17 (   0.00%)
6    o-16        n,p             -1.0955E-10 +/- 1.7204E-17 (   0.00%)
6    o-16        n,d             -1.2082E-11 +/- 8.9008E-19 (   0.00%)
6    o-16        n,t             -2.2926E-17 +/- 0.0000E+00 (   0.00%)
6    o-16        n,alpha         -1.7356E-08 +/- 2.1801E-11 (   0.13%)
6    na-23       total            2.3175E-08 +/- 1.4447E-09 (   6.23%)
6    na-23       scatter          5.7809E-08 +/- 1.4371E-09 (   2.49%)
6    na-23       elastic          5.2606E-08 +/- 1.4371E-09 (   2.73%)
6    na-23       n,n'             5.2023E-09 +/- 7.8251E-15 (   0.00%)
6    na-23       n,2n             4.5318E-13 +/- 0.0000E+00 (   0.00%)
6    na-23       capture         -3.4634E-08 +/- 2.0188E-11 (   0.06%)
6    na-23       n,gamma         -3.4409E-08 +/- 2.0188E-11 (   0.06%)
6    na-23       n,p             -1.5160E-10 +/- 3.2974E-18 (   0.00%)
6    na-23       n,alpha         -7.2671E-11 +/- 5.0907E-18 (   0.00%)
6    mg-24       total            3.2809E-08 +/- 5.8412E-12 (   0.02%)
6    mg-24       scatter          3.5854E-08 +/- 5.8398E-12 (   0.02%)
6    mg-24       elastic          3.3595E-08 +/- 5.8398E-12 (   0.02%)
6    mg-24       n,n'             2.2531E-09 +/- 3.1539E-16 (   0.00%)
6    mg-24       n,2n             4.2480E-17 +/- 0.0000E+00 (   0.00%)
6    mg-24       capture         -3.0449E-09 +/- 2.0795E-15 (   0.00%)
6    mg-24       n,gamma         -2.5024E-09 +/- 2.0794E-15 (   0.00%)
6    mg-24       n,p             -1.3151E-10 +/- 4.3984E-18 (   0.00%)
6    mg-24       n,alpha         -4.1094E-10 +/- 8.3640E-18 (   0.00%)
6    mg-25       total            2.2861E-09 +/- 1.0962E-12 (   0.05%)
6    mg-25       scatter          3.6225E-09 +/- 1.0956E-12 (   0.03%)
6    mg-25       elastic          3.3170E-09 +/- 1.0956E-12 (   0.03%)
6    mg-25       n,n'             2.9207E-10 +/- 1.2629E-16 (   0.00%)
6    mg-25       n,2n             1.3416E-11 +/- 1.1115E-18 (   0.00%)
6    mg-25       capture         -1.3364E-09 +/- 7.2936E-16 (   0.00%)
6    mg-25       n,gamma         -1.1676E-09 +/- 7.2936E-16 (   0.00%)
6    mg-25       n,p             -1.5856E-11 +/- 1.8444E-19 (   0.00%)
6    mg-25       n,alpha         -1.5294E-10 +/- 1.4511E-18 (   0.00%)
6    mg-26       total            3.6092E-09 +/- 4.4230E-13 (   0.01%)
6    mg-26       scatter          3.8717E-09 +/- 4.4228E-13 (   0.01%)
6    mg-26       elastic          3.6643E-09 +/- 4.4228E-13 (   0.01%)
6    mg-26       n,n'             2.0617E-10 +/- 2.7499E-17 (   0.00%)
6    mg-26       n,2n             1.2053E-12 +/- 6.0525E-20 (   0.00%)
6    mg-26       capture         -2.6241E-10 +/- 2.5730E-17 (   0.00%)
6    mg-26       n,gamma         -2.5990E-10 +/- 2.5729E-17 (   0.00%)
6    mg-26       n,p             -3.0666E-13 +/- 3.9578E-20 (   0.00%)
6    mg-26       n,alpha         -2.2079E-12 +/- 2.2172E-19 (   0.00%)
6    si-28       total            1.7607E-07 +/- 1.6638E-09 (   0.94%)
6    si-28       scatter          2.6676E-07 +/- 1.5896E-09 (   0.60%)
6    si-28       elastic          2.5293E-07 +/- 1.5241E-09 (   0.60%)
6    si-28       n,n'             1.3815E-08 +/- 1.7317E-15 (   0.00%)
6    si-28       capture         -9.0687E-08 +/- 6.7580E-11 (   0.07%)
```

(continues on next page)

(continued from previous page)

```
6    si-28       n,gamma          -8.2663E-08 +/- 6.7580E-11 (   0.08%)
6    si-28       n,p              -5.5220E-09 +/- 1.0643E-16 (   0.00%)
```

```
6    si-28       n,d              -8.9972E-12 +/- 1.1284E-18 (   0.00%)
6    si-28       n,alpha          -2.4934E-09 +/- 5.7790E-17 (   0.00%)
6    si-29       total             7.3370E-09 +/- 1.5194E-12 (   0.02%)
6    si-29       scatter           1.0779E-08 +/- 1.5176E-12 (   0.01%)
6    si-29       elastic           9.6163E-09 +/- 1.5176E-12 (   0.02%)
6    si-29       n,n'              1.1542E-09 +/- 1.7759E-16 (   0.00%)
6    si-29       n,2n              8.0511E-12 +/- 8.0235E-19 (   0.00%)
6    si-29       capture          -3.4416E-09 +/- 7.1336E-15 (   0.00%)
6    si-29       n,gamma          -3.0602E-09 +/- 7.1336E-15 (   0.00%)
6    si-29       n,p              -1.4767E-10 +/- 2.5051E-18 (   0.00%)
6    si-29       n,alpha          -2.3373E-10 +/- 3.3600E-18 (   0.00%)
6    si-30       total             4.2864E-09 +/- 9.7142E-13 (   0.02%)
6    si-30       scatter           7.1185E-09 +/- 9.6378E-13 (   0.01%)
6    si-30       elastic           6.7125E-09 +/- 9.6378E-13 (   0.01%)
6    si-30       n,n'              4.0116E-10 +/- 5.5897E-17 (   0.00%)
6    si-30       n,2n              4.8546E-12 +/- 4.4990E-19 (   0.00%)
6    si-30       capture          -2.8321E-09 +/- 1.8920E-14 (   0.00%)
6    si-30       n,gamma          -2.8275E-09 +/- 1.8920E-14 (   0.00%)
6    si-30       n,p              -9.0348E-13 +/- 1.2153E-19 (   0.00%)
6    si-30       n,alpha          -3.7528E-12 +/- 3.1321E-19 (   0.00%)
6    s-32        total            -1.4228E-08 +/- 1.7286E-12 (   0.01%)
6    s-32        scatter           1.9787E-08 +/- 1.7227E-12 (   0.01%)
6    s-32        elastic           1.9229E-08 +/- 1.7227E-12 (   0.01%)
6    s-32        n,n'              5.4483E-10 +/- 8.3488E-17 (   0.00%)
6    s-32        n,2n              9.4026E-17 +/- 0.0000E+00 (   0.00%)
6    s-32        capture          -3.4015E-08 +/- 9.3425E-15 (   0.00%)
6    s-32        n,gamma          -2.3112E-08 +/- 3.3672E-15 (   0.00%)
6    s-32        n,p              -4.5487E-09 +/- 5.5174E-17 (   0.00%)
6    s-32        n,alpha          -6.3539E-09 +/- 7.3703E-15 (   0.00%)
6    s-33        total            -3.5098E-10 +/- 2.0849E-14 (   0.01%)
6    s-33        scatter           1.1379E-10 +/- 1.9431E-14 (   0.02%)
6    s-33        elastic           1.0619E-10 +/- 1.9431E-14 (   0.02%)
6    s-33        n,n'              7.3395E-12 +/- 2.0279E-18 (   0.00%)
6    s-33        n,2n              1.7646E-13 +/- 2.0474E-20 (   0.00%)
6    s-33        capture          -4.6477E-10 +/- 1.4277E-15 (   0.00%)
6    s-33        n,gamma          -1.2051E-10 +/- 3.3390E-17 (   0.00%)
6    s-33        n,p              -3.3567E-11 +/- 1.3744E-17 (   0.00%)
6    s-33        n,alpha          -3.1070E-10 +/- 1.4014E-15 (   0.00%)
6    s-34        total             3.0984E-10 +/- 8.4943E-14 (   0.03%)
6    s-34        scatter           7.5435E-10 +/- 8.4929E-14 (   0.01%)
6    s-34        elastic           7.1661E-10 +/- 8.4929E-14 (   0.01%)
6    s-34        n,n'              3.6985E-11 +/- 4.3271E-18 (   0.00%)
6    s-34        n,2n              2.4941E-13 +/- 1.8198E-20 (   0.00%)
6    s-34        capture          -4.4451E-10 +/- 3.1938E-17 (   0.00%)
6    s-34        n,gamma          -4.3181E-10 +/- 3.1937E-17 (   0.00%)
6    s-34        n,p              -2.5649E-12 +/- 7.4283E-20 (   0.00%)
6    s-34        n,alpha          -1.0132E-11 +/- 1.3880E-19 (   0.00%)
6    s-36        total             2.4367E-12 +/- 4.8470E-16 (   0.02%)
```

```
6    s-36        scatter           3.9693E-12 +/- 4.8442E-16 (   0.01%)
6    s-36        elastic           3.8641E-12 +/- 4.8442E-16 (   0.01%)
6    s-36        n,n'              9.8043E-14 +/- 1.0060E-20 (   0.00%)
6    s-36        n,2n              7.0907E-15 +/- 8.5709E-22 (   0.00%)
6    s-36        capture          -1.5325E-12 +/- 3.2624E-19 (   0.00%)
6    s-36        n,gamma          -1.5317E-12 +/- 3.2624E-19 (   0.00%)
6    s-36        n,p              -1.2365E-17 +/- 1.4329E-24 (   0.00%)
6    s-36        n,alpha          -8.4074E-16 +/- 3.8508E-23 (   0.00%)
6    fe-54       total            -3.6694E-09 +/- 6.3742E-12 (   0.17%)
6    fe-54       scatter           3.8072E-09 +/- 6.3646E-12 (   0.17%)
6    fe-54       elastic           3.6383E-09 +/- 6.3646E-12 (   0.17%)
6    fe-54       n,n'              1.6766E-10 +/- 2.5736E-17 (   0.00%)
```

(continues on next page)

(continued from previous page)

6	fe-54	n,2n	4.3302E-15	+/- 0.0000E+00	(0.00%)
6	fe-54	capture	-7.4766E-09	+/- 9.6690E-15	(0.00%)
6	fe-54	n,gamma	-7.0941E-09	+/- 9.6690E-15	(0.00%)
6	fe-54	n,p	-3.7781E-10	+/- 4.3570E-18	(0.00%)
6	fe-54	n,d	-8.2739E-14	+/- 9.5098E-21	(0.00%)
6	fe-54	n,alpha	-4.5819E-12	+/- 1.0972E-19	(0.00%)
6	fe-56	total	-1.0500E-07	+/- 4.7404E-10	(0.45%)
6	fe-56	scatter	2.1456E-08	+/- 7.5931E-12	(0.04%)
6	fe-56	elastic	1.6631E-08	+/- 7.5931E-12	(0.05%)
6	fe-56	n,n'	4.8137E-09	+/- 2.9475E-15	(0.00%)
6	fe-56	n,2n	1.0449E-11	+/- 7.3407E-19	(0.00%)
6	fe-56	capture	-1.2645E-07	+/- 1.0772E-10	(0.09%)
6	fe-56	n,gamma	-1.2633E-07	+/- 1.0772E-10	(0.09%)
6	fe-56	n,p	-8.2373E-11	+/- 2.2336E-18	(0.00%)
6	fe-56	n,d	-2.1784E-13	+/- 2.2646E-20	(0.00%)
6	fe-56	n,t	-4.5091E-16	+/- 2.5804E-24	(0.00%)
6	fe-56	n,he-3	-5.1613E-17	+/- 1.4066E-24	(0.00%)
6	fe-56	n,alpha	-3.4811E-11	+/- 9.9336E-19	(0.00%)
6	fe-57	total	-1.5428E-09	+/- 1.5175E-12	(0.10%)
6	fe-57	scatter	1.1965E-09	+/- 1.5103E-12	(0.13%)
6	fe-57	elastic	9.2071E-10	+/- 1.5085E-12	(0.16%)
6	fe-57	n,n'	2.7180E-10	+/- 1.9306E-14	(0.01%)
6	fe-57	n,2n	3.9813E-12	+/- 3.4807E-19	(0.00%)
6	fe-57	capture	-2.7393E-09	+/- 7.2413E-15	(0.00%)
6	fe-57	n,gamma	-2.7362E-09	+/- 7.2413E-15	(0.00%)
6	fe-57	n,p	-1.2950E-12	+/- 2.2824E-20	(0.00%)
6	fe-57	n,alpha	-1.7737E-12	+/- 1.9032E-20	(0.00%)
6	fe-58	total	-7.0000E-11	+/- 1.6196E-14	(0.02%)
6	fe-58	scatter	1.1679E-10	+/- 1.6041E-14	(0.01%)
6	fe-58	elastic	1.0304E-10	+/- 1.6040E-14	(0.02%)
6	fe-58	n,n'	1.3592E-11	+/- 9.0752E-18	(0.00%)
6	fe-58	n,2n	1.6128E-13	+/- 1.8159E-20	(0.00%)
6	fe-58	capture	-1.8679E-10	+/- 7.3090E-16	(0.00%)
6	fe-58	n,gamma	-1.8677E-10	+/- 7.3090E-16	(0.00%)
6	fe-58	n,p	-2.8627E-15	+/- 2.6305E-22	(0.00%)
6	fe-58	n,alpha	-1.6727E-14	+/- 1.0033E-21	(0.00%)

6	cu-63	total	-3.1331E-03	+/- 6.7640E-06	(0.22%)
6	cu-63	scatter	5.1503E-04	+/- 4.5047E-06	(0.87%)
6	cu-63	elastic	4.3487E-04	+/- 4.1844E-06	(0.96%)
6	cu-63	n,n'	7.8687E-05	+/- 1.1401E-06	(1.45%)
6	cu-63	n,2n	2.1308E-07	+/- 7.0126E-09	(3.29%)
6	cu-63	capture	-3.6481E-03	+/- 3.1712E-06	(0.09%)
6	cu-63	n,gamma	-3.6222E-03	+/- 3.1710E-06	(0.09%)
6	cu-63	n,p	-2.4693E-05	+/- 3.5515E-08	(0.14%)
6	cu-63	n,d	-5.2482E-07	+/- 1.5798E-09	(0.30%)
6	cu-63	n,he-3	-3.0750E-12	+/- 3.9260E-19	(0.00%)
6	cu-63	n,alpha	-6.7771E-07	+/- 2.7471E-09	(0.41%)
6	cu-65	total	-6.0763E-04	+/- 4.7381E-06	(0.78%)
6	cu-65	scatter	1.7893E-04	+/- 4.2465E-06	(2.37%)
6	cu-65	elastic	1.4608E-04	+/- 4.1860E-06	(2.87%)
6	cu-65	n,n'	3.2461E-05	+/- 4.9054E-07	(1.51%)
6	cu-65	n,2n	3.7791E-07	+/- 7.8368E-09	(2.07%)
6	cu-65	capture	-7.8656E-04	+/- 6.8251E-07	(0.09%)
6	cu-65	n,gamma	-7.8620E-04	+/- 6.8251E-07	(0.09%)
6	cu-65	n,p	-3.1043E-07	+/- 7.8886E-10	(0.25%)
6	cu-65	n,d	-4.0903E-08	+/- 1.8559E-10	(0.45%)
6	cu-65	n,t	-1.0623E-11	+/- 2.7949E-18	(0.00%)
6	cu-65	n,he-3	-3.8976E-13	+/- 1.3313E-20	(0.00%)
6	cu-65	n,alpha	-8.9150E-09	+/- 1.3569E-15	(0.00%)
7	h-1	total	-3.2827E-02	+/- 6.0121E-04	(1.83%)
7	h-1	scatter	2.0058E-02	+/- 5.9506E-04	(2.97%)
7	h-1	elastic	2.0058E-02	+/- 5.9506E-04	(2.97%)
7	h-1	capture	-5.2884E-02	+/- 8.0351E-06	(0.02%)

(continues on next page)

(continued from previous page)

```
7    h-1          n,gamma           -5.2884E-02 +/- 8.0351E-06 (   0.02%)
7    o-16         total              2.6563E-02 +/- 3.0100E-05 (   0.11%)
7    o-16         scatter            2.6725E-02 +/- 3.0097E-05 (   0.11%)
7    o-16         elastic            2.6660E-02 +/- 3.0097E-05 (   0.11%)
7    o-16         n,n'               6.2924E-05 +/- 8.2297E-08 (   0.13%)
7    o-16         n,2n               3.8366E-12 +/- 8.0943E-18 (   0.00%)
7    o-16         capture           -1.6136E-04 +/- 6.2177E-08 (   0.04%)
7    o-16         n,gamma           -1.5133E-05 +/- 2.2991E-09 (   0.02%)
7    o-16         n,p               -7.7305E-07 +/- 2.9519E-09 (   0.38%)
7    o-16         n,d               -7.8828E-08 +/- 5.6744E-10 (   0.72%)
7    o-16         n,t               -2.2174E-13 +/- 1.4667E-18 (   0.00%)
7    o-16         n,alpha           -1.4538E-04 +/- 6.1000E-08 (   0.04%)
```

```
-----------------------------------------------
   Total Sensitivity Coefficients by Nuclide
-----------------------------------------------

Mixture    Nuclide     Atom Density   Sensitivity      Std. Dev. % Std. Dev.
------- -------------  ------------   -----------     ---------- -----------
  1    u-234          5.1835E-06   -9.0979E-04  +/-  3.1204E-07 (   0.03%)
  1    u-235          1.0102E-03    2.0616E-01  +/-  7.0501E-05 (   0.03%)
  1    u-236          5.1395E-06   -2.1805E-04  +/-  1.2725E-07 (   0.06%)
  1    u-238          2.2157E-02   -5.9609E-02  +/-  2.6857E-05 (   0.05%)
  1    o-16           4.6753E-02    1.1444E-02  +/-  3.3431E-05 (   0.29%)
  2    o-16           3.3338E-02    3.6005E-02  +/-  2.9143E-05 (   0.08%)
  4    o-16           1.2461E-02    3.9480E-05  +/-  8.0523E-07 (   2.04%)
  5    o-16           1.4273E-02    5.7694E-05  +/-  6.3654E-07 (   1.10%)
  6    o-16           1.0064E-04    6.9207E-07  +/-  7.5312E-09 (   1.09%)
  7    o-16           3.3338E-02    2.6563E-02  +/-  3.0100E-05 (   0.11%)
  2    h-1            6.6675E-02    2.0159E-01  +/-  4.6480E-04 (   0.23%)
  4    h-1            5.8178E-02   -3.3487E-05  +/-  3.3383E-05 (  99.69%)
  5    h-1            5.6642E-02   -4.6343E-05  +/-  2.5632E-05 (  55.31%)
  7    h-1            6.6675E-02   -3.2827E-02  +/-  6.0121E-04 (   1.83%)
  3    al-27          5.8433E-02   -6.7582E-05  +/-  2.3201E-05 (  34.33%)
  3    cr-50          2.7074E-06   -1.3040E-05  +/-  9.1891E-09 (   0.07%)
  3    cr-52          5.2210E-05   -8.7501E-06  +/-  2.6630E-08 (   0.30%)
  3    cr-53          5.9194E-06   -3.4245E-05  +/-  2.1670E-08 (   0.06%)
  3    cr-54          1.4736E-06   -1.6695E-08  +/-  4.1104E-10 (   2.46%)
  3    cu-63          6.5503E-05   -9.5973E-05  +/-  7.0063E-08 (   0.07%)
  6    cu-63          5.8191E-02   -3.1331E-03  +/-  6.7640E-06 (   0.22%)
  3    cu-65          2.9195E-05   -1.7689E-05  +/-  4.4209E-08 (   0.25%)
  6    cu-65          2.5937E-02   -6.0763E-04  +/-  4.7381E-06 (   0.78%)
  3    mg-24          5.2648E-04    4.5166E-05  +/-  4.5339E-07 (   1.00%)
  6    mg-24          3.4888E-06    3.2809E-08  +/-  5.8412E-12 (   0.02%)
  3    mg-25          6.6651E-05    1.2140E-06  +/-  3.2733E-08 (   2.70%)
  6    mg-25          4.4168E-07    2.2861E-09  +/-  1.0962E-12 (   0.05%)
  3    mg-26          7.3383E-05    5.2992E-06  +/-  3.2698E-08 (   0.62%)
  6    mg-26          4.8629E-07    3.6092E-09  +/-  4.4230E-13 (   0.01%)
  3    mn-55          2.2115E-05   -9.6360E-05  +/-  1.4079E-07 (   0.15%)
  3    ti-46          2.0300E-06   -1.8856E-07  +/-  8.6800E-10 (   0.46%)
  3    ti-47          1.8524E-06   -8.7007E-07  +/-  2.4463E-09 (   0.28%)
  3    ti-48          1.8727E-05   -4.5957E-05  +/-  3.3263E-08 (   0.07%)
  3    ti-49          1.3956E-06   -6.8733E-07  +/-  6.2660E-10 (   0.09%)
  3    ti-50          1.3702E-06    5.1758E-08  +/-  4.1741E-10 (   0.81%)
  3    si-28          3.1918E-04    4.3107E-06  +/-  1.2244E-07 (   2.84%)
  4    si-28          8.8873E-05    1.5659E-07  +/-  3.1168E-09 (   1.99%)
  6    si-28          3.5253E-05    1.7607E-07  +/-  1.6638E-09 (   0.94%)
  3    si-29          1.6161E-05    5.3687E-07  +/-  5.6554E-09 (   1.05%)
  4    si-29          4.5000E-06    8.3353E-09  +/-  7.5001E-15 (   0.00%)
  6    si-29          1.7850E-06    7.3370E-09  +/-  1.5194E-12 (   0.02%)
  3    si-30          1.0728E-05    1.3408E-07  +/-  4.4925E-09 (   3.35%)
  4    si-30          2.9872E-06    5.0684E-09  +/-  6.4263E-15 (   0.00%)
  6    si-30          1.1849E-06    4.2864E-09  +/-  9.7142E-13 (   0.02%)
  3    fe-54          5.9897E-06   -3.9407E-06  +/-  8.5216E-09 (   0.22%)
```

(continues on next page)

(continued from previous page)

```
  6    fe-54          2.2682E-07   -3.6694E-09  +/-  6.3742E-12 (   0.17%)
  3    fe-56          9.3114E-05   -6.9426E-05  +/-  1.3507E-07 (   0.19%)
  6    fe-56          3.5261E-06   -1.0500E-07  +/-  4.7404E-10 (   0.45%)
  3    fe-57          2.1319E-06   -1.4580E-06  +/-  2.4578E-09 (   0.17%)
  6    fe-57          8.0732E-08   -1.5428E-09  +/-  1.5175E-12 (   0.10%)
```

```
3    fe-58          2.8426E-07   -8.6249E-08  +/-  9.2863E-11 (   0.11%)
6    fe-58          1.0764E-08   -7.0000E-11  +/-  1.6196E-14 (   0.02%)
4    c              4.3562E-02    1.4324E-04  +/-  3.4393E-06 (   2.40%)
5    c              3.5648E-02    1.5186E-04  +/-  2.0605E-06 (   1.36%)
6    c              1.5194E-03    9.7668E-06  +/-  1.3853E-07 (   1.42%)
4    ca-40          2.4875E-03   -1.7949E-06  +/-  1.1894E-07 (   6.63%)
4    ca-42          1.6602E-05   -3.3798E-08  +/-  4.2370E-10 (   1.25%)
4    ca-43          3.4641E-06   -2.9992E-07  +/-  8.5309E-10 (   0.28%)
4    ca-44          5.3527E-05   -1.9121E-07  +/-  2.9350E-09 (   1.53%)
4    ca-46          1.0264E-07   -9.1789E-11  +/-  2.9314E-16 (   0.00%)
4    ca-48          4.7984E-06   -2.4743E-08  +/-  1.8724E-10 (   0.76%)
4    s-32           4.5439E-04   -9.3027E-07  +/-  1.5023E-08 (   1.61%)
6    s-32           3.1807E-06   -1.4228E-08  +/-  1.7286E-12 (   0.01%)
4    s-33           3.5865E-06   -1.1925E-08  +/-  1.0473E-14 (   0.00%)
6    s-33           2.5106E-08   -3.5098E-10  +/-  2.0849E-14 (   0.01%)
4    s-34           2.0132E-05    1.2596E-08  +/-  3.9241E-10 (   3.12%)
6    s-34           1.4093E-07    3.0984E-10  +/-  8.4943E-14 (   0.03%)
4    s-36           9.5640E-08    1.2812E-10  +/-  1.6526E-16 (   0.00%)
6    s-36           6.6948E-10    2.4367E-12  +/-  4.8470E-16 (   0.02%)
6    na-23          4.6695E-06    2.3175E-08  +/-  1.4447E-09 (   6.23%)
```

```
 ------------------------------------------------
    Total Sensitivity Coefficients by Mixture
 ------------------------------------------------

 Mixture   Sensitivity      Std. Dev. % Std. Dev.
 -------   -----------     ---------- -----------
      1    1.5686E-01  +/-  8.2519E-05 (   0.05%)
      2    2.3760E-01  +/-  4.6571E-04 (   0.20%)
      3   -3.9956E-04  +/-  2.3207E-05 (   5.81%)
      4    1.4613E-04  +/-  3.3569E-05 (  22.97%)
      5    1.6321E-04  +/-  2.5723E-05 (  15.76%)
      6   -3.7302E-03  +/-  8.2596E-06 (   0.22%)
      7   -6.2632E-03  +/-  6.0196E-04 (   9.61%)

 ------------------------------
    Problem Characterization
 ------------------------------

 median fission group is 222    5.000E-02 to 4.000E-02(eV)
                           average fission group   2.077E+02 +/- 3.212E-01
              average energy(eV) causing fission   1.115E+05 +/- 2.059E+02
  energy(eV) of average lethargy causing fission   1.128E-01 +/- 1.749E-04

 median capture group is 223    4.000E-02 to 3.000E-02(eV)
                           average capture group   2.072E+02 +/- 1.110E-01
              average energy(eV) causing capture   3.771E+04 +/- 4.509E+01
  energy(eV) of average lethargy causing capture   8.751E-02 +/- 4.773E-05
```

Figure 6.3.16. SAMS output for LEU-COMP-THERM-009 case 10 sample problem. (continued)

```
 median scatter group is 221    6.000E-02 to 5.000E-02(eV)
                           average scatter group   1.903E+02 +/- 5.331E-02
              average energy(eV) causing scatter   6.721E+04 +/- 2.875E+01
  energy(eV) of average lethargy causing scatter   3.839E-01 +/- 1.069E-04


 Generating working covariance matrix ...
```

(continues on next page)

(continued from previous page)

```
 Working covariance matrix created for future processing.
```

```
-----------------------------
   Uncertainty Information
-----------------------------


 the relative standard deviation of k-eff (% delta-k/k) due to cross-section covariance data is:

   0.6109 +/- 0.0000 % delta-k/k

  contributions to uncertainty in k-eff (% delta-k/k) by individual energy covariance matrices:

 covariance matrix

              nuclide-reaction        with        nuclide-reaction            % delta-k/k due to this matrix
      ------------------------------      ------------------------------  ------------------------------------
               u-235 nubar                         u-235 nubar                  2.8973E-01 +/- 3.9856E-06
               u-238 n,n'                          u-238 n,n'                   2.6630E-01 +/- 2.9468E-05
               u-235 chi                           u-235 chi                    2.4752E-01 +/- 1.6035E-05
                 h-1 elastic                         h-1 elastic                2.4045E-01 +/- 8.1628E-06
               u-235 n,gamma                       u-235 n,gamma                1.7330E-01 +/- 1.8063E-06
               u-238 n,gamma                       u-238 n,gamma                1.5959E-01 +/- 1.1160E-06
               u-235 fission                       u-235 n,gamma                1.2186E-01 +/- 9.5110E-07
                o-16 elastic                        o-16 elastic                1.1639E-01 +/- 4.4365E-06
               u-235 fission                       u-235 fission                1.0865E-01 +/- 1.2931E-06
               u-238 elastic                       u-238 n,n'                  -9.9362E-02 +/- 5.3145E-06
                 h-1 n,gamma                         h-1 n,gamma                9.8744E-02 +/- 3.7005E-07
               u-238 nubar                         u-238 nubar                  4.2236E-02 +/- 1.3385E-07
               u-238 elastic                       u-238 elastic                2.2575E-02 +/- 4.8689E-07
               al-27 n,gamma                       al-27 n,gamma                1.9265E-02 +/- 7.1802E-08
               al-27 n,n'                          al-27 n,n'                   1.7017E-02 +/- 5.0822E-07
               cu-63 n,gamma                       cu-63 n,gamma                1.4853E-02 +/- 1.0721E-07
               u-238 n,2n                          u-238 n,2n                   1.3281E-02 +/- 1.1785E-07
               u-238 fission                       u-238 fission                1.2837E-02 +/- 1.1196E-08
```

```
u-238 chi                       u-238 chi                    1.1469E-02 +/- 3.3598E-08
al-27 elastic                   al-27 n,n'                  -1.0157E-02 +/- 2.1624E-07
u-238 elastic                   u-238 n,gamma                8.5476E-03 +/- 7.8644E-07
u-235 elastic                   u-235 n,gamma                6.0964E-03 +/- 7.9059E-08
u-235 elastic                   u-235 fission               -4.4946E-03 +/- 7.3563E-08
 o-16 n,alpha                    o-16 n,alpha                4.0401E-03 +/- 2.6708E-09
u-234 n,gamma                   u-234 n,gamma                3.8813E-03 +/- 2.0326E-08
 o-16 n,n'                       o-16 n,n'                   3.8417E-03 +/- 1.1648E-08
al-27 elastic                   al-27 elastic                3.6255E-03 +/- 4.5685E-08
cu-65 n,gamma                   cu-65 n,gamma                3.4344E-03 +/- 5.5487E-09
u-235 n,n'                      u-235 n,n'                   3.3168E-03 +/- 4.5674E-09
cu-63 elastic                   cu-63 elastic                3.2014E-03 +/- 4.7334E-08
 o-16 n,n'                       o-16 elastic                -2.1989E-03 +/- 3.3250E-09
cu-65 elastic                   cu-65 elastic                1.6081E-03 +/- 1.1779E-08
u-235 elastic                   u-235 n,n'                  -1.5190E-03 +/- 1.0084E-09
u-238 elastic                   u-238 fission               -1.2258E-03 +/- 8.6756E-10
u-238 fission                   u-238 n,gamma                1.1071E-03 +/- 1.6581E-10
u-238 elastic                   u-238 n,2n                  -1.0603E-03 +/- 8.3265E-09
 o-16 n,gamma                    o-16 n,gamma                6.6497E-04 +/- 1.4657E-11
u-235 elastic                   u-235 elastic                6.4491E-04 +/- 5.5253E-10
u-236 n,gamma                   u-236 n,gamma                6.4328E-04 +/- 4.8310E-10
al-27 elastic                   al-27 n,gamma                6.3680E-04 +/- 4.9874E-08
u-234 fission                   u-234 fission                5.5055E-04 +/- 1.4300E-11
fe-56 n,gamma                   fe-56 n,gamma                4.6510E-04 +/- 4.5069E-11
```

(continues on next page)

(continued from previous page)

mn-55 n,gamma	mn-55 n,gamma	4.2641E-04 +/- 3.3567E-11
cu-63 n,p	cu-63 n,p	4.1549E-04 +/- 2.0900E-10
ti-48 n,gamma	ti-48 n,gamma	4.1164E-04 +/- 5.1774E-11
al-27 n,p	al-27 n,p	4.0075E-04 +/- 1.5438E-10
cu-63 elastic	cu-63 n,gamma	3.5387E-04 +/- 6.2656E-10
cu-63 elastic	cu-63 n,n'	-3.5353E-04 +/- 8.2827E-10
u-236 fission	u-236 fission	3.0090E-04 +/- 5.0084E-12
u-235 n,2n	u-235 n,2n	2.9426E-04 +/- 7.2192E-11
cr-53 n,gamma	cr-53 n,gamma	2.8295E-04 +/- 1.8411E-11
cu-63 n,n'	cu-63 n,n'	2.7900E-04 +/- 9.5920E-10
mg-24 elastic	mg-24 elastic	2.2982E-04 +/- 3.3498E-10
c elastic	c elastic	1.8308E-04 +/- 1.3865E-10
cu-63 n,n'	cu-63 n,p	1.8186E-04 +/- 2.1087E-10
cu-65 elastic	cu-65 n,gamma	1.7182E-04 +/- 1.7540E-10
cu-65 elastic	cu-65 n,n'	-1.4020E-04 +/- 1.3870E-10
cu-65 n,n'	cu-65 n,n'	1.1168E-04 +/- 1.6558E-10
al-27 elastic	al-27 n,p	1.0924E-04 +/- 5.0806E-11
cr-52 n,gamma	cr-52 n,gamma	9.4888E-05 +/- 2.0408E-12
mg-24 n,n'	mg-24 n,n'	8.9307E-05 +/- 1.6296E-11
si-28 n,gamma	si-28 n,gamma	8.3099E-05 +/- 1.6055E-12
c n,n'	c elastic	-7.0959E-05 +/- 3.4172E-11
cr-50 n,gamma	cr-50 n,gamma	6.1797E-05 +/- 7.7139E-13
al-27 n,alpha	al-27 n,alpha	5.1627E-05 +/- 6.3487E-12
u-234 n,n'	u-234 n,n'	4.9075E-05 +/- 1.3555E-12

c n,n'	c n,n'	4.8570E-05 +/- 3.6800E-11
o-16 n,p	o-16 n,p	4.7167E-05 +/- 3.5470E-12
mn-55 elastic	mn-55 elastic	4.3763E-05 +/- 6.4296E-11
cu-63 n,n'	cu-63 n,gamma	4.1035E-05 +/- 1.8996E-11
u-236 n,n'	u-236 n,n'	3.8927E-05 +/- 1.4655E-12
ca-40 n,gamma	ca-40 n,gamma	3.8300E-05 +/- 2.0407E-12
al-27 n,n'	al-27 n,p	3.8185E-05 +/- 1.9522E-11
fe-54 n,gamma	fe-54 n,gamma	3.2900E-05 +/- 2.4380E-13
mg-26 elastic	mg-26 elastic	3.0966E-05 +/- 4.0780E-12
mg-24 n,gamma	mg-24 n,gamma	2.9109E-05 +/- 1.6944E-13
o-16 elastic	o-16 n,gamma	2.9062E-05 +/- 6.2232E-12
mg-25 elastic	mg-25 elastic	2.5701E-05 +/- 3.0122E-12
u-235 elastic	u-235 n,2n	-2.5517E-05 +/- 6.3158E-12
ca-40 elastic	ca-40 elastic	2.4956E-05 +/- 1.5456E-11
si-28 elastic	si-28 elastic	2.3690E-05 +/- 1.8751E-12
al-27 n,n'	al-27 n,alpha	1.8266E-05 +/- 3.6345E-12
si-28 n,n'	si-28 n,n'	1.8068E-05 +/- 7.2723E-13
fe-57 n,gamma	fe-57 n,gamma	1.7622E-05 +/- 8.0437E-14
cu-63 n,d	cu-63 n,d	1.7414E-05 +/- 9.7220E-13
u-234 elastic	u-234 elastic	1.6232E-05 +/- 1.9361E-12
o-16 n,d	o-16 n,d	1.3977E-05 +/- 5.9002E-13
fe-56 elastic	fe-56 elastic	1.3490E-05 +/- 1.2632E-12
al-27 elastic	al-27 n,alpha	1.1830E-05 +/- 1.6930E-12
mn-55 elastic	mn-55 n,gamma	1.1605E-05 +/- 1.0979E-12
ti-46 n,gamma	ti-46 n,gamma	1.1500E-05 +/- 2.6908E-14
u-236 elastic	u-236 elastic	1.1425E-05 +/- 8.1487E-13
cr-52 elastic	cr-52 elastic	1.1243E-05 +/- 7.7884E-13
si-28 elastic	si-28 n,n'	-1.0955E-05 +/- 3.0843E-13
fe-56 n,n'	fe-56 n,n'	1.0721E-05 +/- 2.4123E-13
al-27 n,n'	al-27 n,gamma	1.0698E-05 +/- 7.9823E-13
mg-25 n,gamma	mg-25 n,gamma	1.0650E-05 +/- 1.6936E-14
si-28 n,p	si-28 n,p	1.0124E-05 +/- 1.7907E-13
cu-65 n,n'	cu-65 n,gamma	9.9993E-06 +/- 9.1006E-13
mg-25 n,n'	mg-25 n,n'	9.7352E-06 +/- 1.3961E-13
ca-43 n,gamma	ca-43 n,gamma	9.5474E-06 +/- 1.2570E-13
cr-50 elastic	cr-50 elastic	9.4119E-06 +/- 7.7575E-13
cu-63 n,n'	cu-63 n,d	9.0483E-06 +/- 9.7927E-13
mg-26 n,n'	mg-26 n,n'	8.6371E-06 +/- 1.2850E-13
c n,gamma	c n,gamma	7.3604E-06 +/- 4.2545E-14

(continues on next page)

(continued from previous page)

cu-65 n,p	cu-65 n,p	7.2791E-06 +/- 1.2170E-13
s-32 elastic	s-32 elastic	6.8662E-06 +/- 8.4130E-13
si-28 n,n'	si-28 n,p	6.6795E-06 +/- 1.9459E-13
fe-56 elastic	fe-56 n,n'	-6.2701E-06 +/- 7.5901E-14
cr-52 n,n'	cr-52 n,n'	6.0843E-06 +/- 8.2734E-14
cr-53 elastic	cr-53 elastic	5.4910E-06 +/- 9.0167E-13
s-32 n,gamma	s-32 n,gamma	5.0179E-06 +/- 3.4498E-14
mn-55 n,n'	mn-55 n,n'	5.0115E-06 +/- 4.1826E-14
si-28 n,alpha	si-28 n,alpha	4.8192E-06 +/- 5.1984E-14

si-30 n,gamma	si-30 n,gamma	4.7027E-06 +/- 1.0526E-13
ti-48 n,n'	ti-48 n,n'	3.9882E-06 +/- 2.8137E-14
u-236 n,2n	u-236 n,2n	3.8665E-06 +/- 6.9405E-15
cu-65 n,n'	cu-65 n,p	3.6181E-06 +/- 1.2145E-13
cr-53 elastic	cr-53 n,gamma	3.2232E-06 +/- 4.5743E-14
cr-52 elastic	cr-52 n,n'	-3.1662E-06 +/- 2.3274E-14
si-28 n,n'	si-28 n,alpha	3.1124E-06 +/- 5.5246E-14
si-29 n,gamma	si-29 n,gamma	2.9509E-06 +/- 0.0000E+00
ti-47 elastic	ti-47 elastic	2.9274E-06 +/- 2.6958E-13
fe-56 elastic	fe-56 n,gamma	2.7043E-06 +/- 3.8368E-14
ti-47 n,gamma	ti-47 n,gamma	2.6204E-06 +/- 0.0000E+00
si-29 elastic	si-29 elastic	2.5109E-06 +/- 2.3146E-14
fe-54 elastic	fe-54 elastic	2.4053E-06 +/- 1.4319E-13
ti-49 elastic	ti-49 elastic	2.2186E-06 +/- 1.6191E-13
si-29 n,n'	si-29 n,n'	2.1944E-06 +/- 9.7431E-15
ca-44 n,gamma	ca-44 n,gamma	2.0769E-06 +/- 5.0792E-15
cu-63 n,alpha	cu-63 n,alpha	1.9167E-06 +/- 1.6316E-14
c n,n'	c n,alpha	1.9077E-06 +/- 1.1726E-13
ti-49 n,gamma	ti-49 n,gamma	1.8792E-06 +/- 0.0000E+00
ti-46 elastic	ti-46 elastic	1.8174E-06 +/- 1.9222E-14
cr-54 n,gamma	cr-54 n,gamma	1.8052E-06 +/- 0.0000E+00
cr-50 elastic	cr-50 n,gamma	-1.6885E-06 +/- 2.2389E-15
al-27 n,2n	al-27 n,2n	1.6378E-06 +/- 1.0328E-13
mg-26 n,gamma	mg-26 n,gamma	1.4435E-06 +/- 0.0000E+00
ca-40 n,n'	ca-40 n,n'	1.3555E-06 +/- 3.3924E-14
si-29 elastic	si-29 n,n'	-1.3141E-06 +/- 2.9905E-15
cu-65 n,d	cu-65 n,d	1.2882E-06 +/- 8.1066E-15
si-30 elastic	si-30 elastic	1.2448E-06 +/- 3.8759E-15
cr-52 elastic	cr-52 n,gamma	-1.2364E-06 +/- 2.5560E-15
cu-63 n,2n	cu-63 n,2n	1.2318E-06 +/- 7.9923E-14
c n,alpha	c n,alpha	1.0668E-06 +/- 6.1427E-14
ti-50 elastic	ti-50 elastic	1.0218E-06 +/- 0.0000E+00
cu-63 n,n'	cu-63 n,alpha	1.0189E-06 +/- 1.5895E-14
cr-53 n,n'	cr-53 n,n'	9.9320E-07 +/- 1.7846E-15
ca-44 elastic	ca-44 elastic	9.9005E-07 +/- 9.0779E-15
ti-46 n,n'	ti-46 n,n'	9.6859E-07 +/- 0.0000E+00
fe-57 elastic	fe-57 elastic	9.5215E-07 +/- 9.6326E-15
ti-47 n,n'	ti-47 n,n'	9.0613E-07 +/- 0.0000E+00
ca-42 n,gamma	ca-42 n,gamma	8.9913E-07 +/- 0.0000E+00
si-30 n,n'	si-30 n,n'	8.6223E-07 +/- 0.0000E+00
al-27 n,n'	al-27 n,2n	-8.2603E-07 +/- 1.6896E-14
cu-65 n,n'	cu-65 n,d	6.9595E-07 +/- 8.0999E-15
fe-54 n,n'	fe-54 n,n'	6.7558E-07 +/- 0.0000E+00
u-234 n,2n	u-234 n,2n	6.5672E-07 +/- 0.0000E+00
cu-63 n,n'	cu-63 n,2n	-5.9352E-07 +/- 2.0173E-14
cr-53 elastic	cr-53 n,n'	-5.8452E-07 +/- 0.0000E+00
fe-58 n,gamma	fe-58 n,gamma	5.8445E-07 +/- 0.0000E+00
si-30 elastic	si-30 n,gamma	5.6922E-07 +/- 2.2346E-14

fe-56 n,n'	fe-56 n,gamma	5.2558E-07 +/- 0.0000E+00
ca-48 n,gamma	ca-48 n,gamma	5.1948E-07 +/- 0.0000E+00
cu-65 n,2n	cu-65 n,2n	4.8462E-07 +/- 7.8007E-15
ca-44 n,n'	ca-44 n,n'	4.5030E-07 +/- 0.0000E+00
si-30 elastic	si-30 n,n'	-4.4830E-07 +/- 0.0000E+00

(continues on next page)

(continued from previous page)

ti-49 n,n'	ti-49 n,n'	4.4467E-07 +/- 0.0000E+00
mg-25 n,2n	mg-25 n,2n	4.2964E-07 +/- 0.0000E+00
cr-52 n,n'	cr-52 n,gamma	4.1052E-07 +/- 0.0000E+00
fe-57 n,n'	fe-57 n,n'	3.9365E-07 +/- 0.0000E+00
si-29 n,alpha	si-29 n,alpha	3.8432E-07 +/- 0.0000E+00
cr-54 elastic	cr-54 elastic	3.6988E-07 +/- 0.0000E+00
si-28 elastic	si-28 n,gamma	3.6913E-07 +/- 0.0000E+00
cr-50 n,n'	cr-50 n,n'	3.5176E-07 +/- 0.0000E+00
fe-54 elastic	fe-54 n,n'	-3.4272E-07 +/- 0.0000E+00
fe-54 n,p	fe-54 n,p	3.2528E-07 +/- 0.0000E+00
cr-50 n,p	cr-50 n,p	3.2089E-07 +/- 0.0000E+00
s-32 n,n'	s-32 n,n'	3.1845E-07 +/- 0.0000E+00
ti-50 n,n'	ti-50 n,n'	3.1488E-07 +/- 0.0000E+00
fe-57 elastic	fe-57 n,n'	-2.9570E-07 +/- 0.0000E+00
si-28 n,n'	si-28 n,gamma	2.8067E-07 +/- 0.0000E+00
si-29 n,n'	si-29 n,alpha	2.6995E-07 +/- 0.0000E+00
s-34 elastic	s-34 elastic	2.5320E-07 +/- 0.0000E+00
s-34 n,gamma	s-34 n,gamma	2.3668E-07 +/- 0.0000E+00
ca-42 elastic	ca-42 elastic	2.3320E-07 +/- 0.0000E+00
fe-54 n,n'	fe-54 n,p	2.3313E-07 +/- 0.0000E+00
cr-50 n,n'	cr-50 n,p	2.3306E-07 +/- 0.0000E+00
cu-65 n,n'	cu-65 n,2n	-2.3141E-07 +/- 2.7300E-15
si-29 n,p	si-29 n,p	2.2789E-07 +/- 0.0000E+00
ca-43 elastic	ca-43 elastic	2.2169E-07 +/- 0.0000E+00
cr-50 elastic	cr-50 n,n'	-2.1587E-07 +/- 0.0000E+00
na-23 elastic	na-23 elastic	2.0908E-07 +/- 2.2505E-15
cu-65 n,alpha	cu-65 n,alpha	2.0897E-07 +/- 0.0000E+00
cr-54 n,n'	cr-54 n,n'	2.0240E-07 +/- 0.0000E+00
fe-56 n,alpha	fe-56 n,alpha	1.8941E-07 +/- 0.0000E+00
cr-52 n,p	cr-52 n,p	1.8760E-07 +/- 0.0000E+00
fe-54 elastic	fe-54 n,gamma	1.8132E-07 +/- 0.0000E+00
si-29 n,n'	si-29 n,p	1.6557E-07 +/- 0.0000E+00
cr-52 n,n'	cr-52 n,p	1.5092E-07 +/- 0.0000E+00
fe-56 n,n'	fe-56 n,alpha	1.4803E-07 +/- 0.0000E+00
fe-56 n,2n	fe-56 n,2n	1.4651E-07 +/- 0.0000E+00
ca-48 elastic	ca-48 elastic	1.4271E-07 +/- 0.0000E+00
ti-50 n,gamma	ti-50 n,gamma	1.3989E-07 +/- 0.0000E+00
s-33 n,gamma	s-33 n,gamma	1.2657E-07 +/- 0.0000E+00
fe-58 elastic	fe-58 elastic	1.2168E-07 +/- 0.0000E+00
cr-54 elastic	cr-54 n,n'	-1.2057E-07 +/- 0.0000E+00
cu-65 n,n'	cu-65 n,alpha	1.1181E-07 +/- 0.0000E+00
ca-42 n,n'	ca-42 n,n'	1.0111E-07 +/- 0.0000E+00
fe-57 elastic	fe-57 n,gamma	9.3777E-08 +/- 0.0000E+00

cr-53 n,n'	cr-53 n,gamma	8.4457E-08 +/- 0.0000E+00
fe-57 n,n'	fe-57 n,gamma	7.6615E-08 +/- 0.0000E+00
fe-56 n,n'	fe-56 n,2n	-7.4856E-08 +/- 0.0000E+00
na-23 n,gamma	na-23 n,gamma	6.9611E-08 +/- 0.0000E+00
mn-55 n,p	mn-55 n,p	5.8834E-08 +/- 0.0000E+00
fe-56 n,p	fe-56 n,p	5.6198E-08 +/- 0.0000E+00
na-23 elastic	na-23 n,n'	-4.8025E-08 +/- 0.0000E+00
fe-56 n,n'	fe-56 n,p	4.5081E-08 +/- 0.0000E+00
cr-50 n,alpha	cr-50 n,alpha	4.4847E-08 +/- 0.0000E+00
cr-52 n,alpha	cr-52 n,alpha	4.3919E-08 +/- 0.0000E+00
fe-58 n,n'	fe-58 n,n'	4.2403E-08 +/- 0.0000E+00
na-23 n,n'	na-23 n,n'	3.9418E-08 +/- 0.0000E+00
s-33 elastic	s-33 elastic	3.7973E-08 +/- 0.0000E+00
cr-50 n,n'	cr-50 n,alpha	3.6143E-08 +/- 0.0000E+00
cr-53 n,alpha	cr-53 n,alpha	3.5247E-08 +/- 0.0000E+00
cr-52 n,n'	cr-52 n,alpha	3.4289E-08 +/- 0.0000E+00
cr-50 n,n'	cr-50 n,gamma	3.3993E-08 +/- 0.0000E+00
mg-26 n,2n	mg-26 n,2n	3.2987E-08 +/- 0.0000E+00
cr-53 n,2n	cr-53 n,2n	3.0637E-08 +/- 0.0000E+00
ca-43 n,n'	ca-43 n,n'	2.9945E-08 +/- 0.0000E+00

(continues on next page)

(continued from previous page)

o-16 n,n'	o-16 n,gamma	2.8202E-08 +/- 0.0000E+00
mn-55 n,2n	mn-55 n,2n	2.8181E-08 +/- 0.0000E+00
si-29 elastic	si-29 n,gamma	2.7325E-08 +/- 0.0000E+00
cr-53 n,n'	cr-53 n,alpha	2.6753E-08 +/- 0.0000E+00
fe-54 n,n'	fe-54 n,gamma	2.5858E-08 +/- 0.0000E+00
si-29 n,2n	si-29 n,2n	2.4668E-08 +/- 0.0000E+00
fe-58 elastic	fe-58 n,n'	-2.3267E-08 +/- 0.0000E+00
si-28 n,d	si-28 n,d	2.2815E-08 +/- 0.0000E+00
si-30 n,n'	si-30 n,gamma	2.1466E-08 +/- 0.0000E+00
fe-58 elastic	fe-58 n,gamma	1.9178E-08 +/- 0.0000E+00
c n,n'	c n,gamma	1.7660E-08 +/- 0.0000E+00
na-23 elastic	na-23 n,gamma	-1.6839E-08 +/- 0.0000E+00
cr-53 n,n'	cr-53 n,2n	-1.6046E-08 +/- 0.0000E+00
fe-57 n,2n	fe-57 n,2n	1.5101E-08 +/- 0.0000E+00
s-34 n,n'	s-34 n,n'	1.4729E-08 +/- 0.0000E+00
si-30 n,2n	si-30 n,2n	1.3545E-08 +/- 0.0000E+00
fe-54 n,alpha	fe-54 n,alpha	1.3005E-08 +/- 0.0000E+00
fe-54 n,n'	fe-54 n,alpha	1.0111E-08 +/- 0.0000E+00
si-29 n,n'	si-29 n,2n	-1.0081E-08 +/- 0.0000E+00
si-30 n,alpha	si-30 n,alpha	8.4209E-09 +/- 0.0000E+00
cr-54 elastic	cr-54 n,gamma	8.3271E-09 +/- 0.0000E+00
fe-57 n,n'	fe-57 n,2n	-7.9525E-09 +/- 0.0000E+00
cr-53 n,p	cr-53 n,p	7.9053E-09 +/- 0.0000E+00
ca-46 elastic	ca-46 elastic	7.6477E-09 +/- 0.0000E+00
fe-58 n,n'	fe-58 n,gamma	7.2381E-09 +/- 0.0000E+00
si-30 n,n'	si-30 n,2n	-7.0930E-09 +/- 0.0000E+00
mn-55 n,alpha	mn-55 n,alpha	7.0079E-09 +/- 0.0000E+00
cr-52 n,2n	cr-52 n,2n	6.5167E-09 +/- 0.0000E+00

si-30 n,n'	si-30 n,alpha	6.4283E-09 +/- 0.0000E+00
ti-48 n,2n	ti-48 n,2n	6.2875E-09 +/- 0.0000E+00
cr-53 n,n'	cr-53 n,p	6.2628E-09 +/- 0.0000E+00
ti-49 n,2n	ti-49 n,2n	5.8476E-09 +/- 0.0000E+00
ca-46 n,gamma	ca-46 n,gamma	5.5651E-09 +/- 0.0000E+00
fe-57 n,alpha	fe-57 n,alpha	5.3090E-09 +/- 0.0000E+00
cr-54 n,n'	cr-54 n,gamma	4.9562E-09 +/- 0.0000E+00
ca-48 n,n'	ca-48 n,n'	4.9102E-09 +/- 0.0000E+00
fe-56 n,d	fe-56 n,d	4.7681E-09 +/- 0.0000E+00
fe-57 n,n'	fe-57 n,alpha	3.9611E-09 +/- 0.0000E+00
s-33 n,n'	s-33 n,n'	3.7151E-09 +/- 0.0000E+00
fe-56 n,n'	fe-56 n,d	3.6804E-09 +/- 0.0000E+00
cr-54 n,2n	cr-54 n,2n	3.6592E-09 +/- 0.0000E+00
si-29 n,n'	si-29 n,gamma	3.4777E-09 +/- 0.0000E+00
fe-57 n,p	fe-57 n,p	3.4511E-09 +/- 0.0000E+00
cr-52 n,n'	cr-52 n,2n	-3.4308E-09 +/- 0.0000E+00
ti-48 n,p	ti-48 n,p	2.9116E-09 +/- 0.0000E+00
fe-57 n,n'	fe-57 n,p	2.6805E-09 +/- 0.0000E+00
ti-47 n,2n	ti-47 n,2n	2.6563E-09 +/- 0.0000E+00
s-36 n,gamma	s-36 n,gamma	2.5689E-09 +/- 0.0000E+00
na-23 n,p	na-23 n,p	2.1365E-09 +/- 0.0000E+00
cr-54 n,n'	cr-54 n,2n	-2.0358E-09 +/- 0.0000E+00
si-30 n,p	si-30 n,p	1.9130E-09 +/- 0.0000E+00
ti-50 n,2n	ti-50 n,2n	1.4840E-09 +/- 0.0000E+00
si-30 n,n'	si-30 n,p	1.4354E-09 +/- 0.0000E+00
c n,n'	c n,d	1.3648E-09 +/- 0.0000E+00
na-23 n,alpha	na-23 n,alpha	1.3262E-09 +/- 0.0000E+00
s-36 elastic	s-36 elastic	1.2923E-09 +/- 0.0000E+00
fe-58 n,2n	fe-58 n,2n	1.2017E-09 +/- 0.0000E+00
o-16 n,n'	o-16 n,2n	-9.4097E-10 +/- 0.0000E+00
mn-55 n,d	mn-55 n,d	8.7967E-10 +/- 0.0000E+00
c n,d	c n,d	8.3498E-10 +/- 0.0000E+00
fe-54 n,d	fe-54 n,d	7.9457E-10 +/- 0.0000E+00
na-23 elastic	na-23 n,p	7.7621E-10 +/- 0.0000E+00
ca-46 n,n'	ca-46 n,n'	7.5538E-10 +/- 0.0000E+00

(continues on next page)

(continued from previous page)

fe-58 n,n'	fe-58 n,2n	-6.7419E-10 +/- 0.0000E+00
cu-65 n,t	cu-65 n,t	6.5557E-10 +/- 0.0000E+00
ti-48 n,alpha	ti-48 n,alpha	6.2508E-10 +/- 0.0000E+00
c n,n'	c n,p	6.0127E-10 +/- 0.0000E+00
na-23 elastic	na-23 n,alpha	4.5436E-10 +/- 0.0000E+00
fe-54 n,2n	fe-54 n,2n	3.8024E-10 +/- 0.0000E+00
c n,p	c n,p	3.6950E-10 +/- 0.0000E+00
cr-54 n,alpha	cr-54 n,alpha	3.3662E-10 +/- 0.0000E+00
cu-65 n,n'	cu-65 n,t	3.3605E-10 +/- 0.0000E+00
ca-44 n,2n	ca-44 n,2n	3.3085E-10 +/- 0.0000E+00
fe-54 n,n'	fe-54 n,2n	-3.2000E-10 +/- 0.0000E+00
cr-54 n,n'	cr-54 n,alpha	2.7557E-10 +/- 0.0000E+00
o-16 n,2n	o-16 n,2n	2.4315E-10 +/- 0.0000E+00

ca-43 n,2n	ca-43 n,2n	1.7648E-10 +/- 0.0000E+00
fe-56 n,n'	fe-56 n,t	1.5852E-10 +/- 0.0000E+00
cr-50 n,d	cr-50 n,d	1.5158E-10 +/- 0.0000E+00
ca-48 n,2n	ca-48 n,2n	1.3661E-10 +/- 0.0000E+00
cr-50 n,n'	cr-50 n,d	1.2412E-10 +/- 0.0000E+00
cu-63 n,he-3	cu-63 n,he-3	1.1688E-10 +/- 0.0000E+00
s-33 n,2n	s-33 n,2n	1.1455E-10 +/- 0.0000E+00
ti-46 n,2n	ti-46 n,2n	8.3916E-11 +/- 0.0000E+00
fe-58 n,alpha	fe-58 n,alpha	8.3616E-11 +/- 0.0000E+00
cr-50 n,2n	cr-50 n,2n	7.9460E-11 +/- 0.0000E+00
s-36 n,n'	s-36 n,n'	7.3825E-11 +/- 0.0000E+00
ca-42 n,2n	ca-42 n,2n	7.0527E-11 +/- 0.0000E+00
fe-58 n,n'	fe-58 n,alpha	6.8070E-11 +/- 0.0000E+00
cu-63 n,n'	cu-63 n,he-3	5.9819E-11 +/- 0.0000E+00
s-34 n,2n	s-34 n,2n	5.5409E-11 +/- 0.0000E+00
cr-50 n,n'	cr-50 n,2n	-4.5867E-11 +/- 0.0000E+00
cr-54 n,p	cr-54 n,p	3.4617E-11 +/- 0.0000E+00
fe-56 n,n'	fe-56 n,he-3	2.9868E-11 +/- 0.0000E+00
mn-55 n,t	mn-55 n,t	2.9356E-11 +/- 0.0000E+00
o-16 n,t	o-16 n,t	2.9008E-11 +/- 0.0000E+00
cr-54 n,n'	cr-54 n,p	2.7365E-11 +/- 0.0000E+00
fe-56 n,t	fe-56 n,t	1.5517E-11 +/- 0.0000E+00
fe-58 n,p	fe-58 n,p	1.3800E-11 +/- 0.0000E+00
fe-58 n,n'	fe-58 n,p	1.1055E-11 +/- 0.0000E+00
cu-65 n,he-3	cu-65 n,he-3	6.2232E-12 +/- 0.0000E+00
mg-24 n,2n	mg-24 n,2n	3.6825E-12 +/- 0.0000E+00
cu-65 n,n'	cu-65 n,he-3	3.1211E-12 +/- 0.0000E+00
na-23 n,2n	na-23 n,2n	2.9396E-12 +/- 0.0000E+00
s-36 n,2n	s-36 n,2n	1.8272E-12 +/- 0.0000E+00
fe-56 n,he-3	fe-56 n,he-3	1.7491E-12 +/- 0.0000E+00
ca-46 n,2n	ca-46 n,2n	1.7030E-12 +/- 0.0000E+00
na-23 elastic	na-23 n,2n	-1.0631E-12 +/- 0.0000E+00
mn-55 n,he-3	mn-55 n,he-3	8.9376E-13 +/- 0.0000E+00
ca-40 n,2n	ca-40 n,2n	4.9330E-13 +/- 0.0000E+00
s-32 n,2n	s-32 n,2n	1.1554E-13 +/- 0.0000E+00

```
Note: relative standard deviation in k-eff can be computed from
individual values by adding the square of the values with positive signs and
subtracting the square of the values with negative signs, then taking the square root


-----------------------------------
 SAMS execution complete
 Elapsed time: 8.10467 minutes
-----------------------------------
```

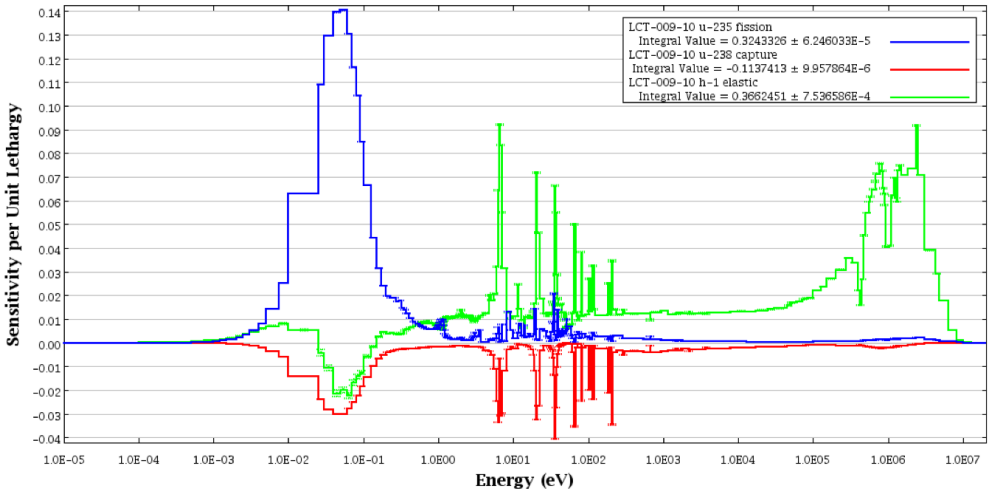


Fig. 6.6.11: Region-integrated sensitivity profiles from test case LEU-COMP-THERM-009 case 10.

6.6.5 ERROR AND WARNING MESSAGES

SAMS will display a message when a predetermined condition is encountered. Error messages are displayed prior to code termination. Warning messages are displayed when a potentially abnormal condition is encountered, but code execution continues.

6.6.5.1 Error Messages

SAMS Error 001: Selected calculation type _____ is not valid.
This message is displayed by VIP when the user enters a calculation type other than kenova or xsdrn.

SAMS Error 002: There was a zero response value.
This message is displayed by SENSITIVITY_SUMMATION when there was no fissile material in the problem or the forward or adjoint flux in all regions containing fissile material was zero. This message can be encountered when the adjoint solution fails to proceed far enough to accumulate active generations for flux accumulation in TSUNAMI-3D models.

SAMS Error 003: Error reading covariance data io=___ mat1=____ mt1=____ mat2=____ mt2=____ nblok=____ ngroup=____
This error is displayed by SAMCAVA when a non-zero return code is obtained when reading from the covariance data library.

SAMS Error 004: Adjoint flux file from KENO was not found on unit ____
This error is displayed by GET_FLUXES when the adjoint flux unit cannot be opened.

SAMS Error 005: Forward flux file from KENO was not found on unit ____
This error is displayed by GET_FLUXES when the forward flux unit cannot be opened.

SAMS Error 006: Adjoint fluxes were not calculated in the source case
This error is displayed by GET_FLUXES when an adjoint data file is found, but it does not contain flux data. The adjoint case should be rerun with flux accumulation turned on.

SAMS Error 007: Only the scalar fluxes were calculated. Execution is stopping.
This message is displayed by GET_FLUXES when the KENO calculation was run with both PNM=0 and NQD=0. The KENO case does not contain sufficient information to produce sensitivity coefficients.

SAMS Error 008: Error reading the forward restart file, record type ____
This message is displayed by GET_FLUXES when an unknown record type from the KENO forward solution is encountered.

SAMS Error 009: Error reading the adjoint restart file, record type ____
This message is displayed by GET_FLUXES if an unknown record type from the KENO adjoint solution is encountered.

SAMS Error 010: Invalid read key ____
This message is displayed by GET_INPUT when an invalid word is encountered in the input before a "read" card is encountered.

SAMS Error 011: Invalid block key ____. The first block must be initial.
This message is displayed by GET_INPUT when the first word after the first "read" is not "initial".

SAMS Error 012: End of input data reached. Expecting more data.
This message is displayed by GET_INPUT when the end of the input stream is unexpectedly encountered.

SAMS Error 013: Ending block keyword ___
does not equal the beginning block keyword ___
This message is displayed by GET_INPUT when the word following "END" does not match the word after "READ" for the current input block.

SAMS Error 014: Unknown read return flag ____
This message is displayed by GET_INPUT when a reading error occurs. This is likely due to an input error.

Error allocating ______
This message is displayed by GET_FLUXES when insufficient computational resources exist to allocate memory for processing the problem. If this message is encountered, try reducing the size of the calculation by reducing the number of mesh intervals (e.g., increase the mesh size, MSH in KENO), reducing the order of flux moments or angular fluxes accumulated (PNM and NQD in KENO), or using a cross-section data library with fewer energy groups.

6.6.5.2 Warning Messages

SAMS Warning 001: Use of mesh flux was requested (mfx=yes), but a mesh size of ____ was requested. Please enter a positive value for msh= in the KENO parameter data. SAMS will continue its calculation with the region fluxes, equivalent to mfx=no.
This message is displayed by GET_FLUXES when MFX=YES is entered in KENO parameter data, but the mesh size MSH is not entered or is non-positive. However, if the user specifies another mesh in a different section of a TSUNAMI-3D input (i.e. in the read GridGeometry block), then TSUNAMI will default to use that mesh.

SAMS Warning 002: The k-eff values for the forward and adjoint calculations differ by more than one percent.
This message is displayed by PRINT_TRANS when the k-eff values from the forward and adjoint calculations differ by more than one percent. This typically indicates poor convergence of the adjoint Monte Carlo calculation. The calculation should be rerun with more histories to improve agreement between the forward and adjoint k-eff values.

SAMS Warning 003: Implicit sensitivities were requested but no implicit datafiles were found. Proceeding with explicit sensitivity coefficients only.

This message is displayed by IMPLICIT_SENSITIVITIES when MAKEIMP is present in the SAMS input, but no datafiles from SENLIB or BONAMIST are found. The calculation will continue with explicit sensitivity coefficients only.

SAMS Warning 004: Implicit sensitivity of ____ ____ to ____ group ____ from ____ had a value of ____ and has been reset to 0.0.

This message is displayed by IMPLICIT_SENSITIVITIES when an implicit sensitivity value greater than that set by LARGEIMP is encountered. The large implicit value is likely due to an inconsistency between the CELLDATA specification for the material identified in the message and the use of that material in the geometry of the model.

SAMS Warning 005: Implicit sensitivity for mixture ____ ____ ____ exceeds the explicit sensitivity by more than an order of magnitude. Implicit sensitivity: ____Explicit sensitivity: ____ Please check for appropriate resonance self-shielding models in celldata.

This message is displayed by IMPLICIT_SENSITIVITIES when the implicit sensitivity value exceeds the explicit sensitivity value by more than an order of magnitude. The large implicit value is possibly due to an inconsistency between the CELLDATA specification for the material identified in the message and the use of that material in the geometry of the model.

SAMS Warning 006: An implicit sensitivity was generated by the resonance self-shielding calculation in ____ for mixture ____ ____ ___ with mixture ____ ____ which was not found in the model. This potentially important information will be skipped. Please check the celldata input.

This message is displayed by IMPLICIT_SENSITIVITIES when implicit data is generated by the resonance self-shielding codes, but a material needed for the calculation is not present in the geometry model. Please check for consistent use of materials in the CELLDATA input and in the geometry model.

SAMS Warning 007: USEANG was requested in the input, but angular fluxes were not computed. USEMOM has been set to true, and flux moments will be used to compute sensitivity coefficients.

This message is displayed by GET_FLUXES when USEANG is true but angular fluxes were not computed in the transport solution. In this case the flux moments computed in the transport solution are used to compute sensitivity coefficients. If neither angular fluxes nor flux moments are found, SAMS Error 007 will stop code execution prior to reaching this warning.

SAMS Warning 008: USEMOM was requested in the input, but flux moments were not computed. USEANG has been set to true, and angular fluxes will be used to compute sensitivity coefficients.

This message is displayed by GET_FLUXES when USEMOM is true but flux moments were not computed in the transport solution. In this case the angular fluxes computed in the transport solution are used to compute sensitivity coefficients. If neither angular fluxes nor flux moments are found, SAMS Error 007 will stop code execution prior to reaching this warning.

6.7 TSAR: TOOL FOR SENSITIVITY ANALYSIS OF REACTIVITY RESPONSES

M. L. Williams and M. A. Jessee

ABSTRACT

TSAR (Tool for Sensitivity Analysis of Reactivity Responses) is a SCALE functional module that computes nuclear data sensitivity coefficients for eigenvalue-difference responses such as reactor reactivity and worth coefficients. Examples include void reactivity, Doppler coefficients, and control rod worths. TSAR reads previously computed sensitivity coefficients for the k-eigenvalues at two states of a reactor system (or for two different systems) and combines them to obtain sensitivity coefficients for the difference. The k-eigenvalue sensitivities are typically obtained using the TSUNAMI-3D or -1D control sequences in SCALE. The reactivity sensitivity coefficients are combined with nuclear data covariance information to determine the uncertainty in the reactivity response.

ACKNOWLEDGMENT

Development of the TSAR code was funded by the U. S. Nuclear Regulatory Commission Office of Research.

6.7.1 INTRODUCTION

The TSUNAMI-1D, -2D, or -3D control sequences in SCALE compute multigroup sensitivity coefficients and uncertainties for the critical multiplication factor k, the reciprocal of the λ-eigenvalue of the neutron transport equation for a multiplying medium. The TSAR module in SCALE performs sensitivity/uncertainty (S/U) calculations for responses represented by the *difference* of two eigenvalues. These types of responses are often of interest in reactor physics applications. For example, TSAR can compute data sensitivities and uncertainties of reactivity responses such as control rod worths, fuel and moderator temperature coefficients, and void coefficients for two defined states of a power reactor [TSARWGC06]. Another potential application is in the analysis of benchmark critical experiments for nuclear data testing and validation studies. Data and methods deficiencies can introduce a computational bias manifested as a trend in calculated critical eigenvalues versus experiment parameters. TSAR can be applied to the *difference* in the computed eigenvalues of two benchmarks to establish the sensitivity of the bias trend to various nuclear data used in the calculations.

TSAR builds upon capabilities of other SCALE modules. TSUNAMI is first used to calculate sensitivities for the multiplication factors of the reference and altered states of the reactor, respectively. TSAR reads the sensitivity data files (.sdf file) produced by TSUNAMI *keff* calculations and uses them to compute relative or absolute sensitivities of an eigenvalue-difference response. The reactivity sensitivities are written to an output file for subsequent applications or visualization. TSAR also combines the calculated reactivity sensitivity coefficients with input nuclear data covariance matrices included in SCALE to determine the uncertainty of the reactivity response.

6.7.2 METHODOLOGY

A detailed description of the S/U methodology for reactivity responses is given in Reference D D; thus, only a brief overview is presented here. The lambda-eigenvalue form of the neutron transport equation for a multiplying medium is given by

$$(\mathbf{L} - \lambda\mathbf{P})\Phi = 0 \tag{6.7.1}$$

where L and P are the loss and production operators, respectively, for the Boltzmann equation describing a multiplying medium and $\lambda = \frac{1}{k}$ is the fundamental lambda-eigenvalue. It is assumed that the system is initially in a well-defined state 1 having a lambda- eigenvalue of λ_1. The reactivity for state 1 is defined as $\rho_1 = 1 - \lambda_1$. Suppose that changes in L and/or P transformed the original system into a new distinct

configuration designated as state 2, with the lambda eigenvalue of λ_2 and static reactivity of $\rho_2 = 1 - \lambda_2$. For example, the configuration change could be caused by moving a control rod or by voiding of the coolant. The reactivity insertion/withdrawal associated with the designated change in conditions is defined as

$$\rho_{1\to 2} = \rho_2 - \rho_1 = \lambda_1 - \lambda_2. \tag{6.7.2}$$

Eq. (6.7.2) defines the eigenvalue-difference (i.e., reactivity) response addressed by TSAR. The code edits the eigenvalues for the two reactor states and the value of the reactivity obtained from Eq. (6.7.2).

6.7.2.1 Reactivity sensitivity coefficients

The relative k-sensitivity coefficient for an arbitrary data parameter α appearing in the transport equation is equal to

$$\mathrm{S}_{\mathrm{k},\alpha} = \frac{\alpha \partial \mathrm{k}}{\mathrm{k} \partial \alpha} = -\frac{\alpha \partial \lambda}{\lambda \partial \alpha} \tag{6.7.3}$$

An analogous expression defines the relative sensitivity coefficient of the reactivity response:

$$\mathrm{S}_{\rho,\alpha} = \frac{\alpha \partial \rho_{1\to 2}}{\rho_{1\to 2} \partial \alpha} \tag{6.7.4}$$

Unlike the multiplication factor, the reactivity response can be negative. This can be source of confusion when interpreting the relative sensitivity coefficient; hence, by convention TSAR defines sensitivities relative to the *absolute value* of the reactivity; thus,

$$\mathrm{S}_{\rho,\alpha} \to \frac{\alpha}{|\rho_{1\to 2}|} \frac{\partial \rho_{1\to 2}}{\partial \alpha}. \tag{6.7.5}$$

In this way, a positive value for the relative sensitivity coefficient means that increasing the value of α always increases the value of the reactivity (i.e., a positive :math:` ho` becomes more positive, and a negative :math:` ho` becomes less negative). Conversely, a negative relative sensitivity coefficient means that increasing α always decreases the reactivity (i.e., a positive :math:` ho` becomes less positive, and a negative :math:` ho` becomes more negative). This convention is used in TSAR for all relative quantities involving the reactivity.

From the definitions in Eq. (6.7.2) and Eq. (6.7.3), Eq. (6.7.4) is simplified to the following expression used in TSAR:

$$\mathrm{S}_{\rho,\alpha} = \frac{\lambda_2 \, \mathrm{S}_{\mathrm{k2},\alpha} - \lambda_1 \, \mathrm{S}_{\mathrm{kl},\alpha}}{|\rho_{1\to 2}|}, \tag{6.7.6}$$

where $\mathrm{S}_{\mathrm{k1},\alpha}$ and $\mathrm{S}_{\mathrm{k2},\alpha}$ are the k-sensitivities for the two states. The relative change in the reactivity response due to an arbitrary relative variation (or uncertainty) in parameter α can be found very easily once the :math:` ho`-sensitivities are determined that is,

$$\frac{\Delta \rho_{1\to 2}}{|\rho_{1\to 2}|} \sim \mathrm{S}_{\rho,\alpha} \frac{\Delta \alpha}{\alpha}. \tag{6.7.7}$$

In cases where the net reactivity change is very small, the denominator of Eq. (6.7.6) approaches zero; thus, the relative sensitivity coefficient can increase without bound. For this reason TSAR provides an input option to compute U*absolute*U rather than U*relative*U sensitivity coefficients. Absolute quantities are indicated here by the presence of a tilde (~), while relative quantities have no tilde. The absolute sensitivity coefficient is

defined in TSAR as the absolute change in the reactivity, expressed in pcm (percent-milli), due to a fractional change in data:

$$S_{\rho,\alpha} = (\lambda_2 \, S_{k2,\alpha} - \lambda_1 \, S_{k1,\alpha}) \times 10^5, \tag{6.7.8}$$

so that

$$\Delta\rho_{1\to 2}(pcm) \sim S_{\rho,\alpha} \frac{\Delta\alpha}{\alpha}. \tag{6.7.9}$$

Prior to executing TSAR, it is necessary to perform TSUNAMI calculations for each reactor state, in order to evaluate the relative k-sensitivity coefficients in Eq. (6.7.3). These are written out in the SDF sensitivity file format and saved for input to TSAR. TSAR reads the two previously prepared files and uses them to evaluate Eq. (6.7.6) or Eq. (6.7.8) for the reactivity sensitivities. The :math:` ho`-sensitivities are then output to another SDF file. As discussed in Sect. 6.1 of the SCALE documentation, the complete sensitivities calculated by TSUNAMI include implicit effects associated with perturbations in resonance self-shielding; hence, the reactivity sensitivities also account for these effects, which can be significant.

TSAR prints the number of different sensitivity profiles that are computed and optionally can edit the input k-sensitivities corresponding to each reactor state, as well the calculated :math:` ho`-sensitivities. Sensitivities may be edited by a sum over group for each nuclide-reaction pair. In addition, the TSAR output SDF file containing multigroup :math:` ho`-sensitivities for each nuclide-reaction pair can be read by Fulcrum to produce plots of the energy-dependent sensitivities. The filename of the reactivity sensitivity data file is given as *job_name.react.sdf*, where *job_name* is the name of the TSAR input file (i.e., *job_name.inp*, *job_name.input*, or simply *job_name*). Additionally, if the user requests edits of the input k-sensitivities using the *print=all* option, the k-sensitivity data files are copied to the return directory as *job_name.kstate1.sdf* and *job_name.kstate2.sdf*. The *print* keyword is further described in the TSAR input section.

6.7.2.2 Reactivity uncertainty analysis

TSAR performs uncertainty analysis for eigenvalue-difference responses in much the same manner as the TSUNAMI sequences for the multiplication factor, except that either absolute or relative uncertainties may be computed. Assume that the transport calculations for the eigenvalues of the two reactor states use "N" input parameters, consisting of nuclear data for all groups, reaction types, and nuclides. The relative sensitivity coefficients for these data can be expressed as components of the N-dimension column vector $\mathbf{S}_\rho$ and similarly for the absolute sensitivities $\mathbf{S}_\rho$. The relative and absolute reactivity variances-indicated as σ_ρ^2 and σ_ρ^2, respectively-are calculated in TSAR as

$$\sigma_\rho^2 = \frac{\sigma_\rho^2}{\rho^2} = \mathbf{S}_{\boldsymbol{\rho}}^{\mathbf{T}} \mathbf{C}_{\boldsymbol{aa}} \mathbf{S}_{\boldsymbol{\rho}} \quad \text{and} \quad \sigma_\rho^2 = \mathbf{S}_{\boldsymbol{\rho}}^{\mathbf{T}} \mathbf{C}_{\boldsymbol{a\alpha}} \mathbf{S}_{\boldsymbol{\rho}}, \tag{6.7.10}$$

where $\mathbf{C}_{\boldsymbol{aa}}$

is the relative covariance matrix describing nuclear data uncertainties and correlations, which are read from the SCALE covariance libraries. Due to the manner in which the absolute sensitivity coefficient is defined in Eq. (6.4.8), the absolute variance of the reactivity still uses the relative covariance matrix of the nuclear data.

The reactivity variance is related to the uncertainties and correlations in the calculated eigenvalues of the two reactor states. It can be shown that expression in Eq. (6.7.10) for the relative variance in an eigenvalue-difference response is equivalent to

$$\sigma_\rho^2 = \frac{\lambda_1^2}{(\lambda_1 - \lambda_2)^2} \sigma_{k1}^2 + \frac{\lambda_2^2}{(\lambda_1 - \lambda_2)^2} \sigma_{k2}^2 - 2c_{1\to 2} \frac{2\lambda_1\lambda_2}{(\lambda_1 - \lambda_2)^2} \sigma_{k1}\sigma_{k2}. \tag{6.7.11}$$

In the above equations, σ_{k1} and σ_{k2} are relative standard deviations of the multiplication factors and the correlation coefficient between the two reactor states is designated as $c_{1\rightarrow 2} \in [-1, 1]$. The eigenvalue calculations of the two states are correlated because they both use the same nuclear data libraries; therefore, the variance in the eigenvalue-difference is not simply the sum of the variances of the eigenvalues. A positive correlation (i.e., $c_{1\rightarrow 2} > 0$) reduces the uncertainty in the reactivity because common uncertainties tend to cancel from the eigenvalue difference. On the other hand, negative correlations increase the reactivity uncertainty. It also can be seen from Eq. (6.7.11). that whenever the difference in the eigenvalues of the two states is small, the relative variance of the reactivity is substantially greater than the individual eigenvalue variances because the coefficients of σ^2_{k1} and σ^2_{k2} are large. Since this is usually the case for reactivity changes in a reactor, relative uncertainties in reactivity responses tend to be much larger than those for eigenvalues. If the reactivity response is close to zero, it is usually preferable to consider absolute rather than relative uncertainties. Whenever the eigenvalue difference is less than 10^{-10}, TSAR will abort the calculation if a relative uncertainty is specified.

TSAR calculates the variance in the reactivity response using the expressions in Eq. (6.7.10). The square root of the variance corresponds to the standard deviation, which indicates the reactivity uncertainty. TSAR edits this value, as well the individual contributions of each nuclide-reaction pair (including cross correlations) to the overall uncertainty.

6.7.2.3 Cross-section-covariance data

The cross-section-covariance data are read from the COVERX-formatted covariance library identified by *coverx=* in the *PARAMETER* data block on the TSAR input file. Cross-section-covariance data files distributed with SCALE are discussed in Sect. 10.3, and the format of the COVERX data file is presented in Sect. 6.3.4.1.6. The recommended SCALE covariance libraries include uncertainty data for all materials in ENDF/B-VII. These data were obtained from a variety of sources, including ENDF/B-VI and VII, JENDL, and approximate values based on uncertainties in measured integral parameters. [TSARWR08]. The *COVARIANCE* data block can be used to override and/or supplement the library uncertainty values for specified nuclide-reaction pairs. The keyword *use_icov* is entered in the *PARAMETER* data block to utilize the covariance data defined in the *COVARIANCE* data block. Additionally, a default uniform standard deviation value can be assigned for any missing covariance data. This default value is defined by the *udcov=* keyword in the *PARAMETER* data block, and the keyword *use_dcov* is entered to activate the option. Warning messages are printed to identify substituted covariance matrices.

When *use_dcov* and/or *use_icov* and *cov_fix* are specified in the *PARAMETER* data block, and a reaction has zero or large (standard deviation > 1000%) values on the diagonal of the covariance matrix, the diagonal elements and off-diagonal terms are replaced according to the user-input criteria. Warning messages are printed to identify the replaced values. Additional options for user-specified covariance data are given in Sect. 6.7.3.2.

In the reactivity-uncertainty-analysis edit, a single asterisk ("*") identifies uncertainty contributions from nuclide-reaction pairs for which the default cross-section-covariance data is applied. Likewise, the markers ("**"), ("**"), ("**"), denote (a) user-input covariance data, (b) covariance library data replaced by default values, and (c) covariance library data replaced by user-input values, respectively. In the HTML output, these uncertainty contributions are distinguished by using unique HTML colors. The different HTML colors are specified by the HTML block keywords *ud_clr=*, *ui_clr=*, *udfix_clr=*, and *uifix_clr=*. The HTML block keyword options are discussed in more detail in Sect. 6.7.3.3.

TSAR computes a problem-specific covariance library that contains cross-section covariances only for the nuclide-reaction pairs (including cross correlations) listed on the k_1-and k_2-sensitivity data files. This covariance library, referred to as the working covariance library, is written in COVERX format like the

SCALE covariance library. The working covariance library contains any default or user-input covariance data for nuclide-reaction pairs that were not on the input covariance library as well as any modified cross-section-covariance data. The working covariance library can be read by the Javapeño plotting tool to visualize the cross-section-covariance data used in the reactivity uncertainty analysis.

6.7.3 INPUT DESCRIPTION

The user input for TSAR consists of a SCALE Analytic Sequence Specification Record (i.e., *=tsar*), an optional title followed by three blocks of data in free-form keyword format, and a SCALE input termination END record. The data blocks begin with **READ KEYNAME**and end with **END KEYNAME**, where **KEYNAME** is the name of an individual data block. The *PARAMETER* data block is the only required block of data, while the *HTML* and *COVARIANCE* data blocks are optional. The data blocks can be entered in any order.

6.7.3.1 Parameter data

The *PARAMETER* block of data is used to specify the name of the previously prepared k-sensitivity data files, the name of the :math:` ho`-sensitivity data file, and other keyword options that control the code execution. The keyword options are listed in Table 6.7.1 along with their default values and description. A keyword that ends with "=" must be followed by additional data. Keywords that do not end with "=" are Boolean flags that are used to turn on certain features of the code, such as the computation of certain data or certain output edits. If the keyword is present for a Boolean entry, the value is set to true. Otherwise, the value is set to false.

Table 6.7.1: Input data for parameter block of TSAR input.

Keyword	Default value	Description
cov_fix (optional)	False	Replace zero and large (standard deviation > *large_cov*) values on diagonal of cross-section-covariance data with user-input values or default values.
coverx= (optional)	44groupcov	Name of cross-section-covariance data file to use in analysis.
large_cov= (optional)	10.0	Cutoff fractional standard deviation value for *cov_fix*. Covariance data with uncertainties larger than *large_cov* are replaced with user-defined or default values. Default =10, which is 1000% uncertainty.
nocovar (optional)	False	If *nocovar* is present, the reactivity uncertainty calculation is bypassed.
`nohtml` (optional)	False	If `nohtml` is present, HTML-formatted output is not generated.
`print=` (optional)	rho	Available options are `rho*`, `*all`, or `none`. If `print=rho`, *rho*−sensitivities edits are generated. If `print=all`, *rho* − k_1, k_2-, sensitivity edits are generated. The `print=none` option turns off all sensitivity edits.
return_work_cov (optional)	False	If *return_work_cov* is present, the working covariance library is copied to the return directory with the file name *job_name.wrk.cov,* where *job_name* is the name of the input file. If *return_work_cov* is not present, the working covariance library remains in the temporary working directory with the file name *job_name.wrk*.

continues on next page

Table 6.7.1 – continued from previous page

Keyword	Default value	Description
sdf_file_1= (required)	n/a	The file name of the initial state k_1-sensitivity data file.
sdf_file_2= (required)	n/a	The file name of the final state k_2-sensitivity data file.
type= (optional)	relative	Available options are *relative* (or *rel*) and *absolute* (or *abs*). If *type=relative*, the output reactivity sensitivity data file contains relative *rho*−sensitivities. If `type=absolute`, the output reactivity sensitivity data file contains absolute *rho*−sensitivities (i.e., pcm units).
udcov= (optional)	0.05	User-defined default value of standard deviation in cross-section data to use for all groups for nuclide-reaction pairs for which cross section covariance data are too large or not available on the input covariance library.
udcov_corr= (optional)	1.0	User-defined default correlation value to use for nuclide-reaction pairs for which cross section covariance data are not available on the input covariance library.
udcov_corr_type= (optional)	zone	User-defined default correlation in cross-section data to use for nuclide-reaction pairs for which cross-section covariance data are not available on the input SCALE covariance library. Allowed values are *long*, *zone*, and *short*. See Table 6.7.2 for details on *long*, *zone*, and *short.*
udcov_therm= (optional)	0.0	User-defined default value of standard deviation in cross-section data to use for thermal data for nuclide-reaction pairs for which cross-section covariance data are too large or not available on the input covariance library. If input, the *udcov_therm* value overrides the *udcov* value in the thermal range (i.e., neutron energies below 0.625 eV).
udcov_inter= (optional)	0.0	User-defined default value of standard deviation in cross-section data to use for intermediate data for nuclide-reaction pairs for which cross section covariance data are too large or not available on the input covariance library. If input, the *udcov_inter* value overrides the *udcov* value in the intermediate range (i.e., neutron energies above 0.625 eV and below 25 keV).
udcov_fast= (optional)	0.0	User-defined default value of standard deviation in cross-section data to use for fast data for nuclide-reaction pairs for which cross section covariance data are too large or not available on the input covariance library. If input, the *udcov_fast* value overrides the *udcov* value in the fast range (i.e., neutron energies above 25 keV).
use_dcov (optional)	False	Use user-defined default cross-section covariance data for nuclide-reaction pairs not included on the input covariance library.

continues on next page

Table 6.7.1 – continued from previous page

Keyword	Default value	Description
use_icov (optional)	False	Use user-defined cross section covarianc data input in the *COVARIANCE* input data block in place of the default values for user-input nuclide-reaction pairs that are not on the input covariance library.

6.7.3.2 User-input covariance data

The *COVARIANCE* data block described in Table 6.7.2 allows the user to specify a covariance matrix for specific nuclide-reaction pairs for which covariance data are not present on the input covariance library or that have zero or large values on the diagonal. The *COVARIANCE* data block must begin with *READ COVARIANCE* and end with *END COVARIANCE*.

Table 6.7.2: Input data for covariance block of TSAR input.

Input parameter	Requirement	Default value	Allowed values	Description
Nuclide	Required	none	Nuclide name or ZA number	Nuclide for which covariance data are to be entered
Reaction	Required	none	Reaction name or ZA number	Reaction for which covariance data are to be entered. See the TSUNAMI-IP manual for available reaction types.
all=	Optional	0.0	any number	Fractional standard deviation value to be applied to all groups
fast=	Optional	0.0	any number	Fractional standard deviation value to be applied to fast groups. If input, the *fast* value overrides the *all* value in the fast range (i.e., neutron energies above 25 keV).
therm=	Optional	0.0	any number	Fractional standard deviation value to be applied to thermal groups. If input, the *therm* value overrides the *all* value in the thermal range (i.e., neutron energies below 0.625 eV).
inter=	Optional	0.0	any number	Fractional standard deviation value to be applied to intermediate groups. If input, the *inter* value overrides the *all* value in the intermediate range (i.e., neutron energies above 0.625 eV and below 25 keV).
corr=	Optional	1.0	any number from -1.0 to 1.0	Correlation between groups

continues on next page

Table 6.7.2 – continued from previous page

Input parameter	Requirement	Default value	Allowed values	Description
corr_type=	Optional	*zone*	*long, short, zone*	Type of correlation applied from group-to-group covariance values *long* - correlation is applied between all groups *short* - correlation is applied only between adjacent groups *zone* - correlation is applied within fast, intermediate, and thermal groups, but no correlation is applied between zones
end	Required			Denotes end of input for current nuclide/reaction

Any MT number or reaction name will be treated as a valid input, but only those present on the k_1- or k_2-sensitivity data files will impact the results. The reaction sensitivity types computed by SAMS from TSUNAMI-1D and TSUNAMI-3D are described in the TSUNAMI-IP manual. An energy-covariance matrix is created for the specified nuclide-reaction pair with the square of the entered standard deviation for the diagonal terms for all groups using the *all*= value. Groups in the fast, intermediate, and thermal energies are then set to the square of the standard deviation value entered for *fast*=, *inter*=, and *therm*=, respectively. The off-diagonal terms of the energy matrix are generated according to the input for *corr*=, and *corr_type*=, with default settings of *1.0* and *zone.* Data entered in this block are used only for missing data and do not override values on the input covariance data file. The SCALE 5.1 input format is supported where data are entered in triplets with the nuclide name (e.g., u-235), then the reaction MT number or name (e.g., 18 or fission), and then a standard deviation value. In this case, the *end* keyword must not be entered. The standard deviation value is applied to all groups with default setting for correlations. These data are only used if *use_icov* is specified in the *PARAMETER* data block. When both *use_icov* and *cov_fix* are specified in the *PARAMETER* data block, and a reaction has zero or large (standard deviation > 1000%) values on the diagonal of the covariance matrix, these values are replaced with the square of the user input standard deviation value, and the corresponding off-diagonal terms are substituted according to the values of *corr* and *corr_type*.

6.7.3.3 HTML data

The optional *HTML* data block is used to customize HTML-formatted output. The *HTML* data block must begin with *READ HTML* and end with *END HTML*. The data input to the *HTML* data block consists of several keywords that are shown, along with their default values and descriptions, in Table 6.7.3. A keyword that ends with "=" must be followed by text data. For color entries, any valid html color name can be entered or the hexadecimal representation can be used if preceded by a # sign. For example, to change the background color of the html output to white, *bg_clr=white* and *bg_clr=#ffffff* have the same effect, because ffffff is the hexadecimal representation of white. An extensive list of available colors for customized output is shown in Sect. 6.3.4.1.7. Please note that not all features are supported by all browsers.

Table 6.7.3: Input data for html block of TSAR input.

Keyword	Default value	Description
bg_clr=	papayawhip	Background color
h1_clr=	maroon	Color used for major headings
h2_clr=	navy	Color used for sub-headings
txt_clr=	black	Color for plain text
lnk_clr=	navy	Color for hyperlinks
lnk_dec=	none	Decoration for hyperlinks. (none, underline, overline, line-through, blink)
vlnk_clr	navy	Color for visited hyperlinks
ud_clr=	blue	Color for values in tables that use default covariance data
ui_clr=	red	Color for values in tables that use user-input covariance data
udfix_clr=	royalblue	Color for values in tables that use default corrected covariance data
uifix_clr=	green	Color for values in tables that use user-input corrected covariance data

6.7.4 SAMPLE PROBLEM AND OUTPUT DESCRIPTION

6.7.4.1 Input and text output

An example of TSAR input is given in Example 6.7.1 Each section of the text output, not shown, is described in order below.

1. Input Listing and Summary of Calculations - The TSAR input data is printed for each data block. Both user-specified and default values for the various keywords are edited.

2. k1-Sensitivity Data File Summary and Sensitivity Coefficients — The header information for the k_1-sensitivity data files follows the input data. This includes the title on the sensitivity data file, the number of energy groups, the number of sensitivity profiles, and the values of k_1 and λ_1. If `print=all` is specified in the *PARAMETER* data block, the energy, region, and mixture-integrated k_1-sensitivity coefficients are edited for each nuclide-reaction pair.

3. k2-Sensitivity Data File Summary and Sensitivity Coefficients — The header information for the k_2-sensitivity data files follows the k_1-sensitivity data file edit. This includes the title on the sensitivity data file, the number of energy groups, the number of sensitivity profiles, and the values of k_2 and λ_1. If *print=all* is specified in the *PARAMETER* data block, the energy, region, and mixture-integrated k_2-sensitivity coefficients are edited for each nuclide-reaction pair.

4. Reactivity Value and Optional ρ-Sensitivity Coefficients — After the k_1 and k_2 sensitivity edits, the reactivity between the two states is edited in pcm units. If *print=all* or *print=rho* is specified in the *PARAMETER* data block, the energy, region, and mixture-integrated ρ-sensitivity coefficients are edited for each nuclide-reaction pair.

5. Reactivity Uncertainty Analysis — Following the edit of the reactivity value and the optional ρ-sensitivity edit, the reactivity uncertainty analysis is printed on the text output. First, a message is printed that states that the working covariance library is being generated. If *PARAMETER* block keywords `use_dcov`, `use_icov`, and/or `cov_fix` are specified, covariance warnings are printed whenever user-input covariance data are included in the reactivity uncertainty analysis. Following the covariance warnings, the total reactivity uncertainty is printed along with the reactivity uncertainty contributions from individual energy covariance matrices. The reactivity uncertainty contributions are sorted in descending order.

Example 6.7.1: TSAR sample problem input.

```
=shell
  cp $SCALE/output/tsunami-1d1.sdf .
  cp $SCALE/output/tsunami-1d1_147HtoU.sdf .
end
=tsar
tsar example problem
read parameter
  sdf_file_1=tsunami-1d1.sdf
  sdf_file_2=tsunami-1d1_147HtoU.sdf
  use_dcov
  use_icov
  cov_fix
  print=all
  udcov_fast=0.10
  udcov_inter=0.15
  udcov_therm=0.08
  udcov_corr_type=zone
  udcov_corr=1.0
  return_work_cov
end parameter
read html
 bg_clr=Aliceblue
 ud_clr=blue
 ui_clr=read
 udfix_clr=green
 uifix_clr=darkorange
end html
end
```

6.7.5 HTML OUTPUT

The input file for the TSAR sample problem shown in Example 6.7.1 is named tsar1.input. In this case, the HTML-formatted output is stored in a file called tsar1.html and additional resources are stored in directories called tsar1.htmd and applet_resources. This section contains example TSAR HTML-formatted output only for demonstration of the interface. The problem does not correspond to the sample problem distributed with SCALE and is included for illustrative purposes only. When tsar1.html is opened in a web browser, the information shown in Fig. 6.7.1 is displayed.

Fig. 6.7.1: Initial screen from TSAR HTML output.

The title of the input file is displayed between the two SCALE logos. Clicking on the SCALE logo will link the user directly to the SCALE website, if Internet access is available. Because this SCALE input file only executed TSAR, only a single output listing is available. The text "1. TSAR" is a hyperlink to view the output from TSAR. Clicking on the "1. TSAR" hyperlink will bring up the information shown in Fig. 6.7.2.

The initial page of output from TSAR is shown in Fig. 6.7.2. Program verification information is shown in the table under the TSAR logo. This table includes information about the code that was executed and the date and time it was run. The menu on the left side of the screen contains hyperlinks to specific portions of the code output. Echoes of the input data are available in the Input Data section. Any errors or warning messages are available in the Messages sections. Results from the code execution are shown in the results section.

Fig. 6.7.2: Program verification screen from TSAR HTML output.

Selecting Input Parameters will reveal the menu of available input data. Selecting Input Parameters causes the table shown in Fig. 6.7.3 to be displayed. Other input data can also be displayed by selecting the desired data from the menu.

General Information

- Program Verification Information

Input Data

- Input Parameters
- Filenames
- HTML Format Options

Results

tsar - Input Parameters
tsar example problem

PARAMETER	VALUE	DESCRIPTION
cov_fix	true	Replace zero and large values on diagonal of cross-section covariance data with user input values and dcov value.
large_cov=	10.0000	Cutoff fractional standard deviation value for cov_fix.
nocovar	false	Flag to cause the reactivity uncertainty edit to be turned off.
nohtml	false	Flag to cause HTML output to not be produced.
print=	all	If print=all (or nprt=2), Sensitivity edits are generated forthe initial state, final state, and reactivity. If print=rho (or nprt=1), reactivity sensitivities are editted. If print=none (or nprt=0), Sensitivity edits are not produced.
return_work_cov	true	Option to copy the working covariance data file back to the return directory.
type=	Relative	Format of the reactivity sensitivity data file.
udcov=	0.0500	User-defined default value of standard deviation in cross-section data to use for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_corr=	1.0000	User-defined default correlation value to use for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_corr_type=	zone	User-defined default correlation in cross-section data to use for nuclide-reaction pairs for which covariance data are not available on the selected data file. (long, zone, short)
udcov_fast=	0.1000	User-defined default value of standard deviation in cross-section data to use for fast data for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_inter=	0.1500	User-defined default value of standard deviation in cross-section data to use for intermediate data for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_therm=	0.0800	User-defined default value of standard deviation in cross-section data to use for thermal data for nuclide-reaction pairs for which covariance data are not available on the selected data file.
use_dcov	true	Use user-defined default covariance data, udcov, for nuclide reaction pairs not included on the covariance data file.
use_icov	true	Use user-defined data input in COVARIANCE input data block in place of udcov value for user input nuclide-reaction pairs that are not on the covariance data file.
sdf1=	41	Logical unit number for the initial state SDF file (lun1).
sdf2=	42	Logical unit number for the final state SDF file (lun2)

Fig. 6.7.3: Input parameters from TSAR HTML output.

Selecting Results causes a menu of available results to be revealed. From this menu, selecting Reactivity Sensitivities causes a submenu to be revealed. From the submenu, the Reactivity Sensitivities can be visualized in tabular format or plot format. Selecting Energy, Region, and Mixture Integrated Sensitivity Coefficients from this submenu causes the information shown in Fig. 6.7.4 to appear.

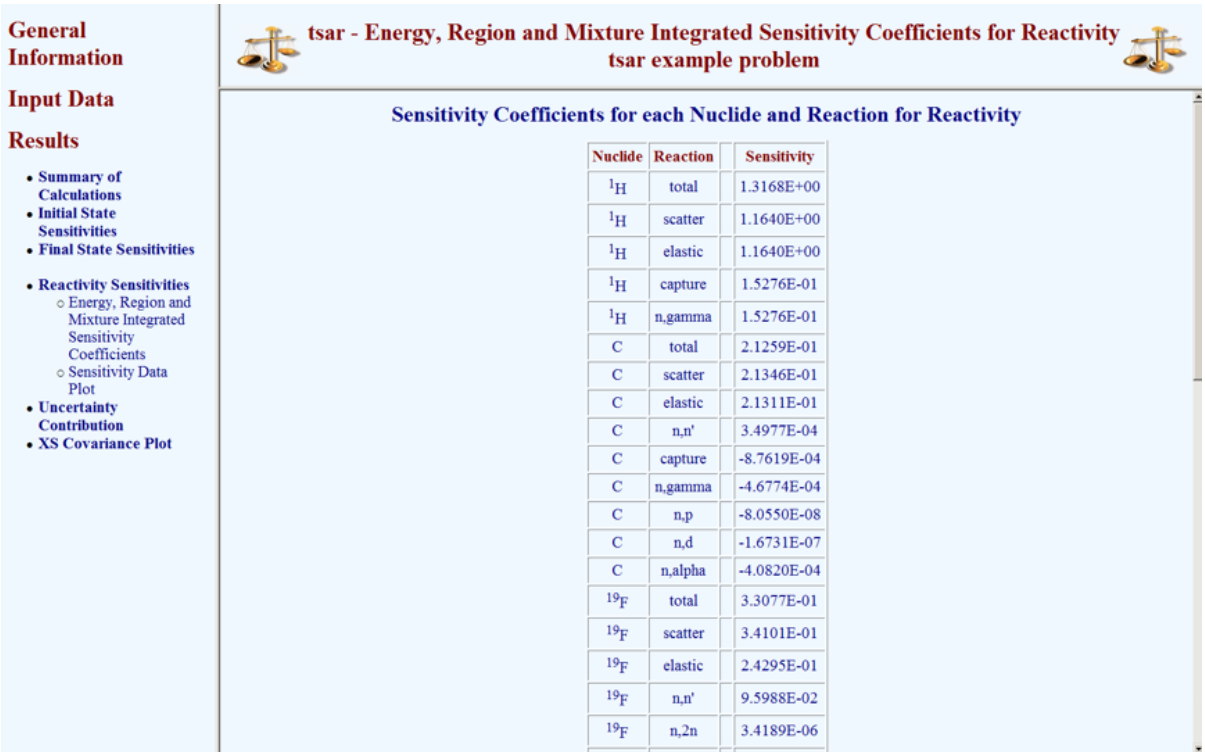


Fig. 6.7.4: Global integral indices from TSAR HTML output.

Plots of sensitivity data from the initial and final states as well as the reactivity sensitivities are available for viewing in the TSAR HTML output. If the *return_work_cov* keyword option is included in the PARAMETER block on the TSAR input file, then the covariance data can be viewed by selecting “XS Covariance Plot” in the results submenu. A Java applet version of Javapeño will appear in the browser window with the working covariance library preloaded. Data can be added to the plot by double-clicking on the list of available data on the right side of Javapeño. The plot shown in Fig. 6.7.5 was produced with this procedure.

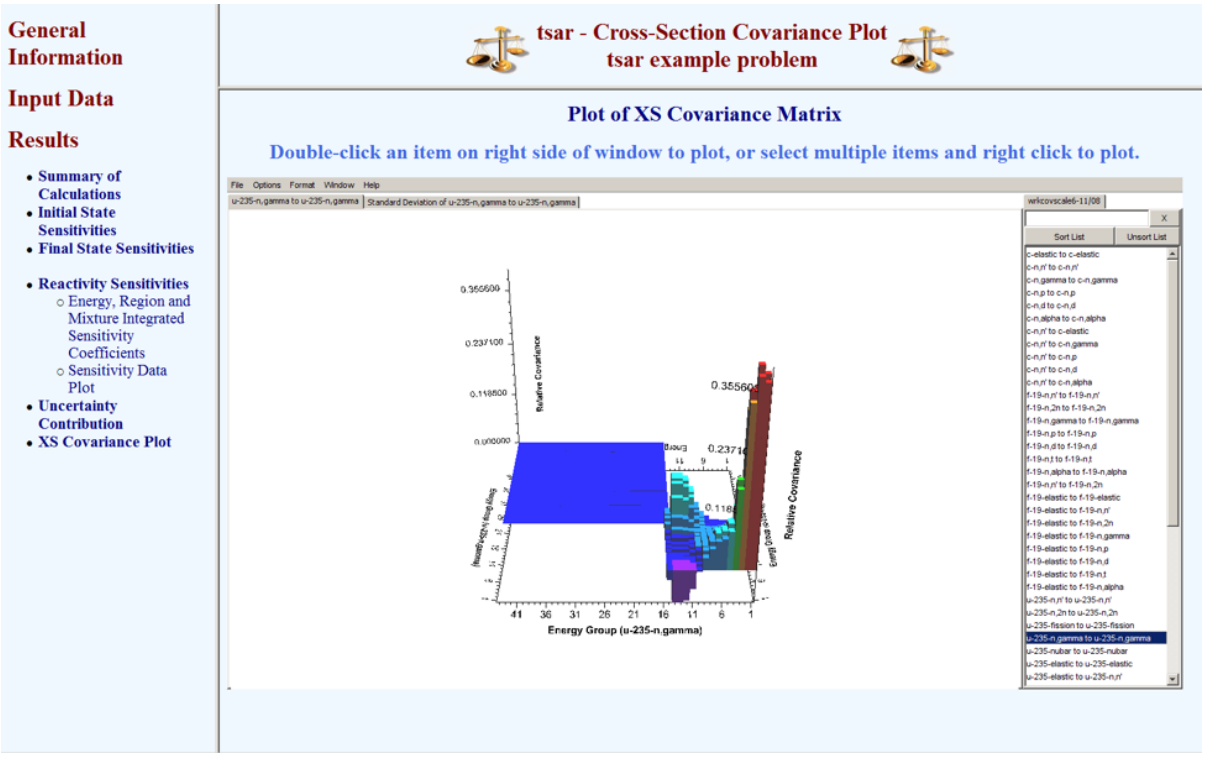


Fig. 6.7.5: Covariance data from TSAR HTML output.

6.8 TSURFER: AN ADJUSTMENT CODE TO DETERMINE BIASES AND UNCERTAINTIES IN NUCLEAR SYSTEM RESPONSES BY CONSOLIDATING DIFFERENTIAL DATA AND BENCHMARK INTEGRAL EXPERIMENTS

M. L. Williams, B. L. Broadhead, M. A. Jessee, J. J. Wagschal1, and R. A. Lefebvre

ABSTRACT

The TSURFER code uses the generalized linear least-squares method to consolidate a prior set of measured integral responses (such as k_{eff}) and corresponding calculated values obtained using the SCALE nuclear analysis code system. The initial estimates for the computed and measured responses are improved by adjusting the experimental values and the nuclear data used in the transport calculations-taking into account their correlated uncertainties-so that the most self-consistent set of data is obtained. This procedure makes the refined estimates of the calculated and measured responses agree within first-order accuracy, while constraining the data variations to minimize a generalized chi-square parameter. Consolidation of the original integral experiment data and calculated results reduces the prior uncertainty in the response estimates because additional information has been incorporated. The method can also address one or more "*application responses*" for which no experimental measurements are available. TSURFER computes an updated estimate for the application responses and provides an estimate for the computational bias and application uncertainty. The methodology is useful in validation studies for criticality safety and reactor analysis.

ACKNOWLEDGMENTS

The authors would like to acknowledge R. L. Childs, formerly of ORNL, for his contributions. Development of the TSURFER code was funded by the U.S. Department of Energy Nuclear Criticality Safety Program.

6.8.1 INTRODUCTION

This report describes the TSURFER code (**T**ool for **S**ensitivity/**U**ncertainty analysis of **R**esponse **F**unctionals using **E**xperimental **R**esults-pronounced "*surfer,*" with silent "T" like TSUNAMI), which is a functional module in the SCALE sensitivity and uncertainty (S/U) analysis methodology (see the TSUNAMI-1D chapter). The main functions of the code are to (a) compute uncertainties in calculated integral responses such as k_{eff}, due to uncertainties in the input nuclear data; (b) reduce discrepancies between the measured and calculated responses by adjusting the nuclear data and experimental values such that the overall consistency is maximized; (c) analyze measured responses from benchmark experiments to establish the bias and associated uncertainty in some application response that has been calculated.

TSURFER utilizes the generalized linear least-squares (GLLS) methodology based on S/U techniques originally developed in the 1970's and 1980's for a variety of applications, including nuclear data evaluation, [TSURFER-PRR+74] fast reactor design studies, [TSURFER-PC88, TSURFER-WML+76] and reactor pressure vessel damage predictions [TSURFER-MWB81]. A recent GLLS application has been in the area of criticality safety analysis, in which critical benchmarks are used to validate the computational methodology for predicting subcritical quantities and configurations of fissile materials [TSURFER-BRH+04]. Similar validation procedures also could be performed for other integral responses of interest for nuclear reactor analysis. These include responses such as reactivity coefficients associated with coolant voiding or Doppler broadening, peak power values, or in-core instrumentation readings. Although not limited to this area, the application of TSURFER to criticality safety validation studies is emphasized here.

6.8.1.1 Application to validation studies

Historically the validity of a calculation performed for some application has been established by considering how well the same calculational methods and nuclear data perform for a set of representative benchmark experiments. While a simple comparison of the computed and experimental results is very informative, it does not fully take advantage of the valuable information provided by the measured integral responses. If the original sets of calculated and experimental responses are consolidated in a consistent manner (i.e., correctly accounting for uncertainties), then the "adjusted" results should be a better estimate for the true responses, since the revised response values are based upon more information than was available in either the original calculations or measurements alone. This is essentially a statement of Bayes Theorem from probability theory, which indicates how prior information (calculated responses) can be evolved by incorporating additional information (integral measurements) into more reliable posterior results (the adjusted responses). The equations used for the GLLS methodology are equivalent to those obtained from Bayes Theorem [TSURFER-Hwa88].

Cross-section libraries for neutron transport calculations are processed from fundamental evaluated nuclear data files such as ENDF/B. Because the "true" values of the nuclear data are not known precisely, it is reasonable to view the ENDF data as being selected from a probability distribution of allowable values. Nuclear data uncertainties are described by covariance matrices that contain variances in the group cross sections for a given nuclide and reaction type, as well as covariances arising from correlations between energy groups, and possibly between reactions and materials. Discrepancies in the ENDF nuclear data caused by uncertainties in the evaluation process propagate to errors in group cross sections, which in general cause computed responses to disagree with the corresponding measured values. The GLLS approach considers potential variations in data parameters and measured integral responses that minimize the differences in measured and calculated integral responses (such as k_{eff}) for a set of benchmark experiments, *taking into account uncertainties and correlations in the ENDF data and in the integral measurements*. Since there are generally many more cross-section values than measured integral responses, the determination of the data modifications is an under-determined problem. If the data variations are not performed in a reasonable

manner, non-physical changes may be obtained. Data adjustments in the GLLS methodology are constrained by the magnitude of the nuclear data uncertainties and their correlations. TSURFER also accounts for uncertainties and correlations in the integral response measurements, arising from experimental uncertainties in parameters such as enrichment, density, impurities, etc. As long as realistic data covariances are used, the resulting data adjustments may be considered the "best estimates"-within the limitations of the GLLS linearity approximation-for realistic data alterations that improve the computed integral responses. It can be shown that the GLLS equations provide the maximum likelihood estimator for the correct nuclear data if the evaluated nuclear data and system parameters obey a multivariate normal probability distribution [TSURFER-Hwa88].

Some previous studies have applied the GLLS methodology to produce an adjusted nuclear data library in order to improve calculations of nuclear reactors with similar characteristics as the experiments used in the adjustment [TSURFER-TTYS88]. In criticality safety analysis this procedure runs the risk of applying the adjusted library to systems beyond the limits for which the data modifications are appropriate. The usual function of TSURFER is not to output an adjusted nuclear data library but rather to obtain an adjusted application response (k_{eff}) and to provide a quantitative estimate for its accuracy. Hence, it is more appropriate to view TSURFER as a tool to establish biases and uncertainties in calculated responses. Nuclear data adjustments are a by-product of this procedure.

Traditional validation of criticality safety calculations estimates the computational bias based on trends in the calculated k_{eff} values versus system parameters such as hydrogen-to-fissile ratios (H/X) or energy of average lethargy causing fission (EALF). These trending parameters are frequently used as measures of "similarity" between critical systems, hence their use as bias-predictors. Recent studies have shown that data sensitivity coefficients, either alone or in combination with cross-section uncertainty information, are good indicators of system similarity. Thus S/U-based indices have also been used in trending analyses, analogously to the commonly used physical parameters.[5] The input data for S/U trending analysis (i.e., calculated and measured responses, sensitivity coefficients, cross-section and experimental uncertainties) are almost identical to those needed for GLLS analysis; therefore, it is not surprising that some aspects of the TSURFER calculations are also used for trending results. However, TSURFER provides an alternative to the trending approach to determine the bias and can address other validation issues. For example, TSURFER is useful when there are few or no existing experiments that are similar to a particular application area, since the GLLS technique can include individual experiments that separately validate portions of the application area, even though none can be considered entirely similar to the application [TSURFER-Gol04].

6.8.1.2 Types of responses

A response corresponds to a particular integral response *type* (e.g., k_{eff}, reaction rate ratio, material worth, radiation dose, etc.) in a particular *nuclear system* (e.g., a benchmark experiment or a proposed storage arrangement of reactor fuel assemblies or a reactor core). In the TSURFER input, responses may be classified either as "*experiments*" or "*applications*" or "*omitted*."

An *experiment* response has both calculated and measured values input to TSURFER, and these play an active role in the GLLS procedure, which minimizes the difference between the two results. A value for the uncertainty in the measured response and any correlations with other experiments are also input for experimental responses. Examples of experimental integral responses are the multiplication factor for the GODIVA critical benchmark experiment, the measured ρ^{28} (ratio of epithermal to thermal capture rate for ^{238}U) in the TRX-1 critical benchmark lattice, or the coolant voiding reactivity in a power reactor.

Applications are responses for which a calculated value is known but no measured value is available. Applications often correspond to hypothetical systems being considered within the context of a design study or a criticality safety analysis for which the computational bias and uncertainty are desired. Examples of application responses are the multiplication factor (subcritical) for a proposed fuel assembly storage rack

or for a shipping cask. An application response plays a passive role in the GLLS procedure. Since the application has no experimental results, it does not impact the active responses included in the consolidation procedure; conversely, the GLLS procedure may modify the calculated application value if it is "similar" to some of the experimental responses. In this case the application response shares similar data sensitivity characteristics with one or more of the active responses and hence will be indirectly affected by the same data adjustments that impact the similar experimental responses. This provides a systematic, well-defined method for utilizing experimental benchmark measurements to establish a bias and uncertainty in the calculation of application response.

A response designated as *omitted* in the TSURFER input neither affects other responses nor is affected by them. These responses are completely isolated from the GLLS procedure. This capability is sometimes useful to easily "turn off" an active system to observe its impact on the application results or on the consistency (chi-square) of the set of remaining experimental responses.

6.8.2 SOURCES OF RESPONSE UNCERTAINTY

Transport calculations of responses such as the neutron multiplication factor inherently have biases and uncertainties due to several factors that can be grouped into three classes:

(A) numerical approximations in the transport code;

(B) system modeling approximations; and

(C) input data uncertainties.

6.8.2.1 Class-A uncertainties (numerical)

Class-A uncertainties are sometimes referred to as "methods uncertainties." In Monte Carlo calculations these may be caused by imperfections in random number generation routines, approximations in techniques for scoring neutron multiplication (e.g., incomplete convergence of fission source distribution, neglect of correlations between generations, etc.), and biases from algorithms used to represent nuclear data and to sample probability distributions, as well as the basic statistical uncertainty that is fundamental to the Monte Carlo method. Deterministic methods have uncertainties from using finite space-energy-direction meshes, truncated (rather than infinite) expansions of functions, incomplete convergence of iterations, and especially self-shielding approximations for the multigroup cross sections. Computational benchmark studies often can establish a reasonable upper limit for these effects, which may be judged either as negligible or as requiring some conservative bias to be applied to the application calculations. Here it is assumed that class-A uncertainties in the calculated response can be made acceptably small (e.g., by running more histories or refining mesh sizes) or at least have been previously quantified and can be bounded by a margin applied to the computation. Hence class (A) uncertainties are considered as systematic "tolerance," and are not addressed in the present GLLS methodology used by TSURFER.

6.8.2.2 Class-B uncertainties (modeling/experimental)

Class-B uncertainties occur because the mathematical model used in the transport computations of an application or an experimental response does not correspond exactly to the "true" system. The response uncertainty caused by modeling effects may be associated with either (i) direct computational simplifications such as omitting or homogenizing some components in the calculation model or (ii) fundamental uncertainties in the material compositions, densities, and dimensions of the experiment. The former are systematic uncertainties similar in effect to Class-A numerical uncertainties and may be addressed in the same manner; that is, by bounding the magnitude of the uncertainty through the applied safety margins. However, the latter are true random uncertainties that in theory have probability distributions, and these may be addressed with the TSURFER code.

Even "clean" critical benchmark experiments have uncertainties in the nominal system parameters-such as fuel enrichment, impurities, densities, critical dimensions, and numerous other components-that may lead to discrepancies in the measured and calculated responses for the system. In TSURFER the impact of these uncertainties is designated as the "experimental uncertainty" in the response, since this uncertainty will be present even if no simplifications or approximations are made in the model used for the transport computation. The terminology is sometimes a source of confusion. For example the measured k_{eff} in a critical experiment usually is known to be unity with a very small uncertainty associated with the long, but finite, stable period. While there is little doubt about k_{eff} for a critical experiment, there may be considerable uncertainty in the system parameter values describing the benchmark configuration. This contribution to the modeling uncertainty could be justifiably considered either "experimental" (because system parameters such as material compositions and dimensions are specified by the experimentalists) or "computational" (because uncertainties in the system parameters affect the calculation model), but in TSURFER they are designated as experimental uncertainties. In any case the uncertainty in each system parameter must be propagated to an uncertainty in the measured response. For a k_{eff} response this may be done experimentally by physically varying the system parameter and measuring the reactivity effect or, more commonly, by performing auxiliary transport calculations to determine the eigenvalue variation. This is discussed in a somewhat more quantitative manner in Sect. 6.8.4.1.

The response uncertainty components associated with the respective modeling uncertainties in system parameters determine the overall experimental uncertainty. Many benchmark experiment descriptions in the *International Handbook of Evaluated Criticality Safety Benchmark Experiments* [TSURFER-Bri06] include information about uncertainties in the system parameters and their estimated impact on the multiplication factor. The standard deviations in k_{eff} due to uncertainties in various system parameters are assigned by the benchmark evaluators based on published or archived experiment descriptions, and sometimes on other considerations.

A complication in specifying experimental uncertainties is how to treat correlations among the experiments. Response correlations in two benchmark experiments may be caused by factors such as use of the same fuel pins and container tank, and common instrumentation (same detectors, hydrometers, etc.). For example, if two different experiments use the same fuel material, then it is not reasonable to conclude that the enrichment in one is too high while the other is too low, even if both differences fall within the specified standard deviation. Reference [TSURFER-WBP01] has shown that these correlations may not be negligible when applying the GLLS technique to a set of benchmark experiments. Only a limited amount of experiment correlation data has been published, but more is expected in future revisions to the *International Handbook of Evaluated Criticality Safety Benchmark Experiments*. TSURFER allows experimental uncertainties caused by uncertainties in system modeling parameters to be input for individual components and correlation coefficients can be specified for the shared system parameters of each response. This approach provides the capability for users to describe the actual sources of benchmark experiment correlations without having to know the overall correlation between two different experiments. See Sect. 6.8.4.1 and Sect. 6.8.4.2.

6.8.2.3 Class-C uncertainties (nuclear data)

In many applications, the major source of uncertainty in the calculated response is due to uncertainties in evaluated nuclear data such as microscopic cross sections, fission spectra, neutron yield (nu-bar), and scattering distributions that are contained in ENDF/B. These arise from uncertainties in experimental nuclear data measurements, as well as uncertainties in the evaluation process itself, which in general combine differential experimental information with nuclear physics theory to generate the basic data in compilations like ENDF/B. Class-C uncertainties are governed by probability distributions. The actual probabilities are unknown, but the evaluated data values are assumed to represent the mean of the distribution, and the evaluated variance represents a measure of the distribution width. Correlations as well as uncertainties in nuclear data

can have a significant impact on the overall uncertainty in the calculated response; thus, it is important to include covariances as well as variances in the TSURFER calculations. The uncertainties in fundamental nuclear data also impact resonance self-shielding of multigroup cross-section values, further contributing to the response uncertainty [TSURFER-WBP01]. In the SCALE S/U methodology the effects of implicit changes in self-shielded cross sections are included in the overall response *sensitivity coefficients*, rather than in the covariance data, so that the fundamental data uncertainties are isolated from problem-specific effects [TSURFER-LKH+08]

Covariance information is currently quite limited in all evaluated nuclear data compilations such as ENDF/B. A more complete library of multigroup uncertainties has been created for SCALE using data from a variety of sources, including ENDF/B-VI and VII, JENDL3.1, and approximate covariances based on uncertainties in measured integral data and nuclear model calculations [TSURFER-LKH+08, TSURFER-WR08]. A detailed description of the SCALE covariance libraries is found in the COVLIB chapter.

The GLLS methodology in TSURFER is mainly concerned with treating Class-C uncertainties due to nuclear data, along with Class-B experimental uncertainties.

6.8.3 ANALYSIS PROCEDURE

6.8.3.1 Functional relation to other SCALE modules

TSURFER is a functional module within the overall SCALE S/U methodology. Other modules in SCALE and outside SCALE perform complementary calculations and provide data files used by TSURFER, as described below.

- PUFF-IV: AMPX code that processes ENDF/B nuclear data covariances and generates multigroup covariance data; creates nuclear data uncertainty files for input to TSURFER.
- TSUNAMI-1D/2D/3D SCALE control sequence that computes sensitivity coefficients for k_{eff} or other responses in a 1D/2D/3D model of the experiment or application system; creates sensitivity files used by TSURFER.
- TSAR: SCALE functional module that computes sensitivity coefficients for eigenvalue-difference responses such as reactor reactivity parameters, using k_{eff} sensitivities from a TSUNAMI sequence; creates sensitivity files used by TSURFER.
- TSUNAMI-IP SCALE functional module that computes similarity and completeness indices for a set of responses with sensitivity coefficients. Prior to running TSURFER, it may be advantageous to perform scoping studies with TSUNAMI-IP to determine if the selected set of benchmark experiments provides adequate "coverage" for data uncertainties that have a significant impact on the application response.

6.8.3.2 Guidelines for TSURFER analysis

Both active and passive responses may be included in the TSURFER calculation. For example, in criticality safety validation procedures, responses of interest typically correspond to the system multiplication factor, k_{eff}. The desired sub-critical design configuration would be a passive application system, while the critical benchmarks used for validation are active experiment systems. TSURFER determines the application bias and uncertainty by propagating data variations obtained from GLLS analysis of the active systems to the calculated multiplication factor of the passive system. The bias represents the change of the application's original k_{eff} as a result of the consolidation of all the active critical benchmark experiments and the adjusted nuclear data parameters. An increase in k_{eff} computed for the applications system response indicates that the calculated value was initially too low (a negative bias), and a decrease that the application's multiplication factor was too high (a positive bias).

TSURFER also computes uncertainties in the initial and adjusted estimates for the system responses (e.g., multiplication factors). These response uncertainties include effects of experimental uncertainties (class-B) and nuclear data uncertainties (class-C) but not the impact of simplifications made to the experiment specifications and numerical approximations (class-A)-which should be included in the safety margin.

Several quantities can be examined to give confidence to the predicted results [TSURFER-BHCP99, TSURFER-BHP99]. The first is the completeness parameter, R, given in the output of the TSUNAMI-IP code. It has been suggested that values of R greater than about 0.7-0.8 for a set of active experiments indicate adequate cross-section coverage for an application response in the GLLS procedure; however, this is a preliminary conclusion that may change as more experience is gained [TSURFER-Gol04].

The *chi-square* (χ^2) statistic indicates the overall consistency of the suite of benchmarks and is key to proper interpretation of the TSURFER results. The value of χ^2 per degree of freedom represents the average discrepancy between the calculated and measured responses, expressed in units of the combined variance of the calculation plus experiment. Values of chi-square per degree of freedom ideally should be within about ±20% of unity, indicating that the calculations and measurements on average agree within about one standard deviation. Results in which this test is not strictly met may still be valid, but in general these should be viewed with skepticism unless the reasons for the test failure are understood. An excessively large χ^2 can lead to unreliable results in the GLLS adjustment. TSURFER provides the total χ^2 value, as well as estimated contributions from each experiment (see Sect. 6.8.4.2.1). Individual χ^2 values suggest which experiments may contain inconsistencies (i.e., the difference between the measured and calculated k_{eff} is larger than their combined uncertainties).

Several methods can be used to improve the initial value of chi-square. One approach is to reevaluate the measurement uncertainties and their correlations for identified discrepant experiments. If the experimental or data uncertainties are underestimated, the data adjustments will correspond to an excessive number of standard deviations, as reflected in high χ^2 values. Values of χ^2 that are too low suggest that the input uncertainty estimates may be too high, and again a reevaluation should be considered. Thus it is quite important to utilize *realistic* (not conservative) estimates for the uncertainties in nuclear data and experimental measurements.

Even when best estimates are used for all input uncertainties, it is not uncommon to encounter a few active responses that are inconsistent with the others, especially when dealing with a large number of benchmark experiments. In this case the best alternative to improve χ^2 is to remove the outliers, either by transforming those experiments into passive responses or by omitting them entirely from the GLLS adjustment. TSURFER provides a "chi-square filtering" procedure that automatically omits inconsistent experiments until a specified target value of chi-square is achieved. Several options are provided to select the experiments to be omitted, as discussed in Sect. 6.8.4.2.1. Omitted experiments should be examined to ensure that simple errors in the problem description are not present.

An internal consistency test such as described in [TSURFER-BHCP99] also may be useful. The consistency test is performed by changing one of the benchmark experiments that is similar (see Sect. 6.8.4.3) to the application response into a passive, pseudo-application response. The predicted bias for this passive response should be close to that of the original application; furthermore, in this case the bias prediction can be checked because this passive response actually has a measured experimental value.

6.8.3.3 Required data for TSURFER

Active and passive responses considered in the GLLS analysis should have sensitivity data provided for each nuclide-reaction pair that significantly impacts the response. The sensitivity coefficients are pre-calculated using other SCALE modules as described in Sect. 6.8.3.1 and are stored in individual files for each response included in the TSURFER analysis. The sensitivity data files must be in one of the SCALE sensitivity data formats described in Data File Formats. The locations of the sensitivity files are specified in the TSURFER input data. It is not required for all sensitivity files to have the same group structures; for example, the sensitivity coefficients for one response may have been computed using a 238-group cross-section library, while sensitivities for another response could have a 44-group structure. Whatever the group structure of the sensitivity data, it is mapped into the same group structure as the covariance file. At present the SCALE covariance files use the SCALE 56-group structure by default.

A file of nuclear data covariances also must be input to the TSURFER calculation. The covariance data file must be in COVERX format described in the COVERX format section. SCALE includes a comprehensive applications-oriented covariance library that includes evaluated covariances taken from ENDF/B-VII, ENDF/B-VI, and JENDL nuclear data files [TSURFER-WR08] described in the COVLIB chapter. Ideally, the covariance files should contain data for all nuclide-reaction pairs on the response sensitivity data files. However, cross-section covariance data are not available for all nuclide-reaction pairs. Nuclide-reaction pairs without available covariance data are omitted from the GLLS analysis, but it is assumed that either the cross-section covariance data values for these pairs are well known (i.e., small uncertainties) or that the sensitivity to these nuclide-reaction pairs is small. Where these assumptions hold, the cross sections for these nuclide-reaction pairs should not be adjusted and can be omitted from the GLLS analysis. For situations where these assumptions are judged to be invalid, the use of GLLS analysis is not appropriate. However, TSURFER provides several input options to define uncertainty values for nuclide-reaction pairs with missing covariance data to assess the impact of the additional covariance data on the GLLS analysis. These input options are discussed in more detail in Sect. 6.8.5.1 and Sect. 6.8.5.3.

6.8.4 TSURFER COMPUTATION METHODOLOGY

A recent detailed derivation of the GLLS formalism is given in [TSURFER-BHCP99]. The general formalism allows cross correlations between the initial integral experiment measurements and the original nuclear data, such as would be present if the calculations used a previously "adjusted" library of nuclear data. Since this is not normally done in SCALE, correlations between the benchmark experiment measurements and the cross-section data in the multigroup libraries are not considered in the TSURFER code; therefore, the GLLS equations presented here are somewhat simplified compared to the more general expressions in [TSURFER-Wil86].

At present, the SCALE cross-section-covariance data files characterize nuclear data uncertainties in terms of relative covariances. Therefore, response sensitivity data in TSURFER are defined in terms of relative changes in the nuclear data. An *absolute* response sensitivity is defined as an absolute change in response due to a relative change in the nuclear data, that is,

$$\tilde{S}_{\alpha}=\alpha\frac{\partial \mathrm{R}}{\partial \alpha} \tag{6.8.1}$$

In this equation, R represents the response, α represents the nuclear data, and the tilde will be used to represent absolute sensitivity and uncertainty data. Likewise, a *relative* response sensitivity is defined as a relative change in response due to a relative change in the nuclear data, that is,

$$\mathrm{S}_{\alpha}=\frac{\alpha}{\mathrm{R}}\frac{\partial \mathrm{R}}{\partial \alpha} \tag{6.8.2}$$

The initial development that follows is for *relative*, rather than *absolute*, response sensitivity and uncertainty parameters. It is then shown how to express the quantities in absolute form for reactivity analysis and mixed relative-absolute form for combined k_{eff} and reactivity analysis. A summary of the notation and definitions used in this section can be found in TSURFER Appendix A: Sensitivity/Uncertainty Notation.

The methodology consists of calculating values for a set of I integral responses (k_{eff}, reaction rates, etc.), some of which have been measured in selected benchmark experiments. Responses with no measured values are the selected "applications," described previously. The set of measured response values $\{m_i ; i=1,2,\ldots, I\}$ can be arranged into an I-dimension column vector designated as **m**. By convention the (unknown) experimental values corresponding to applications are represented by the corresponding calculated values. As discussed in Sect. 6.8.2.2, the measured integral responses have uncertainties—possibly correlated—due to uncertainties in the system parameter specifications. The I × I covariance matrix describing the relative experimental uncertainties is defined to be $\mathbf{C_{mm}}$.

The calculated integral values for each experiment and application are obtained by neutron transport calculations, producing a set of calculated responses $\{k_i ; i=1,2,\ldots, I\}$ arranged in a I-dimension vector **k**. The multigroup cross-section data for all nuclide-reaction pairs used in the transport calculations of all responses comprise a set $\{\alpha_n ; n=1,2,\ldots, M\}$, where M is the number of unique nuclide-reaction pairs multiplied by the number of energy groups. It is convenient to arrange these data into a M-dimensional column vector α, so that the dependence of the initial calculated responses upon the input nuclear data values can be indicated as $\mathbf{k} = \mathbf{k}(\alpha)$. The prior covariance matrix for the nuclear data is equal to the M × M matrix $\mathbf{C}_{\alpha\alpha}$, which contains relative variances along the diagonal and relative covariances in the off-diagonal positions. These data describe uncertainties in the *infinitely dilute* multigroup cross sections.

Nuclear data uncertainties cause uncertainties in the calculated response values. In general, these uncertainties are correlated because the same nuclear data library is used for all the transport calculations. The covariance matrix describing uncertainties in the calculated responses due to class-C uncertainties is designated as $\mathbf{C_{kk}}$. Using expressions for propagation of error (the so called sandwich rule), the following relationship is obtained for the relative uncertainty in the calculated responses:

$$\mathbf{C_{kk}} = \mathbf{S_{k\alpha}} \mathbf{C_{\alpha\alpha}} \mathbf{S^T_{k\alpha}} \tag{6.8.3}$$

where $\mathbf{S_{k\alpha}}$ is the relative sensitivity matrix, whose (i, n) element is equal to the relative sensitivity of the i_{th} response with respect to the n_{th} data value, that is, $\frac{1}{R_i}\frac{\partial R_i}{\partial \alpha_n}\alpha_n$. Sensitivity coefficients appearing in the sensitivity matrix are computed using first-order perturbation theory. A description of the equations used to compute sensitivity coefficients using first order perturbation theory can be found in [TSURFER-Wil86] or in the SAMS chapter. In the SCALE methodology, the sensitivity coefficients consist of an "explicit" component that accounts for the direct impact of the data on the neutron transport calculation, as well as an "implicit" component that accounts for its impact on other self-shielded multigroup data [TSURFER-WBP01]. For example, a variation in the hydrogen multigroup elastic cross section has an explicit effect on k_{eff} through its impact on neutron moderation and leakage in the transport solution and has an implicit effect on the self-shielded ^{238}U multigroup cross sections, which causes additional change in k_{eff}. Because self-shielding effects are addressed through the sensitivity coefficients, the nuclear data uncertainties in the covariance matrix correspond to the infinitely dilute values.

Each row i of the sensitivity matrix contains sensitivity coefficients for all nuclear data used in the transport calculation of response i. These data also can be arranged into an M-component sensitivity *vector* $\mathbf{S}_i$ for a particular response "i", which may be either an experiment or application. For example, the sensitivity vector $\mathbf{S}_i$ is an M dimensional vector whose n^{th} element is equal to the sensitivity coefficient of response "i" to data element α_n as specified previously.

It is often convenient to express covariance matrices in terms of standard deviations [represented as σ_{i} for variable *i*] and correlation coefficients [represented by $\rho_{i,j}$ for the variable pair (*i,j*)]. The correlation coefficient is related to the corresponding covariance value by the equation

$$\rho_{\mathrm{i,j}} = \frac{\mathrm{Cov(i,j)}}{\sigma_{\mathrm{i}}\sigma_{\mathrm{j}}} \tag{6.8.4}$$

Correlation coefficients, which have values between -1 and 1, indicate the degree of correlation between the pair of variables, where a value of 1.0 indicates full correlation, 0.0 no correlation, and -1.0 full anti-correlation. Using matrix notation, relative standard deviations are arranged in a diagonal matrix $\boldsymbol{\sigma}$ and the correlation coefficients in a square matrix $\mathbf{R}$ (symmetrical, but generally non-diagonal). The previously defined covariance matrices can be expressed as follows:

$$\mathbf{C_{mm}} = \sigma_{\mathbf{m}}\mathbf{R_{mm}}\sigma_{\mathbf{m}} \tag{6.8.5}$$

$$\mathbf{C}_{\boldsymbol{\alpha\alpha}} = \sigma_{\boldsymbol{\alpha}}\mathbf{R}_{\boldsymbol{\alpha\alpha}}\sigma_{\boldsymbol{\alpha}} \tag{6.8.6}$$

$$\mathbf{C_{kk}} = \sigma_{\mathbf{k}}\mathbf{R_{kk}}\sigma_{\mathbf{k}} \tag{6.8.7}$$

Eq. (6.6.3) and Eq. (6.8.7) can be substituted into Eq. (6.8.3) and rearranged to relate the nuclear data correlations to the correlations in the computed responses:

$$\mathbf{R_{kk}} = \left[\boldsymbol{\sigma}_{\mathbf{k}}^{-1}\ \mathbf{S}_{\mathbf{k}\boldsymbol{\alpha}}\boldsymbol{\sigma}_{\boldsymbol{\alpha}}\right]\mathbf{R}_{\boldsymbol{\alpha\alpha}}\left[\boldsymbol{\sigma}_{\mathbf{k}}^{-1}\ \mathbf{S}_{\mathbf{k}\boldsymbol{\alpha}}\sigma_{\boldsymbol{\alpha}}\right]^{\mathbf{T}}. \tag{6.8.8}$$

The bracketed term on the right side of the above equation is an I × M matrix whose elements equal the number of relative standard deviations ($\boldsymbol{\sigma}_{\mathbf{k}}$) that the response changes, due to a one relative standard deviation change in the nuclear data ($\boldsymbol{\sigma}_{\boldsymbol{\alpha}}$). Even if the nuclear data are not correlated—that is, $\mathbf{R}_{\boldsymbol{\alpha\alpha}}$ is an identity matrix—$\mathbf{R_{kk}}$ is generally not diagonal.

The expressions thus far describe *relative* response sensitivities and covariances (i.e., uncertainties). Similar expressions can also be derived for absolute quantities. In this report, absolute response sensitivities and covariances are denoted by a tilde, such as $\tilde{\mathbf{C}}_{\mathbf{mm}}$, $\tilde{\mathbf{C}}_{\mathbf{kk}}$, and $\tilde{\mathbf{S}}_{\mathbf{k}\boldsymbol{\alpha}}$, which are explicitly defined in Appendix A.

TSURFER allows for a mixed relative and absolute response sensitivities and covariances to be used in the analysis. In the TSURFER input (described in Sect. 6.8.5), each response is designated as an *absolute*-formatted response or a *relative*-formatted response using the input keywords *"absolute"* or *"relative"*. This flexibility allows for the simultaneous use of both relative-formatted k_{eff} sensitivity data (generated by TSUNAMI modules) and absolute-formatted eigenvalue-difference sensitivity data (generated by TSAR) in the same analysis. In this report, mixed relative-absolute response sensitivities and covariances are denoted by a caret, such as $\hat{\mathbf{C}}_{\mathbf{mm}}$, $\hat{\mathbf{C}}_{\mathbf{kk}}$, and $\hat{\mathbf{S}}_{\mathbf{k}\boldsymbol{\alpha}}$.

6.8.4.1 Representation of experimental uncertainty components

Experimental uncertainties (i.e., type-B uncertainties as described in Sect. 6.8.2.2) may be entered directly in the TSURFER input, or alternatively, it may be specified by defining individual "uncertainty components." The latter approach is useful in defining experimental correlations between measured responses. In this case, an index " ℓ " is introduced to identify the response uncertainty components associated with a particular system parameter, p_{ℓ}. For example, the measured k_{eff} uncertainty components for a particular critical experiment consisting of uranyl nitrate dissolved in water might correspond to the following eight p_{ℓ} contributors, as identified by the value of (ℓ): (1) isotopic composition; (2) fuel concentration in the solution; (3) solution density; (4) excess acid concentration in the solution; (5) fuel impurities; (6) dimension of the solution tank; (7) thickness of the solution tank; (8) composition of the tank [TSURFER-Bri06].

The relative standard deviation of a measured response m_i due to an uncertainty in the system parameter p_ℓ is designated as the uncertainty component $\sigma_{m,i}^{(\ell)}$. Assuming that uncertainties in system parameters are uncorrelated, the response uncertainty component is related to the uncertainty in system parameter p_ℓ by the expression

$$\sigma_{m,i}^{(\ell)} = \frac{1}{m_i}\left(\frac{\partial k_i}{\partial p} p\right)\sigma_p = S_{m_i p}\sigma_p, \tag{6.8.9}$$

where σ_{p_ℓ} is the relative standard deviation of system parameter p:$_\ell$ and $S_{m_i p_\ell}$ is the relative sensitivity coefficient relating p_ℓ to the measured response m_i. In principle, the system parameter values and uncertainties could be treated directly in the TSURFER calculation by providing the sensitivity coefficients $S_{m_i p_\ell}$, thus allowing the experiment parameters to be included in the GLLS adjustment. However, at the present time the response uncertainty components $\boldsymbol{\sigma}_{m,i}^{(\ell)}$ must be determined prior to the TSURFER calculation and are read into TSURFER. Values for the response uncertainty components sometimes can be found in the benchmark experiment specifications [TSURFER-Gol04], or auxiliary sensitivity analysis may be necessary. The relative experimental standard deviation of the i_{th} measured response is calculated from

$$\sigma_{m,i} = \sqrt{\sum_1 \left(\sigma_{m,i}^{(1)}\right)^2} \tag{6.8.10}$$

The (i,i) diagonal element of the relative covariance matrix corresponds to the relative experimental variance in response i, which is equal the square of $\sigma_{m,i}$ above. Note that similar expressions can be derived for uncertainty components using absolute sensitivities and uncertainties. For absolute-formatted responses, the uncertainty components on the TSURFER input must be entered in terms of absolute standard deviation. This is discussed in more detail in Sect. 6.8.5.2.

If a different response j is measured in a benchmark system that shares some or all of the same uncertainty components as response i, then the two experiment responses have correlated uncertainties. In such a case the (i, j) element of the relative covariance matrix $\mathbf{C_{mm}}$ is equal to

$$C_{mm}(i,j) = \sum_\ell \left(\sigma_{m,i}^{(\ell)}\sigma_{i,j}^{(\ell)}\sigma_{m,j}^{(\ell)}\right) \tag{6.8.11}$$

and the total correlation coefficient for response pair (i, j) is

$$R_{mm}(i,j) = \frac{C_{mm}(i,j)}{\sigma_{m,i}\sigma_{m,j}} = \frac{\sum\left(\sigma_{m,i}^{(\ell)}\rho_{i,j}^{(\ell)}\sigma_{m,j}^{(\ell)}\right)}{\sigma_{m,i}\sigma_{m,j}} \tag{6.8.12}$$

where $\rho_{i,j}^{(\ell)}$. is the correlation coefficient for responses i and j due uncertainty component ℓ.

TSURFER allows the user to input text-identifiers for the various experiment uncertainty components, along with the associated values for relative standard deviations, $\sigma_{m,i}^{(\ell)}$. Response correlation coefficients $\rho_{i,j}^{(\ell)}$ can be input for each type of uncertainty component, by response pair (i, j).

The previous discussion applies only to *experiment responses* for which measurements have been performed. In the case of an *application response* for which no experimental measurement is known, the uncertainty is set internally by TSURFER to the large value of 10^{10}, to approximate the "infinite" uncertainty in the unknown measurement, and correlations to other responses are set to zero. The large uncertainty for an application response has the effect of letting the response "float" in a passive manner; that is, the application response has a negligible effect on the adjustment of active responses, but the GLLS consolidation of the active responses with finite uncertainties can impact the adjusted value for the application.

6.8.4.2 Generalized linear least-squares equations

Discrepancies in the calculated and measured responses are defined by the I dimensional column vector

$$\mathbf{d} = \left\{ \mathrm{d}_i = \frac{\mathrm{k_i}(\alpha) - \mathrm{m}_i}{\mathrm{k_i}(\alpha)}, i = 1, \ldots, I \right\} \tag{6.8.13}$$

In TSURFER the components of **d** corresponding to application responses are set to zero because applications have no measured values. Using the standard formula for propagation of error and assuming no correlations between k and m, the relative uncertainty matrix for the discrepancy vector **d** can be expressed as the I by I matrix:

$$\mathbf{C}_{\mathrm{dd}} = \mathbf{C_{kk}} + \mathbf{F_{m/k}} \mathbf{C_{mm}} \mathbf{F_{m/k}} = \mathbf{S_{k\alpha}} \mathbf{C}_{\alpha\alpha} \mathbf{S}^{\mathbf{T}}_{\mathbf{k}\alpha} + \mathbf{F_{m/k}} \mathbf{C_{mm}} \mathbf{F_{m/k}} \tag{6.8.14}$$

where the expression in Eq. (6.8.3) was substituted for $\mathbf{C_{kk}}$, and $\mathbf{F_{m/k}}$ is an I × I diagonal matrix containing m/k factors, that is, $\frac{E}{C}$ factors (ratio of experimental to calculated response values). The inverse of the matrix $\mathbf{C_{dd}}$ appears in several expressions presented later in this section. In TSURFER the inversion is performed using routines from the LINPACK software package.

The goal of the GLLS method is to vary the nuclear data ($\alpha \rightarrow \alpha'$) and the measured integral responses ($\mathrm{m} \rightarrow \mathrm{m}'$), such that they are most consistent with their respective uncertainty matrices, $\mathbf{C}_{\alpha\alpha}$ and $\mathbf{C_{mm}}$. This is done by minimizing chi-square, expressed as

$$\begin{aligned} \chi^2 &= \left[\frac{\boldsymbol{\alpha}' - \boldsymbol{\alpha}}{\boldsymbol{\alpha}} \right]^{\mathrm{T}} \mathbf{C}^{-1}_{\alpha\alpha} \left[\frac{\boldsymbol{\alpha}' - \boldsymbol{\alpha}}{\boldsymbol{\alpha}} \right] + \left[\frac{\mathbf{m}' - \mathbf{m}}{\mathbf{m}} \right]^{\mathrm{T}} \mathbf{C}^{-1}_{\mathbf{mm}} \left[\frac{\mathbf{m}' - \mathbf{m}}{\mathbf{m}} \right] \\ &= [\boldsymbol{\Delta\alpha}]^{\mathrm{T}} \mathbf{C}^{-1}_{\alpha\alpha} [\boldsymbol{\Delta\alpha}] + [\boldsymbol{\Delta\mathbf{m}}]^{\mathrm{T}} \mathbf{C}^{-1}_{\mathbf{mm}} [\boldsymbol{\Delta\mathbf{m}}] \end{aligned} \tag{6.8.15}$$

where $\Delta\alpha_i = \frac{\alpha'_i - \alpha_i}{\alpha_i}$ and $\Delta \mathrm{m}_i = \frac{\mathrm{m}'_i - \mathrm{m}_i}{\mathrm{m}_i}$. Eq. (6.8.15) is rearranged to give

$$\chi^2 = \left[\boldsymbol{\sigma}^{-1}_{\boldsymbol{\alpha}} \boldsymbol{\Delta\alpha} \right]^{\mathbf{T}} \mathbf{R}^{-1}_{\alpha\alpha} \left[\boldsymbol{\sigma}^{-1}_{\boldsymbol{\alpha}} \boldsymbol{\Delta\alpha} \right] + \left[\boldsymbol{\sigma}^{-1}_{\mathbf{m}} \boldsymbol{\Delta\mathbf{m}} \right]^{\mathbf{T}} \mathbf{R}^{-1}_{\mathbf{mm}} \left[\boldsymbol{\sigma}^{-1}_{\mathbf{m}} \boldsymbol{\Delta\mathbf{m}} \right] \tag{6.8.16}$$

Eq. (6.8.16) expresses the variations in the nuclear data and measured responses in units of their respective standard deviations; that is, $\left[\boldsymbol{\sigma}^{-\mathbf{1}}_{\boldsymbol{\alpha}} \boldsymbol{\Delta\alpha} \right]$ and $\left[\boldsymbol{\sigma}^{-1}_{\mathbf{m}} \boldsymbol{\Delta\mathbf{m}} \right]$

Chi-square is a quadratic form indicating the squared magnitude of the combined data variations with respect to their uncertainties. This is easily seen for the simple case in which $\mathbf{R}_{\alpha\alpha}{}^{-1}$ and $\mathbf{R_{mm}}^{-1}$ in Eq. (6.8.16) are identity matrices, so that Eq. (6.8.16) reduces to just the diagonal contributions:

$$\chi^2 \rightarrow \sum_{\mathrm{n}=1}^{\mathrm{M}} \left(\frac{\alpha'_n - \alpha_n}{\sigma_{\alpha_n}} \right)^2 + \sum_{\mathrm{i}=1}^{\mathrm{I}} \left(\frac{\mathrm{m}'_\mathrm{i} - \mathrm{m}_\mathrm{i}}{\sigma_{\mathrm{m}_i}} \right)^2 \tag{6.8.17}$$

The first term on the on the right side of Eq. (6.8.17) is equal to the sum of the squares of the individual nuclear data variations expressed in units of their standard deviations while the second term represents a similar quantity for the measured integral responses. In the general case where correlations exist, the inverse matrices in Eq. (6.8.16) are not diagonal, and the value of chi-square must be evaluated using the indicated matrix multiplication.

Thus it can be seen that the GLLS method determines adjustments in the nuclear data and experimental measurements that (a) make the calculated and measured responses agree [i.e., $k' = k'(\alpha') = m'$ within the limitations of first-order sensitivity theory], and (b) minimize Eq. (6.8.15) so that the adjustments are most

consistent with the data uncertainties. Although many possible combinations of data variations may make $k' = m'$, there is a unique set that also minimizes χ^2.

Note: In TSURFER the term "chi-square" normally is meant to signify the minimum value of the quadratic form in Eq. (6.8.15). The significance of this minimum value is discussed in Sect. 6.8.4.2.1.

The following variations minimize Eq. (6.8.15), subject to the constraint $k'(\alpha') = m'$ [TSURFER-Wil86] and the linearity condition $[\boldsymbol{\Delta k}] = \mathbf{S_{k\alpha}}[\boldsymbol{\Delta\alpha}]$ where $\Delta k_i = \frac{k_i' - k_i}{k_i}$:

$$\boldsymbol{\Delta\alpha} = -\left[\mathbf{C_{\alpha\alpha}} \mathbf{S_{k\alpha}^T} \mathbf{C_{dd}^{-1}}\right] \mathbf{d} \tag{6.8.18}$$

$$\boldsymbol{\Delta m} = \left[\mathbf{C_{mm}} \mathbf{F_{m/k}} \mathbf{C_{dd}^{-1}}\right] \mathbf{d} \tag{6.8.19}$$

In the above equations the initial response discrepancy vector $\mathbf{d}$ is operated on by the transformation matrix in square brackets to obtain the desired variations in nuclear data and integral measurements; thus, it is the discrepancy components that drive the adjustments. If the linearity assumption is valid, then the changes in the calculated responses are found to be

$$\boldsymbol{\Delta k} = \mathbf{F_{m/k}} \boldsymbol{\Delta m} - \mathbf{d} = \mathbf{S_{k\alpha}} \boldsymbol{\Delta\alpha}. \tag{6.8.20}$$

Eq. (6.8.20) relates the adjustments in calculated responses, measured responses, and nuclear data.

As previously discussed, consolidation of the calculated and measured responses reduces the prior uncertainties for α', $\mathbf{m}$, and $\mathbf{k}$ because additional knowledge has been incorporated. This is indicated by their modified covariance matrices $\mathbf{C_{\alpha'\alpha'}}$, $\mathbf{C_{m'm'}}$, $\mathbf{C_{k'k'}}$, respectively, given by [TSURFER-Wil86]

$$\mathbf{C_{\alpha'\alpha'}} = \mathbf{C_{\alpha\alpha}} - \left[\mathbf{C_{\alpha\alpha}} \mathbf{S_{k\alpha}^T} \mathbf{C_{dd}^{-1}} \mathbf{S_{k\alpha}} \mathbf{C_{\alpha\alpha}}\right] \tag{6.8.21}$$

$$\mathbf{C_{m'm'}} = \mathbf{C_{mm}} - \left[\mathbf{C_{mm}} \mathbf{F_{m/k}} \mathbf{C_{dd}^{-1}} \mathbf{F_{m/k}} \mathbf{C_{mm}}\right] \tag{6.8.22}$$

$$\mathbf{C_{k'k'}} = \mathbf{C_{kk}} - \left[\mathbf{C_{kk}} \mathbf{C_{dd}^{-1}} \mathbf{C_{kk}}\right]. \tag{6.8.23}$$

If all the responses on the TSURFER input are relative formatted, then the adjusted data and response values edited by TSURFER are obtained from Eq. (6.8.18)-Eq. (6.8.20), while the square roots of diagonal elements in Eq. (6.8.21)-Eq. (6.8.23) correspond to the edited values for adjusted uncertainties in the nuclear data and in the experiment responses, respectively.

The adjustment formulas must be modified slightly to be consistent with the absolute-formatted responses. In the following expressions, absolute response covariance and response sensitivity data are denoted by a tilde [see Appendix A.]:

$$\tilde{\mathbf{d}} = \mathbf{k}(\alpha) - \mathbf{m} \tag{6.8.24}$$

$$\tilde{\mathbf{C}}_{\mathbf{dd}} = \tilde{\mathbf{C}}_{\mathbf{kk}} + \tilde{\mathbf{C}}_{\mathbf{mm}} = \tilde{\mathbf{S}}_{\mathbf{k}\alpha} \mathbf{C}_{\alpha\alpha} \tilde{\mathbf{S}}_{\mathbf{k}\alpha}^{\mathbf{T}} + \tilde{\mathbf{C}}_{\mathbf{mm}} \tag{6.8.25}$$

$$\Delta\tilde{\boldsymbol{\alpha}} = \boldsymbol{\alpha}' - \boldsymbol{\alpha} = -\left[\mathbf{C}_{\alpha\alpha} \tilde{\mathbf{S}}_{\mathbf{k}\alpha}^{\mathbf{T}} \tilde{\mathbf{C}}_{\mathbf{dd}}^{-\mathbf{1}}\right] \tilde{\mathbf{d}} \tag{6.8.26}$$

$$\Delta\tilde{\mathbf{m}} = \mathbf{m}' - \mathbf{m} = \left[\tilde{\mathbf{C}}_{\mathbf{mm}} \tilde{\mathbf{C}}_{\mathbf{dd}}^{-1}\right] \tilde{\mathbf{d}} \tag{6.8.27}$$

$$\Delta\tilde{\mathbf{k}} = \mathbf{k}' - \mathbf{k} = (\mathbf{m}' - \mathbf{m}) - \tilde{\mathbf{d}} = \mathbf{S}_{\mathbf{k}\alpha} (\boldsymbol{\alpha}' - \boldsymbol{\alpha}) \tag{6.8.28}$$

Relative covariances for the posterior values of the nuclear data and measured responses are given as

$$\mathbf{C}_{\alpha'\alpha'} = \mathbf{C}_{\alpha\alpha} - \left[\mathbf{C}_{\alpha\alpha}\tilde{\mathbf{S}}^{\mathbf{T}}_{\mathbf{k}\alpha}\right]\tilde{\mathbf{C}}^{-1}_{\mathbf{dd}}\left[\tilde{\mathbf{S}}_{\mathbf{k}\alpha}\mathbf{C}_{\alpha\alpha}\right] \tag{6.8.29}$$

$$\tilde{\mathbf{C}}_{\mathbf{m}'\mathbf{m}'} = \tilde{\mathbf{C}}_{\mathbf{mm}} - \left[\tilde{\mathbf{C}}_{\mathbf{mm}}\tilde{\mathbf{C}}^{-1}_{\mathbf{dd}}\tilde{\mathbf{C}}_{\mathbf{mm}}\right] \tag{6.8.30}$$

If all the responses on the TSURFER input are absolute formatted, the adjusted data and response values edited by TSURFER are obtained from Eq. (6.8.26)-Eq. (6.8.28), while the square roots of diagonal elements in Eq. (6.8.29)-Eq. (6.8.30) correspond to the edited values for adjusted uncertainties in the nuclear data and in the experiment responses, respectively.

The adjustment formulas again must be modified slightly given a set of mixed relative/absolute-formatted responses. In the following expressions, mixed response covariance and response sensitivity data are denoted by a caret (see Appendix A.), and $\hat{\mathbf{F}}_{\mathbf{m/k}}$ is an I × I diagonal matrix containing m/k factors for relative-formatted responses or a value of one for absolute-formatted responses:

$$\hat{\mathbf{d}} = \begin{cases} \frac{\mathbf{k}(\alpha)_i - \mathbf{m}_i}{\mathbf{k}(\alpha)_i} & relative \\ \mathbf{k}(\alpha)_i - \mathbf{m}_i & absolute \end{cases} \tag{6.8.31}$$

$$\Delta\hat{\mathbf{m}}_{\mathbf{i}} = \begin{cases} \frac{\mathbf{m}'_{\mathrm{i}} - \mathbf{m}_{\mathrm{i}}}{\mathbf{m}_{\mathrm{i}}} & relative \\ \mathbf{m}'_{\mathrm{i}} - \mathbf{m}_{\mathrm{i}} & absolute \end{cases} \tag{6.8.32}$$

$$\Delta\hat{\mathbf{k}}_{\mathrm{i}} = \begin{cases} \frac{\mathbf{k}'(\alpha')_i - \mathbf{k}(\alpha)_i}{\mathbf{k}(\alpha)_i} & relative \\ \mathbf{k}(\alpha)_i - \mathbf{k}(\alpha)_i & absolute \end{cases} \tag{6.8.33}$$

$$\hat{\mathbf{C}}^{-1}_{\mathbf{dd}} = \hat{\mathbf{C}}_{\mathbf{kk}} + \hat{\mathbf{F}}_{\mathbf{m/k}}\hat{\mathbf{C}}_{\mathbf{mm}}\hat{\mathbf{F}}_{\mathbf{m/k}} = \hat{\mathbf{S}}_{\mathbf{k}\alpha}\mathbf{C}_{\mathbf{dd}}\hat{\mathbf{S}}^{\mathbf{T}}_{\mathbf{k}\alpha} + \hat{\mathbf{F}}_{\mathbf{m/k}}\hat{\mathbf{C}}_{\mathbf{mm}}\hat{\mathbf{F}}_{\mathbf{m/k}} \tag{6.8.34}$$

$$\Delta\hat{\alpha} = -\left[\mathbf{C}_{\mathbf{dd}}\hat{\mathbf{S}}^{\mathbf{T}}_{\mathbf{k}\alpha}\hat{\mathbf{C}}^{-1}_{\mathbf{dd}}\right]\hat{\mathbf{d}} \tag{6.8.35}$$

$$\Delta\hat{\mathbf{m}} = \left[\hat{\mathbf{C}}_{\mathbf{mm}}\hat{\mathbf{F}}_{\mathbf{m/k}}\hat{\mathbf{C}}^{-1}_{\mathbf{dd}}\right]\hat{\mathbf{d}} \tag{6.8.36}$$

$$\Delta\mathbf{k} = \hat{\mathbf{S}}_{\mathbf{k}\alpha}\Delta\hat{\alpha} \tag{6.8.37}$$

Covariances for the posterior values of the nuclear data and measured responses are given as

$$\mathbf{C}_{\alpha'\alpha'} = \mathbf{C}_{\alpha\alpha} - \left[\mathbf{C}_{\alpha\alpha}\hat{\mathbf{S}}^{\mathbf{T}}_{\mathbf{k}\alpha}\right]\hat{\mathbf{C}}^{-1}_{\mathbf{dd}}\left[\hat{\mathbf{S}}_{\mathbf{k}\alpha}\mathbf{C}_{\alpha\alpha}\right] \tag{6.8.38}$$

$$\hat{\mathbf{C}}_{\mathbf{m}'\mathbf{m}'} = \hat{\mathbf{C}}_{\mathbf{mm}} - \left[\hat{\mathbf{C}}_{\mathbf{mm}}\hat{\mathbf{F}}_{\mathbf{m/k}}\hat{\mathbf{C}}^{-1}_{\mathbf{dd}}\hat{\mathbf{F}}_{\mathbf{m/k}}\hat{\mathbf{C}}_{\mathbf{mm}}\right] \tag{6.8.39}$$

If responses on the TSURFER input are both relative formatted and absolute formatted, the adjusted data and response values edited by TSURFER are obtained from Eqs. Eq. (6.8.35)-Eq. (6.8.37), while the square roots of diagonal elements in Eqs. Eq. (6.8.38)-Eq. (6.8.39) correspond to the edited values for adjusted uncertainties in the nuclear data and in the experiment responses, respectively.

Consistency relations and chi-square filtering

The variations for Δm and $\Delta\alpha$ defined by Eq. Eq. (6.8.18) and Eq. Eq. (6.8.19) are those that give the smallest value of the quadratic form χ^2. This minimum χ^2 value is found by substituting these equations into Eq. Eq. (6.8.15):

$$\chi^2_{\min} = \mathbf{d}^{\mathbf{T}}\mathbf{C}^{-1}_{\mathbf{dd}}\mathbf{d} = \mathbf{d}^{\mathbf{T}}\left[\mathbf{C}_{\mathbf{kk}} + \mathbf{F}_{\mathbf{m/k}}\mathbf{C}_{\mathbf{mm}}\mathbf{F}_{\mathbf{m/k}}\right]^{-1}\mathbf{d} \tag{6.8.40}$$

It is interesting to observe that Eq. Eq. (6.8.40) does not depend upon adjustments in nuclear data or integral experiments and physically expresses a measure of the initial discrepancies ($\mathbf{d}$) in all responses, compared to their combined calculation and experiment uncertainties ($\mathbf{C_{kk}} + \mathbf{F_{m/k}C_{mm}F_{m/k}}$). In fact, the parameter is identical to the chi-square statistic discussed in Sect. 6.8.3.2 that denotes consistency between the calculations and measurements. Equation Eq. (6.8.40) can be viewed as an inherent limit on the consistency of the GLLS adjustment procedure. If the initial calculated and measured responses are not consistent with their stated uncertainties, then adjustments in nuclear data and experiment values obtained by TSURFER cannot be consistent either.

TSURFER provides an option for "chi-square filtering" to ensure that a given set of benchmark experiments is consistent; that is, that the input responses have an acceptable $\chi^2_{\min}$ defined by Eq. (6.8.40). The code progressively removes individual experiments until the calculated $\chi^2_{\min}$ is less than the input target value "*target_chi"*. Each iteration removes one experiment estimated to have the greatest impact on chi-square per degree of freedom. The method used to assess individual contributions to $\chi^2_{\min}$ is specified by input parameter "*chi_sq_filter"*, which refers to the different approaches described below.

Independent Chi-Square Option (*chi_sq_filter=independent*).

The consistency of the i-th measured and calculated response values, disregarding any other integral response, is equal to the discrepancy in the measured and calculated value squared divided by the variance of the discrepancy of the i-th response:

$$\chi^2_{\mathbf{ind,i}} = \frac{(\mathbf{k_i} - \mathbf{m_i})^2}{\sigma^2_{\mathbf{ki}} + \sigma^2_{\mathbf{mi}}} \tag{6.8.41}$$

Equation Eq. (6.8.41) is strictly valid only when no correlations exist, but it may be a useful approximation to estimate the experiment having the greatest impact on chi-square per degree of freedom. Hence, this expression is called the "*independent chi-square*" approximation in TSURFER. This approximation executes fast since no matrix inversions are required.

Diagonal Chi-Square Option (*chi_sq_filter=diagonal*)

The "*diagonal chi-square*" approach uses diagonal values of the original inverse $\mathbf{C_{dd}}$ matrix to estimate the experiment having the greatest impact on chi-square per degree of freedom:

$$\chi^2_{\mathbf{dia,i}} \equiv (\mathbf{k_i} - \mathbf{m_i})^2 \, \mathbf{C^{-1}_{dd}}(\mathbf{i},\mathbf{i}) \tag{6.8.42}$$

In this method the correlations in all responses are taken into account to some extent. The original $\mathbf{C^{-1}_{dd}}$. is used in each iteration; therefore, the diagonal chi-square method requires only a single matrix inversion.

Iterative-Diagonal Chi-Square Option (*chi_sq_filter=iter_diag*).

This approach is identical to the diagonal chi-square method, except that an updated value of is computed each iteration to re-evaluate the total chi-square from Eq. Eq. (6.8.40). Thus one matrix inversion is performed per iteration.

Delta Chi-Square Option (*chi_sq_filter=delta_chi*).

The most rigorous method to determine the impact of an individual response on the overall consistency is called the "*delta chi-square*" method in TSURFER. This method [TSURFER-YWMW80] calculates the change in chi-square whenever a particular response is omitted for the analysis; that is, omitting the i*th* response results in

$$\Delta\chi^2_{\mathrm{i}} = \left[\mathbf{d^T C^{-1}_{dd} d}\right] - \left[\mathbf{d}^{\mathrm{T}}_{\neq \mathrm{i}} \left(\mathbf{C}^{\neq \mathrm{i}}_{\mathbf{dd}}\right)^{-1} \mathbf{d}_{\neq \mathrm{i}}\right] \tag{6.8.43}$$

where $\mathbf{d}_{\neq i}$ and $\mathbf{C}_{\mathbf{dd}}^{\neq \mathrm{i}}$ are, respectively, the discrepancy vector and discrepancy covariance with response i omitted. While Eq. Eq. (6.8.43) is the most rigorous method, it also requires the most computation effort. A matrix inversion must be performed for every omitted response, in each iteration. For SCALE 6.3, the runtime of this method was reduced by using the matrix inversion lemma to compute the inverse of each submatrix, according to the method of [TSURFER-JRCMPR16].

It has been observed that independent chi-square and diagonal chi-square options execute fast but often eliminate more experiments than necessary to obtain the target chi-square value. The diagonal chi-square option is somewhat faster than the iterative-diagonal chi-square option but also sometimes omits more than the minimum number of experiments. The delta chi-square option is currently default in TSURFER.

Expressions for computational bias

The computational "bias" is defined here as the observed difference between a calculated and measured response. In conventional validation studies the expected bias in an application response (for which there is no measurement, by definition) often is estimated as the sample mean of the biases for a set of benchmark experiments and the uncertainty in the application bias is estimated by the sample standard deviation of the experimental biases.

The GLLS technique provides another method to compute the bias of an application response. The application response bias β_a is defined as the expected deviation of the original calculated response $\mathrm{k_a}$ from the best estimate of the measured response, which is unknown but has some probability distribution. Note that if the application response actually *did* have a prior measured value $\mathrm{m_a}$, then the best estimate for the experiment value would be the final adjusted value $\mathrm{m_a}$' obtained from the GLLS procedure. For this reason the notation $\mathrm{m_a}$' is used here to represent the (unknown) best estimate for the application's projected measured response, so that

$$\beta_a = E\left[k_a - m'_a\right] \tag{6.8.44}$$

where E is the expectation operator. The application's projected experiment value can be expressed as $m'_a = k_a\left(\alpha'\right) - \delta m_a$, where δm_a represents the difference between the best computed response obtained with the adjusted data α' and the expected value of the experimental measurement. Therefore Eq. Eq. (6.8.44) can be expressed

$$\beta_a = E\left[k_a - k_a\left(\alpha'\right) + \delta m_a\right] = k_a - k_a\left(\alpha'\right) + E\left[\delta m_a\right] \tag{6.8.45}$$

Recall that all *experiment* responses are sure to have $\delta m_i = 0$, because the GLLS procedure forces k'=m' within the approximation of first-order theory. However, $\delta m_a\,(= k'_a - m'_a)$ for the application is not guaranteed to be zero, since there is no known measured value. Nevertheless the application response calculated using the best cross sections α' should approach the desired (unknown) experiment value if a "sufficient" number of experiments similar to the application of interest are considered [TSURFER-BHCP99] so that under these conditions $E\left[\delta m_a\right] \rightarrow 0$ for the application as well. More details concerning the suitable degree of similarity and the sufficient number of experiments necessary for convergence of the GLLS methodology are discussed in other publications [TSURFER-BHCP99, TSURFER-BHP99, TSURFER-MWB81, TSURFER-PC88]. TSURFER also provides an automated procedure to examine the convergence of the bias, which is described in Sect. 6.8.4.4.

Assuming an adequate benchmark data base such that $E\left[\delta m_a\right] \rightarrow 0$, Eq. Eq. (6.8.45) simplifies to

$$\beta_a = k_a - k'_a\left(\alpha'\right) \sim -\left(k_a\right)\mathbf{S}_a^{\mathbf{T}}\Delta\boldsymbol{\alpha} \tag{6.8.46}$$

or, stated in absolute terms,

$$\beta_a \sim -\mathbf{S_a^T}\Delta\boldsymbol{\alpha}. \tag{6.8.47}$$

In the above equations $\mathbf{S_a}$ is the column vector of relative sensitivities for the application response. A negative bias indicates that the original computed value was too low; therefore, the adjusted application result will be higher than the original ($k'_a > k_a$). Similarly, a positive bias means that the original response was calculated too high, and therefore $k'_a < k_a$.

6.8.4.3 Expressions for response similarity parameters

TSURFER estimates the similarity in a pair of responses using one of three internally computed similarity coefficients-respectively designated as E, G, and C-specified by the input parameter *"sim_type"*. These are essentially equivalent to the corresponding similarity coefficients described in [TSURFER-BRH+04], although there are slight differences in the definitions of E and G. Similarity coefficients are defined so that a value of zero indicates no similarity between the systems, and unity is maximum similarity. It is also theoretically possible, but unusual, to have a negative similarity in the range [-1,0], indicating systems that are "anti-correlated" in some sense—in which case they are treated as completely dissimilar. Input parameter *"sim_min"* specifies the minimum similarity coefficient (compared to a specified reference application) of systems to be included in the GLLS procedure. TSURFER also optionally edits the I by I similarity matrix whose elements are the similarity coefficients for every response-pair combination (including both experimental and application responses).

The three types of similarity coefficients used in TSURFER are described below. In these expressions, $\mathbf{S_i}$ is defined as the sensitivity *vector* (not matrix) for a particular response "i" which may be an experiment or application. The magnitude of the sensitivity vector corresponds to the L2 norm: $|\mathbf{S_i}| = \sqrt{\mathbf{S_i^T S_i}}$.

The* E *similarity parameter (*sim_type=E*)

The E-similarity coefficient relating two responses i and j is defined analogously to the cosine of the angle between two direction vectors:

$$\mathrm{E_{i,j}} \equiv \frac{\mathbf{S_i^T S_j}}{|\mathbf{S_i}|\,|\mathbf{S_j}|} \tag{6.8.48}$$

A value of E=1.0 corresponds to the case when **Si** and **Sj** are "parallel", such as would occur when the two sensitivity vectors are proportional. A value of E = 0.0 corresponds to the case when **Si** and **Sj** are "perpendicular", such as occurs when the two sensitivity vectors have no common components (i.e., for every non-zero component in one, the corresponding component is zero in the other). Thus, E indicates the "relative direction" of the two sensitivity vectors in an N-dimensional vector space, with the assumption that the larger the parallel component, the greater the similarity. In theory, E could also take on the negative values in the interval [-1,0] if the two responses are anti-parallel. In addition, the E coefficient is the same for absolute-formatted sensitivities (i.e., $\tilde{\mathbf{S}}_\mathbf{i}$ and $\tilde{\mathbf{S}}_j$) or mixed relative-absolute sensitivities (e.g., $\tilde{\mathbf{S}}_\mathbf{i}$ and $\mathbf{S}_j$).

The G similarity parameter (sim_type=G)

The G-similarity coefficient for responses i and j is defined as,

$$G_{ij} \equiv 1 - \frac{\left|\mathbf{S_i} - \mathbf{S_j}\right|^2}{|\mathbf{S_i}|^2 + \left|\mathbf{S_j}\right|^2} = \frac{\mathbf{S_i^T S_j}}{|\mathbf{S}|^2}, \tag{6.8.49}$$

where $\overline{|\mathbf{S}|^2} \equiv \frac{|\mathbf{S_i}|^2 + |\mathbf{S_j}|^2}{2}$.

As seen in the last term, the G parameter is similar to the E parameter, except the denominators are different. The effect of the different normalization is that G will be unity only if S_i and S_j are *identical*, while E indicates maximum similarity if they are *proportional*. It is important to note that the expression for the G parameter in the TSUNAMI-IP manual is different from Eq. Eq. (6.8.49). In both the TSURFER and TSUNAMI-IP formulations of G, the calculated parameter depends on the sensitivity format. It is recommended that Eq. Eq. (6.8.49) be used with relative-formatted sensitivities to calculate G for k_{eff} responses.

The* C *similarity parameter (*sim_type=C*)

The C similarity coefficient represents the correlation in two calculated responses due to the shared uncertainty from common nuclear data. While E and G similarity coefficients only depend on the sensitivity vectors of two responses, the C parameter also involves cross-section covariance data. The C-similarity parameter for responses i and j is the value of the correlation coefficient (ρ_{ij}) in position (i, j) of the R_{kk} correlation matrix; thus,

$$C_{ij} \equiv \rho_{ij} = R_{kk}(i, j) = \frac{\mathbf{S_i^T} C_{\alpha\alpha} \mathbf{S_j}}{\boldsymbol{\sigma_i \sigma_j}}. \tag{6.8.50}$$

The C coefficient has the usual interpretation of a correlation coefficient: 1.0 implies that the two responses are completely correlated by their nuclear data; 0.0 means no correlation; and -1.0 means full anti-correlation. The C coefficient is the same for absolute-formatted sensitivities (i.e., $\tilde{\mathbf{S}}_\mathbf{i}$ and $\tilde{\mathbf{S}}_j$) or mixed relative-absolute sensitivities (e.g., $\tilde{\mathbf{S}}_\mathbf{i}$ and $\mathbf{S}_j$).

6.8.4.4 Convergence of reference application response

It is sometimes useful to consider how the GLLS procedure "converges" the estimated bias in an application response, as the number and similarity of integral experiment responses included in the analysis is increased [TSURFER-Wil86]. In TSURFER, the bias-convergence can be edited for any one of the application responses, called the "reference application," which is defined by the value of "*ref_app*" in the TSURFER input. Inserting the keyword *"calc_cumul_effect"* on the TSURFER input activates the option to edit the cumulative impact of increasing the number of benchmark experiments in the GLLS calculation. In this case, the range of similarity coefficients [0.0→1.0] is subdivided in bins of constant width set by the TSURFER input parameter *"bin_width,"* and the experiment responses are sorted into the bins according their similarity to the reference application response. Any experiments with negative similarity coefficients are included in the first bin. Each bin of benchmark experiments is successively added to the GLLS calculation, going from low to high response similarity, until the whole suite of benchmarks is included. Ideally, the calculated reference application bias (β_a) should converge and stabilize at some value as the number and similarity of the experiment responses increases. Under these conditions the value $E[\delta m_a]$ in Eq. Eq. (6.8.45) is approximately zero.

6.8.5 TSURFER INPUT DESCRIPTION

The user input for TSURFER is described in this section. The input consists of an optional title on a single line followed by one required and four optional blocks of data which are identified in Table 6.8.1 and individually described in subsequent subsections. These data blocks must begin with **READ KEYNAME**and end with **END KEYNAME**, where **KEYNAME** is the name of an individual data block. The *PARAMETER* data block, if requested, should be entered first after the optional title. The *HTML, COVARIANCE,* and *RESPONSE* data blocks may follow in any order. If the *CORR* data block is necessary to specify experiment correlations, it should be the last block of data on the input. All keyword inputs are internally translated to lowercase with the exception of sensitivity data filenames and their file paths.

Table 6.8.1: Table keynames and descriptions for TSURFER input data blocks.

Keyname	**Description**	**Required/Optional**
PARAMETER	Parameters that specify the covariance data file, chi-square filtering options, similarity filtering options, output edit options, and approximate cross-section covariance data options can be entered in this section.	Optional
RESPONSE *EXPERIMENTS*[a] *APPLICATIONS*[a]	File paths to sensitivity data files representing experiments or applications are input in this section. Measured response values and measured response uncertainties are also input in this section.	Required
COVARIANCE	User-input standard deviation for nuclide-reaction pairs for which cross-section-covariance data are not available can be entered in this section.	Optional
HTML	Parameters to customize the HTML-formatted output can be entered in this section.	Optional
CORR	Correlations between measured responses and measured response uncertainty components can be entered in this section.	Optional
[a] The TSUNAMI-IP block keynames *EXPERIMENTS* and *APPLICATIONS* are also allowed. By default, sensitivity data files listed in *RESPONSE* or *EXPERIMENTS* data blocks are designated experiment responses and files listed in the *APPLICATIONS* data block are designated application responses. The response designation can be easily changed by the *use* keyword in the response definition record described in Sect. 6.8.5.2.		

6.8.5.1 Parameter block

The *PARAMETER* data block is used to specify various keyword options used to control the execution of the code. These options include the name of the cross-section covariance data file, output edits, default covariance data, and chi-square or similarity filtering options. The parameter block must begin with *READ PARAMETER* and end with *END PARAMETER*. The data input to the parameter data block consist of numerous keywords that are shown, along with their default values and descriptions, in

Table 6.8.2. A keyword that ends with "=" must be followed by numeric data or a character string. Keywords that do not end with "=" are Boolean flags that are used to turn on certain features of the code, such as the computation of certain data or certain output edits. If the keyword is present for a Boolean entry, the value is set to true. Otherwise, the value is set to false. All *PARAMETER* block keywords are optional.

Table 6.8.2: Input data for the Parameter block of TSURFER.

Keyword	**Default value**	**Description**
absolute or *abs*	False	Use absolute sensitivities and uncertainties for all applications and experiments in the analysis unless specifically overridden by experiment or application input.
adj_cov_cut	0.000001	Cutoff value for including an adjusted cross-section-covariance matrix in the post adjustment analysis and data file. If a nuclide-reaction to nuclide-reaction covariance contains no values exceeding adj_cov_cut,the matrix is excluded from further analysis. Note that adj_cov_cut represents a variance, not a standard deviation.
adjcut	0.00001	Cutoff value for the cross-section adjustment edit. If the maximum (absolute value) multigroup cross-section adjustment for a given nuclide-reaction pair is less than *adjcut*, then the nuclide-reaction pair is not included in the cross-section adjustment edit.
bin_widtha	0.01	Size of the similarity bins for the cumulative iteration edits.
cov_fix	False	Replace zero and large (standard deviation >1000%) values on diagonal of cross-section-covariance data with user-input values and default values.
coverx=	56group-cov7.1	Name of cross-section covariance data file to use in analysis. See the COVLIB chapter of SCALE documentation for detailed description of the available covariance library.
calc_cumul_effect[a]	False	Perform cumulative iteration edit.

continues on next page

Table 6.8.2 – continued from previous page

chi_sq_filter [b]	delta_chi	Method used for chi-square filtering. Allowed values are *independent*, *diagonal*, *iter_diag*, and *delta_chi*.
def_min=	0.001	Minimum sensitivity criteria to adjust nuclear data. The minimum sensitivity criteria is only applied to nuclide-reaction pairs with **missing** covariance data and if *use_dcov* or *use_icov* is entered.
large_cov=	10.0	Cutoff fractional standard deviation value for *cov_fix*. Cross-section-covariance data with uncertainties larger than *large_cov* are replaced with user-input or default values. Default =10, which is 1000% uncertainty.
nohtml	False	If *nohtml* is present, HTML-formatted output is not generated.
print=	regular	Level of ouput edits discussed below. Options are *minimum* and *regular*.
print_adjustments	False	Option to print cross-section adjustment edit.
print_adj_corr	False	Option to print the adjusted response correlation matrix.
print_init_corr	False	Option to print the initial response correlation matrix.
print_sim_matrix	False	Option to print the initial response similarity matrix.
ref_app= [a]	First Application on Input	If application systems are included, *ref_app* is the index to the reference application response. Additional output edits are given for the reference application described in Sample Problem Input and Output Description.
relative or *rel*	True	Use relative sensitivities and uncertainties for all applications and experiments in the analysis. This is the default and keyword *relative* is not required.
return_adj_cov	False	Option to create a COVERX-formatted covariance data file of the adjusted cross-section-covariance matrix. If *return_adj_cov* is present, the adjusted covariance data file is returned to the working directory with the file name *job_name.adj.cov* where *job_name* is the name of the input file.

continues on next page

Table 6.8.2 – continued from previous page

return_work_cov	False	If *return_work_cov* is present, the working covariance library is copied to the return directory with the file name *job_name.wrk.cov* where *job_name* is the name of the input file. If *return_work_cov* is not present, the working covariance library remains in the temporary working directory with the file name *job_name.wrk*.
sim_min= [c]	-1	Minimum similarity coefficient of experimental responses with the reference application response to be included in the adjustment.
sim_type= [c]	None	Criteria to calculate initial response similarity matrix. Allowed values are *none, E, C,* and *G.*
target_chi= [b]	1.2	Target chi-square per degree of freedom for consistency acceptance. If *target_chi=0.0*, chi-square filtering is not performed.
udcov= (optional)	0.05	User-defined default value of standard deviation in cross-section data to use for all groups for nuclide-reaction pairs for which cross-section-covariance data are not available on the input covariance library.
udcov_corr= (optional)	1.0	User-defined default correlation value to use for nuclide-reaction pairs for which cross-section-covariance data are not available on the input covariance library.
udcov_corr_type= (optional)	zone	User-defined default correlation in cross-section data to use for nuclide-reaction pairs for which cross-section-covariance data are not available on the input SCALE covariance library. Allowed values are *long*, *zone*, and *short.*
udcov_therm= (optional)	0.0	User-defined default value of standard deviation in cross-section data to use for thermal data for nuclide-reaction pairs for which cross-section-covariance data are not available on the input covariance library.
udcov_inter= (optional)	0.0	User-defined default value of standard deviation in cross-section data to use for intermediate data for nuclide-reaction pairs for which cross-section-covaria nce data are not available on the input covariance library.

continues on next page

Table 6.8.2 – continued from previous page

udcov_fast= (optional)	0.0	User-defined default value of standard deviation in cross-section data to use for fast data for nuclide-reaction pairs for which cross-section-covariance data are not available on the input covariance library.
uncert_long	False	Prints extended table of uncertainty in response due to covariance data.
use_dcov	False	Use default cross-section-covariance data for nuclide-reaction pairs not included on the input covariance data file.
use_diff_groups=	true	Permit sensitivity data files to have different energy group structures. This parameter is now always true and does not need to be set.
usename	False	Use the name of the sensitivity data file as the default response identifier in the TSURFER output.
use_icov	False	Use user-defined cross-section-covariance data input in the *COVARIANCE* input data block in place of the default values for user-input nuclide-reaction pairs that are not on the input covariance data file.
[a] See Sect. 6.8.4.4 for additional information on bias convergence analysis. [b] See Sect. 6.8.4.2.1 for additional inforomation on chi-square filtering methods. [c] See Sect. 6.8.4.3 for additional information on similarity coefficients.		

The PARAMETER block keyword *print* controls the general level of the TSURFER output print. The minimum print level *"print=minimum"* summarizes the input values for experimental responses and uncertainties, edits chi-square values, and prints GLLS results for the application responses. The regular print option *"print=regular"* additionally shows GLLS results computed for all adjusted experimental responses.

The *PARAMETER* block keywords-*use_dcov, udcov, udcov_therm, udcov_inter, udcov_fast, udcov_corr,* and *udcov_corr_type*-are used to specify the default covariance data for nuclide-reaction pairs that do not have covariance data available on the SCALE covariance data file. The Boolean flag keyword *use_dcov* activates the use of default covariance data for nuclide-reaction pairs with missing covariance data. The *udcov* keyword specifies a default relative standard deviation for all energy groups. The keywords *udcov_therm, udcov_inter,* and *udcov_fast* can be used to specify the default relative standard deviation for the thermal energy groups, intermediate energy groups, and fast energy groups, respectively. If either *udcov_therm, udcov_inter,* or *udcov_fast* are omitted from the input, the default uncertainty applied for their respective energy groups is the *udcov* value. The keyword *udcov_corr* specifies the correlation coefficient for the default covariance data, and *udcov_corr_type* specifies the correlation type. The correlation type options are (a) *long* - apply correlation coefficient in all energy groups, (b) *short* - apply correlation coefficient in adjacent groups, and (c) *zone* - apply correlation within fast, intermediate, and thermal groups, but no correlation is applied between different group ranges.

For additional user control over the approximate cross-section covariance data, the *COVARIANCE* data block can be used to input uncertainty values for particular nuclide-reaction pairs. To utilize the covariance data generated by user-input in the *COVARIANCE* data block, the keyword *use_icov* must be entered in the *PARAMETER* data block. Approximate covariance data specified in the *COVARIANCE* data block are

referred to as user-input data. The input for the COVARIANCE data block is described in more detail in Sect. 6.8.5.3.

When *use_dcov* and/or *use_icov* and *cov_fix* are specified in the *PARAMETER* data block, and a reaction has zero or large (standard deviation > 1000%) values on the diagonal of the covariance matrix, these values are replaced with the square of the user-input or default standard deviation value, and the corresponding off-diagonal terms are substituted according to the user-input or default correlation values. Warning messages are printed to identify which values were replaced and which standard deviation value was used in the replacement. The maximum relative standard deviation in which to apply the covariance correction can be specified by the user with the *large_cov* keyword.

The *def_min* keyword is used to determine if the default or user-input covariance data is applied for nuclide-reaction pairs with missing covariance data. For each nuclide-reaction pair with missing covariance data, TSURFER calculates the maximum, absolute-value, groupwise response sensitivity over all active (i.e., experiment) and passive (i.e., application) responses on the input. If the maximum sensitivity value for a given nuclide-reaction pair is greater than *def_min*, the default or user-input covariance data is applied and the cross-section data for the nuclide-reaction pair is adjusted in the analysis. If the *relative* keyword is entered in the *PARAMETER* data block, the value of *def_min* is interpreted as a relative-formatted sensitivity. Likewise, if the *absolute* keyword is entered in the *PARAMETER* data block, the value of *def_min* is interpreted as an absolute-formatted sensitivity. If both *relative* and *absolute* are entered, the last keyword in the *PARAMETER* data block sets the format for both *def_min* and the response sensitivity data files. If both *relative* and *absolute* are omitted, the default format for *def_min* is relative. The minimum sensitivity criterion is slightly different if both relative-formatted k_{eff} responses and absolute-formatted eigenvalue-difference (reactivity) responses are included in the analysis. In this case, the minimum sensitivity criteria can be entered for each response in the *RESPONSE* block described in the next section.

During the GLLS analysis, TSURFER computes a new covariance data file that contains cross-section-covariance data only for the nuclide-reaction pairs that are listed in the response sensitivity data files. The new covariance data file, referred to as the working covariance data file, is written in COVERX format like the input SCALE covariance data file. The working covariance data file contains any default or user-input cross-section-covariance data for nuclide-reaction pairs that were not in the input SCALE covariance data file as well as any corrected cross-section-covariance data if the *cov_fix* keyword is entered on the input. The working covariance data file can be read by the data plotting tool in Fulcrum to visualize the cross section covariance data used in the analysis.

6.8.5.2 RESPONSE block

In the *RESPONSE* data block, sensitivity data files are designated as either application responses or experiment responses. The *RESPONSE* data block is also used to specify experimental response values, experimental response uncertainties, and uncertainties of experimental response components. The TSUNAMI-IP block keynames *EXPERIMENTS* and *APPLICATIONS* are also allowed. Each data block must begin with *READ KEYNAME* and end with *END KEYNAME* where *KEYNAME* can be *APPLICATIONS, EXPERIMENTS,* or *RESPONSE*.

By default, sensitivity data files listed in *RESPONSE* or *EXPERIMENTS* data blocks are designated as experiment responses, while files listed in the *APPLICATIONS* data block are designated as application responses. Multiple *RESPONSE*, *EXPERIMENTS*, and *APPLICATIONS* data blocks are allowed, and they can be entered in any order. However, the order of the sensitivity data files in the TSURFER input is important when defining experiment correlations. Two recommended input methods are (a) define all experiment and application responses in a single *RESPONSE* data block using the *use* keyword and specify the role of each

response in the analysis or (b) define all experiment responses in a single *EXPERIMENTS* block, and define all application responses in a single *APPLICATIONS* block.

Inside each data block, sensitivity data files are listed using response definition records. A response definition record is a single line of input that contains the sensitivity data filename, two required keywords and eight optional keywords shown in parentheses. The sensitivity data filename and keywords can be entered in any order, with the following format:

filename (use=R) (name=N) (type=T) ev=E uv=U (cv=C) (nu=P) (omit) (abs) (rel) (msen=M)

where

filename = sensitivity data filename. The filename can include the file path.

R = adjustment role. Allowed values are *appl*, *expt*, and *omit*. The default value is *expt* in the RESPONSE or EXPERIMENT block and *appl* in the APPLICATION block.

N = 20-character maximum alphanumeric response identifier in TSURFER output.

T = 8-character maximum alphanumeric identifier for the response type (e.g., "keff", "gpt", or "rho"). The response-type identifier is used in various output edits along with the response name identifier.

E = experimental value of the response.

U = uncertainty value of the response.

C = calculated value of the response.

P = number of uncertainty components to characterize the experiment uncertainty for this response.

omit - Optional keyword used to omit the response from the analysis. This can also be done by the *use=omit* keyword specification.

abs - Optional keyword that specifies absolute sensitivities, absolute experiment uncertainties, and absolute components of uncertainty that are used for this response. The keyword *absolute* is also valid.

rel - Optional keyword that specifies relative sensitivities, relative experiment uncertainties, and relative components of uncertainty that are used for this response. The keyword *relative* is also valid.

M = minimum sensitivity criteria for this response. This value will replace the def_min value in the PARAMETER block to determine if nuclide-reaction pairs with missing covariance data are included in the adjustment.

Case-sensitive filenames and file paths are allowed for sensitivity data filename. However, **spaces are not allowed in the filenames or file paths.** The sensitivity data filename is limited to 80 characters, and the total length of the response definition must not exceed 255 characters. The *use* keyword specifies the role of the response in the GLLS analysis. "*use=expt"* designates the corresponding sensitivity data file as an experiment response. Likewise, "*use=appl"* designates the corresponding data file as an application response. In addition, the user can omit the sensitivity data file from the analysis by entering either '*use=omit"* or simply *omit* on the response definition record. If the *use* keyword is not included, the role of the response is determined by the data block name; that is, "*use=appl"* is implied for the *APPLICATIONS* block and "*use=expt"* is implied for the *EXPERIMENTS* and *RESPONSE* blocks.

By default, TSURFER identifies responses in the output according to the title on the sensitivity data files. For files that have the same titles, or have long or non-descriptive titles, the *usename* keyword in the *PARAMETER* data block can be used to identify the response by their sensitivity data filename. Although filenames are unique, they can also be non-descriptive. For this reason, the *name* keyword on the response definition record can be used to create a new identifier for the response in the TSURFER output. Similarly, the *type* keyword can be used to identify the response type in the output. The default response type is *"keff"* for k_{eff} responses and *"rho"* for eigenvalue-difference responses. It may be useful to include a sequence number in the response name, in order to more easily associate the response number to the input response data. For example, the response names for the first three responses entered could be *name=1_GODIVA, name=2_JEZEBEL,* and *name=3_ZPR4*. In the *CORR* data block and in the printed output, responses are identified by their sequence number (i.e., the order read in), so it is convenient to show this number in the response Name, especially when dealing with a large number of responses.

The measured value of the response (*ev*=) and the measured uncertainty *(uv*=) are required for experiment responses. For application or omitted responses, the *ev* and *uv* keywords are permitted but are not required. The calculated response value is read from the sensitivity data file, but can be overridden by the *cv* keyword. The *nu*= keyword defines the number of uncertainty components that characterize the experiment response uncertainty. If the experiment response uncertainty is given in terms of uncertainty components, the *uv*= keyword specification is optional. An uncertainty component definition record follows the response definition record if the *nu*= keyword specification is given. The uncertainty component definition record has the following format:

ucmp1 val1 ucmp2 val2 . . . . . ucmpP valP,

where

uncmp1 = 4-character alphanumeric identifier for the 1st uncertainty component,

val1 = experiment uncertainty for component *uncmp1,*

Uncmp2 = 4-character alphanumeric identifier for the 2nd uncertainty component,

val2 = experiment uncertainty for component *uncmp2,*

uncmpP = 4-character alphanumeric identifier for the Pth uncertainty component, and

valP = experiment uncertainty for component *uncmpP.*

The uncertainty component definition record contains *nu=P* pairs of alphanumeric identifiers and numeric values. The measured uncertainty value is determined by Eq. Eq. (6.8.10).

The keywords *abs* and *rel* are used to determine the format of sensitivity and uncertainty data on the response definition record and the uncertainty component definition record. For a k_{eff} response, the following four input definitions are equivalent:

```
1)    name=exp_001 ev=1.001 uv=0.005000 rel C:\sensitivity\k_critical_a.sdf

2)    name=exp_001 ev=1.001 uv=0.005005 abs C:\sensitivity\k_critical_a.sdf

3)    name=exp_001 ev=1.001 nu=2        rel C:\sensitivity\k_critical_a.sdf
       enri 0.003000 sden 0.004000

4)    name=exp_001 ev=1.001 nu=2        abs C:\sensitivity\k_critical_a.sdf
       enri 0.003003 sden 0.004004
```

In the example above, the measured k_{eff} is 1.001 ± 0.5% or 1.001 ± 0.005005. (Although most critical experiments have measured k_{eff}= 1, this contrived example reveals the difference between the *absolute* format and the *response* format.) The sensitivity data filename is given as *C:sensitivityk_critical_a.sdf*, and the experiment response is referred to as *exp_001* in the TSURFER output. In 1), the *relative* format is used to specify the relative standard deviation of the measured response as 0.005. In 2), the *absolute* format is used to specify the absolute standard deviation of the measured response as 0.005005. Because the TSUNAMI-generated sensitivity data file is in relative format, TSURFER internally renormalizes the relative sensitivities to absolute sensitivities. In 3), the *relative* format is used to specify the relative standard deviation of k_{eff} due to two components (*enri* and *sden*) as 0.003 and 0.004, respectively. Using Eq. Eq. (6.8.10), the relative standard deviation of k_{eff} is computed to be 0.005. In 4), the *absolute* format is used to specify the absolute standard deviation of k_{eff} due to two components (*enri* and *sden*) as 0.003003 and 0.004004, respectively. The absolute standard deviation of k_{eff} is computed to be 0.005005. Like 2), the sensitivity data file is internally converted to contain absolute sensitivities.

For a second example, the following input definitions are equivalent for an eigenvalue-difference, or reactivity, response:

```
1)    C:\sensitivity\reactivity.sdf ev=15.0000  uv=3.0 abs

2)    C:\sensitivity\reactivity.sdf ev=0.00015  uv=0.2 rel
```

In this example, the measured reactivity is 15 pcm (percent-mille) ± 3 pcm or 0.00015 ± 20%. TSAR creates reactivity sensitivity files in either (a) *absolute* format where the calculated reactivity response and sensitivities are in pcm units or (b) *relative* format with relative sensitivities and the calculated reactivity response are not in pcm units. The TSURFER response definition records are designed to be consistent with the TSAR formats. In 1), the *absolute* format is used to specify the absolute standard deviation of the measured response as 3 pcm. In 2), the *relative* format is used to specify the measured reactivity response as 0.00015 with a relative standard deviation of 0.20 (or 20%). Because TSAR-generated reactivity sensitivity data files may be in absolute format or relative format, TSURFER internally renormalizes the reactivity sensitivity data file to the user-requested format. On occasion, it is desired to adjust a set of nuclear data with zero-valued reactivity responses (i.e., *ev=0.0*). For this case, the *absolute* format should be used because the relative standard deviation of the measured response approaches infinity.

If the *abs* or *rel* keywords are not included on the response definition record, the default format is determined by the *abs* or *rel* keywords in the PARAMETER data block. If more than one formatting keyword is entered in either the PARAMETER data block or response definition record, the last keyword entered sets the format. As an example, the following inputs are equivalent:

1)

```
read parameter
  relative
end parameter
read response
  C:\sensitivity\k_critical_a.sdf ev=1.0 uv=0.005 rel
  C:\sensitivity\reactivity.sdf   ev=0.0 uv=3.0   abs
end response
```

2)

```
read parameter
  relative
end parameter
```

(continues on next page)

(continued from previous page)

```
read response
  C:\sensitivity\k_critical_a.sdf ev=1.0 uv=0.005
  C:\sensitivity\reactivity.sdf   ev=0.0 uv=3.0   abs
end response
```

3)

```
read parameter
  absolute
end parameter
read response
  C:\sensitivity\k_critical_a.sdf ev=1.0 uv=0.005 rel
  C:\sensitivity\reactivity.sdf   ev=0.0 uv=3.0
end response
```

4)

```
read response
  C:\sensitivity\k_critical_a.sdf ev=1.0 uv=0.005
  C:\sensitivity\reactivity.sdf   ev=0.0 uv=3.0   abs
end response
```

In this example, two experiment responses are given. The first response is a relative-formatted k_{eff} response. The second response is an absolute-formatted reactivity response. In 1), the format is determined by the formatting keyword on the response definition record. In 2), the relative-format is set as the default format by the *PARAMETER* block and the absolute format for the reactivity response is specified on its response definition record. In 3), the absolute-format is set as the default format by the *PARAMETER* block and the relative format for the k_{eff} is set by its response definition record. Case 4) is the same as case 2) where the default relative format is applied if no *PARAMETER* block is included.

The final optional keyword for the response definition record is *msen=M*. This record sets the minimum sensitivity criteria for nuclide-reaction pairs with missing covariance data. The keyword is useful when dealing with mixed formatted responses. For example, the following input contains three relative formatted k_{eff} responses and two absolute-formatted reactivity responses.

```
1)
read parameter
  use_dcov
  udcov=0.05
  def_min=0.00001
  relative
end parameter
read response
  C:\sensitivity\godiva.sdf       ev=1.0 uv=0.005
  C:\sensitivity\zpr.sdf          ev=1.0 uv=0.005
  C:\sensitivity\jezebel.sdf      ev=1.0 uv=0.005
  C:\sensitivity\void_react_1.sdf ev=0.0 uv=3.0   abs msen=0.1
  C:\sensitivity\void_react_2.sdf ev=0.0 uv=3.0   abs msen=0.1
end response
```

In this example, the *PARAMETER* block keywords initialize all the responses that follow as relative-formatted responses and the minimum sensitivity criteria for applying default covariance data is 0.00001 or 0.001%. This criteria is used for the three k_{eff} responses. In the response definition records for the two reactivity responses, the absolute-format is specified and the minimum sensitivity criteria is 0.1 pcm. Therefore, default covariance data is used for a nuclide-reaction pair with missing cross-section covariance data if at least one k_{eff} sensitivity for one of the three k_{eff} responses is greater than 0.001% or at least one reactivity sensitivity

for one of the two reactivity responses is greater than 0.1 pcm. Similar to the example above, this example has the following equivalent input:

```
2)
read parameter
  use_dcov
  udcov=0.05
  def_min=0.1
  absolute
end parameter
read response
  C:\sensitivity\godiva.sdf       ev=1.0 uv=0.005 rel msen=0.00001
  C:\sensitivity\zpr.sdf          ev=1.0 uv=0.005 rel msen=0.00001
  C:\sensitivity\jezebel.sdf      ev=1.0 uv=0.005 rel msen=0.00001
  C:\sensitivity\void_react_1.sdf ev=0.0 uv=3.0
  C:\sensitivity\void_react_2.sdf ev=0.0 uv=3.0
end response
```

6.8.5.3 COVARIANCE block

The *COVARIANCE* data block allows the user to specify a covariance matrix for specific nuclide-reaction pairs for which covariance data are not present on the input SCALE covariance library or that have zero or large values on the diagonal. The *COVARIANCE* data block must begin with *READ COVARIANCE* and end with *END COVARIANCE*. The available *COVARIANCE* data block keywords and their default values are given in Table 6.8.3.

Table 6.8.3: Input data for the Covariance block of TSURFER.

Input parameter	**Requirement**	**Default value**	**Allowed values**	**Description**
Nuclide	Required	none	Nuclide name or ZA number	Nuclide for which covariance data are to be entered
Reaction	Required	none	Reaction name or ZA number	Reaction for which covariance data are to be entered
all=	Optional	0.0	any number	Fractional standard deviation value to be applied to all groups
fast=	Optional	0.0	any number	Fractional standard deviation value to be applied to fast groups
therm=	Optional	0.0	any number	Fractional standard deviation value to be applied to thermal groups
inter=	Optional	0.0	any number	Fractional standard deviation value to be applied to intermediate groups
corr=	Optional	1.0	any number from -1.0 to 1.0	Correlation between groups

continues on next page

Table 6.8.3 – continued from previous page

Input parameter	Requirement	Default value	Allowed values	Description
corr_type=	Optional	*zone*	*long, short, zone*	Type of correlation applied from group-to-group covariance values *long* - correlation is applied between all groups *short* - correlation is applied only between adjacent groups *zone* - correlation is applied within fast, intermediate, and thermal groups, but no correlation is applied between zones
end	Required			Denotes end of input for current nuclide/reaction (must not start in column 1)

Any MT number or reaction name will be treated as a valid input, but only those present on the response sensitivity data files will produce useful information. The available reaction sensitivity types are shown in table *Reaction Sensitivity Types Computed by SAMS* in the TSUNAMI-IP manual. An energy-covariance matrix is created for the specified nuclide-reaction pair with the square of the entered standard deviation for the diagonal terms for all groups using the *all=* value. Groups in the fast, intermediate, and thermal energies are then set to the square of the standard deviation value entered for *fast=*, *inter=*, and *therm=*, respectively. The off-diagonal terms of the energy matrix are generated according to the input for *corr=*, and *corr_type=*, with default settings of *1.0* and *zone.* Data entered in this block do not override data present on the covariance data file. The SCALE 5.1 input format is supported where data are entered in triplets with the nuclide name or ZA identifier (e.g., u-235 or 92235), then the reaction MT name or number (e.g., 18 or fission), and then a standard deviation value. In this case, the *end* keyword must not be entered. These data are only used if *use_icov* is specified in the *PARAMETER* data block. When both *use_icov* and *cov_fix* are specified in the *PARAMETER* data block, and a reaction has zero or large (standard deviation > 1000%) values on the diagonal of the covariance matrix, these values are replaced with the square of the user input standard deviation value, and the corresponding off-diagonal terms are substituted according to the values of *corr* and *corr_type*.

6.8.5.4 HTML block

The optional *HTML* data block is used to customize HTML-formatted output. The *HTML* data block must begin with *READ HTML* and end with *END HTML.* The data input in the *HTML* data block consist of several keywords that are shown, along with their default values and descriptions, in Table 6.8.4.

A keyword that ends with "=" must be followed by text data. For color entries, any valid html color name can be entered or the hexadecimal representation can be used if preceded by a # sign. For example, to change the background color of the html output to white, *bg_clr=white* and *bg_clr=#ffffff* have the same effect, because *ffffff* is the hexadecimal representation of white. An extensive list of available colors for customized output is shown in HTML colors. Please note that not all features are supported by all browsers.

Table 6.8.4: Input data for HTML block of TSURFER

Keyword	Default value	Description
bg_clr=	*papayawhip*	Background color.
h1_clr=	*maroon*	Color used for major headings.
h2_clr=	*navy*	Color used for sub-headings.
txt_clr=	*black*	Color for plain text.
lnk_clr=	*navy*	Color for hyperlinks.
lnk_dec=	*none*	Decoration for hyperlinks. (none, underline, overline, line-through, blink).
vlnk_clr	*navy*	Color for visited hyperlinks.
ud_clr=	*blue*	Color for values in tables that use default covariance data.
ui_clr=	*red*	Color for values in tables that use user-input covariance data.
udfix_clr=	*royalblue*	Color for values in tables that use default corrected covariance data.
uifix_clr=	*green*	Color for values in tables that use user-input corrected covariance data.

6.8.5.5 CORR block

The CORR block specifies correlation coefficients between different experiment responses. When present, this block must be the last data block in the input file. The correlation block must begin with *READ CORR* and end with *END PARAMETER*. Correlation coefficients for experimental response uncertainties may be entered either as the total correlation coefficient for a pair of responses; or for a particular uncertainty-component shared by two responses. Values for correlation coefficients are input in the form:

corr_typ (i,j)=ρ ..... {repeat for I=1,N } end

where:

`corr_typ` = 4-character alphanumeric identifier for the response uncertainty component previously defined in the READ RESPONSES block (i.e., *ucmp1*, *umcp2*, etc.). `corr_typ` may also equal *totl* indicating that the total correlation is entered. The `corr_typ` identifier may be omitted and the total correlation is assumed.

N = number of responses in TSURFER input.

ρ = the correlation coefficient at the specified position in the correlation matrix for each uncertainty component.

i,j = the row and column index to the correlation matrix for each uncertainty component. Correlation coefficients can be entered the following five ways:

1) Element-by-element - (i,j)= ρ
2) By row - (i,j1:j2)= ρ
3) By column - (i1:j2,j)= ρ
4) By block - (i1:i2,j1:j2)= ρ. This option can be used to set a large block of the correlation matrix to one number. All diagonal elements in the block are reset to 1.0.
5) By shorthand block - (i1:i2)= ρ. This is identical to (i1:i2,i1:i2)= ρ. All diagonal elements in the block are reset to 1.0.

TSURFER initializes each correlation matrix as an N by N identity matrix. Therefore, all uncorrelated elements (i.e., values equal to 0) do not have to be entered. The correlation matrix can be specified using multiple lines of input with each line having a maximum of 255 characters. As each correlation coefficient is

processed, the symmetric element of the correlation matrix is assigned to the same value. Therefore only the upper or lower triangular portion of each correlation matrix must be specified. For example, given the following *RESPONSE* block:

```
read response
  name=1_k  C:\sensitivity\k_critical_a.sdf ev=1.0 uv=0.005
  name=2_k  C:\sensitivity\k_critical_b.sdf ev=1.0 uv=0.005
  name=3_k  C:\sensitivity\k_critical_c.sdf ev=1.0 uv=0.005
  name=4_k  C:\sensitivity\k_critical_d.sdf ev=1.0 uv=0.005
end response
```

then the following forms of the CORR block are equivalent in specifying the 4 x 4 total correlation matrix as:

$$\begin{bmatrix} 1 & .2 & .3 & .2 \\ .2 & 1 & .2 & .2 \\ .3 & .2 & 1 & .1 \\ .2 & .2 & .1 & 1 \end{bmatrix} \tag{6.8.51}$$

1) Specify the upper triangular portion of the matrix element by element:

```
read corr
   totl (1,2)=.2 (1,3)=.3 (1,4)=.2 (2,3)=.2 (2,4)=.2 (3,4)=.1 end
end corr
```

2) Specify the lower triangular portion of the matrix element by element:

```
read corr
   totl (2,1)=.2 (3,1)=.3 (4,1)=.2 (3,2)=.2 (4,2)=.2 (4,3)=.1 end
end corr
```

3) Use the colon character to specify multiple elements at one time in the upper triangular portion of the matrix:

```
read corr
   totl (1,2)=.2 (1,3)=.3 (1,4)=.2 (2,3:4)=.2 (3,4)=.1 end
end corr
```

Values entered for the total correlation matrix will override any component correlations entered. Correlation coefficients may be specified in the *CORR* data block for responses that share one or more of the same uncertainty components. The value of *corr_typ* must correspond to one of the 4-character alphanumeric identifiers given to an uncertainty component. Only those uncertainty components that appear in more than one response description should be entered, since these are the only ones with correlations. An END keyword is required to terminate the data of an individual uncertainty component, and the input is repeated for each type of correlated uncertainty component.

Note: The experiment covariance matrix should be positive definite to ensure a physical result for all possible sensitivities. If the input correlation values do not satisfy this constraint, a warning message is printed. Use of several fully correlated uncertainties can lead to an over-constrained system, which may result in a non-positive-definite covariance matrix. In order to help avoid this problem, correlation values usually should be limited to a maximum of 0.95, suggesting that a small random component is always present.

As an example of correlation matrices for uncertainty components, consider the following input:

```
read response
  name=1_k  C:\sensitivity\k_critical_a.sdf  ev=1.0  nu=2
     enri=0.003 soln=0.004
  name=2_k  C:\sensitivity\k_critical_b.sdf  ev=1.0  nu=2
     enri=0.0005 soln=0.0012
  name=3_k  C:\sensitivity\k_application.sdf use=app
end response
read corr
   enri (1,2)=0.5 end
   soln (1,2)=0.8 end
end corr
```

In this example, two uncertainty components are used, identified as *enri* and *soln*. Using the propagation of error formula Eq. Eq. (6.8.10) , the relative standard deviation of the experiment responses *1_k* and *2_k* are determined as $\sqrt{(0.003^2 + 0.004^2)} = 0.005$ and $\sqrt{(0.0005^2 + 0.0012^2)} = 0.0013$, respectively. The correlation matrix *enri* and *soln* are given as

$$\begin{bmatrix} 1 & .5 \\ .5 & 1 \end{bmatrix} \tag{6.8.52}$$

and

$$\begin{bmatrix} 1 & .8 \\ .8 & 1 \end{bmatrix} \tag{6.8.53}$$

Using Eq. Eq. (6.8.11), the relative covariance between response *1_k* and *2_k* is calculated as: $0.003 * 0.5 * 0.0005$ (*enri* component) + $0.0005 * 0.8^*0.0012$ (*soln* component) = 1.23E-6.

6.8.6 SAMPLE PROBLEM INPUT AND OUTPUT DESCRIPTION

6.8.6.1 Input and text output

An example TSURFER input is given in Example 6.8.1 and the text output is shown in Example 6.8.2-Example 6.8.14. In this sample problem, 40 k_{eff} responses are specified in the *RESPONSE* block, 37 experiments and 3 applications. All calculation options are turned on including the use of default and user-input covariance data, similarity filtering to the reference application, chi-square filtering for consistency, and bias convergence analysis. Each section of the text output is described in order below. Some of the figures of the text output have been truncated from their original length. The examples of output are for illustrative purposes and only demonstrate the format of the TSURFER results.

1. Echo of Input (Example 6.8.2) - The TSURFER input data are printed for the *PARAMETER, HTML*, and *COVARIANCE* data blocks. Both user-specified and default values for the various keywords are edited.
2. Covariance Warnings (Example 6.8.3) - If the PARAMETER block keywords *use_dcov* and/or *use_icov* and/or *cov_fix* are entered, covariance warnings are listed that specify the nuclide-reaction pairs for which approximate covariance data is applied.
3. Listing of Input Responses (Example 6.8.4) - Various information is listed for each response. This includes the response index, name, title, adjustment role (e.g., *expt*), type, calculated response value, measured response value, and similarity coefficient to the reference application. The similarity coefficient column is only edited if a reference application is listed on the input.
4. Experiment Uncertainties (Example 6.8.5) - The experiment standard deviations, as well as any input uncertainty components, are edited for each measured response. When uncertainty components are given, the total standard deviation is computed from Eq. .

5. Chi-square summary (Example 6.8.6) - Different chi-squared values are edited based on the GLLS analysis. This includes the initial value of chi-squared, the target value of chi-squared based on the *target_chi* keyword, and the final value of chi-squared. The independent and diagonal chi-squares are also edited.

6. Correlation Matrices (Example 6.8.7-Example 6.8.9) - Correlation matrices are printed after the chi-squared edit in the following order: (1) response similarity matrix if *print_sim_matrix* is entered in the input, (2) the prior calculated response correlation matrix and prior measured response correlation matrix if *print_init*_corr is entered in the input, and (3) the adjusted response correlation matrix if *print_adj_corr* is entered in the input. The value of the keyword **sim_type** designates the type of similarity coefficient appearing in the similarity matrix. See Sect. 6.6.4.3 for description of the types of similarity coefficients. The response correlation matrices are defined in Appendix A.

7. Cumulative Convergence Edit (Example 6.8.10) - The cumulative convergence edit follows the printout of the requested correlation matrices. TSURFER only performs the cumulative convergence calculation if the keyword *calc_cumul_effect* is entered in the input. Four columns of data are printed that specify the cumulative range number, the maximum similarity coefficient allowed for each adjustment, the number of experiments with similarity coefficients within the specified range, and the computed application bias for each range, shown as o/v(A-C)/C, where A represents the adjusted k_{eff} value and C represents the original calculated k_{eff} value.

8. Summary of Adjustments (Example 6.8.11) - The adjustment summary table is an 11-column table that summarizes the prior and posterior values of each response. The 11 columns include the response adjustment role (i.e., *expt, appl,* or *omit*), the name and type identifiers, the prior and posterior uncertainties of each response, the independent and diagonal chi-squared values, and the change in the response between the prior and posterior values.

9. Summary of Adjusted Responses (Example 6.8.12) - Following the adjustment summary table, the adjusted values of each response are listed in tabular format. The adjusted uncertainty values of the response are also included.

10. Application and Bias Summary (Example 6.8.13) - The application and bias summary table follows the adjusted response table. This edit is only printed if applications are specified on the TSURFER input. For each application, the following values are tabulated: the name and type of the response, the prior and posterior values of the application response, the prior and posterior values of the application uncertainty, and the application bias as determined by Eq. . If the application is a relative-formatted response, the fractional bias is also included in the table. Following this table, a second table is printed that lists the contribution to the reference application bias for each nuclide-reaction pair used in the analysis. The nuclide-reaction pairs are listed in descending order based on the fraction of bias L1-norm, defined as

$$f_x = \frac{\sum_g \left| S_{x,g} \Delta\alpha_{x,g} \right|}{\sum_{x'} \sum_g \left| S_{x',g} \Delta\alpha_{x',g} \right|} \tag{6.8.54}$$

11. Multigroup Cross-Section Adjustment Table (Example 6.8.14) - The Multigroup cross-section adjustment tables are printed if the *print_adjustments* keyword is included in the TSURFER input. For each nuclide-reaction pair, a table is printed that includes the relative adjustment of each multigroup cross-section, and the prior and posterior values of the cross-section uncertainty. If an application is included in the TSURFER input, the bias contribution and fraction of bias L1-norm are also edited. The order of the nuclide-reaction pairs is determined by the fraction of bias L1-norm.

Example 6.8.1: TSURFER sample input.

```
=tsurfer
TSURFER sample problem
read parameter
'
 chi_sq_filter=delta_chi target_chi=3.0
'
 calc_cumul_effect bin_width=0.01
'
 ref=40 sim_type=c sim_min=0.3
'
 use_dcov use_icov cov_fix coverx=44groupcov udcov=0.05 def_min=0.0
'
 print=regular print_adjustments print_adjustments print_adj_corr print_init_corr return_work_cov return_adj_
↪cov
'
uncert_long=false
'
end parameter
read covariance
  u-235 elastic all=0.07 end
  u-238 elastic all=0.06 end
end covariance
read response
nam=1_hst001-1 hst001-001.sdf  ev=1.0 nu=4
   pyra 0.0042 ucna 0.0021  B10a 0.0030  H/Ua 0.0042
nam=2_hst001-1  hst001-02.sdf  ev=1.0 nu=4
   pyra 0.0032 ucna 0.0025  B10a 0.0032  H/Ua 0.0040
nam=3_hst001-1  use=appl hst001-03.sdf
nam=4_hst001-1  hst001-04.sdf  ev=1.0 nu=3
   pyra 0.0039 ucna 0.0015   H/Ua 0.0025
nam=5_hst001-1  hst001-05.sdf  ev=1.0 nu=4
   pyra 0.0040 ucna 0.0021  B10a 0.0025  H/Ua 0.0032
nam=6_hst001-1  hst001-06.sdf  ev=1.0 nu=3
          ucna 0.0021  B10a 0.0030  H/Ua 0.0042
nam=7_hst001-1  hst001-07.sdf  ev=1.0 nu=4
   pyra 0.0022 ucna 0.0022  B10a 0.0033  H/Ua 0.0042
nam=8_hst001-1  hst001-08.sdf  ev=1.0 nu=4
   pyra 0.0022 ucna 0.0025  B10a 0.0036  H/Ua 0.0045
nam=9_hst001-1  hst001-09.sdf  ev=1.0 nu=2
   ucnb 0.0021 H/Ub 0.0042
nam=10_hst001-1  hst001-10.sdf  ev=1.0 nu=2
         ucnb 0.0028    H/Ub 0.0050
nam=11_hst001-1  hst002-01.sdf  ev=1.0 nu=2
         ucnb 0.0031    H/Ub 0.0032
nam=12_hst001-1  hst002-03.sdf  ev=1.0 nu=2
         ucnb 0.0021     H/Ub 0.0042
Figure 6.6.1.  TSURFER sample input.
nam=13_hst001-1  hst002-09.sdf  ev=1.0 nu=2
        ucnb 0.0021  H/Ub 0.0042
nam=14_hst001-1  hst003-03.sdf  ev=1.0 nu=2
        ucnb 0.0011    H/Ub 0.0022
nam=15_hst001-1  hst003-08.sdf  ev=1.0 nu=2
       ucnb 0.0021     H/Ub 0.0042
nam=16_hst001-1  hst003-18.sdf  ev=1.0 nu=2
       ucnb 0.0021     H/Ub 0.0042
nam=17_hst001-1  hst004-003.sdf  ev=1.0 nu=2
       ucnb 0.0021     H/Ub 0.0042
nam=18_hst001-1  hst021-030.sdf  ev=1.0 nu=2
       ucnb 0.0021     H/Ub 0.0042
nam=19_hst001-1  hst025-02.sdf  ev=1.0 nu=2
   ucnb 0.0021  H/Ub 0.0042
nam=20_hst001-1  hst025-04.sdf  ev=1.0 nu=2
   ucnb 0.0021   H/Ub 0.0042
```

(continues on next page)

(continued from previous page)

```
nam=21_hst001-1  hst025-05.sdf  ev=1.0 nu=4
   pyrc 0.0042 ucnc 0.0021  B10c 0.0030  H/Uc 0.0042
nam=22_hst001-1  hst027-01.sdf  ev=1.0 nu=4
   pyrc 0.0045 ucnc 0.0025  B10c 0.0033  H/Uc 0.0040
nam=23_hst001-1  hst29i-01.sdf  ev=1.0 nu=4
   pyrc 0.0039 ucnc 0.0021  B10c 0.0030  H/Uc 0.0045
nam=24_hst001-1  hst29i-02.sdf  ev=1.0 nu=4
   pyrc 0.0042 ucnc 0.0030  B10c 0.0020  H/Uc 0.0022
nam=25_hst001-1  hst29i-03.sdf  ev=1.0 nu=4
   pyrc 0.0042 ucnc 0.0021  B10c 0.0030  H/Uc 0.0042
nam=26_hst001-1  hst29i-04.sdf  ev=1.0 nu=4
   pyrc 0.0022 ucnc 0.0031  B10c 0.0040  H/Uc 0.0036
nam=27_hst001-1  hst29i-05.sdf  ev=1.0 nu=4
   pyrc 0.0042 ucnc 0.0021  B10c 0.0030  H/Uc 0.0042
nam=28_hst001-1  hst29i-06.sdf  ev=1.0 nu=4
   pyrc 0.0032 ucnc 0.0021  B10c 0.0045  H/Uc 0.0032
nam=29_hst001-1  hst29i-07.sdf  ev=1.0 nu=4
   pyrc 0.0042 ucnc 0.0025  B10c 0.0030  H/Uc 0.0036
nam=30_hst001-1  hst30i-01.sdf  ev=1.0 nu=3
    ucnc 0.0021  B10c 0.0030  H/Uc 0.0042
nam=31_hst001-1  hst30i-02.sdf  ev=1.0 nu=3
    ucnc 0.0031  B10c 0.0030  H/Uc 0.0042
nam=32_hst001-1  use=appl hst30i-03.sdf ev=1.0 uv=0.003
nam=33_hst001-1  hst30i-04.sdf  ev=1.0 nu=3
   ucnc 0.0031  B10c 0.0040  H/Uc 0.0038
nam=34_hst001-1  hst30i-05.sdf  ev=1.0 nu=3
    ucnc 0.0021  B10c 0.0030  H/Uc 0.0042
nam=35_hst001-1  hst30i-06.sdf  ev=1.0 nu=3
    ucnc 0.0025  B10c 0.0040  H/Uc 0.00436
Figure 6.6.1.  TSURFER sample input (continued).
nam=36_hst001-1  hst30i-07.sdf  ev=1.0 nu=3
   ucnc 0.0031  B10c 0.0040  H/Uc 0.0038
nam=37_hst001-1  hst31i-01.sdf  ev=1.0 nu=3
    ucnc 0.0021  B10c 0.0030  H/Uc 0.0042
nam=38_hst001-1  hst31i-02.sdf  ev=1.0 nu=3
   ucnc 0.0041  B10c 0.0020  H/Uc 0.0032
nam=39_hst001-1  hst31i-03.sdf  ev=1.0 nu=3
    ucnc 0.0021  B10c 0.0030  H/Uc 0.0042
nam=40_hst001-1 use=appl hst31i-04.sdf
end response
read corr
pyra (1,2)=0.2 (1,4)=0.6 (1,5)=0.5 (1,7)=0.2 (1,8)=0.6
     (2,4)=0.6 (2,5)=0.5 (2,7)=0.2 (2,8)=0.6 (4,5)=0.6
     (4,7)=0.5 (4,8)=0.5 (5,7)=0.2 (5,8)=0.6 (7,8)=0.6  end
ucna (1,2)=0.2 (1,4)=0.2 (1,5)=0.2 (1,6)=0.2 (1,7)=0.2
     (1,8)=0.2 (2,4)=0.2 (2,5)=0.2 (2,6)=0.2 (2,7)=0.2
     (2,8)=0.2 (4,5)=0.6 (4,6)=0.6 (4,7)=0.5 (4,8)=0.5
     (5,6)=0.8 (5,7)=0.2 (5,8)=0.6 (6,7)=0.8 (6,8)=0.6
     (7,8)=0.6                                      end
H/Ua (1,2)=0.2 (1,4)=0.2 (1,5)=0.2 (1,6)=0.2 (1,7)=0.2
     (1,8)=0.2 (2,4)=0.2 (2,5)=0.2 (2,6)=0.2 (2,7)=0.2
     (2,8)=0.2 (4,5)=0.6 (4,6)=0.6 (4,7)=0.5 (4,8)=0.5
     (5,6)=0.8 (5,7)=0.2 (5,8)=0.6 (6,7)=0.8 (6,8)=0.6
     (7,8)=0.6                                     end
ucnb (9,10)=0.2 (9,11)=0.2 (9,12)=0.2 (9,13)=0.2 (9,14)=0.2
     (9,15)=0.2 (9,16)=0.2 (9,17)=0.2 (9,18)=0.2 (9,19)=0.2
     (9,20)=0.2 (10,11)=0.2 (10,12)=0.2 (10,13)=0.2
     (10,14)=0.2 (10,15)=0.2 (10,16)=0.2 (10,17)=0.2
     (10,18)=0.2 (10,19)=0.2 (10,20)=0.2 (11,12)=0.2
     (11,13)=0.2 (11,14)=0.2 (11,15)=0.2 (11,16)=0.2
     (11,17)=0.2 (11,18)=0.2 (11,19)=0.2 (11,20)=0.2
     (12,13)=0.2 (12,14)=0.2 (12,15)=0.2 (12,16)=0.2
     (12,17)=0.2 (12,18)=0.2 (12,19)=0.2 (12,20)=0.2
     (13,14)=0.2 (13,15)=0.2 (13,16)=0.2 (13,17)=0.2
     (13,18)=0.2 (13,19)=0.2 (13,20)=0.2 (14,15)=0.2
```

(continues on next page)

(continued from previous page)

```
     (14,16)=0.2 (14,17)=0.2 (14,18)=0.2 (14,19)=0.2
     (14,20)=0.2 (15,16)=0.2 (15,17)=0.2 (15,18)=0.2
     (15,19)=0.2 (15,20)=0.2 (16,17)=0.2 (16,18)=0.2
     (16,19)=0.2 (16,20)=0.2 (17,18)=0.2 (17,19)=0.2
     (17,20)=0.2 (18,19)=0.2  (19,20)=0.2           end
H/Ub (9,10)=0.2  (9,11)=0.2 (9,12)=0.2 (9,13)=0.2 (9,14)=0.2
     (9,15)=0.2  (9,16)=0.2 (9,17)=0.2 (9,18)=0.2 (9,19)=0.2
     (9,20)=0.2  (10,11)=0.2 (10,12)=0.2 (10,13)=0.2
     (10,14)=0.2 (10,15)=0.2 (10,16)=0.2 (10,17)=0.2
     (10,18)=0.2 (10,19)=0.2 (10,20)=0.2 (11,12)=0.2
     (11,13)=0.2 (11,14)=0.2 (11,15)=0.2 (11,16)=0.2
     (11,17)=0.2 (11,18)=0.2 (11,19)=0.2 (11,20)=0.2
     (12,13)=0.2 (12,14)=0.2 (12,15)=0.2 (12,16)=0.2
     (12,17)=0.2 (12,18)=0.2 (12,19)=0.2 (12,20)=0.2
     (13,14)=0.2 (13,15)=0.2 (13,16)=0.2 (13,17)=0.2
     (13,18)=0.2 (13,19)=0.2 (13,20)=0.2 (14,15)=0.2
     (14,16)=0.2 (14,17)=0.2 (14,18)=0.2 (14,19)=0.2
     (14,20)=0.2 (15,16)=0.2 (15,17)=0.2 (15,18)=0.2
     (15,19)=0.2 (15,20)=0.2 (16,17)=0.2 (16,18)=0.2
     (16,19)=0.2 (16,20)=0.2 (17,18)=0.2 (17,19)=0.2
     (17,20)=0.2 (18,19)=0.2  (19,20)=0.2             end
ucnc (21:31,21:31)=0.2 (21:31,33:39)=0.2 (33:39,33:39)=0.2 end
end corr
end
```

Example 6.8.2: Echo of TSURFER input parameters.

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+                                                                                  +
+                                T S U R F E R                                     +
+                                                                                  +
+    TITLE: tsurfer sample problem                                                 +
+                                                                                  +
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                         I N P U T  D A T A

 PARAMETER         VALUE       DESCRIPTION

 absolute          false       Print uncertainty values and penalty
                               assessments in absolute format.  This is
                               the default format. Relative format can be
                               specified using the "rel" keyword in the
                               APPLICATIONS, EXPERIMENTS, or RESPONSE
                               input blocks.


 adjcut             1.0000E-05 Cutoff value for the cross-section
                               adjustment edits. If the maximum
                               group-wise relative adjustment for a given
                               cross-section is less than adjcut, then it
                               is omitted from the cross-section
                               adjustment table.
 adj_cov_cut        1.0000E-06 Cutoff value for including an adjusted
                               cross-section-covariance matrix in the
                               post adjustment analysis and data file.
                               If a nuclide-reaction to nuclide-reaction
                               covariance contains no values exceeding
                               adj_cov_cut,the matrix is excluded from
                               further analysis.  Note that adj_cov_cut
                               represents a variance, not a standard
                               deviation.
```

(continues on next page)

(continued from previous page)

```
bin_width=          1.000E-02  Size of similarity bins for cumulative
                               iteration edits.


cov_fix             true       Replace zero and large values on diagonal
                               of  cross-section covariance data with
                               user input values and dcov value.


cov_unit=           33         Logical unit for cross-section covariance
                               data.


calc_cumul_effect true         Perform cumulative iteration edit.


chi_sq_filter=      delta_chi  Method used for chi^2 filter analysis.
                               Possible values are:

                               independent - use independent chi^2
                                             filtering method.

                               diagonal    - use diagonal    chi^2
                                             filtering method.



                               iter_diag   - use iterative diagonal
                                             chi^2 filtering method.

                               delta_chi   - use iterative delta-chi
                                             chi^2 filtering method.

large_cov=          10.0000    Cutoff fractional standard deviation
                               value for cov_fix.


nohtml              false      Flag to cause HTML output to not be
                               produced.


print=              regular    Level of output edits for this analysis
                               (minimum or regular).


print_sim_matrix    false      Option to print similarity matrix.


print_adjustments   true       Option to print 1-D cross-section
                               adjustments.


print_init_corr     true       Option to print initial response
                               correlation matrix.


print_adj_corr      true       Option to adjusted response correlation
                               matrix.


ref_app=            40         Index to reference application response.

relative            true       Print uncertainty values and penalty
                               assessments in relative format.  This is
                               the default format. Absolute format can be
```

(continues on next page)

(continued from previous page)

```
                              specified using the "abs" keyword in the
                              APPLICATIONS, EXPERIMENTS, or RESPONSE
                              input blocks.


return_work_cov   true        Option to copy the working covariance
                              data file back to the return directory.


return_adj_cov    true        Option to copy the adjusted covariance
                              data file back to the return directory. If
                              return_adj_cov is false, the adjusted
                              covariance data file is not created.


sen_unit=         41          Logical unit for sensitivity data files.


sim_type=         C           Criteria used to calculate sim matrix.

                              Possible values are:

                              E - Calculate the similarity matrix
                                 using Esum correlation coefficients.

                              G - Calculate the similarity matrix
                                 using Gm  correlation coefficients.

                              C - Calculate the similarity matrix
                                 using Ck  correlation coefficients.


target_chi=        3.000E+00  Target chi-square per degree of freedom
                              for consistency acceptance.


udcov=             0.0500     User-defined default value of standard
                              deviation in cross-section data to use for
                              nuclide-reaction pairs for which
                              covariance data are not available on the
                              selected data file.


udcov_corr=        1.0000     User-defined default correlation value to
                              use for nuclide-reaction pairs for which
                              covariance data are not available on the
                              selected data file.


udcov_corr_type=  zone        User-defined default correlation in
                              cross-section data to use for
                              nuclide-reaction pairs for which
                              covariance data are not available on the
                              selected data file.  (long, zone, short)


udcov_fast=        0.0000     User-defined default value of standard
                              deviation in cross-section data to use for
                              fast data for nuclide-reaction pairs for
                              which covariance data are not available on
                              the selected data file.


udcov_inter=       0.0000     User-defined default value of standard
                              deviation in cross-section data to use for
```

(continues on next page)

(continued from previous page)

```
                              intermediate data for nuclide-reaction
                              pairs for which covariance data are not
                              available on the selected data file.


udcov_therm=       0.0000     User-defined default value of standard
                              deviation in cross-section data to use for
                              thermal data for nuclide-reaction pairs
                              for which covariance data are not
                              available on the selected data file.

uncert_long       false       Prints extended table of uncertainty in
                              response due to covariance data.

use_dcov          true        Use user-defined default covariance data,
                              udcov, for nuclide reaction pairs not
                              included on the covariance data file.


use_diff_groups   true        Allow sensitivity data files to have
                              different energy group structures.

usename           false       Use the name of the sensitivity data file
                              in place of its title in all output.


use_icov          true        Use user-defined data input in COVARIANCE
                              input data block in place of udcov value
                              for user input nuclide-reaction pairs that
                              are not on the covariance data file.



USER COVARIANCE DATA

  ZA   NUCLIDE REACTION  MT      ALL      THERMAL     INTER       FAST     CORREL    TYPE
------ ------- -------- ----  ---------- ---------- ---------- ---------- --------  ------
 92235 u-235   elastic     2   7.00E-02   0.00E+00   0.00E+00   0.00E+00    1.00     zone
 92238 u-238   elastic     2   6.00E-02   0.00E+00   0.00E+00   0.00E+00    1.00     zone


HTML Format Options


PARAMETER          VALUE        DESCRIPTION
---------          -----        -----------
  bg_clr=          papayawhip   Background color

  h1_clr=          maroon       Color used for major headings

  h2_clr=          navy         Color used for sub-headings

  txt_clr=         black        Color for plain text

  lnk_clr=         navy         Color for hyperlinks

  lnk_dec=         none         Decoration for hyperlinks (none, underline, overline, line-through, blink)

  vlnk_clr=        navy         Color for visited hyperlinks

                    <<<<  GENERALIZED LEAST-SQUARE ANALYSIS >>>>
```

(continues on next page)

(continued from previous page)

```
 Name of cross section COV file          : 44groupcov
 Name of working cross section COV file  : tsurfer.wrk.cov
 Name of adjusted cross section COV file : tsurfer.adj.cov
 Name of adjusted cross section PLT file : tsurfer.xs-adjust.plt
 Number of groups on COV file            :   44
           =>All sensitivity coefficients will be converted into COV group structure
```

Example 6.8.3: TSURFER covariance warnings edit.

```
Generating working covariance matrix ...


 ------------------------------------------------------------------------------
 Covariance Warnings in creating working COVERX library
 ------------------------------------------------------------------------------

 WARNING: cov_fix applied for  b-10 n,p
   Default standard deviation data value 0.0500 will replace 0.0000 for group 15
   Default standard deviation data value 0.0500 will replace 0.0000 for group 16
   Default standard deviation data value 0.0500 will replace 0.0000 for group 17
   Default standard deviation data value 0.0500 will replace 0.0000 for group 18
   Default standard deviation data value 0.0500 will replace 0.0000 for group 19
   Default standard deviation data value 0.0500 will replace 0.0000 for group 20
   Default standard deviation data value 0.0500 will replace 0.0000 for group 21
   Default standard deviation data value 0.0500 will replace 0.0000 for group 22
   Default standard deviation data value 0.0500 will replace 0.0000 for group 23
   Default standard deviation data value 0.0500 will replace 0.0000 for group 24
   Default standard deviation data value 0.0500 will replace 0.0000 for group 25
   Default standard deviation data value 0.0500 will replace 0.0000 for group 26
   Default standard deviation data value 0.0500 will replace 0.0000 for group 27
   Default standard deviation data value 0.0500 will replace 0.0000 for group 28
   Default standard deviation data value 0.0500 will replace 0.0000 for group 29
   Default standard deviation data value 0.0500 will replace 0.0000 for group 30
   Default standard deviation data value 0.0500 will replace 0.0000 for group 31
   Default standard deviation data value 0.0500 will replace 0.0000 for group 32
   Default standard deviation data value 0.0500 will replace 0.0000 for group 33
   Default standard deviation data value 0.0500 will replace 0.0000 for group 34
   Default standard deviation data value 0.0500 will replace 0.0000 for group 35
   Default standard deviation data value 0.0500 will replace 0.0000 for group 36
   Default standard deviation data value 0.0500 will replace 0.0000 for group 37
   Default standard deviation data value 0.0500 will replace 0.0000 for group 38
   Default standard deviation data value 0.0500 will replace 0.0000 for group 39
   Default standard deviation data value 0.0500 will replace 0.0000 for group 40
   Default standard deviation data value 0.0500 will replace 0.0000 for group 41
   Default standard deviation data value 0.0500 will replace 0.0000 for group 42
   Default standard deviation data value 0.0500 will replace 0.0000 for group 43
   Default standard deviation data value 0.0500 will replace 0.0000 for group 44

 ...
 Working covariance matrix created for future processing.
```

Example 6.8.4: TSURFER response list edit.

```
Number of Input Sensitivity Files = 40
   =>Number of Applications (passive) Included in GLLSM:   3
   =>Number of Benchmarks (active) Included in GLLSM   :  37
   =>Number of Responses (active+passive) used in GLLSM:  40
   =>Number of Input Systems Omitted from GLLSM(*)     :   0

                        ** Description of Prior Responses **
  RESP.#   EXPERIMENT NAME        SENS. TITLE     USE   TYPE     CALC         EXP      Ck W/ REFERENCE␣
↪APPLICATION
```

(continues on next page)

(continued from previous page)

1	1_hst001-1	r1	expt	keff	1.0015E+00	1.0000E+00	9.597E-01
2	2_hst001-1	r2	expt	keff	9.9852E-01	1.0000E+00	9.569E-01
3	3_hst001-1	r3	appl	keff	1.0024E+00	<(NA)>	9.589E-01
4	4_hst001-1	r4	expt	keff	1.0008E+00	1.0000E+00	9.558E-01
5	5_hst001-1	r5	expt	keff	1.0011E+00	1.0000E+00	9.640E-01
6	6_hst001-1	r6	expt	keff	1.0046E+00	1.0000E+00	9.640E-01
7	7_hst001-1	r7	expt	keff	9.9994E-01	1.0000E+00	9.593E-01
8	8_hst001-1	r8	expt	keff	9.9975E-01	1.0000E+00	9.589E-01
9	9_hst001-1	r9	expt	keff	9.9634E-01	1.0000E+00	9.561E-01
10	10_hst001-1	r10	expt	keff	9.9521E-01	1.0000E+00	9.660E-01
11	11_hst001-1	rot2 tank in cen	expt	keff	1.0046E+00	1.0000E+00	9.723E-01
12	12_hst001-1	rot7 tank in cen	expt	keff	1.0010E+00	1.0000E+00	9.609E-01
13	13_hst001-1	rot38 tank in ce	expt	keff	1.0009E+00	1.0000E+00	9.661E-01
14	14_hst001-1	rot4 tank in cen	expt	keff	1.0019E+00	1.0000E+00	9.638E-01
15	15_hst001-1	rot14 tank in ce	expt	keff	1.0049E+00	1.0000E+00	9.621E-01
16	16_hst001-1	rot29 tank in ce	expt	keff	9.9888E-01	1.0000E+00	9.641E-01
17	17_hst001-1	ol3ne 15.5 in. s	expt	keff	1.0031E+00	1.0000E+00	3.882E-01
18	18_hst001-1	case 30 experime	expt	keff	9.9962E-01	1.0000E+00	9.858E-01
19	19_hst001-1	heu-sol-therm-02	expt	keff	1.0023E+00	1.0000E+00	9.720E-01
20	20_hst001-1	heu-sol-therm-02	expt	keff	1.0031E+00	1.0000E+00	9.744E-01
21	21_hst001-1	heu-sol-therm-02	expt	keff	1.0060E+00	1.0000E+00	9.767E-01
22	22_hst001-1	heu-sol-therm-02	expt	keff	9.9845E-01	1.0000E+00	9.660E-01
23	23_hst001-1	heu-sol-therm-02	expt	keff	1.0050E+00	1.0000E+00	9.828E-01
24	24_hst001-1	heu-sol-therm-02	expt	keff	1.0086E+00	1.0000E+00	9.937E-01
25	25_hst001-1	heu-sol-therm-02	expt	keff	1.0014E+00	1.0000E+00	9.952E-01
26	26_hst001-1	heu-sol-therm-02	expt	keff	9.9857E-01	1.0000E+00	9.988E-01
27	27_hst001-1	heu-sol-therm-02	expt	keff	1.0045E+00	1.0000E+00	9.989E-01
28	28_hst001-1	heu-sol-therm-02	expt	keff	1.0051E+00	1.0000E+00	9.971E-01
29	29_hst001-1	heu-sol-therm-02	expt	keff	1.0057E+00	1.0000E+00	9.931E-01
30	30_hst001-1	heu-sol-therm-03	expt	keff	9.9998E-01	1.0000E+00	9.798E-01
31	31_hst001-1	heu-sol-therm-03	expt	keff	1.0014E+00	1.0000E+00	9.862E-01
32	32_hst001-1	heu-sol-therm-03	appl	keff	9.9990E-01	<(NA)>	9.852E-01
33	33_hst001-1	heu-sol-therm-03	expt	keff	1.0083E+00	1.0000E+00	9.820E-01
34	34_hst001-1	heu-sol-therm-03	expt	keff	1.0036E+00	1.0000E+00	9.910E-01
35	35_hst001-1	heu-sol-therm-03	expt	keff	1.0048E+00	1.0000E+00	9.936E-01
36	36_hst001-1	heu-sol-therm-03	expt	keff	1.0029E+00	1.0000E+00	9.979E-01
37	37_hst001-1	heu-sol-therm-03	expt	keff	1.0045E+00	1.0000E+00	9.944E-01
38	38_hst001-1	heu-sol-therm-03	expt	keff	1.0051E+00	1.0000E+00	9.978E-01
39	39_hst001-1	heu-sol-therm-03	expt	keff	1.0043E+00	1.0000E+00	9.967E-01
40	40_hst001-1	heu-sol-therm-03	appl	keff	1.0017E+00	<(NA)>	1.000E+00
31	31_hst001-1	heu-sol-therm-03	expt	keff	9.9987E-01	1.0000E+00	9.565E-01
32	32_hst001-1	heu-sol-therm-03	appl	keff	9.9894E-01	<(NA)>	9.527E-01
33	33_hst001-1	heu-sol-therm-03	expt	keff	1.0069E+00	1.0000E+00	9.782E-01
34	34_hst001-1	heu-sol-therm-03	expt	keff	1.0033E+00	1.0000E+00	9.820E-01
35	35_hst001-1	heu-sol-therm-03	expt	keff	1.0042E+00	1.0000E+00	9.831E-01
36	36_hst001-1	heu-sol-therm-03	expt	keff	1.0026E+00	1.0000E+00	9.878E-01
37	37_hst001-1	heu-sol-therm-03	expt	keff	1.0039E+00	1.0000E+00	9.842E-01
38	38_hst001-1	heu-sol-therm-03	expt	keff	1.0040E+00	1.0000E+00	9.800E-01
39	39_hst001-1	heu-sol-therm-03	expt	keff	1.0028E+00	1.0000E+00	9.868E-01
40	40_hst001-1	heu-sol-therm-03	appl	keff	1.0007E+00	<(NA)>	1.000E+00

ave. calc - exp value |C-E| = 3.1229E-03
sample stand. dev. in |C-E| = 2.2236E-03
Chi-fission spectrum sensitivities are constrained.

Example 6.8.5: Uncertainty components edit.

```
** Input Experiment Response Uncertainties **

  RESP.#  TYPE   UNC. UNITS     TOTAL      UNCERT.  COMPONENTS (std. dev.):
                             STD DEV      pyra       ucna       b10a       h/ua       ucnb       h/ub       pyrc       ucnc       b10c       h/uc
  1     keff  % dk/k          6.97782E-1     4.2000E-1  2.1000E-1  3.0000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  2     keff  % dk/k          6.53682E-1     3.2000E-1  2.5000E-1  3.2000E-1  4.0000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  3     keff  % dk/k          0.00000E+0     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  4     keff  % dk/k          4.86929E-1     3.9000E-1  1.5000E-1  0.0000E+0  2.5000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  5     keff  % dk/k          6.07454E-1     4.0000E-1  2.1000E-1  2.5000E-1  3.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  6     keff  % dk/k          5.57225E-1     0.0000E+0  2.1000E-1  3.0000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  7     keff  % dk/k          6.18142E-1     2.2000E-1  2.2000E-1  3.3000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  8     keff  % dk/k          6.65582E-1     2.2000E-1  2.5000E-1  3.6000E-1  4.5000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  9     keff  % dk/k          4.69574E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  10    keff  % dk/k          5.73062E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.8000E-1  5.0000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  11    keff  % dk/k          4.45533E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  3.1000E-1  3.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  12    keff  % dk/k          4.69574E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  13    keff  % dk/k          4.69574E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  14    keff  % dk/k          2.45967E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  1.1000E-1  2.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  15    keff  % dk/k          4.69574E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  16    keff  % dk/k          4.69574E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  17    keff  % dk/k          4.69574E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  18    keff  % dk/k          4.69574E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  19    keff  % dk/k          4.69574E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  20    keff  % dk/k          4.69574E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  4.2000E-1  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  21    keff  % dk/k          6.97782E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  4.2000E-1  2.1000E-1  3.0000E-1  4.2000E-1
  22    keff  % dk/k          7.30685E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  4.5000E-1  2.5000E-1  3.3000E-1  4.0000E-1
  23    keff  % dk/k          6.99071E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  3.9000E-1  2.1000E-1  3.0000E-1  4.5000E-1
  24    keff  % dk/k          5.95651E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  4.2000E-1  3.0000E-1  2.0000E-1  2.2000E-1
  25    keff  % dk/k          6.97782E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  4.2000E-1  2.1000E-1  3.0000E-1  4.2000E-1
  26    keff  % dk/k          6.58863E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.2000E-1  3.1000E-1  4.0000E-1  3.6000E-1
  27    keff  % dk/k          6.97782E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  4.2000E-1  2.1000E-1  3.0000E-1  4.2000E-1
  28    keff  % dk/k          6.71863E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  3.2000E-1  2.1000E-1  4.5000E-1  3.2000E-1
  29    keff  % dk/k          6.77126E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  4.2000E-1  2.5000E-1  3.0000E-1  3.6000E-1
  30    keff  % dk/k          5.57225E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  3.0000E-1  4.2000E-1
  31    keff  % dk/k          6.02080E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  3.1000E-1  3.0000E-1  4.2000E-1
  32    keff  % dk/k          0.00000E+0     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
  33    keff  % dk/k          6.32851E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  3.1000E-1  4.0000E-1  3.8000E-1
  34    keff  % dk/k          5.57225E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  3.0000E-1  4.2000E-1
  35    keff  % dk/k          6.42336E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.5000E-1  4.0000E-1  4.3600E-1
  36    keff  % dk/k          6.32851E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  3.1000E-1  4.0000E-1  3.8000E-1
  37    keff  % dk/k          5.57225E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  3.0000E-1  4.2000E-1
  38    keff  % dk/k          5.57225E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  4.1000E-1  2.0000E-1  3.2000E-1
  39    keff  % dk/k          5.57225E-1     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  2.1000E-1  3.0000E-1  4.2000E-1
  40    keff  % dk/k          0.00000E+0     0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0  0.0000E+0
```

Example 6.8.6: Chi-squared analysis edit.

```
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
                  CHI-SQUARED ANALYSIS FOR INCLUDED EXPERIMENTS


     INITIAL CHI-SQUARE PER DEGR. OF FREEDOM  =          2.781E-01
     TARGET  CHI-SQUARE PER DEGR. OF FREEDOM  =          3.000E+00
     FINAL   CHI-SQUARE PER DEGR. OF FREEDOM  =          2.781E-01

       * final number of degrees of freedom               37

       * final chi-square per degr. of freedom....
          diagonal contribution                        4.556E-01

       * final chi-square per degr. of freedom....
          without correlations                         1.561E-01

  ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
```

Example 6.8.7: Prior experiment correlation matrix edit.

```
Prior Experimental-Response Correlation Matrix

   resp   resp  1      resp  2      resp  3      resp  4      resp  5      resp  6      resp  7      resp  8
    1    1.000E+00    1.556E-01    0.000E+00    3.696E-01    2.824E-01    1.134E-01    1.461E-01    2.234E-01
    2    1.556E-01    1.000E+00    0.000E+00    3.216E-01    2.521E-01    1.211E-01    1.452E-01    2.086E-01
    3    0.000E+00    0.000E+00    1.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00
    4    3.696E-01    3.216E-01    0.000E+00    1.000E+00    5.426E-01    3.018E-01    3.718E-01    3.638E-01
    5    2.824E-01    2.521E-01    0.000E+00    5.426E-01    1.000E+00    4.219E-01    1.431E-01    4.222E-01
    6    1.134E-01    1.211E-01    0.000E+00    3.018E-01    4.219E-01    1.000E+00    5.170E-01    3.907E-01
    7    1.461E-01    1.452E-01    0.000E+00    3.718E-01    1.431E-01    5.170E-01    1.000E+00    4.264E-01
    8    2.234E-01    2.086E-01    0.000E+00    3.638E-01    4.222E-01    3.907E-01    4.264E-01    1.000E+00
    9    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00
       resp   10 thru resp   40 same as above

   resp   resp  9      resp 10      resp 11      resp 12      resp 13      resp 14      resp 15      resp 16
    1    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00
       resp    2 thru resp    8 same as above
    9    1.000E+00    1.998E-01    1.907E-01    2.000E-01    2.000E-01    2.000E-01    2.000E-01    2.000E-01
   10    1.998E-01    1.000E+00    1.933E-01    1.998E-01    1.998E-01    1.998E-01    1.998E-01    1.998E-01
   11    1.907E-01    1.933E-01    1.000E+00    1.907E-01    1.907E-01    1.907E-01    1.907E-01    1.907E-01
   12    2.000E-01    1.998E-01    1.907E-01    1.000E+00    2.000E-01    2.000E-01    2.000E-01    2.000E-01
   13    2.000E-01    1.998E-01    1.907E-01    2.000E-01    1.000E+00    2.000E-01    2.000E-01    2.000E-01
   14    2.000E-01    1.998E-01    1.907E-01    2.000E-01    2.000E-01    1.000E+00    2.000E-01    2.000E-01
   15    2.000E-01    1.998E-01    1.907E-01    2.000E-01    2.000E-01    2.000E-01    1.000E+00    2.000E-01
   16    2.000E-01    1.998E-01    1.907E-01    2.000E-01    2.000E-01    2.000E-01    2.000E-01    1.000E+00
   17    2.000E-01    1.998E-01    1.907E-01    2.000E-01    2.000E-01    2.000E-01    2.000E-01    2.000E-01
       resp   18 thru resp   20 same as above
   21    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00
       resp   22 thru resp   40 same as above

   resp   resp 17      resp 18      resp 19      resp 20      resp 21      resp 22      resp 23      resp 24
    1    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00
       resp    2 thru resp    8 same as above
    9    2.000E-01    2.000E-01    2.000E-01    2.000E-01    0.000E+00    0.000E+00    0.000E+00    0.000E+00
   10    1.998E-01    1.998E-01    1.998E-01    1.998E-01    0.000E+00    0.000E+00    0.000E+00    0.000E+00
   11    1.907E-01    1.907E-01    1.907E-01    1.907E-01    0.000E+00    0.000E+00    0.000E+00    0.000E+00
   12    2.000E-01    2.000E-01    2.000E-01    2.000E-01    0.000E+00    0.000E+00    0.000E+00    0.000E+00
       resp   13 thru resp   16 same as above
   17    1.000E+00    2.000E-01    2.000E-01    2.000E-01    0.000E+00    0.000E+00    0.000E+00    0.000E+00
   18    2.000E-01    1.000E+00    2.000E-01    0.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00
```

(continues on next page)

(continued from previous page)

```
    19    2.000E-01    2.000E-01    1.000E+00    2.000E-01    0.000E+00    0.000E+00    0.000E+00    0.000E+00
    20    2.000E-01    0.000E+00    2.000E-01    1.000E+00    0.000E+00    0.000E+00    0.000E+00    0.000E+00
    21    0.000E+00    0.000E+00    0.000E+00    0.000E+00    1.000E+00    2.059E-02    1.808E-02    3.032E-02
...
```

Example 6.8.8: Prior calculated response correlation matrix edit.

```
Prior Calculated-Response Correlation Matrix (omitted responses are 0)

    resp   resp  1      resp  2      resp  3      resp  4      resp  5      resp  6      resp  7      resp  8
     1    1.000E+00    9.941E-01    9.988E-01    9.922E-01    9.918E-01    9.932E-01    9.988E-01    9.988E-01
     2    9.941E-01    1.000E+00    9.925E-01    9.987E-01    9.768E-01    9.790E-01    9.921E-01    9.928E-01
     3    9.988E-01    9.925E-01    1.000E+00    9.931E-01    9.931E-01    9.945E-01    1.000E+00    1.000E+00
     4    9.922E-01    9.987E-01    9.931E-01    1.000E+00    9.769E-01    9.791E-01    9.926E-01    9.933E-01
     5    9.918E-01    9.768E-01    9.931E-01    9.769E-01    1.000E+00    9.999E-01    9.936E-01    9.928E-01
     6    9.932E-01    9.790E-01    9.945E-01    9.791E-01    9.999E-01    1.000E+00    9.950E-01    9.943E-01
     7    9.988E-01    9.921E-01    1.000E+00    9.926E-01    9.936E-01    9.950E-01    1.000E+00    1.000E+00
     8    9.988E-01    9.928E-01    1.000E+00    9.933E-01    9.928E-01    9.943E-01    1.000E+00    1.000E+00
     9    9.922E-01    9.986E-01    9.931E-01    1.000E+00    9.770E-01    9.792E-01    9.927E-01    9.934E-01
    10    9.934E-01    9.793E-01    9.947E-01    9.794E-01    9.998E-01    1.000E+00    9.951E-01    9.944E-01
    11    9.923E-01    9.847E-01    9.927E-01    9.836E-01    9.883E-01    9.896E-01    9.926E-01    9.927E-01
    12    9.943E-01    9.992E-01    9.923E-01    9.977E-01    9.786E-01    9.805E-01    9.920E-01    9.925E-01
    13    9.901E-01    9.748E-01    9.913E-01    9.750E-01    9.995E-01    9.992E-01    9.919E-01    9.909E-01
    14    9.995E-01    9.929E-01    9.983E-01    9.909E-01    9.935E-01    9.946E-01    9.984E-01    9.982E-01
    15    9.985E-01    9.918E-01    9.997E-01    9.924E-01    9.945E-01    9.957E-01    9.998E-01    9.996E-01
    16    9.932E-01    9.983E-01    9.942E-01    9.994E-01    9.805E-01    9.825E-01    9.938E-01    9.944E-01
    17    3.190E-01    3.558E-01    3.167E-01    3.588E-01    3.060E-01    3.064E-01    3.164E-01    3.175E-01
    18    9.745E-01    9.771E-01    9.756E-01    9.771E-01    9.692E-01    9.705E-01    9.754E-01    9.757E-01
    19    9.782E-01    9.598E-01    9.774E-01    9.579E-01    9.925E-01    9.915E-01    9.783E-01    9.771E-01
    20    9.804E-01    9.626E-01    9.785E-01    9.592E-01    9.923E-01    9.915E-01    9.793E-01    9.781E-01
    21    9.860E-01    9.715E-01    9.831E-01    9.670E-01    9.909E-01    9.907E-01    9.836E-01    9.829E-01
    22    9.994E-01    9.918E-01    9.979E-01    9.893E-01    9.939E-01    9.950E-01    9.980E-01    9.978E-01
    23    9.715E-01    9.696E-01    9.715E-01    9.684E-01    9.659E-01    9.673E-01    9.713E-01    9.717E-01
    24    9.738E-01    9.709E-01    9.735E-01    9.697E-01    9.723E-01    9.731E-01    9.735E-01    9.736E-01
    25    9.697E-01    9.666E-01    9.694E-01    9.654E-01    9.696E-01    9.702E-01    9.695E-01    9.695E-01
    26    9.581E-01    9.543E-01    9.576E-01    9.531E-01    9.621E-01    9.622E-01    9.579E-01    9.576E-01
    27    9.653E-01    9.621E-01    9.647E-01    9.609E-01    9.675E-01    9.678E-01    9.650E-01    9.647E-01
    28    9.756E-01    9.732E-01    9.748E-01    9.719E-01    9.750E-01    9.756E-01    9.750E-01    9.749E-01
    29    9.833E-01    9.815E-01    9.826E-01    9.803E-01    9.798E-01    9.807E-01    9.826E-01    9.826E-01
    30    9.828E-01    9.674E-01    9.828E-01    9.659E-01    9.914E-01    9.912E-01    9.833E-01    9.826E-01
    31    9.663E-01    9.478E-01    9.663E-01    9.461E-01    9.828E-01    9.817E-01    9.671E-01    9.659E-01
    32    9.511E-01    9.310E-01    9.511E-01    9.293E-01    9.717E-01    9.702E-01    9.521E-01    9.507E-01
    33    9.875E-01    9.864E-01    9.870E-01    9.854E-01    9.796E-01    9.810E-01    9.868E-01    9.871E-01
    34    9.830E-01    9.809E-01    9.825E-01    9.798E-01    9.788E-01    9.799E-01    9.825E-01    9.826E-01
    35    9.773E-01    9.749E-01    9.770E-01    9.738E-01    9.750E-01    9.759E-01    9.770E-01    9.771E-01
    36    9.680E-01    9.649E-01    9.676E-01    9.637E-01    9.692E-01    9.697E-01    9.678E-01    9.676E-01
    37    9.738E-01    9.711E-01    9.736E-01    9.701E-01    9.725E-01    9.733E-01    9.736E-01    9.737E-01
```

Example 6.8.9: Adjusted response correlation matrix edit.

```
Adjusted-Response Correlation Matrix

    resp   resp  1      resp  2      resp  3      resp  4      resp  5      resp  6      resp  7      resp  8
     1    1.000E+00    8.916E-01    9.629E-01    8.530E-01    8.277E-01    8.526E-01    9.625E-01    9.634E-01
     2    8.916E-01    1.000E+00    8.490E-01    9.648E-01    5.905E-01    6.214E-01    8.413E-01    8.529E-01
     3    9.629E-01    8.490E-01    1.000E+00    8.796E-01    8.663E-01    8.917E-01    9.997E-01    9.998E-01
     4    8.530E-01    9.648E-01    8.796E-01    1.000E+00    6.189E-01    6.501E-01    8.721E-01    8.838E-01
     5    8.277E-01    5.905E-01    8.663E-01    6.189E-01    1.000E+00    9.985E-01    8.765E-01    8.606E-01
     6    8.526E-01    6.214E-01    8.917E-01    6.501E-01    9.985E-01    1.000E+00    9.009E-01    8.865E-01
     7    9.625E-01    8.413E-01    9.997E-01    8.721E-01    8.765E-01    9.009E-01    1.000E+00    9.992E-01
     8    9.634E-01    8.529E-01    9.998E-01    8.838E-01    8.606E-01    8.865E-01    9.992E-01    1.000E+00
```

(continues on next page)

(continued from previous page)

9	8.534E-01	9.646E-01	8.799E-01	9.999E-01	6.208E-01	6.518E-01	8.726E-01	8.842E-01
10	8.541E-01	6.235E-01	8.941E-01	6.520E-01	9.968E-01	9.990E-01	9.033E-01	8.888E-01
11	8.150E-01	6.935E-01	8.277E-01	6.829E-01	7.351E-01	7.573E-01	8.238E-01	8.291E-01
12	8.904E-01	9.808E-01	8.354E-01	9.423E-01	6.150E-01	6.419E-01	8.309E-01	8.392E-01
13	7.950E-01	5.588E-01	8.308E-01	5.896E-01	9.877E-01	9.833E-01	8.433E-01	8.244E-01
14	9.869E-01	8.703E-01	9.488E-01	8.280E-01	8.583E-01	8.786E-01	9.509E-01	9.466E-01
15	9.537E-01	8.341E-01	9.916E-01	8.669E-01	8.944E-01	9.158E-01	9.936E-01	9.896E-01
16	8.567E-01	9.549E-01	8.878E-01	9.898E-01	6.606E-01	6.891E-01	8.817E-01	8.902E-01
17	1.856E-01	2.321E-01	1.808E-01	2.363E-01	1.675E-01	1.688E-01	1.812E-01	1.819E-01
18	5.222E-01	5.967E-01	5.551E-01	6.082E-01	4.698E-01	4.846E-01	5.491E-01	5.553E-01
19	6.311E-01	3.866E-01	6.173E-01	3.713E-01	8.690E-01	8.530E-01	6.318E-01	6.111E-01
20	6.622E-01	4.169E-01	6.154E-01	3.653E-01	8.540E-01	8.398E-01	6.286E-01	6.100E-01
21	7.295E-01	5.167E-01	6.495E-01	4.285E-01	7.865E-01	7.834E-01	6.582E-01	6.456E-01
22	9.864E-01	8.495E-01	9.378E-01	7.951E-01	8.603E-01	8.801E-01	9.396E-01	9.365E-01
23	4.880E-01	4.903E-01	4.928E-01	4.794E-01	3.958E-01	4.132E-01	4.863E-01	4.972E-01
24	5.166E-01	5.003E-01	5.130E-01	4.885E-01	5.022E-01	5.113E-01	5.123E-01	5.148E-01
25	4.678E-01	4.510E-01	4.650E-01	4.406E-01	4.786E-01	4.847E-01	4.654E-01	4.662E-01
26	3.542E-01	3.359E-01	3.504E-01	3.294E-01	4.347E-01	4.321E-01	3.551E-01	3.493E-01
27	4.191E-01	4.060E-01	4.111E-01	3.969E-01	4.747E-01	4.745E-01	4.154E-01	4.104E-01
28	5.372E-01	5.292E-01	5.228E-01	5.150E-01	5.446E-01	5.493E-01	5.259E-01	5.229E-01
29	6.543E-01	6.511E-01	6.374E-01	6.342E-01	6.022E-01	6.136E-01	6.379E-01	6.391E-01
30	6.667E-01	4.381E-01	6.730E-01	4.322E-01	8.187E-01	8.159E-01	6.816E-01	6.694E-01
31	4.770E-01	2.395E-01	4.821E-01	2.341E-01	7.442E-01	7.284E-01	4.955E-01	4.767E-01
32	3.553E-01	1.294E-01	3.622E-01	1.263E-01	6.524E-01	6.326E-01	3.763E-01	3.564E-01
33	7.158E-01	7.218E-01	7.060E-01	7.095E-01	5.640E-01	5.864E-01	7.012E-01	7.099E-01
34	6.492E-01	6.403E-01	6.387E-01	6.257E-01	5.781E-01	5.927E-01	6.370E-01	6.413E-01
35	5.624E-01	5.510E-01	5.579E-01	5.404E-01	5.327E-01	5.433E-01	5.572E-01	5.598E-01
36	4.467E-01	4.327E-01	4.414E-01	4.238E-01	4.848E-01	4.872E-01	4.443E-01	4.412E-01

Example 6.8.10: Bias convergence edit.

```
*** Cumulative Conv. of (A-C)/C For Application ***

     =>Edited for reference response:   40
     =>Based on similarity parameter :   Ck
     =>Minimum similarity included   :   0.300
     =>Similarity bin width for edit :   0.010

     CUM RANGE   MAX. SIM   NO. EXP     %(A-C)/C
        NO.      INCLUDED  INCLUDED
         1        0.390        1        -0.057
         2        0.400        1        -0.057
         3        0.410        1        -0.057
         4        0.420        1        -0.057
         5        0.430        1        -0.057
         6        0.440        1        -0.057
         7        0.450        1        -0.057
         8        0.460        1        -0.057
         9        0.470        1        -0.057
        10        0.480        1        -0.057
        11        0.490        1        -0.057
        12        0.500        1        -0.057
        13        0.510        1        -0.057
        14        0.520        1        -0.057
        15        0.530        1        -0.057
        16        0.540        1        -0.057
        17        0.550        1        -0.057
        18        0.560        1        -0.057
        19        0.570        1        -0.057
        20        0.580        1        -0.057
        21        0.590        1        -0.057
        22        0.600        1        -0.057
        23        0.610        1        -0.057
```

(continues on next page)

(continued from previous page)

24	0.620	1	-0.057
25	0.630	1	-0.057
26	0.640	1	-0.057
27	0.650	1	-0.057
28	0.660	1	-0.057
29	0.670	1	-0.057
30	0.680	1	-0.057
31	0.690	1	-0.057
32	0.700	1	-0.057
33	0.710	1	-0.057
34	0.720	1	-0.057
35	0.730	1	-0.057
36	0.740	1	-0.057
37	0.750	1	-0.057
38	0.760	1	-0.057
39	0.770	1	-0.057
40	0.780	1	-0.057
41	0.790	1	-0.057
42	0.800	1	-0.057
43	0.810	1	-0.057
44	0.820	1	-0.057
45	0.830	1	-0.057
46	0.840	1	-0.057
47	0.850	1	-0.057
48	0.860	1	-0.057
49	0.870	1	-0.057
50	0.880	1	-0.057
51	0.890	1	-0.057
52	0.900	1	-0.057
53	0.910	1	-0.057
54	0.920	1	-0.057
55	0.930	1	-0.057
56	0.940	1	-0.057
57	0.950	1	-0.057
58	0.960	7	0.063
59	0.970	16	-0.076
60	0.980	21	-0.133
61	0.990	25	-0.170
62	1.000	37	-0.246

Example 6.8.11: Adjustment summary edit.

```
_____________________________________________
|                                           |
| NOTATION:  calc = prior calculated value  |
|            exp  = prior experimental value|
|            adj  = adjusted calculated value|
|                 = adjusted experimental value|
|___________________________________________|
```

++ R E S P O N S E ++	%(calc-exp) /calc	%(adj- exp) /exp	%(adj-calc) /calc	%st.dev exp	%st.dev old calc	%st.dev new adj	indepndnt chi-sq.	diagonal chi-sq
EXPT keff 1_hst001-1	1.4808E-01	-7.8217E-02	-2.2618E-01	6.9778E-01	9.2613E-01	1.6039E-01	1.6325E-02	5.2540E-02
EXPT keff 2_hst001-1	-1.4852E-01	-3.5635E-01	-2.0836E-01	6.5368E-01	9.4664E-01	1.7812E-01	1.6652E-02	5.6775E-02
*APPL keff 3_hst001-1	NA	NA	-2.2029E-01	NA	9.2936E-01	1.6290E-01	NA	NA
EXPT keff 4_hst001-1	7.4944E-02	-1.2371E-01	-1.9856E-01	4.8693E-01	9.4552E-01	1.8134E-01	4.9671E-03	3.9194E-02
EXPT keff 5_hst001-1	1.1098E-01	-1.1748E-01	-2.2833E-01	6.0745E-01	8.2656E-01	1.5674E-01	1.1714E-02	6.0017E-02
EXPT keff 6_hst001-1	4.5304E-01	2.2589E-01	-2.2818E-01	5.5723E-01	8.3667E-01	1.5545E-01	2.0367E-01	1.1014E+00
EXPT keff 7_hst001-1	-5.7003E-03	-2.2571E-01	-2.2002E-01	6.1814E-01	9.2336E-01	1.6172E-01	2.6316E-05	1.4614E-04
EXPT keff 8_hst001-1	-2.4606E-02	-2.4519E-01	-2.2065E-01	6.6558E-01	9.3166E-01	1.6335E-01	4.6176E-04	2.0174E-03
EXPT keff 9_hst001-1	-3.6745E-01	-5.6401E-01	-1.9864E-01	4.6957E-01	9.4345E-01	1.8067E-01	1.2139E-01	6.6398E-01
EXPT keff 10_hst001-1	-4.8080E-01	-7.0685E-01	-2.2945E-01	5.7306E-01	8.2869E-01	1.5179E-01	2.2702E-01	8.0333E-01
EXPT keff 11_hst001-1	4.5571E-01	1.8845E-01	-2.6812E-01	4.4553E-01	8.7207E-01	1.5382E-01	2.1696E-01	1.1550E+00
EXPT keff 12_hst001-1	1.0180E-01	-1.0271E-01	-2.0440E-01	4.6957E-01	8.9651E-01	1.6755E-01	1.0122E-02	5.2290E-02
EXPT keff 13_hst001-1	9.1516E-02	-1.2988E-01	-2.2127E-01	4.6957E-01	7.9206E-01	1.5393E-01	9.8828E-03	4.2829E-02
EXPT keff 14_hst001-1	1.9343E-01	-2.9197E-02	-2.2257E-01	2.4597E-01	8.8292E-01	1.5096E-01	4.4550E-02	5.6670E-01
EXPT keff 15_hst001-1	4.8573E-01	2.7272E-01	-2.1434E-01	4.6957E-01	8.9044E-01	1.5510E-01	2.3331E-01	1.2327E+00
EXPT keff 16_hst001-1	-1.1253E-01	-3.1683E-01	-2.0466E-01	4.6957E-01	8.9137E-01	1.6221E-01	1.2468E-02	6.4284E-02
EXPT keff 17_hst001-1	3.0586E-01	-5.2167E-02	-3.5787E-01	4.6957E-01	1.0537E+00	4.0553E-01	7.0374E-02	8.2612E-02
EXPT keff 18_hst001-1	-3.8014E-02	-2.9288E-01	-2.5497E-01	4.6957E-01	7.4743E-01	1.5210E-01	1.8543E-03	7.2266E-03
EXPT keff 19_hst001-1	2.2897E-01	-2.3461E-02	-2.5238E-01	4.6957E-01	7.3482E-01	1.5593E-01	6.9036E-02	2.6221E-01
EXPT keff 20_hst001-1	3.1242E-01	5.1984E-02	-2.6060E-01	4.6957E-01	7.4192E-01	1.5158E-01	1.2683E-01	4.9994E-01
EXPT keff 21_hst001-1	5.9168E-01	3.2443E-01	-2.6916E-01	6.9778E-01	7.7116E-01	1.4698E-01	3.2540E-01	7.1086E-01
EXPT keff 22_hst001-1	-1.5504E-01	-3.8713E-01	-2.3269E-01	7.3068E-01	8.7855E-01	1.4818E-01	1.8385E-02	4.3978E-02
EXPT keff 23_hst001-1	4.9385E-01	1.9665E-01	-2.9817E-01	6.9907E-01	7.8336E-01	1.6584E-01	2.2221E-01	4.8883E-01

...

Example 6.8.12: Adjusted response edit.

```
** Description of Adjusted Responses **

RESP.#   EXPERIMENT NAME        SENS. TITLE     USE   TYPE       ADJUSTED RESPONSE
1      1_hst001-1           r1                expt  keff    9.9922E-01 +/-  1.6063E-03
2      2_hst001-1           r2                expt  keff    9.9644E-01 +/-  1.7785E-03
3      3_hst001-1           r3                appl  keff    1.0002E+00 +/-  1.6330E-03
4      4_hst001-1           r4                expt  keff    9.9876E-01 +/-  1.8147E-03
5      5_hst001-1           r5                expt  keff    9.9883E-01 +/-  1.5691E-03
6      6_hst001-1           r6                expt  keff    1.0023E+00 +/-  1.5616E-03
7      7_hst001-1           r7                expt  keff    9.9774E-01 +/-  1.6171E-03
8      8_hst001-1           r8                expt  keff    9.9755E-01 +/-  1.6331E-03
9      9_hst001-1           r9                expt  keff    9.9436E-01 +/-  1.8001E-03
10      10_hst001-1          r10               expt  keff    9.9293E-01 +/-  1.5106E-03
11      11_hst001-1          rot2 tank in cen  expt  keff    1.0019E+00 +/-  1.5453E-03
12      12_hst001-1          rot7 tank in cen  expt  keff    9.9897E-01 +/-  1.6772E-03
13      13_hst001-1          rot38 tank in ce  expt  keff    9.9870E-01 +/-  1.5407E-03
14      14_hst001-1          rot4 tank in cen  expt  keff    9.9971E-01 +/-  1.5125E-03
15      15_hst001-1          rot14 tank in ce  expt  keff    1.0027E+00 +/-  1.5586E-03
16      16_hst001-1          rot29 tank in ce  expt  keff    9.9683E-01 +/-  1.6202E-03
17      17_hst001-1          ol3ne 15.5 in. s  expt  keff    9.9948E-01 +/-  4.0678E-03
18      18_hst001-1          case 30 experime  expt  keff    9.9707E-01 +/-  1.5204E-03
19      19_hst001-1          heu-sol-therm-02  expt  keff    9.9977E-01 +/-  1.5629E-03
20      20_hst001-1          heu-sol-therm-02  expt  keff    1.0005E+00 +/-  1.5205E-03
21      21_hst001-1          heu-sol-therm-02  expt  keff    1.0032E+00 +/-  1.4786E-03
22      22_hst001-1          heu-sol-therm-02  expt  keff    9.9613E-01 +/-  1.4795E-03
23      23_hst001-1          heu-sol-therm-02  expt  keff    1.0020E+00 +/-  1.6666E-03
24      24_hst001-1          heu-sol-therm-02  expt  keff    1.0059E+00 +/-  1.3850E-03
25      25_hst001-1          heu-sol-therm-02  expt  keff    9.9870E-01 +/-  1.3849E-03
26      26_hst001-1          heu-sol-therm-02  expt  keff    9.9596E-01 +/-  1.4012E-03
27      27_hst001-1          heu-sol-therm-02  expt  keff    1.0019E+00 +/-  1.3609E-03
...
```

Example 6.8.13: Application bias summary edit.

```
  ____________________________________________________________________________
  |                                                                          |
  |                                                                          |
  |                      APPLICATION AND BIAS SUMMARY                        |
  |                                                                          |
  |__________________________________________________________________________|


APPLICATION              TYPE CALC VALUE  PRIOR UNC % REL. BIAS %    BIAS     ADJ VALUE   ADJ UNC %
------------------------ ---- ----------- ----------- ----------- ----------- ----------- -----------
3_hst001-1               keff  1.0024E+00  9.2936E-01  2.2029E-01  2.2083E-03  1.0002E+00  1.6290E-01
32_hst001-1              keff  9.9990E-01  6.2270E-01  2.7119E-01  2.7116E-03  9.9719E-01  1.5301E-01
40_hst001-1              keff  1.0017E+00  6.0802E-01  2.4635E-01  2.4677E-03  9.9922E-01  1.3718E-01

NOTE: The relative bias and uncertainty values are normalized to the calculated response value.


_____________________________________________________________________________
|                                                                           |
|                                                                           |
|  CONTRIBUTION TO THE REFERENCE APPLICATION BIAS FROM NUCLIDE-REACTION PAIRS  |
|                                                                           |
|___________________________________________________________________________|


                       CONTRIB. TO BIAS   FRACTION OF BIAS
```

(continues on next page)

(continued from previous page)

NUCLIDE	REACTION	% dk/k	L1-NORM
u-235	nubar	1.2213E-01	3.1407E-01
u-235	chi	1.0203E-01	2.6237E-01
u-235	n,gamma	5.6672E-02	1.4647E-01
h-1	elastic	-4.9761E-02	1.2798E-01
u-235	fission	2.3867E-02	6.1390E-02
o-16	elastic	-2.0793E-02	5.3522E-02
fe	n,gamma	6.2393E-03	1.6044E-02
fe	elastic	2.1543E-03	5.5536E-03
h-1	n,gamma	1.0344E-03	2.6600E-03
fe	n,n'	9.3138E-04	2.3950E-03
cr	n,gamma	7.2797E-04	1.8720E-03
ni	n,gamma	3.7890E-04	9.7433E-04
u-234	n,gamma	-3.0719E-04	8.0394E-04

Figure 6.6.13. Application bias summary edit

cr	elastic	2.4825E-04	6.3853E-04
u-238	n,gamma	2.4072E-04	6.1902E-04
b-10	n,alpha	2.1488E-04	5.5255E-04
n-14	n,p	2.1113E-04	5.4291E-04
u-235	n,n'	-1.5322E-04	3.9401E-04
ni	elastic	1.0020E-04	2.5830E-04
mn-55	n,gamma	8.5311E-05	2.1938E-04
cr	n,n'	4.9517E-05	1.2733E-04
n-14	elastic	-4.0198E-05	1.0883E-04
u-235	elastic	4.1221E-06	9.1845E-05
o-16	n,alpha	3.5175E-05	9.0453E-05
u-238	elastic	8.6693E-06	4.1429E-05
b-11	elastic	1.3516E-05	3.6278E-05
ni	n,n'	1.0318E-05	2.6532E-05
mn-55	elastic	1.0216E-05	2.6270E-05
u-234	fission	-7.9124E-06	2.0356E-05
u-236	n,gamma	-5.3796E-06	1.4106E-05
u-238	n,n'	4.7750E-06	1.2292E-05
n-14	n,gamma	4.2428E-06	1.0911E-05
b-10	n,n'	3.5966E-06	9.5464E-06
n-14	n,alpha	3.5931E-06	9.2396E-06
n-14	n,n'	5.9234E-07	7.7528E-06
c	elastic	2.2321E-06	6.1840E-06
b-10	elastic	-8.9644E-07	6.1541E-06
b-10	n,p	1.5897E-06	4.0880E-06
ni	n,p	1.3670E-06	3.5153E-06
c	n,n'	-1.0409E-06	2.6767E-06
u-234	n,n'	-8.8825E-07	2.4901E-06
o-16	n,n'	-7.8209E-07	2.3129E-06
u-236	fission	-7.4671E-07	1.9550E-06
si	elastic	6.9545E-07	1.8014E-06
u-234	nubar	-6.3015E-07	1.6282E-06
u-235	n,2n	-4.4596E-07	1.1468E-06
mn-55	n,n'	4.2551E-07	1.0942E-06
u-238	nubar	3.9169E-07	1.0072E-06
fe	n,p	3.7711E-07	9.6973E-07
ti	elastic	2.3945E-07	6.1575E-07
ti	n,gamma	1.6831E-07	4.3281E-07
b-11	n,n'	1.0274E-07	2.6419E-07
o-16	n,gamma	1.0048E-07	2.5839E-07
ti	n,n'	7.4571E-08	1.9176E-07
si	n,n'	5.2998E-08	1.3628E-07
u-236	nubar	-5.0705E-08	1.3039E-07

...

Example 6.8.14: Cross-section adjustment edit.

```
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
       Multigroup Cross Section Changes Inferred from GLSS Adjustment
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::



**************** u-235       chi

  group   delta-XS   st. dev.   st. dev.   rel sen  contribution to   fraction of
  number    (%)       old(%)     new(%)     coeff.   appl. bias (%)  bias - L1-norm

     1    8.798E-06  8.125E-01  8.125E-01  1.585E-04   -1.395E-09     3.586E-09
     2   -1.194E-04  2.310E-01  2.309E-01  3.486E-04    4.160E-08     1.070E-07
     3   -1.044E-04  2.811E-01  2.811E-01  6.741E-04    7.035E-08     1.809E-07
     4   -1.079E-04  2.298E-01  2.298E-01  2.170E-03    2.342E-07     6.023E-07
     5   -9.178E-05  3.071E-01  3.071E-01  1.384E-03    1.270E-07     3.265E-07
     6   -5.405E-05  2.888E-01  2.887E-01  4.064E-04    2.197E-08     5.648E-08
     7   -5.449E-05  2.887E-01  2.886E-01  1.693E-03    9.228E-08     2.373E-07
     8   -8.681E-05  2.444E-01  2.443E-01  1.896E-03    1.646E-07     4.232E-07
     9   -1.706E-04  2.544E-01  2.543E-01  2.392E-03    4.082E-07     1.050E-06
    10   -2.719E-04  1.946E-01  1.945E-01  3.378E-03    9.187E-07     2.362E-06
    11   -3.341E-04  1.828E-01  1.827E-01  3.793E-03    1.267E-06     3.259E-06
    12   -4.506E-04  2.503E-01  2.502E-01  2.920E-03    1.316E-06     3.384E-06
    13   -1.078E-03  3.416E-01  3.415E-01  8.418E-04    9.071E-07     2.333E-06
    14   -1.230E-03  2.896E-01  2.894E-01  5.316E-03    6.538E-06     1.681E-05
    15   -1.187E-03  2.036E-01  2.034E-01  1.087E-02    1.289E-05     3.316E-05
    16   -3.654E-03  4.411E-01  4.393E-01  2.317E-02    8.467E-05     2.177E-04
    17   -4.655E-02  3.057E-01  2.995E-01  3.238E-02    1.507E-03     3.875E-03
    18   -1.238E-01  3.167E-01  2.829E-01  2.907E-02    3.598E-03     9.251E-03
    19   -1.238E-01  3.167E-01  2.829E-01  1.322E-02    1.636E-03     4.208E-03
    20   -1.238E-01  3.167E-01  2.829E-01  4.597E-03    5.689E-04     1.463E-03
    21   -1.257E-01  3.109E-01  2.752E-01  1.735E-03    2.182E-04     5.610E-04
    22   -1.337E-01  3.108E-01  2.703E-01  5.343E-03    7.141E-04     1.836E-03
    23   -1.337E-01  3.108E-01  2.703E-01  4.306E-03    5.755E-04     1.480E-03
    24   -1.337E-01  3.108E-01  2.703E-01  1.447E-02    1.934E-03     4.973E-03
    25   -1.337E-01  3.108E-01  2.703E-01  1.678E-02    2.242E-03     5.766E-03
    26   -1.337E-01  3.108E-01  2.703E-01  2.346E-02    3.136E-03     8.065E-03
    27   -1.337E-01  3.108E-01  2.703E-01  4.807E-03    6.424E-04     1.652E-03
    28   -1.337E-01  3.108E-01  2.703E-01  5.858E-03    7.830E-04     2.013E-03
    29   -1.337E-01  3.108E-01  2.703E-01  7.113E-03    9.507E-04     2.445E-03
    30   -1.337E-01  3.108E-01  2.703E-01  1.844E-02    2.464E-03     6.337E-03
    31   -1.337E-01  3.108E-01  2.703E-01  1.076E-02    1.438E-03     3.699E-03
    32   -1.337E-01  3.108E-01  2.703E-01  1.164E-02    1.555E-03     3.999E-03
    33   -1.337E-01  3.108E-01  2.703E-01  1.316E-02    1.759E-03     4.524E-03
    34   -1.337E-01  3.108E-01  2.703E-01  3.669E-02    4.904E-03     1.261E-02
    35   -1.337E-01  3.108E-01  2.703E-01  7.544E-02    1.008E-02     2.593E-02
    36   -1.337E-01  3.108E-01  2.703E-01  9.663E-02    1.292E-02     3.321E-02
    37   -1.337E-01  3.108E-01  2.703E-01  1.070E-01    1.430E-02     3.678E-02
    38   -1.337E-01  3.108E-01  2.703E-01  7.069E-02    9.448E-03     2.430E-02
    39   -1.337E-01  3.108E-01  2.703E-01  8.255E-02    1.103E-02     2.837E-02
    40   -1.337E-01  3.108E-01  2.703E-01  4.197E-02    5.609E-03     1.442E-02
    41   -1.336E-01  3.106E-01  2.700E-01  1.461E-01    1.951E-02     5.017E-02
    42   -1.331E-01  3.108E-01  2.706E-01  2.156E-02    2.871E-03     7.383E-03
    43   -1.331E-01  3.108E-01  2.706E-01  3.154E-02    4.199E-03     1.080E-02
    44   -1.331E-01  3.108E-01  2.706E-01  1.072E-02    1.427E-03     3.670E-03
 Total   -3.625E+00 ---------- ----------  9.994E-01    1.221E-01     3.141E-01

 NOTE: The contribution to the application bias is tabulated in units of % dk/k.
       The contribution to the application bias is normalized to the calculated response value.
 NOTE: The fraction of the bias L1-norm is equal to the absolute value of the bias contribution
       divided by the sum of the absolute value of all groupwise bias contributions.
```

(continues on next page)

(continued from previous page)

```
**********  u-235       chi

  group   delta-XS   st. dev.   st. dev.   rel sen  contribution to   fraction of
  number    (%)       old(%)     new(%)     coeff.   appl. bias (%)  bias - L1-norm

     1    3.011E+00  1.165E+01  8.267E+00 -2.143E-03    6.453E-03     1.659E-02
     2    2.165E+00  8.336E+00  5.877E+00 -5.079E-03    1.100E-02     2.828E-02
     3    1.528E+00  5.869E+00  4.122E+00 -1.255E-02    1.917E-02     4.930E-02
     4    8.165E-01  3.138E+00  2.200E+00 -2.584E-02    2.110E-02     5.426E-02
     5    3.741E-01  1.451E+00  1.024E+00 -8.856E-03    3.313E-03     8.518E-03
     6    2.299E-01  9.050E-01  6.464E-01 -1.880E-03    4.321E-04     1.111E-03
     7    7.697E-02  3.463E-01  2.717E-01 -3.955E-03    3.044E-04     7.827E-04

...

                              <<<<< NORMAL END OF CALCULATION >>>>>
```

6.8.6.2 HTML Output

The input file for the TSURFER sample problem is named tsurfer.inp. In this case, the HTML-formatted output is stored in a file called tsurfer.html and additional resources are stored in directories called tsurfer.htmd and applet_resources. This section contains example TSURFER HTML-formatted output only for demonstration of the interface. When tsurfer.html is opened in a web browser, the information shown in Fig. 6.8.1 is displayed. The title of the input file is displayed between the two SCALE logos. Because this SCALE input file only executed TSURFER, only a single-output listing is available. The text "1. TSURFER" is a hyperlink to view the output from TSURFER. Clicking on the "1. TSURFER" hyperlink will bring up the information shown in Fig. 6.8.2. Clicking on the SCALE logos will link the user to the SCALE website, if external internet access is available.

Fig. 6.8.1: Initial screen from TSURFER HTML output.

The initial page of output from TSURFER is shown in Fig. 6.8.2. Program verification information is shown in the table under the TSURFER logo. This table includes information about the code that was executed and the date and time it was run. The menu on the left side of the screen contains hyperlinks to specific portions of the code output. Echoes of the input data are available in the Input Data section. Any errors or warning messages are available in the Messages sections. Results from the code execution are shown in the results section.

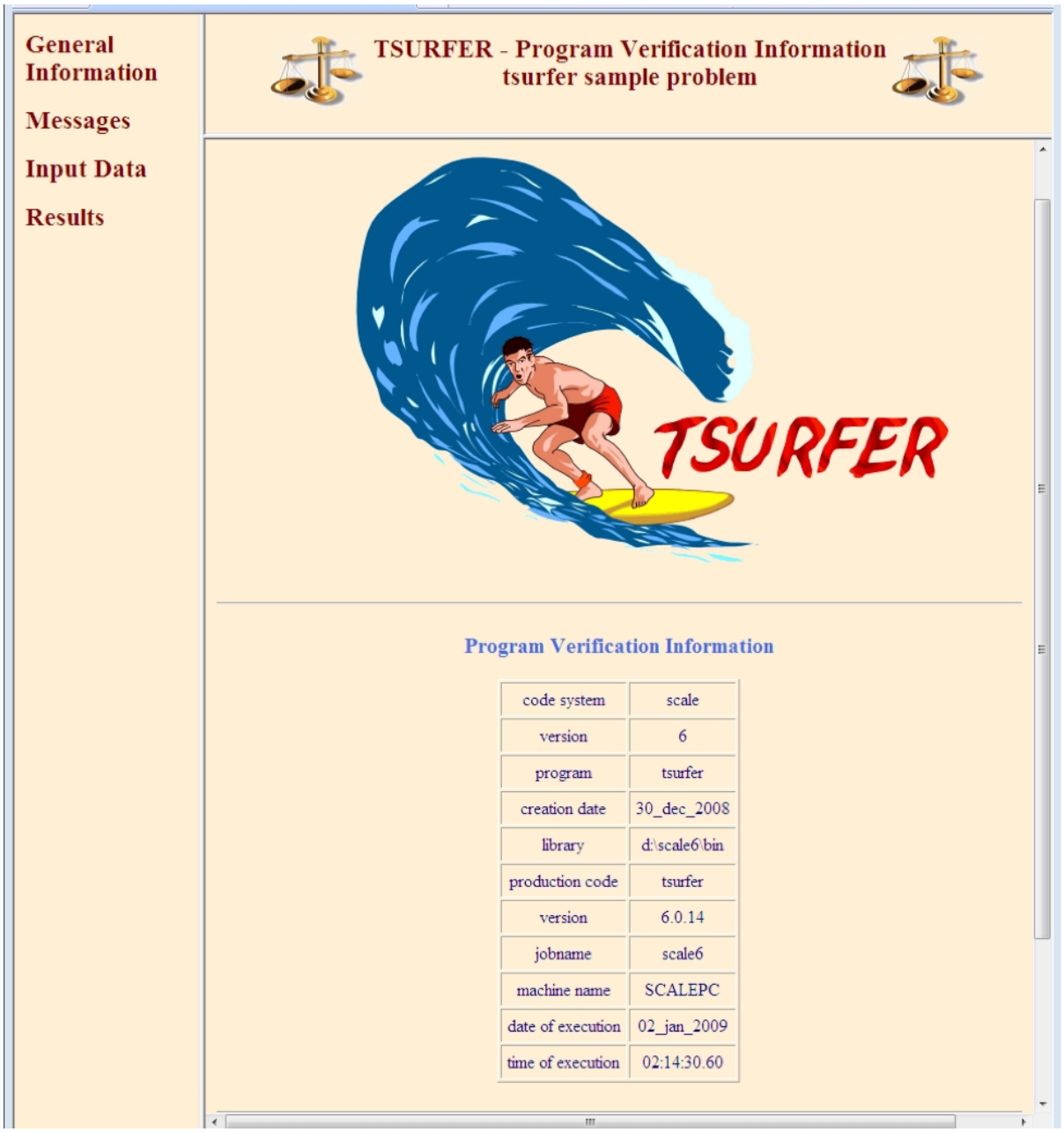
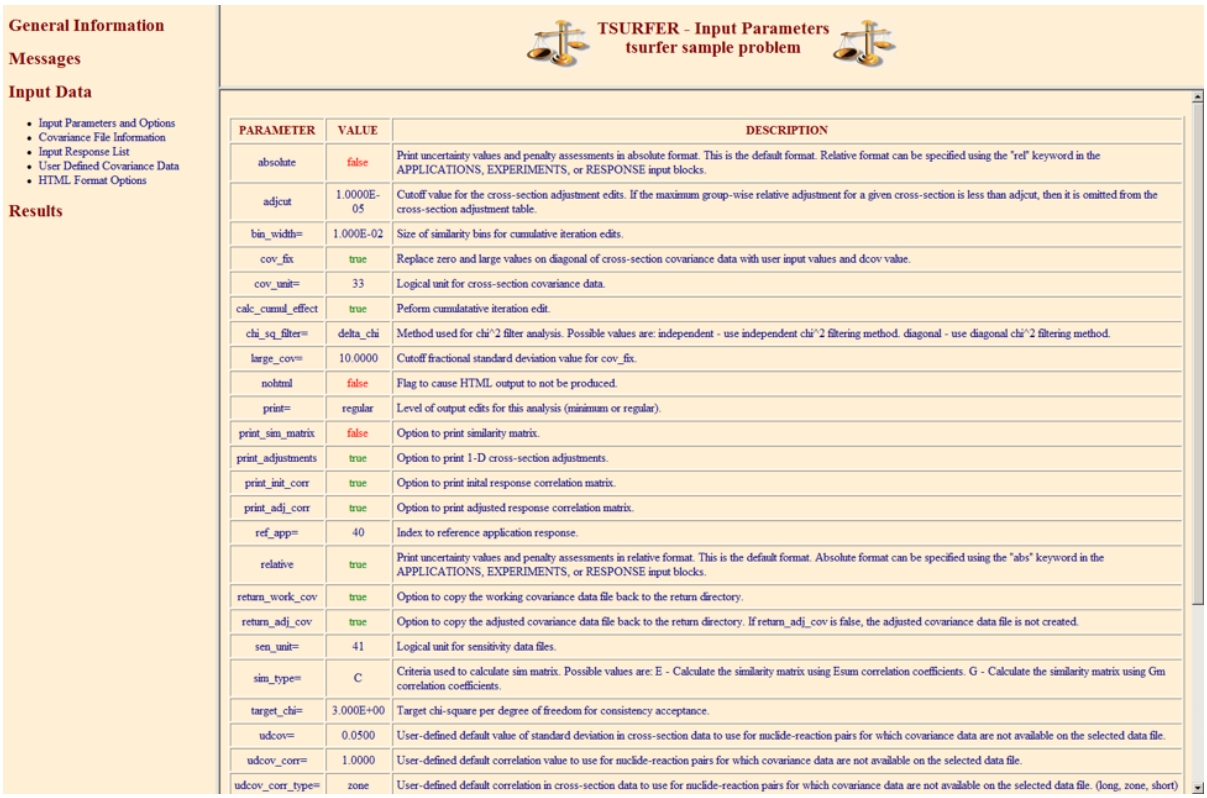


Fig. 6.8.2: Program verification screen from TSURFER HTML output.

Selecting Input Parameters will reveal the menu of available input data. Selecting Input Parameters causes the table shown in Fig. 6.8.3 to be displayed. Other input data can also be displayed by selecting the desired data from the menu.



PARAMETER	VALUE	DESCRIPTION
absolute	false	Print uncertainty values and penalty assessments in absolute format. This is the default format. Relative format can be specified using the "rel" keyword in the APPLICATIONS, EXPERIMENTS, or RESPONSE input blocks.
adjcut	1.0000E-05	Cutoff value for the cross-section adjustment edits. If the maximum group-wise relative adjustment for a given cross-section is less than adjcut, then it is omitted from the cross-section adjustment table.
bin_width=	1.000E-02	Size of similarity bins for cumulative iteration edits.
cov_fix	true	Replace zero and large values on diagonal of cross-section covariance data with user input values and dcov value.
cov_unit=	33	Logical unit for cross-section covariance data.
calc_cumul_effect	true	Peform cumulatative iteration edit.
chi_sq_filter=	delta_chi	Method used for chi^2 filter analysis. Possible values are: independent - use independent chi^2 filtering method. diagonal - use diagonal chi^2 filtering method.
large_cov=	10.0000	Cutoff fractional standard deviation value for cov_fix.
nohtml	false	Flag to cause HTML output to not be produced.
print=	regular	Level of output edits for this analysis (minimum or regular).
print_sim_matrix	false	Option to print similarity matrix.
print_adjustments	true	Option to print 1-D cross-section adjustments.
print_init_corr	true	Option to print inital response correlation matrix.
print_adj_corr	true	Option to print adjusted response correlation matrix.
ref_app=	40	Index to reference application response.
relative	true	Print uncertainty values and penalty assessments in relative format. This is the default format. Absolute format can be specified using the "abs" keyword in the APPLICATIONS, EXPERIMENTS, or RESPONSE input blocks.
return_work_cov	true	Option to copy the working covariance data file back to the return directory.
return_adj_cov	true	Option to copy the adjusted covariance data file back to the return directory. If return_adj_cov is false, the adjusted covariance data file is not created.
sen_unit=	41	Logical unit for sensitivity data files.
sim_type=	C	Criteria used to calculate sim matrix. Possible values are: E - Calculate the similarity matrix using Esum correlation coefficients. G - Calculate the similarity matrix using Gm correlation coefficients.
target_chi=	3.000E+00	Target chi-square per degree of freedom for consistency acceptance.
udcov=	0.0500	User-defined default value of standard deviation in cross-section data to use for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_corr=	1.0000	User-defined default correlation value to use for nuclide-reaction pairs for which covariance data are not available on the selected data file.
udcov_corr_type=	zone	User-defined default correlation in cross-section data to use for nuclide-reaction pairs for which covariance data are not available on the selected data file. (long, zone, short)

Fig. 6.8.3: Input parameters from TSURFER HTML output.

Selecting Messages will reveal a menu of available messages. Selecting Warning Messages from the Messages section of the menu causes the information shown in Fig. 6.8.4 to appear. The Warning Messages edit contains all warning messages that were generated during the execution of the code. If errors were encountered in the code execution, an Error Messages item would have also been available in the menu under Messages.

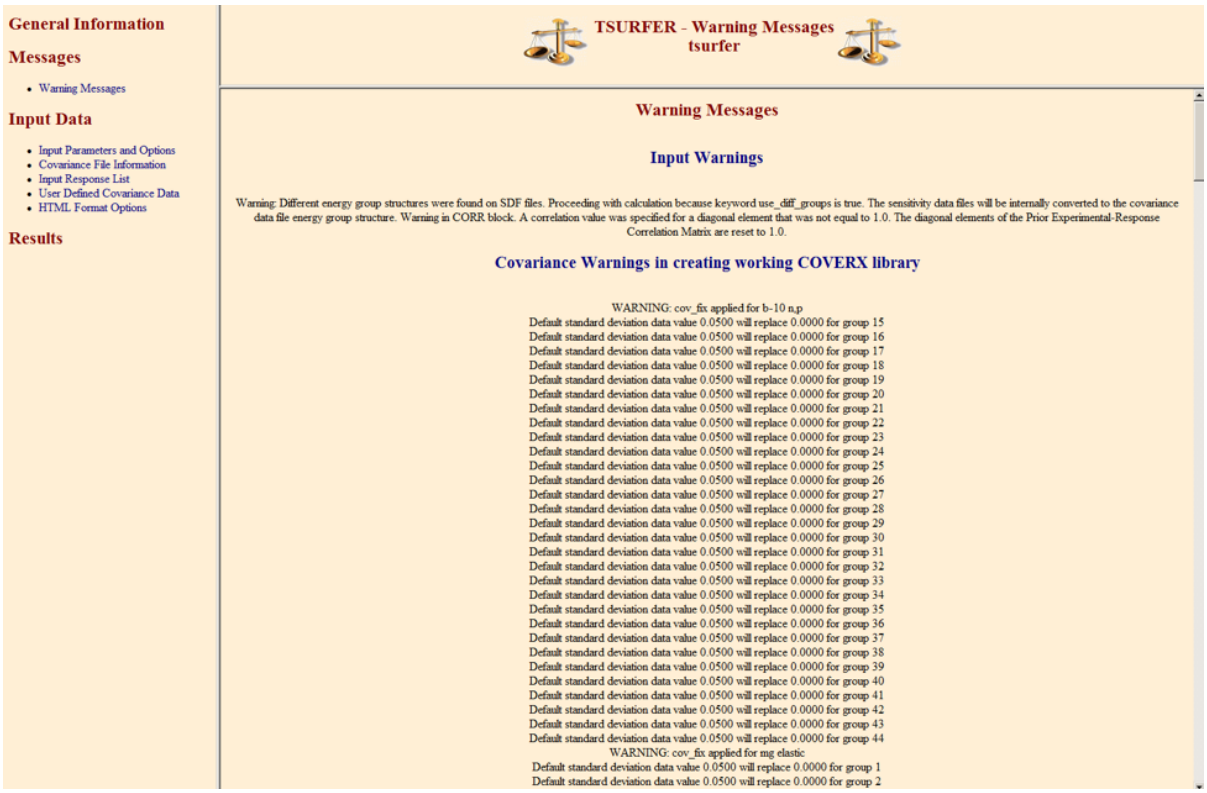
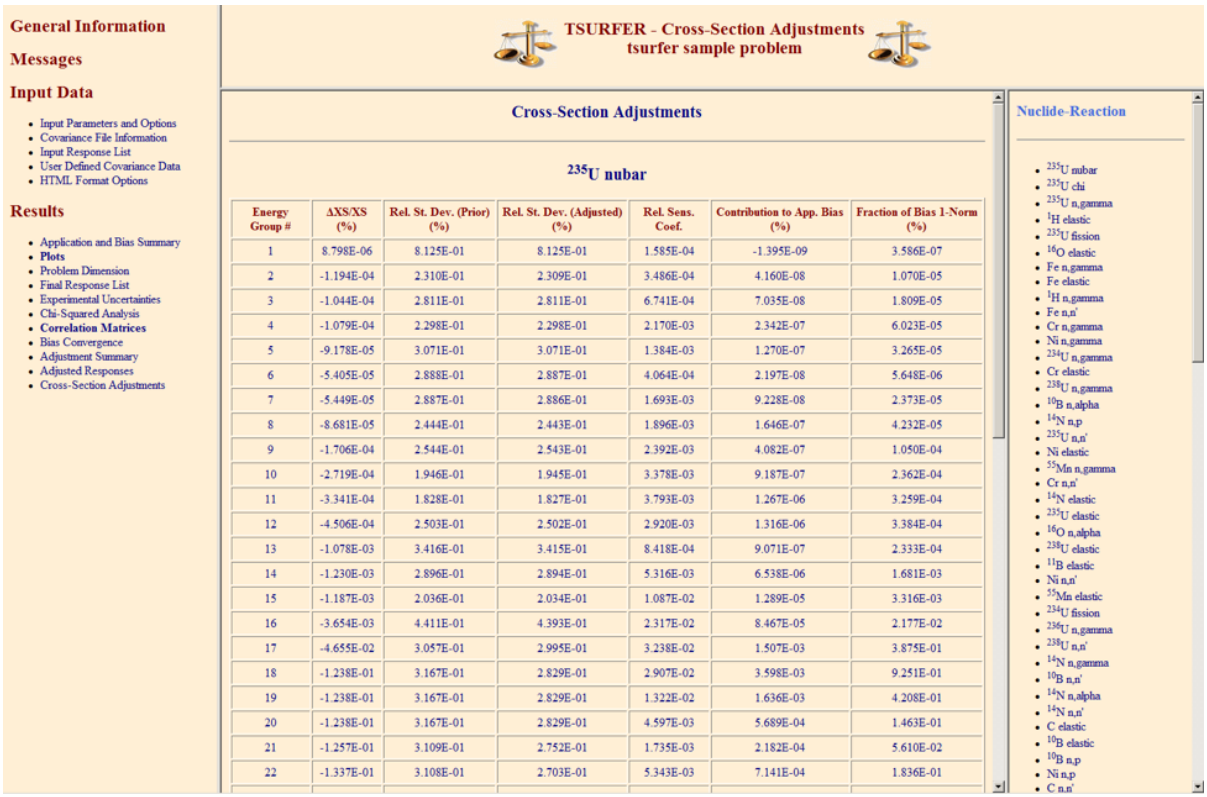


Fig. 6.8.4: Warning messages from TSURFER HTML output.

Selecting Results causes a menu of available results to be revealed. From this menu, selecting Cross-Section Adjustments causes a menu on the right to be revealed. From this menu, nuclide-reaction pairs can be selected to visualize their cross-section adjustments in tabular format. The U-235 nubar adjustments are shown in Fig. 6.8.5.



General Information

Messages

Input Data

- Input Parameters and Options
- Covariance File Information
- Input Response List
- User Defined Covariance Data
- HTML Format Options

Results

- Application and Bias Summary
- **Plots**
- Problem Dimension
- Final Response List
- Experimental Uncertainties
- Chi-Squared Analysis
- **Correlation Matrices**
- Bias Convergence
- Adjustment Summary
- Adjusted Responses
- Cross-Section Adjustments

TSURFER - Cross-Section Adjustments
tsurfer sample problem

Cross-Section Adjustments

^{235}U nubar

Energy Group #	ΔXS/XS (%)	Rel. St. Dev. (Prior) (%)	Rel. St. Dev. (Adjusted) (%)	Rel. Sens. Coef.	Contribution to App. Bias (%)	Fraction of Bias 1-Norm (%)
1	8.798E-06	8.125E-01	8.125E-01	1.585E-04	-1.395E-09	3.586E-07
2	-1.194E-04	2.310E-01	2.309E-01	3.486E-04	4.160E-08	1.070E-05
3	-1.044E-04	2.811E-01	2.811E-01	6.741E-04	7.035E-08	1.809E-05
4	-1.079E-04	2.298E-01	2.298E-01	2.170E-03	2.342E-07	6.023E-05
5	-9.178E-05	3.071E-01	3.071E-01	1.384E-03	1.270E-07	3.265E-05
6	-5.405E-05	2.888E-01	2.887E-01	4.064E-04	2.197E-08	5.648E-06
7	-5.449E-05	2.887E-01	2.886E-01	1.693E-03	9.228E-08	2.373E-05
8	-8.681E-05	2.444E-01	2.443E-01	1.896E-03	1.646E-07	4.232E-05
9	-1.706E-04	2.544E-01	2.543E-01	2.392E-03	4.082E-07	1.050E-04
10	-2.719E-04	1.946E-01	1.945E-01	3.378E-03	9.187E-07	2.362E-04
11	-3.341E-04	1.828E-01	1.827E-01	3.793E-03	1.267E-06	3.259E-04
12	-4.506E-04	2.503E-01	2.502E-01	2.920E-03	1.316E-06	3.384E-04
13	-1.078E-03	3.416E-01	3.415E-01	8.418E-04	9.071E-07	2.333E-04
14	-1.230E-03	2.896E-01	2.894E-01	5.316E-03	6.538E-06	1.681E-03
15	-1.187E-03	2.036E-01	2.034E-01	1.087E-02	1.289E-05	3.316E-03
16	-3.654E-03	4.411E-01	4.393E-01	2.317E-02	8.467E-05	2.177E-02
17	-4.655E-02	3.057E-01	2.995E-01	3.238E-02	1.507E-03	3.875E-01
18	-1.238E-01	3.167E-01	2.829E-01	2.907E-02	3.598E-03	9.251E-01
19	-1.238E-01	3.167E-01	2.829E-01	1.322E-02	1.636E-03	4.208E-01
20	-1.238E-01	3.167E-01	2.829E-01	4.597E-03	5.689E-04	1.463E-01
21	-1.257E-01	3.109E-01	2.752E-01	1.735E-03	2.182E-04	5.610E-02
22	-1.337E-01	3.108E-01	2.703E-01	5.343E-03	7.141E-04	1.836E-01

Nuclide-Reaction

- ^{235}U nubar
- ^{235}U chi
- ^{235}U n,gamma
- ^{1}H elastic
- ^{235}U fission
- ^{16}O elastic
- Fe n,gamma
- Fe elastic
- ^{1}H n,gamma
- Fe n,n'
- Cr n,gamma
- Ni n,gamma
- ^{234}U n,gamma
- Cr elastic
- ^{238}U n,gamma
- ^{10}B n,alpha
- ^{14}N n,p
- ^{235}U n,n'
- Ni elastic
- ^{55}Mn n,gamma
- Cr n,n'
- ^{14}N elastic
- ^{235}U elastic
- ^{16}O n,alpha
- ^{238}U elastic
- ^{11}B elastic
- Ni n,n'
- ^{55}Mn elastic
- ^{234}U fission
- ^{236}U n,gamma
- ^{238}U n,n'
- ^{14}N n,gamma
- ^{10}B n,n'
- ^{14}N n,alpha
- ^{14}N n,n'
- C elastic
- ^{10}B elastic
- ^{10}B n,p
- Ni n,p
- C n,n'

Fig. 6.8.5: Cross-section adjustments from TSURFER HTML output.

Various plots can also be viewed in the TSURFER HTML output. Selecting “Plots” in the Results menu brings up a submenu of various TSURFER plots. The correlation matrices may be viewed by selecting “Correlation Matrices” in the Plots submenu. A Java applet version of Javapeño will appear in the browser window with the correlation matrices preloaded. Data can be added to the plot by double-clicking on the list of available data on the right side of Javapeño. The plot shown in Fig. 6.8.6 was produced with this procedure.

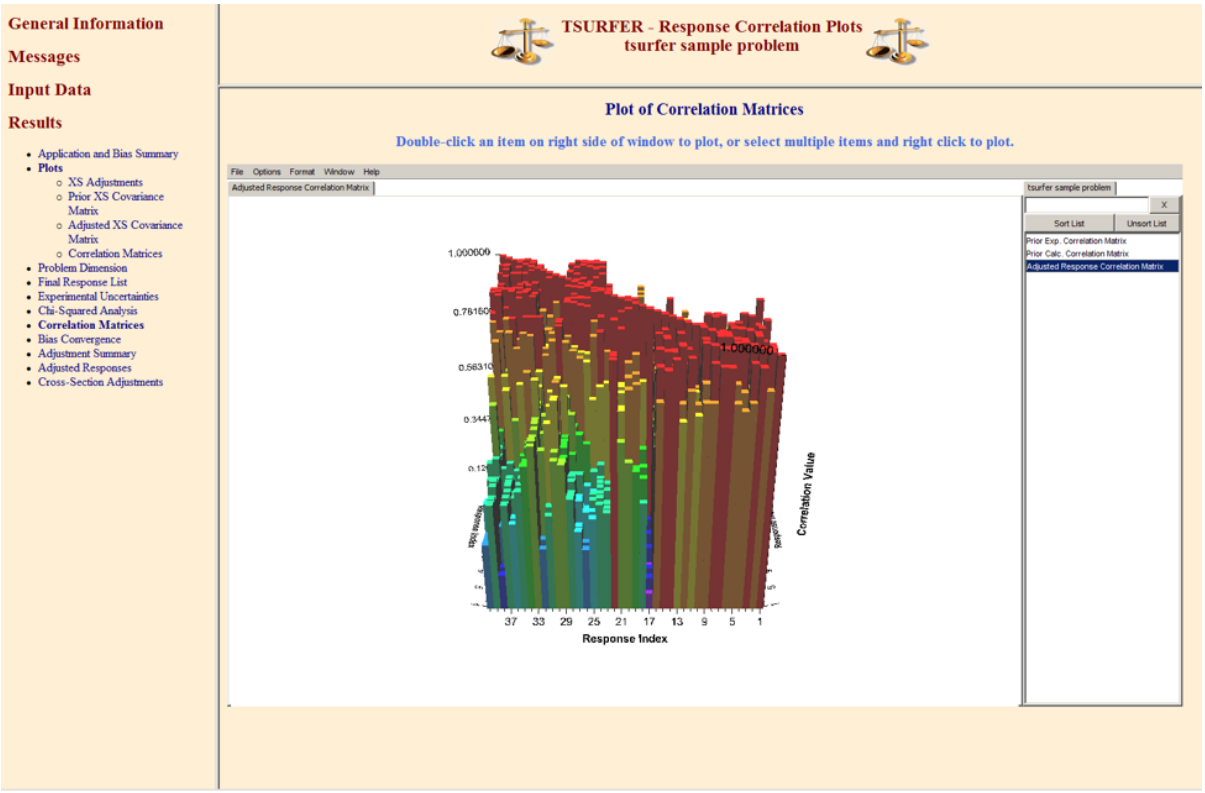


Fig. 6.8.6: Three-dimensional plot of correlation matrix in TSURFER HTML output.

6.8.7 APPENDICES

6.8.7.1 TSURFER Appendix A: Sensitivity/Uncertainty Notation

In the following expressions, the notation E[X] represents the expected value of random variable X, which is equal to the integral of X weighted by its probability density function over the range of allowable values.

Basic variables

I =

number of integral response (experiment and applications) used in GLLS analysis

M =

number of nuclear data parameters used in transport calculations (i.e., number of unique nuclide-reaction pairs multiplied by the number of energy groups)

$\boldsymbol{\alpha}$ =

M dimensional vector of prior nuclear data parameters, where component-i = α_i

A =

M by M diagonal matrix of prior nuclear data parameters, where diagonal element **A**(i,i) = α_i

m =

I dimensional vector of prior measured responses, where component-i = m_i

M =

I by I diagonal matrix of prior measured responses, where diagonal element **M**(i,i) = m_i

$\mathbf{k}(\boldsymbol{\alpha})$ =

I dimensional vector of prior calculated responses obtained with nuclear data $\boldsymbol{\alpha}$, where component I = k_i

K =

I by I diagonal matrix of prior calculated responses, where diagonal element **K**(i,i) = k_i

$\mathbf{F_{m/k}}$ =

I by I diagonal matrix of "E/C" values =

$$\mathbf{MK}^{-1} = \mathbf{K^{-1}M} \tag{6.8.55}$$

where diagonal element

$$F_{m/k}(i,i) = \frac{m_i}{k_i} \tag{6.8.56}$$

$\hat{\mathbf{F}}_{\mathbf{m/k}}$ =

I by I diagonal matrix, where diagonal element

$$F_{m/k}\,(i,i) = \frac{m_i}{k_i} \tag{6.8.57}$$

for a relative-formatted response and $F_{m/k}(i,i) = 1$

$\boldsymbol{\alpha}'$ =

M dimensional vector of adjusted nuclear data parameters produced by GLLS procedure

m' =

I dimensional vector of adjusted measured responses produced by GLLS procedure

$\mathbf{k}'(\boldsymbol{\alpha}')$ =

I dimensional vector of adjusted calculated responses obtained with modified nuclear data $\boldsymbol{\alpha}'$

Note: $\mathbf{k}'(\boldsymbol{\alpha}') = \mathbf{m}'$, due to GLLS adjustment procedure.

$\tilde{\mathbf{d}}$ =

original absolute discrepancy vector = $\mathbf{k} - \mathbf{m}$, where component-i= $k_i - m_i$

d =

original relative discrepancy vector = $\mathbf{K}^{-1}(\mathbf{k} - \mathbf{m})$, where component-i = $(k_i - m_i)\,/k_i$

$\hat{\mathbf{d}}$

original mixed absolute-relative discrepancy vector, where component-i = $(k_i - m_i)/k_i$ for a relative-formatted response and $(k_i - m_i)$ for an absolute-formatted response

$[\Delta\boldsymbol{\alpha}]$ =

M dimensional vector of relative variations in nuclear data = $\mathbf{A}^{-1}(\boldsymbol{\alpha}' - \boldsymbol{\alpha})$ where component-i = $\frac{\alpha_i' - \alpha_i}{\alpha_i}$

$[\Delta\mathbf{m}]$ =

I dimensional vector of relative variations in measured responses = $\mathbf{M}^{-1}(\mathbf{m}' - \mathbf{m})$ where component-i = $\frac{m_i' - m_i}{m_i} \rightarrow \frac{k_i' - m_i}{m_i}$

$[\Delta\mathbf{m}]$ =

I dimensional vector of absolute variations in measured responses = $\mathbf{m}' - \mathbf{m}$ where component-i $m_i' - m_i \rightarrow k_i' - m_i$

$[\Delta\hat{\mathbf{m}}]$ =

I dimensional vector of mixed absolute-relative variations in measured responses, where component-i = $\frac{m_i' - m_i}{m_i}$ for a relative-formatted response and $m_i' - m_i$ for an absolute-formatted response

$[\Delta\mathbf{k}]$ =

I dimensional vector of relative variations in calculated responses = $\mathbf{K}^{-1}(\mathbf{k}' - \mathbf{k})$ where component-i = $\frac{k_i' - k_i}{k_i}$

$[\Delta\mathbf{k}]$ =

I dimensional vector of absolute variations in calculated responses = $\mathbf{k}' - \mathbf{k}$, where component-i = $k_i' - k_i$

$[\Delta\hat{\mathbf{k}}]$ =

I dimensional vector of mixed absolute-relative variations in calculated responses, where component-i = $k_i' - k_i$ for an absolute formatted response

Sensitivity Relations

$\widetilde{\mathbf{S}}_{\mathbf{k}\boldsymbol{\alpha}}$ =

I by M absolute sensitivity matrix; where element $\widetilde{\mathbf{S}}_{\mathbf{k}\boldsymbol{\alpha}}(i, n) = \alpha_n \frac{\partial k_i}{\partial \alpha_n}$

$\mathbf{S}_{\mathbf{k}\boldsymbol{\alpha}}$ =

I by M relative sensitivity matrix = $\mathbf{K}^{-1}\mathbf{S}_{\mathbf{k}\boldsymbol{\alpha}}$, where element $\mathbf{S}_{k\alpha}(i, n) = \frac{\alpha_n}{k_i}\frac{\partial k_i}{\partial \alpha_n}$.

$\hat{\mathbf{S}}_{\mathbf{k}\boldsymbol{\alpha}}$ =

I by M mixed absolute-relative sensitivity matrix, where element $\hat{\mathbf{S}}_{\mathbf{k}\boldsymbol{\alpha}}(i, n) = \frac{\alpha_n}{k_i}\frac{\partial k_i}{\partial \alpha_n}$ if response-i is relative-formatted and $\hat{\mathbf{S}}_{\mathbf{k}\boldsymbol{\alpha}}(i, n) = \alpha_n \frac{\partial \mathbf{k}_i}{\partial \alpha_n}$ if response-i is absolute-formatted

$$\begin{aligned} [\Delta\mathbf{k}] &= \quad \mathbf{S}_{\mathbf{k}\boldsymbol{\alpha}}[\Delta\boldsymbol{\alpha}] \\ [\Delta\mathbf{k}] &= \mathbf{S}_{\mathbf{k}\boldsymbol{\alpha}}[\Delta\boldsymbol{\alpha}] \\ [\Delta\hat{\mathbf{k}}] &= \hat{\mathbf{S}}_{\mathbf{k}\boldsymbol{\alpha}}[\Delta\boldsymbol{\alpha}] \end{aligned} \tag{6.8.58}$$

Absolute covariances

$\tilde{\mathbf{C}}_{\mathbf{mm}}$ =

I by I covariance matrix for prior measured experiment responses where element $\tilde{\mathbf{C}}_{\mathbf{mm}}(\mathrm{i,j}) = E\left(\delta m_i\, \delta m_j\right)$

$\tilde{\mathbf{C}}_{\mathbf{kk}}$ =

I by I covariance matrix for prior calculated responses, where element $\tilde{\mathbf{C}}_{\mathbf{kk}}(\mathrm{i,j}) = E\left(\delta k_i\, \delta k_j\right)$

$\tilde{\mathbf{C}}_{\mathbf{dd}}$ =

I by I covariance matrix for the discrepancies (k-m), where element $\tilde{\mathbf{C}}_{\mathbf{dd}}(\mathrm{i,j}) = E\left(\delta d_i\, \delta d_j\right) = \mathrm{E}\left(\delta\left(\mathrm{k_i} - \mathrm{m_i}\right)\delta\left(\mathrm{k_j} - \mathrm{m_j}\right)\right)$

$\tilde{\mathbf{C}}_{\mathbf{k'k'}}$ =

I by I covariance matrix for adjusted responses, where element $\tilde{\mathbf{C}}_{\mathbf{k'k'}}(\mathrm{i,j}) = E\left(\delta k'_i\, \delta k'_j\right)$

$\boldsymbol{\sigma}_{\mathbf{m}}$ =

I by I diagonal matrix containing standard deviations in prior measured responses, where diagonal element $\widetilde{\sigma}_{\mathrm{m}}\,(\mathrm{ii}) = \sqrt{\widetilde{\mathrm{C}}_{\mathrm{mm}}(\mathrm{i,i})}$

$\boldsymbol{\sigma}_{\mathbf{k}}$ =

I by I diagonal matrix containing standard deviations in prior calculated responses, where diagonal element $\widetilde{\sigma}_{\mathrm{k}}(\mathrm{i,i}) = \sqrt{\widetilde{\mathrm{C}}_{\mathrm{kk}}(\mathrm{i,i})}$

$\boldsymbol{\sigma}_{\mathbf{k'}}$

Relative covariances

$C_{\mathbf{mm}}$ =

I by I relative covariance matrix for prior measured responses, $= \mathbf{M}^{-1}\left[\tilde{\mathbf{C}}_{\mathbf{mm}}\right]\mathbf{M}^{-1}$ $C_{mm}(i,j) = \frac{\mathrm{C_{mm}(i,j)}}{\mathrm{m_i m_j}}$

$C_{\alpha\alpha}$ =

M by M relative covariance matrix for prior nuclear data, where element $\widetilde{C}_{\alpha\alpha}(i,j) = \frac{\mathrm{E}\left(\delta\alpha_\mathrm{i}\delta\alpha_\mathrm{j}\right)}{\alpha_\mathrm{i}\alpha_\mathrm{j}}$

$C_{\mathbf{kk}}$ =

I by I relative covariance matrix for prior calculated responses $= \mathbf{K}^{-1}\,[\mathbf{C}_{\mathrm{kk}}]\,\mathbf{K}^{-1}$ where element $C_{kk}(i,j) = \frac{\mathrm{C_{kk}(i,j)}}{\mathrm{k_i k_j}}$

$C_{\mathbf{dd}}$

I by I relative covariance matrix for response discrepancies; $= \mathbf{K}^{-1}\,[\mathbf{C}_{\mathbf{dd}}]\,\mathbf{K}^{-1}$, where element $C_{dd}(i,j) = \frac{\mathrm{C_{dd}(i,j)}}{\mathrm{k_i k_j}}$

$\boldsymbol{\sigma}_{\mathbf{m}}$ =

I by I diagonal matrix containing relative standard deviations in measured responses, where diagonal element $\sigma_{\mathrm{m}}\,(\mathrm{i,i}) = \sqrt{\mathrm{C_{mm}(i,i)}}$

$\boldsymbol{\sigma}_{\mathbf{k}}$ =

I by I diagonal matrix containing relative standard deviations in calculated responses, where diagonal element $\sigma_{k'}(i,i) = \sqrt{C_{k'k'}(i,i)}$

$\boldsymbol{\sigma_\alpha}$ =

M by M diagonal matrix containing standard deviations in nuclear data, where diagonal element $\boldsymbol{\sigma_\alpha}(i,i) = \sqrt{C_{\alpha\alpha}(i,i)}$

Mixed absolute-relative covariances

If response-i and response-j are both absolute formatted, then

$$\begin{aligned} \hat{C}_{kk}(i,j) &= C_{kk}(i,j) \\ \hat{C}_{mm}(i,j) &= C_{mm}(i,j) \\ \hat{C}_{dd}(i,j) &= C_{dd}(i,j) \\ \hat{C}_{k'k'}(i,j) &= C_{k'k'}(i,j) \end{aligned} \tag{6.8.59}$$

Likewise, if both response-i and response-j are relative-formatted, then

$$\begin{aligned} \hat{C}_{kk}(i,j) = C_{kk}(i,j) &= \frac{C_{kk}(i,j)}{k_i k_j} \\ \hat{C}_{mm}(i,j) = C_{mm}(i,j) &= \frac{C_{mm}(i,j)}{m_i m_j} \\ \hat{C}_{dd}(i,j) = C_{dd}(i,j) &= \frac{C_{dd}(i,j)}{d_i d_j} \\ \hat{C}_{k'k'}(i,j) = C_{kk'}(i,j) &= \frac{C_{kk'}(i,j)}{k'_i k'_j} \end{aligned} \tag{6.8.60}$$

If response-i is absolute-formatted and response-j is relative-formatted, then

$$\begin{aligned} \hat{C}_{kk}(i,j) &= \frac{C_{kk}(i,j)}{k_j} \\ \hat{C}_{mm}(i,j) &= \frac{C_{mm}(i,j)}{m_j} \\ \hat{C}_{dd}(i,j) = \frac{C_{dd}(i,j)}{d_j} \hat{C}_{k'k'}(i,j) &= \frac{C_{k'k'}(i,j)}{k'_j} \end{aligned} \tag{6.8.61}$$

Similar expressions can be derived if response-i is relative-formatted, and response-j is absolute-formatted. The I by I diagonal matrices of standard deviation values are the following:

$$\hat{\sigma}_k(i,i) = \begin{cases} \sigma_k(i,i) & \text{absolute-formatted} \\ \sigma_k(i,i) & \text{relative-formatted} \end{cases} \tag{6.8.62}$$

$$\hat{\sigma}_m(i,i) = \begin{cases} \sigma_m(i,i) & \text{absolute-formatted} \\ \sigma_m(i,i) & \text{relative-formatted} \end{cases} \tag{6.8.63}$$

$$\hat{\sigma}_d(i,i) = \begin{cases} \sigma_d(i,i) & \text{absolute-formatted} \\ \sigma_d(i,i) & \text{relative-formatted} \end{cases} \tag{6.8.64}$$

$$\hat{\sigma}_{k'}(i,i) = \begin{cases} \sigma_{k'}(i,i) & \text{absolute-formatted} \\ \sigma_{k'}(i,i) & \text{relative-formatted} \end{cases} \tag{6.8.65}$$

Correlation matrices

$\mathbf{R_{kk}}$ =

I by I correlation matrix for prior calculated responses, where element $R_{kk}(i,j) = \frac{C_{kk}(i,j)}{\sigma_k(i,i)\sigma_k(j,j)} = \frac{C_{kk}(i,j)}{\sigma_k(i,i)\sigma_k(j,j)} = \frac{\hat{C}_{kk}(i,j)}{\hat{\sigma}_k(i,i)\hat{\sigma}_k(j,j)}$

$\mathbf{R_{mm}}$ =

I by I correlation matrix for prior measured responses, where element $R_{mm}(i,j)$ $\frac{C_{mm}(i,j)}{\sigma_m(i,i)\sigma_m(j,j)}$ = $\frac{C_{mm}(i,j)}{\sigma_m(i,i)\sigma_m(j,j)} = \frac{\hat{C}_{mm}(i,j)}{\hat{\sigma}_m(i,i)\hat{\sigma}_m(j,j)}$

$\mathbf{R_{\alpha\alpha}}$ =

M by M correlation matrix for prior nuclear data, where element $R_{\alpha\alpha}(i,j) = \frac{C_{\alpha\alpha}(i,j)}{\sigma_\alpha(i,i)\sigma_\alpha(j,j)}$

$\mathbf{R_{k'k'}}$ =

I by I correlation matrix for adjusted responses, where element $R_{k'k'}$ (i, j) = $\frac{C_{k'k'}(i,j)}{\sigma_{k'}(i,i)\sigma_{k'}(j,j)}$ = $\frac{C_{k'k'}(i,j)}{\sigma_{k'}(i,i)\sigma_{k'}(j,j)} = \frac{\hat{C}_{k'k'}(i,j)}{\hat{\sigma}_{k'}(i,i)\hat{\sigma}_{k'}(j,j)}$

7. MATERIAL SPECIFICATION AND CROSS SECTION PROCESSING

Introductiom by M. L. Williams and B. T. Rearden

XSProc (Cross Section Processing) provides material input and multigroup (MG) cross section preparation for most SCALE sequences. XSProc allows users to specify problem materials using easily remembered and easily recognizable keywords associated with mixtures, elements, nuclides, and fissile solutions provided in the SCALE **Standard Composition Library**. For MG calculations, XSProc provides cross section temperature correction and resonance self-shielding as well as energy group collapse and spatial homogenization for systems that can be represented in *celldata* input as infinite media, finite 1D/2D systems, or repeating structures of 1D/2D systems, such as uniform arrays of fuel units. Improved resonance self-shielding treatment for nonuniform lattices can be achieved through the use of the **MCDancoff** (Monte Carlo Dancoff) code that generates Dancoff factors for generalized 3D geometries for subsequent use in XSProc. Cross sections are generated on a microscopic and/or macroscopic basis as needed. Although XSProc is most often used as part of an integrated sequence, it can be run without subsequent calculations to generate problem-dependent MG data for use in other tools.

This chapter provides detailed descriptions of the methods and modules used for self-shielding. Self-shielding calculations are effectively a problem-specific extension of the processing procedures used to create the SCALE cross section libraries. SCALE includes continuous energy (CE) and several MG (MG) cross section libraries described in the chapter on SCALE Cross Section Libraries. The AMPX nuclear data processing system [MAT-WWCD15] was used to convert evaluated data from ENDF/B into CE cross sections, which were then averaged into problem-independent MG data at a reference temperature of 300K, weighted with a generic energy spectrum (see the SCALE Cross Section Libraries chapter). After being transformed in probability distributions by AMPX, the CE data require no further modifications for application to a specific problem except for possible interpolation to the required temperatures. However, in MG calculations, reaction rates depend strongly on the problem-specific energy distribution of the flux, which implies that the problem-independent MG data on the library should be modified into problem-dependent values representative of the actual flux spectrum rather than the library generic spectrum. The neutron energy spectrum is especially sensitive to the concentrations and heterogeneous arrangement of resonance absorbers, which may dramatically reduce the flux at the resonance peaks of a nuclide, thus reducing its own reaction rate–a phenomenon known as self-shielding. In general, the higher the concentration of a resonance nuclide and the more the interaction between heterogeneous lumps (e.g. fuel pins), the greater the degree of self-shielding for the nuclide.

Reference [MAT-Wil11] gives a general description of the SCALE self-shielding methods. The individual computational modules perform distinct functions within the overall all self-shielding methodology of XSProc. More theoretical details about individual computational modules are given in Sect. 7.2 through Sect. 7.9. XSProc provides capabilities for two different types of self-shielding methods, which are summarized below.

Bondarenko Method

The Bondarenko approach [MAT-IB64] uses MG cross sections pre-computed over a range of self-shielding conditions, varying from negligibly (infinitely dilute) to highly self-shielded. Based on the following approximations [MAT-StammlerA83] it can be shown that the degree of self-shielding in both homogeneous and heterogeneous systems depends only on a single parameter called the background cross section, "sigma0," and on the Doppler broadening temperature:

(a) neglect of resonance interference effects,

(b) intermediate resonance approximation, and

(c) equivalence theory.

During the SCALE MG library processing with AMPX, self-shielded cross sections are computed using a CE flux calculated at several background cross section values and temperatures. These are used to calculate ratios of the shielded to unshielded cross sections, called "Bondarenko factors" (a.k.a. shielding factors or f-factors). As described in the SCALE Cross Section Libraries chapter, Bondarenko factors are tabulated on the SCALE libraries as a function of sigma0 values and Doppler temperatures for all energy groups of each nuclide.

Bondarenko factors are multiplicative correction factors that convert the generic unshielded data into problem-dependent self-shielded values. The BONAMI computational module performs self-shielding calculations with the Bondarenko method by using the input concentrations and unit cell geometry to calculate a sigma0 value for each nuclide and then interpolating the appropriate MG shielding factors from the tabulated library values.

CENTRM/PMC Method

Self-shielding calculations with BONAMI are fast and are always performed for all SCALE MG sequences. However, due to the approximations (a)–(c) listed in the previous section, a more rigorous method is also provided which can replace the BONAMI results over a specified energy range, usually encompassing the resolved resonance ranges of important absorber nuclides. This approach is designated as the CENTRM/PMC method, named after the two main computational modules, although several additional modules are also used. CENTRM/PMC eliminates the main approximations of the BONAMI approach by performing detailed neutron transport calculations with a combination of MG and CE cross sections for the actual problem-dependent compositions and unit cell descriptions [MAT-WA95]. This provides a problem-dependent pointwise (PW) flux spectrum for averaging MG cross sections, which reflects resonance cross-interference effects, an accurate slowing down treatment, and geometry-specific transport calculations using several available options. Shielded MG cross sections processed with CENTRM/PMC are usually more accurate than BONAMI, so it is the default for most SCALE MG sequences. However, depending on the selected transport option, CENTRM/PMC may run considerably longer than BONAMI alone.

The CENTRM/PMC methodology first executes BONAMI, which provides shielded cross sections outside the specified range of the PW flux calculation. Then the computational module CRAWDAD reads CE cross section files and bound thermal scatter kernels and interpolates the data to the desired temperatures for CENTRM. Using a combination of shielded MG data from BONAMI and CE data from CRAWDAD, CENTRM calculates PW flux spectra by solving the deterministic neutron transport equation for all unit cells described in the input. CENTRM calculations cover the energy interval 10^{-5} eV to 2×10^{7} eV spanned by the SCALE MG libraries. This energy range is subdivided into three sections: (a) upper MG range: E>*demax*, (b) PW range: *demin*<E<*demax*, and (c) lower MG range: E<*demin*, where *demin* and *demax* are the boundaries of the PW range, which can be defined by user input. The default values are *demin*=10^{-3} eV and *demax*=2×10^{4}. The values encompass the resolved resonance ranges of essentially all actinide and fission product nuclides. MG transport calculations are performed in the upper and lower ranges, which are coupled to the PW transport calculation by the scattering sources.

Several methods are available for the CENTRM transport solutions within each energy range, and the default methods can be changed through parameters in the XSProc input. The discrete S_n method is default for homogeneous media and for arbitrary one dimensional (1D) slab, spherical, and cylindrical geometries with general boundary conditions. A unit cell model is used for self-shielding arrays of spherical or cylindrical fuel regions. For the common case of a square-pitch lattice with cylindrical fuel pins, the default transport solver is the 2D method of characteristics (MoC). The CENTRM MoC solution exactly models the outer

rectangular cell surface using a reflected boundary condition. CENTRM also has an option for discrete S_n calculations using a 1D Wigner-Seitz cell with a white outer boundary condition. The 1D cell model is always used for spherical fuel arrays (e.g., pebbles), and can also be selected as a faster alternative than MoC for cylindrical fuel lattices. Finally, a two-region collision probability method can be used for any type of array. The two-region solver executes very fast but is usually more approximate than the MoC and S_n methods.

After CENTRM computes the average PW flux for each material zone, PMC uses the spectra to process the CE cross sections into problem-specific MG values for each material zone. A typical energy grid for the flux solution consists of 50,000–90,000 points, providing good resolution of the spectral fine-structure caused by resonance self-shielding. PMC has several options for processing the MG data, such as correcting for resonance absorption effects on the elastic removal. Shielded cross sections from PMC may also be used to perform an optional MG eigenvalue calculation with the XSDRNPM S_n module for cell-averaging and/or group collapsing of the MG values.

A variation of the standard CENTRM/PMC method is used to perform self-shielding for doubly heterogeneous cells in which cylindrical or spherical fuel elements, composed of small spherical fuel particles dispersed in a moderator material, are distributed in an array configuration. Self-shielding of this type of system requires multiple CENTRM/PMC passes, effectively representing the two levels of heterogeneity [MAT-GW05]. First-level CENTRM calculations are performed for each type of fuel particle using a spherical unit cell to represent the array of multi-layered fuel particles distributed in the moderator matrix. Space-dependent CE fluxes from these calculations are used in the CHOPS module to compute CE disadvantage factors (fuel-average flux divided by cell-average flux) for generating cell-averaged, CE cross sections representative of the homogenized fuel compact. The spatially averaged CE cross sections are used in a second-level CENTRM transport calculation corresponding to a 1D unit cell model for the array of fuel elements, with homogenized number densities for the fuel compact. The CE flux spectrum from this calculation is used in PMC to process the final MG, problem-dependent cross sections. This entire procedure is transparent to the user and has been automated in XSProc. Reference 2 provides more details about the SCALE treatment for doubly heterogeneous fuel.

Treatment of Non-Uniform Lattice Effects

For self-shielding of lattice configurations, both the BONAMI and CENTRM/PMC approaches assume that the fuel is arranged in an infinite, uniform array of identical cells. For most pins in an actual lattice, the uniform-array approximation is satisfactory; however, self-shielding of some cells may be affected by boundary effects along the edge of the array or by the presence of water holes or control rods. These effects can be treated by incorporating a nonuniform Dancoff factor into the self-shielding calculations for the affected cells. The SCALE module MCDancoff performs a simplified one-group Monte Carlo calculation to compute Dancoff factors for arbitrary absorber mixtures within a complex (nonuniform) 3D array. The input for MCDancoff is described in Sect. 7.8. This module must be run as a standalone executable prior to the self-shielding calculations for a given sequence, and the computed Dancoff factors must be entered as XSProc input. The input Dancoff factor is used directly in defining the background cross section for BONAMI calculations. In the CENTRM/PMC methodology, the input Dancoff factor is used in CENTRM to calculate a Dancoff-equivalent unit cell, which defines a uniform lattice pitch that produces the same Dancoff value as the nonuniform lattice. The CENTRM transport calculation then proceeds as usual using 2D MoC or 1D S_n for the unit cell.

7.1 XSPROC: THE MATERIAL AND CROSS SECTION PROCESSING MODULE FOR SCALE

M. L. Williams, L. M. Petrie, R. A. Lefebvre, K. T. Clarno, J. P. Lefebvre, U. Merturyek, D. Wiarda, and B. T. Rearden

ABSTRACT

The material and cross section processing module of SCALE (XSProc) was improved for the 6.2 release to prepare data for continuous-energy and multigroup calculations. XSProc expands material input from Standard Composition Library definitions into atom number densities and, for multigroup calculations, performs cross section resonance self-shielding, energy group collapse, and spatial homogenization. XSProc implements capabilities for problem-dependent temperature interpolation, calculation of Dancoff factors, resonance self-shielding using Bondarenko factors with full-range intermediate resonance treatment, as well as use of continuous energy resonance self-shielding in the resolved resonance region. The SCALE 6.3 XSProc maintains the same architecture with SCALE 6.2 which integrated and enhanced the capabilities previously implemented independently in BONAMI, CENTRM, PMC, WORKER, ICE, and XSDRNPM, along with some additional capabilities that were provided by MIPLIB and SCALELIB. The SCALE 6.3 XSProc development focuses on improving accuracy, applicability and stability for any advanced reactor analysis including Light Water Reactors (LWR), Pebble and Prismatic High Temperature Gas-cooled Reactors (HTGR), Molten Salt Reactors (MSR), Sodium- and Lead-cooled Fast Reactors (SFR and LFR) without any limitation.

ACKNOWLEDGMENTS

Development and maintenance of XSProc and related codes and methods have been sponsored by the US Nuclear Regulatory Commission (NRC) and the US Department of Energy (DOE).

VERSION INFORMATION

XSProc has evolved from the concept of a Material Information Processor library (MIPLIB) that used alphanumeric material specifications, which was initially proposed and developed by R. M. Westfall. J. R. Knight and J. A. Bucholz expanded and refined MIPLIB in early SCALE releases.

SCALE 6.1 (2011) and 6.2 (2016): M. L. Williams and L. M. Petrie led development and improvement for XSProc and contributors include, R. A. Lefebvre, K. T. Clarno, J. P. Lefebvre, U. Merturyek, D. Wiarda, B. T. Rearden, S. Goluoglu, D. F. Hollenbach, N. F. Landers, J. A. Bucholz, C. F. Weber, and C. M. Hopper. With the SCALE modernization initiative beginning in SCALE 6.2, MIPLIB is no longer part of the XSProc analysis, but the original concepts and input formatting were preserved in the new implementation.

SCALE 6.3 (2021): K. S. Kim led responsibility and contributors include A. M. Holcomb, D. Wiarda, R. Bostelmann, M. A. Jessee, and W. A. Wieselquist. The SCALE 6.3 XSProc has been focused on improving accuracy, applicability and stability for advanced reactor analysis including high temperature gas cooled reactors and fast spectrum reactors.

7.1.1 INTRODUCTION

Self-shielding of multigroup cross sections is required in SCALE sequences for criticality safety, reactor physics, radiation shielding, and sensitivity analysis. In all previous versions of SCALE, resonance self-shielding calculations were done by executing a series of stand-alone executable codes, each dedicated to a specific aspect of the self-shielding operations. Each sequence had its own unique internal coding to launch the executable codes. Multigroup (MG) and continuous-energy (CE) cross sections and other data were passed between the individual executable codes by external I/O, which could require a substantial amount of clock time. In the modern version of SCALE, all self-shielding operations are consolidated into a single

driver module named XSProc, and the stand-alone executable codes have been transformed into callable "computational modules" [XSProc-RLL+15]. The functions of XSProc are to (a) read input data, (b) generate in-memory data structures (objects) containing problem-definition information (compositions, cell geometries, computation options), as well as self-shielding information (MG and CE cross sections and fluxes), and (c) execute appropriate computational modules for the requested self-shielding option. Calculated results produced by one module may be stored in the internal data objects and passed to other modules through application program interfaces (APIs). At the completion of XSProc the self-shielded MG cross sections on the data objects can be passed along to transport solvers for continued execution of the control sequence or can be written to an external AMPX library file.

In the future, XSProc will be extended to parallel computations in which self-shielding calculations are done simultaneously for multiple types of unit cells. At the present time, however, XSProc is limited to serial computations; but even in serial mode it typically requires less time than older versions of SCALE to process shielded cross sections, and significant speedups have been observed for heavily I/O bound problems. Integrating the self-shielding capabilities into a single module has a number of additional benefits as well, including maintainability, extensibility, and the ability to easily replace an entire computational module with a future implementation containing new features. Additionally, the size of the problem-dependent MG library generated by XSProc may be greatly reduced compared to previous versions of SCALE because macroscopic cross sections are stored rather than a general-purpose library of microscopic data.

7.1.2 TECHNIQUES

XSProc integrates and enhances the capabilities previously implemented independently in BONAMI, CENTRM, PMC, WORKER, ICE, and XSDRNPM, as well as other capabilities formerly provided by MIPLIB and SCALELIB. It provides capabilities for problem-dependent temperature interpolation of both CE and MG nuclear data, calculation of Dancoff factors, and resonance self-shielding of MG cross sections using several available options. XSProc produces shielded microscopic data for each nuclide or macroscopic data for each material. Additionally, a flux-weighting spectrum can be applied to collapse cross sections to a coarser group structure and/or to integrate over volumes for homogenized cross sections. The flux-weighting spectrum can be input by the user or calculated using one-dimensional (1-D) coupled neutron/gamma transport model. These operations are performed by the sequences CSAS-MG, CSAS1, CSASI, and T-XSEC described in Sect. 7.1.3.2.

7.1.2.1 Overview of XSProc procedures

XSProc reads the COMPOSITION and CELL DATA blocks of the SCALE input, which are described in the following sections. After reading the user input data, XSProc loads the specified MG library to be self-shielded and, depending on the selected self-shielding method, additional CE data files for nuclides appearing in the problem specification. Finally XSProc performs MG self-shielding calculations for all compositions by calling APIs to computational modules such as BONAMI (**BON**darenko **AM**PX **I**nterpolator), CRAWDAD (**C**ode to **R**ead **A**nd **W**rite **DA**ta for **D**iscretized solution), CENTRM (**C**ontinuous **EN**ergy **TR**ansport **M**odule), PMC (**P**roduce **M**ultigroup **C**ross sections), CHOPS (**C**ompute **HO**mogenized **P**ointwise **S**tuff), CAJUN (**C**E **AJ**AX **UN**iter), WAX (**W**orking **AJAX**), XSDRNPM (**X**-**S**ection **D**evelopment for **R**eactor **N**ucleonics with **P**etrie **M**odifications), and/or MIXMACRO to provide a problem-dependent cross section library. Many computational modules have been modernized compared to earlier executable codes distributed in previous versions of SCALE.

Like earlier versions of SCALE, XSProc provides several options for self-shielding an input MG library [XSProc-Wil11]. The first, based on the Bondarenko method [XSProc-IB64], uses the computational module BONAMI. BONAMI is always used to compute self-shielded cross sections for all energy groups. If *parm=bonami* is specified, the shielded cross sections provided by BONAMI are the final values output from

XSProc. However the Bondarenko method has several limitations, especially in the resolved resonance range. Therefore XSProc provides another self-shielding method, with several computation options, which often produces more accurate MG data in the resolved resonance and thermal energy ranges. If *parm=centrm* or *parm=2region* is specified on the sequence line, XSProc calls APIs for the modules CRAWDAD, CENTRM, and PMC to compute CE flux spectra for processing problem-specific, self-shielded cross sections "on the fly [XSProc-WA95]. CENTRM performs MG transport calculations in the fast and lower energy ranges, coupled to pointwise (PW) transport calculations that use CE cross sections in the resonance range. PMC uses the PW flux spectra from CENTRM to compute MG values, which replace the previous values obtained from BONAMI over the specified range of the CE calculation. The original BONAMI shielded cross sections are retained for all other groups.

The CENTRM/PMC approach is the default for criticality and lattice physics calculations, while the BONAMI-only method is default for radiation shielding calculations. The end results of an XSProc calculation are self-shielded macroscopic and/or microscopic MG cross sections stored in memory for subsequent transport calculations; or alternatively a shielded MG AMPX library can be written to an external file and saved for future use.

7.1.2.2 Standard composition material processing

A primary function of the XSProc module is to expand user input in the COMPOSITION block into nuclear number densities (nuclei/b-cm) for every nuclide in each defined mixture. Mixtures can be specified through the direct use of materials presented in the Standard Composition Library, which includes individual nuclides, elements with natural abundances, numerous compounds, alloys and mixtures found in engineering practice, as well several variations of fissile solutions. Additionally, users may define their own materials as atom percent or weight percent combinations. Nuclear masses and theoretical densities are provided in the Standard Composition Library, and methods are available to determine equilibrium states for fissile solutions. Input options for composition data are described in Sect. 7.1.3.3 with several examples provided in Appendix A.

7.1.2.3 Unit cells for MG resonance self-shielding

XSProc utilizes a unit cell description to provide information for resonance self-shielding calculations of the input mixtures. As many unit cells as needed to describe the problem may be specified; however, each mixture (other than 0 for a void mixture) can appear only in one unit cell in the CELLDATA block. If a nuclide appears in more than one mixture, multiple sets of self-shielded cross sections are calculated for the nuclide-one for each mixture in each unit cell. Four types of cells are available for self-shielding calculations: **INFHOMMEDIUM**, **LATTICECELL**, **MULTIREGION**, and **DOUBLEHET**. The default calculation type is CENTRM/PMC for CSAS, TRITON, and TSUNAMI sequences and BONAMI for MAVRIC. All materials not specified in a unit cell are treated as infinite homogeneous media and shielded with BONAMI only, unless the mixture contains a fissionable nuclide, in which case an infinite medium CENTRM/PMC model is used. Note that previous versions of SCALE used infinite medium CENTRM/PMC calculations for all unassigned mixtures. The default type of self-shielding calculation can be overridden, as described in Sect. 7.1.3.2. The following is a brief description of the types of unit cells that can be input in CELLDATA and the computation procedures used.

INFHOMMEDIUM (infinite homogeneous medium) Treatment

The **INFHOMMEDIUM** treatment is best suited for large masses of materials where the size of each material is large compared with the average mean-free path of the material or where the fraction of the material that is a mean-free path from the surface of the material is very small. When **INFHOMMEDIUM** cell is specified, the material in the unit cell is treated as an infinite homogeneous lump. Systems composed of small fuel lumps or resonance nuclides sandwiched between moderating regions should not be treated as infinite homogeneous media. In these cases a MULTIREGION or LATTICECELL geometry should be used.

LATTICECELL Treatment

The **LATTICECELL** model is appropriate for arrays of resonance absorber mixtures-with or without clad-arranged in a square or a triangular pitch configuration within a moderator. Annular fuel (e.g., with an internal moderator in the center) can also be addressed. Input data for the **LATTICECELL** treatment are described in Sect. 7.1.3.5. Self-shielded cross sections are generated for each material zone in a unit cell of the lattice. If a nuclide appears in more than one zone, self-shielded cross sections are produced for each zone where the nuclide is present. Limitations of the **LATTICECELL** treatment are listed below.

1. The cell description is limited to unit cells for arrays of spherical, plate (slab), or cylindrical fuel bodies. In the case of cylindrical pins in a square-pitch lattice, the default (*parm=centrm*) self-shielding calculation uses the CENTRM method of characteristics (MoC) option to represent the 2D rectangular unit cell with reflected boundary conditions. By default, self-shielding for all other arrays uses a CENTRM 1D S_N calculation for the unit cell (spherical and cylindrical geometries use Wigner-Seitz cells). If *parm=bonami* is specified, heterogeneous self-shielding effects are treated by equivalence theory [XSProc-Wil11] The computation option *parm=2region*, described in Sect. 7.1.3.1, can also be used for self-shielding lattice cells.

2. Only predefined choices of cell configurations are available. The available options are described in detail in Sect. 7.1.3.5.

3. The basic treatment for **LATTICECELL** assumes an infinite, uniform array of unit cells. This assumption is a good approximation for interior fuel regions within a large, uniform array. The approximation becomes less rigorous for fuel regions on the periphery of the array or adjacent to a nonuniformity (e.g., control rod, water hole, etc.) in the lattice. For some cases it may be desirable to address this issue by specifying a different lattice cell for this type of fuel pin and using a modified procedure to define an effective unit cell, as described below.

**** LATTICECELL treatment for nonuniform arrays*.

Nonuniform lattice effects may be treated in CENTRM calculation by specifying the keyword **DAN2PITCH=***dancoff* in the optional CENTRM DATA (see Sect. 7.1.3.9). In this approach, the SCALE standalone code MCDancoff must be run prior to the self-shielding calculation in order to compute Dancoff factors for the fuel regions of interest in the nonuniform lattice configuration. MCDancoff performs a simplified one-group Monte Carlo calculation to compute Dancoff factors for complex geometries (see Sect. 7.8). The Dancoff value for the fuel region of interest is assigned to the DAN2PITCH keyword in the input for the corresponding cell. Using an iterative procedure, CENTRM computes the pitch of a uniform lattice that has the same Dancoff value as the nonuniform lattice.

MULTIREGION Treatment

The **MULTIREGION** treatment is appropriate for 1-D geometric regions where the geometry effects may be important, but the limited number of zones and boundary conditions in the **LATTICECELL** treatment are not applicable. The **MULTIREGION** unit cell allows more flexibility in the placement of the mixtures but requires all regions of the cell to have the same geometric shape (i.e., slab, cylinder, sphere, buckled slab, or buckled cylinder). Lattice arrangements can be approximated by specifying a non-vacuum boundary condition on the outer boundary. See Sect. 7.1.3.6 for more details. Limitations of the **MULTIREGION** cell treatment are listed below.

1. A **MULTIREGION** cell is limited to a 1-D approximation of the system being represented. An exact 1D model can be defined for the following multizone geometries with vacuum boundary conditions: spheres, infinitely long cylinders, and slabs; and for an infinite array of slabs with reflected or periodic boundaries.
2. The shape of the outer boundary of the **MULTIREGION** cell is the same as the shape of the inner regions. Cells with curved outer surfaces cannot be stacked physically to create arrays; however, arrays can be approximated by a Wigner-Sietz cell with a white outer boundary condition, where the outer radius is defined to preserve the area of the true rectangular or hexagonal unit cell.
3. Boundary conditions available in a **MULTIREGION** problem include vacuum (eliminated at the boundary), reflected (reflected about the normal to the surface at the point of impact), periodic (a particle exiting the surface effectively enters an identical cell having the same orientation and continues traveling in the same direction), and white (isotropic return about the point of impact). Reflected and periodic boundary conditions on a slab can represent real physical situations but are not valid on a curved outer surface. A single, non-interacting cell has a vacuum outer boundary condition. If the cell outer boundary condition is not a vacuum boundary, the unit cell approximates some type of array.
4. When using the CENTRM/PMC self-shielding method, the MULTIREGION cell model must include fissionable material. This can be accomplished by adding a trace amount of a fissionable material to one or more mixtures, or by modeling a region of homogenized fuel and water, or by adding a thin (e.g., 1e-6 cm-thick) layer containing at least a trace of a fissionable nuclide on the periphery of the problem.

DOUBLEHET Treatment

DOUBLEHET cells use a specialized CENTRM/PMC calculational approach to treat resonance self-shielding in "doubly heterogeneous" systems. The fuel for these systems typically consists of small, heterogeneous, spherical fuel particles (grains) embedded in a moderator matrix to form the fuel compact. The fuel-grain/matrix compact constitutes the first level of heterogeneity. Cylindrical(rod). spherical (pebble), or slab fuel elements composed of the compact material are arranged in a moderating medium to form a regular or irregular lattice, producing the second level of heterogeneity. The fuel elements are also referred to as "macro cells." Advanced reactor fuel designs that use TRISO (tri-material, isotopic) or fully ceramic microencapsulated (FCM) fuel require the **DOUBLEHET** treatment to account for both levels of heterogeneities in the self-shielding calculations. Simply ignoring the double-heterogeneity by volume-weighting the fuel grains and matrix material into a homogenized compact mixture can result in a large reactivity bias.

In the **DOUBLEHET** cell input, keywords and the geometry description for grains are similar to those of the **MULTIREGION** treatment, while keywords and the geometry for the fuel element (macro-cell) are similar to those of the **LATTICECELL** treatment. The following rules apply to the **DOUBLEHET** cell treatment and must be followed. Violation of any rules may cause a fatal error.

1. As many grain types as needed may be specified for each unique fuel element. Note that grain type is different from the number of grains of a certain type. For example, a fuel element that contains both UO_2 and PuO_2 grains has two grain types. The same fuel element may contain 10000 UO_2 grains and 5000 PuO_2 grains. In this case, the number of grains of type UO_2 is 10000, and the number of grains of type PuO_2 is 5000.

2. As many fuel elements as needed may be specified, each requiring its own **DOUBLEHET** cell. This may be the case for systems with many fuel elements at different fuel enrichments, burnable poisons, etc. Each fuel element may have one or more grain types.

3. Since the grains are homogenized into a new mixture to be used in the fuel element (macro-cell) cell calculation, a unique fuel mixture number must be entered. XSProc creates a new material with the new mixture number designated by the keyword f*uelmix*=, containing all the nuclides that are homogenized. The user must assign the new mixture number in the transport solver geometry (e.g., KENO) input unless a cell-weighted mixture is created.

4. The type of lattice or array configuration for the fuel-element may be spheres on a triangular pitch (**SPHTRIANGP**), spheres on a square pitch (**SPHSQUAREP**), annular spheres on a triangular pitch (**ASPHTRIANGP)**, annular spheres on a square pitch (**ASPHSQUAREP)**, cylindrical rods on a triangular pitch (**TRIANGPITCH**), cylindrical rods on a square pitch (**SQUAREPITCH**),annular cylinderical rods on a triangular pitch (**ATRIANGPITCH)**, annular cylindrical rods on a square pitch (**ASQUAREPITCH)**, a symmetric slab (**SYMMSLABCELL)**, or an asymmetric slab (**ASYMSLABCELL)**.

5. If there is only one grain type for a fuel element, the user must enter either the pitch, the aggregate number of particles in the element, or the volume fraction for the grains. The code needs the pitch and will directly use it if entered. If pitch is not given, then the volume fraction (if given) is used to calculate the pitch. If neither the pitch nor the volume fraction is given, then the number of particles is used to calculate the pitch and the volume fraction. The user should only enter one of these items.

   If the fuel matrix contains more than one grain type, all types are homogenized into a single mixture for the compact. As for the one grain type case, the pitch is needed for the spherical cell calculations. However, the pitch by itself is not sufficient to perform the homogenization. Since each grain's volume is known (grain dimensions must always be entered), entering the number of particles for each grain type essentially provides the total volume of each grain type and therefore enables the calculation of the volume fraction and the pitch. Likewise, entering the volume fraction for each grain type essentially provides the total volume of each grain type and therefore enables the calculation of the number of particles and the pitch. Therefore, one of these two quantities must be entered for multiple grain types. In these cases, since pitch is not given, the available matrix material is distributed around the grains of each grain type proportional to the grain volume and is used to calculate the corresponding pitch. Over-specification is allowed as long as the values are not inconsistent to greater than 0.01%.

6. For cylindrical rods and for slabs, fuel height must also be specified. For slabs the slab width must also be specified.

7. The CENTRM calculation option must be S_n.

7.1.2.4 Cell weighting of MG cross sections

Cell-weighted self-shielded cross sections are created when **CELLMIX**= is specified in a **LATTICECELL** or **MULTIREGION** cell input. In this case, after finishing the self-shielding calculations for all mixtures in the cell, XSProc calls the computational module XSDRNPM, which solves the 1-D MG transport equation to obtain k_∞ and space-dependent MG fluxes for the cell. The resultant fluxes are used to compute MG flux disadvantage factors for processing cell-weighted cross sections of all nuclides in the cell. When the cell-weighted cross sections are used with *homogenized* number densities of the cell nuclides, the reaction rates of the homogenized mixture preserve the spatially averaged reactions rates of the heterogeneous configuration. The user must input a new mixture ID to identify the homogenized mixture associated with the cell-weighted cross sections. **This homogenized mixture should not be used in the heterogeneous geometry data for other transport codes such as KENO, NEWT, etc.** Instead, the cell-homogenized mixture that is created should be used at the location of the original cell. Also, cell weighted homogenized cross sections should not be used in MG sensitivity data calculations performed using the TSUNAMI sequences.

7.1.3 XSPROC INPUT DATA GUIDE

XSProc input data are entered in free form, allowing alphanumeric data, floating-point data, and integer data to be entered in an unstructured manner. Up to 252 characters per line are allowed. Data can usually start or end in any column. Each data entry must be followed by one or more blanks to terminate the data entry. For numeric data, either a comma or a blank can be used to terminate each data entry. Integers may be entered for floating values. For example, 10 will be interpreted as 10.0 if a floating point value is required. Imbedded blanks are not allowed within a data entry unless an E precedes a single blank as in an unsigned exponent in a floating-point number. For example, 1.0E 4 would be correctly interpreted as 1.0×10^4. A number with a negative exponent must include an "E". For example 1.0-4 cannot be used for 1.0E-4.

The word "END" is a special data item. An END may have a name or label associated with it. The name or label associated with an END is separated from the END by a single blank and is a maximum of 12 characters long. *At least two blanks or a new line MUST follow every labeled and unlabeled END. WARNING: It is the user's responsibility to ensure compliance with this restriction. Failure to observe this restriction can result in the use of incorrect or incomplete data without the benefit of warning or error messages.*

Multiple entries of the same data value can be achieved by specifying the number of times the data value is to be entered, followed by either R, *, or $, followed by the data value to be repeated. Imbedded blanks are not allowed between the number of repeats and the repeat flag. For example, 5R12, 5*12, 5$12, or 5R 12, etc., will enter five successive 12's in the input data. Multiple zeros can be specified as nZ where n is the number of zeroes to be entered.

7.1.3.1 XSProc data checking and resonance processing options

To check the XSProc input data, run CSAS-MG and specify PARM=CHECK or PARM=CHK after the sequence specification as shown below.

```
=CSAS-MG PARM=CHK
```

In this case the actual XSProc cross section processing calculations are not performed. The input data are checked, the problem description is printed, appropriate error and warning messages are printed, and a table of additional data is printed.

Resonance processing will automatically be performed by the default method for the sequence selected. The default methods are CENTRM/PMC for CSAS, TRITON, and TSUNAMI sequences and BONAMI for the MAVRIC sequences. Alternatively, a resonance processing procedure may be chosen by entering

PARM=*option*, where *option* CENTRM selects the recommended CENTRM/PMC transport method for each cell type, *option* 2REGION selects the CENTRM/PMC two-region calculation, and *option* BONAMI applies full range Bondarenko factors to all energy groups without utilizing CENTRM/PMC. For example, to run CSAS1X sequence using only BONAMI for self-shielding, rather than the default CENTRM/PMC method, enter the computational sequence specification shown below.

```
=CSAS1X PARM=BONAMI
```

Multiple PARM options are specified by enclosing parameters in parenthesis, such as

```
=CSAS1X PARM=(CHK, BONAMI)
```

XSProc resonance self-shielding options are summarized below.

PARM=BONAMI.

This is the fastest MG processing method. It performs resonance self-shielding for all energy groups using the Bondarenko method. BONAMI computes the appropriate background cross section of a given unit cell and then interpolates the corresponding shielding factor from Bondarenko factors on the MG library. Dancoff factors needed to evaluate the background cross section for lattices are computed internally, but these can be overridden by input values in the MORE DATA block. More details on this method are given in the BONAMI section of the manual.

PARM=CENTRM.

This executes the CENTRM/PMC modules to process shielded MG cross sections using CE flux spectra calculated with the recommended type of CE transport solver for the designated type of cell. The CENTRM-recommended CE transport solvers are (a) infinite homogeneous medium calculation for INFHOMMEDIUM cells; (b) 2D MoC transport calculation for a LATTICECELL consisting of cylindrical fuel pins in a square lattice; and (c) 1-D discrete S_n transport for all other LATTICECELLs and for all MULTIREGION cells. The recommended type of transport solver can be overridden for individual cells, as well as for selected energy ranges, by using the CENTRM DATA block described in Sect. 7.1.3.9.

PARM=2REGION.

The CENTRM two-region (2R) option computes the PW flux using a simplified collision probability method for an absorber (e.g., fuel) region surrounded by an external moderator region which has an asymptotic energy spectrum. To account for the heterogeneous effects of a lattice, a correction known as the Dancoff factor is applied to the escape probabilities in the 2R calculation (see the CENTRM chapter of the SCALE manual). These Dancoff factors are calculated internally by XSProc for a uniform array of mixtures in slab, spherical, or cylindrical geometries. These mixture-dependent Dancoff factors can be modified by user input using the DAN parameters contained in the MORE DATA block, as defined in Sect. 7.1.3.8.

Note on CENTRM/PMC self-shielding options:

The energy range of the CENTRM flux calculation is subdivided into three sections: fast, PW, and low energy. PMC only computes self-shielded cross sections for groups within the PW range defined by parameters *demax* and *demin*, which, respectively, define the upper and lower energies of the CENTRM PW flux calculation. Problem-dependent cross sections for groups in the fast and low energy ranges are obtained with the more approximate BONAMI method. Default values for parameters *demax* and *demin* are defined appropriately for self-shielding of important resonance materials in thermal reactor systems. The PW self-shielding range can be extended or decreased for individual cells by modifying these parameters using CENTRM DATA.

7.1.3.2 XSProc input data

The types of input data required for XSProc are given in Table 7.1.1, and individual entries are explained in the text following the table. The title, cross section library name (either CE or MG), and standard composition specification data (**READ COMP** input block) are required for all sequences that use XSProc. The name of the cross section library is used to determine if the transport solver is executed using CE or MG data (e.g., CE or MG KENO calculations). The unit cell descriptions (**READ CELL** input block) are only used for MG self-shielding calculations. If the specified sequence executes in CE mode, the cell data input can be omitted, or it will be skipped if present. If the cell data information is omitted for MG calculations, all mixtures are self-shielded using the infinite medium approximation.

There are seven standard SCALE sequences that run just XSProc, and produce a MG cross section library or libraries.

=XSPROC produces three libraries with an optional fourth library.

- **sysin.microLib** is a self-shielded library of the individual nuclides in the problem for use in a later transport calculation,
- **sysin.macroLib** is a self-shielded library of the mixture cross sections in the problem for use in a later transport calculation,
- **sysin.smallMicroLib** is a self-shielded library of specific reaction rate cross sections and the elastic and total inelastic scattering transfer matrices for later use in calculating reaction rates and sensitivity values, and
- **sysin.xsdrnWeightedLib** is an optional library produced if the input specifies having **XSDRN** do a weighting calculation. This can be a cell weighted and/or a group collapse calculation. The library can be either individual nuclides or mixtures, depending on input.

=CSAS-MG produces an **ft04f001** library that is equivalent to the **sysin.microLib**. With appropriate input it can also produce an **ft03f001** which is equivalent to **sysin.xsdrnWeightedLib** above.

=CSASI or **=CSASIX** produce an **ft04f001** library that is equivalent to **sysin.microLib**, and an **ft02f001** library that is equivalent to **sysin.macroLib**. CSASIX will run an **XSDRN** on the first cell without any MOREDATA input. With appropriate input they both can produce an **ft03f001** that is the equivalent of **sysin.xsdrnWeightedLib**.

=CSAS1 or **=CSAS1X** produce an **ft04f001** library that is the equivalent of **sysin.microLib**. Both sequences will run an **XSDRN** on the first cell. With appropriate input, they both can produce an **ft03f001** that is the equivalent of **sysin.xsdrnWeightLib**.

=T-XSEC produces an **ft04f001** library that is equivalent to **sysin.macroLib**. and an **ft44f001** library that is equivalent to sysin.microLib.

The reactions (MT numbers) written to each library are listed in the `SequenceNeutronMT.txt` file located in the etc directory installed with SCALE.

Table 7.1.1: Outline of XSProc input data

Data Position	Type of Data	Data Entry	Comments
1	Title	Enter title	Limit to 80 characters
2	Cross section library name		The currently available libraries are listed in the table *Standard SCALE cross-section libraries* of the XSLib chapter.
3	Standard composition specification data	Enter the appropriate data	Begin this data block with **READ COMP** and terminate with **END COMP**. See Section **7.1.3.2**
4	Unit cell(s) description for MG calculations only		Begin this data block with **READ CELL (or CELLDATA)**
	a. Type of self shielding calculation	**INFHOMMEDIUM** **LATTICECELL** **MULTIREGION** **DOUBLEHET**	These are the available options. See the explanation in Section Error! Reference source not found..
	b. Unit cell geometry specification	Enter the appropriate data	See Section **7.1.3.4** for **INFHOMMEDIUM**. See Section **7.1.3.5** for **LATTICECELL**. See Section **7.1.3.6** for **MULTIREGION**. See Section **7.1.3.7** for **DOUBLEHET.**
	c. Optional **MORE** parameter data	Enter the desired data	Begin this data block with **MORE DATA** (or **MOREDATA**) and terminate with **END MORE** (or END **MOREDATA**). Use only if MORE parameter data are to be entered; otherwise, omit these data entirely. See Section **7.1.3.8**
	d. Optional **CENTRM** parameter data	Enter the desired data	Begin this data block with **CENTRM DATA** (or **CENTRMDATA**) and terminate with **END CENTRM** (or END **CENTRMDATA**). Use only if CENTRM parameter data are to be entered; otherwise, omit these data entirely.
	e. End of unit cell data		Terminate with **END CELL (or END CELLDATA)**

Repeat positions 4a–4d as needed to specify all unit cells. Position 4 data are applicable to the MG calculations only.

1. TITLE. An 80-character maximum title is required. The title is the first 80 characters of the XSPROC data.
2. CROSS SECTION LIBRARY NAME. This item specifies the cross section library that is to be used in the calculation. See Table *Standard SCALE cross-section libraries* in the XSLIB chapter of the SCALE manual for a discussion of the available libraries.
3. The keywords **READ COMP** followed by the standard compositions specifications. These data are used to define mixtures used in the problem. See Sect. 7.1.3.3 and Table 7.1.2 for a description of the standard composition specification data. These data are required for every problem. After all mixtures have been entered, the keywords **END COMP** must be entered.
4. The keywords **READ CELLDATA** followed by the input describing each unit cell as defined below. After all unit cells are described, the keywords **END CELLDATA** terminate this input block.

a. TYPE OF CALCULATION. The options are **INFHOMMEDIUM**, **LATTICECELL**, **MULTIREGION**, **DOUBLEHET**, or nothing. A description of these cell types and the associated computational methods are provided in Sect. 7.1.2.3. If all input mixtures are to be treated as infinite homogeneous media, the **CELLDATA** block can be omitted. In this case the self-shielding calculations will not account for any geometrical effects, so users should be careful in applying this approach. Similarly, mixtures not explicitly assigned to a cell are treated as infinite homogeneous media in the manner discussed in Sect. 7.1.2.3.

b. CELL GEOMETRY SPECIFICATION. See Sect. 7.1.3.4 and Table 7.1.7 for an explanation of the optional unit cell data associated with an **INFHOMMEDIUM** problem. See Sect. 7.1.3.5 for an explanation of the data associated with **LATTICECELL** problems. Sect. 7.1.3.6 explains the data required for a **MULTIREGION** problem. Sect. 7.1.3.7 explains the required data for a **DOUBLEHET** problem. The **DOUBLEHET** input may be thought of as a combination of **MULTIREGION** input for the fuel grains and **LATTICECELL** input for the fuel element.

c. OPTIONAL MORE PARAMETER DATA. This option allows certain defaulted parameters to be re-specified by the user. This block begins with **MORE DATA** and is used by XSDRN. These data apply only to the unit cell immediately preceding them. Data placed prior to all unit cell data apply to all materials not listed in any unit cell and are treated as infinite homogeneous media. Omit these data unless they are needed. This block ends with **END MORE**. See Sect. 7.1.3.8.

d. OPTIONAL CENTRM PARAMETER DATA. This optional data block begins with **CENTRM DATA** and ends with **END CENTRM**. These data allow the user to override default data for CENTRM and PMC. These data apply only to the unit cell immediately preceding them. Data placed prior to all unit cell data apply to all materials not listed in any unit cell and are treated as infinite homogeneous media.

7.1.3.3 Standard composition specification data

Mixtures utilized in the problem are defined using standard composition specification data. The standard composition input begins with the keywords **READ COMP**, followed by standard composition specifications for all mixtures in the problem. When all mixtures have been described, enter the words **END COMP** to signal the completion of this block of data. XSProc computes macroscopic cross sections for all mixtures defined in the **COMP** block.

The required input for the standard composition specification data varies, depending on the type of standard composition material. However, every standard composition specification must include the following:

1. a standard composition material name.

2. a mixture number (MX) that contains this material, and
3. a terminator for the standard composition specification data (enter the word END).

The types of standard compositions in SCALE are (a) basic mixtures, (b) fissile solutions, (c) chemical compounds, and (d) alloys. The four general options for inputting these types of data are shown in Table 7.1.2. For some cases, more than one option could possibly be used to specify the mixture. The user may select whichever options are most convenient to define a particular mixture, and these may be entered in any order.

Table 7.1.2: Outline of standard composition specification options (Mixtures can be defined using one or more of these options in any order).

Input data name	Comments
READ COMP	Enter once for a problem. Enter the words **READ COMP** prior to entering any standard composition data.
sc	This option is used for defining basic mixtures. Enter one of the alphanumeric identifiers, symbols, or names from Standard Composition Library tables *Isotopes in standard composition library*, *Isotopes and their natural abundances*, *Elements and special nuclide symbols*, *Compounds*, or *Alloys and mixtures* in place of SC. This indicates the isotope, nuclide, compound, or alloy that will make up this standard composition. See **Table 6.1.3.3.0.2** for additional required and optional data for each standard composition.
SOLUTION	This option is used to specify a fissile solution mixture. See **Table 6.1.3.3.0.3** for additional required and optional data for each solution. End the data with an **END**.
ATOM	This option creates a chemical compound mixture composed of the specified nuclide in the compound. Each nuclide is entered followed by the relative number of atoms of the nuclide in the compound. All compounds must begin with the four letters **ATOM** followed by up to eight additional alphanumeric characters. See **Table 6.1.3.3.0.4** for additional required and optional data for each compound.
WTPT	This option creates a mixture/alloy composed of the specified nuclides in the mixture/alloy. Each nuclide is entered followed by the weight percent of the nuclide in the mixture/alloy. All mixture/alloys must begin with the four letters **WTPT** followed by up to eight additional alphanumeric characters. See **Table 6.1.3.3.0.5** for additional required and optional data for each arbitrary physical mixture or alloy.
END COMP	Enter once for a problem. Enter the exact words **END COMP** when all the standard composition components have been described. At least two blanks or a new line must follow the words **END COMP** prior to continuing data entry.

Names of the standard composition materials (the alphanumeric identifiers) appearing in the **COMP** block input must be selected from the tables of elements, compounds, solutions, and alloys given in the SCALE manual section describing the Standard Composition Library. An error message will be printed if the user enters an invalid standard composition material name or if any isotopes in the compound do not exist in the specified library

Input data to define each of the standard composition types in Table 7.1.2 are summarized in Table 7.1.3 through Table 7.1.6. Optional input is indicated by curly brackets { }. *Since some of the input is not keyword based, the order of entries is important in the standard composition specification*. The temperature specification is used for Doppler broadening and/or determination of the proper thermal scattering data. Input material densities are not modified for temperature effects. Additional description of the standard composition input for each type of material is given following all the tables. As in the tables, input parameters enclosed by

curly brackets { } indicate that these are optional.

continues on next page

Table 7.1.3 – continued from previous page

Table 7.1.3: Standard composition specification for basic mixtures.

Entry no.	Data name	Data type	Entry requirement	Comment
1	**sc**	Standard composition name	Always	Enter one of the alphanumeric identifiers, symbols, or names from Tables *Isotopes in standard composition library*, *Isotopes and their natural abundances*, *Elements and special nuclide symbols*, *Compounds*, or *Alloys and mixtures* in the STDCMP chapter in place of **sc**. This indicates the isotope, nuclide, compound, or alloy that will make up this standard composition. This entry is required.
2	*mx*	Mixture ID number	Always	Skip at least one blank after SC prior to entering the mixture number. This must be an integer greater than zero. This entry is required.
3	**DEN**=*roth*	Density	Optional	If the density of a basic standard composition (*roth*) is to be entered, add **DEN** = *roth*, where *roth* is the density in g/cm^3, following the mixture ID number with at least one space between *mx* and the keyword **DEN**.
4	{**VF**=}*vf*	Density multiplier	See comment	Default value is 1. Enter the density multiplier (density fraction, volume fraction, or a combination). **VF** = 0 indicates that syntax 2 is to be used and that the next number entered will be ADEN. In this case entries 7-8 are omitted.
5	*aden*	Number density	**VF** = 0	Atom density of the nuclide in atoms / barn-cm. This can only be entered for elements and isotopes. See chapter STDCMP tables *Isotopes in standard composition library*, *Isotopes and their natural abundances*, and *Elements and special nuclide symbols*.
6	*temp*	Temperature (K)	See comment	Default value is 300 K. This entry may be omitted if the default temperature is acceptable and *iza* / *wpt* data are not entered.
7	*iza*	Isotope's ZA number	**VF** ≠ 0	Enter for each isotope in a multiple isotope nuclide. Omit if **VF** = 0 or the default values are acceptable. Entries 7 and 8 are entered in pairs until each isotope in the nuclide is defined. See STDCMP chapter tables *Elements and special nuclide symbols*, *Compounds*, and *Alloys and mixtures* for isotope IDs contained in nuclides, compounds, and alloys.
8	*wtp*	weight percent of isotope	**VF** ≠ 0	Enter the weight percent of the isotope in the nuclide. Omit if **VF** = 0 or the default values are acceptable. For each multiple isotope nuclide, the weight percents of the isotopes must sum to 100. Entries 7 and 8 are entered in pairs until each isotope in the nuclide is defined.
9	**END**	Terminate the standard composition	Always	This terminates the data for a standard composition. Enter the keyword **END** to terminate standard composition. A tag, up to 12 characters long, may follow the keyword **END** preceded by a single blank. At least two blanks or an end of line must separate this entry from the next entry.

continues on next page

Table 7.1.3 – continued from previous page

continues on next page

Table 7.1.4 – continued from previous page

Table 7.1.4: Standard composition specification for solutions.

Entry no.	Data name	Data type	Entry requirement	Comment
1	**SOLUTION**	keyword	Always	Identifies a new fissile solution. See the Standard Composition Library documentation.
2	{**MIX**=} *mx*	mixture number	Always	If the mixture number is not keyworded, it must immediately follow the SOLUTION keyword.
*3a	**RHO[***name***]**= *rho*	metal density	Optional	The density of the metal in *name* in grams/liter
*3b	**MOLAR[***name***]**= *molar*	molarity	Optional	moles of *name* per liter of solution
*3c	**MASSFRAC[***name***]**= *massfrac*	relative density	Optional	grams of metal in *name*/gram of solution
*3d	**MOLEFRAC[***name***]**= *molefrac*	fractional molarity	Optional	moles of *name* per mole of solution
*3e	**MOLALITY[***name***]**= *molality*	molality	Optional	moles of *name* per kilogram of water
4	**DENSITY**= *density*	solution density	Optional	density of the solution in grams/milliliter
5	**TEMPERATURE**= *temperature*	temperature	Optional	temperature of the solution in Kelvin
6	**VOL_FRAC**= *vf*	density multiplier	Optional	an overall, after the fact multiplier of the solution density
7	**END SOLUTION**	keyword	Always	ends the solution input

* Nuclides that occur in an item 3 will, by default, have naturally occurring isotopics. If this is not appropriate, the desired isotope and weight percent of each isotope making up the nuclide can be input in pairs following the value associated with the specified item 3.

continues on next page

Table 7.1.4 – continued from previous page

continues on next page

Table 7.1.5 – continued from previous page

Table 7.1.5: Standard composition specification for chemical compounds.

Entry no.	Data name	Data type	Entry requirement	Comment
1	**ATOM**	Arbitrary compound name	Always	Enter the four characters ATOM followed by up to 12 additional alphanumeric characters in place of ATOM. As many compounds as required may be entered, but each must have a unique name. This entry is required.
2	*mx*	Mixture ID number	Always	Skip at least one blank after the compound name prior to entering the mixture number. This must be an integer greater than zero. This entry is required.
3	*roth*	Density	Always	This is the mixture density in g/cm^3.
4	*nel*	Number of *ncza* entries	Always	This is the number of elements or nuclides that make up the compound.
5	*ncza*	Nuclide ID number	Always	Repeat entries 6 and 7 for each element in the arbitrary compound before entering *temp*. Enter the ID number from the far right column of tables *Isotopes in Standard composition library* or *Isotopes and their natural abundances*. (Premixed standard compositions cannot be used in a chemical compound definition.)
6	*atpm*	Atoms per molecule	Always	Enter the number of atoms of this element per molecule of compound following each *ncza*. Repeat entries 6 and 7 for each element in the arbitrary compound before entering *temp*.
7	*vf*	Density multiplier	Always	Enter the density multiplier (density fraction, volume fraction, or a combination). A value of 1 means the material density is *roth*.
8	*temp*	Temperature (K)	See comment	Default value is 300 K. This entry may be omitted if the default temperature is acceptable and *iza* / *wpt* data are not entered.
9	*iza*	Isotope's ZA number	Optional	Enter for each isotope in a multiple isotope nuclide. Entries 9 and 10 are entered in pairs until each isotope in the nuclide is defined. See table *Isotopes and their natural abundances* for multiple isotope nuclide IDs and table *Elements and special nuclide symbols* in the Standard Composition Library for a list of isotopes in a multiple isotope nuclide.
10	*wtp*	weight percent of isotope	Optional	Enter the weight percent of the isotope in the nuclide. For each nuclide the weight percents must sum to 100. Entries 9 and 10 are entered in pairs until each isotope in the nuclide is defined.
11	**END**	Terminate the standard composition	Always	This terminates the data for an compound. Enter the keyword **END** to terminate compound. A tag, up to 12 characters long, may follow the keyword **END** preceded by a single blank. At least two blanks or a new line must separate this entry from the next entry.

continues on next page

Table 7.1.5 – continued from previous page

continues on next page

Table 7.1.6 – continued from previous page

Table 7.1.6: Input specification for user-specified mixture/alloy data.

Entry no.	Data name	Data type	Entry requirement	Comment
1	**WTPT**	Mixture/alloy name	Always	Enter the four characters **WTPT** followed by up to 12 additional alphanumeric characters in place of WTPT. As many physical mixtures or alloys as required may be entered but each must have a unique name. This entry is required.
2	*mx*	Mixture ID number	Always	Skip at least one blank after the alloy name prior to entering the mixture number. This must be an integer greater than zero. This entry is required.
3	*roth*	Density	Always	This is the mixture density in g/cm^3.
4	*nel*	Number of *ncza* entries	Always	This is the number of elements or nuclides that make up the mixture/alloy.
5	*ncza*	Nuclide or element ID number	Always	Repeat the sequences 6 and 7 for each element in the mixture/alloy before entering *temp*. Enter the ID number from the far right column of table *Isotopes in Standard composition library* or *Isotopes and their natural abundances* in the Standard Composition Library section. (Premixed standard compositions cannot be used in an arbitrary mixture/alloy definition.)
6	*wpct*	Weight percent of nuclide or element	Always	Enter the weight percent of this element in the mixture/alloy following each *ncza*. Weight percents must sum to 100. Repeat the sequence 6 and 7 for each element in the mixture before entering *temp*.
7	*vf*	Density multiplier	Always	Enter the density multiplier (density fraction, volume fraction, or a combination). A value of 1 means the material density is *roth*.
8	*temp*	Temperature (K)	See comment	Default value is 300 K. This entry may be omitted if the default temperature is acceptable and *iza* / *wpt* data is not entered.
9	*iza*	Isotope's ZA number	Optional	Enter for each isotope in a multiple isotope nuclide. Entries 9 and 10 are entered in pairs until each isotope in the nuclide is defined. See the Standard Composition Library tables *Isotopes and their natural abundances* in for multiple isotope nuclide IDs and *Elements and special nuclide symbols* for a list of isotopes in a multiple isotope nuclide.
10	*wtp*	Weight percent of isotope	Optional	Enter the weight percent of the isotope in the nuclide. For each nuclide the weight percents must sum to 100 for each isotope in a multiple isotope nuclide. Entries 9 and 10 are entered in pairs until each isotope in the nuclide is defined.
11	**END**	Terminate the standard composition	Always	This terminates the data for an arbitrary mixture. Enter the keyword **END** to terminate arbitrary mixture/alloy. Repeat entries 1 through 11 until all the mixtures have been defined. A tag, up to 12 characters long, may follow the keyword **END** preceded by a single blank. At least two blanks or a new line must separate this entry from the next entry.

continues on next page

STANDARD COMPOSITION INPUT FOR BASIC MIXTURES (see Table 7.1.3).

Two input syntaxes are available for standard composition specifications of basic mixtures in the **COMP** block. The first uses information (e.g., densities, atomic weights, physical constants, etc.) contained in the Standard Composition Library, along with user specified input, to automatically compute the number densities for mixture components. In the second option, the user computes the nuclide number densities, and inputs these directly for each component of the mixture. XSProc recognizes syntax 2 if the third entry of the composition specification is zero, as shown below. It is allowable to use syntax 1 for some standard composition specifications and syntax 2 for others. The two syntaxes to define basic mixtures with the standard composition specifications are shown below.

syntax 1: Standard Composition Library data used to compute number densities.

sc *mx* **DEN**=*roth* {**VF**=}*vf temp* iza_1 wtp_1 ... iza_N wtp_N **END**

syntax 2: User input number densities

sc *mx* 0 *aden temp* **END**

The definitions for these input parameters are given below.

A1. **sc**

STANDARD COMPOSITION MATERIAL NAME. This corresponds to one of the material names given in the Standard Composition Library for isotopes, elements, thermal moderators and activity materials, chemical compounds, and alloys/mixtures. Some types of these materials require entering certain data such as the volume fraction or theoretical density and other engineering-type data. For standard compositions containing more than one isotope of an element (such as UO_2), the user is free to specify the weight percent for each isotope, such that they total 100%. See the Basic standard composition specifications section for examples of basic standard compositions.

A2. *mx*

MIXTURE ID NUMBER. An arbitrary mixture number is required on every standard composition specification for both syntaxes. It defines the mixture that contains the material defined by the standard composition specification data. The mixture numbers are utilized in the CELLDATA block Cell Block for INFHOMMEDIUM, LATTICECELL, MULTIREGION, or DOUBLEHET problems and the geometry data.

A3. **DEN**=*roth*

MIXTURE DENSITY. The keyword **DEN** is assigned a value of *roth,* where *roth* is the specified density of the mixture component in g/cm^3. It should always be entered for materials that contain enriched multi-isotopic nuclides. The effective density of the material component is equal to the product of *roth* and *vf.* An example of this is demonstrated in Appendix A.

A4. {**VF**=}*vf*

VOLUME FRACTION. The keyword **VF** is assigned a value of *vf.* It is also allowable to omit the keyword **VF=** and just enter the value *vf* . The default value of the volume fraction is 1.0. The volume fraction can be interpreted as

a. the volume fraction of this standard composition component in the mixture,

b. the density of the standard composition component in this application divided by the theoretical or default density given in the Standard Composition Library, or

c. the product of (a) and (b).

Appendix A discusses the interaction between *roth* and *vf*. For example, assume a homogenized mixture representing the water moderator and Zircaloy cladding around a fuel pin is to be described. If the volume of the clad is 5.32 cc and the volume of the water moderator is 44.68 cc, the mixture can be described using H_2O with a volume fraction of 0.8936 [i.e., 44.68/(44.68+5.32)] and ZIRCALOY with a volume fraction of 0.1064 [i.e., 5.32/(44.68+5.32)].

A5. *aden*

NUMBER DENSITY (not used for syntax 1, but required for 2). The number density is entered ONLY if 3rd entry on the standard composition specification is entered as zero. The number density is entered in units of atoms per barn-cm.

A6. *temp*

TEMPERATURE. The default value of the temperature is 300 K. The temperature can be omitted if entries A7 and A8 are also omitted.

A7. *iza*

ISOTOPE ZA NUMBER. Enter a value for each isotope in the standard composition component, entry 1. Do not enter a value if the volume fraction, **VF**, is zero (A4 above).

The ZA number of the isotope is entered if the user wishes to specify the isotopic distribution. This is done by entering *iza* and *wtp* for each isotope until all the desired isotopes have been described. In most cases the "ZA" ID number is (A+1000*Z), where A is the atomic mass or weight of the isotope, and Z is the atomic number. For example, the ZA number for ^{235}U is 92235.

Entries A7 and A8 can be skipped if the default values listed in Table 7.1.2 are acceptable.

A8. *wtp*

WEIGHT PERCENT OF THE ISOTOPE. If entry A7 is entered, a value must also be entered for A8. The weight percent of the isotope is the percent of this isotope in the element. The weight percent of all specified isotopes of the element must sum to 100 (± 0.01).

A9. **END**

The word **END** is entered to terminate the input data for a standard composition component. This **END** can have a label associated with it that can be as long as 12 characters. The label is optional, and if entered must be preceded from the **END** by a single blank. At least two blanks or a new line must separate this item from the next data entry.

STANDARD COMPOSITION INPUT FOR FISSILE SOLUTIONS (see Table 7.1.4).

Syntax:

```
SOLUTION {MIX=}mx RHO[fuelsalt]=fd (iza_i  wtp_i) MOLAR[acid]=aml
     MASSFRAC[name]=mfrac  MOLEFRAC[name] =molfrac
     MOLALITY[name]=molal  DENSITY=roth
     TEMPERATURE=temp VOL_FRAC=vf
END SOLUTION
```

where

mx is the mixture number,

fuelsalt is the Standard Composition Library component name of one of the fissile compounds

fd is the fuel density in grams of uranium or plutonium per liter of solution

acid is one of the Standard Composition Library acid compounds (e.g., HNO3 or HFACID)

name is one of the Standard Composition Library solution components

aml is the acid molarity of the *acid* component (moles of *acid/* liter of solution)

mfrac is the mass fraction of *name* in the solution (grams of metal in *name*/gram solution)

molfrac is the mole fraction of *name* in the solution (moles of *name*/mole solution)

molal is the mass fraction of *name* in the solution (moles of *name*/kg water)

roth is the density of the solution,

vf is the density multiplier (ratio of actual to theoretical density of the solution),

temp is the temperature in Kelvin,

iza is the isotope ID number from table *Available fissile solution components*, and

wtp is the weight percent of the isotope in the material.

Below are the input data for fissile solutions.

1. **SOLUTION**

   Keyword starting a solution specification. Solutions require the specification of the mixture and at least one component. Current possible components are given in the Standard Composition Library table, *Available fissile solution components*. Only the mixture number and one component are required. Appendix A contains examples of the input data for solutions.

2. *mx*

   MIXTURE ID NUMBER. A mixture number is required on every standard composition specification. It defines the mixture that contains the material defined by the standard composition specification data. The mixture numbers are utilized in the Unit Cell Specification for INFHOMMEDIUM, LATTICECELL, or MULTIREGION.

3. **RHO**[*fuelsalt*]=*fd*
MOLAR[*acid*]=*aml*
MASSFRAC[*name*]=*mfrac*
MOLEFRAC[*name*]=*molfrac*
MOLALITY[*name*]=*molal*

KEYWORD PARAMETERS TO DEFINE CONCENTRATIONS OF SOLUTION COMPONENTS. Each keyword specifies the unit, the component name from the Standard Composition Library and the component value, as shown Table 7.1.4. Up to three components can be specified for a solution if one is an acid. After the value, the isotopic enrichments of the nuclides can be given as pairs of isotope IDs and weight percent. **NOTE: the square brackets [ ] containing the component name are required.**

4. **DENSITY=***roth*

   Keyword specifying the overall solution density as grams per cubic centimeter or as a "?", meaning it is to be solved for. Solving for the density is the default behavior, but the density can be given, and a component value can be solved for instead.

5. **TEMPERATURE=***temp*

   Keyword defining temperature of the solution. The default value is 300 K.

6. **VOLFRAC=***vf*

   Keyword defining volume fraction - the default volume fraction is 1.0. This value must be greater than 0.0. The volume fraction can be interpreted as: a. the volume fraction of this solution specification in the mixture, b. the density of the solution in this application divided by the calculated density of the solution, or c. the product of (a) and (b).

7. **END SOLUTION**

STANDARD COMPOSITION INPUT FOR CHEMICAL COMPOUNDS (see Table 7.1.5)

Syntax:

ATOMnn *mx roth nel* $ncza_1$ $atpm_1$ … $ncza_{\text{nel}}$ $atpm_{\text{nel}}$ {*vf* {*temp* {iza_1 wtp_1 …} } } **END**

Below are the data for user-defined chemical compounds.

C1. **ATOMnn**

COMPOUND NAME. User-specified compounds (also defined as "arbitrary" in older versions of SCALE) require the user to provide all the information normally found in the Standard Composition Library. This option allows specifying a compound not available in the Standard Composition Library by utilizing nuclides and elements available in the library. An user-specified compound name must start with the four characters "**ATOM**." A maximum of twelve characters is allowed for the compound name, and imbedded blanks are not allowed.

C2. *mx*

MIXTURE ID NUMBER. A mixture number is required on every standard composition specification. It defines the mixture that contains the material defined by the compound specification data. The mixture numbers are utilized in the Unit Cell Specification for **INFHOMMEDIUM**, **LATTICECELL**, or **MULTIREGION** problems and the KENO V.a or KENO-VI geometry data.

C3. *roth*

MIXTURE DENSITY. The density of the arbitrary material is entered in units of g/cm^3. *roth* and *vf* interact to produce the density of the mixture used in the problem. Note that this is a required entry and does not use "**DEN**=" keyword.

C4. *nel*

NUMBER OF ELEMENTS IN THE MATERIAL. Enter the number of components from the Standard Composition Library that are to be used to define this material.

C5. *ncza*

ID NUMBER. This is the "ZA" ID number for the element or isotope. Usually, *ncza*=A+1000*Z, where A is the atomic mass or weight of the nuclide, and Z is the atomic number.

C6. *atpm*

ATOMIC or ELEMENT ABUNDANCE. Enter the number of atoms of this element per molecule of compound. Repeat the sequence *ncza* and *atpm* (C5 and C6) for every element in the compound before going to entry C7.

C7. *vf*

VOLUME FRACTION. The default value of the volume fraction is 1.0. This value must be greater than 0.0. The volume fraction can be interpreted as

a. the volume fraction of this compound in the mixture,

b. the density of the compound in this application divided by the input density of the compound, or

c. the product of (a) and (b).

C8. *temp*

TEMPERATURE. The default value of the temperature is 300 K. The temperature can be omitted if entries C9 and C10 are also omitted.

C9. *iza*

ISOTOPE ZA NUMBER. Enter a value for each isotope in the element in the compound. The ZA number of the isotope is entered if the user wishes to specify the isotopic distribution. This is done by entering *iza* and *wtp* for each isotope until all the desired isotopes have been described. In most cases the "ZA" ID number is (A+1000*Z), where A is the atomic mass or weight of the isotope, and Z is the atomic number.

Entries C9 and C10 can be skipped if the default values listed in Table 7.1.2 of Sect. 7.1 are acceptable.

C10. *wtp* WEIGHT PERCENT OF THE ISOTOPE. If entry C9 is entered, a value must also be entered for C10. The weight percent of the isotope is the percent of this isotope in the element. The weight percents of all specified isotopes of the element must sum to 100 (± 0.01).

Repeat the sequence *iza wtp* until the sum of the *wtp*s sum to 100. The sequence *iza wtp* is repeated until all of the desired isotopes have been specified.

C11. **END**

The word **END** is entered to terminate the input data for compound. This **END** can have a label associated with it that can be as long as 12 characters. The label is optional, and if entered must be preceded from the **END** by a single blank. At least two blanks or a new line must separate item C11 from the next data entry.

STANDARD COMPOSITION INPUT FOR MIXTURES AND ALLOYS (see Table 7.1.6)

Syntax:

WTPTnn *mx roth nel* $ncza_1$ $wpct_1$ ... $ncza_{\text{nel}}$ $wpct_{\text{nel}}$ {*vf* {*temp* {iza_1 wtp_1 ... } }} **END**

Below are the input data for arbitrary (i.e., user-defined) physical mixture or alloy.

D1. **WTPTnn**

ARBITRARY MIXTURE/ALLOY NAME. The arbitrary user-specified mixture/alloy option allows specifying a mixture or an alloy not available in the Standard Composition Library by utilizing the nuclides and elements available in the library. An arbitrary mixture/alloy name must start with the four characters "**WTPT**." A maximum of 12 characters is allowed for the arbitrary mixture/alloy name. Imbedded blanks are not allowed in an arbitrary mixture/alloy name. Appendix A contains input data for arbitrary mixture/alloys.

D2. *mx*

MIXTURE ID NUMBER. A mixture number is required on every standard composition specification. It defines the mixture that contains the material defined by the arbitrary compound specification data. The mixture numbers are utilized in the Unit Cell Specification for **INFHOMMEDIUM**, **LATTICECELL**, **MULTIREGION**, or **DOUBLEHET** problems and the KENO V.a or KENO-VI geometry data.

D3. *roth*

MIXTURE DENSITY. The density of the arbitrary material is entered in units of g/cm^3. *roth* and *vf* interact to produce the density of the mixture used in the problem. Note that this is a required entry and does not use "**DEN**=" keyword.

D4. *nel*

NUMBER OF ELEMENTS IN THE MATERIAL. Enter the number of components from the Standard Composition Library that are to be used to define this arbitrary material.

D5. *ncza*

ID NUMBER. This is the "ZA" ID number for the element or isotope. Usually, *ncza*=A+1000*Z, where A is the atomic mass or weight of the nuclide, and Z is the atomic number.

D6. *wpct*

ATOMIC or ELEMENT ABUNDANCE. Enter the weight percent of this element in the arbitrary alloy. The sum of all the weight percents for each specified element in the arbitrary alloy MUST be 100.0. Repeat the sequence *ncza* and *wpct* (D5 and D6) for every element in the arbitrary mixture/alloy before going to entry D7.

D7. *vf*

VOLUME FRACTION. The default value of the volume fraction is 1.0. This value must be greater than 0.0. The volume fraction can be interpreted as:

a. the volume fraction of this mixture or alloy in the mixture,

b. the density of the mixture or alloy in this application divided by the input density (*roth*) of the mixture or alloy, or

c. the product of (a) and (b).

D8. *temp*

TEMPERATURE. The default value of the temperature is 300 K. The temperature can be omitted if entries D9 and D10 are also omitted.

D9. *iza*

ISOTOPE ZA NUMBER. Enter a value for each isotope in the element in the arbitrary alloy. The ZA number of the isotope is entered if the user wishes to specify the isotopic distribution. This is done by entering *iza* and *wtp* for each isotope until all the desired isotopes have been described. In most cases the "ZA" ID number is (A+1000*Z), where A is the atomic mass or weight of the isotope, and Z is the atomic number.

Entries D9 and D10 can be skipped if the default values listed in Table 7.1.2 are acceptable.

D10. *wtp*

WEIGHT PERCENT OF THE ISOTOPE. If entry D9 is entered, a value must also be entered for D10. The weight percent of the isotope is the percent of this isotope in the element. Weight percents of all specified isotopes of the element must sum to 100 (±0.01).

D11. **END**

The word **END** is entered to terminate the input data for an arbitrary compound. This **END** can have a label associated with it that can be as long as 12 characters. The label is optional and if entered must be preceded from the **END** by a single blank. At least two blanks or a new line must separate this item from the next data entry.

7.1.3.4 Unit cell specification for infinite homogeneous problems

This section describes the unit cell data that can be entered for an **INFHOMMEDIUM** problem. Additional information is available in Appendix B.

Syntax:

INFHOMMEDIUM *mx* {**CELLMIX**{=}*mix*} **END**

The data required to specify the unit cell for an **INFHOMMEDIUM** unit cell are given in Table 7.1.7. The individual entries are explained in the following text.

1. **celltype**

   INFHOMMEDIUM. The keyword **INFHOMMEDIUM** is entered to indicate this unit cell contains one mixture with no geometry corrections. This data must be entered. The keyword may be truncated to any number of characters as long as the characters present are identical from the beginning of the keyword (i.e., INF is acceptable). All mixtures not in a defined unit cell are by default processed as infhommedium.

2. *mx*

   MIXTURE NUMBER. The mixture number defines the mixture to be used in the cell. This data must be entered. Be sure the mixture number entered is defined in the standard composition data.

3. **CELLMIX**=*mix*

CELL-WEIGHTED MIXTURE NUMBER. (the = sign can be replaced by a space if desired). Enter ONLY if a cell-weighted mixture is to be generated. Enter a unique mixture number to be used by XSDRN to create the cell-weighted mixture (Sect. 7.1.2.4). For **INFHOMMEDIUM** cells, cross sections for the cell mixture are equal to the shielded values of the original mixture.

4. **END**

   The word **END** is entered to terminate the **INFHOMMEDIUM** data. An optional label can be associated with this **END**. The label can be as many as 12 characters long and is separated from the **END** by a single blank. At least two blanks must follow this entry.

Table 7.1.7: Unit cell specifications for INFHOMMEDIUM problems.

Entry no.	Input data	Data type	Comments
1	**INFHOMMEDIUM**	Keyword	Keyword to begin infhommedium unit cell. Enter the keyword INFHOMMEDIUM. This word may be truncated to any number of letters as long as they exactly replicate the beginning part of the keyword (e.g., INF is acceptable).
2	*mx*	Cell mixture number	Specifies the mixture number to be used in the cell.
3	**CELLMIX**= *mix*	Keyword + new mixture number	Enter the keyword **CELLMIX**= followed immediately by a unique positive integer (*mix*). The integer will be a new mixture number that has the neutronic properties of the self-shielded unit cell.[a]
4	**END**		Terminate **INFHOMMEDIUM** data

7.1.3.5 Unit cell specification for LATTICECELL problems

This section describes the unit cell input data for a **LATTICECELL** problem. The **LATTICECELL** description is especially suited to self-shield arrays of repeated cells such as a fuel assembly lattice. The unit cell specification plays a major role in providing accurate problem-dependent cross sections using the computational procedures described in Sect. 7.1.2.3. Unit cells are limited to (a) infinitely long cylindrical rods in a square or triangular lattice, (b) spheres in a cubic or triangular lattice, (c) a symmetric array of slabs, or (d) an asymmetric array of slabs. Both "regular" and "annular" fuel geometries can be used in **LATTICECELL** problems. "Regular" cells allow a concentric spherical, cylindrical, or symmetric slab configuration, where the central region is fuel, surrounded by an optional gap, an optional clad, and an external moderator. "Annular" cells also allow concentric spherical, cylindrical, or asymmetric slab configurations, but the central region corresponds to an inner moderator region which is surrounded by a fuel region having an optional gap and optional clad on each side of the fuel. An inner gap may be specified inside the fuel region, and an outer gap may be specified outside the fuel region. Similarly an inner clad may be specified inside the fuel region, and an outer clad may be specified outside the fuel region. For both regular and annular fuel cells, the outer boundary of the unit cell is determined from the square or triangular pitch of the array.

Regular cells are **SQUAREPITCH**, **TRIANGPITCH**, **SPHSQUAREP**, **SPHTRIANGP**, and **SYMMSLABCELL**.

Annular cells are **ASQUAREPITCH** (or **ASQP**), **ATRIANGPITCH** (or **ATRP**), **ASPHSQUAREP** (or **ASSP**), **ASPHTRIANGP** (or **ASTP**), and **ASYMSLABCELL**

Syntax:

celltype ctp PITCH (or **HPITCH**) *pitch mm* **FUELD (or FUELR**) *fuel mf*

GAPD (or GAPR) *gap mg* **CLADD (or CLADR**) *clad mc*

IMODD (or IMODR) *imod mim* **IGAPD** (**or IGAPR**) *igap mig*

ICLADD (or **ICLADR**) *iclad mic* {**CELLMIX**=*mix*} **END**

The unit cell geometry data required to specify a LATTICECELL problem are given in Table 7.1.8. The individual entries are explained in the text below.

1. **celltype**

   LATTICECELL. The keyword **LATTICECELL** is entered to indicate this unit cell contains mixtures that are positioned in a regular array. This data must be entered. The keyword may be truncated to any number of characters as long as the characters present are identical from the beginning of the keyword (e.g., **LAT** is acceptable). This unit cell is normally used for regular arrays of materials such as fuel pins in an assembly.

2. **ctp**

   TYPE OF LATTICE. This defines the type of lattice or array configuration. Any one of the following alphanumeric descriptions may be used. Note that the alphanumeric description must be separated from subsequent data entries by one or more blanks. Fig. 7.1.1 Mixture and position data are entered using keywords. Mixture number 0 may be entered for void and may be used multiple times in each and all unit cells. For regular cells, the minimum requirement is that a fuel region and a moderator region are specified and no other inner components are specified. For annular cells, the minimum requirement is the fuel and outer moderator and inner moderator regions are specified. Regular and annular cell configurations are specified as shown below.

   Regular Cells

   SQUAREPITCH is used for an array of cylinders arranged in a square lattice, as shown in Fig. 7.1.1. The clad and/or gap can be omitted.

   TRIANGPITCH is used for an array of cylinders arranged in a triangular-pitch lattice as shown in Fig. 7.1.2. The clad and/or gap can be omitted.

   SPHSQUAREP is used for an array of spheres arranged in a square-pitch lattice. A cross section view through a cell is represented by Fig. 7.1.1. The clad and/or gap can be omitted.

   SPHTRIANGP is used for an array of spheres arranged in a triangular-pitch (dodecahedral) lattice. A cross section view through a cell is represented by Fig. 7.1.2. The clad and/or gap can be omitted.

   SYMMSLABCELL is used for an infinite array of symmetric slab cells, as shown in Fig. 7.1.3. The clad and/or gap can be omitted.

   Annular Cells

   ASQUAREPITCH or **ASQP** is used for annular cylindrical rods in a square-pitch lattice as shown in Fig. 7.1.4. The inner and outer clad and gap are independently entered so they must be different materials and dimensions. Note that each mixture in the problem can be used only once and in only one zone of a cell.

   ATRIANGPITCH or **ATRP** is used for annular cylindrical rods in a triangular-pitch lattice as shown in Fig. 7.1.5. The inner and outer clad and gap are independently entered, so they must be different materials and dimensions.

   ASPHSQUAREP or **ASSP** is used for spherical shells in a square-pitch lattice as shown in Fig. 7.1.4. The inner and outer clad and gap are independently entered, so they must be different materials and dimensions.

   ASPHTRIANGP or **ASTP** is used for spherical shells in a triangular-pitch (dodecahedral) lattice as shown in Fig. 7.1.5. The inner and outer clad and gap are independently entered, so they must be different materials and dimensions.

ASYMSLABCELL is used for a periodic, but asymmetric, array of slabs as shown in Fig. 7.1.6. The inner and outer clad and gap are independently entered, so they may be different materials and dimensions.

3. **PITCH** or **HPITCH**

   ARRAY PITCH. This is the center-to-center spacing or half-spacing between the fuel lumps (rods, pellets, or slabs), *pitch*, in cm followed by the outer moderator material number, mm, as shown in Fig. 7.1.1 through Fig. 7.1.6.

4. **FUELD** or **FUELR**

   OUTSIDE DIMENSION OF FUEL. This is the outside diameter or radius of the fuel, fuel, in cm followed by the fuel mixture number, *mf*, as shown in Fig. 7.1.1 through Fig. 7.1.6.

5. **GAPD** or **GAPR**

   OUTSIDE DIMENSION OF OUTER GAP. Enter only if outer gap is present. This is the outside diameter or radius of the outer gap, *gap*, in cm followed by the gap mixture number, mg, as shown in Fig. 7.1.1 through Fig. 7.1.6.

6. **CLADD** or **CLADR**

   OUTSIDE DIMENSION OF OUTER CLAD. Enter ONLY if a clad is present. This is the outside diameter or radius of the outer clad, *clad*, in cm followed by the clad mixture number, mc, as shown in Fig. 7.1.1 through Fig. 7.1.6.

7. **IMODD** or **IMODR**

   DIMENSION OF INNER MODERATOR. Enter ONLY if an annular cell is specified. This is the outside diameter or radius of the inner moderator, *imod*, in cm followed by the inner moderator mixture number, *mim*, as shown in Fig. 7.1.4 through Fig. 7.1.6.

8. **IGAPD** or **IGAPR**

   OUTSIDE DIMENSION OF INNER GAP. Enter ONLY if an annular cell is specified and inner gap is present. This is the outside diame*ter or radius of the inner gap, *igap*, in cm followed by the inner gap mixture number, *mig*, as shown in Fig. 7.1.4 through Fig. 7.1.6.

9. **ICLADD** or **ICLADR**

   OUTSIDE DIMENSION OF INNER CLAD. Enter ONLY if an annular cell is specified and inner clad is present. This is the outside diameter or radius of the inner clad, *iclad*, in cm followed by the inner clad mixture number, *mic*, as shown in Fig. 7.1.4 through Fig. 7.1.6.

10. {**CELLMIX=}mix**

    CELL-WEIGHTED MIXTURE NUMBER. [the = sign can be replaced by a space if desired). Enter ONLY if a cell-weighted mixture is to be generated. Enter a unique mixture number to be used by XSDRN to create the cell-weighted mixture (Sect. 7.1.2.4).

11. **END**

    The word **END** is entered to terminate the **LATTICECELL** data. An optional label can be associated with this **END**. The label can be as many as 12 characters long and is separated from the **END** by a single blank. At least two blanks must follow this entry. Must not start in column 1.

continues on next page

Table 7.1.8 – continued from previous page

Table 7.1.8: Unit cell specification for LATTICECELL problems

Entry no.	Input keyword		Comments		
1	**LATTICECELL**		Keyword to begin LATTICECELL unit cell. Enter the keyword **LATTICECELL**. This word may be truncated to any number of letters as long as they exactly replicate the beginning part of the keyword (e.g., **LAT** is acceptable).		
2			One of the following keywords is specified. This keyword determines the type of lattice or array configuration and which subsequent data need to be specified.		
	SQUAREPITCH		Used for cylindrical rods in a square pitch.		
	ASQUAREPITCH		Used for annular cylindrical rods in a square pitch.		
	ASQP		Used for annular cylindrical rods in a square pitch.		
	TRIANGPITCH		Used for cylindrical rods in a triangular pitch.		
	ATRIANGPITCH		Used for annular cylindrical rods in a triangular pitch.		
	ATRP		Used for annular cylindrical rods in a triangular pitch.		
	SPHSQUAREP		Used for spherical pellets in a cubic lattice.		
	ASPHSQUAREP		Used for annular spherical pellets in a cubic lattice.		
	ASSP		Used for annular spherical pellets in a cubic lattice.		
	SPHTRIANGP		Used for spherical pellets in a bi-centered or face-centered hexagonal close-packed lattice.		
	ASPHTRIANGP or **ASTP**		Used for annular spherical pellets in a bi-centered or face-centered hexagonal close-packed lattice.		
	SYMMSLABCELL		Used for a symmetric array of slabs.		
	ASYMSLABCELL		Used for a periodic but asymmetric array of slabs.		
Enter the following keywords and subordinate data as required to specify the unit cell. Each dimension can be entered as a diameter or radius using the appropriate keyword. The following keywords may be entered in any order.					
3	**PITCH**	**HPITCH**	Cell pitch or half-pitch (cm) + moderator mixture	Always	The pitch is the center-to-center spacing (cm) between fuel lumps. For asymmetric slab cell, **PITCH** is the center-to-center distance between inner moderators (cm).
4	**FUELD**	**FUELR**	Outside diameter or radius of fuel (cm) + fuel mixture	Always	Outside diameter (radius) of fuel (cm) or the thickness (half-thickness) of the fuel in a slab. For asymmetric slab cells, **FUELR** is measured from the centerline of the inner moderator.
5	**GAPD**	**GAPR**	Outside diameter or radius of outer gap (cm) + gap mixture	Outer gap Present	OMIT IF NO GAP between the fuel and clad outer clad. A mixture number of zero is often used. For asymmetric slab cells, **GAPR** is measured from the centerline of the inner moderator.
6	**CLADD**	**CLADR**	Outside diameter or radius of outer clad (cm) + clad mixture	Outer clad Present	OMIT IF NO CLAD between fuel and outer moderator. For asymmetric slab cells, **CLADR** is measured from the centerline of the inner moderator.
7	**IMODD**	**IMODR**	Outside diameter or radius of inner moderator (cm) + moderator mixture	Annular cell	Dimension of inner moderator mixture inside the fuel. For a slab, this is the moderator on the other side of the slab. Enter for annular cells only.
8	**IGAPD**	**IGAPR**	Outside diameter or radius of inner gap (cm) + gap mixture	Annular cell + Inner Gap Present	OMIT IF NO GAP between the inner clad and fuel. For an asymmetric slab, **IGAPR** is the distance from the center of the inner moderator to the outside of the inner gap (cm).
9	**ICLADD**	**ICLADR**	Outside diameter or radius of inner clad (cm) + clad mixture	Annular cell + Inner Clad Present	OMIT IF NO CLAD between fuel and inner moderator. For an asymmetric slab, **ICLADR** is the distance from the center of the inner moderator to the outside of the inner clad (cm).
10	**CELLMIX**		Unique mixture number	Cell-weighted mixture	Used if a cell-weighted mixture is specified. Calls XSDRN to produce a cell-weighted mixture (Section **7.1.2.4**).
11	**END**		Terminate **LATTICECELL** data	Always	Terminate the **LATTICECELL** input data by entering the word **END**. Do not start in column 1. At least two blanks or a new line must follow **END**.

continues on next page

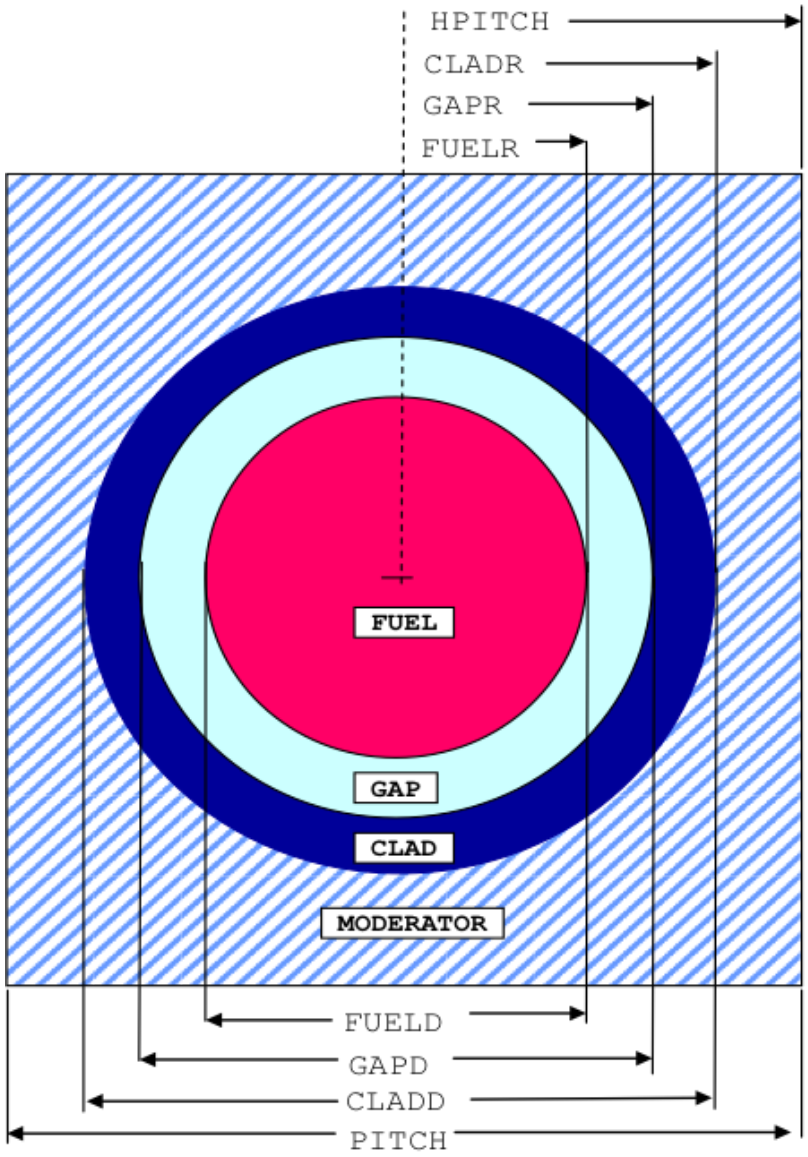


Fig. 7.1.1: Arrangement of materials in a SQUAREPITCH and SPHSQUAREP unit cell.

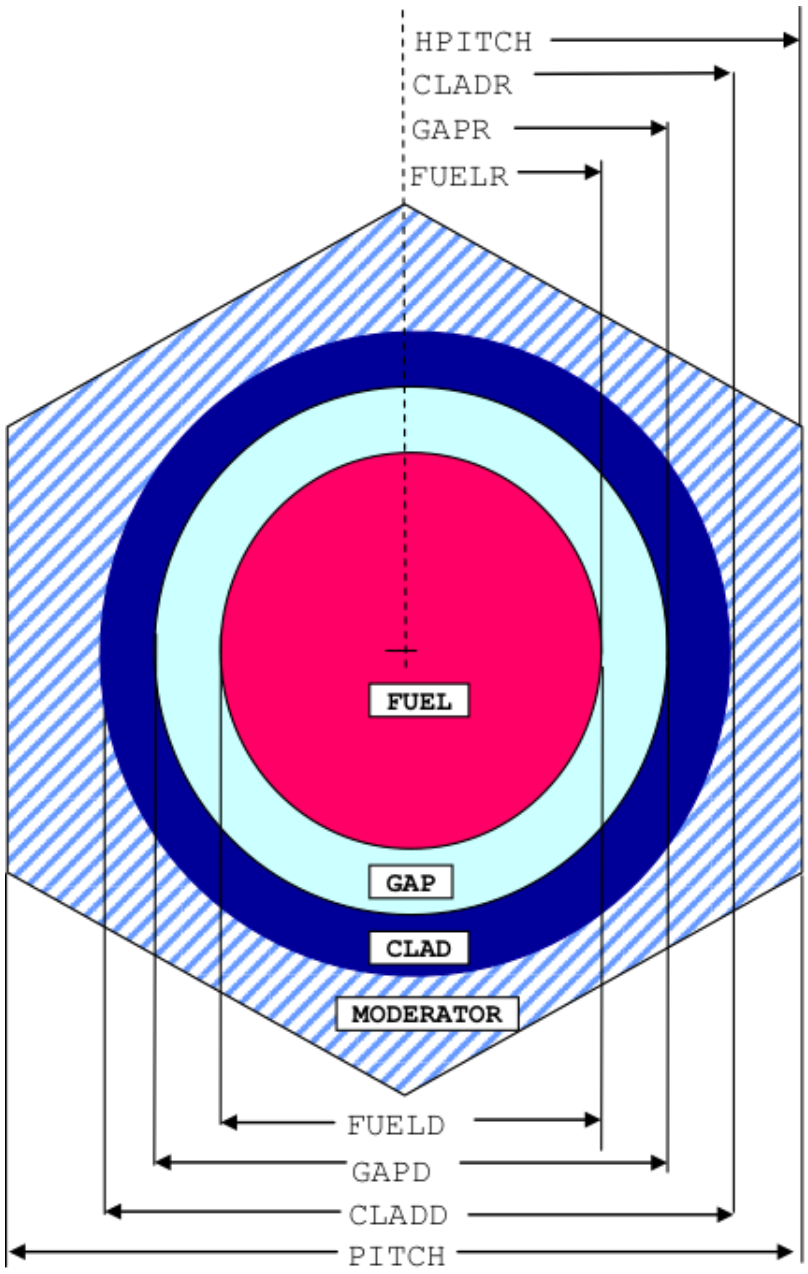


Fig. 7.1.2: Arrangement of materials in a TRIANGPITCH and SPHTRIANGP unit cell.

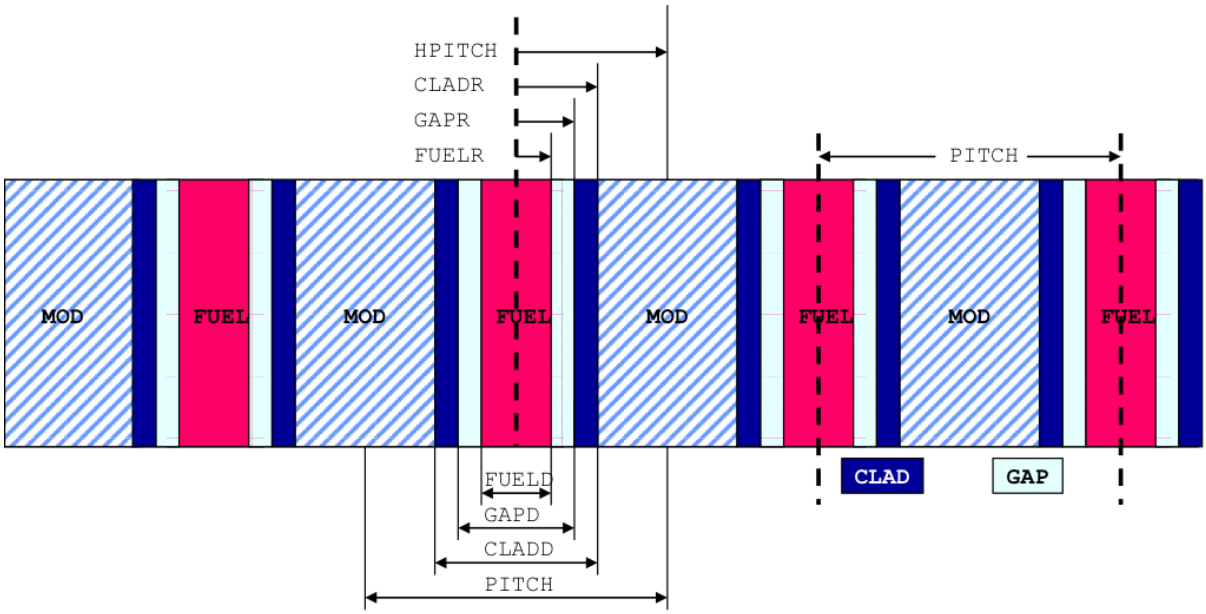


Fig. 7.1.3: Arrangement of materials in a SYMMSLABCELL unit cell having reflected left and right boundary conditions.

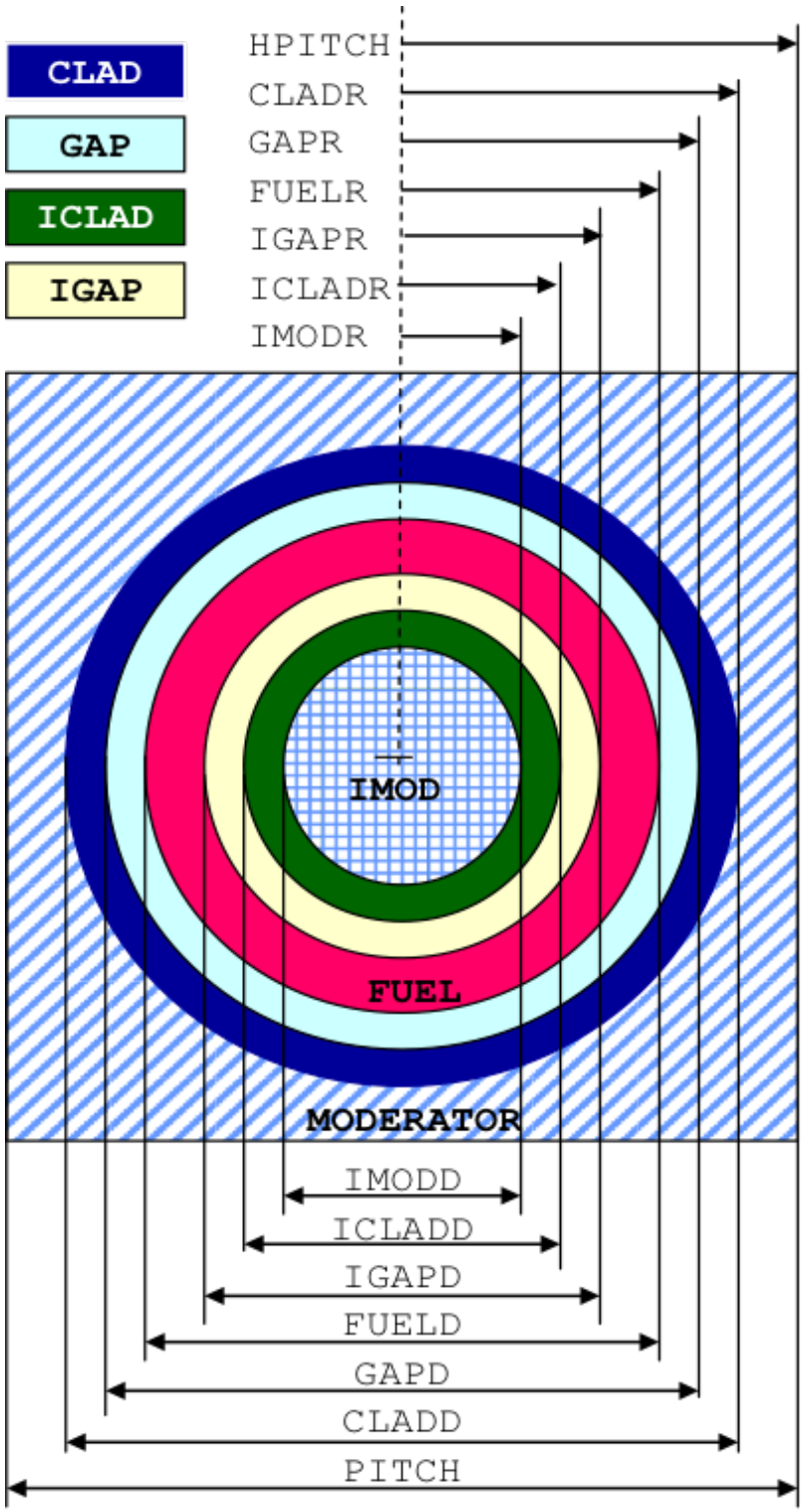


Fig. 7.1.4: Arrangement of materials in an ASQUAREPITCH and ASPHSQUAREP unit cell.

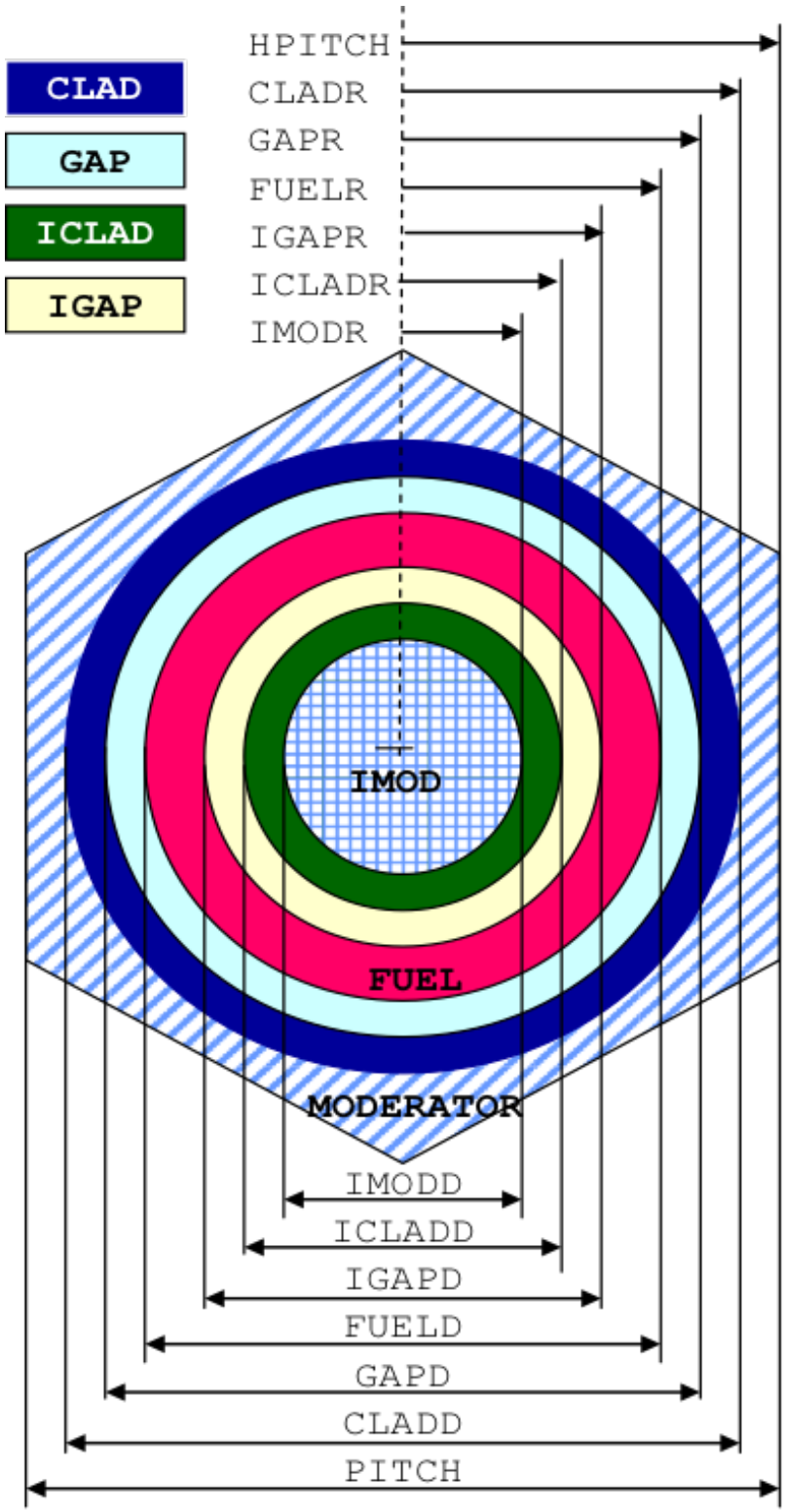


Fig. 7.1.5: Arrangement of materials in an ATRIANGPITCH and ASPHTRIANGP unit cell.

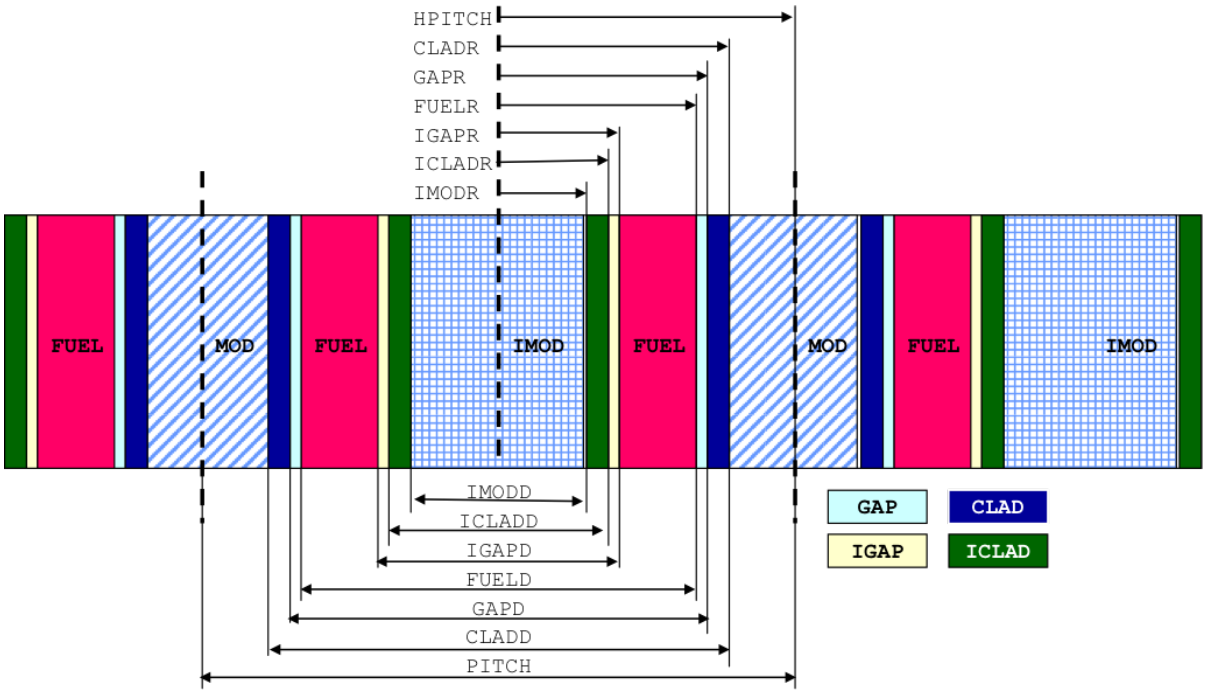


Fig. 7.1.6: Arrangement of materials in an ASYMSLABCELL unit cell having reflected left and right boundary conditions.

7.1.3.6 Unit cell specification for MULTIREGION cells

A **MULTIREGION** cell can be used to define a 1-D geometric arrangement that is more general than allowed by a **LATTICECELL**. It can also be used for large geometric regions where the geometry effects for the cross sections are small. For CENTRM/PMC self-shielding, lattice effects can be approximated by applying reflected, periodic, or white external boundary conditions to a MULTIREGION cell. HOWEVER, MULTIREGION CELLS SHOULD NOT BE USED FOR BONAMI-ONLY SELF-SHIELDING OF AN ARRAY UNIT CELL. In this case a LATTICECELL should always be used for BONAMI self-shielding in order to incorporate the proper Dancoff effects.

The data required for a MULTIREGION cell are given in Table 7.1.9 and explained in the following text.

1. **celltype**

   MULTIREGION. The keyword **MULTIREGION** is used to represent arbitrary 1-D geometries, with no restrictions the on number or placement of mixtures in the cell. The keyword may be truncated to any number of characters as long as the characters presented are identical from the beginning of the keyword (i.e., M is acceptable).

2. **cs**

   TYPE OF GEOMETRY. The type of geometry must always be specified for a **MULTIREGION** cell. The available geometry options are listed below.

   SLAB. This is used to describe a slab geometry.

   CYLINDRICAL. This is used to describe cylindrical geometry.

   SPHERICAL. This is used to describe spherical geometry.

BUCKLEDSLAB. This is used for slab geometry with a buckling correction for the two transverse directions. Inactive in SCALE 6.2 and later.

BUCKLEDCYL. This is used for cylindrical geometry with a buckling correction in the axial direction. Inactive in SCALE 6.2 and later.

3. **RIGHT_BDY**

RIGHT BOUNDARY CONDITION. This is defaulted to **VACUUM**. The available options and their qualifications are listed below.

VACUUM. This imposes a vacuum at the boundary of the system.

REFLECTED. This imposes mirror image reflection at the boundary. Do not use for **CYLINDRICAL** or **SPHERICAL**.

PERIODIC. This imposes periodic reflection at the boundary. Do not use for **CYLINDRICAL** or **SPHERICAL**.

WHITE. This imposes isotropic return at the boundary.

4. **LEFT_BDY**

LEFT BOUNDARY CONDITION. This is defaulted to **REFLECTED**. The available options and their qualifications are listed below.

VACUUM. This imposes a vacuum at the boundary of the system.

REFLECTED. This imposes mirror image reflection at the boundary. For **CYLINDRICAL** or **SPHERICAL**, this is the only valid boundary condition because the left boundary corresponds to the centerline of the cylinder or the center of the sphere.

PERIODIC. This imposes periodic reflection at the boundary.

WHITE. This imposes isotropic return at the boundary.

5. **ORIGIN**

LOCATION OF LEFT BOUNDARY ON THE ORIGIN. The default value of **ORIGIN** is 0.0. This is the only value allowed for **CYLINDRICAL** or **SPHERICAL** geometry. For **SLAB**s, enter the location of the left boundary on the X-axis perpendicular to the slab (in cm).

6. **DY**

BUCKLING HEIGHT. This is the buckling height in cm. It corresponds to one of the transverse dimensions of an actual 3-D slab assembly or the length of a finite cylinder. Inactive in SCALE 6.2 and later.

7. **DZ**

BUCKLING DEPTH. This is the buckling width in cm. It corresponds to the second transverse dimension of an actual 3-D slab assembly. Inactive in SCALE 6.2 and later.

8. **CELLMIX**

CELL-WEIGHTED MIXTURE NUMBER. Enter ONLY if a cell-weighted mixture is required. Enter a unique mixture number used to create a cell-weighted homogeneous mixture (Sect. 7.1.2.4).

9. **END**

   The word **END** is entered to terminate these data before entering the zone description data. It must not be entered in columns 1 through 3, and at least two blanks must separate it from the zone description. A label can be associated with this **END**. The label can be a maximum of 12 characters and is separated from the **END** by a single blank. At least two blanks must follow this entry.

   The zone description data are entered at this point. Entries 10 and 11 are entered for each zone, and the sequence is repeated until all the desired zones have been described. To terminate the data, enter the words END ZONE. Zone dimensions must be in increasing order.

10. **mxz**

    MIXTURE NUMBER IN THE ZONE. Enter the mixture number of the material that is present in the zone. Enter a zero for a void. Repeat the sequence of entries 10 and 11 for each zone. Mixtures other than zero must not be used more than once in a cell and may be used in no more than one cell.

11. **rz**

    OUTSIDE RADIUS OF THE ZONE. Enter the outside dimension of the zone in cm.

    In **SLAB** geometry, **rz** is the location of the zone's right boundary on the X-axis. Repeat the sequence of entries 10 and 11 for each zone.

12. **END ZONE**

    Is used to terminate the **MULTIREGION** zone data. Enter the words **END ZONE** when all the zones have been described. Note that **ZONE** is a label associated with this **END**. This label can be as long as 12 characters, but the first four characters must be **ZONE**. At least two blanks must follow this entry.

continues on next page

Table 7.1.9 – continued from previous page

Table 7.1.9: Unit cell specification for MULTIREGION problems.

Entry no.	Input keyword	Data following keyword	Entry requirement	Comments
1	**MULTIREGION**		Always	Keyword to begin multiregion unit cell. Enter the keyword **MULTIREGION**. This word may be truncated to any number of letters as long as they exactly replicate the beginning part of the keyword (e.g., **M** is acceptable).
2			Always	One of the following keywords is specified. This keyword determines the type of unit cell geometry.
	SLAB			Used for slab geometry.
	CYLINDRICAL			Used for cylindrical geometry.
	SPHERICAL			Used for spherical geometry.
	BUCKLEDSLAB			Used for slab geometry with a buckling correction for the two transverse directions.
	BUCKLEDCYL			Used for cylindrical geometry with a buckling correction in the axial direction.
3	**RIGHT_BDY**	**VACUUM** **REFLECTED** **PERIODIC** **WHITE**	Optional for all geometries	Default is **VACUUM**. Describes the right/outer boundary condition. This provides a non-return condition at the boundary. **REFLECTED** or **PERIODIC** not allowed for cylindrical or spherical.
4	**LEFT_BDY**	**VACUUM** **REFLECTED** **PERIODIC** **WHITE**	Optional for slab type geometries	Default is **REFLECTED**. Describes the left/inner boundary condition. Do not change for cylindrical or spherical. **VACUUM** provides a non-return condition at the boundary. **WHITE** provides isotropic return at the boundary.
5	**ORIGIN**	Left boundary location (cm)	Optional for slab type geometries	Default is 0.0. Should not be changed for cylindrical or spherical geometry.
6	**DY**	Buckling height (cm)		OMIT FOR **SLAB**, **CYLINDRICAL**, and **SPHERICAL**. This corresponds to one of the transverse dimensions of an actual 3-D slab assembly or to the length of a finite cylinder.
7	**DZ**	Bucking depth (cm)		OMIT UNLESS **BUCKLEDSLAB** IS SPECIFIED. This is the buckling depth corresponding to the second transverse dimension of a 3-D slab assembly.

(continued)

Entry no.	Input keyword	Data following keyword	Entry requirement	Comments
8	**CELLMIX**	Unique mixture no.	Cell-weighted mixture required	Used if a cell-weighted mixture is specified. Enter a unique mixture number in the problem (See **Cell weighting of MG cross sections**).
9	**END**		Always	Enter the word **END**. Do not start in column 1. At least two blanks or a new line must separate **END** from the next entry.
10		Zone mixture number	Always	Enter mixture number in zone. Repeat 10 and 11 until all zones are specified.
11		Zone outside radius (cm)	Always	Enter outside radius of zone. Repeat 10 and 11 until all zones are specified.
12	**END ZONE**		Always	Terminates MULTIREGION data. At least two blanks must follow.

continues on next page

Table 7.1.9 – continued from previous page

7.1.3.7 Unit cell specification for doubly heterogeneous (DOUBLEHET) cells

The data required for a **DOUBLEHET** cell are given in Table 7.1.10 and explained in the following text.

Details about the computation procedures for **DOUBLEHET** cells can be found in Sect. 7.1.2.3.

"Grain" refers to a spherical fuel particle surrounded by one or more coating zones and does not include the matrix material the grains are in. "Grain type" refers to a grain that has specified dimensions and mixtures such as a 0.025-cm-radius UO_2 fuel kernel with a 0.01-cm-thick carbon coating. Another grain type could be a 0.012-cm-radius PuO_2 fuel kernel with a 0.01-cm-thick carbon coating. The user must first define all grain types in a fuel element. Next, all fuel element–related data must be entered.

Since all grains and the matrix material are homogenized into a single uniform mixture for the fuel element, there are restrictions on how each grain type must be defined so that the volume fraction of each grain type within the homogenized fuel mixture can be determined. Related entries are **PITCH**, **NUMPAR** (number of particles), and **VF** (volume fraction). If there is only one grain type for a fuel element, the code needs the pitch and will directly use the input value if entered. If **PITCH** is not given, then the **VF** (if given) is used to calculate the pitch. If neither **PITCH** nor **VF** is given, then **NUMPAR** is used to calculate the pitch and the volume fraction. The user should only enter one of these items.

If more than one grain type is present, additional information is needed since all grain types are homogenized into a single mixture. Similar to the one grain type case, the pitch is needed to perform the CENTRM spherical cell calculations. However, the pitch by itself is not sufficient to perform the homogenization. Therefore, the user needs to input **VF** or **NUMPAR** for each grain type. Since each grain's volume is known (grain dimensions must always be entered), entering **NUMPAR** or **VF** for each grain type essentially provides the total volume of each grain type and therefore enables the calculation of the other unknowns (**VF** or **NUMPAR**, and **PITCH**). In this case, since pitch is not given, the available matrix material is distributed around the grains of each grain type proportional to the grain volume to calculate the corresponding pitch.

Syntax:

DOUBLEHET *fuelmix* **END**

GF(**D**|**R**)=*fuel mg* (**COAT**(**D**|**R**)=*coat mc*)|(**COATT**=*coat mc*) {**H**}**PITCH**=*mod* **MATRIX**=*mm* **NUMPAR**=*npar* **VF**=*vf* **END GRAIN**

mct ctp FUEL(**D**|**R**)=*mfuel* {**FUELH**=*hfuel*} {**FUELW**=wfuel} {**GAP**(**D**|**R**)=*mgap mmg*} {**CLAD**(**D**|**R**)=*mclad mmc*} {**H**}**PITCH**=*mpitch mmm* {**CELLMIX**=*mcmx*} **END**

1. **celltype**

   DOUBLEHET. The keyword DOUBLEHET is used to represent a doubly heterogeneous problem such as fuel units that are composed of grains of fuel.

2. *fuelmix*

   HOMOGENIZED MIXTURE NUMBER. Enter a unique mixture number to be used for the homogenized grains and matrix material.

3. **END**

   The word **END** is entered to terminate these data before entering the grain and fuel element description data. It must not be entered in columns 1 through 3, and at least two blanks must separate it from the zone description. A label can be associated with this **END**. The label can be a maximum of 12 characters and is separated from the **END** by a single blank. At least two blanks must follow this entry.

The grain description data are entered at this point. Entries 5 through 12 are entered for each grain, and the sequence is repeated until all the grains have been described. To terminate the data, enter the words **END GRAIN**. Data may be entered in any order.

4. **PITCH** or **HPITCH**

   EQUIVALENT CELL DIMENSION. This is the equivalent spherical diameter (or radius), in cm, of the "average" unit cell for this grain, as shown in Fig. 7.1.7. Physically, the volume of the average unit cell is equal to the volume of the fuel element divided by the total number of all grain types.

5. **GFD** or **GFR**

   OUTSIDE DIMENSION OF FUEL. This is the outside diameter or radius of the fuel zone in a grain, *fuel*, in cm followed by the fuel mixture number, *mg*, as shown in Fig. 7.1.7.

6. **COATD** or **COATR**

   OUTSIDE DIMENSION OF COATING. This is the outside diameter or radius of a coating zone, *coat*, in cm followed by the coating mixture number, *mc*, as shown in Fig. 7.1.7. As many coating-mixture pairs as desired may be entered. If the coating dimensions are entered using COATD or COATR, then the COATT keyword should not be used.

7. **COATT**

   THICKNESS OF COATING. This is the thickness of a coating zone, *coat*, in cm followed by the coating mixture number, *mc*, as shown in :numref`fig7-1-7`. As many coating-mixture pairs as desired may be entered. If the coating dimensions are entered using COATT, then the COATD or COATR keyword should not be used.

8. **MATRIX**

   MIXTURE NUMBER OF THE MATRIX MATERIAL. This is the mixture number, *mm*, of the matrix material that encloses the grains.

9. **NUMPAR**

   NUMBER OF PARTICLES. This is the number of grains, *npar*, of this type in each fuel element.

10. **VF**

    VOLUME FRACTION. This is the volume fraction, *vf*, of grains of this type in each fuel element's fuel zone. A fuel element's fuel zone is entered using the entry number 16-**FUELD** (or **FUELR**).

11. **END GRAIN**

    This is used to terminate the grain zone data for this grain type. At least two blanks must follow this entry.

REPEAT ENTRIES 4-11 FOR EACH GRAIN TYPE IN A FUEL ELEMENT.

12. **mct**

    TYPE OF FUEL ELEMENT (macro cell type). One of the keywords **PEBBLE** or **ROD** or SLAB is entered to indicate the type of the fuel element, i.e., the second layer of heterogeneity. This data must be entered. The keyword may NOT be truncated. **PEBBLE** is used for spherical fuel elements; **ROD** is used for cylindrical fuel elements; and SLAB for plate fuel elements.

13. **ctp**

TYPE OF LATTICE. This defines the type of lattice or array configuration. Any one of the following alphanumeric descriptions may be used. Note that the alphanumeric description must be separated from subsequent data entries by one or more blanks. Fig. 7.1.1 Mixture and position data are entered using keywords. Mixture number 0 may be entered for void and may be used multiple times in each and all unit cells. For regular cells, the minimum requirement is that a fuel region and a moderator region are specified and no inner components are specified. For annular cells, the minimum requirement is the fuel and outer moderator and inner moderator regions are specified. Regular and annular cell configurations are specified as shown below.

Regular Cells

SQUAREPITCH is used for an array of cylinders arranged in a square lattice, as shown in Fig. 7.1.1. The clad and/or gap can be omitted.

TRIANGPITCH is used for an array of cylinders arranged in a triangular-pitch lattice as shown in Fig. 7.1.2. The clad and/or gap can be omitted.

SPHSQUAREP is used for an array of spheres arranged in a square-pitch lattice. A cross section view through a cell is represented by Fig. 7.1.1. The clad and/or gap can be omitted.

SPHTRIANGP is used for an array of spheres arranged in a triangular-pitch (dodecahedral) lattice. A cross section view through a cell is represented by Fig. 7.1.2. The clad and/or gap can be omitted.

SYMMSLABCELL is used for an infinite array of symmetric slab cells, as shown in Fig. 7.1.3. The clad and/or gap can be omitted.

Annular Cells

ASQUAREPITCH or **ASQP** is used for annular cylindrical rods in a square-pitch lattice as shown in Fig. 7.1.4. The inner and outer clad and gap are independently entered so they may be different materials and dimensions.

ATRIANGPITCH or **ATRP** is used for annular cylindrical rods in a triangular-pitch lattice as shown in Fig. 7.1.5. The inner and outer clad and gap are independently entered, so they may be different materials and dimensions.

ASPHSQUAREP or **ASSP** is used for spherical shells in a square-pitch lattice as shown in Fig. 7.1.4. The inner and outer clad and gap are independently entered, so they may be different materials and dimensions.

ASPHTRIANGP or **ASTP** is used for spherical shells in a triangular-pitch (dodecahedral) lattice as shown in Fig. 7.1.5. The inner and outer clad and gap are independently entered, so they may be different materials and dimensions.

ASYMSLABCELL is used for a periodic, but asymmetric, array of slabs as shown in Fig. 7.1.6. The inner and outer clad and gap are independently entered, so they may be different materials and dimensions.

14. **PITCH** or **HPITCH**

ARRAY PITCH. This is the center-to-center spacing or half-spacing between the fuel lumps (pebbles or rods or slabs), *mpitch*, in cm followed by the outer moderator material number, *mmm*, as shown in Fig. 7.1.1 and Fig. 7.1.2.

15. **FUELD** or **FUELR**

    OUTSIDE DIMENSION OF FUEL. This is the outside dimension (diameter or radius for sphere/cylinder or x-thickness for slab) of the fuel region, *mfuel*, in cm, as shown in Fig. 7.1.1 and Fig. 7.1.2.

16. **FUELH**

    HEIGHT OF FUEL ROD OR SLAB. This is the height (z-dimension) of the fuel plate, *hfuel*, in cm. (only used to compute volume of fuel plate).

17. **FUELW**

    WIDTH OF FUEL ROD or slab. This is the width/depth (y-dimension) of the fuel plate, *wfuel*, in cm. (only used to compute volume of fuel plate).

18. **GAPD** or **GAPR**

    OUTSIDE DIMENSION OF GAP. Enter only if outer gap is present. This is the outside diameter or radius of the outer gap, *mgap*, in cm followed by the gap mixture number, *mmg*, as shown in Fig. 7.1.1 and Fig. 7.1.2.

19. **CLADD** or **CLADR**

    OUTSIDE DIMENSION OF CLAD. Enter ONLY if a clad is present. This is the outside diameter or radius of the outer clad, *mclad*, in cm followed by the clad mixture number, *mmc*, as shown in Fig. 7.1.1 and Fig. 7.1.2.

20. **CELLMIX**

    CELL-WEIGHTED MIXTURE NUMBER. Enter ONLY if cell-weighted mixture, *mcmx*, is to be created.

21. **IMODD** or **IMODR**

    DIMENSION OF INNER MODERATOR. Enter ONLY if an annular cell is specified. This is the outside diameter or radius of the inner moderator, *imod*, in cm followed by the inner moderator mixture number, *mim*, as shown in Fig. 7.1.4 through Fig. 7.1.6.

22. **IGAPD** or **IGAPR**

    OUTSIDE DIMENSION OF INNER GAP. Enter ONLY if an annular cell is specified and inner gap is present. This is the outside diameter or radius of the inner gap, *igap*, in cm followed by the inner gap mixture number, *mig*, as shown in Fig. 7.1.4 through Fig. 7.1.6.

23. **ICLADD** or **ICLADR**

    OUTSIDE DIMENSION OF INNER CLAD. Enter ONLY if an annular cell is specified and inner clad is present. This is the outside diameter or radius of the inner clad, *iclad*, in cm followed by the inner clad mixture number, *mic*, as shown in Fig. 7.1.4 through Fig. 7.1.6.

24. **END**

    The word **END** is entered to terminate the **DOUBLEHET** data. An optional label can be associated with this **END**. The label can be as many as 12 characters long and is separated from the **END** by a single blank. At least two blanks must follow this entry.

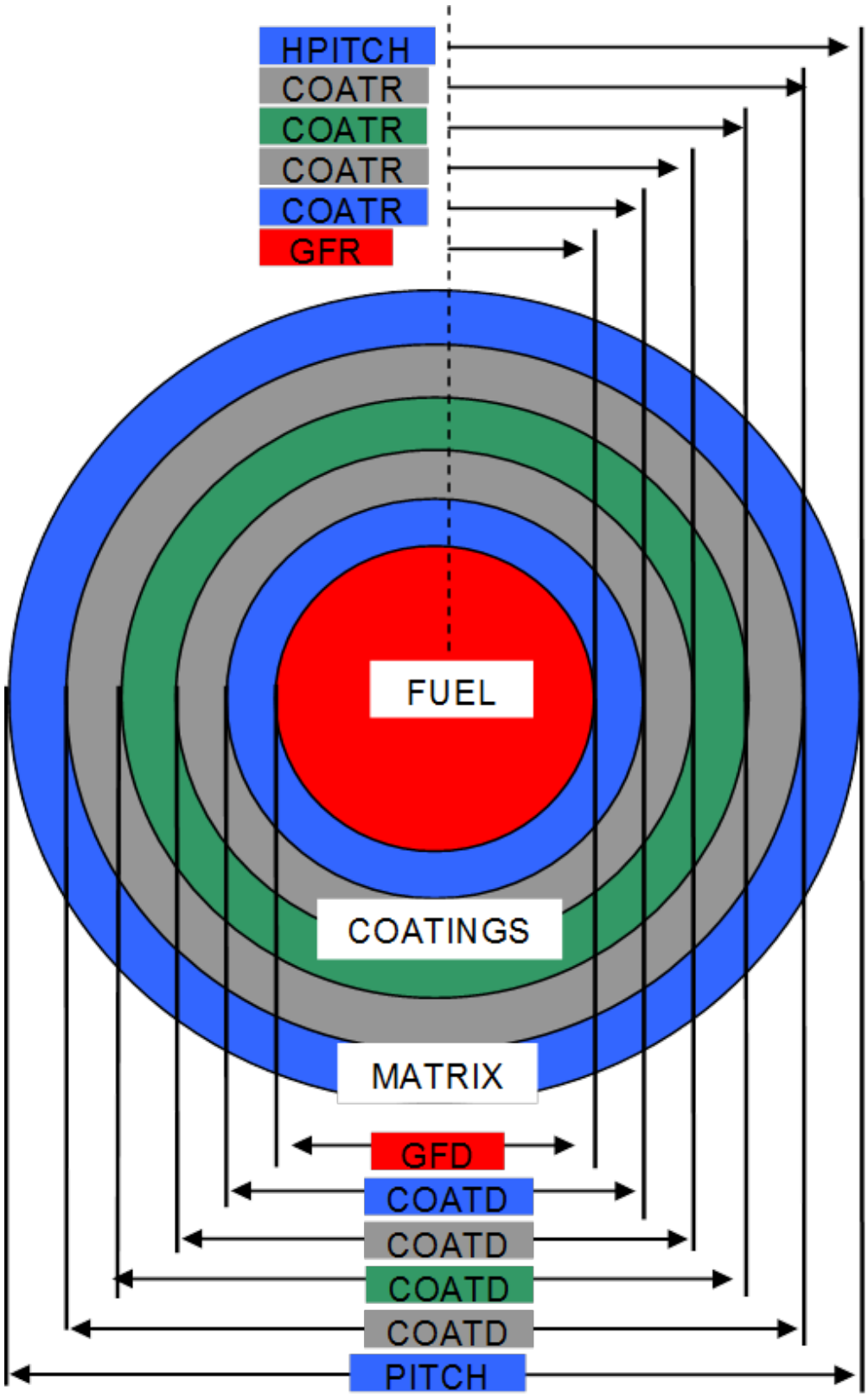


Fig. 7.1.7: Arrangement of materials in a grain (first level cell) in a DOUBLEHET unit cell.

Table 7.1.10: Unit cell specification for DOUBLEHET problems

GEOMETRY (region)

Format: READ GEOM enter geometry region data here END GEOM

See sections on Geometry, Geometry data, Multiple entries in geometry data, and Alternative Sample problem mockups.

GEOMETRY REGION DATA consist of SIMPLE GEOMETRY REGION DATA and EXTENDED GEOMETRY REGION DATA.

ENTER GEOMETRY DATA IN THE FOLLOWING FORM:

OPTIONAL GLOBAL SPECIFICATION

UNIT n

OPTIONAL GEOMETRY COMMENT

GEOMETRY REGION DATA and/or EXTENDED GEOMETRY REGION DATA

****

ENTER SIMPLE REGION DATA IN THE FOLLOWING FORM:

GLOBAL Enter only to specify this unit as the global unit.

UNIT n

COM=delim comment delim This optional comment can be up to 132 characters long. It must begin and end with a delimiter (i.e. “ , ‘ , * , ^ , or !)

fgeom mix no. bias ID dimensions optional origin data (ORIGIN coordinates) optional chord data (CHORD distance) optional center data (CENTER type X-offset Y-offset Z-offset)

Enter as many geometry description specifications as necessary to describe the unit and as many units as necessary to describe the system.

SIMPLE GEOMETRY REGION INPUT DATA REQUIREMENTS

TYPE OF DATA	TYPE 1 DATA	TYPE 2 DATA	TYPE 3 DATA	TYPE 4 DATA	TYPE 5 DATA	TYPE 6 DATA
fgeom	SPHERE	XCYLINDER	YCYLINDER	CYLINDER	CUBE	CUBOID
	HEMISPHERE	XHEMICYL+Y	YHEMICYL+X	ZCYLINDER		
	HEMISPHE+X	XHEMICYL–Y	YHEMICYL–X	ZHEMICYL+X		
	HEMISPHE–X	XHEMICYL+Z	YHEMICYL+Z	ZHEMICYL–X		
	HEMISPHE+Y	XHEMICYL–Z	YHEMICYL–Z	ZHEMICYL+Y		
	HEMISPHE–Y			ZHEMICYL–Y		
	HEMISPHE+Z					
	HEMISPHE–Z					
dimensions	R (radius)	R +X –X	R +Y –Y	R +Z –Z	+X –X	+X –X +Y –Y +Z –Z
optional origin coordinates*	Enter the X Y Z coordinates of origin	Enter the Y Z coordinates of centerline	Enter the X Z coordinates of centerline	Enter the X Y coordinates of centerline	omit	omit
optional chord data**	Enter the dist. to plane	Enter the dist. to plane	Enter the dist. to plane	Enter the dist. to plane	omit	omit
optional center data***	Center type	X-offset	X-offset	Z-offset	omit	omit

*Enter ORIG or ORIGIN for *fgeom*.

**Enter CHORD for *fgeom*.

***Center data are only applicable when flux moments (PNM>0) or angular fluxes (NQD>0) are requested with the coordinate transform (TFM) set to YES.

NOTE: Chord data are not applicable for SPHERE, XCYLINDER, YCYLINDER, CYLINDER, ZCYLINDER, CUBE, or CUBOID.
Origin data are not applicable for a CUBE or CUBOID.

Table 7.1.11: Unit cell specification for DOUBLEHET problems (continued).

GEOMETRY (region) (continued)

ENTER GEOMETRY DATA IN THE FOLLOWING FORM:

fgeom ref. ID bias ID thickness per region origin coordinates nreg

EXTENDED GEOMETRY REGION INPUT DATA REQUIREMENTS

TYPE OF DATA	TYPE 1 DATA	TYPE 2 DATA	TYPE 3 DATA
fgeom	ARRAY	HOLE	REPLICATE REFLECTOR
ref. ID	array no.	emplaced unit number	mixture number in generated regions
bias ID	omit for ARRAY	omit	first bias ID
thickness / region	omit	omit	variable[a]
origin coordinates	Enter the X Y Z coordinates of the most negative point of the array	Enter the X Y Z coordinates of the origin	omit
nreg	omit	omit	number of regions to be generated

[a]The number of dimensions to be entered is the same as the region preceding the replicate or reflector specification because the g enerated regions have that shape. The value of the dimensions is the thickness of each generated region of material on that surface and must therefore be a positive value.

continues on next page

Table 7.1.12: Unit cell specification for DOUBLEHET problems (continued).

(continued)

14	**PITCH** or **HPITCH**	Macro cell array pitch + moderator mixture	Always	ARRAY PITCH. This is the center-to-center spacing or half-spacing between the fuel lumps (pebbles or rods) in cm followed by the outer moderator material number, as shown in Figure 1 and Figure 2.
15	**FUELD** or **FUELR**	Macro cell fuel diameter or radius	Always	OUTSIDE DIMENSION OF FUEL. This is the outside diameter or radius of the fuel in cm, as shown in Figure 1 and Figure 2.
16	**FUELH**	Macro cell fuel height	Optional	HEIGHT OF FUEL ROD OR SLAB. This is the height of the fuel in cm.
17	**FUELW**	Macro cell fuel width	Optional	WIDTH OF FUEL SLAB. This is the width of the fuel in a plate element, in cm.
18	**GAPD** or **GAPR**	Macro cell gap diameter or radius + gap mixture	Optional	OUTSIDE DIMENSION OF GAP. Enter only if outer gap is present. This is the outside diameter or radius of the outer gap in cm followed by the gap mixture number, as shown in Figure 1 and Figure 2.
19	**CLADD** or **CLADR**	Macro cell clad diameter or radius + clad mixture	Optional	OUTSIDE DIMENSION OF CLAD. Enter ONLY if a clad is present. This is the outside diameter or radius of the outer clad in cm followed by the clad mixture number, as shown in Figure 1 and Figure 2.
20	**IMODD or IMODR**	Macro cell inner moderator diameter or radius + moderator mixture	Optional	OUTSIDE DIMENSION OF INNER MODERATOR . Enter ONLY for annular cells or asymmetric slab cells
21	**IGAPD or IGAPR**	Macro cell inner gap diameter or radius + gap mixture	Optional	OUTSIDE DIMENSION OF INNER GAP Enter ONLY if Annular cell + Inner Gap Present
22	**ICLADD or ICLADR**	Macro cell inner clad diameter or radius + clad mixture	Optional	OUTSIDE DIMENSION OF INNER CLAD. Enter ONLY if Annular cell + Inner Clad Present
23	**CELLMIX**	Cell-weighted mixture number	Optional	CELL-WEIGHTED MIXTURE NUMBER. Enter ONLY if cell-weighted macro-cell mixture is to be created (**Cell weighting of MG cross sections**).
24	**END**		Always	The word **END** is entered to terminate the **DOUBLEHET** data. An optional label can be associated with this **END**. The label can be as many as 12 characters long and is separated from the **END** by a single blank. At least two blanks must follow this entry.

continues on next page

7.1.3.8 Optional MORE DATA parameter data

MORE DATA is an optional sub-block of the **READ CELL** block. **MORE DATA** parameters allow certain default options in BONAMI and XSDRNPM to be modified for individual cell calculations. Each **MORE DATA** sub-block applies only to the unit cell immediately preceding it. However a **MORE DATA** sub-block placed prior to all unit cell definitions applies to all mixtures not assigned to a unit cell, which are treated as infinite homogeneous media. If the default parameters are acceptable, this section of input data should be omitted in its entirety. Non-default values for one or more of the parameters can be specified by entering the words **MORE DATA** followed by the desired keyword parameters and their associated values. One or more of the parameters can be entered in any order. Default values are used for parameters that are not entered. Each parameter is entered by spelling its name, followed immediately by an equal sign and the value to be entered. There should not be a blank between the parameter name and the equal sign. Each parameter specification must be separated from the rest by at least one blank. For example, if an XSDRNPM calculation is performed for particular unit cell (e.g., *cellmix*= is specified),

MORE DATA ISN=16 **EPS**=0.00008 **END MORE**

would result in using an S16 angular quadrature set and tightening the convergence criteria to 0.00008 in the XSDRNPM calculation.

A description of each entry is given. (Also see sections on BONAMI and XSDRNPM input description.)

1. **MORE DATA** These words, followed by one or more blanks, are entered ONLY if optional parameter data are to be entered. Entries 2 through 42 can be entered in any order.
2. **NSENSX** This is the XSDRNPM sensitivity output file for TSUNAMI sequences.
3. **CROSSEDT** BONAMI CROSS SECTION EDIT. Cross section print option for BONAMI 0/1 –no/yes (default is 0).
4. **FFACTEDT** BONDARENKO FACTOR EDIT. Bondarenko factor (f-factor) print option 0/1 –no/yes (default is 0).
5. **ISSOPT** BONAMI BACKGROUND XSEC OPTIONS. BONAMI background cross section selection option if > 1000 potential; otherwise, total cross section is used (default is –1).
6. **IROPT** BONAMI IR/NR CALCULATION OPTION. BONAMI uses intermediate resonance (IR) if iropt=1 and narrow resonance (NR) approximation for iropt=0 (default is 0).
7. **BELLOPT** BELL FACTOR OPTION. Optional user-defined bell factor calculation option (default is -1).
8. **BELLFACT** BELL FACTOR. Optional user-defined bell factor for BONAMI (default is 0.0).
9. **ESCXSOPT** ESCAPE CROSS SECTION CALC OPTION. Escape cross section calculation for IR calculations. 0/1 =consistent/inconsis tent (default is 0).
10. **BONAMIEPS** BONAMI CONVERGENCE CRITERIA. BONAMI Bondarenko iteration convergence criteria (default is 0.001).
11. **LBARIN** INPUT MEAN CORD LENGTH. Mean cord length for each zone (default is 0.00).
12. **ADJTHERM** ADJUST 1D THERMAL CROSS SECTIONS TO MATCH SUM OF 2D CROSS SECTIONS. Flag determining whether 1-D cross sections are scaled to match the 2-D cross sections or the 2-D cross sections are scaled to match the 1-D cross sections.

13. **EXSIG** ESCAPE CROSS SECTION. External escape cross section for BONAMI (default is 0.00).

14. **IEVT** XSDRNPM CALCULATION TYPE. The type of calculation to be performed- fixed source, eigenvalue, alpha, zone width search, outer radius search, buckling search, direct buckling search (default is 1).

15. **ICLC** THEORY OPTION. Number of outer iterations to use an alternative theory (diffusion, infinite medium, or B_N) before using discrete ordinates. Negative values indicate alternative theory (default is 0).

16. **IPVT** PARAMETRIC EIGENVALUE SEARCH. 0 – none; 1 – search for eigenvalue equal PV; 2 – alpha search (default is 0).

17. **IPP** WEIGHTED CROSS SECTION PRINT. 2 -> No print; -1 -> 1-D edit; 0-N – edit through PN cross section arrays (default is 2).

18. **IFLU** GENERALIZED ADJOINT CALCULATION. 0 is a standard calculation; 1 is a generalized adjoint calculation (default is 0).

19. **IFSN** FISSION SOURCE SUPPRESSION. Non-zero suppresses the fission source in a fixed source calculation (default is 0).

20. **IQM** VOLUMETRIC FIXED SOURCES. The number of volumetric sources in a fixed source problem (default is 0).

21. **IPM** BOUNDARY FIXED SOURCES. The number of boundary sources in a fixed source problem (default is 0).

22. **XNF** SOURCE NORMALIZATION FACTOR. The value used to normalize the problem source (default is 1.0).

23. **VSC** VOID STREAMING CORRECTION. The height of a void streaming path in a cylinder or slab in centimeters (default is 0.0).

24. **EV** EIGENVALUE GUESS. Starting eigenvalue guess for a search calculation (default is 0.0).

25. **EQL** INITIAL SEARCH CONVERGENCE. Initial eigenvalue search convergence (default is 0.0001).

26. **XNPM** DAMPING FACTOR. Damping factor used in search calculations (default is 0.75).

27. **ISN** ORDER OF ANGULAR QUADRATURE FOR XSDRNPM. Quadrature sets are geometry-dependent quantities that are defaulted to order 8 by the XSProc for **LATTICECELL** and cylindrical **MULTIREGION**. The default is 32 for **MULTIREGION** slabs and spheres. See the automatic quadrature generator and Appendix B for a more detailed explanation.

28. **SZF** SPATIAL MESH SIZE FACTOR FOR XSDRNPM. The size of the mesh intervals can be adjusted by entering a value for **SZF**, which is a multiplier of the mesh size. The default value is 1.0. A value between zero and 1.0 yields a finer mesh; a value greater than 1.0 yields a coarser mesh. If **SZF** $\leq$ 0, the user specifies the number of mesh intervals in each zone immediately following the **MORE DATA** block. If **SZF** = 0, the interval spacing is automatically generated, while if **SZF** < 0 the intervals are equally spaced intervals in each zone.

29. **IIM** MAXIMUM NUMBER OF INNER ITERATIONS FOR XSDRNPM. This is the maximum number of inner iterations to be used in the XSDRNPM calculation. The default value is 20. See Appendix B for a more detailed explanation.

30. **ICM** MAXIMUM NUMBER OF OUTER ITERATIONS FOR XSDRNPM. This is the maximum number of outer iterations to be used in the XSDRNPM calculation. The default value is 25. If the calculation reaches the outer iteration limit, a larger value should be used. See Appendix B for a more detailed explanation.

31. **EPS** OVERALL CONVERGENCE CRITERIA FOR XSDRNPM. This is used by XSDRNPM after each outer iteration to determine if the problem has converged. The default value of **EPS** is 0.00001. A value less than 0.00001 tightens the convergence criteria; a larger value loosens the convergence criteria.

32. **PTC** POINTWISE CONVERGENCE CRITERIA FOR XSDRNPM. This is the point flux convergence criteria used by XSDRNPM to determine if convergence has been achieved after an inner iteration. The default value for PTC is 0.000001. A smaller value tightens convergence; a larger value loosens it.

33. **BKL** BUCKLING FACTOR FOR XSDRNPM. A buckling factor should be used ONLY for a **MULTIREGION BUCKLEDSLAB** or **BUCKLEDCYL** problem. Because cylinders are assumed to be infinitely long and slabs are assumed to be infinite in both transverse directions, the analytic sequence may tend to overestimate the total flux for a finite system. A buckling correction can be used to approximate the leakage from the system in the transverse direction(s). The extrapolation distance factor, **BKL**, is defaulted to 1.420892.

34. **IUS** UPSCATTER SCALING FLAG for XSDRNPM. This option allows the use of upscatter scaling to accelerate the solution or force convergence. The default value is zero, in which case upscatter scaling is not used. **IUS**=1 facilitates upscatter scaling. Guidelines are not available to indicate when upscatter scaling is needed. Some problems will not converge with it, and some will not converge without it. See Appendix B for a more detailed explanation.

35. **DAN**(mm) DANCOFF FACTOR for the specified mixtures used in BONAMI and in the CENTRM 2REGION option. This value overrides the internally computed Dancoff factor used in the resonance correction for the specified mixture *mm*. The Dancoff data are entered in the form **DAN**(mm) = Dancoff factor. Note that the parentheses must be entered as part of the data, and the mixture number, mm, must be enclosed in the parentheses. See Appendix B for additional details. (Note: this is not to be confused with the DAN2PITCH parameter in CENTRMDATA)

36. **BAL** BALANCE TABLE PRINT FLAG for XSDRNPM. This allows control of the balance table print from **XSDRNPM**. The default value is **FINE**. **BAL**=**NONE** suppresses the balance table print. **BAL**=**ALL** prints all of the balance tables. **BAL**=**FINE** prints only the fine-group balance tables. See Appendix B for additional details.

37. **DY** FIRST TRANSVERSE DIMENSION for XSDRNPM. This is the first transverse dimension, in cm, used in a buckling correction to calculate the leakage normal to the principal calculation direction (the height of a slab or cylinder). It should only be entered if XSDRNPM is to create cell-weighted cross sections and/or calculate the eigenvalue of a cylinder or slab system of finite height for a **LATTICECELL** problem. **DY**= is defaulted to an infinite height, or is set to **DY** for a buckled **MULTIREGION** cell description. A value entered here overrides any buckling height value entered in the **MULTIREGION** data.

38. **DZ** SECOND TRANSVERSE DIMENSION for XSDRNPM. This is the second transverse dimension, in cm, used for a buckling correction for a slab of finite width. It should only be entered if XSDRNPM is to create cell-weighted cross sections and/or calculate the eigenvalue of a **LATTICECELL** slab of finite width. **DZ**= is defaulted to an infinite width, or is set to **DZ** for a buckled **MULTIREGION** slab cell of finite width. A value entered here overrides any buckling depth value entered in the **MULTIREGION** data.

39. **COF** DIFFUSION COEFFICIENT FOR TRANSVERSE LEAKAGE CORRECTIONS IN XSDRNPM. The default value is 3. The available options are as follows.

    COF=0 sets a transport-corrected cross section for each zone

    COF=1 use a spatially averaged diffusion coefficient for each zone

    COF=2 use a diffusion coefficient for all zones that is one-third of the diffusion coefficient determined from the spatially averaged transport cross section for all zones

    COF=3 use a flux and volume weighting across all zones

    See Appendix B or XSDRNPM Input/Output Assignments in the XSDRNPM chapter, 3$ array, variable **IPN** for more details.

40. **NT3** UNIT WHERE XSDRNPM WRITES THE WEIGHTED LIBRARY. If XSDRN does a weighting calculation, this is the unit number it uses to write the weighted library on (default is 3).

41. **NT4** UNIT WHERE XSDRNPM WRITES THE ANGULAR FLUXES. XSDRN writes the angular fluxes on this unit if it is non-zero (default is 16).

42. **ADJ** Adjoint mode flag for XSDRNPM. Set to 1 to cause XSDRNPM to solve the adjoint problem (default is 0).

43. **NTA** UNIT WHERE XSDRNPM WRITES THE ACTIVITIES. XSDRN writes the calculated activities on this unit if it is non-zero (default is 75).

44. **NBU** UNIT WHERE XSDRNPM WRITES BALANCE TABLES. If the balance tables file is to be saved, enter the unit number where it is to be written (default is 76).

45. **NTC** UNIT WHERE XSDRNPM WRITES THE DERIVED DATA. XSDRN writes the derived input data on this unit if it is non-zero (default is 73).

46. **NTD** UNIT WHERE XSDRNPM WRITES THE DATA FOR A SENSITIVITY ANALYSIS. XSDRN writes the data for a sensitivity analysis on this unit if it is non-zero (default is 0).

47. **FRD** UNIT WHERE XSDRNPM READS INPUT FLUX GUESS. If greater than 0, a flux guess will be read from this unit.

48. **FWR** UNIT WHERE XSDRNPM WRITES OUPUT FLUX. If greater than 0, the space-dependent multigroup scalar flux is written in binary format to this unit.

49. **WGT** CROSS SECTION WEIGHTING FLAG for XSDRNPM. The default is 0, not to perform cross section weighting. To turn on cross section weighting, a positive value should be entered. A value of 1 will weight the cross sections by nuclide; 2 will weight by mixture.

50. **ZMD**(iz) ZONE WIDTH MODIFIERs for an XSDRNPM search problem. This allows entering a zone width modifier for zone iz in the XSDRNPM problem description. The zone width data are entered in the following form:

    ZMD(iz)=modifier

    Note that the parentheses must be entered as part of the keyword. The zone number iz, to which the modifier is applied, must be enclosed in the parentheses. The modifier is entered after the equal sign. See the "Dimension Search Calculations" description in the XSDRNPM chapter for more information.

51. **INT**(iz) NUMBER OF MESH INTERVALS FOR ZONE IZ in XSDRNPM. The default is 0, which causes the number to be calculated. The data are entered in the following form:

    INT(iz)=number

Note that the parentheses must be entered as part of the keyword. The zone number iz, for which the number of intervals is specified, must be enclosed in the parentheses. The number of intervals is entered after the equal sign.

52. **KEF** DESIRED VALUE OF k_{EFF} for an XSDRNPM zone width search. The default value is 1.0. If it is desired to search for some other value, such as 0.9, then input it here.

53. **KFM** The first eigenvalue modifier used in an XSDRNPM search. This value is used to make the first change in the XSDRNPM search. The default value is -0.1. A user may sometimes need to change this to make the search converge.

54. **ID1** SCALAR FLUX PRINT CONTROL. The default value is -1, which suppresses printing the scalar fluxes in XSDRNPM. See the XSDRNPM Input/Output Assignments section in the XSDRNPM chapter, 2$ array, variable **ID1** for allowed values and corresponding actions.

55. **ISCT** ORDER OF SCATTERING for XSDRMPM. The default is 5 for all libraries.

56. **ICON** TYPE OF WEIGHTING (see Cross-Section Weighting section in the XSDRNPM chapter).

    INNERCELL - followed by integer N (zones in the cell). Cell weighting is performed over the N innermost regions in the problem. Nuclides outside these regions are not weighted.

    CELL - cell weighting

    ZONE - zone weighting

    REGION - region weighting

57. **IGMF** NUMBER OF GROUPS IN COLLAPSED LIBRARY. Enter number of groups after equal sign, followed by group lower energy boundaries (eV) in descending order.

58. **ITP** COLLAPSED OUTPUT FORMAT. The default is 0.

    0-19 - cross sections are written only in the AMPX weighted library formats on logical 3. A weighted library is always written when IFG=1.

    The various values of ITP (modulo 10) are used to select the different transport cross section weighting options mentioned earlier. The options are as follows:

    ITP = 0, 10, ... $\sqrt{(\psi_1^g + (DG\psi))^2}$

    ITP = 1 ,11, ... absolute value of current

    ITP = 2, 12, ... $DB^2\psi_g$ + outside leakage

    ITP = 3, 13, ... $\frac{\psi}{\Sigma_t^g}$

    ITP = 4, 14, ... $DB\psi_g$

    ITP = Other values are reserved for future development and should not be used.

59. **GAMMA_MT_LIST** LIST OF GAMMA 1D REACTIONS ASSOCIATED WITH INPUT. A list of 1-D gamma reactions to be included on a condensed library for later use gamma_mt_list= numberEntries mt1 mt2 ... mt_numberEntries.

60. **NEUTRON_MT_LIST** LIST OF NEUTRON 1D REACTIONS ASSOCIATED WITH INPUT. A list of 1-D neutron reactions to be included on a condensed library for later use neutron_mt_list= numberEntries mt1 mt2 ... mt_numberEntries.

61. **NEUTRON_2D_LIST** LIST OF NEUTRON 2D ARRAYS FOR THE MICRO LIBRARY. This list flags the finalizer to place 2-D arrays (currently MT 2, 4, 16) on the micro library for use in SAMS.
62. **ACTIVITY** Enter:

    IAZ (number of activities)

    IAI (calculate activities by zone or interval)

        0 – zone

        1 – interval

    LACFX (unit number to which activities are written)

    LAZ (IAZ sets of numbers consisting of the nuclide and process numbers for each activity)
63. **BAND** NUMBER OF REBALANCE BANDS for XSDRNPM (default is 1).
64. **IPRT** CROSS SECTION PRINT CONTROL. The default value is -2, which suppresses printing the cross sections in XSDRNPM. See XSDRNPM chapter, 2$ array, variable IPRT for allowed value, and corresponding actions.
65. **GRAIN_K** Flag to control execution of XSDRNPM after each grain calculation for a **DOUBLEHET** cell.
66. **SOURCE**(iz) ZONE SOURCE for an XSDRNPM fixed source problem. This allows entering a source spectrum for zone iz in the XSDRNPM problem description. The source spectrum data are entered in the following form:

    SOURCE(iz)= numEntries spectrum_grp_1 . . . spectrum_grp_numEntries

    Note that the parentheses must be entered as part of the keyword. The zone number, iz, to which the spectrum is applied, must be enclosed by the parentheses. The numEntries follows the equal sign and must be less than or equal to the number of energy groups for the problem. It is followed by numEntries numbers defining the spectrum for the first numEntries groups for zone iz. Groups not defined are set to zero. The spectrum applies uniformly to zone iz. A different spectrum may be entered for different zones.
67. **END MORE** Terminate the optional parameter data.

7.1.3.9 Optional CENTRM DATA parameter data

The CENTRM DATA block defines input parameter values for the CENTRM, PMC and CRAWDAD modules. XSProc defines default values for these parameters which are adequate for most applications. If all default values are acceptable, this section of input data can be omitted. The CENTRM DATA block applies only to the unit cell immediately preceding it. CENTRM DATA placed prior to all unit cell data applies to all materials not listed in any unit cell. Parameter values are assigned by entering the words **CENTRM DATA** followed by the desired keyword parameters and their associated values. One or more parameters can be entered in any order. There should not be a blank between the parameter name and the equal sign. Each parameter specification must be separated from the rest by at least one blank. For example,

CENTRM DATA ISN=16 **PTC**=0.0008 **N1D**=1 **END CENTRM DATA**

A description of CENTRM DATA parameters is given below.

1. **CENTRM DATA** These words, followed by one or more blanks, are entered ONLY if optional parameter data are to be entered. They must precede all other optional parameter data. Entries 2 through 42 can be entered in any order.

2. **ISN** ORDER OF SN ANGULAR QUADRATURE FOR CENTRM. SN Quadrature sets are geometry-dependent quantities. Default value for **ISN** is 6 (only used for **NFST** and **NTHR**=0; and **NPXS**=1).

3. **ISCT** LEGENDRE POLYNOMIAL P_N ORDER OF SCATTERING. These are used to determine the number of moments calculated for the scattering cross sections. Default value is 0 for 2-D MoC option and 1 for 1-D S_n, which have been found adequate for nearly all cases.

4. **IIM** MAXIMUM NUMBER OF INNER ITERATIONS. This is the maximum number of inner iterations for Sn transport calculations in CENTRM. Default value is 10.

5. **IUP** MAXIMUM NUMBER OF OUTER ITERATIONS IN THERMAL RANGE. This is the maximum number of outer iterations used to converge PW flux changes caused by upscattering in the thermal range. Default value is 3. More iterations (~ 15) may be required for higher accuracy in some cases.

6. **NFST** FAST RANGE MULTIGROUP CALCULATION OPTION, E > **DEMAX**. This determines what type of calculation is done above **DEMAX**. The options are (0) S:sub:*N*, (1) diffusion theory, (2) homogenized infinite medium, (3) zonewise infinite medium, or (6) 2D MoC lattice cell [NOTE: NFST=4,5 are deprecated]. Default value is 0 (S_N).

7. **NTHR** THERMAL RANGE MULTIGROUP CALCULATION OPTION, E < **DEMIN**. This determines what type of calculation is done below **DEMIN**. The options include (0) S:sub:*N*, (1) diffusion, (2) homogenized infinite medium, (3) zonewise infinite medium, or (6) 2-D MoC lattice cell [NOTE: NTHR=4,5 are deprecated]. Default value is 0 (S_N).

8. **NPXS** POINTWISE RANGE MULTIGROUP CALCULATION OPTION, **DEMIN** < E < **DEMAX**. This determines what type of calculation is done between **DEMIN** and **DEMAX**. The options include (0) MG calculation, (1) 1-D S_N, (2) collision probability, (3) homogenized infinite medium, (4) zonewise infinite medium, (5) two-region, or (6) 2-D MoC lattice cell. Default value is 1 (S_N), except for square-pitch LATTICECELL where the default is 6 (2D MoC).

9. **ISVAR** LINEARIZATION OPTION. This determines if the MG source and/or the cross sections are linearized in CENTRM calculations. Options for linearizing are (0) neither, (1) source, (2) cross section, or (3) both. Default value is 3.

10. **ISCTI** LEGENDRE POLYNOMIAL P_N ORDER OF SCATTERING IN THE INELASTIC RANGE. These are used to determine the number of moments calculated for the inelastic scattering cross sections. Default value is 0, isotropic.

11. **NMF6** INELASTIC FLAG. This determines if inelastic data are used. The options are to include (-1) no inelastic data, (0) discrete inelastic data, and (1) discrete inelastic and continuum. Default value is -1. Use of **NMF6**=1 is not recommended due to long running times.

12. **IPRT** MIXTURE CROSS-SECTION OUTPUT OPTION. This determines the output of cross section. The options include (-3) none, (-2) output macro PW cross sections to file "_centrm.pw.macroxs", (-1) 1-D MG cross sections, (N) P_0 to P_N MG 2-D matrices. Default value is -3, none.

13. **ID1** FLUX EDIT OPTION. This option determines the output of flux energy spectra. The options are (-1) none, (0) print MG fluxes, (1) also print MG flux moments, (2) save CE fluxes on output file, "_centrm.pw_flux". Default value is -1.

14. **KERNEL** BOUND KERNELS. This indicates use of CENTRM PW thermal kernel data [S(α,:math:*beta*)] for bound nuclides if **KERNEL**=1. If **KERNEL**=0, all thermal kernels are treated as free gas; Default is 1, use bound scattering kernels if available.

15. **IPBT** PRINT GROUP SUMMARY TABLES. Group summary tables for each zone are printed in CENTRM if greater than 0. Default is 0. Balance ratios are not computed in thermal groups or for MoC option.

16. **IPN** GROUP DIFFUSION COEFFICENT. Used for DB^2 loss term. See XSDRNPM chapter for more information. Default is 2.

17. **IXPRT** PRINT OPTION FOR CENTRM. This value is >0 if more information is printed to output. Default value is 0, minimum output.

18. **MLIM** MASS VALUE RESTRICTION ON ORDER OF SCATTERING. Nuclides with mass ratios greater than **MLIM** are limited to a **NLIM** order of scattering. Default value is 100.

19. **NLIM** ORDER OF SCATTERING RESTRICTION. This is the limiting order of scattering for all nuclides with mass ratios greater than **MLIM**. Default value is 0.

20. **EPS** INTEGRAL CONVERGENCE CRITERIA. This is used by CENTRM after each outer iteration to determine if the problem has converged. Default value is 0.001. A value less than 0.0001 tightens the convergence criteria; a larger value loosens the convergence criteria.

21. **PTC** POINTWISE CONVERGENCE CRITERIA. This is the point flux convergence criteria used by CENTRM to determine if convergence has been achieved after an inner iteration. Default value is 0.0001. A smaller value tightens convergence; a larger value loosens it.

22. **B2** MATERIAL BUCKLING FACTOR (cm^{-2}). This is used with a buckled system. If a buckled system is specified for a unit cell, the code will use this value. Default value is 0.0.

23. **DEMIN** LOWEST ENERGY OF POINTWISE FLUX CALCULATION. This value is the lowest energy (eV) for which CENTRM calculates PW fluxes. Default is 0.001 eV.

24. **DEMAX** HIGHEST ENERGY OF POINTWISE FLUX CALCULATION. This value is the highest energy (eV) for which CENTRM calculates PW fluxes. Default is 20,000.0 eV, which encompasses the resolved resonance range of all actinides. It is recommended that DEMAX be <500 keV.

25. **TOLE** CENTRM PW THINNING TOLERANCE. This is the tolerance used to thin the PW material cross sections after they are mixed. Default value is 0.001.

26. **FLET** FRACTIONAL LETHARGY CONSTRAINT. This is the maximum lethargy difference between points in the flux solution energy mesh. Smaller values increase the number of energy points. Default value is 0.1.

27. **DAN2PITCH** CENTRM DANCOFF FACTOR SEARCH. Fuel Dancoff factor to search for a Dancoff-equivalent pitch used in the CENTRM cell calculation. Only applicable in LATTICECELL and DOUBLEHET cases with fuel in center region, with SN or MoC transport solvers. Default is 0, which indicates no pitch modification. NOTE! This option should not be used to enter Dancoff factors for the CENTRM *2REGION* transport option-use EDAN(m) array in **MOREDATA** for these values.

28. **MRANGE** PMC GROUP CROSS-SECTION PROCESSING RANGE. This option determines the range over which the group cross sections will be processed. The options are (0) compute new group cross section over the PW range, (1) over the resolved resonance range of each nuclide, or (2) over the PW flux range (**DEMAX** to **DEMIN**). Default value is 2.

29. **N2D** PMC ELASTIC MATRIX PROCESSING FLAG. This option determines how MG P_N elastic scattering matrices are obtained. Options are (-2) perform operations in both (-1) and (2); (-1) compute P0 self-scatter, then renormalize matrix to shielded 1-D elastic values; (0) normalize original scatter

matrix to shielded 1-D elastic values; (1) compute new P_N moments of elastic matrix using scalar flux and S-wave kinematics for both thermal and epithermal energy ranges; or (2) use flux-moments to compute "consistent PN" correction for diagonal elements of elastic P_N components. Default value is -1. For unit cell calculations in reactor lattices, option -2 may improve results. NOTE: option 0 is always used in thermal range except for option 1.

30. **IXTR3** PMC P_N ORDER FLAG. This option determines the maximum order of Legendre moments to be retained on output MG library. The default is 5; i.e., retain scattering moments up to P_5 if available on the input MG library. If (-1) is entered, all elastic moments on the MG library are included.

31. **NPRT** PMC PRINT FLAG. This option determines what is printed to output. The options include (-1) minimum output, (0) standard output, (1) print 1-D cross sections, (2) print both 1-D and 2-D cross sections. Default value is -1, minimum output.

32. **NWT** PMC MULTIGROUP SPATIAL-WEIGHTING FLAG. This option determines if the MG data are (0) zone-weighted or (1) cell-weighted. Default value is 0.

33. **MTT** PMC MT PROCESSING FLAG. This option determines if reaction MTs are processed individually or treat dependencies explicitly. If **MTT**=0 all MTs are processed independently; if **MTT**=1, all MTs are processed except 1, 27, and 101. These are then computed as follows: MT101 = sum MT102 - 114, MT27 = MT18 + MT101, MT1 = MT2 + MT4 + MT16 + MT17 + MT27. Default value is 1.

34. **N1D** PMC WEIGHTING FUNCTION FLAG. This is used to determine if (0) flux weighting or (1) current weighting is used to collapse the cross sections. Default value is 0, flux weighting.

35. **PMC_DILUTE** PMC DILUTE BACKGROUND CROSS SECTION. The background cross section σ_0 value above which nuclide cross sections are not processed in PMC but the BONAMI cross sections are used instead. No resonance shielding corrections are performed for materials with background cross sections greater than *pmc_dilute*. Higher values of *pmc_dilute* result in more nuclides being processed. The default value is 1.0E10.

36. **MTOUT** PW REACTION TYPES. Reactions included by CRAWDAD on PW library for MG processing in PMC: (0) all; (1) output only MTs 1, 2, 4, 102, 18, 452, 455, 456 (and 107 for ^{10}B or ^{7}Li); (2) all from option (1) and all inelastic MTs and 16. Default is **MTOUT**=1 for **NMF6**=-1 and **MTOUT**=2 for **NMF6**>-1.

37. **IBR** CENTRM RIGHT BOUNDARY TYPE. Type of boundary condition on right boundary of unit cell for CENTRM **LATTICECELL** calculations. See allowable IBR values in CENTRM. Default is white (**IBR** = 3) for 1D SN; 2D MoC transport option always uses reflected.

38. **IBL** CENTRM LEFT BOUNDARY TYPE. Same as **IBR**, but for left boundary. Default is reflected (**IBL** = 1).

39. **ALUMP** MASS LUMPING FRACTION. A value in range [0.0, 1.0] indicates fractional mass lumping criterion for CENTRM. Value of 0 indicates no lumping applied. For example, **ALUMP**=0.3 means that materials are combined into one or more lumps such that their masses are within +/-30% of the effective lump mass, while preserving the slowing-down power. This approximation reduces execution time. Default value is 0.2.

40. **PMC_OMIT** PMC NUCLIDES SKIPPED. PMC normally processes problem-dependent (e.g., self-shielded) MG cross sections for all materials. If **PMC_OMIT**=1, processing is only performed for materials contained in fuel mixtures. Default value is 0 (all materials processed).

41. **PXSMEM** CENTRM PW DATA STORAGE. Option to store PW data in memory or in external file during centrm execution. If **PXSMEM**=1, PW cross section data are stored by group in external scratch file during CENTRM calculation; if **PXSMEM**=0 (default value), all PW cross sections are kept in memory.
42. **MOCMESH** CENTRM MOC MESH OPTION. Pre-defined space mesh intervals for CENTRM MoC calculation: 0=>coarse mesh (1 interval per zone); 1=>regular mesh (4 intervals in fuel, 2 in moderator, 1 in others); 2=> fine mesh (8 in fuel, 4 in moderator, 1 in others). Default=0.
43. **MOCRAY** CENTRM MOC RAY SPACING. Distance between characteristic rays in CENTRM MoC calculation. Default=0.02.
44. **MOCPOL** CENTRM NUMBER OF MOC POLAR ANGLES. Allowable values are 2, 3, 4. Default=3 (only used for **NPXS**=6).
45. **MOCAZI** CENTRM NUMBER OF MOC AZIMUTHAL ANGLES. Allowable values are 2–16. Default=8 (only used for **NPXS**=6).
46. **MOCZONE_INT** CENTRM MOC MESH BY ZONE. User-defined mesh intervals by zone; e.g., moczone_int(1)=5 defines five intervals for zone 1; zero value means not used. This overrides the predefined meshs described by **MOCMESH**
47. **ISRC** CENTRM SOURCE TYPE. CENTRM can use a fission-spectrum source (isrc=1), an input source spectrum (isrc=0), or a combination(isrc=3) for transport. Default=1.
48. **XNF** CENTRM SOURCE NORMALIZATION. The integrated source (fission-spectrum and/or fixed source spectrum) is normalized to this value. Default=1.0.
49. **ITERP** CRAWDAD TEMPERATURE INTERPOLATION METHOD. Method to use for CE cross section interpolation 0=>combination of square-root(T) and finite-difference; 1=>only square-root(T); 2=> only finite-difference. Default is 0.
50. **END CENTRM** The word **END** is entered to terminate the optional parameter data. A label can be associated with this **END**. The label can be as long as 12 characters but must be preceded by a single blank. If this **END** is entered without a label, it must not begin in column 1. At least two blanks must follow this entry.

7.1.4 APPENDICES

7.1.4.1 XSProc: Standard Composition Examples

Standard composition fundamentals

The standard composition specification data are used to define mixtures using standardized engineering data entered in a free-form format. The XSProc uses the standard composition specification data and information from the Standard Composition Library to provide number densities for each nuclide of every defined mixture according to Eq. (7.1.1).

$$NO = \frac{RHO \times AVN \times C}{AWT}, \tag{7.1.1}$$

where

NO is the number density of the nuclide in atoms/b-cm,

RHO is the actual density of the nuclide in g/cm^3,

AVN is Avogadro's number, $6.02214199 \times 10^{23}$, in atoms/mol,

C is a constant, 10^{-24} cm^2/b,

AWT is the atomic or molecular weight of the nuclide in g/mol.

The actual density, RHO, is defined by

$$RHO = ROTH \times VF \times WGTF, \tag{7.1.2}$$

where

RHO is the actual density of the standard composition in g/cm^3,

ROTH is either the specified density of the standard composition or the theoretical density of the standard composition in g/cm^3,

VF is a density multiplier compatible with ROTH as defined by Eq. Eq. (7.1.3),

WGTF is the weight fraction of the nuclide in the standard composition. This value is automatically obtained by the code from the Standard Composition Library. WGTF is 1.0 for a single nuclide standard composition.

$$VF = DFRAC \times VFRAC, \tag{7.1.3}$$

where

VF is the density multiplier,

DFRAC is the density fraction,

VFRAC is the volume fraction.

To illustrate the interaction between ROTH and VF, consider an Inconel having a density of 8.5 g/cm^3. It is 7.0% by weight iron, 15.5% chromium, and 77.5% nickel. The Inconel occupies a volume of 4 cm^3.

Method 1:

To describe the iron, enter 8.5 for ROTH and 0.07 for VF.

To describe the chromium, enter 8.5 for ROTH and 0.155 for VF.

To describe the nickel, enter 8.5 for ROTH and 0.775 for VF.

Method 2:

Do not enter the density, and by default the theoretical density of each component will be used for ROTH. DFRAC will be the ratio of the specified density to the theoretical density. The specified density of each component is the density of the Inconel × the weight fraction of that component.

Thus, the density of the iron is $8.5 \times 0.07 = 0.595$ g/cm^3

chromium is $8.5 \times 0.155 = 1.318$ g/cm^3

nickel is $8.5 \times 0.775 = 6.588$ g/cm^3.

To calculate DFRAC, the theoretical density of each material must be obtained from the table *Elements and special nuclide symbols* in the STDCMP chapter. These values are

7.86 g/cm^3 for iron

8.90 g/cm^3 for nickel

7.20 g/cm^3 for chromium

The DFRAC entered for the iron is 0.595/7.86 = 0.0757

for the nickel is 1.318/8.90 = 0.1481

for the chromium is 6.588/7.20 = 0.9163.

Since there are no volumetric corrections, VFRAC is 1.0 and the values of DFRAC are entered for VF.

Method 3:

Assume the Inconel, which occupies 4 cm^3, is to be spread over a volume of 5 cm^3. Then the volume fraction, VFRAC, is 4 cm^3/5 cm^3 = 0.8 and can be combined with the density fraction, DFRAC, to obtain the density multiplier, VF.

To describe the iron, enter 8.5 for ROTH and 0.07 × 0.8 = 0.056 for VF

To describe the chromium, enter 8.5 for ROTH and 0.155 × 0.8 = 0.124 for VF

To describe the nickel, enter 8.5 for ROTH and 0.775 × 0.8 = 0.620 for VF.

Alternatively, the volume fraction can be applied to the density before it is entered. Then the ROTH can be entered as 8.5 g/cm^3 × 0.8 = 6.8 g/cm^3, and DFRAC is entered for the density multiplier, VF.

To describe the iron, enter 6.8 for ROTH and 0.07 for VF

for chromium, enter 6.8 for ROTH and 0.155 for VF

for nickel, enter 6.8 for ROTH and 0.775 for VF.

Method 4:

Assume the Inconel, which occupies 4 cm^3, is to be spread over a volume of 5 cm^3. Then the volume fraction, VFRAC, is 4 cm^3/5 cm^3 = 0.8. Do not enter the density, and by default the theoretical density of each component will be used for ROTH.

VF is then entered as the product of VFRAC and DFRAC according to Eq. . The specified density of each component is the density of the Inconel × the weight fraction of that component.

Thus, the density of the iron is 8.5 × 0.07 = 0.595 g/cm^3

Chromium is 8.5 × 0.155 = 1.318 g/cm^3

Nickel is 8.5 × 0.775 = 6.588 g/cm^3.

To calculate DFRAC, the theoretical density of each material must be obtained from Table 7.2.3. These values are

7.86 g/cm^3 for iron 8.90 g/cm^3 for nickel 7.20 g/cm^3 for chromium.

Then DFRAC for the iron is 0.595/7.86 = 0.0756

for nickel is 1.318/8.90 = 0.1481

for chromium is 6.588/7.20 = 0.9150.

Then VF is DFRAC × VFRAC

VF for the iron is 0.0757 × 0.8 = 0.0606

for nickel is 0.1481 × 0.8 = 0.1185

for chromium is 0.9150 × .8 = 0.7320.

Basic standard composition specifications

EXAMPLE 1. Material name is given. Create a mixture 3 that is Plexiglas.

Since no other information is given, the information on the Standard Composition Library can be assumed to be adequate. Therefore, the only data to be entered are the standard composition name and the mixture number

```
PLEXIGLAS  3  END
```

EXAMPLE 2. Material name and density (g/cm^3) are given.

Create a mixture 3 that is Plexiglas at a density of 1.15 g/cm^3. Since no other data are specified, the defaults from the Standard Composition Library will be used. Therefore, the only data to be entered are the standard composition name, the mixture number, and the density.

```
PLEXIGLAS  3  DEN=1.15  END
```

EXAMPLE 3. Material name and number density (atoms/b-cm) are given. Create a mixture 2 that is aluminum having a number density of 0.060244.

```
AL  2  0  .060244  END
```

EXAMPLE 4. Material name, density (g/cm^3) and isotopic abundance are given.

Create a mixture 1 that is uranium metal at 18.76 g/cm^3 whose isotopic composition is 93.2 wt % ^{235}U, 5.6 wt % ^{238}U, and 1.0 wt % ^{234}U, and 0.2 wt % ^{236}U. This example uses the DEN= keyword to enter the density and define the standard composition. Example 5 demonstrates another method of defining the standard composition.

```
URANIUM   1  DEN=18.76 1 300  92235  93.2  92238  5.6  92234  1.0  92236  0.2  END
```

EXAMPLE 5. Material name, density (g/cm^3) and isotopic abundance are given.

Create a mixture 7 defining B_4C with a density of 2.45 g/cm^3. The boron is 40 wt % ^{10}B and 60 wt % ^{11}B. This example utilizes the **DEN**= keyword. Example 6 illustrates an alternative description.

```
B4C 7  DEN=2.45  1.0 300  5010  40.0  5011  60.0  END
```

EXAMPLE 6. Material name, density (g/cm^3) and isotopic abundance are given.

Create a mixture 7 defining B_4C with a density of 2.45 g/cm^3. The boron is 40 wt % ^{10}B and 60 wt % ^{11}B. This example incorporates the known density into the density multiplier, *vf*, rather than using the **DEN**= keyword. The default density for B_4C given in the COMPOUNDS table in the SCL section 7.2 is equal to 2.52 g/cm^3.

```
B4C  7  0.9722 300  5010  40.0  5011  60.0  END
```

Note: In the above examples, the actual density is input for materials containing enriched multi-isotope nuclides (uranium in Examples 4 and 5 and boron in Examples 6 and 7). The default density should never be used for enriched materials, especially low atomic mass neutron absorbers such as boron and lithium. The default density is a fixed value for nominal conditions and naturally occurring distributions of isotopes. Use of the default density for enriched materials will likely result in incorrect number densities

---

User-defined (arbitrary) chemical compound specifications

The user-defined compound option allows the user to specify materials that are not found in the Standard Composition Library and can be specified by the number of atoms of each element or isotope that are contained in the molecule. To define a user-defined compound, the first four characters of the standard composition component name must be **ATOM**. The remaining characters of the standard composition component name are chosen by the user. The maximum length of the standard composition name is 16 characters. All the information that would normally be found in the Standard Composition Library must be entered in the user-defined compound specification. Sect. 7.1.3.3 contains data input details for arbitrary compounds.

EXAMPLE 1. Density and chemical equation are given.

Create a mixture 3 that is a hydraulic fluid, C_2H_6SiO, with a density of 0.97 g/cm^3. The input data for this user-defined compound are given below:

```
ATOM     3  0.97  4 6000 2 1001 6 14000 1 8000 1 END
```

EXAMPLE 2. Density and chemical equation are given. Create a mixture 7, TBP, also known as phosphoric acid tributyl ester or tributylphosphate, $(C_4H_9O)_3PO$, having a density of 0.973 g/cm^3.

```
ATOMtbp          7  0.973  4 1001 27 6000 12 8016 4 15031 1 end
```

User-defined (arbitrary) mixture/alloy specifications

The user-defined compound or alloy option allows the user to specify materials that are not found in the Standard Composition Library and are defined by specifying the weight percent of each element or isotope contained in the material. To define a user-defined weight percent mixture, the first four characters of the standard composition component name must be *wtpt*. The remaining characters of the standard composition component name are chosen by the user. The maximum length of the standard composition name is 16 characters. All the information that would normally be found in the Standard Composition Library must be entered in the arbitrary mixture/alloy specification. Sect. 7.1.3.3 contains data input details for user-defined compounds.

EXAMPLE 1. Density and weight percents are given.

Create a mixture 5 that defines a borated aluminum that is 2.5 wt % natural boron. The density of the borated aluminum is 2.65 g/cm^3.

```
SOLUTION MIX=2 RHO[UO2(NO3)2]=415 92235 92.6 92238 5.9 92234 1 92236
0.5 MASSFRAC[HNO3]=6.339-6 TEMPERATURE=293 END SOLUTION
```

EXAMPLE 2. Density, weight percents, and isotopic abundance are given.

Create a mixture 5 that defines a borated aluminum that is 2.5 wt % boron. The boron is 90 wt % ^{10}B and 10 wt % ^{11}B. The density of the borated aluminum is 2.65 g/cm^3. The minimum generic input specification for this arbitrary material is

```
WTPTBAL  5  2.65  2  5000 2.5  13027 97.5  1  293  5010 90.  5011 10. END
```

Fissile solution specifications

Solutions of fissile materials are available in the XSProc. A list of the available solution salts and acids is given in the table *Available fissile solution components* in Sect. 7.2.3. When the XSProc processes a solution, it breaks the solution into its component parts (basic standard composition specifications) and uses the solution density to calculate the volume fractions.

EXAMPLE 1. Fuel density, excess acid and isotopic abundance are given.

> Create a mixture 2 that is a highly enriched uranyl nitrate solution with 415 g/L and 0.39 mg of excess nitrate per gram of solution. The uranium isotopic content is 92.6 wt % ^{235}U, 5.9 wt % ^{238}U, 1.0 wt % ^{234}U, and 0.5 wt % ^{236}U. The temperature is 293 Kelvin.

```
SOLUTION MIX=2 RHO[UO2(NO3)2]=415 92235 92.6 92238 5.9 92234 1 92236
0.5 MASSFRAC[HNO3]=6.339-6 TEMPERATURE=293 END SOLUTION
```

where

> The molecular weight of NO_3 is 62.0049 g/mole, of H is 1.0078 g/mole, so the grams of excess H per gram of solution is 1.0078 / 62.0049 × (0.39 mg/g) × (1 g/1000 mg) = 6.339 × 10^{-6}.

Combinations of standard composition materials to define a mixture

Frequently more than one standard composition is required to define a mixture. This section contains such examples.

EXAMPLE 1. Boral from B_4C and Aluminum.

> Create a mixture 6 that is Boral, 15 wt % B_4C and 85 wt % Al, having a density of 2.64 g/cm^3. Natural boron is used in the B_4C. Note that Example 2 demonstrates the use of the keyword **DEN=** to enter the density of the mixture and avoid having to look up the theoretical density from the table *Isotopes in standard composition library,* in the section 7.2.2, and calculate the density multiplier (VF)

```
B4C  6  0.1571  END
 AL  6  0.8305  END
```

EXAMPLE 2. Boral from B_4C and Aluminum.

> This is the same problem as Example 1 using a different method of specifying the input data. Create a mixture 6 that is Boral, 15 wt % B_4C and 85 wt % Al, having a density of 2.64 g/cm^3. Natural boron is used in the B_4C.

```
B4C  6  DEN=2.64  0.15  END
 AL  6  DEN=2.64  0.85  END
```

EXAMPLE 3. Boral from Boron, Carbon, and Aluminum.

> If neither Boral nor B_4C were available in the Standard Composition Library, Boral could be described as follows:
>
> Create a mixture 2 that is Boral composed of 35 wt % B_4C and 65 wt % aluminum with an overall density of 2.64 g/cm^3. The boron is natural boron.

vf is the density multiplier. (The density multiplier is the ratio of actual to theoretical density.) From the Standard Composition Library chapter, table *Isotopes in standard composition library*, the theoretical density of aluminum is 2.702 g/cm^3; boron is 2.37 g/cm^3; and carbon is 2.1 g/cm^3. The density multiplier, *vf*, for Al is (0.65)(2.64)/2.702 = 0.63509. The isotopic abundances in natural boron are known to have some variability. Here it is assumed that natural boron is 18.4309 wt % ^{10}B at 10.0129 amu and 81.5691 wt % ^{11}B at 11.0096 amu. C is 12.000 amu.

Convert the weight percents to atom percents for the natural boron where w denotes weight fraction, a denotes atom fraction, and M denotes atomic mass:

$$w_{B10} = 0.184309 \equiv \frac{a_{B10}M_{B10}}{a_{B10}M_{B10} + a_{B11}M_{B11}} = \frac{a_{B10}(10.0129)}{a_{B10}(10.0129) + (1 - a_{B10}))(11.0093)} \tag{7.1.4}$$

Solving for a_{B10} gives:

$$[a_{B10}=0.184309 = \frac{(0.184309)\ (11.0093)}{(0.184309)\ (11.0093)\text{-}(0.184309)\ (10.0129)\text{+}(10.0129)} = 19.900 \tag{7.1.5}$$

Therefore the atom percent of ^{11}B is, aB_{11} = 80.1 a%.

Similarly, the mass of the B_4C molecule is

$$[(0.199 \times 4 \times 10.0129) + (0.801 \times 4 \times 11.0093) + (12.000)] = 55.24407 \text{ amu.}$$

The mass of the boron is (55.24407 - 12.000) = 43.24407 amu.

The *vf* of boron would be $\left(\frac{43.24407}{55.24407}\right)\left(\frac{(0.35)(2.64)}{2.37}\right) = 0.30519$

The *vf* of C would be

$$\left(\frac{12.0000}{55.24407}\right)\left(\frac{(0.35)(2.64)}{2.1}\right) = 0.09558 \tag{7.1.6}$$

$$\left(\frac{12.000}{55.25045}\right)\left[\frac{(0.35)(2.64)}{2.30}\right] = 0.08725 \tag{7.1.7}$$

The standard composition input data for the Boral follows:

```
AL     2       0.63509 END
BORON  2       0.30519 END
C      2       0.09558 END
```

EXAMPLE 4. Boral from ^{10}B, ^{11}B, Carbon, and Aluminum.

Create a mixture 2 that is Boral composed of 35 wt % B_4C and 65 wt % aluminum. The Boral density is 2.64 g/cm^3. The boron is natural boron.

vf is the density multiplier. Use 0.63581 for AL and 0.08725 for C as explained in Example 3 above. From the Standard Composition Library chapter, *Isotopes in standard composition library* table, the theoretical density of ^{10}B is 1.00 g/cm^3 and ^{11}B is 1.00 g/cm^3. As computed in Example 3, the mass of the B_4C molecule is 55.25045 amu, and the boron is 19.764 atom % ^{10}B and 80.236 atom % ^{11}B. The mass of ^{10}B is 10.0129 amu and the ^{11}B is 11.0096. Thus, the *vf* of ^{10}B is

$$\left(\frac{(4)(0.199)(10.0129)}{55.24407}\right)\left(\frac{(0.35)(2.64)}{1.0}\right) = 0.13331\ . \tag{7.1.8}$$

The *vf* of ^{11}B is

$$\left(\frac{(4)(0.801)(11.0093)}{55.24407}\right)\left(\frac{(0.35)(2.64)}{1.0}\right) = 0.58998\ . \tag{7.1.9}$$

The standard composition input data for the Boral are given as

```
AL     2        0.63509 END
B-10   2        0.13331 END
B-11   2        0.58998 END
C      2        0.09558 END
```

EXAMPLE 5. Specify all of the number densities in a mixture.

Create a mixture 1 that is vermiculite, defined as

hydrogen at a number density of 6.8614-4 atoms/b-cm

oxygen at a number density of 2.0566-3 atoms/b-cm

magnesium at a number density of 3.5780-4 atoms/b-cm

aluminum at a number density of 1.9816-4 atoms/b-cm

silicon at a number density of 4.4580-4 atoms/b-cm

potassium at a number density of 1.0207-4 atoms/b-cm

iron at a number density of 7.7416-5 atoms/b-cm

In this example we use the 2nd syntax option described in Sect. 7.1.3.3, in which the 3rd entry must be 0. The standard composition input data for the vermiculite are given below:

```
H    1          0        6.8614-4         END
O    1          0        2.0566-3         END
MG   1          0        3.5780-4         END
AL   1          0        1.9816-4         END
SI   1          0        4.4580-4         END
K    1          0        1.0207-4         END
FE   1          0        7.7416-5         END
```

Combinations of user-defined compound and user-defined mixture/alloy to define a mixture

Mixtures can usually be created using only basic standard composition specifications. Occasionally, it is convenient to create two or more user-defined materials for a given mixture. This procedure is demonstrated in the following example.

EXAMPLE 1. Specify Boral using a user-defined compound and user-defined mixture/alloy.

Create a mixture 6 that is Boral, 15 wt % B_4C and 85 wt % Al, having a density of 2.64 g/cm^3. Natural boron is used in the B_4C. Boral can be described in several ways. For demonstration purposes, it will be described as a combination of a user-defined compound and user-defined mixture/alloy. This is not necessary, because both B_4C and Al are available as standard compositions. A method of describing the Boral without using user-defined compounds or user-defined mixtures/alloys is given in Examples 1 and 2 of Sect. 7.1.4.1.6. The minimum generic input specifications for this user-defined compound and alloy are

```
ATOM-B4C   6       2.64    2       5000            4       6012    1       0.15    END
WTPT-AL    6       2.64    1       13027           100.0           0.85    END
```

Combinations of solutions to define a mixture

This section demonstrates the use of more than one solution definition to describe a single mixture. The assumptions used in processing the cross sections are likely to be inadequate for solutions of mixed oxides of uranium and plutonium. Therefore, this section is given purely for demonstration purposes.

EXAMPLE 1. Solution of uranyl nitrate and plutonium nitrate.

Note that the assumptions used in processing the cross sections are likely to only be adequate for CENTRM/PMC calculations of mixed-oxide solutions. This example is given purely for demonstration purposes. Create a mixture 1 consisting of a mixture of plutonium nitrate solution and uranyl nitrate solution. The specific gravity of the mixed solution is 1.4828. The solution contains 325.89 g (U + Pu)/L soln. The acid molarity of the solution is 0.53. In this solution 77.22 wt % of the U+Pu is uranium. The isotopic abundance of the uranium is 0.008% ^{234}U, 0.7% ^{235}U, 0.052% ^{236}U, and 99.24% ^{238}U. The isotopic abundance of the plutonium is 0.028% ^{238}Pu, 91.114% ^{239}Pu, 8.34% ^{240}Pu, 0.426% ^{241}Pu, and 0.092% ^{242}Pu. Note that a single quote in the first column indicates a comment line in SCALE input.

```
'   Uranium density of 77.22% of 325.89 g/L
SOLUTION  MIX=1  RHO[UO2(NO3)2]=251.65  92234 .008 92235 .700 92236 .052
                                        92238 99.240
'   Plutonium density if 22.78% of 325.89 g/L
                 RHO[PU(NO3)4]=74.24  94238 .028 94239 91.114 94240 8.34
                                      94241 .426 94242 .092
'   Acid molarity is 0.53 M
                 MOLAR[HNO3]=0.53
'   Specifying the density over specifies the problem, which means the solution may
'   not be in thermodynamic equilibrium.  The specification below adds about 0.3%
'   extra hydrogen to the problem
                 DENSITY=1.4828
END SOLUTION
```

Combinations of basic and user-defined standard compositions to define a mixture

EXAMPLE 1. Burnable poison from B_4C and Al_2O_3.

Create a mixture 6 that is a burnable poison with a density of 3.7 g/cm^3 and composed of Al_2O_3 and B_4C. The material is 1.395 wt % B_4C. The boron is natural boron. This material can be easily specified using a combination of user-defined material to describe the Al_2O_3 and a simple standard composition to define the B_4C. The minimum generic input specification for this user-defined material and the standard composition are

The density multiplier of the B_4C is the density of the material times the weight percent, divided by the theoretical density of B_4C [$(3.7 \times 0.01395)/2.52$] or 0.02048; the density multiplier of the Al_2O_3 is 1.0 - 0.01395 or 0.98605 (the theoretical density of B_4C was obtained from *Isotopes in standard composition library* table in the STDCMP chapter).

The input data for the burnable poison are given below:

```
ATOM-AL2O3  6  3.70  2  13027  2  8016  3  0.98605  END
B4C  6  2.048-2  END
```

The B_4C input can be specified using the **DEN**= parameter as shown below:

```
ATOM-AL2O3  6  3.70  2  13027  2  8016  3  0.98605  END
B4C  6  DEN=3.7  0.01395  END
```

The fraction of B_4C in the mixture is ((3.7 × 0.01395)/2.52) = 0.02048. The fraction of Al_2O_3 in the mixture is 1.0 - 0.02048 = 0.979518. The density of the Al_2O_3 can be calculated as shown below.

$$F_{Al_2O_3} * Density_{Al_2O_3} + 0.020482 * 2.52 = 3.7 = Density \ \ of \ \ the \ \ mixture$$

$$Density_{Al_2O_3} = \frac{3.7 - 0.020482 * 2.52}{0.979518} = 3.72467$$

Input data using the density of Al_2O_3 are given below:

```
ATOM-AL2O3  6  3.72467  2  13027  2  8016  3  END
B4C  6  2.048-2  END
```

EXAMPLE 2. Borated water from H_3BO_3 and water.

Create a mixture 2 that is borated water at 4350 parts per million (ppm) by weight, resulting from the addition of boric acid, H_3BO_3 to water. The density of the borated water is 1.0078 g/cm^3 (see "Specific Gravity of Boric Acid Solutions," Handbook of Chemistry, 1162, Compiled and Edited by Norbert A. Lange, Ph.D, 1956.). The solution temperature is 15ºC and the boron is natural boron.

An easy way to describe this mixture is to use a combination of a user-defined compound to describe the boric acid, and a basic composition to describe the water.

STEP 1. INPUT DATA TO DESCRIBE THE USER-DEFINED COMPOUNDThe generic input data for the boric acid are given below. The actual input data are derived in steps 2 through 5.

```
ATOMH3BO3  2  0.025066  3  5000  1  1001  3  8016  3  1.0  288.15  END
```

STEP 2. AUXILIARY CALCULATIONS FOR THE USER-DEFINED COMPOUND INPUT DATA

In calculating the molecular weights, use the atomic weights from SCALE, which are available in the table *Isotopes in standard composition library* in Sect. 7.2.2 of the SCALE manual. The atomic weights used in SCALE may differ from some periodic tables. The SCALE atomic weights used in this problem are listed below:

H (1001) 1.0078

O (8016) 15.9949

^{10}B 10.0129

^{11}B 11.0093

The natural boron abundance, in weight percent, is defined to be:

^{10}B 18.4309

^{11}B 81.5691

The molecular weight of natural boron is given by

DEN nat B/AWT nat B = DEN ^{10}B/AWT ^{10}B + DEN ^{11}B/AWT ^{11}B

DEN ^{10}B = WTF ^{10}B × DEN nat B

DEN ^{11}B = WTF ^{11}B × DEN nat B

where:

DEN is density in g/cm^3,

AWT is the atomic weight in g/mol,

WTF is the weight fraction of the isotope.

Substituting,

DEN nat B/AWT nat B = DEN nat B × ((WTF ^{10}B/AWT ^{10}B) + (WTF ^{11}B/AWT ^{11}B))

Solving for AWT nat B yields:

AWT nat B = 1/((WTF ^{10}B/AWT ^{10}B) + (WTF ^{11}B/AWT ^{11}B))

The atomic weight of natural boron is thus

1.0/((0.184309 g ^{10}B/g nat B/10.0129 g ^{10}B/mol ^{10}B) + (0.815691 g ^{11}B/g nat B/11.0093 g /mol ^{11}B)) = 10.81103 g nat B/mol nat B

The molecular weight of the boric acid, H_3BO_3 is given by:

(3 × 1.0078) + 10.81103 + (3 × 15.9949) = 61.8191

Calculate the grams of boric acid in a gram of solution:

Boric acid, H_3BO_3 is 61.8222 g/mol

Natural boron is 10.81261 g/mol

(4350 × 10^{-6} g B/g soln) × (1 mol/10.81261 g B) × (61.8191 g boric acid/mol) =

0.024874 g boric acid/g soln (2.4874 wt %)

Interpolating from the referenced page from Lange's Handbook of Chemistry, the specific gravity of the boric acid solution at 2.4872 weight percent is 1.0087. This value is based on water at 15°. Therefore, the density of the boric acid solution is 1.0087 × 0.99913 g/cm^3 = 1.0078 g soln/cm^3.

Calculate ROTH, the theoretical density of the boric acid.

1.0078 g soln/cm^3 × 0.024874 g boric acid/g soln = 0.025068 g boric acid/cm^3

STEP 3. DESCRIBE THE BASIC STANDARD COMPOSITION INPUT DATA

```
H2O     2  0.984507  288.15  END
```

where the volume fraction =0.984506 (see step 4 auxiliary calculations below)

STEP 4. AUXILIARY CALCULATIONS FOR THE BASIC STANDARD COMPOSITION INPUT DATA

Calculate the volume fraction of the water in the solution, assuming 0.9982 is the theoretical density of water from Table 7.2.4. Each gram of solution contains 0.024872 g of boric acid, so there is 0.975128 g of water in each gram of solution. The volume fraction of water is then given by:

(1.0078 g soln/cm^3 × 0.975128 g water/g soln)/0.9982 g water/cm^3 = 0.984506

STEP 5. CREATE THE MIXTURE FOR BORATED WATER

```
ATOMH3BO3  2  0.025068  3  5000  1  1001  3  8016  3  1.0  288.15  END
H2O                  2  0.984506  288.15  END
```

Combinations of basic and solution standard compositions to define a mixture

The solution specification is the easiest way of specifying the solutions listed in the *Available fissile solution components* table in Sect. 7.2.3. A combination of solution and basic standard compositions can be used to describe a mixture that contains more than just a solution as demonstrated in the following example.

EXAMPLE 1. Uranyl nitrate solution containing gadolinium.

> Create a 4.306% enriched uranyl nitrate solution containing 0.184 g gadolinium per liter. The uranium in the nitrate is 95.65% ^{238}U, 0.022% ^{236}U, 4.306% ^{235}U, and 0.022% ^{234}U. The uranium concentration is 195.8 g U/L and the specific gravity of the uranyl nitrate is 1.254. There is no excess acid in the solution. The presence of the gadolinium is assumed to produce no significant change in the solution density. The solution is defined to be mixture 3.

```
SOLUTION  MIX=3
  RHO[UO2(NO3)2]=195.8 92238 95.65 92236 0.022 92235 4.306 92234 0.022
  VOL_FRAC=0.99985
  DENSITY=1.254
END SOLUTION
GD  3  0.000184  293  END
```

Combinations of user-defined compound and solution to define a mixture

The solution specification is the easiest way of specifying the solutions listed in the *Available fissile solution components* table in Sect. 7.2.3 of the SCALE manual. A solution specification and user-defined compound specification can be used to describe a mixture that contains more than just a solution as demonstrated in the following example.

EXAMPLE 1. Uranyl nitrate solution with gadolinium nitrate.

> Create a 4.306% enriched uranyl nitrate solution containing gadolinium in the form of $Gd(NO_3)_3$. The uranium in the nitrate is 95.65% ^{238}U, 0.022% ^{236}U, 4.306% ^{235}U, and 0.022% ^{234}U. The uranium concentration is 195.8 g U/L and the density of the uranyl nitrate is 1.254. There is no excess acid in the solution. The concentration of the gadolinium is 0.184 g/L. The volume fraction of the mixture that is uranyl nitrate (0.99985 = 1.254/ (1.254 + 0.000184)). The solution is defined to be mixture 3.

```
SOLUTION  MIX=3
  RHO[UO2(NO3)2]=195.8 92238 95.65 92236 0.022 92235 4.306 92234 0.022
  VOL_FRAC=0.99985
  DENSITY=1.254
END SOLUTION
```

The density of the gadolinium is given as 0.184 g/L. To describe the user-defined compound, the density of the $Gd(NO_3)_3$ is needed. The atomic weights from the Standard Composition Library are:

> Gd 157.25
>
> N 14.0067
>
> O 15.999

Therefore, the density of the $Gd(NO_3)_3$ = 0.000184 g Gd/cm^3 × (157.25 + 3(14.0067 + 3(15.999))/157.25) = 0.0004017 g/cm^3.

The input data for this user-defined compound are given below:

```
ATOMGD(NO3)3  3  .0004017  3  64000  1  7014  3  8016  9  1.0  300  END
```

The complete input data for the mixture of uranyl nitrate and gadolinium nitrate are given as:

```
SOLUTION  MIX=3
  RHO[UO2(NO3)2]=195.8 92238 95.65 92236 0.022 92235 4.306 92234 0.022
  VOL_FRAC=0.99985
  DENSITY=1.254
END SOLUTION
ATOMGD(NO3)3  3  .0004017  3  64000  1  7014  3  8016  9  1.0  300  END
```

Note: Since the default temperature (300 K) is to be used, it can be omitted from the user-defined compound standard composition. The temperature must be entered if the standard composition contains a multiple-isotope nuclide whose isotopic abundance is to be specified.

7.1.4.2 XSProc Standard Composition Examples

Infinite homogeneous medium unit cell data

EXAMPLE 1. A single mixture 1.

> Consider a single cylindrical configuration of mixture 1, composed of 10% enriched UO_2 having a radius of 35 cm and a height of 20 cm. This fuel region is sufficient large to model as an infinite medium. Mixture 100 may be used in subsequent multigroup neutron transport calculations.

```
INFHOMMEDIUM  1  CELLMIX=10  END
```

XSDRNPM will calculate the eigenvalue of an infinite mass of 10% enriched UO_2.

LATTICECELL unit cell data

Examples of "regular" **LATTICECELL** unit cells are given in Examples 1–5, and examples of "annular" **LATTICECELL** unit cells are given in Examples 6–10 below.

EXAMPLE 1. SQUAREPITCH (infinitely long cylindrical pins in a square-pitched array).

> Consider a large array of UO_2 fuel pins having a fuel O.D. of 0.79 cm, a 0.015-cm gap, and a 0.06-cm-thick aluminum clad. The array is a square-pitched array with a center-to-center spacing of 1.60 cm and is completely flooded with water. In the standard composition data, UO_2 is defined to be mixture 1, the aluminum clad is defined to be mixture 2, and the water moderator is defined to be mixture 3.

```
LATTICECELL  SQUAREPITCH PITCH=1.60 3 FUELD=0.79 1 CLADD=0.94 2 GAPD=0.82 0 END
```

EXAMPLE 1a. SQUAREPITCH (infinitely long cylindrical pins with multiple fuel zones in a square-pitched array).

Consider a large array of UO2 fuel pins having a fuel radius of 0.79 cm with three equi-volume fuel zones with different compositions, a 0.015-cm gap, and a 0.06-cm-thick aluminum clad. The array is a square-pitched array with a center-to-center spacing of 1.60 cm and is completely flooded with water. In the standard composition data, UO2 zones are defined to be mixtures 1 through 3, the aluminum clad is defined to be mixture 4, and the water moderator is defined to be mixture 5.

```
LATTICECELL  SQUAREPITCH      PITCH=1.60 5
                           FUELR=0.228053 1
                           FUELR=0.322516 2
                           FUELR=0.395000 3
                           CLADR=0.47 4
                           GAPR=0.41 0 END
```

EXAMPLE 2. TRIANGPITCH (infinitely long cylinders in a triangular-pitched array).

Consider an array of UO_2 pins with a diameter of 0.635 m. The outside diameter of the clad is 0.78 cm. There is no gap between the fuel and clad. The array is a triangular-pitched array with a center-to-center spacing of 5.0 cm and is flooded with water. In the standard composition data, the UO_2 is defined to be mixture 1, the aluminum is defined to be mixture 2, and the water moderator is defined to be mixture 3.

```
LATTICECELL  TRIANGPITCH PITCH=5.0  3  FUELD=.635  1  CLAD=.78  2  END
```

EXAMPLE 3. SPHSQUAREP (spheres in a square-pitched array).

Consider a large array of U_3O_8 spheres having a fuel O.D. of 18.6 cm, with an aluminum clad that is 0.18 cm thick. The array is a triangular-pitched array with a center-to-center spacing of 19.0 cm and is unmoderated. In the standard composition data, the aluminum is defined to be mixture 1 and the U_3O_8 is defined to be mixture 2. There is no moderator material, so 0 will be used to represent a void. Also, have XSDRNPM make a cell weighted material 20 from this unit cell.

```
LATTICECELL SPHSQUAREP PITCH=19.0 0 FUELD=18.6 2 CLADD=18.96 1 CELLMIX=20 END
```

EXAMPLE 4. SPHTRIANGP (spheres in a triangular-pitched array).

Consider a large array of U_3O_8 spheres having a fuel O.D. of 18.6 cm, with an aluminum clad that is 0.18 cm thick. The array is a triangular-pitched array with a center-to-center spacing of 19.0 cm and is flooded with water. In the standard composition data, the aluminum is defined to be mixture 1, the U_3O_8 is defined to be mixture 2, and the water moderator is defined to be mixture 3.

```
LATTICECELL  SPHTRIANGP  PITCH=19.0  3  FUELD=18.6  2 CLADD=18.96  1  END
```

EXAMPLE 5. SYMMSLABCELL (slabs repeated in a symmetric fashion).

Consider a system of alternating slabs of U_3O_8 and low-density water. Each U_3O_8 region is 1.27 cm thick, and each water region is 15.0 cm thick. In the standard composition data, the U_3O_8 is defined to be mixture 1, and the low-density water is defined to be mixture 2.

```
LATTICECELL  SYMMSLABCELL PITCH=16.27  2 FUELD=1.27  1  END
```

EXAMPLE 5a. SYMMSLABCELL (slabs repeated in a symmetric fashion).

Consider a system of alternating slabs of U_3O_8 and low-density water. Each U_3O_8 region is 1.27 cm thick, and each water region is 15.0 cm thick. The U_3O_8 regions have a 0.01-cm gap and 0.24-cm-thick aluminum clad on each face. In the standard composition data, the U_3O_8 is defined to be mixture 1, the aluminum is defined to be mixture 2, and the low-density water is defined to be mixture 3. Also, have XSDRNPM make a cell-weighted material 100 from this unit cell.

```
LATTICECELL  SYMMSLABCELL  PITCH=16.77  3  FUELD=1.27  1
CLADD=1.77  2  GAPD=1.29  0  CELLMIX=100  END
```

EXAMPLE 6. ASQUAREPITCH (infinitely long annular cylindrical rods in a square-pitched array).

Consider an array of uranium metal pipes having an inside diameter of 5.0 cm and an outer diameter of 6.75 cm. A gap of 0.025 cm and a clad of 0.25 cm exist on both the inner and outer surfaces of the fuel. The fuel rods are arranged in a square-pitched array. The center-to-center spacing is 8.0 cm. The array is completely flooded with water. In the standard composition data, the uranium metal is defined to be mixture 1, the outer clad is mixture 2, the inner clad is mixture 7, the inner moderator is Plexiglas and is mixture 3, the gap is a void, and the external moderator is water, defined to be mixture 4.

```
LATTICECELL  ASQUAREPITCH  PITCH=8.0  4  FUELD=6.75  1  GAPD=6.8  0
CLADD=7.3  2  IMODD=4.45  3  ICLADD=4.95 7  IGAPD=5.0  0  END
```

EXAMPLE 6a. ASQUAREPITCH (infinitely long annular cylindrical rods in a square-pitched array).

Consider an array of uranium metal pipes having an inside diameter of 5.0 cm and an outer diameter of 6.75 cm arranged in a square-pitched array. The center-to-center spacing is 8.0 cm. The array is completely flooded with water. In the standard composition data, the uranium metal is defined to be mixture 1, the water moderator is defined to be mixture 2, and the inside water moderator is defined as mixture 3.

```
LATTICECELL  ASQUAREPITCH  PITCH=8.0  2  FUELD=6.75  1  IMODD=5.0  3  END
```

Note: This problem defines two water mixtures. If mixture 2 were entered twice, i.e., for both the inner and outer moderator, an error message would be printed and the calculation terminated.

EXAMPLE 7. ATRIANGPITCH (infinitely long annular cylindrical rods in a triangular-pitched array).

Consider an array of uranium metal pipes having an inside diameter of 8.0 cm and a wall thickness of 0.75 cm arranged in a square-pitched array. The center-to-center spacing is 9.75 cm. The array is completely flooded with water. A Plexiglas rod fills the center of the uranium pipe. In the standard compositions data, the uranium metal is defined to be mixture 1, the Plexiglas is defined to be mixture 2, and the external water moderator is mixture 3.

```
LATTICECELL  ATRIANGPITCH  PITCH=9.75  3  FUELD=9.5  1  IMODD=8.0  2  END
```

EXAMPLE 8. ASPHSQUAREP (spherical annuli in a square-pitched array).

Consider a large array of hollow U_3O_8 spheres having a fuel I.D. of 8.0 cm and O.D. of 18.6 cm. The centers of the spheres are empty. The external moderator is water. The spheres are stacked in a square-pitched array with a center-to-center spacing of 19.0 cm. In the standard composition data, the U_3O_8 is defined to be mixture 1, and the water is defined to be mixture 2. The centers of the spheres are defined to be void, mixture 0.

```
LATTICECELL  ASPHSQUAREP  HPITCH=9.5  2  FUELR=9.3  1  IMODR=4.0  0  END
```

EXAMPLE 9. ASPHTRIANGP (spheres in a triangular-pitched array).

Consider a large array of hollow U_3O_8 spheres having a fuel I.D. of 8.0 cm and a fuel O.D. of 18.6 cm. A 0.18-cm-thick aluminum clad exists outside the fuel. The interior of each sphere is void. The array is a triangular-pitched array with a center-to-center spacing of 19.0 cm and is flooded with water. In the standard composition data, the aluminum is defined to be mixture 1, the U_3O_8 is defined to be mixture 2, and the water moderator is defined to be mixture 3. The void in the center of each sphere is entered as mixture 0.

```
LATTICECELL ASPHTRIANGP HPITCH=9.5  3 FUELR=9.3  2 IMODR=4.0  0 CLADR=9.48  1 END
```

EXAMPLE 10. ASYMSLABCELL (repeated slabs having different moderator conditions on the left and right boundaries).

Consider an array of U_3O_8 slabs with an inner moderator region composed of full-density water with a half thickness of 8.0 cm, and a low-density water outer moderator with a 16 cm half thickness of 16 cm half thickness. Each U_3O_8 slab is 10.54 cm thick. In the standard composition data, the U_3O_8 is defined to be mixture 1, the full density water is defined to be mixture 2, and the low-density water is mixture 3. Also, have XSDRNPM create a cell weighted mixture 100 from this unit cell.

```
LATTICECELL ASYMSLABCELL CELLMIX=100 IMODR=8.0 2 FUELR=18.54 1  HPITCH=34.54 3 END
```

EXAMPLE 10a. ASYMSLABCELL (repeated slabs having different moderator conditions on the left and right boundaries).

Consider an array of U_3O_8 fuel plates with an inner moderator region of full-density water with a half-thickness of 8.0 cm, and with a 16 cm thick low-density outer moderator. Each fuel plate includes a 10.54 cm thick U_3O_8 slab with a 0.01 cm gap and 0.24-cm-thick aluminum clad on each face. In the standard composition data, the U_3O_8 is defined to be mixture 1, the full density water is defined to be mixture 2, and the low-density water is mixture 3, the inner aluminum is mixture 4, the outer aluminum clad is mixture 5, and both gaps are voids.

```
LATTICECELL ASYMSLABCELL IMODR=8.0 2 ICLADR=8.24 5 IGAPR=8.25 0 FUELR=18.79 1
GAPR 18.80 0 CLADR 19.04 4 HPITCH=27.04 3 END
```

MULTIREGION unit cell data

Examples of **MULTIREGION** unit cells follow:

EXAMPLE 1. SLAB.

Consider a 5-cm-thick slab of fuel (mixture 1) with 0.5 cm of aluminum (mixture 3) and 15 cm of water (mixture 2) on each face. The unit cell data for this problem could be entered as follows:

```
MULTIREGION  SLAB  LEFT_BDY=REFLECTED  RIGHT_BDY=VACUUM  ORIGIN=0  END
1  2.5  3  3.0  2  18.0  END ZONE
```

EXAMPLE 2. CYLINDRICAL.

Consider a large array of fuel pins. Each pin is UO_2 (mixture 1) with a radius of 0.465 cm, a 0.009-cm gap (mixture 0), and a Zircaloy clad (mixture 9) 0.062 cm thick, centered in a water (mixture 8) region surrounded by a flooded support structure represented by homogenized water and Zircaloy (mixture 10). The outer radius of the water-Zircaloy region is 0.844 cm and it is 0.037 cm thick. This problem cannot be described as a **LATTICECELL** problem because the **LATTICECELL** configuration is limited to fuel-gap-clad-cell boundary and this problem is fuel-gap-clad-moderator-outer region. When **MULTIREGION** is used, lattice effects are accounted for by specifying a **WHITE**, **PERIODIC**, or **REFLECTED** right boundary condition, as long as the CENTRM/PMC self-shielding method is used. **MULTIREGION** cells should not be used for arrays if BONAMI-only method is specified

```
MULTIREGION  CYLINDRICAL RIGHT_BDY=WHITE  END
1  0.465  0  0.474  9  0.536  8  0.807  10  0.844  END ZONE
```

EXAMPLE 3. SPHERICAL.

Describe a bare sphere of uranium metal 8.72 cm in radius. The uranium metal is defined to be mixture 1. Also, have XSDRNPM create a cell weighted mixture 100 and calculate and eigenvalue. The unit cell data for this problem could be entered as follows:

```
MULTIREGION  SPHERICAL  CELLMIX=100  END   1  8.72    END ZONE
```

EXAMPLE 4. BUCKLEDSLAB.

Consider a plate of fuel 4 cm thick, reflected by 3 cm of water on both faces. The plate is 32 cm tall and 16 cm deep. The fuel is mixture 1 and the water is mixture 2. Also, have XSDRNPM create a cell weighted mixture 100 and calculate and eigenvalue.

```
MULTIREGION  BUCKLEDSLAB  CELLMIX=100  LEFT_BDY=REFLECTED  RIGHT_BDY=VACUUM
DY=32 DZ=16.0  END  1  2.0  2  5.0  END ZONE
```

EXAMPLE 5. BUCKLEDCYL.

Consider a solution of uranyl nitrate contained in a cylindrical stainless-steel container reflected by 33 cm of water. The inside dimensions of the steel container are 7.62 cm in radius and 130.0 cm tall. The steel is 0.15 cm thick. The uranyl nitrate is defined to be mixture 1, the steel is defined to be mixture 2, and the water is defined to be mixture 3.

```
MULTIREGION  BUCKLEDCYL  DY=130  END
1  7.62  2  7.77  3  40.77  END ZONE
```

DOUBLEHET unit cell data

Unit cell data are always required for **DOUBLEHET** calculations. As many unit cells as needed may be defined in the problem. If **CELLMIX**=*mx* is entered after the fuel element (macro cell) description, XSProc calls XSDRNPM to calculate the eigenvalue of the cell and to create a homogenized cell-weighted cross section having the characteristics of the doubly-heterogeneous cell configuration.

EXAMPLE 1: A doubly-heterogeneous spherical fuel element with 15,000 UO_2 particles in a graphite matrix.

Grain fuel radius is 0.025 cm. Grain contains one coating layer that is 0.009-cm-thick. Pebbles are in a triangular pitch on a 6.4-cm-pitch. Fuel pebble fuel zone is 2.5-cm in radius and contains a 0.5-cm-thick graphite clad that contains small amounts of ^{10}B. Pebbles are surrounded by ^{4}He. Assume the composition block is below:

```
' UO2 FUEL KERNEL
U-235  1 0 1.92585E-3 293.6 END
O      1 0 4.64272E-2 293.6 END
' FIRST COATING
C      2 0 5.26449E-2 293.6 END
' GRAPHITE MATRIX
C      6 0 8.77414E-2 293.6 END
' CARBON PEBBLE OUTER COATING
C      7 0 8.77414E-2 293.6 END
B-10   7 0 9.64977E-9 293.6 END
HE-4   8 0 2.65156E-5 293.6 END
```

The cell data for the **DOUBLEHET** cell follows:

```
DOUBLEHET FUELMIX=10 END
 GFR=0.025  1 COATT=0.009 2 MATRIX=6 NUMPAR=15000 END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.2 8 FUELR=2.5 CLADR=3.0 7  END
```

In this case we designated the homogenized mixture as mixture 10. If we have a KENO V.a or KENO-VI input section, we would use mixture 10 in that section. Note that the keyword "**FUELR**=" is followed by the fuel dimension only, i.e., no mixture number. That is because the fuel mixture number is specified with "**FUELMIX**=" and therefore need not be repeated.

EXAMPLE 2: Same as Example 1, except volume fraction of the grain type is known and is 0.037732.

```
DOUBLEHET  RIGHT_BDY=WHITE FUELMIX=10 END
 GFR=0.025  1 COATT=0.009 2 MATRIX=6 VF=0.037732 END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.2 8 FUELR=2.5 CLADR=3.0 7  END
```

EXAMPLE 3: Same as Example 1, except halfpitch of the grain type is known and is 0.10137 cm.

```
DOUBLEHET FUELMIX=10 END
 GFR=0.025  1 COATT=0.009 2 HPITCH=0.10137 MATRIX=6 END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.2 8 FUELR=2.5 CLADR=3.0 7  END
```

EXAMPLE 4: A doubly-heterogeneous spherical fuel element with 10,000 UO2 particles and 5,000 PuO2 particles in a graphite matrix.

> Grain fuel radii for UO_2 and PuO_2 particles are 0.025 cm and 0.012 cm, respectively. UO_2 grains contain one coating layer that is 0.009 cm-thick. PuO_2 grains contain one coating layer that is 0.0095 cm-thick. Pebbles are in a triangular pitch on a 6.4-cm-pitch. Fuel pebble fuel zone is 2.5-cm in radius and contains a 0.5-cm-thick graphite clad that contains small amounts of ^{10}B. Pebbles are surrounded by ^{4}He. Assume the composition block is given below:

```
' UO2 FUEL KERNEL
U-235  1 0 1.92585E-3 293.6 END
O      1 0 4.64272E-2 293.6 END
' FIRST COATING
C      2 0 5.26449E-2 293.6 END
' GRAPHITE MATRIX
C      6 0 8.77414E-2 293.6 END
' CARBON PEBBLE OUTER COATING
C      7 0 8.77414E-2 293.6 END
B-10   7 0 9.64977E-9 293.6 END
HE-4   8 0 2.65156E-5 293.6 END
' PUO2 FUEL KERNEL
PU-239  11 0 1.24470E-02 293.6 END
O       11 0 4.60983E-02 293.6 END
' FIRST COATING
```

(continues on next page)

(continued from previous page)

```
C       12 0 5.26449E-2 293.6 END
' GRAPHITE MATRIX
C       16 0 8.77414E-2 293.6 END
```

The cell data for the **DOUBLEHET** cell follows:

```
DOUBLEHET FUELMIX=10 END
 GFR=0.025  1 COATT=0.009 2 MATRIX=6 NUMPAR=10000 END GRAIN
 GFR=0.012 11 COATT=0.0095 12 MATRIX=16 NUMPAR=5000 END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.2 8 FUELR=2.5 CLADR=3.0 7 END
```

Since number of particles is entered, the total volume fraction and the pitch can be calculated by the code.

EXAMPLE 5: Same as Example 4 above except the volume fractions of UO_2 and PuO_2 grains are 0.02511 and 0.00318, respectively.

```
DOUBLEHET  RIGHT_BDY=WHITE FUELMIX=10 END
 GFR=0.025  1 COATT=0.009 2 MATRIX=6 VF=0.02511 END GRAIN
 GFR=0.012 11 COATT=0.0095 12 MATRIX=16 VF=0.00318 END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.2 8 FUELR=2.5 CLADR=3.0 7 END
```

EXAMPLE 6: Same as Example 4 above except pitch is also known.

UO_2 grains have a pitch of 0.25 cm. PuO_2 grains have a pitch of 0.20 cm.

```
DOUBLEHET FUELMIX=10 END
 GFR=0.025  1 COATT=0.009 2
 MATRIX=6 NUMPAR=10000 PITCH=0.25 END GRAIN
 GFR=0.012  11 COATT=0.0095 12
 MATRIX=16 NUMPAR=5000 PITCH=0.20 END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.2 8 FUELR=2.5 CLADR=3.0 7 END
```

Since number of particles is sufficient to perform the homogenization, it is used. However, instead of calculating the pitch for the 1-D cell calculation for each grain type, the user input pitch is used. Hence, the calculated k_{eff} of Example 6 will be different from those of Examples 4 and 5.

EXAMPLE 7: Same as Example 6 except the doubly-heterogeneous cell will be cell-weighted.

The final cell-weighted mixture number is 17.

```
DOUBLEHET FUELMIX=10 END
 GFR=0.025  1 COATT=0.009 2
 NUMPAR=10000 PITCH=0.25 MATRIX=6 END GRAIN
 GFR=0.012  11 COATT=0.0095 12
 NUMPAR=5000 PITCH=0.20 MATRIX=16 END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.2 8 FUELR=2.5 CLADR=3.0 7 CELLMIX=17 END
```

EXAMPLE 8: A doubly-heterogeneous spherical fuel element with 15,000 UO_2 particles in a graphite matrix.

Grain fuel radius is 0.012 cm. Grain contains four coating layers that are 0.0095, 0.004, 0.0035, and 0.004-cm-thick. Pebbles are in a square pitch on a 6.0 cm-pitch. Fuel pebble fuel zone is 2.5-cm in radius and contains a 0.5-cm-thick graphite clad that contains small amounts of ^{10}B. Pebbles are surrounded by ^{4}He. Assume the composition block is given below:

```
' UO2 FUEL KERNEL
U-235  1 0 1.92585E-3 293.6 END
O       1 0 4.64272E-2 293.6 END
' FIRST COATING
```

(continues on next page)

(continued from previous page)

```
C       2 0 5.26449E-2 293.6 END
' INNER PYRO CARBON
C       3 0 9.52621E-2 293.6 END
' SILICON CARBIDE
C       4 0 4.77240E-2 293.6 END
SI      4 0 4.77240E-2 293.6 END
' OUTER PYRO CARBON
C       5 0 9.52621E-2 293.6 END
' GRAPHITE MATRIX
GRAPHITE 6 0 8.77414E-2 293.6 END
' CARBON PEBBLE OUTER COATING
C       7 0 8.77414E-2 293.6 END
B-10    7 0 9.64977E-9 293.6 END
HE-4    8 0 2.65156E-5 293.6 END
```

The cell data for the **DOUBLEHET** cell follows:

```
DOUBLEHET FUELMIX=10 END
 GFR=0.012  1 COATT=0.0095 2 COATT=0.004 3 COATT=0.0035 4 COATT=0.004 5 MATRIX=6 NUMPAR=15000 VF=0.0245 END␣
↪GRAIN
PEBBLE SPHSQUAREP RIGHT_BDY=WHITE HPITCH=3.0 8 FUELR=2.5 CLADR=3.0 7 END
```

Note that the grains are overspecified and the numbers are inconsistent. A **VF** value of 0.0245 results in a total number of particles of 10652.32 which is considerably less than 15,000. In this case, the code will issue a warning to this effect and will use **VF** value in the calculations (i.e., ignore **NUMPAR**=15000 entry).

EXAMPLE 9: Similar to Example 8 except radii for grain regions are entered.

```
DOUBLEHET FUELMIX=10 END
 GFR=0.012  1 COATR=0.0215 2 COATR=0.0255 3 COATR=0.029 4 COATR=0.033 5 MATRIX=6 NUMPAR=15000 VF=0.0245 END␣
↪GRAIN
PEBBLE SPHSQUAREP RIGHT_BDY=WHITE HPITCH=3.0 8 FUELR=2.5 CLADR=3.0 7 END
```

EXAMPLE 10: A doubly-heterogeneous spherical fuel element with two UO_2 grain types.

> First grain type has a fuel radius of 0.025 cm. Second grain type fuel radius is 0.004 cm. First grain type has one coating that is 0.009-cm-thick. Second grain type has two coatings each 0.004-cm-thick. Each grain type has a volume fraction of 0.45. Pebbles are in a triangular pitch on a 7.0-cm-pitch. Fuel pebble fuel zone is 2.5-cm in radius and contains a 0.5-cm-thick graphite clad that contains small amounts of ^{10}B and ^{11}B. Pebbles are surrounded by ^{4}He. Assume the composition block is given below:

```
' FUEL KERNEL
U-238   1 0 2.12877E-2 END
U-235   1 0 1.92585E-3 END
O       1 0 4.64272E-2 END
B-10    1 0 1.14694E-7 END
B-11    1 0 4.64570E-7 END
' FIRST COATING
C       2 0 5.26449E-2 END
' INNER PYRO CARBON
C       3 0 9.52621E-2 END
' SILICON CARBIDE
C       4 0 4.77240E-2 END
SI      4 0 4.77240E-2 END
' FUEL KERNEL
U-238   5 0 2.12877E-2 END
U-235   5 0 1.92585E-3 END
O       5 0 4.64272E-2 END
```

(continues on next page)

(continued from previous page)

```
B-10    5 0 1.14694E-7 END
B-11    5 0 4.64570E-7 END
' GRAPHITE MATRIX
C       6 0 8.77414E-2 END
B-10    6 0 9.64977E-9 END
B-11    6 0 3.90864E-8 END
' CARBON PEBBLE OUTER COATING
C       7 0 8.77414E-2 END
B-10    7 0 9.64977E-9 END
B-11    7 0 3.90864E-8 END
' HELIUM
HE      8 0.000164 END
' GRAPHITE MATRIX
C       9 0 8.77414E-2 END
B-10    9 0 9.64977E-9 END
B-11    9 0 3.90864E-8 END
```

The cell data for the **DOUBLEHET** cell follows:

```
DOUBLEHET FUELMIX=10 END
 GFR=0.025  1 COATR=0.034 2  MATRIX=6 VF=0.45 END GRAIN
 COATT=0.004 3 GFR=0.4 5 COATT=0.004 4 MATRIX=9  VF=0.45    END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.5 8 FUELD=5.0
  CLADD=6.0 7 END
```

EXAMPLE 11: A doubly-heterogeneous hexagonal block type fuel element with UO_2 grains in a cylindrical fuel region.

> Grain fuel radius is 0.025 cm. Grain coating is 0.009-cm-thick. Grains have a volume fraction of 0.45. Hexagonal rods are in a 7-cm triangular pitch. Fuel rod fuel zone is 2.5-cm in radius, 10-cm-high and contains a 0.5-cm-thick graphite clad that contains small amounts of ^{10}B. Assume the composition block is below:

```
' FUEL KERNEL
U-238   1 0 2.12877E-2 END
U-235   1 0 1.92585E-3 END
O       1 0 4.64272E-2 END
B-10    1 0 1.14694E-7 END
' FIRST COATING
C       2 0 5.26449E-2 END
' GRAPHITE MATRIX
C       6 0 8.77414E-2 END
B-10    6 0 9.64977E-9 END
' CARBON PEBBLE OUTER COATING
C       7 0 8.77414E-2 END
B-10    7 0 9.64977E-9 END
' IRON CLADDING
FE      8 END
```

The cell data for the **DOUBLEHET** cell follows:

```
DOUBLEHET FUELMIX=10 END
 GFR=0.025  1 COATR=0.034 2  MATRIX=6 VF=0.45 END GRAIN
ROD TRIANGP RIGHT_BDY=WHITE HPITCH=3.5 7 FUELD=5.0
  FUELH=10 END
```

EXAMPLE 12: This is the same as Example 11 except the fuel elements (cylindrical rods) have 0.05 cm-thick iron cladding.

The cell data for the **DOUBLEHET** cell follows:

```
DOUBLEHET FUELMIX=10 END
 GFR=0.025  1 COATR=0.034 2  MATRIX=6 VF=0.45 END GRAIN
ROD TRIANGP RIGHT_BDY=WHITE HPITCH=3.5 7 FUELR=2.5
  CLADD=5.1 8 FUELH=10 END
```

Optional parameter data

The optional parameter data provide a means of providing additional information or alternative data to the cross-section processing codes. There are two types of optional parameter data. The first type of data is used by XSDRNPM and BONAMI for cross-section processing and cell-weighting cross sections. This type of data is initiated using the keywords **MORE DATA** and ends with the keywords **END MORE**. This input is described in Sect. 7.1.3.8. The second type of optional parameter data is used by CENTRM and PMC for cross-section processing. This type of data is initiated using the keywords **CENTRM DATA** and ends with the keywords **END CENTRM**. This input is described in Sect. 7.1.3.9. It is possible to input both types of data for a unit cell. The optional parameter data specified apply only to the unit cell that immediately precedes it.

MORE DATA examples

Consider a problem in which it is desirable to increase the number of inner iterations in XSDRNPM to 30 and to tighten the overall convergence criteria to a value of 0.000075. This could be accomplished by entering the data as follows:

```
MORE DATA   IIM=30  EPS=0.000075  END
```

The order of the data entry is not important, and it can be continued across several lines. However, a keyword and its value cannot be separated across lines. The terminator for the optional parameter data, END, must not begin in column 1 unless you assign a name to it. An alternative method of entering the above data is given below.

```
MORE DATA
  IIM=30  EPS=0.000075
END MORE
```

or,

```
MORE DATA  IIM=30  EPS=0.000075  END MORE DATA
```

CENTRM DATA examples

Consider a problem in which it is desirable to increase the upper energy of the CENTRM CE transport calculation from the default of 20000 eV to a value 50000 eV, and to extend the default lower energy from 0.001 eV to 0.0001. This is accomplished by entering the data as follows:

```
CENTRM DATA  DEMAX=50000  DEMIN=0.0001 END CENTRM
```

As with the **MORE DATA** block, an alternative method of entering the above data is given below.

```
CENTRMDATA
  DEMAX=50000  DEMIN=0.0001
END CENTRMDATA
```

CENTRM and PMC computation options can also be controlled with **CENTRM DATA.** A complete description of the CENTRM/PMC computational methods and options can be found the corresponding sections of the SCALE manual. The following example specifies that:

(a) discrete-level inelastic scattering will be used in CENTRM and processed in PMC [nmf6];

(b) the CENTRM 1D discrete S_N transport solver will be used in the upper MG energy range [nfst] and the CE energy range [npxs], while the infinitie medium model will be used for the thermal energy range [nthr];

(c) a P3 scattering order [isct] will be used in the transport calculations;

(d) PMC will perform "consistent PN" corrections on Legendre moments of the 2D elastic matrices [n2d]; (e) additional output information will be provided by CENTRM [ixprt] and by PMC [nprt].

```
CENTRM DATA  NMF6=0 NFST=0 NTHR=2 ISCT=3
       N2D=-2 IXPRT=1  NPRT=1    END CENTRM DATA
```

7.1.4.3 Examples of Complete XSProc Input Data

Infinite homogeneous medium input data

Examples of XSProc input data for infinite homogeneous media problems are given below. In these cases the cross section library name "fine_n" indicates that the latest recommended fine-group SCALE library will used in the calculations.

EXAMPLE 1. Default cell definition.

> Consider a cylindrical billet of 20 wt % enriched UO_2, having a density of 10.85 g/cm^3 that is 26 cm in diameter and 26 cm tall.

The average mean-free path in the uranium dioxide is on the order of 2.5 cm. Because only a small fraction of the billet is within a mean-free path of the surface, the material can be treated as an infinite homogeneous medium; therefore the CELL DATA block can be omitted. The XSProc data follows:

```
20%  ENRICHED  UO2  BILLET
fine_n
READ COMP
UO2  1  0.99  293  92235  20  92238  80  END
END  COMP
```

The volume fraction used for the UO_2, 0.99, is calculated by dividing the actual density by the theoretical density obtained from the *Isotopes in standard composition library* table in the STDCMP chapter, (10.85/10.96). Since the enrichment was specified as 20%, it is assumed that the remainder is ^{238}U.

An alternative input data description follows:

```
20% ENRICHED UO2 BILLET
fine_n
READ COMP
UO2 1 DEN=10.85  1  293  92235  20  92238  80  END
END COMP
```

EXAMPLE 2. Specify the cell definition.

> Consider a 5-liter Plexiglas bottle with an inner radius of 9.525 cm and inner height of 17.78 cm that is filled with highly enriched uranyl nitrate solution at 415 g/L and 0.39 mg of excess nitrate per gram of solution. The uranium isotopic content of the nitrate solution is 92.6 wt % ^{235}U, 5.9 wt % ^{238}U, 1.0 wt % ^{234}U, and 0.5 wt % ^{236}U. Solution density will be calculated from the given data.

The size of the nitrate solution is on the order of 16 to 20 cm in diameter and height. The average mean-free path in the nitrate solution is on the order of 0.5 cm. Therefore, infinite homogeneous medium is an appropriate choice for this problem. By default BONAMI is used for self-shielding the infinite medium of Plexiglas, while CENTRM is used to shield the infinite medium fissile solution.

```
SET UP 5 LITER URANYL NITRATE SOLUTION IN A PLEXIGLAS CONTAINER
fine_n
READ COMP
PLEXIGLAS   1  END
SOLUTION   MIX=2   RHO[UO2(NO3)2]=415
           92235   92.6   92238   5.9   92236   0.5
END SOLUTION
END  COMP
READ CELLDATA
INFHOMMEDIUM  2  END
END CELLDATA
```

LATTICECELL input data

Examples of XSProc input data for **LATTICECELL** problems are given below.

EXAMPLE 1. SQUAREPITCH ARRAY.

> Consider an infinite planar array (infinite in X and Y and one layer in Z) of 20 wt % enriched U metal rods with a 1-cm pitch. Each fuel rod is bare uranium metal, 0.75 cm OD × 30.0 cm long. The rods are submerged in water.

Because the diameter of the fuel rod, 0.75 cm, is only slightly larger than the average mean-free path in the uranium metal, approximately 0.5, and because the configuration is a regular array, **LATTICECELL** is the appropriate choice for proper cross-section processing. The *parm* field is not provided, so the default CENTRM/PMC self-shielding method is used. XSProc data follows:

```
INFINITE  PLANAR  ARRAY  OF  20%  U  METAL  RODS
fine_n
READ COMP
URANIUM  1  1  293  92235  20  92238  80  END
H2O      2  END
END  COMP
READ CELLDATA
LATTICECELL  SQUAREPITCH   PITCH=1.0  2  FUELD=0.75  1  END
END CELLDATA
```

Since the MORE DATA and CENTRM DATA blocks were omitted, default options will be used in the self-shielding calculations. The default CENTRM/PMC computation options for a square pitch lattice cell are the method-of-characteristics (MoC) method with P0 scatter in CENTRM calculations.

EXAMPLE 2. SQUAREPITCH PWR LATTICE.

> Consider an infinite, uniform planar array (infinite in X and Y and one layer in Z) of PWR-like fuel pins of 2.35% enriched UO_2 clad with zirconium. The density of the UO_2 is 9.21 g/cm^3. The fuel in each pin is 0.823 cm in diameter, the clad is 0.9627 cm in diameter, and the length of each pin is 366 cm. The fuel pins are separated by 0.3124 cm of water in the horizontal plane.

LATTICECELL is the appropriate choice for cross-section processing. Assume that all defaults are appropriate; thus the CENTRM/PMC methodology is used, and the MORE DATA and CELL DATA blocks are not entered. The input cross section library named "broad_n" indicates that the recommended broad group SCALE library will be used. In this case CENTRM uses the 2D MoC transport solver. The XSProc data follows:

```
PWR-LIKE FUEL BUNDLE; uniform infinite array model.
broad_n
READ COMP
UO2   1  .84  293.  92235  2.35  92238  97.65  END
ZR    2  1  END
H2O   3  1  END
END  COMP
READ CELLDATA
LATTICECELL  SQUAREPITCH  PITCH=1.2751  3  FUELD=0.823  1  CLADD=0.9627  2  END
END CELLDATA
```

EXAMPLE 3. SQUAREPITCH PWR LATTICE, with non-uniform Dancoff.

This example is a single PWR assembly of fuel pins of the type described above, contained in a water pool. The interior pins in the assembly can be self-shielded using the same uniform, infinite lattice model in previous example. However self-shielding of the outer boundary-edge pins will be modified to account for being adjacent to a water reflector, rather than surrounded on all sides by similar pins. This requires that the MCDancoff module be executed previously to obtain non-uniform Dancoff factors for the edge pins. The average edge-pin value of 0.61 is used to represent Dancoff factors of all boundary pins. The default CENTRM MoC transport solver is used for both cells, but the original pitch of 1.2751 cm for the second cell (i.e., boundary pin) is modified to a new pitch corresponding to a Dancoff value of 0.61.

```
PWR-LIKE FUEL BUNDLE, with boundary-pin corrections
broad_n
READ COMP
' mixtures for interior pins
UO2   1  .84  293.  92235  2.35  92238  97.65  END
ZR    2  1  END
H2O   3  1  END
' mixtures for boundary pins
UO2   4  .84  293.  92235  2.35  92238  97.65  END
ZR    5  1  END
H2O   6  1  END
END  COMP
READ CELLDATA
LATTICECELL  SQUAREPITCH PITCH=1.2751 3 FUELD=0.823 1 CLADD=0.9627 2  END
LATTICECELL  SQUAREPITCH PITCH=1.2751 6 FUELD=0.823 4 CLADD=0.9627 5  END
  CENTRM DATA  DAN2PITCH=0.61    END CENTRM
END CELLDATA
```

EXAMPLE 6. SPHTRIANGP ARRAY.

> Consider an infinite array of spherical pellets of 2.67% enriched UO_2 with a density of 10.3 g/cm^3 and a diameter of 1.0724 cm arranged in a "triangular" pitch, flooded with borated water at 4350 ppm. The boron is natural boron; the borated water is created by adding boric acid, H_3BO_3, and has a density of 1.0078 g/cm^3. The temperature is 15°.

Because the diameter of the fuel pellet, 1.0724 cm, is smaller than the average mean-free path in the UO_2, approximately 1.5 cm, and because the configuration is a regular array, **LATTICECELL** is the appropriate choice for proper cross-section processing.

The density fraction for the UO_2 is the ratio of actual to theoretical density (10.3/10.96 = 0.9398). Assume that the U is all ^{235}U and ^{238}U. See Sect. 7.1.4.1.9 for how to define borated water.

The XSProc data follows:

```
SPHERICAL  PELLETS  IN  BORATED  WATER
fine_n
```

(continues on next page)

(continued from previous page)

```
READ COMP
UO2   1  .9398  288  92235  2.67  92238  97.33  END
ATOMH3BO3  2  0.025066  3  5000  1  1001  3  8016  3
       1.0  288  END
H2O   2  0.984507  288  END
END  COMP
READ CELLDATA
LATTICECELL  SPHTRIANGP  PITCH=1.1440  2  FUELD=1.0724  1  END
END CELLDATA
```

MULTIREGION input data

Examples of XSProc input data for **MULTIREGION** problems are given below.

EXAMPLE 1. SPHERICAL.

> Consider a small highly enriched uranium sphere supported by a Plexiglas collar in a tank of water. The uranium metal sphere has a diameter of 13.1075 cm, is 97.67% enriched, and has a density of 18.794 g/cm^3. The cylindrical Plexiglas collar has a 4.1275-cm-radius central hole, extends to a radius of 12.7 cm and is 2.54 cm thick. The water filled tank is 60 cm in diameter.

The density fraction of the uranium metal is the ratio of actual to theoretical density, where the theoretical density is obtained from the *Isotopes in standard composition library* table in section 7.2.1. Thus, the density multiplier is 18.794/19.05 = 0.9866. The abundance of uranium is not stated beyond 97.67% enriched, so it is reasonable to assume the remainder is ^{238}U. The Plexiglas collar is not significantly different from water and does not surround the fuel, so it can be ignored. If it is ignored, the problem becomes a 1-D geometry that can be defined using the **MULTIREGION** type of calculation, and the eigenvalue of the system can be obtained without additional data by executing CSAS1. However, the Plexiglas has been included in this data so it can be passed to a code such as KENO V.a which can describe the geometry rigorously. The XSProc data follow:

```
SMALL  WATER  REFLECTED  SPHERE  ON  PLEXIGLAS  COLLAR
fine_n
READ COMP
URANIUM    1  .9866  293.  92235  97.67  92238  2.33  END
PLEXIGLAS  2  END
H2O        3  END
END  COMP
READ CELLDATA
MULTIREGION SPHERICAL RIGHT_BDY=VACUUM END 1 6.55375 3 30.0 END ZONE
END CELLDATA
```

EXAMPLE 2. BUCKLEDSLAB.

> This example features a 93.2% enriched uranyl-fluoride solution inside a rectangular Plexiglas container immersed in water. The fissile solution contains 578.7 g of UO_2F_2 per liter and has no excess acid. The critical thickness of the fuel is 5.384 cm. The finite height of the fuel slab is 147.32 cm, and the depth is 71.58 cm. The Plexiglas container is 1.905 cm thick and is reflected by 20.32 cm of water.

The half thickness of the fuel (2.692) will be used with a reflected left boundary and a vacuum right boundary (default). The XSProc data follow:

```
CRITICAL SLAB EXPERIMENT USING URANYL-FLUORIDE SOLUTION
fine_n
READ COMP
SOLUTION  MIX=1  RHO[UO2F2]=578.7
```

(continues on next page)

(continued from previous page)

```
          92235  93.2  92238  6.8  TEMP=300
END SOLUTION
PLEXIGLAS  2  END
H2O        3  END
END  COMP
READ CELLDATA
MULTIREGION  BUCKLEDSLAB  LEFT_BDY=REFLECTED
DY=71.58 DZ=147.32  END  1  2.692  2  4.597  3  24.917  END ZONE
END CELLDATA
```

DOUBLEHET input data

EXAMPLE 1: A doubly-heterogeneous spherical fuel element with 15,000 UO_2 particles in a graphite matrix.

Grain fuel radius is 0.025 cm. Grain contains one coating layer that is 0.009-cm-thick. Pebbles are in a triangular pitch on a 6.4-cm-pitch. Fuel pebble fuel zone is 2.5 cm in radius and contains a 0.5-cm-thick graphite clad that contains small amounts of ^{10}B. Pebbles are surrounded by ^{4}He. In this case we designated the homogenized mixture as mixture 10. If we have a KENO V.a or KENO-VI input section, we would use mixture 10 in that section. Note that the keyword "FUELR=" is followed by the fuel dimension only, i.e., no mixture number. That is because the fuel mixture number is specified with "FUELMIX=" and therefore need not be repeated.

```
INFINITE ARRAY OF UO2-FUELLED PEBBLES
fine_n
READ COMP
' UO2 FUEL KERNEL
U-235  1 0 1.92585E-3 293.6 END
O      1 0 4.64272E-2 293.6 END
' FIRST COATING
C      2 0 5.26449E-2 293.6 END
' GRAPHITE MATRIX
C      6 0 8.77414E-2 293.6 END
' CARBON PEBBLE OUTER COATING
C      7 0 8.77414E-2 293.6 END
B-10   7 0 9.64977E-9 293.6 END
HE-4   8 0 2.65156E-5 293.6 END
END COMP
READ CELLDATA
DOUBLEHET  RIGHT_BDY=WHITE FUELMIX=10 END
 GFR=0.025  1 COATT=0.009 2 MATRIX=6 NUMPAR=15000 END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.2 8 FUELR=2.5 CLADR=3.0 7  END
END CELLDATA
```

EXAMPLE 2: A doubly-heterogeneous spherical fuel element with 10,000 UO_2 particles and 5,000 PuO_2 particles in a graphite matrix.

Grain fuel radii for UO_2 and PuO_2 particles are 0.025 cm and 0.012 cm, respectively. UO_2 grains contain one coating layer that is 0.009 cm thick. PuO_2 grains contain one coating layer that is 0.0095-cm-thick. Pebbles are in a triangular pitch on a 6.4-cm-pitch. Fuel pebble fuel zone is 2.5-cm in radius and contains a 0.5-cm-thick graphite clad that contains small amounts of ^{10}B. Pebbles are surrounded by ^{4}He. Since number of particles is entered, the total volume fraction and the pitch can be calculated by the code.

```
INFINITE ARRAY OF UO2- AND PUO2-FUELLED PEBBLES
fine_n
READ COMP
' UO2 FUEL KERNEL
U-235  1 0 1.92585E-3 293.6 END
```

(continues on next page)

(continued from previous page)

```
O       1 0 4.64272E-2 293.6 END
' FIRST COATING
C       2 0 5.26449E-2 293.6 END
' GRAPHITE MATRIX
C       6 0 8.77414E-2 293.6 END
' CARBON PEBBLE OUTER COATING
C       7 0 8.77414E-2 293.6 END
B-10    7 0 9.64977E-9 293.6 END
HE-4    8 0 2.65156E-5 293.6 END
' PUO2 FUEL KERNEL
PU-239  11 0 1.24470E-02 293.6 END
O       11 0 4.60983E-02 293.6 END
' FIRST COATING
C       12 0 5.26449E-2 293.6 END
' GRAPHITE MATRIX
C       16 0 8.77414E-2 293.6 END
END COMP
READ CELLDATA
DOUBLEHET  RIGHT_BDY=WHITE FUELMIX=10 END
 GFR=0.025  1 COATT=0.009 2 MATRIX=6 NUMPAR=10000 END GRAIN
 GFR=0.012 11 COATT=0.0095 12 MATRIX=16 NUMPAR=5000 END GRAIN
PEBBLE SPHTRIANGP RIGHT_BDY=WHITE HPITCH=3.2 8 FUELR=2.5 CLADR=3.0 7 END
END CELLDATA
```

EXAMPLE 3: A doubly-heterogeneous slab fuel element with flibe salt coolant

Grain fuel radii for UO_2 particles are 0.025 cm. The UO_2 grains contain four coating layers with thicknesses of 0.01, 0.0035, 0.003, and 0.004 cm, respectively. The fuel grains are embedded in a carbon matrix material to form the fuel compact. The x-dimension of fuel plate consists of a 0.5 cm (half-thickness) fuel compact region, a carbon clad with outer dimension of 1.27, followed by the flibe coolant with an outer reflected dimension of 1.62 cm. The width (y-dimension) of the slab plate is 22.5 cm and the height (z-dimension) is 500 cm. The y and z dimensions are only used to define volumes for the fuel plate.

```
slab doublehet sample problem: double-het for slab
v7.1-252n
read comp
' fuel kernel
u-238  1 0 2.12877e-2 293.6 end
u-235  1 0 1.92585e-3 293.6 end
o      1 0 4.64272e-2 293.6 end
b-10   1 0 1.14694e-7 293.6 end
b-11   1 0 4.64570e-7 293.6 end
' first coating
c      2 0 5.26449e-2 293.6 end
' inner pyro carbon
c      3 0 9.52621e-2 293.6 end
' silicon carbide
c      4 0 4.77240e-2 293.6 end
si     4 0 4.77240e-2 293.6 end
' outer pyro carbon
c      5 0 9.52621e-2 293.6 end
' graphite matrix
c      6 0 8.77414e-2 293.6 end
b-10   6 0 9.64977e-9 293.6 end
b-11   6 0 3.90864e-8 293.6 end
' carbon slab outer coating
c      7 0 8.77414e-2 293.6 end
b-10   7 0 9.64977e-9 293.6 end
b-11   7 0 3.90864e-8 293.6 end
Li-6        8    0   1.38344E-06   948.15  end
Li-7        8    0   2.37205E-02   948.15  end
```

(continues on next page)

(continued from previous page)

```
Be          8    0   1.18609E-02   948.15  end
F           8    0   4.74437E-02   948.15  end
end comp
read celldata
  doublehet  fuelmix=10 end
    gfr=0.02135   1
    coatt=0.01    2
    coatt=0.0035  3
    coatt=0.003   4
    coatt=0.004   5
    vf=0.4
    matrix=6
    end grain
  slab symmslabcell
    hpitch=1.62   8
    cladr=1.27    7
    fuelr=0.5
    fuelh=500
    fuelw=22.500
  end
  centrm data ixprt=1 isn=8 end centrm
end celldata
```

EXAMPLE 4: A doubly-heterogeneous triangular-pitch fuel element with 1,302 UO_2 particles in a graphite matrix with the DAN2PITCH option for grain.

Grain fuel radius for UO2 particles are 0.02125 cm. The UO2 grains contain four coating layers with radii of 0.03125, 0.03525, 0.03875, and 0.04275 cm, respectively. The fuel grains are embedded in a carbon matrix material to form the fuel compact. Fuel compact is in a triangular pitch on a 1.8796-cm-pitch. Fuel zone is 0.6225-cm in radius and there is a 0.0125 cm gap between fuel and graphite moderator. Since number of particles is entered, the total volume fraction and the pitch can be calculated by the code. Dancoff factor of 0.6552 is inputted for a grain to consider neutron leakage effect.

```
DH_dan2pitch_nonuniform
v7.1-252
read composition
 u-235       1 0 3.6676E-03   600.0   end
 u-238       1 0 1.9742E-02   600.0   end
 o-16        1 0 3.5114E-02   600.0   end
 c           1 0 1.1705E-02   600.0   end
 c           2 0 5.2646E-02   600.0   end
 c           3 0 9.5263E-02   600.0   end
 si-28       4 0 4.4159E-02   600.0   end
 si-29       4 0 2.2433E-03   600.0   end
 si-30       4 0 1.4805E-03   600.0   end
 c           4 0 4.7883E-02   600.0   end
 c           5 0 9.5263E-02   600.0   end
 c-graphite  6 0 7.2701E-02   600.0   end
 he          7 0 2.4006E-05   600.0   end
 c-graphite  8 0 9.2756E-02   600.0   end
end composition

read celldata
 doublehet  fuelmix=9 end
  gfr=0.02125   1
  coatr=0.03125 2
  coatr=0.03525 3
  coatr=0.03875 4
  coatr=0.04275 5
  numpar=1302
  matrix=6  end grain
```

(continues on next page)

(continued from previous page)

```
 centrm data alump=0.0 dan2pitch=0.6562 end centrm
 rod triangpitch
  fuelr=0.6225
  gapr=0.635 7
  hpitch=0.9398 8
  fuelh=1.000
  right_bdy=white left_bdy=reflected end
 centrm data iup=12 isn=16  alump=0.0  end centrm
end celldata
```

Two methods of specifying a fissile solution

The standard composition specification data offer flexibility in the choice of input data. This section illustrates two methods of specifying the same fissile solution.

Create a mixture 3 that is aqueous uranyl nitrate solution:

$UO_2(NO_3)_2$, solution density = 1.555 g cm^3/

0.2669 g U/g-soln., 0.415 g U/ cm^3; excess nitrate = 0.39 mg/g-soln

Uranium isotopic content: 92.6 wt % U-235 5.9 wt % U-238

1.0 wt % U-234 and 0.5 wt % U-236

The SCALE atomic weights used in this problem are listed as follows:

H 1.0078

O 15.999

N 14.0067

U-234 234.041

U-235 235.0439

U-236 236.0456

U-238 238.0508

Two methods of describing the uranyl nitrate solution will be demonstrated. Method 1 is more rigorous, and method 2 is easier and as accurate.

METHOD 1:

This method involves breaking the solution into its component parts [(HNO_3, $UO_2(NO_3)_2$, and H_2O)] and entering the basic standard composition specifications for each.

1. Calculate the density of the HNO_3 0.39×10^{-3} g NO_3/g soln × [(62.997 g HNO_3/mole HNO_3)/(61.990 g NO_3/mole NO_3)] × 1.555 g soln/ cm^3soln = 6.16×10^{-4} g HNO_3/cc soln.
2. Calculate the density fraction of HNO_3 (actual density/theoretical density). In the Standard Composition Library the theoretical density of HNO_3 is 1.0. $6.16 \times 10^{-4}/1.0 = 6.16 \times 10^{-4}$.
3. Calculate the molecular weight of the uranium

The number of atoms in a mole of uranium is the sum of the number of atoms of each isotope in the mole of uranium.

Let AU = the average molecular weight of uranium, g U/mole U

GU = the density of uranium in g/cm^3.

Then the number of atoms in a mol of uranium =

$(6.023 \times 10^{+23} * 10^{-24} * GU)/AU$

or 0.6023 * GU/AU.

The weight fraction of each isotope is the weight % * 100.

Therefore, F235 = 0.926, the weight fraction of U-235 in the U

F238 = 0.059, the weight fraction of U-238 in the U

F236 = 0.005, the weight fraction of U-236 in the U

F234 = 0.010, the weight fraction of U-234 in the U

A235 = 235.0442, the molecular weight of U-235

A238 = 238.0510, the molecular weight of U-238

A236 = 236.0458, the molecular weight of U-236

A234 = 234.0406, the molecular weight of U-234.

Then the number of atoms of isotopes in a mol of uranium =

$6.023 \times 10^{+23} * 10^{-24} * ((GU*F235/A235) + (GU*F238/A238) +$

GU*F236/A236) + (GU*F234/A234))

or

0.6023*GU * (0.926/235.0442 + 0.059/238.0510 +

0.005/236.0458 + 0.010/234.0406).

Because the number of atoms of uranium equals the sum of the atoms of isotopes,

0.6023 * GU/AU = 0.6023 * GU *(0.926/235.0442 + 0.059/238.0510 +

0.005/236.0458 + 0.010/234.0406)

1/AU = 0.926/235.0442 + 0.059/238.0510 + 0.005/236.0458 + 0.010/234.0406

AU = 235.2144.

4. Calculate the molecular weight of the $UO_2(NO_3)_2$.

   $235.2144 + (8 \times 15.9954) + (2 \times 14.0033) = 391.184$ g $UO_2(NO_3)_2$/mole

5. Calculate the density of $UO_2(NO_3)_2$

   0.415 g U/cc × [(391.184 g $UO_2(NO_3)_2$/mol)/(235.2144 g U/mole)] =

   0.69018 g $UO_2(NO_3)_2$/ cm^3.soln.

Calculate the density fraction (actual density/theoretical density) of $UO_2(NO_3)_2$.

[In the Standard Composition Library the theoretical density of $UO_2(NO_3)_2$ is given as 2.2030 g/cm^3.]

The density fraction is 0.69018/2.2030 = 0.31329.

6. Calculate the amount of water in the solution

   1.555 g soln/ cm^3. soln - 6.16 × 10^{-4} g HNO_3/cm^3 soln - 0.69018 g $UO_2(NO_3)_2$LL/ cm^3. soln = 0.8642 g H_2O/cc soln.

7. Calculate the density fraction (actual density/theoretical density) of water.

```
HNO3        3   6.16-4   293  END
UO2(NO3)2   3  .31329  293  92235  92.6  92238  5.9  92234  1.0
                   92236   0.5  END
H2O         3    .86575  293  END
```

METHOD 2:

This method utilizes the solution option available in the standard composition specification data. Because the density is specified in the input data, this method should yield correct number densities that should agree with method 1 except for calculational round-off.

1. Calculate the fuel density

   0.415 g U/cc is 415 g U/L.

2. The molecular weight of nitrate NO_3 is 61.9895.

3. Calculate the molarity of the solution.

   0.39 mg nitrate/g soln × 1000 cm^{3}soln/L soln × 1 g/1000 mg × 1.555 g soln/ cm^{3}soln = 0.60645 g excess nitrate/L soln.

   A 1-molar solution is 1 mole of acid/L of solution:

   (For nitric acid 1 molar is 1 normal because there is only one atom of hydrogen per molecule of acid in HNO_3.)

   (0.60645 g nitrate/L soln)/(61.9895 g NO_3/mole NO_3) = 9.783 × 10^{-3} mole nitrate/L is identical to mole of acid/L, which is identical to molarity.

4. The density fraction of the solution is 1.0. Do not try to use the density of the solution divided by the theoretical density of $UO_2(NO_3)_2$ from the Standard Composition Library for your density multiplier. The $UO_2(NO_3)_2$ listed there is the solid, not the solution.

   The solution specification data follow:

```
SOLUTION      MIX=1   RHO[UO2(NO3)2] = 415    92235   92.6    92238   5.9
                      92234    1.0    92236   0.5
              MOLAR [HNO3] = 9.783-3
              TEMP = 293      DENSITY = 1.555 END SOLUTION
```

Comparison of number densities from the two methods

The number densities of methods 1 and 2 should agree within the limits of the input data. The density multipliers in method 1 are 5 digits and the density multipliers in method 2 are 4 digits. Therefore, the number densities calculated by the two methods should agree to 4 or 5 digits.

	Method 1	Method 2
Nuclide number	Atom density	Atom density
92235	9.84603E-04	9.84603E-04
92238	6.19415E-05	6.19415E-05
92234	1.06784E-05	1.06784E-05
92236	5.29387E-06	5.29387E-06
07014	2.13092E-03	2.13092E-03
08016	3.74135E-02	3.7410E-02
01001	5.77973E-02	5.77983E-02

Multiple unit cells in a single problem

Consider a problem that involves three different UO_2 fuel assemblies: a 1.98%-enriched assembly, a 2.64%-enriched assembly, and a 2.96%-enriched assembly. All fuel rods are UO_2 at 10.138 g/cm^3 and are 0.94 cm in diameter. The Zircaloy-4 clad has an inside radius of 0.4875 cm and an outside radius of 0.545 cm. The rod pitch is 1.44 cm. Each fuel assembly is a 15 × 15 array of fuel pins with water holes, instrumentation holes, and burnable poison rods. For cross-section processing, the presence of the water holes, instrumentation holes, and burnable poison rods in the assemblies are ignored.

The following XSProc input use the CENTRM/PMC method for self-shielding three latticecells with different fuel enrichments. The remaining mixture (SS-304), not specified in a unit cell, is processed as an infinite homogeneous medium using the BONAMI method. Each mixture can appear only in a single zone of one unit cell. For square pitch latticecells the default CENTRM transport solver is MoC with P0 scatter; however in this input, the solver for the 3rd cell is modified through CENTRM DATA to use the two-region approximation for the CE calculation [npxs=5], and discrete S_N transport calculation with P1 anisotropic scatteringfor the MG solutions in the fast and thermal energy ranges [nfst=0, nthr=0].

```
DEMONSTRATION PROBLEM WITH MULTIPLE RESONANCE CORRECTIONS REQUIRED
broad_n
READ COMP
UO2        1  .925    300  92235  1.98  92238  98.02  END
UO2        2  .925    300  92235  2.64  92238  97.36  END
UO2        3  .925    300  92235  2.96  92238  97.04  END
ZIRC4      4  1.0     300  END
H2O        5  1.0     300  END
ZIRC4      6  1.0     300  END
H2O        7  1.0     300  END
ZIRC4      8  1.0     300  END
H2O        9  1.0     300  END
SS304     10  1.0     300  END
END  COMP
READ CELLDATA
LATTICECELL SQUAREPITCH PITCH=1.44 5 FUELD=0.94  1 CLADD=1.09  4 GAPD=0.975  0 END
LATTICECELL SQUAREPITCH PITCH=1.44 7 FUELD=0.94  2 CLADD=1.09  6 GAPD=0.975  0 END
LATTICECELL SQUAREPITCH PITCH=1.44 9 FUELD=0.94  3 CLADD=1.09  8 GAPD=0.975  0 END
CENTRM DATA  npxs=5 nthr=0 nfst=0 isct=1    END CENTRM DATA
END CELLDATA
```

Multiple fissile mixtures in a single unit cell

The following problem involves large units having the bulk of their fissile material more than one mean-free path away from the surface of the unit. The interaction between the units that occurs in the resonance range is a very small fraction of the total interaction because an overwhelming percentage of the interaction occurs deep within each unit. Therefore, the resonance range interaction between the units can be ignored, and the default infinite homogeneous medium cross-section processing in the resonance range can be considered adequate for this particular application.

Consider a problem that consists of four 20.96-kg 93.2%-enriched uranium metal cylinders, density 18.76 g/cm^3, and four 5-liters Plexiglas bottles filled with highly enriched uranyl nitrate solution at 415 g/L, a specific gravity of 1.555, and 0.39 mg of excess nitrate per gram of solution. The isotopic content of the uranium metal is 93.2 wt % ^{235}U, 5.6 wt % ^{238}U, 1.0 wt % ^{234}U, and 0.2 wt % ^{236}U. The uranium isotopic content of the nitrate solution is 92.6 wt % ^{235}U, 5.9 wt % ^{238}U, 1.0 wt % ^{234}U and 0.5 wt % ^{236}U. The size of the metal cylinders is between 10 and 12 cm in diameter and height, and the size of the nitrate solution is on the order of 16 and 20 cm in diameter and height. The average mean-free path in the uranium metal is on the order of 1.5 cm, and the average mean free path in the nitrate solution is on the order of 0.5 cm. Therefore, infinite homogeneous medium is an appropriate choice for this problem and the use of CENTRM/PMC is valid.

See Examples 1–4 of Sect. 7.1.4.1.2 for data input details for the Plexiglas and uranium metal. See Example 1 of Sect. 7.1.4.1.5 for data input details for the uranyl nitrate solution. The XSProc data for this problem follow:

```
SET  UP  4 AQUEOUS  4  METAL
fine_n
READ COMP
URANIUM  1  0.985  293  92235  93.2  92238  5.6  92234  1.0  92236  0.2  END
SOLUTION 2  RHO[UO2(NO3)2]=415  92235 92.6 92238 5.9 92234 1.0 92236 0.5
            MOLAR[HNO3]=9.783-3  DENSITY=1.555  TEMPERATURE=293  END SOLUTION
PLEXIGLAS 3  END
END COMP
```

Consider the same materials above except rearrange them so that a 10 cm diameter uranium metal sphere sits inside a 50 cm diameter spherical tank of uranyl nitrate solution having a 1-cm thick Plexiglas wall. This problem can be modeled in SCALE but only CENTRM/PMC will treat the resonance processing correctly. This problem is modeled below.

```
SET  UP  4 AQUEOUS  4  METAL
fine_n
READ COMP
URANIUM   1  0.985  293   92235  93.2  92238  5.6  92234  1.0  92236  0.2  END
SOLUTION  2  RHO[UO2(NO3)2]=415  92235 92.6 92238 5.9 92234 1.0 92236 0.5
             MOLAR[HNO3]=9.783-3  DENSITY=1.555  TEMPERATURE=293  END SOLUTION
PLEXIGLAS  3  END
END  COMP
READ CELLDATA
MULTIREGION SPHERICAL END 1 5.0 2 25.0 3 26.0 END ZONE
END CELLDATA
```

Cell weighting an infinite homogeneous problem

Cell weighting an infinite homogeneous medium has no effect on the cross sections because there is only one zone and one set of cross sections. However, a cell-weighted mixture number can still be supplied using the keyword **CELLMIX**= followed by an unique mixture number. This cell-weighted mixture number can be used in subsequent codes and will produce results similar to the cross sections of the original mixture.

EXAMPLE 1

This problem would probably be run with CSAS1 to provide the k-infinity of 20%-enriched UO_2.

```
20%  ENRICHED  UO2  BILLET
fine_n
READ COMP
UO2  1  0.99  293  92235  20  92238  80  END
END COMP
READ CELLDATA
INFHOMMEDIUM  1  CELLMIX=100  END
END CELLDATA
```

Cell weighting a LATTICECELL problem

Cell weighting used with a **LATTICECELL** problem creates cell-weighted homogeneous cross sections that represent the characteristics of the heterogeneous unit cell. This cell-weighted mixture can then be used in a subsequent code for the overall volume where the cells are located without having to mock up the actual 3-D heterogeneous array of cells. This cell-weighted homogeneous mixture is designated by the user with the keyword **CELLMIX**= immediately followed by an unused mixture number. This needs to follow immediately after the cell description. Note that the mixtures used in the unit cell data cannot be used in a subsequent code because they have been flux weighted to create the user specified mixture. Therefore, if a mixture used in the unit cell description is also to be used in a subsequent code, another mixture must be created that is identical except for the mixture number. Every mixture that is to be used in a subsequent code except zero (i.e., void) must be defined in the standard composition data.

A byproduct of the cell-weighting calculation is the eigenvalue (k-effective) of an infinite array of the cell described as the unit cell.

EXAMPLE 1

Consider a cylindrical stainless steel tank filled with spherical pellets of 2.67%-enriched UO_2 arranged in a close-packed "triangular" pitch, flooded with borated water at 4350 ppm. The cylindrical stainless tank is sitting in a larger tank filled with borated water at 4350 ppm.

The data for the UO_2 and borated water were developed in detail in Example 3 of Sect. 7.1.4.3.2. The stainless steel must be defined, and mixture 3 was chosen because mixture 1 was the UO_2 and mixture 2 was the borated water. Because the borated water will be used as a reflector for the stainless steel tank and has been used in the unit cell data, it must be repeated with a different mixture number (in this case, as mixture 4).

In the subsequent calculation, user specified cell mixture 100 will be used to represent the UO_2 pellets in the borated water, mixture 3 will represent the stainless steel tank, and mixture 4 will represent the borated water reflector around the stainless-steel tank.

The XSProc data for creating the cell-weighted cross sections on mixture 100 follow:

```
SPHERICAL  PELLETS  IN  BORATED  WATER
fine_n
READ COMP
UO2        1  .9398  293.  92235  2.67  92238  97.33  END
ATOMH3BO3  2  0.025066  3  5000  1  1001  3  8016  3  1.0  293  END
H2O        2  0.984507  293  END
SS304      3  1.0  293  END
ATOMH3BO3  4  0.025066  3  5000  1  1001  3  8016  3  1.0  293  END
H2O        4  0.984507  293  END
END  COMP
READ CELLDATA
LATTICECELL  SPHTRIANGP   PITCH  1.0724  2  FUELD  1.0724  1  CELLMIX=100  END
END CELLDATA
```

Cell weighting a MULTIREGION problem

A **MULTIREGION** problem is cell weighted primarily to obtain a cell-weighted homogeneous cross section that represents the characteristics of the heterogeneous unit cell. The eigenvalue obtained for a **MULTIREGION** problem with cylindrical or spherical geometry having a white boundary condition specified on the right boundary approximates an infinite array of the cells. A vacuum boundary condition would represent a single cell. A slab with reflected boundary conditions for both boundaries represents an infinite array of slab cells. The cell-weighted cross sections for spherical or cylindrical geometries with a white right boundary condition do not use a Dancoff correction and thus may not be accurate for representing a large array of the specified units.

EXAMPLE 1

Consider a small, highly enriched uranium sphere supported by a Plexiglas collar in a tank of water. The uranium metal sphere has a diameter of 13.1075 cm, is 97.67% enriched, and has a density of 18.794 g/cm^3. The cylindrical Plexiglas collar has a 4.1275-cm radius central hole, extends to a radius of 12.7 cm and is 2.54 cm thick. The water-filled tank is 60 cm in diameter.

The Plexiglas collar is not significantly different from water and does not surround the fuel, so it will be ignored. Because this makes the problem a 1-D geometry, it can be defined using the **MULTIREGION** type of calculation and the eigenvalue of the system can be obtained without additional data by executing CSAS1 with CENTRM/PMC, if PARM=CENTRM is specified on the command line. The abundance of uranium is not stated beyond 97.67% enriched, so assume the remainder is ^{238}U. The XSProc data follow:

```
=CSAS5
SMALL  WATER  REFLECTED  SPHERE  ON  PLEXIGLAS  COLLAR
fine_n
READ COMP
URANIUM    1  DEN=18.794  1  293.  92235  97.67  92238  2.33  END
H2O        2  END
END  COMP
READ CELLDATA
MULTI SPHERICAL CELLMIX=100  END   1  6.5537  2  30.0  END ZONE
END CELLDATA
•
•
•
KENO DATA THAT USES MIX=100 FOR A HOMOGENEOUS SPHERE OF 30-CM RADIUS GOES HERE.
•
•
END
```

7.2 STANDARD COMPOSITION LIBRARY

L. M. Petrie, R. A. Lefebvre, and D. Wiarda

ABSTRACT

The SCALE Standard Composition Library provides a flexible and convenient means of generating models that include many types of materials. Users may specify materials as individual nuclides; elements with tabulated natural abundances; compounds, alloys, mixtures, and fissile solutions commonly encountered in engineering practice.

ACKNOWLEDGMENTS

This work was originally funded by the U.S. Nuclear Regulatory Commission through the Office of Nuclear Material Safety and Safeguards and the Office of Nuclear Regulatory Research. Continuing support is provided by the U.S. Department of Energy Nuclear Criticality Safety Program.

7.2.1 INTRODUCTION

The Standard Composition Library has been included within the SCALE system to provide the user with a simple and straightforward method of specifying the material mixtures for a given problem. The library consists of over 700 mixtures and isotopes commonly used within criticality safety, shielding, and reactor physics models. This library is tabulated in Sect. 7.2.2 with the various features of each composition. Additionally, a number of standard fissile solutions available for use are tabulated in Sect. 7.2.3.

The Standard Composition Library is created using the COMPOZ module, and can be updated by users as desired.

7.2.2 THE STANDARD COMPOSITION LIBRARY

The Standard Composition Library describes the various isotopes, elements (both symbols and full name), and compounds/alloys that can be used to define the material mixtures for a given problem. Typically, the alphanumeric description of one or more of these materials will be used to define a material mixture.

When formulating a mixture, it is often necessary to know the density (g/cm^3) of the various constituent materials. For convenience, the reference values given in the library have been listed in Table 7.2.2 through Table 7.2.5. Note that the given reference values represent the actual theoretical density, except in the case of isotopes and some individual nuclides where a default value of 1.0 g/cm^3 is used. The actual theoretical densities are fixed values at naturally occurring or nominal conditions. These default densities should not be used for materials containing enriched isotopes, especially light elements with isotopes that are strong absorbers such as boron, B_4C, or lithium. When densities are recalculated by other codes such as KENO, the recalculated densities may differ slightly from those given in the Standard Composition Library depending on the cross-section library specified in the calculation. Note that the recalculated mixture densities are presented for information purposes and are not used in the calculations.

Note that not all nuclides in the Standard Composition Library are available on each cross-section library, and user are encourage to review the code output for warning messages regarding composition data. Refer to the cross section library chapter for listings of available nuclides in each cross-section library.

Multiple sets of iron, nickel, and chromium nuclides are available in the Standard Composition Library. These sets correspond to different weighting functions used in generating the multigroup cross sections. One special weighting function corresponds to $1/[E\ \sigma_t(E)]$, where $\sigma_t(E)$ is the total cross section of the referenced nuclide or alloy. Entries have been added to the isotopic distribution table so that standard weighted isotopes will be requested if the desired nuclide is not on the specified library.

The nuclide identifying numbers (IDs) are listed in Table 7.2.1. Typically, the ID is 1000*Z + A, where Z and A are the charge and mass numbers for the nuclide (e.g., 1001 for ^{1}H and 8016 for ^{16}O). Exceptions to this rule include metastable nuclides, nuclides with bound thermal scattering data, and nuclides whose cross sections have a special weighting. Also, elements with isotopic mixtures (typically natural abundance) have IDs of Z* 1000.

If a nuclide identifier is listed in Table 7.2.1, it can be accessed and used in a user-specified material (a.k.a., arbitrary material). User-specified materials require the user to provide all the information normally found in the Standard Composition Library. Refer to the XSProc manual for details on how to input user-specified materials.

Several materials contain multiple isotopes of a single element. For these materials, the user is free to specify the isotopic distribution as discussed in the XSProc chapter. Alternatively, the user may elect not to enter this data, thereby telling the code to assume the default values shown in the tables. In describing a user-specified material, a multiple isotope ID of Z * 1000 can be used to denote the elements of Table 7.2.2. Note that even if the natural abundances from Table 7.2.2 are accessed through elemental specification, not all nuclides are necessarily present on each cross-section library. Refer to the Cross Section Library chapter for listings of available nuclides in each cross-section library.

Atomic masses for the isotopes were taken from "The Ame2003 atomic mass evaluation (II)" by G. Audi et al. [STDCOMPAWT03]. The atom percents of the isotopic distribution table were taken from "Isotopic Compositions of the Elements, 2001" by J. K. Bohlke [STDCOMPBDLDB+05]. Densities were taken from several sources, including the CRC Handbook of Chemistry and Physics [STDCOMPWea01]. Gases and explicit isotopes were changed to all have a theoretical density of 1.0 g/cm^3.

To more fully document the composition of each compound and/or document the assumptions used in producing the associated cross-section data, a brief description is given in the tables where needed.

Table 7.2.1: Isotopes in standard composition library.

Name	Identifier	Mass (amu)
h-1	1001	1.0078
h-2	1002	2.0141
h-3	1003	3.0161
h-4	1004	4.0278
h-5	1005	5.0353
h-6	1006	6.0449
h-7	1007	7.0528
he-3	2003	3.016
he-4	2004	4.0026
he-5	2005	5.0122
he-6	2006	6.0189
he-7	2007	7.028
he-8	2008	8.0339
he-9	2009	9.044
he-10	2010	10.0524
li-3	3003	3.0308
li-4	3004	4.0272
li-5	3005	5.0125

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
li-6	3006	6.0151
li-7	3007	7.016
li-8	3008	8.0225
li-9	3009	9.0268
li-10	3010	10.0355
li-11	3011	11.0438
li-12	3012	12.0538
be-5	4005	5.0408
be-6	4006	6.0197
be-7	4007	7.0169
be-8	4008	8.0053
be-9	4009	9.0122
be-10	4010	10.0135
be-11	4011	11.0217
be-12	4012	12.0269
be-13	4013	13.0357
be-14	4014	14.0429
be-15	4015	15.0535
be-16	4016	16.0619
b-6	5006	6.0468
b-7	5007	7.0299
b-8	5008	8.0246
b-9	5009	9.0133
b-10	5010	10.0129
b-11	5011	11.0093
b-12	5012	12.0143
b-13	5013	13.0178
b-14	5014	14.0254
b-15	5015	15.0311
b-16	5016	16.0398
b-17	5017	17.047
b-18	5018	18.0562
b-19	5019	19.0637
c-8	6008	8.0377
c-9	6009	9.031
c-10	6010	10.0169
c-11	6011	11.0114
c-12	6012	12.0
c-13	6013	13.0034
c-14	6014	14.0032
c-15	6015	15.0106
c-16	6016	16.0147
c-17	6017	17.0226

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
c-18	6018	18.0268
c-19	6019	19.0348
c-20	6020	20.0403
c-21	6021	21.0493
c-22	6022	22.0572
n-10	7010	10.0416
n-11	7011	11.0261
n-12	7012	12.0186
n-13	7013	13.0057
n-14	7014	14.0031
n-15	7015	15.0001
n-16	7016	16.0061
n-17	7017	17.0084
n-18	7018	18.0141
n-19	7019	19.017
n-20	7020	20.0234
n-21	7021	21.0271
n-22	7022	22.0344
n-23	7023	23.0412
n-24	7024	24.051
n-25	7025	25.0607
o-12	8012	12.0344
o-13	8013	13.0248
o-14	8014	14.0086
o-15	8015	15.0031
o-16	8016	15.9949
o-17	8017	16.9991
o-18	8018	17.9992
o-19	8019	19.0036
o-20	8020	20.0041
o-21	8021	21.0087
o-22	8022	22.01
o-23	8023	23.0157
o-24	8024	24.0205
o-25	8025	25.0295
o-26	8026	26.0383
o-27	8027	27.0483
o-28	8028	28.0578
f-14	9014	14.0351
f-15	9015	15.018
f-16	9016	16.0115
f-17	9017	17.0021
f-18	9018	18.0009

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
f-19	9019	18.9984
f-20	9020	20.0
f-21	9021	21.0
f-22	9022	22.003
f-23	9023	23.0036
f-24	9024	24.0081
f-25	9025	25.0121
f-26	9026	26.0196
f-27	9027	27.0268
f-28	9028	28.0357
f-29	9029	29.0433
f-30	9030	30.0525
f-31	9031	31.0604
ne-16	10016	16.0258
ne-17	10017	17.0177
ne-18	10018	18.0057
ne-19	10019	19.0019
ne-20	10020	19.9924
ne-21	10021	20.9939
ne-22	10022	21.9914
ne-23	10023	22.9945
ne-24	10024	23.9936
ne-25	10025	24.9977
ne-26	10026	26.0005
ne-27	10027	27.0076
ne-28	10028	28.0121
ne-29	10029	29.0194
ne-30	10030	30.0248
ne-31	10031	31.0331
ne-32	10032	32.04
ne-33	10033	33.0494
ne-34	10034	34.057
na-18	11018	18.026
na-19	11019	19.0139
na-20	11020	20.0074
na-21	11021	20.9977
na-22	11022	21.9944
na-23	11023	22.9898
na-24	11024	23.991
na-25	11025	24.99
na-26	11026	25.9926
na-27	11027	26.9941
na-28	11028	27.9989

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
na-29	11029	29.0029
na-30	11030	30.009
na-31	11031	31.0136
na-32	11032	32.0205
na-33	11033	33.0267
na-34	11034	34.0352
na-35	11035	35.0425
na-36	11036	36.0515
na-37	11037	37.0593
mg-19	12019	19.0355
mg-20	12020	20.0189
mg-21	12021	21.0117
mg-22	12022	21.9996
mg-23	12023	22.9941
mg-24	12024	23.985
mg-25	12025	24.9858
mg-26	12026	25.9826
mg-27	12027	26.9843
mg-28	12028	27.9839
mg-29	12029	28.9886
mg-30	12030	29.9904
mg-31	12031	30.9965
mg-32	12032	31.999
mg-33	12033	33.0052
mg-34	12034	34.0095
mg-35	12035	35.0173
mg-36	12036	36.023
mg-37	12037	37.0314
mg-38	12038	38.0376
mg-39	12039	39.0468
mg-40	12040	40.0539
al-21	13021	21.028
al-22	13022	22.0195
al-23	13023	23.0073
al-24	13024	23.9999
al-25	13025	24.9904
al-26	13026	25.9869
al-27	13027	26.9815
al-28	13028	27.9819
al-29	13029	28.9804
al-30	13030	29.983
al-31	13031	30.9839
al-32	13032	31.9881

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
al-33	13033	32.9908
al-34	13034	33.9968
al-35	13035	34.9999
al-36	13036	36.0062
al-37	13037	37.0107
al-38	13038	38.0172
al-39	13039	39.023
al-40	13040	40.0314
al-41	13041	41.0383
al-42	13042	42.0469
si-22	14022	22.0345
si-23	14023	23.0255
si-24	14024	24.0116
si-25	14025	25.0041
si-26	14026	25.9923
si-27	14027	26.9867
si-28	14028	27.9769
si-29	14029	28.9765
si-30	14030	29.9738
si-31	14031	30.9754
si-32	14032	31.9741
si-33	14033	32.978
si-34	14034	33.9786
si-35	14035	34.9846
si-36	14036	35.9866
si-37	14037	36.9929
si-38	14038	37.9956
si-39	14039	39.0021
si-40	14040	40.0059
si-41	14041	41.0146
si-42	14042	42.0198
si-43	14043	43.0287
si-44	14044	44.0353
p-24	15024	24.0343
p-25	15025	25.0203
p-26	15026	26.0118
p-27	15027	26.9992
p-28	15028	27.9923
p-29	15029	28.9818
p-30	15030	29.9783
p-31	15031	30.9738
p-32	15032	31.9739
p-33	15033	32.9717

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
p-34	15034	33.9736
p-35	15035	34.9733
p-36	15036	35.9783
p-37	15037	36.9796
p-38	15038	37.9842
p-39	15039	38.9862
p-40	15040	39.9913
p-41	15041	40.9943
p-42	15042	42.001
p-43	15043	43.0062
p-44	15044	44.013
p-45	15045	45.0192
p-46	15046	46.0274
s-26	16026	26.0279
s-27	16027	27.0188
s-28	16028	28.0044
s-29	16029	28.9966
s-30	16030	29.9849
s-31	16031	30.9795
s-32	16032	31.9721
s-33	16033	32.9715
s-34	16034	33.9679
s-35	16035	34.969
s-36	16036	35.9671
s-37	16037	36.9711
s-38	16038	37.9712
s-39	16039	38.9751
s-40	16040	39.9754
s-41	16041	40.9796
s-42	16042	41.981
s-43	16043	42.9871
s-44	16044	43.9902
s-45	16045	44.9965
s-46	16046	46.0008
s-47	16047	47.0086
s-48	16048	48.0142
s-49	16049	49.0236
cl-28	17028	28.0285
cl-29	17029	29.0141
cl-30	17030	30.0048
cl-31	17031	30.9924
cl-32	17032	31.9857
cl-33	17033	32.9775

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
cl-34	17034	33.9738
cl-35	17035	34.9688
cl-36	17036	35.9683
cl-37	17037	36.9659
cl-38	17038	37.968
cl-39	17039	38.968
cl-40	17040	39.9704
cl-41	17041	40.9707
cl-42	17042	41.9733
cl-43	17043	42.974
cl-44	17044	43.9783
cl-45	17045	44.9803
cl-46	17046	45.9842
cl-47	17047	46.9887
cl-48	17048	47.9949
cl-49	17049	49.0003
cl-50	17050	50.0078
cl-51	17051	51.0145
ar-30	18030	30.0216
ar-31	18031	31.0121
ar-32	18032	31.9976
ar-33	18033	32.9899
ar-34	18034	33.9803
ar-35	18035	34.9753
ar-36	18036	35.9675
ar-37	18037	36.9668
ar-38	18038	37.9627
ar-39	18039	38.9643
ar-40	18040	39.9624
ar-41	18041	40.9645
ar-42	18042	41.9631
ar-43	18043	42.9656
ar-44	18044	43.9649
ar-45	18045	44.968
ar-46	18046	45.9681
ar-47	18047	46.9722
ar-48	18048	47.9745
ar-49	18049	48.9805
ar-50	18050	49.9844
ar-51	18051	50.9916
ar-52	18052	51.9968
ar-53	18053	53.0049
k-32	19032	32.0219

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
k-33	19033	33.0073
k-34	19034	33.9984
k-35	19035	34.988
k-36	19036	35.9813
k-37	19037	36.9734
k-38	19038	37.9691
k-39	19039	38.9637
k-40	19040	39.964
k-41	19041	40.9618
k-42	19042	41.9624
k-43	19043	42.9607
k-44	19044	43.9616
k-45	19045	44.9607
k-46	19046	45.962
k-47	19047	46.9617
k-48	19048	47.9655
k-49	19049	48.9674
k-50	19050	49.9728
k-51	19051	50.9764
k-52	19052	51.9826
k-53	19053	52.9871
k-54	19054	53.9942
k-55	19055	54.9997
ca-34	20034	34.0141
ca-35	20035	35.0049
ca-36	20036	35.9931
ca-37	20037	36.9859
ca-38	20038	37.9763
ca-39	20039	38.9707
ca-40	20040	39.9626
ca-41	20041	40.9623
ca-42	20042	41.9586
ca-43	20043	42.9588
ca-44	20044	43.9555
ca-45	20045	44.9562
ca-46	20046	45.9537
ca-47	20047	46.9546
ca-48	20048	47.9525
ca-49	20049	48.9557
ca-50	20050	49.9575
ca-51	20051	50.9615
ca-52	20052	51.9651
ca-53	20053	52.9701

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ca-54	20054	53.9743
ca-55	20055	54.9805
ca-56	20056	55.9856
ca-57	20057	56.9924
sc-36	21036	36.0149
sc-37	21037	37.0031
sc-38	21038	37.9947
sc-39	21039	38.9848
sc-40	21040	39.978
sc-41	21041	40.9692
sc-42	21042	41.9655
sc-43	21043	42.9612
sc-44	21044	43.9594
sc-45	21045	44.9559
sc-46	21046	45.9552
sc-47	21047	46.9524
sc-48	21048	47.9522
sc-49	21049	48.95
sc-50	21050	49.9522
sc-51	21051	50.9536
sc-52	21052	51.9567
sc-53	21053	52.9596
sc-54	21054	53.9633
sc-55	21055	54.9682
sc-56	21056	55.9729
sc-57	21057	56.9778
sc-58	21058	57.9837
sc-59	21059	58.9892
sc-60	21060	59.9957
ti-38	22038	38.0098
ti-39	22039	39.0016
ti-40	22040	39.9905
ti-41	22041	40.9832
ti-42	22042	41.973
ti-43	22043	42.9685
ti-44	22044	43.9597
ti-45	22045	44.9581
ti-46	22046	45.9526
ti-47	22047	46.9518
ti-48	22048	47.9479
ti-49	22049	48.9479
ti-50	22050	49.9448
ti-51	22051	50.9466

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ti-52	22052	51.9469
ti-53	22053	52.9497
ti-54	22054	53.951
ti-55	22055	54.9553
ti-56	22056	55.9582
ti-57	22057	56.964
ti-58	22058	57.967
ti-59	22059	58.9729
ti-60	22060	59.9768
ti-61	22061	60.9832
ti-62	22062	61.9875
ti-63	22063	62.9944
v-40	23040	40.0111
v-41	23041	40.9998
v-42	23042	41.9912
v-43	23043	42.9807
v-44	23044	43.9741
v-45	23045	44.9658
v-46	23046	45.9602
v-47	23047	46.9549
v-48	23048	47.9523
v-49	23049	48.9485
v-50	23050	49.9472
v-51	23051	50.944
v-52	23052	51.9448
v-53	23053	52.9443
v-54	23054	53.9464
v-55	23055	54.9472
v-56	23056	55.9505
v-57	23057	56.9526
v-58	23058	57.9568
v-59	23059	58.9602
v-60	23060	59.965
v-61	23061	60.9685
v-62	23062	61.9738
v-63	23063	62.9776
v-64	23064	63.9835
v-65	23065	64.9879
cr-42	24042	42.0064
cr-43	24043	42.9977
cr-44	24044	43.9855
cr-45	24045	44.9796
cr-46	24046	45.9684

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
cr-47	24047	46.9629
cr-48	24048	47.954
cr-49	24049	48.9513
cr-50	24050	49.946
cr-51	24051	50.9448
cr-52	24052	51.9405
cr-53	24053	52.9407
cr-54	24054	53.9389
cr-55	24055	54.9408
cr-56	24056	55.9407
cr-57	24057	56.9436
cr-58	24058	57.9444
cr-59	24059	58.9486
cr-60	24060	59.9501
cr-61	24061	60.9547
cr-62	24062	61.9566
cr-63	24063	62.9619
cr-64	24064	63.9644
cr-65	24065	64.9702
cr-66	24066	65.9734
cr-67	24067	66.9796
mn-44	25044	44.0069
mn-45	25045	44.9945
mn-46	25046	45.9867
mn-47	25047	46.9761
mn-48	25048	47.9685
mn-49	25049	48.9596
mn-50	25050	49.9542
mn-51	25051	50.9482
mn-52	25052	51.9456
mn-53	25053	52.9413
mn-54	25054	53.9404
mn-55	25055	54.938
mn-56	25056	55.9389
mn-57	25057	56.9383
mn-58	25058	57.94
mn-59	25059	58.9404
mn-60	25060	59.9429
mn-61	25061	60.9446
mn-62	25062	61.9484
mn-63	25063	62.9502
mn-64	25064	63.9543
mn-65	25065	64.9563

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
mn-66	25066	65.9611
mn-67	25067	66.9641
mn-68	25068	67.9693
mn-69	25069	68.9728
fe-45	26045	45.0146
fe-46	26046	46.0008
fe-47	26047	46.9929
fe-48	26048	47.9805
fe-49	26049	48.9736
fe-50	26050	49.963
fe-51	26051	50.9568
fe-52	26052	51.9481
fe-53	26053	52.9453
fe-54	26054	53.9396
fe-55	26055	54.9383
fe-56	26056	55.9349
fe-57	26057	56.9354
fe-58	26058	57.9333
fe-59	26059	58.9349
fe-60	26060	59.9341
fe-61	26061	60.9367
fe-62	26062	61.9368
fe-63	26063	62.9404
fe-64	26064	63.9412
fe-65	26065	64.9454
fe-66	26066	65.9468
fe-67	26067	66.951
fe-68	26068	67.9537
fe-69	26069	68.9588
fe-70	26070	69.9615
fe-71	26071	70.9667
fe-72	26072	71.9696
co-47	27047	47.0115
co-48	27048	48.0018
co-49	27049	48.9897
co-50	27050	49.9815
co-51	27051	50.9707
co-52	27052	51.9636
co-53	27053	52.9542
co-54	27054	53.9485
co-55	27055	54.942
co-56	27056	55.9398
co-57	27057	56.9363

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
co-58	27058	57.9357
co-58m	1027058	
co-59	27059	58.9332
co-60	27060	59.9338
co-61	27061	60.9325
co-62	27062	61.9341
co-63	27063	62.9336
co-64	27064	63.9358
co-65	27065	64.9365
co-66	27066	65.9398
co-67	27067	66.9409
co-68	27068	67.9449
co-69	27069	68.9463
co-70	27070	69.951
co-71	27071	70.9529
co-72	27072	71.9578
co-73	27073	72.9602
co-74	27074	73.9654
co-75	27075	74.9683
ni-48	28048	48.0197
ni-49	28049	49.0097
ni-50	28050	49.9959
ni-51	28051	50.9877
ni-52	28052	51.9757
ni-53	28053	52.9685
ni-54	28054	53.9579
ni-55	28055	54.9513
ni-56	28056	55.9421
ni-57	28057	56.9398
ni-58	28058	57.9353
ni-59	28059	58.9343
ni-60	28060	59.9308
ni-61	28061	60.9311
ni-62	28062	61.9283
ni-63	28063	62.9297
ni-64	28064	63.928
ni-65	28065	64.9301
ni-66	28066	65.9291
ni-67	28067	66.9316
ni-68	28068	67.9319
ni-69	28069	68.9356
ni-70	28070	69.9365
ni-71	28071	70.9407

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ni-72	28072	71.9421
ni-73	28073	72.9465
ni-74	28074	73.9481
ni-75	28075	74.9529
ni-76	28076	75.9553
ni-77	28077	76.9605
ni-78	28078	77.9632
cu-52	29052	51.9972
cu-53	29053	52.9855
cu-54	29054	53.9767
cu-55	29055	54.966
cu-56	29056	55.9586
cu-57	29057	56.9492
cu-58	29058	57.9445
cu-59	29059	58.9395
cu-60	29060	59.9374
cu-61	29061	60.9335
cu-62	29062	61.9326
cu-63	29063	62.9296
cu-64	29064	63.9298
cu-65	29065	64.9278
cu-66	29066	65.9289
cu-67	29067	66.9277
cu-68	29068	67.9296
cu-69	29069	68.9294
cu-70	29070	69.9324
cu-71	29071	70.9327
cu-72	29072	71.9358
cu-73	29073	72.9367
cu-74	29074	73.9399
cu-75	29075	74.9419
cu-76	29076	75.9453
cu-77	29077	76.9479
cu-78	29078	77.952
cu-79	29079	78.9546
cu-80	29080	79.9609
zn-54	30054	53.993
zn-55	30055	54.984
zn-56	30056	55.9724
zn-57	30057	56.9648
zn-58	30058	57.9546
zn-59	30059	58.9493
zn-60	30060	59.9418

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
zn-61	30061	60.9395
zn-62	30062	61.9343
zn-63	30063	62.9332
zn-64	30064	63.9291
zn-65	30065	64.9292
zn-66	30066	65.926
zn-67	30067	66.9271
zn-68	30068	67.9248
zn-69	30069	68.9266
zn-70	30070	69.9253
zn-71	30071	70.9277
zn-72	30072	71.9269
zn-73	30073	72.9298
zn-74	30074	73.9295
zn-75	30075	74.9329
zn-76	30076	75.9333
zn-77	30077	76.937
zn-78	30078	77.9384
zn-79	30079	78.9426
zn-80	30080	79.9443
zn-81	30081	80.9505
zn-82	30082	81.9544
zn-83	30083	82.961
ga-56	31056	55.9949
ga-57	31057	56.9829
ga-58	31058	57.9743
ga-59	31059	58.9634
ga-60	31060	59.9571
ga-61	31061	60.9495
ga-62	31062	61.9442
ga-63	31063	62.9393
ga-64	31064	63.9368
ga-65	31065	64.9327
ga-66	31066	65.9316
ga-67	31067	66.9282
ga-68	31068	67.928
ga-69	31069	68.9256
ga-70	31070	69.926
ga-71	31071	70.9247
ga-72	31072	71.9264
ga-73	31073	72.9252
ga-74	31074	73.9269
ga-75	31075	74.9265

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ga-76	31076	75.9288
ga-77	31077	76.9292
ga-78	31078	77.9316
ga-79	31079	78.9329
ga-80	31080	79.9365
ga-81	31081	80.9378
ga-82	31082	81.943
ga-83	31083	82.947
ga-84	31084	83.9527
ga-85	31085	84.957
ga-86	31086	85.9631
ge-58	32058	57.991
ge-59	32059	58.9818
ge-60	32060	59.9702
ge-61	32061	60.9638
ge-62	32062	61.9547
ge-63	32063	62.9496
ge-64	32064	63.9417
ge-65	32065	64.9394
ge-66	32066	65.9338
ge-67	32067	66.9327
ge-68	32068	67.9281
ge-69	32069	68.928
ge-70	32070	69.9242
ge-71	32071	70.9249
ge-72	32072	71.9221
ge-73	32073	72.9235
ge-74	32074	73.9212
ge-75	32075	74.9229
ge-76	32076	75.9214
ge-77	32077	76.9236
ge-78	32078	77.9229
ge-79	32079	78.9254
ge-80	32080	79.9254
ge-81	32081	80.9288
ge-82	32082	81.9296
ge-83	32083	82.9346
ge-84	32084	83.9375
ge-85	32085	84.943
ge-86	32086	85.9465
ge-87	32087	86.9525
ge-88	32088	87.9569
ge-89	32089	88.9638

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
as-60	33060	59.9931
as-61	33061	60.9806
as-62	33062	61.9732
as-63	33063	62.9637
as-64	33064	63.9576
as-65	33065	64.9496
as-66	33066	65.9447
as-67	33067	66.9392
as-68	33068	67.9368
as-69	33069	68.9323
as-70	33070	69.9309
as-71	33071	70.9271
as-72	33072	71.9268
as-73	33073	72.9238
as-74	33074	73.9239
as-75	33075	74.9216
as-76	33076	75.9224
as-77	33077	76.9206
as-78	33078	77.9218
as-79	33079	78.921
as-80	33080	79.9225
as-81	33081	80.9221
as-82	33082	81.9245
as-83	33083	82.925
as-84	33084	83.9291
as-85	33085	84.932
as-86	33086	85.9365
as-87	33087	86.9399
as-88	33088	87.9449
as-89	33089	88.9494
as-90	33090	89.9555
as-91	33091	90.9604
as-92	33092	91.9668
se-65	34065	64.9647
se-66	34066	65.9552
se-67	34067	66.9501
se-68	34068	67.9418
se-69	34069	68.9396
se-70	34070	69.9334
se-71	34071	70.9322
se-72	34072	71.9271
se-73	34073	72.9268
se-74	34074	73.9225

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
se-75	34075	74.9225
se-76	34076	75.9192
se-77	34077	76.9199
se-78	34078	77.9173
se-79	34079	78.9185
se-80	34080	79.9165
se-81	34081	80.918
se-82	34082	81.9167
se-83	34083	82.9191
se-84	34084	83.9185
se-85	34085	84.9222
se-86	34086	85.9243
se-87	34087	86.9285
se-88	34088	87.9314
se-89	34089	88.9364
se-90	34090	89.94
se-91	34091	90.946
se-92	34092	91.9499
se-93	34093	92.9563
se-94	34094	93.9605
br-67	35067	66.9648
br-68	35068	67.9585
br-69	35069	68.9501
br-70	35070	69.9448
br-71	35071	70.9387
br-72	35072	71.9366
br-73	35073	72.9317
br-74	35074	73.9299
br-75	35075	74.9258
br-76	35076	75.9245
br-77	35077	76.9214
br-78	35078	77.9212
br-79	35079	78.9183
br-80	35080	79.9185
br-81	35081	80.9163
br-82	35082	81.9168
br-83	35083	82.9152
br-84	35084	83.9165
br-85	35085	84.9156
br-86	35086	85.9188
br-87	35087	86.9207
br-88	35088	87.9241
br-89	35089	88.9264

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
br-90	35090	89.9306
br-91	35091	90.934
br-92	35092	91.9393
br-93	35093	92.943
br-94	35094	93.9487
br-95	35095	94.9529
br-96	35096	95.9585
br-97	35097	96.9628
kr-69	36069	68.9652
kr-70	36070	69.9553
kr-71	36071	70.9496
kr-72	36072	71.9421
kr-73	36073	72.9393
kr-74	36074	73.9331
kr-75	36075	74.9309
kr-76	36076	75.9259
kr-77	36077	76.9247
kr-78	36078	77.9204
kr-79	36079	78.9201
kr-80	36080	79.9164
kr-81	36081	80.9166
kr-82	36082	81.9135
kr-83	36083	82.9141
kr-84	36084	83.9115
kr-85	36085	84.9125
kr-86	36086	85.9106
kr-87	36087	86.9134
kr-88	36088	87.9145
kr-89	36089	88.9176
kr-90	36090	89.9195
kr-91	36091	90.9234
kr-92	36092	91.9262
kr-93	36093	92.9313
kr-94	36094	93.9344
kr-95	36095	94.9398
kr-96	36096	95.9431
kr-97	36097	96.9486
kr-98	36098	97.9519
kr-99	36099	98.9576
kr-100	36100	99.9611
rb-71	37071	70.9653
rb-72	37072	71.9591
rb-73	37073	72.9506

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
rb-74	37074	73.9443
rb-75	37075	74.9386
rb-76	37076	75.9351
rb-77	37077	76.9304
rb-78	37078	77.9281
rb-79	37079	78.924
rb-80	37080	79.9225
rb-81	37081	80.919
rb-82	37082	81.9182
rb-83	37083	82.9151
rb-84	37084	83.9144
rb-85	37085	84.9118
rb-86	37086	85.9112
rb-87	37087	86.9092
rb-88	37088	87.9113
rb-89	37089	88.9123
rb-90	37090	89.9148
rb-91	37091	90.9165
rb-92	37092	91.9197
rb-93	37093	92.922
rb-94	37094	93.9264
rb-95	37095	94.9293
rb-96	37096	95.9343
rb-97	37097	96.9373
rb-98	37098	97.9418
rb-99	37099	98.9454
rb-100	37100	99.9499
rb-101	37101	100.9532
rb-102	37102	101.9589
sr-73	38073	72.966
sr-74	38074	73.9563
sr-75	38075	74.95
sr-76	38076	75.9418
sr-77	38077	76.938
sr-78	38078	77.9322
sr-79	38079	78.9297
sr-80	38080	79.9245
sr-81	38081	80.9232
sr-82	38082	81.9184
sr-83	38083	82.9176
sr-84	38084	83.9134
sr-85	38085	84.9129
sr-86	38086	85.9093

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
sr-87	38087	86.9089
sr-88	38088	87.9056
sr-89	38089	88.9074
sr-90	38090	89.9077
sr-91	38091	90.9102
sr-92	38092	91.911
sr-93	38093	92.914
sr-94	38094	93.9154
sr-95	38095	94.9194
sr-96	38096	95.9217
sr-97	38097	96.9261
sr-98	38098	97.9285
sr-99	38099	98.9332
sr-100	38100	99.9353
sr-101	38101	100.9405
sr-102	38102	101.943
sr-103	38103	102.949
sr-104	38104	103.9523
sr-105	38105	104.9586
y-76	39076	75.9585
y-77	39077	76.9497
y-78	39078	77.9436
y-79	39079	78.9373
y-80	39080	79.9343
y-81	39081	80.9291
y-82	39082	81.9268
y-83	39083	82.9223
y-84	39084	83.9204
y-85	39085	84.9164
y-86	39086	85.9149
y-87	39087	86.9109
y-88	39088	87.9095
y-89	39089	88.9059
y-90	39090	89.9072
y-91	39091	90.9073
y-92	39092	91.909
y-93	39093	92.9096
y-94	39094	93.9116
y-95	39095	94.9128
y-96	39096	95.9159
y-97	39097	96.9181
y-98	39098	97.9222
y-99	39099	98.9246

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
y-100	39100	99.9278
y-101	39101	100.9303
y-102	39102	101.9336
y-103	39103	102.9367
y-104	39104	103.941
y-105	39105	104.9449
y-106	39106	105.9498
y-107	39107	106.9541
y-108	39108	107.9595
zr-78	40078	77.9552
zr-79	40079	78.9492
zr-80	40080	79.9404
zr-81	40081	80.9372
zr-82	40082	81.9311
zr-83	40083	82.9286
zr-84	40084	83.9232
zr-85	40085	84.9215
zr-86	40086	85.9165
zr-87	40087	86.9148
zr-88	40088	87.9102
zr-89	40089	88.9089
zr-90	40090	89.9047
zr-91	40091	90.9056
zr-92	40092	91.905
zr-93	40093	92.9065
zr-94	40094	93.9063
zr-95	40095	94.908
zr-96	40096	95.9083
zr-97	40097	96.9109
zr-98	40098	97.9127
zr-99	40099	98.9165
zr-100	40100	99.9178
zr-101	40101	100.9211
zr-102	40102	101.923
zr-103	40103	102.9266
zr-104	40104	103.9288
zr-105	40105	104.9331
zr-106	40106	105.9359
zr-107	40107	106.9408
zr-108	40108	107.944
zr-109	40109	108.9492
zr-110	40110	109.9529
nb-81	41081	80.949

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
nb-82	41082	81.9431
nb-83	41083	82.9367
nb-84	41084	83.9336
nb-85	41085	84.9279
nb-86	41086	85.925
nb-87	41087	86.9204
nb-88	41088	87.9183
nb-89	41089	88.9134
nb-90	41090	89.9113
nb-91	41091	90.907
nb-92	41092	91.9072
nb-93	41093	92.9064
nb-94	41094	93.9073
nb-95	41095	94.9068
nb-96	41096	95.9081
nb-97	41097	96.9081
nb-98	41098	97.9103
nb-99	41099	98.9116
nb-100	41100	99.9142
nb-101	41101	100.9153
nb-102	41102	101.918
nb-103	41103	102.9191
nb-104	41104	103.9225
nb-105	41105	104.9239
nb-106	41106	105.928
nb-107	41107	106.9303
nb-108	41108	107.9348
nb-109	41109	108.9376
nb-110	41110	109.9424
nb-111	41111	110.9456
nb-112	41112	111.9508
nb-113	41113	112.9547
mo-83	42083	82.9487
mo-84	42084	83.9401
mo-85	42085	84.9365
mo-86	42086	85.9307
mo-87	42087	86.9273
mo-88	42088	87.922
mo-89	42089	88.9195
mo-90	42090	89.9139
mo-91	42091	90.9118
mo-92	42092	91.9068
mo-93	42093	92.9068

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
mo-94	42094	93.9051
mo-95	42095	94.9058
mo-96	42096	95.9047
mo-97	42097	96.906
mo-98	42098	97.9054
mo-99	42099	98.9077
mo-100	42100	99.9075
mo-101	42101	100.9103
mo-102	42102	101.9103
mo-103	42103	102.9132
mo-104	42104	103.9138
mo-105	42105	104.917
mo-106	42106	105.9181
mo-107	42107	106.9217
mo-108	42108	107.9234
mo-109	42109	108.9278
mo-110	42110	109.9297
mo-111	42111	110.9344
mo-112	42112	111.9368
mo-113	42113	112.9419
mo-114	42114	113.9449
mo-115	42115	114.9503
tc-85	43085	84.9488
tc-86	43086	85.9429
tc-87	43087	86.9365
tc-88	43088	87.9327
tc-89	43089	88.9272
tc-90	43090	89.9236
tc-91	43091	90.9184
tc-92	43092	91.9153
tc-93	43093	92.9102
tc-94	43094	93.9097
tc-95	43095	94.9077
tc-96	43096	95.9079
tc-97	43097	96.9064
tc-98	43098	97.9072
tc-99	43099	98.9062
tc-100	43100	99.9077
tc-101	43101	100.9073
tc-102	43102	101.9092
tc-103	43103	102.9092
tc-104	43104	103.9115
tc-105	43105	104.9117

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
tc-106	43106	105.9144
tc-107	43107	106.9151
tc-108	43108	107.9185
tc-109	43109	108.92
tc-110	43110	109.9238
tc-111	43111	110.9257
tc-112	43112	111.9292
tc-113	43113	112.9316
tc-114	43114	113.9359
tc-115	43115	114.9387
tc-116	43116	115.9434
tc-117	43117	116.9465
tc-118	43118	117.9515
ru-87	44087	86.9492
ru-88	44088	87.9403
ru-89	44089	88.9361
ru-90	44090	89.9299
ru-91	44091	90.9263
ru-92	44092	91.9201
ru-93	44093	92.9171
ru-94	44094	93.9114
ru-95	44095	94.9104
ru-96	44096	95.9076
ru-97	44097	96.9076
ru-98	44098	97.9053
ru-99	44099	98.9059
ru-100	44100	99.9042
ru-101	44101	100.9056
ru-102	44102	101.9044
ru-103	44103	102.9063
ru-104	44104	103.9054
ru-105	44105	104.9078
ru-106	44106	105.9073
ru-107	44107	106.9099
ru-108	44108	107.9102
ru-109	44109	108.9132
ru-110	44110	109.9141
ru-111	44111	110.9177
ru-112	44112	111.919
ru-113	44113	112.9225
ru-114	44114	113.9243
ru-115	44115	114.9287
ru-116	44116	115.9308

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ru-117	44117	116.9356
ru-118	44118	117.9378
ru-119	44119	118.9428
ru-120	44120	119.9453
rh-89	45089	88.9488
rh-90	45090	89.9429
rh-91	45091	90.9365
rh-92	45092	91.932
rh-93	45093	92.9257
rh-94	45094	93.9217
rh-95	45095	94.9159
rh-96	45096	95.9145
rh-97	45097	96.9113
rh-98	45098	97.9107
rh-99	45099	98.9081
rh-100	45100	99.9081
rh-101	45101	100.9062
rh-102	45102	101.9068
rh-103	45103	102.9055
rh-104	45104	103.9067
rh-105	45105	104.9057
rh-106	45106	105.9073
rh-107	45107	106.9068
rh-108	45108	107.9087
rh-109	45109	108.9087
rh-110	45110	109.9111
rh-111	45111	110.9116
rh-112	45112	111.9144
rh-113	45113	112.9155
rh-114	45114	113.9188
rh-115	45115	114.9203
rh-116	45116	115.9241
rh-117	45117	116.926
rh-118	45118	117.9301
rh-119	45119	118.9321
rh-120	45120	119.9364
rh-121	45121	120.9387
rh-122	45122	121.9432
pd-91	46091	90.9491
pd-92	46092	91.9404
pd-93	46093	92.9359
pd-94	46094	93.9288
pd-95	46095	94.9247

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
pd-96	46096	95.9182
pd-97	46097	96.9165
pd-98	46098	97.9127
pd-99	46099	98.9118
pd-100	46100	99.9085
pd-101	46101	100.9083
pd-102	46102	101.9056
pd-103	46103	102.9061
pd-104	46104	103.904
pd-105	46105	104.9051
pd-106	46106	105.9035
pd-107	46107	106.9051
pd-108	46108	107.9039
pd-109	46109	108.906
pd-110	46110	109.9052
pd-111	46111	110.9077
pd-112	46112	111.9073
pd-113	46113	112.9101
pd-114	46114	113.9104
pd-115	46115	114.9137
pd-116	46116	115.9142
pd-117	46117	116.9178
pd-118	46118	117.919
pd-119	46119	118.9231
pd-120	46120	119.9247
pd-121	46121	120.9289
pd-122	46122	121.9305
pd-123	46123	122.9349
pd-124	46124	123.9369
ag-93	47093	92.9498
ag-94	47094	93.9428
ag-95	47095	94.9355
ag-96	47096	95.9307
ag-97	47097	96.924
ag-98	47098	97.9216
ag-99	47099	98.9176
ag-100	47100	99.9161
ag-101	47101	100.9128
ag-102	47102	101.9117
ag-103	47103	102.909
ag-104	47104	103.9086
ag-105	47105	104.9065
ag-106	47106	105.9067

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ag-107	47107	106.9051
ag-108	47108	107.906
ag-109	47109	108.9047
ag-110	47110	109.9061
ag-110m	1047110	
ag-111	47111	110.9053
ag-112	47112	111.907
ag-113	47113	112.9066
ag-114	47114	113.9088
ag-115	47115	114.9088
ag-116	47116	115.9114
ag-117	47117	116.9117
ag-118	47118	117.9146
ag-119	47119	118.9157
ag-120	47120	119.9188
ag-121	47121	120.9199
ag-122	47122	121.9235
ag-123	47123	122.9249
ag-124	47124	123.9286
ag-125	47125	124.9304
ag-126	47126	125.9345
ag-127	47127	126.9368
ag-128	47128	127.9412
ag-129	47129	128.9437
ag-130	47130	129.9505
cd-95	48095	94.9499
cd-96	48096	95.9398
cd-97	48097	96.9349
cd-98	48098	97.9274
cd-99	48099	98.925
cd-100	48100	99.9203
cd-101	48101	100.9187
cd-102	48102	101.9145
cd-103	48103	102.9134
cd-104	48104	103.9099
cd-105	48105	104.9095
cd-106	48106	105.9065
cd-107	48107	106.9066
cd-108	48108	107.9042
cd-109	48109	108.905
cd-110	48110	109.903
cd-111	48111	110.9042
cd-112	48112	111.9028

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
cd-113	48113	112.9044
cd-114	48114	113.9034
cd-115	48115	114.9054
cd-115m	1048115	
cd-116	48116	115.9048
cd-117	48117	116.9072
cd-118	48118	117.9069
cd-119	48119	118.9099
cd-120	48120	119.9099
cd-121	48121	120.913
cd-122	48122	121.9133
cd-123	48123	122.917
cd-124	48124	123.9176
cd-125	48125	124.9212
cd-126	48126	125.9223
cd-127	48127	126.9264
cd-128	48128	127.9278
cd-129	48129	128.9321
cd-130	48130	129.9339
cd-131	48131	130.9407
cd-132	48132	131.9456
in-97	49097	96.9495
in-98	49098	97.9421
in-99	49099	98.9342
in-100	49100	99.9311
in-101	49101	100.9263
in-102	49102	101.9241
in-103	49103	102.9199
in-104	49104	103.9183
in-105	49105	104.9147
in-106	49106	105.9135
in-107	49107	106.9103
in-108	49108	107.9097
in-109	49109	108.9072
in-110	49110	109.9072
in-111	49111	110.9051
in-112	49112	111.9055
in-113	49113	112.9041
in-114	49114	113.9049
in-115	49115	114.9039
in-116	49116	115.9053
in-117	49117	116.9045
in-118	49118	117.9063

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
in-119	49119	118.9059
in-120	49120	119.908
in-121	49121	120.9079
in-122	49122	121.9103
in-123	49123	122.9104
in-124	49124	123.9132
in-125	49125	124.9136
in-126	49126	125.9165
in-127	49127	126.9174
in-128	49128	127.9202
in-129	49129	128.9217
in-130	49130	129.925
in-131	49131	130.9268
in-132	49132	131.933
in-133	49133	132.9378
in-134	49134	133.9442
in-135	49135	134.9493
sn-99	50099	98.9493
sn-100	50100	99.939
sn-101	50101	100.9361
sn-102	50102	101.9303
sn-103	50103	102.9281
sn-104	50104	103.9231
sn-105	50105	104.9213
sn-106	50106	105.9169
sn-107	50107	106.9156
sn-108	50108	107.9119
sn-109	50109	108.9113
sn-110	50110	109.9078
sn-111	50111	110.9077
sn-112	50112	111.9048
sn-113	50113	112.9052
sn-114	50114	113.9028
sn-115	50115	114.9033
sn-116	50116	115.9017
sn-117	50117	116.9029
sn-118	50118	117.9016
sn-119	50119	118.9033
sn-120	50120	119.9022
sn-121	50121	120.9042
sn-122	50122	121.9034
sn-123	50123	122.9057
sn-124	50124	123.9053

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
sn-125	50125	124.9078
sn-126	50126	125.9077
sn-127	50127	126.9104
sn-128	50128	127.9105
sn-129	50129	128.9135
sn-130	50130	129.914
sn-131	50131	130.917
sn-132	50132	131.9178
sn-133	50133	132.9238
sn-134	50134	133.9283
sn-135	50135	134.9347
sn-136	50136	135.9393
sn-137	50137	136.946
sb-103	51103	102.9397
sb-104	51104	103.9365
sb-105	51105	104.9315
sb-106	51106	105.9288
sb-107	51107	106.9241
sb-108	51108	107.9222
sb-109	51109	108.9181
sb-110	51110	109.9167
sb-111	51111	110.9132
sb-112	51112	111.9124
sb-113	51113	112.9094
sb-114	51114	113.9093
sb-115	51115	114.9066
sb-116	51116	115.9068
sb-117	51117	116.9048
sb-118	51118	117.9055
sb-119	51119	118.9039
sb-120	51120	119.9051
sb-121	51121	120.9038
sb-122	51122	121.9052
sb-123	51123	122.9042
sb-124	51124	123.9059
sb-125	51125	124.9053
sb-126	51126	125.9072
sb-127	51127	126.9069
sb-128	51128	127.9092
sb-129	51129	128.9091
sb-130	51130	129.9117
sb-131	51131	130.912
sb-132	51132	131.9145

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
sb-133	51133	132.9153
sb-134	51134	133.9204
sb-135	51135	134.9252
sb-136	51136	135.9303
sb-137	51137	136.9353
sb-138	51138	137.9408
sb-139	51139	138.946
te-105	52105	104.9436
te-106	52106	105.9375
te-107	52107	106.935
te-108	52108	107.9294
te-109	52109	108.9274
te-110	52110	109.9224
te-111	52111	110.9211
te-112	52112	111.917
te-113	52113	112.9159
te-114	52114	113.9121
te-115	52115	114.9119
te-116	52116	115.9085
te-117	52117	116.9087
te-118	52118	117.9058
te-119	52119	118.9064
te-120	52120	119.904
te-121	52121	120.9049
te-122	52122	121.903
te-123	52123	122.9043
te-124	52124	123.9028
te-125	52125	124.9044
te-126	52126	125.9033
te-127	52127	126.9052
te-127m	1052127	
te-128	52128	127.9045
te-129	52129	128.9066
te-129m	1052129	
te-130	52130	129.9062
te-131	52131	130.9085
te-132	52132	131.9086
te-133	52133	132.911
te-134	52134	133.9114
te-135	52135	134.9164
te-136	52136	135.9201
te-137	52137	136.9253
te-138	52138	137.9292

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
te-139	52139	138.9347
te-140	52140	139.9388
te-141	52141	140.9447
te-142	52142	141.9491
i-108	53108	107.9435
i-109	53109	108.9381
i-110	53110	109.9352
i-111	53111	110.9303
i-112	53112	111.928
i-113	53113	112.9236
i-114	53114	113.9219
i-115	53115	114.9181
i-116	53116	115.9168
i-117	53117	116.9137
i-118	53118	117.9131
i-119	53119	118.9101
i-120	53120	119.91
i-121	53121	120.9074
i-122	53122	121.9076
i-123	53123	122.9056
i-124	53124	123.9062
i-125	53125	124.9046
i-126	53126	125.9056
i-127	53127	126.9045
i-128	53128	127.9058
i-129	53129	128.905
i-130	53130	129.9067
i-131	53131	130.9061
i-132	53132	131.908
i-133	53133	132.9078
i-134	53134	133.9097
i-135	53135	134.91
i-136	53136	135.9147
i-137	53137	136.9179
i-138	53138	137.9223
i-139	53139	138.9261
i-140	53140	139.931
i-141	53141	140.935
i-142	53142	141.9402
i-143	53143	142.9446
i-144	53144	143.95
xe-110	54110	109.9443
xe-111	54111	110.9416

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
xe-112	54112	111.9356
xe-113	54113	112.9333
xe-114	54114	113.928
xe-115	54115	114.9263
xe-116	54116	115.9216
xe-117	54117	116.9204
xe-118	54118	117.9162
xe-119	54119	118.9154
xe-120	54120	119.9118
xe-121	54121	120.9115
xe-122	54122	121.9084
xe-123	54123	122.9085
xe-124	54124	123.9059
xe-125	54125	124.9064
xe-126	54126	125.9043
xe-127	54127	126.9052
xe-128	54128	127.9035
xe-129	54129	128.9048
xe-130	54130	129.9035
xe-131	54131	130.9051
xe-132	54132	131.9041
xe-133	54133	132.9059
xe-134	54134	133.9054
xe-135	54135	134.9072
xe-136	54136	135.9072
xe-137	54137	136.9116
xe-138	54138	137.914
xe-139	54139	138.9188
xe-140	54140	139.9216
xe-141	54141	140.9267
xe-142	54142	141.9297
xe-143	54143	142.9351
xe-144	54144	143.9385
xe-145	54145	144.9441
xe-146	54146	145.9478
xe-147	54147	146.9536
cs-112	55112	111.9503
cs-113	55113	112.9445
cs-114	55114	113.9415
cs-115	55115	114.9359
cs-116	55116	115.9334
cs-117	55117	116.9287
cs-118	55118	117.9266

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
cs-119	55119	118.9224
cs-120	55120	119.9207
cs-121	55121	120.9172
cs-122	55122	121.9161
cs-123	55123	122.913
cs-124	55124	123.9123
cs-125	55125	124.9097
cs-126	55126	125.9094
cs-127	55127	126.9074
cs-128	55128	127.9078
cs-129	55129	128.9061
cs-130	55130	129.9067
cs-131	55131	130.9055
cs-132	55132	131.9064
cs-133	55133	132.9055
cs-134	55134	133.9067
cs-135	55135	134.906
cs-136	55136	135.9073
cs-137	55137	136.9071
cs-138	55138	137.911
cs-139	55139	138.9134
cs-140	55140	139.9173
cs-141	55141	140.92
cs-142	55142	141.9243
cs-143	55143	142.9274
cs-144	55144	143.9321
cs-145	55145	144.9355
cs-146	55146	145.9403
cs-147	55147	146.9442
cs-148	55148	147.9492
cs-149	55149	148.9529
cs-150	55150	149.9582
cs-151	55151	150.9622
ba-114	56114	113.9507
ba-115	56115	114.9474
ba-116	56116	115.9414
ba-117	56117	116.9385
ba-118	56118	117.933
ba-119	56119	118.9307
ba-120	56120	119.926
ba-121	56121	120.924
ba-122	56122	121.9199
ba-123	56123	122.9188

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ba-124	56124	123.9151
ba-125	56125	124.9145
ba-126	56126	125.9112
ba-127	56127	126.9111
ba-128	56128	127.9083
ba-129	56129	128.9087
ba-130	56130	129.9063
ba-131	56131	130.9069
ba-132	56132	131.9051
ba-133	56133	132.906
ba-134	56134	133.9045
ba-135	56135	134.9057
ba-136	56136	135.9046
ba-137	56137	136.9058
ba-138	56138	137.9052
ba-139	56139	138.9088
ba-140	56140	139.9106
ba-141	56141	140.9144
ba-142	56142	141.9164
ba-143	56143	142.9206
ba-144	56144	143.9229
ba-145	56145	144.9276
ba-146	56146	145.9302
ba-147	56147	146.935
ba-148	56148	147.9377
ba-149	56149	148.9426
ba-150	56150	149.9457
ba-151	56151	150.9508
ba-152	56152	151.9543
ba-153	56153	152.9596
la-117	57117	116.9501
la-118	57118	117.9467
la-119	57119	118.941
la-120	57120	119.9381
la-121	57121	120.933
la-122	57122	121.9307
la-123	57123	122.9262
la-124	57124	123.9246
la-125	57125	124.9208
la-126	57126	125.9195
la-127	57127	126.9164
la-128	57128	127.9156
la-129	57129	128.9127

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
la-130	57130	129.9124
la-131	57131	130.9101
la-132	57132	131.9101
la-133	57133	132.9082
la-134	57134	133.9085
la-135	57135	134.907
la-136	57136	135.9076
la-137	57137	136.9065
la-138	57138	137.9071
la-139	57139	138.9064
la-140	57140	139.9095
la-141	57141	140.911
la-142	57142	141.9141
la-143	57143	142.9161
la-144	57144	143.9196
la-145	57145	144.9216
la-146	57146	145.9258
la-147	57147	146.9282
la-148	57148	147.9322
la-149	57149	148.9347
la-150	57150	149.9388
la-151	57151	150.9417
la-152	57152	151.9462
la-153	57153	152.9496
la-154	57154	153.9545
la-155	57155	154.9583
ce-119	58119	118.9528
ce-120	58120	119.9466
ce-121	58121	120.9434
ce-122	58122	121.9379
ce-123	58123	122.9354
ce-124	58124	123.9304
ce-125	58125	124.9284
ce-126	58126	125.924
ce-127	58127	126.9227
ce-128	58128	127.9189
ce-129	58129	128.9181
ce-130	58130	129.9147
ce-131	58131	130.9144
ce-132	58132	131.9115
ce-133	58133	132.9115
ce-134	58134	133.9089
ce-135	58135	134.9091

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ce-136	58136	135.9072
ce-137	58137	136.9078
ce-138	58138	137.906
ce-139	58139	138.9066
ce-140	58140	139.9054
ce-141	58141	140.9083
ce-142	58142	141.9092
ce-143	58143	142.9124
ce-144	58144	143.9137
ce-145	58145	144.9172
ce-146	58146	145.9188
ce-147	58147	146.9227
ce-148	58148	147.9244
ce-149	58149	148.9284
ce-150	58150	149.9304
ce-151	58151	150.934
ce-152	58152	151.9365
ce-153	58153	152.9406
ce-154	58154	153.9434
ce-155	58155	154.948
ce-156	58156	155.9513
ce-157	58157	156.9563
pr-121	59121	120.9554
pr-122	59122	121.9518
pr-123	59123	122.946
pr-124	59124	123.943
pr-125	59125	124.9378
pr-126	59126	125.9353
pr-127	59127	126.9308
pr-128	59128	127.9288
pr-129	59129	128.9251
pr-130	59130	129.9236
pr-131	59131	130.9203
pr-132	59132	131.9193
pr-133	59133	132.9163
pr-134	59134	133.9157
pr-135	59135	134.9131
pr-136	59136	135.9127
pr-137	59137	136.9107
pr-138	59138	137.9108
pr-139	59139	138.9089
pr-140	59140	139.9091
pr-141	59141	140.9077

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
pr-142	59142	141.91
pr-143	59143	142.9108
pr-144	59144	143.9133
pr-145	59145	144.9145
pr-146	59146	145.9176
pr-147	59147	146.919
pr-148	59148	147.9221
pr-149	59149	148.9237
pr-150	59150	149.9267
pr-151	59151	150.9283
pr-152	59152	151.9315
pr-153	59153	152.9338
pr-154	59154	153.9375
pr-155	59155	154.9401
pr-156	59156	155.9443
pr-157	59157	156.9474
pr-158	59158	157.952
pr-159	59159	158.9555
nd-124	60124	123.9522
nd-125	60125	124.9489
nd-126	60126	125.9432
nd-127	60127	126.9405
nd-128	60128	127.9354
nd-129	60129	128.9332
nd-130	60130	129.9285
nd-131	60131	130.9272
nd-132	60132	131.9233
nd-133	60133	132.9223
nd-134	60134	133.9188
nd-135	60135	134.9182
nd-136	60136	135.915
nd-137	60137	136.9146
nd-138	60138	137.912
nd-139	60139	138.912
nd-140	60140	139.9095
nd-141	60141	140.9096
nd-142	60142	141.9077
nd-143	60143	142.9098
nd-144	60144	143.9101
nd-145	60145	144.9126
nd-146	60146	145.9131
nd-147	60147	146.9161
nd-148	60148	147.9169

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
nd-149	60149	148.9202
nd-150	60150	149.9209
nd-151	60151	150.9238
nd-152	60152	151.9247
nd-153	60153	152.9277
nd-154	60154	153.9295
nd-155	60155	154.9329
nd-156	60156	155.935
nd-157	60157	156.939
nd-158	60158	157.9416
nd-159	60159	158.9461
nd-160	60160	159.9491
nd-161	60161	160.9539
pm-126	61126	125.9575
pm-127	61127	126.9516
pm-128	61128	127.9484
pm-129	61129	128.9432
pm-130	61130	129.9404
pm-131	61131	130.9359
pm-132	61132	131.9337
pm-133	61133	132.9298
pm-134	61134	133.9283
pm-135	61135	134.9249
pm-136	61136	135.9236
pm-137	61137	136.9205
pm-138	61138	137.9196
pm-139	61139	138.9168
pm-140	61140	139.916
pm-141	61141	140.9136
pm-142	61142	141.9129
pm-143	61143	142.9109
pm-144	61144	143.9126
pm-145	61145	144.9128
pm-146	61146	145.9147
pm-147	61147	146.9151
pm-148	61148	147.9175
pm-148m	1061148	
pm-149	61149	148.9183
pm-150	61150	149.921
pm-151	61151	150.9212
pm-152	61152	151.9235
pm-153	61153	152.9241
pm-154	61154	153.9265

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
pm-155	61155	154.9281
pm-156	61156	155.9311
pm-157	61157	156.933
pm-158	61158	157.9366
pm-159	61159	158.939
pm-160	61160	159.943
pm-161	61161	160.9459
pm-162	61162	161.9503
pm-163	61163	162.9537
sm-128	62128	127.9581
sm-129	62129	128.9546
sm-130	62130	129.9489
sm-131	62131	130.9461
sm-132	62132	131.9407
sm-133	62133	132.9387
sm-134	62134	133.934
sm-135	62135	134.9325
sm-136	62136	135.9283
sm-137	62137	136.927
sm-138	62138	137.9232
sm-139	62139	138.9223
sm-140	62140	139.919
sm-141	62141	140.9185
sm-142	62142	141.9152
sm-143	62143	142.9146
sm-144	62144	143.912
sm-145	62145	144.9134
sm-146	62146	145.913
sm-147	62147	146.9149
sm-148	62148	147.9148
sm-149	62149	148.9172
sm-150	62150	149.9173
sm-151	62151	150.9199
sm-152	62152	151.9197
sm-153	62153	152.9221
sm-154	62154	153.9222
sm-155	62155	154.9246
sm-156	62156	155.9255
sm-157	62157	156.9284
sm-158	62158	157.93
sm-159	62159	158.9332
sm-160	62160	159.9351
sm-161	62161	160.9388

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
sm-162	62162	161.9412
sm-163	62163	162.9454
sm-164	62164	163.9483
sm-165	62165	164.953
eu-130	63130	129.9636
eu-131	63131	130.9577
eu-132	63132	131.9544
eu-133	63133	132.9492
eu-134	63134	133.9465
eu-135	63135	134.9418
eu-136	63136	135.9396
eu-137	63137	136.9356
eu-138	63138	137.9337
eu-139	63139	138.9298
eu-140	63140	139.9281
eu-141	63141	140.9249
eu-142	63142	141.9234
eu-143	63143	142.9203
eu-144	63144	143.9188
eu-145	63145	144.9163
eu-146	63146	145.9172
eu-147	63147	146.9167
eu-148	63148	147.9181
eu-149	63149	148.9179
eu-150	63150	149.9197
eu-151	63151	150.9198
eu-152	63152	151.9217
eu-153	63153	152.9212
eu-154	63154	153.923
eu-155	63155	154.9229
eu-156	63156	155.9247
eu-157	63157	156.9254
eu-158	63158	157.9279
eu-159	63159	158.9291
eu-160	63160	159.932
eu-161	63161	160.9337
eu-162	63162	161.937
eu-163	63163	162.9392
eu-164	63164	163.943
eu-165	63165	164.9457
eu-166	63166	165.95
eu-167	63167	166.9532
gd-134	64134	133.9554

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
gd-135	64135	134.9526
gd-136	64136	135.9473
gd-137	64137	136.945
gd-138	64138	137.9401
gd-139	64139	138.9382
gd-140	64140	139.9337
gd-141	64141	140.9321
gd-142	64142	141.9281
gd-143	64143	142.9267
gd-144	64144	143.923
gd-145	64145	144.9217
gd-146	64146	145.9183
gd-147	64147	146.9191
gd-148	64148	147.9181
gd-149	64149	148.9193
gd-150	64150	149.9187
gd-151	64151	150.9203
gd-152	64152	151.9198
gd-153	64153	152.9218
gd-154	64154	153.9209
gd-155	64155	154.9226
gd-156	64156	155.9221
gd-157	64157	156.924
gd-158	64158	157.9241
gd-159	64159	158.9264
gd-160	64160	159.927
gd-161	64161	160.9297
gd-162	64162	161.931
gd-163	64163	162.934
gd-164	64164	163.9359
gd-165	64165	164.9394
gd-166	64166	165.9416
gd-167	64167	166.9456
gd-168	64168	167.9484
gd-169	64169	168.9529
tb-136	65136	135.9614
tb-137	65137	136.956
tb-138	65138	137.9532
tb-139	65139	138.9483
tb-140	65140	139.9458
tb-141	65141	140.9415
tb-142	65142	141.9387
tb-143	65143	142.9351

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
tb-144	65144	143.933
tb-145	65145	144.9293
tb-146	65146	145.9272
tb-147	65147	146.9241
tb-148	65148	147.9243
tb-149	65149	148.9232
tb-150	65150	149.9237
tb-151	65151	150.9231
tb-152	65152	151.9241
tb-153	65153	152.9234
tb-154	65154	153.9247
tb-155	65155	154.9235
tb-156	65156	155.9247
tb-157	65157	156.924
tb-158	65158	157.9254
tb-159	65159	158.9254
tb-160	65160	159.9272
tb-161	65161	160.9276
tb-162	65162	161.9295
tb-163	65163	162.9306
tb-164	65164	163.9333
tb-165	65165	164.9349
tb-166	65166	165.938
tb-167	65167	166.94
tb-168	65168	167.9436
tb-169	65169	168.9462
tb-170	65170	169.9503
tb-171	65171	170.9533
dy-138	66138	137.9625
dy-139	66139	138.9595
dy-140	66140	139.954
dy-141	66141	140.9514
dy-142	66142	141.9464
dy-143	66143	142.9438
dy-144	66144	143.9393
dy-145	66145	144.9374
dy-146	66146	145.9328
dy-147	66147	146.9311
dy-148	66148	147.9272
dy-149	66149	148.9273
dy-150	66150	149.9256
dy-151	66151	150.9262
dy-152	66152	151.9247

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
dy-153	66153	152.9258
dy-154	66154	153.9244
dy-155	66155	154.9258
dy-156	66156	155.9243
dy-157	66157	156.9255
dy-158	66158	157.9244
dy-159	66159	158.9257
dy-160	66160	159.9252
dy-161	66161	160.9269
dy-162	66162	161.9268
dy-163	66163	162.9287
dy-164	66164	163.9292
dy-165	66165	164.9317
dy-166	66166	165.9328
dy-167	66167	166.9357
dy-168	66168	167.9371
dy-169	66169	168.9403
dy-170	66170	169.9424
dy-171	66171	170.9462
dy-172	66172	171.9488
dy-173	66173	172.953
ho-140	67140	139.9685
ho-141	67141	140.9631
ho-142	67142	141.9598
ho-143	67143	142.9546
ho-144	67144	143.9515
ho-145	67145	144.9472
ho-146	67146	145.9446
ho-147	67147	146.9401
ho-148	67148	147.9377
ho-149	67149	148.9338
ho-150	67150	149.9335
ho-151	67151	150.9317
ho-152	67152	151.9317
ho-153	67153	152.9302
ho-154	67154	153.9306
ho-155	67155	154.9291
ho-156	67156	155.9298
ho-157	67157	156.9283
ho-158	67158	157.9289
ho-159	67159	158.9277
ho-160	67160	159.9287
ho-161	67161	160.9279

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ho-162	67162	161.9291
ho-163	67163	162.9287
ho-164	67164	163.9302
ho-165	67165	164.9303
ho-166	67166	165.9323
ho-166m	1067166	
ho-167	67167	166.9331
ho-168	67168	167.9355
ho-169	67169	168.9369
ho-170	67170	169.9396
ho-171	67171	170.9415
ho-172	67172	171.9448
ho-173	67173	172.9473
ho-174	67174	173.9512
ho-175	67175	174.9541
er-143	68143	142.9663
er-144	68144	143.9604
er-145	68145	144.9574
er-146	68146	145.952
er-147	68147	146.9495
er-148	68148	147.9445
er-149	68149	148.9423
er-150	68150	149.9379
er-151	68151	150.9375
er-152	68152	151.935
er-153	68153	152.9351
er-154	68154	153.9328
er-155	68155	154.9332
er-156	68156	155.9311
er-157	68157	156.9319
er-158	68158	157.9299
er-159	68159	158.9307
er-160	68160	159.9291
er-161	68161	160.93
er-162	68162	161.9288
er-163	68163	162.93
er-164	68164	163.9292
er-165	68165	164.9307
er-166	68166	165.9303
er-167	68167	166.9321
er-168	68168	167.9324
er-169	68169	168.9346
er-170	68170	169.9355

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
er-171	68171	170.938
er-172	68172	171.9394
er-173	68173	172.9424
er-174	68174	173.9442
er-175	68175	174.9478
er-176	68176	175.9501
er-177	68177	176.9541
tm-145	69145	144.9701
tm-146	69146	145.9664
tm-147	69147	146.961
tm-148	69148	147.9578
tm-149	69149	148.9527
tm-150	69150	149.95
tm-151	69151	150.9455
tm-152	69152	151.9444
tm-153	69153	152.942
tm-154	69154	153.9416
tm-155	69155	154.9392
tm-156	69156	155.939
tm-157	69157	156.937
tm-158	69158	157.937
tm-159	69159	158.935
tm-160	69160	159.9353
tm-161	69161	160.9335
tm-162	69162	161.934
tm-163	69163	162.9326
tm-164	69164	163.9336
tm-165	69165	164.9324
tm-166	69166	165.9335
tm-167	69167	166.9328
tm-168	69168	167.9342
tm-169	69169	168.9342
tm-170	69170	169.9358
tm-171	69171	170.9364
tm-172	69172	171.9384
tm-173	69173	172.9396
tm-174	69174	173.9422
tm-175	69175	174.9438
tm-176	69176	175.947
tm-177	69177	176.949
tm-178	69178	177.9526
tm-179	69179	178.9553
yb-148	70148	147.9674

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
yb-149	70149	148.964
yb-150	70150	149.9584
yb-151	70151	150.9554
yb-152	70152	151.9503
yb-153	70153	152.9495
yb-154	70154	153.9464
yb-155	70155	154.9458
yb-156	70156	155.9428
yb-157	70157	156.9426
yb-158	70158	157.9399
yb-159	70159	158.94
yb-160	70160	159.9375
yb-161	70161	160.9379
yb-162	70162	161.9358
yb-163	70163	162.9363
yb-164	70164	163.9345
yb-165	70165	164.9353
yb-166	70166	165.9339
yb-167	70167	166.935
yb-168	70168	167.9339
yb-169	70169	168.9352
yb-170	70170	169.9348
yb-171	70171	170.9363
yb-172	70172	171.9364
yb-173	70173	172.9382
yb-174	70174	173.9389
yb-175	70175	174.9413
yb-176	70176	175.9426
yb-177	70177	176.9453
yb-178	70178	177.9467
yb-179	70179	178.9502
yb-180	70180	179.9523
yb-181	70181	180.9561
lu-150	71150	149.9732
lu-151	71151	150.9676
lu-152	71152	151.9641
lu-153	71153	152.9588
lu-154	71154	153.9575
lu-155	71155	154.9543
lu-156	71156	155.953
lu-157	71157	156.9501
lu-158	71158	157.9493
lu-159	71159	158.9466

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
lu-160	71160	159.946
lu-161	71161	160.9436
lu-162	71162	161.9433
lu-163	71163	162.9412
lu-164	71164	163.9413
lu-165	71165	164.9394
lu-166	71166	165.9399
lu-167	71167	166.9383
lu-168	71168	167.9387
lu-169	71169	168.9377
lu-170	71170	169.9385
lu-171	71171	170.9379
lu-172	71172	171.9391
lu-173	71173	172.9389
lu-174	71174	173.9403
lu-175	71175	174.9408
lu-176	71176	175.9427
lu-177	71177	176.9438
lu-178	71178	177.946
lu-179	71179	178.9473
lu-180	71180	179.9499
lu-181	71181	180.952
lu-182	71182	181.955
lu-183	71183	182.9576
lu-184	71184	183.9609
hf-153	72153	152.9707
hf-154	72154	153.9649
hf-155	72155	154.9634
hf-156	72156	155.9594
hf-157	72157	156.9584
hf-158	72158	157.9548
hf-159	72159	158.954
hf-160	72160	159.9507
hf-161	72161	160.9503
hf-162	72162	161.9472
hf-163	72163	162.9471
hf-164	72164	163.9444
hf-165	72165	164.9446
hf-166	72166	165.9422
hf-167	72167	166.9426
hf-168	72168	167.9406
hf-169	72169	168.9413
hf-170	72170	169.9396

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
hf-171	72171	170.9405
hf-172	72172	171.9395
hf-173	72173	172.9405
hf-174	72174	173.94
hf-175	72175	174.9415
hf-176	72176	175.9414
hf-177	72177	176.9432
hf-178	72178	177.9437
hf-179	72179	178.9458
hf-180	72180	179.9465
hf-181	72181	180.9491
hf-182	72182	181.9505
hf-183	72183	182.9535
hf-184	72184	183.9554
hf-185	72185	184.9588
hf-186	72186	185.9609
hf-187	72187	186.9646
hf-188	72188	187.9668
ta-155	73155	154.9746
ta-156	73156	155.9723
ta-157	73157	156.9682
ta-158	73158	157.9667
ta-159	73159	158.963
ta-160	73160	159.9615
ta-161	73161	160.9584
ta-162	73162	161.9573
ta-163	73163	162.9543
ta-164	73164	163.9535
ta-165	73165	164.9508
ta-166	73166	165.9505
ta-167	73167	166.9481
ta-168	73168	167.948
ta-169	73169	168.946
ta-170	73170	169.9462
ta-171	73171	170.9445
ta-172	73172	171.9449
ta-173	73173	172.9438
ta-174	73174	173.9444
ta-175	73175	174.9437
ta-176	73176	175.9449
ta-177	73177	176.9445
ta-178	73178	177.9458
ta-179	73179	178.9459

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ta-180	73180	179.9475
ta-181	73181	180.948
ta-182	73182	181.9501
ta-183	73183	182.9514
ta-184	73184	183.954
ta-185	73185	184.9556
ta-186	73186	185.9586
ta-187	73187	186.9605
ta-188	73188	187.9637
ta-189	73189	188.9658
ta-190	73190	189.9692
w-158	74158	157.9746
w-159	74159	158.9729
w-160	74160	159.9685
w-161	74161	160.9674
w-162	74162	161.9635
w-163	74163	162.9625
w-164	74164	163.959
w-165	74165	164.9583
w-166	74166	165.955
w-167	74167	166.9548
w-168	74168	167.9518
w-169	74169	168.9518
w-170	74170	169.9492
w-171	74171	170.9494
w-172	74172	171.9473
w-173	74173	172.9477
w-174	74174	173.9461
w-175	74175	174.9467
w-176	74176	175.9456
w-177	74177	176.9466
w-178	74178	177.9459
w-179	74179	178.9471
w-180	74180	179.9467
w-181	74181	180.9482
w-182	74182	181.9482
w-183	74183	182.9502
w-184	74184	183.9509
w-185	74185	184.9534
w-186	74186	185.9544
w-187	74187	186.9572
w-188	74188	187.9585
w-189	74189	188.9619

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
w-190	74190	189.9632
w-191	74191	190.9666
w-192	74192	191.9682
re-160	75160	159.9821
re-161	75161	160.9776
re-162	75162	161.976
re-163	75163	162.9721
re-164	75164	163.9703
re-165	75165	164.9671
re-166	75166	165.9658
re-167	75167	166.9626
re-168	75168	167.9616
re-169	75169	168.9588
re-170	75170	169.9582
re-171	75171	170.9557
re-172	75172	171.9554
re-173	75173	172.9532
re-174	75174	173.9531
re-175	75175	174.9514
re-176	75176	175.9516
re-177	75177	176.9503
re-178	75178	177.951
re-179	75179	178.95
re-180	75180	179.9508
re-181	75181	180.9501
re-182	75182	181.9512
re-183	75183	182.9508
re-184	75184	183.9525
re-185	75185	184.953
re-186	75186	185.955
re-187	75187	186.9557
re-188	75188	187.9581
re-189	75189	188.9592
re-190	75190	189.9618
re-191	75191	190.9631
re-192	75192	191.966
re-193	75193	192.9675
re-194	75194	193.9704
os-162	76162	161.9844
os-163	76163	162.9827
os-164	76164	163.978
os-165	76165	164.9768
os-166	76166	165.9727

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
os-167	76167	166.9716
os-168	76168	167.9678
os-169	76169	168.967
os-170	76170	169.9636
os-171	76171	170.9632
os-172	76172	171.96
os-173	76173	172.9598
os-174	76174	173.9571
os-175	76175	174.957
os-176	76176	175.9548
os-177	76177	176.955
os-178	76178	177.9532
os-179	76179	178.9538
os-180	76180	179.9524
os-181	76181	180.9532
os-182	76182	181.9521
os-183	76183	182.9531
os-184	76184	183.9525
os-185	76185	184.954
os-186	76186	185.9538
os-187	76187	186.9557
os-188	76188	187.9558
os-189	76189	188.9581
os-190	76190	189.9585
os-191	76191	190.9609
os-192	76192	191.9615
os-193	76193	192.9642
os-194	76194	193.9652
os-195	76195	194.9681
os-196	76196	195.9696
ir-164	77164	163.9922
ir-165	77165	164.9875
ir-166	77166	165.9858
ir-167	77167	166.9817
ir-168	77168	167.9799
ir-169	77169	168.9763
ir-170	77170	169.975
ir-171	77171	170.9716
ir-172	77172	171.9705
ir-173	77173	172.9675
ir-174	77174	173.9669
ir-175	77175	174.9641
ir-176	77176	175.9637

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ir-177	77177	176.9613
ir-178	77178	177.9611
ir-179	77179	178.9591
ir-180	77180	179.9592
ir-181	77181	180.9576
ir-182	77182	181.9581
ir-183	77183	182.9568
ir-184	77184	183.9575
ir-185	77185	184.9567
ir-186	77186	185.9579
ir-187	77187	186.9574
ir-188	77188	187.9588
ir-189	77189	188.9587
ir-190	77190	189.9606
ir-191	77191	190.9606
ir-192	77192	191.9626
ir-193	77193	192.9629
ir-194	77194	193.9651
ir-195	77195	194.966
ir-196	77196	195.9684
ir-197	77197	196.9697
ir-198	77198	197.9723
ir-199	77199	198.9738
pt-166	78166	165.9949
pt-167	78167	166.993
pt-168	78168	167.9881
pt-169	78169	168.9867
pt-170	78170	169.9825
pt-171	78171	170.9812
pt-172	78172	171.9774
pt-173	78173	172.9764
pt-174	78174	173.9728
pt-175	78175	174.9724
pt-176	78176	175.9689
pt-177	78177	176.9685
pt-178	78178	177.9657
pt-179	78179	178.9654
pt-180	78180	179.963
pt-181	78181	180.9631
pt-182	78182	181.9612
pt-183	78183	182.9616
pt-184	78184	183.9599
pt-185	78185	184.9606

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
pt-186	78186	185.9594
pt-187	78187	186.9606
pt-188	78188	187.9594
pt-189	78189	188.9608
pt-190	78190	189.9599
pt-191	78191	190.9617
pt-192	78192	191.961
pt-193	78193	192.963
pt-194	78194	193.9627
pt-195	78195	194.9648
pt-196	78196	195.965
pt-197	78197	196.9673
pt-198	78198	197.9679
pt-199	78199	198.9706
pt-200	78200	199.9714
pt-201	78201	200.9745
pt-202	78202	201.9757
au-169	79169	168.9981
au-170	79170	169.9961
au-171	79171	170.9919
au-172	79172	171.99
au-173	79173	172.9862
au-174	79174	173.9848
au-175	79175	174.9813
au-176	79176	175.9801
au-177	79177	176.9769
au-178	79178	177.976
au-179	79179	178.9732
au-180	79180	179.9725
au-181	79181	180.9701
au-182	79182	181.9696
au-183	79183	182.9676
au-184	79184	183.9675
au-185	79185	184.9658
au-186	79186	185.9659
au-187	79187	186.9646
au-188	79188	187.9653
au-189	79189	188.9639
au-190	79190	189.9647
au-191	79191	190.9637
au-192	79192	191.9648
au-193	79193	192.9642
au-194	79194	193.9654

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
au-195	79195	194.965
au-196	79196	195.9666
au-197	79197	196.9666
au-198	79198	197.9682
au-199	79199	198.9688
au-200	79200	199.9707
au-201	79201	200.9717
au-202	79202	201.9738
au-203	79203	202.9752
au-204	79204	203.9777
au-205	79205	204.9799
hg-171	80171	171.0038
hg-172	80172	171.9988
hg-173	80173	172.9972
hg-174	80174	173.9929
hg-175	80175	174.9914
hg-176	80176	175.9874
hg-177	80177	176.9863
hg-178	80178	177.9825
hg-179	80179	178.9818
hg-180	80180	179.9783
hg-181	80181	180.9778
hg-182	80182	181.9747
hg-183	80183	182.9745
hg-184	80184	183.9717
hg-185	80185	184.9719
hg-186	80186	185.9694
hg-187	80187	186.9698
hg-188	80188	187.9676
hg-189	80189	188.9682
hg-190	80190	189.9663
hg-191	80191	190.9672
hg-192	80192	191.9656
hg-193	80193	192.9667
hg-194	80194	193.9654
hg-195	80195	194.9667
hg-196	80196	195.9658
hg-197	80197	196.9672
hg-198	80198	197.9668
hg-199	80199	198.9683
hg-200	80200	199.9683
hg-201	80201	200.9703
hg-202	80202	201.9706

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
hg-203	80203	202.9729
hg-204	80204	203.9735
hg-205	80205	204.9761
hg-206	80206	205.9775
hg-207	80207	206.9826
hg-208	80208	207.9859
hg-209	80209	208.991
hg-210	80210	209.9945
tl-176	81176	176.0006
tl-177	81177	176.9964
tl-178	81178	177.9949
tl-179	81179	178.9911
tl-180	81180	179.9899
tl-181	81181	180.9863
tl-182	81182	181.9857
tl-183	81183	182.9822
tl-184	81184	183.9819
tl-185	81185	184.9788
tl-186	81186	185.9783
tl-187	81187	186.9759
tl-188	81188	187.976
tl-189	81189	188.9736
tl-190	81190	189.9739
tl-191	81191	190.9718
tl-192	81192	191.9722
tl-193	81193	192.9707
tl-194	81194	193.9712
tl-195	81195	194.9698
tl-196	81196	195.9705
tl-197	81197	196.9696
tl-198	81198	197.9705
tl-199	81199	198.9699
tl-200	81200	199.971
tl-201	81201	200.9708
tl-202	81202	201.9721
tl-203	81203	202.9723
tl-204	81204	203.9739
tl-205	81205	204.9744
tl-206	81206	205.9761
tl-207	81207	206.9774
tl-208	81208	207.982
tl-209	81209	208.9854
tl-210	81210	209.9901

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
tl-211	81211	210.9935
tl-212	81212	211.9982
pb-178	82178	178.0038
pb-179	82179	179.0022
pb-180	82180	179.9979
pb-181	82181	180.9966
pb-182	82182	181.9927
pb-183	82183	182.9919
pb-184	82184	183.9881
pb-185	82185	184.9876
pb-186	82186	185.9842
pb-187	82187	186.9839
pb-188	82188	187.9809
pb-189	82189	188.9808
pb-190	82190	189.9781
pb-191	82191	190.9783
pb-192	82192	191.9758
pb-193	82193	192.9762
pb-194	82194	193.974
pb-195	82195	194.9745
pb-196	82196	195.9728
pb-197	82197	196.9734
pb-198	82198	197.972
pb-199	82199	198.9729
pb-200	82200	199.9718
pb-201	82201	200.9729
pb-202	82202	201.9722
pb-203	82203	202.9734
pb-204	82204	203.973
pb-205	82205	204.9745
pb-206	82206	205.9745
pb-207	82207	206.9759
pb-208	82208	207.9767
pb-209	82209	208.9811
pb-210	82210	209.9842
pb-211	82211	210.9887
pb-212	82212	211.9919
pb-213	82213	212.9966
pb-214	82214	213.9998
pb-215	82215	215.0048
bi-184	83184	184.0011
bi-185	83185	184.9976
bi-186	83186	185.9966

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
bi-187	83187	186.9932
bi-188	83188	187.9923
bi-189	83189	188.9892
bi-190	83190	189.9883
bi-191	83191	190.9858
bi-192	83192	191.9855
bi-193	83193	192.983
bi-194	83194	193.9828
bi-195	83195	194.9807
bi-196	83196	195.9807
bi-197	83197	196.9789
bi-198	83198	197.9792
bi-199	83199	198.9777
bi-200	83200	199.9781
bi-201	83201	200.977
bi-202	83202	201.9777
bi-203	83203	202.9769
bi-204	83204	203.9778
bi-205	83205	204.9774
bi-206	83206	205.9785
bi-207	83207	206.9785
bi-208	83208	207.9797
bi-209	83209	208.9804
bi-210	83210	209.9841
bi-211	83211	210.9873
bi-212	83212	211.9913
bi-213	83213	212.9944
bi-214	83214	213.9987
bi-215	83215	215.0018
bi-216	83216	216.0063
bi-217	83217	217.0095
bi-218	83218	218.0143
po-188	84188	187.9994
po-189	84189	188.9985
po-190	84190	189.9951
po-191	84191	190.9946
po-192	84192	191.9913
po-193	84193	192.991
po-194	84194	193.9882
po-195	84195	194.9881
po-196	84196	195.9855
po-197	84197	196.9857
po-198	84198	197.9834

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
po-199	84199	198.9837
po-200	84200	199.9818
po-201	84201	200.9823
po-202	84202	201.9808
po-203	84203	202.9814
po-204	84204	203.9803
po-205	84205	204.9812
po-206	84206	205.9805
po-207	84207	206.9816
po-208	84208	207.9812
po-209	84209	208.9824
po-210	84210	209.9829
po-211	84211	210.9866
po-212	84212	211.9889
po-213	84213	212.9929
po-214	84214	213.9952
po-215	84215	214.9994
po-216	84216	216.0019
po-217	84217	217.0063
po-218	84218	218.009
po-219	84219	219.0137
po-220	84220	220.0166
at-193	85193	192.9998
at-194	85194	193.9987
at-195	85195	194.9963
at-196	85196	195.9958
at-197	85197	196.9932
at-198	85198	197.9928
at-199	85199	198.9905
at-200	85200	199.9904
at-201	85201	200.9884
at-202	85202	201.9886
at-203	85203	202.9869
at-204	85204	203.9872
at-205	85205	204.9861
at-206	85206	205.9867
at-207	85207	206.9858
at-208	85208	207.9866
at-209	85209	208.9862
at-210	85210	209.9872
at-211	85211	210.9875
at-212	85212	211.9908
at-213	85213	212.9929

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
at-214	85214	213.9964
at-215	85215	214.9987
at-216	85216	216.0024
at-217	85217	217.0047
at-218	85218	218.0087
at-219	85219	219.0112
at-220	85220	220.0154
at-221	85221	221.0181
at-222	85222	222.0223
at-223	85223	223.0252
rn-195	86195	195.0054
rn-196	86196	196.0021
rn-197	86197	197.0016
rn-198	86198	197.9987
rn-199	86199	198.9984
rn-200	86200	199.9957
rn-201	86201	200.9956
rn-202	86202	201.9933
rn-203	86203	202.9934
rn-204	86204	203.9914
rn-205	86205	204.9917
rn-206	86206	205.9902
rn-207	86207	206.9907
rn-208	86208	207.9896
rn-209	86209	208.9904
rn-210	86210	209.9897
rn-211	86211	210.9906
rn-212	86212	211.9907
rn-213	86213	212.9939
rn-214	86214	213.9954
rn-215	86215	214.9987
rn-216	86216	216.0003
rn-217	86217	217.0039
rn-218	86218	218.0056
rn-219	86219	219.0095
rn-220	86220	220.0114
rn-221	86221	221.0155
rn-222	86222	222.0176
rn-223	86223	223.0218
rn-224	86224	224.0241
rn-225	86225	225.0284
rn-226	86226	226.0309
rn-227	86227	227.0354

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
rn-228	86228	228.038
fr-199	87199	199.0073
fr-200	87200	200.0066
fr-201	87201	201.0039
fr-202	87202	202.0034
fr-203	87203	203.0009
fr-204	87204	204.0007
fr-205	87205	204.9986
fr-206	87206	205.9987
fr-207	87207	206.9969
fr-208	87208	207.9971
fr-209	87209	208.996
fr-210	87210	209.9964
fr-211	87211	210.9955
fr-212	87212	211.9962
fr-213	87213	212.9962
fr-214	87214	213.999
fr-215	87215	215.0003
fr-216	87216	216.0032
fr-217	87217	217.0046
fr-218	87218	218.0076
fr-219	87219	219.0092
fr-220	87220	220.0123
fr-221	87221	221.0143
fr-222	87222	222.0175
fr-223	87223	223.0197
fr-224	87224	224.0233
fr-225	87225	225.0256
fr-226	87226	226.0294
fr-227	87227	227.0318
fr-228	87228	228.0357
fr-229	87229	229.0385
fr-230	87230	230.0425
fr-231	87231	231.0454
fr-232	87232	232.0498
ra-202	88202	202.0099
ra-203	88203	203.0093
ra-204	88204	204.0065
ra-205	88205	205.0063
ra-206	88206	206.0038
ra-207	88207	207.0038
ra-208	88208	208.0018
ra-209	88209	209.002

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ra-210	88210	210.0005
ra-211	88211	211.0009
ra-212	88212	211.9998
ra-213	88213	213.0004
ra-214	88214	214.0001
ra-215	88215	215.0027
ra-216	88216	216.0035
ra-217	88217	217.0063
ra-218	88218	218.0071
ra-219	88219	219.0101
ra-220	88220	220.011
ra-221	88221	221.0139
ra-222	88222	222.0154
ra-223	88223	223.0185
ra-224	88224	224.0202
ra-225	88225	225.0236
ra-226	88226	226.0254
ra-227	88227	227.0292
ra-228	88228	228.0311
ra-229	88229	229.035
ra-230	88230	230.0371
ra-231	88231	231.0412
ra-232	88232	232.0436
ra-233	88233	233.0481
ra-234	88234	234.0507
ac-206	89206	206.0145
ac-207	89207	207.0119
ac-208	89208	208.0116
ac-209	89209	209.0095
ac-210	89210	210.0094
ac-211	89211	211.0077
ac-212	89212	212.0078
ac-213	89213	213.0066
ac-214	89214	214.0069
ac-215	89215	215.0065
ac-216	89216	216.0087
ac-217	89217	217.0094
ac-218	89218	218.0116
ac-219	89219	219.0124
ac-220	89220	220.0148
ac-221	89221	221.0156
ac-222	89222	222.0178
ac-223	89223	223.0191

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
ac-224	89224	224.0217
ac-225	89225	225.0232
ac-226	89226	226.0261
ac-227	89227	227.0278
ac-228	89228	228.031
ac-229	89229	229.033
ac-230	89230	230.0363
ac-231	89231	231.0386
ac-232	89232	232.042
ac-233	89233	233.0446
ac-234	89234	234.0484
ac-235	89235	235.0512
ac-236	89236	236.0553
th-209	90209	209.0177
th-210	90210	210.0151
th-211	90211	211.0149
th-212	90212	212.013
th-213	90213	213.013
th-214	90214	214.0115
th-215	90215	215.0117
th-216	90216	216.0111
th-217	90217	217.0131
th-218	90218	218.0133
th-219	90219	219.0155
th-220	90220	220.0157
th-221	90221	221.0182
th-222	90222	222.0185
th-223	90223	223.0208
th-224	90224	224.0215
th-225	90225	225.024
th-226	90226	226.0249
th-227	90227	227.0277
th-228	90228	228.0287
th-229	90229	229.0318
th-230	90230	230.0331
th-231	90231	231.0363
th-232	90232	232.0381
th-233	90233	233.0416
th-234	90234	234.0436
th-235	90235	235.0475
th-236	90236	236.0499
th-237	90237	237.0539
th-238	90238	238.0565

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
pa-212	91212	212.0232
pa-213	91213	213.0211
pa-214	91214	214.0209
pa-215	91215	215.0192
pa-216	91216	216.0191
pa-217	91217	217.0183
pa-218	91218	218.02
pa-219	91219	219.0199
pa-220	91220	220.0219
pa-221	91221	221.0219
pa-222	91222	222.0237
pa-223	91223	223.024
pa-224	91224	224.0256
pa-225	91225	225.0261
pa-226	91226	226.028
pa-227	91227	227.0288
pa-228	91228	228.0311
pa-229	91229	229.0321
pa-230	91230	230.0345
pa-231	91231	231.0359
pa-232	91232	232.0386
pa-233	91233	233.0403
pa-234	91234	234.0433
pa-235	91235	235.0454
pa-236	91236	236.0487
pa-237	91237	237.0511
pa-238	91238	238.0545
pa-239	91239	239.0573
pa-240	91240	240.061
u-217	92217	217.0244
u-218	92218	218.0235
u-219	92219	219.0249
u-220	92220	220.0247
u-221	92221	221.0264
u-222	92222	222.0261
u-223	92223	223.0277
u-224	92224	224.0276
u-225	92225	225.0294
u-226	92226	226.0293
u-227	92227	227.0312
u-228	92228	228.0314
u-229	92229	229.0335
u-230	92230	230.0339

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
u-231	92231	231.0363
u-232	92232	232.0372
u-233	92233	233.0396
u-234	92234	234.041
u-235	92235	235.0439
u-236	92236	236.0456
u-237	92237	237.0487
u-238	92238	238.0508
u-239	92239	239.0543
u-240	92240	240.0566
u-241	92241	241.0603
u-242	92242	242.0629
np-225	93225	225.0339
np-226	93226	226.0352
np-227	93227	227.035
np-228	93228	228.0362
np-229	93229	229.0363
np-230	93230	230.0378
np-231	93231	231.0383
np-232	93232	232.0401
np-233	93233	233.0407
np-234	93234	234.0429
np-235	93235	235.0441
np-236	93236	236.0466
np-237	93237	237.0482
np-238	93238	238.0509
np-239	93239	239.0529
np-240	93240	240.0562
np-241	93241	241.0582
np-242	93242	242.0616
np-243	93243	243.0643
np-244	93244	244.0679
pu-228	94228	228.0387
pu-229	94229	229.0401
pu-230	94230	230.0397
pu-231	94231	231.0411
pu-232	94232	232.0412
pu-233	94233	233.043
pu-234	94234	234.0433
pu-235	94235	235.0453
pu-236	94236	236.0461
pu-237	94237	237.0484
pu-238	94238	238.0496

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
pu-239	94239	239.0522
pu-240	94240	240.0538
pu-241	94241	241.0569
pu-242	94242	242.0587
pu-243	94243	243.062
pu-244	94244	244.0642
pu-245	94245	245.0677
pu-246	94246	246.0702
pu-247	94247	247.0741
am-231	95231	231.0456
am-232	95232	232.0466
am-233	95233	233.0464
am-234	95234	234.0478
am-235	95235	235.0479
am-236	95236	236.0496
am-237	95237	237.05
am-238	95238	238.052
am-239	95239	239.053
am-240	95240	240.0553
am-241	95241	241.0568
am-242	95242	242.0596
am-242m	1095242	
am-243	95243	243.0614
am-244	95244	244.0643
am-244m	1095244	
am-245	95245	245.0665
am-246	95246	246.0698
am-247	95247	247.0721
am-248	95248	248.0757
am-249	95249	249.0785
cm-233	96233	233.0508
cm-234	96234	234.0502
cm-235	96235	235.0514
cm-236	96236	236.0514
cm-237	96237	237.0529
cm-238	96238	238.053
cm-239	96239	239.055
cm-240	96240	240.0555
cm-241	96241	241.0576
cm-242	96242	242.0588
cm-243	96243	243.0614
cm-244	96244	244.0627
cm-245	96245	245.0655

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
cm-246	96246	246.0672
cm-247	96247	247.0703
cm-248	96248	248.0724
cm-249	96249	249.0759
cm-250	96250	250.0784
cm-251	96251	251.0823
cm-252	96252	252.0849
bk-235	97235	235.0566
bk-236	97236	236.0573
bk-237	97237	237.057
bk-238	97238	238.0583
bk-239	97239	239.0583
bk-240	97240	240.0598
bk-241	97241	241.0602
bk-242	97242	242.062
bk-243	97243	243.063
bk-244	97244	244.0652
bk-245	97245	245.0664
bk-246	97246	246.0687
bk-247	97247	247.0703
bk-248	97248	248.0731
bk-249	97249	249.075
bk-250	97250	250.0783
bk-251	97251	251.0808
bk-252	97252	252.0843
bk-253	97253	253.0869
bk-254	97254	254.0906
cf-237	98237	237.0621
cf-238	98238	238.0614
cf-239	98239	239.0624
cf-240	98240	240.0623
cf-241	98241	241.0637
cf-242	98242	242.0637
cf-243	98243	243.0654
cf-244	98244	244.066
cf-245	98245	245.0681
cf-246	98246	246.0688
cf-247	98247	247.071
cf-248	98248	248.0722
cf-249	98249	249.0748
cf-250	98250	250.0764
cf-251	98251	251.0796
cf-252	98252	252.0816

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
cf-253	98253	253.0851
cf-254	98254	254.0873
cf-255	98255	255.091
cf-256	98256	256.0934
es-240	99240	240.0689
es-241	99241	241.0685
es-242	99242	242.0697
es-243	99243	243.0695
es-244	99244	244.0709
es-245	99245	245.0713
es-246	99246	246.0729
es-247	99247	247.0737
es-248	99248	248.0755
es-249	99249	249.0764
es-250	99250	250.0786
es-251	99251	251.08
es-252	99252	252.083
es-253	99253	253.0848
es-254	99254	254.088
es-255	99255	255.0903
es-256	99256	256.0936
es-257	99257	257.096
es-258	99258	258.0995
fm-242	100242	242.0734
fm-243	100243	243.0744
fm-244	100244	244.0741
fm-245	100245	245.0754
fm-246	100246	246.0753
fm-247	100247	247.0768
fm-248	100248	248.0772
fm-249	100249	249.079
fm-250	100250	250.0795
fm-251	100251	251.0816
fm-252	100252	252.0825
fm-253	100253	253.0852
fm-254	100254	254.0869
fm-255	100255	255.09
fm-256	100256	256.0918
fm-257	100257	257.0951
fm-258	100258	258.0971
fm-259	100259	259.1006
fm-260	100260	260.1027
md-245	101245	245.0808

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
md-246	101246	246.0819
md-247	101247	247.0816
md-248	101248	248.0828
md-249	101249	249.083
md-250	101250	250.0844
md-251	101251	251.0848
md-252	101252	252.0866
md-253	101253	253.0873
md-254	101254	254.0897
md-255	101255	255.0911
md-256	101256	256.0941
md-257	101257	257.0956
md-258	101258	258.0984
md-259	101259	259.1005
md-260	101260	260.1036
md-261	101261	261.1057
md-262	101262	262.1089
no-248	102248	248.0866
no-249	102249	249.0878
no-250	102250	250.0875
no-251	102251	251.089
no-252	102252	252.089
no-253	102253	253.0907
no-254	102254	254.091
no-255	102255	255.0932
no-256	102256	256.0943
no-257	102257	257.0969
no-258	102258	258.0982
no-259	102259	259.101
no-260	102260	260.1026
no-261	102261	261.1057
no-262	102262	262.1073
no-263	102263	263.1106
no-264	102264	264.1123
lr-251	103251	251.0944
lr-252	103252	252.0954
lr-253	103253	253.0952
lr-254	103254	254.0965
lr-255	103255	255.0967
lr-256	103256	256.0986
lr-257	103257	257.0995
lr-258	103258	258.1018
lr-259	103259	259.1029

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
lr-260	103260	260.1055
lr-261	103261	261.1069
lr-262	103262	262.1096
lr-263	103263	263.1113
lr-264	103264	264.114
lr-265	103265	265.1158
lr-266	103266	266.1193
rf-253	104253	253.1007
rf-254	104254	254.1002
rf-255	104255	255.1013
rf-256	104256	256.1012
rf-257	104257	257.103
rf-258	104258	258.1035
rf-259	104259	259.1057
rf-260	104260	260.1064
rf-261	104261	261.1088
rf-262	104262	262.1099
rf-263	104263	263.1125
rf-264	104264	264.114
rf-265	104265	265.1167
rf-266	104266	266.118
rf-267	104267	267.1215
rf-268	104268	268.1236
db-255	105255	255.1074
db-256	105256	256.1081
db-257	105257	257.1077
db-258	105258	258.1092
db-259	105259	259.1096
db-260	105260	260.1113
db-261	105261	261.1121
db-262	105262	262.1141
db-263	105263	263.115
db-264	105264	264.1174
db-265	105265	265.1186
db-266	105266	266.121
db-267	105267	267.1224
db-268	105268	268.1255
db-269	105269	269.1275
db-270	105270	270.1307
sg-258	106258	258.1132
sg-259	106259	259.1145
sg-260	106260	260.1144
sg-261	106261	261.1161

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
sg-262	106262	262.1164
sg-263	106263	263.1183
sg-264	106264	264.1189
sg-265	106265	265.1211
sg-266	106266	266.1221
sg-267	106267	267.1244
sg-268	106268	268.1256
sg-269	106269	269.1288
sg-270	106270	270.1303
sg-271	106271	271.1335
sg-272	106272	272.1352
sg-273	106273	273.1382
bh-260	107260	260.122
bh-261	107261	261.1217
bh-262	107262	262.1229
bh-263	107263	263.123
bh-264	107264	264.1246
bh-265	107265	265.1252
bh-266	107266	266.127
bh-267	107267	267.1277
bh-268	107268	268.1298
bh-269	107269	269.1307
bh-270	107270	270.1336
bh-271	107271	271.1352
bh-272	107272	272.138
bh-273	107273	273.1396
bh-274	107274	274.1424
bh-275	107275	275.1443
hs-263	108263	263.1286
hs-264	108264	264.1284
hs-265	108265	265.1301
hs-266	108266	266.1301
hs-267	108267	267.1318
hs-268	108268	268.1322
hs-269	108269	269.1341
hs-270	108270	270.1346
hs-271	108271	271.1377
hs-272	108272	272.139
hs-273	108273	273.142
hs-274	108274	274.1431
hs-275	108275	275.1459
hs-276	108276	276.1472
hs-277	108277	277.1498

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
mt-265	109265	265.1361
mt-266	109266	266.1373
mt-267	109267	267.1373
mt-268	109268	268.1387
mt-269	109269	269.1391
mt-270	109270	270.1407
mt-271	109271	271.1411
mt-272	109272	272.1437
mt-273	109273	273.1449
mt-274	109274	274.1475
mt-275	109275	275.1487
mt-276	109276	276.1512
mt-277	109277	277.1524
mt-278	109278	278.1548
mt-279	109279	279.1562
ds-267	110267	267.1443
ds-268	110268	268.1438
ds-269	110269	269.1451
ds-270	110270	270.1447
ds-271	110271	271.1461
ds-272	110272	272.1463
ds-273	110273	273.1489
ds-274	110274	274.1495
ds-275	110275	275.1522
ds-276	110276	276.153
ds-277	110277	277.1556
ds-278	110278	278.1565
ds-279	110279	279.1589
ds-280	110280	280.1598
ds-281	110281	281.162
rg-272	111272	272.1536
rg-273	111273	273.1537
rg-274	111274	274.1557
rg-275	111275	275.1561
rg-276	111276	276.1585
rg-277	111277	277.1595
rg-278	111278	278.1616
rg-279	111279	279.1625
rg-280	111280	280.1645
rg-281	111281	281.1654
rg-282	111282	282.1675
rg-283	111283	283.1684
sg-277	112277	277.1639

continues on next page

Table 7.2.1 – continued from previous page

Name	Identifier	Mass (amu)
sg-278	112278	278.1643
sg-279	112279	279.1666
sg-280	112280	280.1671
sg-281	112281	281.1693
sg-282	112282	282.1698
sg-283	112283	283.1718
sg-284	112284	284.1724
sg-285	112285	285.1741
uut-283	113283	283.1765
uut-284	113284	284.1781
uut-285	113285	285.1787
uut-286	113286	286.1805
uut-287	113287	287.1811
uuq-285	114285	285.1837
uuq-286	114286	286.1839
uuq-287	114287	287.1856
uuq-288	114288	288.1857
uuq-289	114289	289.1873
uup-287	115287	287.1912
uup-288	115288	288.1925
uup-289	115289	289.1927
uup-290	115290	290.1942
uup-291	115291	291.1944
uuh-289	116289	289.1989
uuh-290	116290	290.1986
uuh-291	116291	291.2
uuh-292	116292	292.1998
uus-291	117291	291.2066
uus-292	117292	292.2076
uuo-293	118293	293.2147

Table 7.2.2: Elements and their natural abundances.

ID	Symbol	Name	Mass (AMU)	Density	Isotopic distribution	Atom %
1000	h	hydrogen	1.0079	1.0	1001	99.9885
					1002	0.0115
2000	he	helium	4.0026	1.0	2003	0.0001
					2004	99.9999
3000	li	lithium	6.941	0.534	3006	7.59
					3007	92.41
4000	be	beryllium	9.0122	1.85	4009	100.0
5000	b	boron	10.811	2.37	5010	19.9
					5011	80.1

continues on next page

Table 7.2.2 – continued from previous page

ID	Symbol	Name	Mass (AMU)	Density	Isotopic distribution	Atom %
6000	c	carbon	12.0107	2.1	6012	98.93
					6013	1.07
7000	n	nitrogen	14.0067	1.0	7014	99.636
					7015	0.364
8000	o	oxygen	15.9994	1.0	8016	99.757
					8017	0.038
					8018	0.205
9000	f	fluorine	18.9984	1.0	9019	100.0
10000	ne	neon	20.1797	1.0	10020	90.48
					10021	0.27
					10022	9.25
11000	na	sodium	22.9898	0.97	11023	100.0
12000	mg	magnesium	24.305	1.74	12024	78.99
					12025	10.0
					12026	11.01
13000	al	aluminum	26.9815	2.702	13027	100.0
14000	si	silicon	28.0855	2.33	14028	92.223
					14029	4.685
					14030	3.092
15000	p	phosphorus	30.9738	1.82	15031	100.0
16000	s	sulfur	32.065	2.07	16032	94.99
					16033	0.75
					16034	4.25
					16036	0.01
17000	cl	chlorine	35.453	1.0	17035	75.76
					17037	24.24
18000	ar	argon	39.948	1.0	18036	0.3365
					18038	0.0632
					18040	99.6003
19000	k	potassium	39.0983	0.86	19039	93.2581
					19040	0.0117
					19041	6.7302
20000	ca	calcium	40.078	1.55	20040	96.941
					20042	0.647
					20043	0.135
					20044	2.086
					20046	0.004
					20048	0.187
21000	sc	scandium	44.9559	2.989	21045	100.0
22000	ti	titanium	47.867	4.5	22046	8.25
					22047	7.44
					22048	73.72

continues on next page

Table 7.2.2 – continued from previous page

ID	Symbol	Name	Mass (AMU)	Density	Isotopic distribution	Atom %
					22049	5.41
					22050	5.18
23000	v	vanadium	50.9415	5.96	23050	0.25
					23051	99.75
24000	cr	chromium	51.9961	7.2	24050	4.345
					24052	83.789
					24053	9.501
					24054	2.365
25000	mn	manganese	54.938	7.2	25055	100.0
26000	fe	iron	55.845	7.86	26054	5.845
					26056	91.754
					26057	2.119
					26058	0.282
27000	co	cobalt	58.9332	8.9	27059	100.0
28000	ni	nickel	58.6934	8.9	28058	68.0769
					28060	26.2231
					28061	1.1399
					28062	3.6345
					28064	0.9256
29000	cu	copper	63.546	8.92	29063	69.15
					29065	30.85
30000	zn	zinc	65.38	7.14	30064	48.268
					30066	27.975
					30067	4.102
					30068	19.024
					30070	0.631
31000	ga	gallium	69.723	5.904	31069	60.108
					31071	39.892
32000	ge	germanium	72.64	5.35	32070	20.38
					32072	27.31
					32073	7.76
					32074	36.72
					32076	7.83
33000	as	arsenic	74.9216	5.73	33075	100.0
34000	se	selenium	78.96	4.81	34074	0.89
					34076	9.37
					34077	7.63
					34078	23.77
					34080	49.61
					34082	8.73
35000	br	bromine	79.904	3.12	35079	50.69
					35081	49.31

continues on next page

Table 7.2.2 – continued from previous page

ID	Symbol	Name	Mass (AMU)	Density	Isotopic distribution	Atom %
36000	kr	krypton	83.798	1.0	36078	0.355
					36080	2.286
					36082	11.593
					36083	11.5
					36084	56.987
					36086	17.279
37000	rb	rubidium	85.4678	1.532	37085	72.17
					37087	27.83
38000	sr	strontium	87.62	2.6	38084	0.56
					38086	9.86
					38087	7.0
					38088	82.58
39000	y	yttrium	88.9059	4.469	39089	100.0
40000	zr	zirconium	91.224	6.49	40090	51.45
					40091	11.22
					40092	17.15
					40094	17.38
					40096	2.8
41000	nb	niobium	92.9064	8.57	41093	100.0
42000	mo	molybdenum	95.96	10.2	42092	14.77
					42094	9.23
					42095	15.9
					42096	16.68
					42097	9.56
					42098	24.19
					42100	9.67
44000	ru	ruthenium	101.07	12.3	44096	5.54
					44098	1.87
					44099	12.76
					44100	12.6
					44101	17.06
					44102	31.55
					44104	18.62
45000	rh	rhodium	102.9055	12.4	45103	100.0
46000	pd	palladium	106.42	12.02	46102	1.02
					46104	11.14
					46105	22.33
					46106	27.33
					46108	26.46
					46110	11.72
47000	ag	silver	107.8682	10.5	47107	51.839

continues on next page

Table 7.2.2 – continued from previous page

ID	Symbol	Name	Mass (AMU)	Density	Isotopic distribution	Atom %
					47109	48.161
48000	cd	cadmium	112.411	8.642	48106	1.25
					48108	0.89
					48110	12.49
					48111	12.8
					48112	24.13
					48113	12.22
					48114	28.73
					48116	7.49
49000	in	indium	114.818	7.3	49113	4.29
					49115	95.71
50000	sn	tin	118.71	7.31	50112	0.97
					50114	0.66
					50115	0.34
					50116	14.54
					50117	7.68
					50118	24.22
					50119	8.59
					50120	32.58
					50122	4.63
					50124	5.79
51000	sb	antimony	121.76	6.684	51121	57.21
					51123	42.79
52000	te	tellurium	127.6	6.25	52120	0.09
					52122	2.55
					52123	0.89
					52124	4.74
					52125	7.07
					52126	18.84
					52128	31.74
					52130	34.08
53000	i	iodine	126.9045	4.93	53127	100.0
54000	xe	xenon	131.293	1.0	54124	0.0952
					54126	0.089
					54128	1.9102
					54129	26.4006
					54130	4.071
					54131	21.2324
					54132	26.9086
					54134	10.4357
					54136	8.8573
55000	cs	cesium	132.9055	1.879	55133	100.0

continues on next page

Table 7.2.2 – continued from previous page

ID	Symbol	Name	Mass (AMU)	Density	Isotopic distribution	Atom %
56000	ba	barium	137.327	3.51	56130	0.106
					56132	0.101
					56134	2.417
					56135	6.592
					56136	7.854
					56137	11.232
					56138	71.698
57000	la	lanthanum	138.9055	6.145	57138	0.09
					57139	99.91
58000	ce	cerium	140.116	6.657	58136	0.185
					58138	0.251
					58140	88.45
					58142	11.114
59000	pr	praseodymium	140.9077	6.773	59141	100.0
60000	nd	neodymium	144.242	6.8	60142	27.2
					60143	12.2
					60144	23.8
					60145	8.3
					60146	17.2
					60148	5.7
					60150	5.6
62000	sm	samarium	150.36	7.52	62144	3.07
					62147	14.99
					62148	11.24
					62149	13.82
					62150	7.38
					62152	26.75
					62154	22.75
63000	eu	europium	151.964	5.243	63151	47.81
					63153	52.19
64000	gd	gadolinium	157.25	7.9	64152	0.2
					64154	2.18
					64155	14.8
					64156	20.47
					64157	15.65
					64158	24.84
					64160	21.86
65000	tb	terbium	158.9254	8.229	65159	100.0
66000	dy	dysprosium	162.5	8.55	66156	0.056
					66158	0.095
					66160	2.329
					66161	18.889

continues on next page

Table 7.2.2 – continued from previous page

ID	Symbol	Name	Mass (AMU)	Density	Isotopic distribution	Atom %
					66162	25.475
					66163	24.896
					66164	28.26
67000	ho	holmium	164.9303	8.795	67165	100.0
68000	er	erbium	167.259	9.006	68162	0.139
					68164	1.601
					68166	33.503
					68167	22.869
					68168	26.978
					68170	14.91
69000	tm	thulium	168.9342	9.321	69169	100.0
70000	yb	ytterbium	173.054	6.965	70168	0.13
					70170	3.04
					70171	14.28
					70172	21.83
					70173	16.13
					70174	31.83
					70176	12.76
71000	lu	lutetium	174.9668	9.84	71175	97.41
					71176	2.59
72000	hf	hafnium	178.49	13.31	72174	0.16
					72176	5.26
					72177	18.6
					72178	27.28
					72179	13.62
					72180	35.08
73000	ta	tantalum	180.9479	16.6	73180	0.012
					73181	99.988
74000	w	tungsten	183.84	19.35	74180	0.12
					74182	26.5
					74183	14.31
					74184	30.64
					74186	28.43
75000	re	rhenium	186.207	20.53	75185	37.4
					75187	62.6
76000	os	osmium	190.23	22.48	76184	0.02
					76186	1.59
					76187	1.96
					76188	13.24
					76189	16.15
					76190	26.26
					76192	40.78

continues on next page

Table 7.2.2 – continued from previous page

ID	Symbol	Name	Mass (AMU)	Density	Isotopic distribution	Atom %
77000	ir	iridium	192.217	22.421	77191	37.3
					77193	62.7
78000	pt	platinum	195.084	21.45	78190	0.014
					78192	0.782
					78194	32.967
					78195	33.832
					78196	25.242
					78198	7.163
79000	au	gold	196.9666	18.88	79197	100.0
80000	hg	mercury	200.59	13.546	80196	0.15
					80198	9.97
					80199	16.87
					80200	23.1
					80201	13.18
					80202	29.86
					80204	6.87
81000	tl	thallium	204.3833	11.85	81203	29.52
					81205	70.48
82000	pb	lead	207.2	11.344	82204	1.4
					82206	24.1
					82207	22.1
					82208	52.4
83000	bi	bismuth	208.9804	9.8	83209	100.0
90000	th	thorium	232.0381	11.7	90232	100.0
91000	pa	protactinium	231.0359	15.37	91231	100.0
92000	u	uranium	238.0289	19.05	92234	0.0054
					92235	0.7204
					92238	99.2742

*If the column for the chemical symbol has a value, the name and the chemical symbol refer to the same composition. Otherwise, the chemical symbol refers to a different composition. In case of monoisotopic elements, such as ^{209}Bi, the chemical symbol refers directly to SCALE ID 83209 instead of 83000. See Table 7.2.3 for details.

Table 7.2.3: Elements and special nuclide symbols.

Name	Description	ID	Density
activities		900	1.0
1/vabsorber		999	1.0
d	Deuterium in heavy water with S(α,β) thermal kernel	1002	1.0
h-liquid_ch4	Liquid methane at 100 K	1001001	1.0
albound	Al metal with S(α,β) thermal kernel	1013027	2.702
zr90-zr5h8	Zr-90 with the S(α,β) thermal kernel for Zr in zrh2 and zr5h8	1040090	1.0
zr91-zr5h8	Zr-91 with the S(α,β) thermal kernel for Zr in zrh2 and zr5h8	1040091	1.0
zr92-zr5h8	Zr-92 with the S(α,β) thermal kernel for Zr in zrh2 and zr5h8	1040092	1.0
zr93-zr5h8	Zr-93 with the S(α,β) thermal kernel for Zr in zrh2 and zr5h8	1040093	1.0
zr94-zr5h8	Zr-94 with the S(α,β) thermal kernel for Zr in zrh2 and zr5h8	1040094	1.0
zr95-zr5h8	Zr-95 with the S(α,β) thermal kernel for Zr in zrh2 and zr5h8	1040095	1.0
zr96-zr5h8	Zr-96 with the S(α,β) thermal kernel for Zr in zrh2 and zr5h8	1040096	1.0
h-solid_ch4	Solid methane at 22 K	2001001	1.0
bebound	Beryllium metal with a S(α,β) thermal kernel	3004009	1.85
h-cryo_ortho	H at cryogenic temperatures with ortho form	4001001	1.0
d-cryo_ortho	D at cryogenic temperatures with ortho form	4001002	1.0
h-cryo_para	H at cryogenic temperatures with para form	5001001	1.0
d-cryo_para	D at cryogenic temperatures with para form	5001002	1.0
be-beo	Beryllium in beryllium oxide with a S(α,β) thermal kernel	5004009	1.0
o-beo	Oxygen in beryllium oxide with a S(α,β) thermal kernel	5008016	1.0
h-benzene	Benzene with a S(α,β) thermal kernel	6001001	1.0
h-zrh2	Hydrogen in zirconium hydride with a S(α,β) thermal kernel	7001001	1.0
hfreegas	Hydrogen with a free gas thermal kernel	8001001	1.0
dfreegas	Deuterium with a free gas thermal kernel	8001002	1.0
h-poly	Hydrogen in polyethylene with a S(α,β) thermal kernel	9001001	1.0

Table 7.2.4: Compounds.

Name	Description	Density	ID	Atoms per molecule
al2o3		3.97	13000	2
			8000	3
b4c	Boron carbide: B_4C; natural isotopic distribution obtained by default	2.52	5000	4
			6000	1
balsa	Balsa wood: $C_{6`H:sub:`10}O_5$	0.125	6000	6
			1000	10
			8000	5
benzene	Benzene with a S(α,β) thermal kernel	0.8765	5006000	6
			6001001	6
beo	Beryllium oxide with a S(α,β) thermal kernel	3.0	5009	1
			5008016	1
d2o	Heavy water: D_2O	1.1054	1002	2
			8000	1
gd2o3		7.07	64000	2
			8000	3
graphite	Graphite carbon	2.3	3006000	1
h2o	Water with only ^{1}H and ^{16}O with S(α,β) thermal kernels	0.9982	1000	2
			8000	1
h2o-x(e)-hr		0.9982	3001001	2
			8000	1
hfacid	Hydrafluoric acid: HF	1.0	1000	1
			9000	1
hno3	Nitric acid: HNO_3	1.0	1000	1
			7000	1
			8000	3
nor-par(h2o)	Normal Paraffin 13: $C_{13}H_{28}$, uses hydrogen in water thermal kernel	0.76	1001	28
			6000	13
norpar13	Normal paraffin 13: $C_{13}H_{28}$	0.76	9001001	28
			6000	13
oak	Oak wood: $C_6H_{10}O_5$	0.7	6000	6
			1000	10
			8000	5
para(h2o)	Paraffin: $C_{25}H_{52}$, uses hydrogen in water S(α,β) thermal kernel	0.9	1001	52
			6000	25

continues on next page

Table 7.2.4 – continued from previous page

Name	Description	Density	ID	Atoms per molecule
paraffin	Paraffin: $C_{25}H_{52}$, uses hydrogen in polyethylene S(α,β) thermal kernel	0.9	9001001	52
			6000	25
plexiglas	Plexiglas: $C_5H_8O_2$	1.18	1000	8
			6000	5
			8000	2
plexiglass	Plexiglas: $C_5H_8O_2$ Maintained for backward compatibility	1.18	1000	8
			6000	5
			8000	2
poly(h2o)	Polyethylene: CH_2, uses hydrogen in water S(α,β) thermal kernel	0.92	1001	2
			6000	1
polyethylene	Polyethylene: CH_2, uses hydrogen in polyethylene S(α,β) thermal kernel	0.92	9001001	2
			6000	1
polyvinylcl	Polyvinyl chloride: C_2H_3Cl, uses hydrogen in polyethylene S(α,β) thermal kernel	1.6	17000	1
			9001001	3
			6000	2
pu(no3)4	Plutonium nitrate: $Pu(NO_3)_4$	2.447	94000	1
			7000	4
			8000	12
puc	Plutonium carbide: PuC	13.6	94000	1
			6000	1
puf4	Plutonium tetrafluoride: PuF_4	7.0	94000	1
			9000	4
pun	Plutonium nitride: PuN	14.25	94000	1
			7000	1
puo2	Plutonium oxide: PuO_2	11.46	94000	1
			8000	2
pvc	Polyvinyl chloride: C_2H_3Cl, uses hydrogen in polyethylene S(α,β) thermal kernel	1.6	17000	1
			9001001	3
			6000	2

continues on next page

Table 7.2.4 – continued from previous page

Name	Description	Density	ID	Atoms per molecule
pvc(h2o)	Polyvinyl chloride: C_2H_3Cl uses hydrogen in water S(α,β) thermal kernel	1.6	17000	1
			1001	3
			6000	2
redwood	Redwood: $C_6H_{10}O_5$	0.387	6000	6
			1000	10
			8000	5
tbp	Tributyl phosphate: $(C_4H_9)_3PO_4$, uses hydrogen in polyethylene S(α,β) thermal kernel	0.9724	9001001	27
			6000	12
			8000	4
			15000	1
tbp(h2o)	Tributyl phosphate: $(C_4H_9)_3PO_4$, uses hydrogen in water S(α,β) thermal kernel	0.9724	1001	27
			6000	12
			8016	4
			15031	1
th(no3)4		2.3752	90000	1
			7000	4
			8000	12
thf4		6.1	90000	1
			9000	4
u3o8	Uranium oxide (yellowcake): U_3O_8	8.3	92000	3
			8000	8
uc	Uranium carbide: UC	13.63	92000	1
			6000	1
uf4	Uranium tetrafluoride: UF_4	6.7	92000	1
			9000	4
uf6	Uranium hexafluoride: UF_6	4.68	92000	1
			9000	6
un	Uranium nitride: UN	14.31	92000	1
			7000	1
uo2	Uranium dioxide: UO_2	10.96	92000	1
			8000	2
uo2(no3)2	Uranyl nitrate: $UO_2(NO_3)_2$	2.203	92000	1
			7000	2
			8000	8
uo2f2	Uranyl fluoride: UO_2F_2	6.37	92000	1

continues on next page

Table 7.2.4 – continued from previous page

Name	Description	Density	ID	Atoms per molecule
			8000	2
			9000	2
uo3	Uranium trioxide: UO_3	7.29	92000	1
			8000	3
water	H_2O, with natural abundance hydrogen and oxygen and the $S(\alpha,\beta)$ thermal kernel for hydrogen in water	0.9982	1000	2
			8000	1
zr5h8	Zirconium hydride as a mixture of ZrH and ZrH2, with an effective composition of Zr_5H_8	5.61	1040000	5
			7001001	8
zrh2	Zirconium hydride: ZrH_2	5.61	1040000	1
			7001001	2

Table 7.2.5: Alloys and mixtures.

Name	Description	Density	ID	Weight %
carbonsteel	Carbon steel	7.8212	26000	99.0
			6000	1.0
dry-air		1.20000-3	6000	0.0126
			7000	76.5081
			8000	23.4793
granite		2.66	1000	0.0336
			8000	47.8286
			9000	0.0901
			11000	2.2501
			12000	0.1449
			13000	7.4752
			14000	32.8046
			15000	0.0393
			19000	5.0108
			20000	1.1876
			22000	0.252
			25000	0.0465
			26000	2.8367
inconel	Inconel-600*	8.3	14000	2.5
			22000	2.5
			404024000	15.0
			404026000	7.0

continues on next page

Table 7.2.5 – continued from previous page

Name	Description	Density	ID	Weight %
			404028000	73.0
inconels	Inconel-600* (maintained for backward compatibility)	8.3	14000	2.5
			22000	2.5
			24000	15.0
			26000	7.0
			28000	73.0
kero(h2o)	Average kerosene (uses hydrogen for water S(α,β) thermal kernel)	0.82	6000	84.0
			1001	16.0
kerosene	Average kerosene (uses hydrogen for polyethylene S(α,β) thermal kernel)	0.82	6000	84.0
			9001001	16.0
limestone		2.15	1000	0.0863
			6000	11.3548
			8000	49.6925
			11000	0.0371
			12000	4.7656
			13000	0.4294
			14000	2.4299
			15000	0.0175
			16000	0.0251
			19000	0.2744
			20000	30.4731
			22000	0.036
			26000	0.3783
mgconcrete	Magnuson's Concrete	2.147	26000	0.5595
			1000	0.3319
			6000	10.5321
			8000	49.943
			11000	0.1411
			12000	9.42
			13000	0.7859
			14000	4.2101
			16000	0.2483
			17000	0.0523
			19000	0.9445
			20000	22.6318
			22000	0.1488
			25000	0.0512
orconcrete	Oak Ridge Concrete	2.2994	26000	0.7784
			1000	0.6187
			6000	17.52

continues on next page

Table 7.2.5 – continued from previous page

Name	Description	Density	ID	Weight %
			8000	41.02
			11000	0.0271
			12000	3.265
			13000	1.083
			14000	3.448
			19000	0.1138
			20000	32.13
pyrex	Pyrex	2.23	5000	3.7
			13000	1.0
			8000	53.5
			14000	37.7
			11000	4.1
reg-concrete	Regulatory Concrete (developed for U.S. NRC)	2.3	26000	1.4
			1000	1.0
			13000	3.4
			20000	4.4
			8000	53.2
			14000	33.7
			11000	2.9
rfconcrete	Rocky Flats Concrete	2.321	26000	1.01
			1000	0.75
			6000	5.52
			7000	0.02
			8000	48.49
			11000	0.63
			12000	1.25
			13000	2.17
			14000	15.5
			16000	0.19
			19000	1.37
			20000	23.0
			22000	0.1
ss304	Stainless steel-304 (using nuclide with special weighting for ENDF/B-V data only)	7.94	6000	0.08
			14000	1.0
			15000	0.045
			304024000	19.0
			25000	2.0
			304026000	68.375
			304028000	9.5
ss304s	Stainless steel-304 (using standard nuclides instead of special weighted nuclides)	7.94	6000	0.08

continues on next page

Table 7.2.5 – continued from previous page

Name	Description	Density	ID	Weight %
			14000	1.0
			15000	0.045
			24000	19.0
			25000	2.0
			26000	68.375
			28000	9.5
ss316	Stainless steel-3l6 (using nuclide with special weighting for ENDF/B-V data only)	8.03	6000	0.08
			14000	1.0
			15000	0.045
			304024000	17.0
			25000	2.0
			304026000	65.375
			304028000	12.0
			42000	2.5
ss316s	Stainless steel-316 (using standard nuclides instead of special weighted nuclides)	8.03	6000	0.08
			14000	1.0
			15000	0.045
			24000	17.0
			25000	2.0
			26000	65.375
			28000	12.0
			42000	2.5
u(.27)metal	Depleted uranium metal having a fixed isotope distribution [to specify a different distribution, the user should use URANIUM instead of U(.27)METAL]	19.05	92235	0.27
			92238	99.73
zircalloy	No longer available	6.56	302040000	100
zirc2	Zircaloy-2	6.56	40000	98.25
			50000	1.45
			26000	0.135
			24000	0.1
			28000	0.055
			72000	0.01
zirc4	Zircaloy-4	6.56	40000	98.23
			50000	1.45
			26000	0.21
			24000	0.1
			72000	0.01
* Not to be confused with the more common Inconel 718.				

7.2.3 TABLE OF FISSILE SOLUTIONS

The Standard Composition Library (through) describes the various compounds, alloys, elements, and isotopes one may use in defining the material mixtures for a given problem. In addition to the various materials listed there, one is also free to use any of the fissile solutions listed in Table 7.2.6. Indeed, the user is encouraged to treat the solutions listed in Table 7.2.6 as he would any other standard composition. Using empirical fits to experimental data, the code will then automatically calculate the density of the solution, or the user can explicitly specify the density. The code then calculates the volume fraction corresponding to the heavy metal, acid, and water components of the solution. A fissile solution starts with the keyword, SOLUTION, after which one or two salts and the corresponding acid may be specified. Input specifications for fissile solutions can be found in the XSPROC chapter.

Table 7.2.6: Available fissile solution components.

Name of component	**Nuclides in component**
Nitrate solutions	
$UO_2(NO_3)_2$	92000 7000 8000
$Pu(NO_3)_4$	94000 7000 8000
$Th(NO_3)_4$	90000 7000 8000
HNO_3 ACID	1000 7000 8000
Fluoride solutions	
UO_2F_2	92000 8000 9000
PuF_4	94000 9000
ThF_4	90000 9000
HFACID	1000 9000

7.3 BONAMI: RESONANCE SELF-SHIELDING BY THE BONDARENKO METHOD

U. Mertyurek, M. L. Williams, and K. S. Kim

ABSTRACT

BONAMI is a module of the SCALE code system that is used to perform Bondarenko calculations for resonance self-shielding. BONAMI obtains problem-independent cross sections and Bondarenko shielding factors from a multigroup (MG) AMPX master library, and it creates a MG AMPX working library of self-shielded, problem-dependent cross sections. Several options may be used to compute the background cross section values using the narrow resonance or intermediate resonance approximations, with and without Bondarenko iterations. A novel interpolation scheme is used that avoids many of the problems exhibited by other interpolation methods for the Bondarenko factors. BONAMI is most commonly used in automated SCALE sequences and is fully integrated within the SCALE cross section processing module, XSProc. New spatially dependent self-shielding capability has been incorporated into BONAMI for SCALE 6.3.

Acknowledgments

The authors express gratitude to W. A. Wieselquist and M. A. Jessee for their supervision of the SCALE project and review of the manuscript. The authors acknowledge N. M. Greene, formerly of ORNL, for his original development of and contributions to the BONAMI module and methodology.

7.3.1 INTRODUCTION

BONAMI (**BON**darenko **AM**PX **I**nterpolator) is a SCALE module that performs resonance self-shielding calculations based on the Bondarenko method [BONAMIIB64]. It reads Bondarenko shielding factors ("f-factors") and infinitely dilute microscopic cross sections from a problem-*independent* nuclear data library processed by the AMPX system [BONAMIWWCD15], interpolates the tabulated shielding factors to appropriate temperatures and background cross sections for each nuclide in the system, and produces a self-shielded, problem-dependent data set.

The code performs self-shielding for an arbitrary number of mixtures using either the narrow resonance (NR) or intermediate resonance (IR) approximation [BONAMIGC62]. The latter capability was introduced in SCALE 6.2. BONAMI has several options for computing background cross sections, which may include Bondarenko iterations to approximately account for the impact of resonance interference for multiple resonance absorbers. Heterogeneous effects are treated using equivalence theory based on an "escape cross section" for arrays of slabs, cylinders, or spheres. During the execution of a typical SCALE computational sequence using XSProc, Dancoff factors for uniform lattices of square- or triangular-pitched units are calculated automatically for BONAMI by numerical integration over the chord length distribution. However, for non-uniform lattices-such as those containing water holes, control rods, and so on-the SCALE module MCDancoff can be run to compute Dancoff factors using Monte Carlo for an arbitrary 3D configuration, and these values are then provided in the sequence input.

The conventional BONAMI approach in SCALE 6.2 introduces significant biases of reactivity and intra pin self-shielded cross sections for multiple fuel rings with uniform or nonuniform temperature profile. New spatially dependent resonance self-shielding method [BONAMIKHW21] has been developed and implemented into BONAMI for SCALE 6.3.

The major advantages of the Bondarenko approach are its simplicity and speed compared with SCALE's more rigorous CENTRM/PMC self-shielding method, which performs a pointwise (PW) deterministic transport calculation "on the fly" to compute multigroup (MG) self-shielded cross sections. With the availability of IR theory in BONAMI, accurate results can be obtained for a variety of system types without the computation expense of CENTRM/PMC.

7.3.2 BONDARENKO SELF-SHIELDING THEORY

In MG resonance self-shielding calculations, one is interested in calculating effective cross sections of the form

$$\sigma_{X,g}^{(r)} = \frac{\int_g \sigma_X^{(r)}(E)\Phi(E)\mathrm{dE}}{\int_g \Phi(E)\mathrm{dE}}, \tag{7.3.1}$$

where $\sigma_{X,g}^{(r)}$ is the shielded MG cross section for reaction type X of resonance nuclide r in group g; $\sigma_X^{(r)}(E)$ is a PW cross section; and $\Phi(E)$ is the PW weighting function, which approximates the flux spectrum per unit of energy for the system of interest. PW cross section values are known from processing evaluated data in ENDF/B files; therefore, resonance self-shielding depends mainly on determining the problem-dependent flux spectrum $\Phi(E)$, which may exhibit significant fine structure variations as a result of resonance reactions.

The essence of the Bondarenko method is to parameterize the flux spectrum corresponding to varying degrees of self-shielding, represented by the background cross section parameter σ_0 (called "sigma-zero") and the Doppler broadening temperature T. Hence,

$$\Phi(\mathrm{E}) \rightarrow \Phi(\mathrm{E}; \sigma_{0,\mathrm{g}}^{(\mathrm{r})}, \mathrm{T}) \quad , \quad \mathrm{E} \in \mathrm{g} \; ; \text{and } \sigma_{X,g}^{(r)} \rightarrow \sigma_{X,g}^{(r)}(\sigma_{0,g}^{(r)}, \mathrm{T}) \tag{7.3.2}$$

With this approach, it is possible to preprocess MG data for different background cross sections representing varying degrees of resonance self-shielding. This allows the MG averaging to be performed during the original MG library processing, so that BONAMI can do a simple interpolation on the background cross section and temperature to obtain self-shielded cross sections. This procedure is much faster than the CENTRM/PMC method in SCALE, which computes a PW flux spectrum by solving the neutron transport equation on a PW energy mesh in CENTRM and then evaluates Eq. (7.3.1). in PMC "on the fly" during a sequence execution.

BONAMI performs two main tasks: (a) computation of background cross sections for all nuclides in each mixture in the system and (b) interpolation of shielded cross sections from the library values tabulated vs. background cross sections and temperature. The BONAMI calculation is essentially isolated from the computation of the tabulated shielded cross sections, which is performed by the AMPX processing code system-the only connection is through the definition of the background cross section used in processing the library values. Various approximations can be used to parameterize the flux spectrum in terms of a background XS, as required by the Bondarenko method. We will first consider several approaches to representing the flux in an infinite medium, which lead to different definitions of the background cross section. BONAMI's use of equivalence theory to extend the homogeneous methods to address heterogeneous systems, such as reactor lattices, is discussed in the following section.

7.3.2.1 Parameterized Flux Spectra

Several approximations can be applied to the infinite medium transport equation to parameterize the flux spectrum in terms of a background XS, as required by the Bondarenko method. The resulting homogeneous spectra are used in AMPX to process MG cross sections which can also can be applied to heterogeneous systems (i.e., lattices) by using equivalence theory; thus the key step is determining approximations that provide parameterized solutions for homogeneous media. The neutron transport equation for a homogeneous medium at temperature T, containing a resonance nuclide r mixed with other nuclides can be expressed as

$$\left(\Sigma_t^{(r)}(E,T) + \sum_{j\neq r}\Sigma_t^{(j)}(E,T)\right)\Phi(E,T) = S^{(r)}(E,T) + \sum_{j\neq r} S^{(j)}(E,T), \tag{7.3.3}$$

where $\Sigma_t^{(r)}(E,T)$, $S^{(r)}(E,T)$ are the macroscopic total XS and elastic scattering source for r, respectively; and $\Sigma_t^{(j)}(E,T)$, $S^{(j)}(E,T)$ are the macroscopic total cross section and elastic source, respectively, for a nuclide j. The cross sections in all these expressions are Doppler-broadened to the temperature of the medium. The nuclides in the summations (i.e., all nuclides except r) are called background nuclides for the resonance absorber r.

The NR approximation can be used to approximate scattering sources of nuclides for which the neutron energy loss is large compared with the practical widths of resonances for the absorber materials of interest. Applying the NR approximation for the scattering source of background material j gives

$$S^{(j)}(E,T) \rightarrow \Sigma_p^{(j)} C(E) \text{for j = a NR-scatterer nuclide} \tag{7.3.4}$$

where C(E) is a slowly varying function representative of the asymptotic (i.e., no absorption) flux in a homogeneous medium, which approximates the flux between resonances. In the resolved resonance range of most important resonance absorbers, the asymptotic flux per unit energy is represented as,

$$C(E) = \frac{\Phi_\infty}{E}, \tag{7.3.5}$$

where Φ_∞ is an arbitrary normalization constant that cancels from the MG cross section expression. In the thermal range a Maxwellian spectrum is used for C(E), and in the fast range a fission spectrum is used. The

SCALE Cross Section Libraries section of the SCALE documentation gives analytical expressions for C(E) used in AMPX to process MG data with the NR approximation. AMPX also has an option to input numerical values for C(E), obtained for example from a PW slowing-down calculation with CENTRM. This method has been used to process MG data for some nuclides on the SCALE libraries.

Conversely, the wide resonance (WR) approximation has been used to represent elastic scattering sources of nuclides for which the neutron energy loss is small compared with the practical width of the resonance. This approximation tends to be more accurate for heavy nuclides and for lower energies. The limit of infinite mass is usually assumed, so the WR approximation is sometimes called the infinite mass (IM) approximation. Because of the assumption of IM, there is no energy loss due to collisions with WR scatterers. Applying the WR approximation for the slowing-down source of background nuclide *j* gives

$$\mathrm{S}^{(j)}(\mathrm{E,T}) \rightarrow \Sigma_s^{(j)}(\mathrm{E,T})\Phi(\mathrm{E,T}); \text{for } j = \text{a WR-scatterer nuclide} \tag{7.3.6}$$

The IR approximation was proposed in the 1960s for scatterers with slowing-down properties intermediate between those of NR and WR scatterers [BONAMIGC62]. The IR method represents the scattering source for arbitrary nuclide *j* by a linear combination of NR and WR expressions. This is done by introducing an IR parameter usually called lambda, such that

$$\mathrm{S}^{(\mathrm{j})}(\mathrm{E,T}) \;\rightarrow\; \underbrace{\lambda_{\mathrm{g}}^{(\mathrm{j})}\Sigma_{\mathrm{p}}^{(\mathrm{j})}\,C(E)}_{\textbf{NRscatterer}} \;+\; (1-\lambda_{\mathrm{g}}^{(\mathrm{j})})\;\underbrace{\Sigma_{\mathrm{s}}^{(\mathrm{j})}(\mathrm{E,T})\Phi(\mathrm{E,T})}_{\textbf{WRscatterer}} \quad ; \quad \mathrm{E}\in \mathrm{g}\,. \tag{7.3.7}$$

A value of $\lambda = 1$ reduces Eq. (7.3.7) to the NR expression, whereas math:*lambda = 0* reduces the equation to the WR expression. Fractional math:*lambda*s are for IR scatterers. Since the type of scatterer can change with the energy, the IR lambdas are functions of the energy group as well as the nuclide. The math:*lambda* values represent the moderation "effectiveness" of a given nuclide, compared to hydrogen. The AMPX module LAMBDA was used to compute the IR parameters on the SCALE libraries. (See AMPX documentation distributed with SCALE) Substituting Eq. (7.3.7) into Eq. (7.3.3) and then dividing by the absorber number density *N(r)* gives the following IR approximation for the infinite medium transport equation in energy group g

$$\left(\sigma_{\mathrm{t}}^{(\mathrm{r})}(\mathrm{E,T}) + \sigma_0^{(\mathrm{r})}(\mathrm{E,T})\right)\Phi^{(\mathrm{r})}(\mathrm{E,T}) \;=\; \frac{1}{\mathrm{N}^{(\mathrm{r})}}\mathrm{S}^{(\mathrm{r})}(\mathrm{E,T}) \;+\; \frac{1}{\mathrm{N}^{(\mathrm{r})}}\sum_{j\neq r}\lambda_{\mathrm{g}}^{(\mathrm{j})}\,\Sigma_{\mathrm{p}}^{(\mathrm{j})}C(E) \tag{7.3.8}$$

where the background cross section of *r* in the homogeneous medium is defined as

$$\sigma_0^{(\mathrm{r})}(\mathrm{E,T}) \;=\; \frac{1}{\mathrm{N}^{(\mathrm{r})}}\;\sum_{j\neq r}\left(\Sigma_{\mathrm{a}}^{(\mathrm{j})}(\mathrm{E,T}) + \lambda_{\mathrm{g}}^{(\mathrm{j})}\,\Sigma_{\mathrm{s}}^{(\mathrm{j})}(\mathrm{E,T})\right) \tag{7.3.9}$$

Although Eq. (7.3.8) provides the flux spectrum as a function of the background cross section $\sigma_0^{(r)}(u,T)$ it is not in a form that can be preprocessed when the MG library is generated, because the energy variation of $\sigma_0^{(r)}(E,T)$ must be known. If the total cross sections of the background nuclides in Eq. (7.3.9) have different energy variations, the shape of $\sigma_0^{(r)}(E,T)$ depends on their relative concentrations-which are not known when the MG library is processed. However, if the cross sections in Eq. (7.3.9) are independent of energy, so that the background cross section is *constant*, Eq. (7.3.8) can be solved for any arbitrary value of $\sigma_0^{(r)}$ as a parameter. This obviously occurs for the special case in which nuclide *r* is the only resonance nuclide in the mixture; i.e., the background materials are nonabsorbing moderators for which the total cross section is equal to the potential cross section. In this case, $\sigma_0^{(r)}(E,T) \;\rightarrow\; \sigma_{0,g}^{(r)}$, where

$$\sigma_{0,g}^{(r)} \;=\; \frac{1}{N^{(r)}}\sum_{j\neq i}N^{(j)}\,\lambda_g^{(j)}\sigma_p^{(j)} \tag{7.3.10}$$

If the mixture contains multiple resonance absorbers, as is usually the case, other approximations must be made to obtain a constant background cross section.

The approximation of "no resonance interference" assumes that resonances of background nuclides do not overlap with those of nuclide *r*, so their total cross sections can be approximated by the potential values within resonances of *r* where self-shielding occurs. In this approximation, the expression in Eq. (7.3.10) is also used for the background cross section.

Another approximation is to represent the energy-dependent cross sections of the background nuclides by their group-averaged (i.e., self-shielded cross) values; thus

$$\sigma_a^{(j)}(E,T) \quad \rightarrow \quad \sigma_{a,g}^{(j)} \quad ; \qquad \sigma_s^{(j)}(E,T) \quad \rightarrow \quad \sigma_{s,g}^{(j)} \text{ for } E \in g \tag{7.3.11}$$

In this case, the background cross section in Eq. (7.3.9) for nuclide *r* is the group-dependent expression,

$$\sigma_{0,g}^{(\mathrm{r})} \;=\; \frac{1}{\mathrm{N}^{(\mathrm{r})}} \sum_{j \neq r} \left(\Sigma_{\mathrm{a,g}}^{(\mathrm{j})} + \lambda_{\mathrm{g}}^{(\mathrm{j})} \Sigma_{\mathrm{s,g}}^{(\mathrm{j})} \right) \tag{7.3.12}$$

An equation similar to Eq. (7.3.12) is used for the background cross sections of all resonance nuclides; thus the self-shielded cross sections of each resonance absorber depend on the shielded cross sections of all other resonance absorbers in the mixture. When self-shielding operations are performed with BONAMI for this approximation, "Bondarenko" iterations are performed to account for the inter-dependence of the shielded cross sections.

Assuming that $\sigma_0^{(r)}$ is represented as a groupwise-constant based on one of the previous approximations, several methods can be used to obtain a parameterized flux spectrum for preprocessing Bondarenko data in the MG libraries. In the simpliest approach, the scattering source of the resonance nuclide *r* in Eq. (7.3.8) is represented by the NR approximation, $\mathrm{S}^{(\mathrm{r})}(\mathrm{E,T})$ to $\Sigma_{\mathrm{p}}^{(\mathrm{r})} C(E)$. In this case, Eq. (7.3.8) can be solved analytically to obtain the following expression for the flux spectrum used to process MG data as a function of $\sigma_0^{(r)}$:

$$\Phi^{(\mathrm{r})}(\mathrm{E};\sigma_0^{(\mathrm{r})},\mathrm{T}) \;=\; \frac{\sigma_{\mathrm{p}}^{(\mathrm{r})} \;+\; \frac{1}{\mathrm{N}^{(\mathrm{r})}} \sum\limits_{j \neq r} \Sigma_{\mathrm{p}}^{(\mathrm{j})}}{\sigma_{\mathrm{t}}^{(\mathrm{r})}(\mathrm{E,T}) + \sigma_0^{(\mathrm{r})}} C(E) \quad \rightarrow \quad \frac{C(E)}{\sigma_{\mathrm{t}}^{(\mathrm{r})}(\mathrm{E,T}) + \sigma_0^{(\mathrm{r})}} \tag{7.3.13}$$

where C(E) includes is an arbitrary constant multiplier that cancels from Eq. (7.3.1).

A more accurate approach that does not require using the NR approximation is to directly solve the IR form of the neutron transport equation using PW cross sections, with the assumption of no interference between mixed absorber resonances. The IRFfactor module of AMPX uses XSProc to calculate the self-shielded flux spectrum for MG data processing using one of two options:

(a) A homogeneous model corresponding to an infinite medium of the resonance nuclide mixed with hydrogen, in which the ratio of the absorber to hydrogen number densities is varied in CENTRM to obtain the desired background cross section values;

(b) A heterogeneous model corresponding to a 2D unit cell from an infinite lattice, in which the cell geometry (e.g., pitch) as well as the absorber number density is varied in CENTRM to obtain the desired background cross section values.

Both of these models provide a numerical solution for the flux spectrum. Details on these approaches are given in reference 2.

7.3.2.2 Self-Shielded Cross Section Data in SCALE Libraries

The AMPX code system processes self-shielded cross sections using the flux expressions described in the preceding section. For MG libraries in SCALE-6.2 and later versions, the NR approximation in Eq. (7.3.13) is used to represent the flux spectrum for nuclides with masses below A=40, since the NR approximation is generally accurate for low-mass nuclides and/or high energies. The standard AMPX weight functions are used to represent C(E) over the entire energy range for all nuclides with A<40, except for hydrogen and oxygen which use a calculated C(E) from CENTRM. The NR approximation with a calculated C(E) function is also used to represent the spectrum above the resolved resonance range for nuclides with A>40; but in the resolved resonance range of these nuclides, AMPX processes shielded cross sections with flux spectra obtained from CENTRM calculations using either a homogeneous or heterogeneous model. Regardless of the method used to obtain the flux spectrum, the parameterized shielded cross sections for absorber nuclide "r" are computed from the expression,

$$\sigma_{\mathrm{X,g}}^{(\mathrm{r})}(\sigma_0^{(r)},T) \quad = \quad \frac{\int_g \ \sigma_X^{(r)}(E,T)\ \Phi(E;\ \sigma_0^{(r)},T)\,dE}{\int_g \ \Phi(E;\ \sigma_0^{(r)},T)\ dE} \quad , \tag{7.3.14}$$

where $\Phi(E;\ \sigma_0^{(r)},T)$ is the flux for a given value of $\sigma_0^{(r)}$ and T.

Rather than storing self-shielded cross sections in the master library, AMPX converts them to Bondarenko shielding factors, also called f-factors, defined as the ratio of the shielded cross section to the infinitely dilute cross section. Thus the MG libraries in SCALE contain Bondarenko data consisting of f-factors defined as

$$f_{\mathrm{X,g}}^{(\mathrm{r})}(\sigma_0,T) \quad = \quad \frac{\sigma_{\mathrm{X,g}}^{(\mathrm{r})}(\sigma_0,T)}{\sigma_{\mathrm{X,g}}^{(\mathrm{r})}(\infty)} \quad , \tag{7.3.15}$$

and infinitely dilute cross sections defined as,

$$\sigma_{\mathrm{X,g}}^{(\mathrm{r})}(\infty) \quad = \quad \sigma_{\mathrm{X,g}}^{(\mathrm{r})}(\sigma_0=\infty, T=T_{ref}) \rightarrow \quad \frac{\int_g \ \sigma_X^{(r)}(E,T_{ref})\,C(E)\ dE}{\int_g \ C(E)\ dE} \quad . \tag{7.3.16}$$

In AMPX, the reference temperature for the infinitely dilute cross section is normally taken to be 293 K. Bondarenko data on SCALE libraries are provided for all energy groups and for five reaction types: total, radiative capture, fission, within-group scattering, and elastic scatter. Recent SCALE libraries include f-factors at ~10–30 background cross section values (depending on nuclide) ranging from ~10^{-3} to ~10^{10} barns, which span the range of self-shielding conditions. Typically the f-factor data are tabulated at five temperature values. Background cross sections and temperatures available for each nuclide in the SCALE MG libraries are given in the SCALE Cross Section Libraries chapter.

7.3.2.3 Background Cross Section Options in BONAMI

To compute self-shielded cross sections for nuclide *r*, BONAMI first computes the appropriate background cross section for the system of interest and then interpolates the library Bondarenko data to obtain the f-factor corresponding to this σ_0 and nuclide temperature. Several options are available in BONAMI to compute the background cross section, based on Eq. (7.3.10) and Eq. (7.3.12) in the preceding section. The options are specified by input parameter "**iropt**" and have the following definitions:

(a) iropt = 0 => NR approximation with Bondarenko iterations:

Background cross sections for all nuclides are computed using Eq. (7.3.12) with math:*lambda = 1*; therefore,

$$\sigma_0^{(\mathrm{r})} = \frac{1}{\mathrm{N}^{(\mathrm{r})}} \sum_{j \neq r} \Sigma_{\mathrm{t,g}}^{(\mathrm{j})}. \tag{7.3.17}$$

Since the background cross section for each nuclide depends on the shielded total cross sections of all other nuclides in the mixture, "Bondarenko iterations" are performed in BONAMI to obtain a consistent set of shielded cross sections. Bondarenko iterations provide a crude method of accounting for resonance interference effects that are ignored by the approximation for $\sigma_0^{(r)}$ in Eq. (7.3.10). The BONAMI iterative algorithm generally converges in a few iterations. Prior to SCALE-6.2, this option was the only one available in BONAMI, and it is still the default for XSProc.

(b) iropt = 1 => IR approximation with no resonance interference (potential cross sections):

Background cross sections for all nuclides are computed using Eq. (7.3.10). No Bondarenko iterations are needed.

(c) iropt t = 2 => IR approximation with Bondarenko iterations, but no resonance scattering:

Background cross sections for all nuclides are computed using Eq. (7.3.12) with the scattering cross section approximated by the potential value; therefore,

$$\sigma_0^{(\mathrm{r})} = \frac{1}{\mathrm{N}^{(\mathrm{r})}} \sum_{j \neq r} \left(\Sigma_{\mathrm{a,g}}^{(\mathrm{j})} + \lambda_{\mathrm{g}}^{(\mathrm{j})} \Sigma_{\mathrm{p}}^{(\mathrm{j})} \right) \tag{7.3.18}$$

Since the background cross section for each resonance nuclide includes the shielded absorption cross sections of all other nuclides, Bondarenko interactions are performed.

(d) iropt = 3 => IR approximation with Bondarenko iterations:

Background cross sections for all nuclides are computed using the full IR expression in Eq. (7.3.12). Bondarenko interactions are performed.

Computation of the background cross sections in BONAMI generally requires group-dependent values for the IR parameter math:*lambda*. These are calculated by a module in AMPX during the library process and are stored in the MG libraries under the reaction identifier (MT number), MT=2000.

7.3.2.4 Self-Shielded Cross Sections for Heterogeneous Media

Equivalence theory can be used to obtain shielded cross sections for heterogeneous systems containing one or more "lumps" of resonance absorber mixtures separated by moderators, such as reactor lattices. It can be shown that if the fuel escape probability is represented by the Wigner rational approximation, the collision probability formulation of the neutron transport equation for an absorber body in a heterogeneous medium can be reduced to a form identical to Eq. (7.3.3). This can be done for an "equivalent" infinite homogeneous medium consisting of the same absorber body mixture plus an additional NR scatterer with a constant cross section called the "escape cross section" [BONAMILam66]. Equivalence theory states that the self-shielded cross section for resonance absorber *r* in the heterogeneous medium is equal to the self-shielded cross section of *r* in the equivalent infinite homogeneous medium; therefore the f-factors that were calculated for homogenous mixtures can also be used to compute self-shielded cross sections for heterogeneous media by simply interpolating the tabulated f-factors in the library to the modified sigma-zero value of

$$\hat{\sigma}_0^{(r)} = \sigma_0^{(r)} + \sigma_{esc}^{(r)} \tag{7.3.19}$$

where,

$\hat{\sigma}_0^{(r)}$ = background cross section of r in the absorber lump of the heterogeneous system;

$\sigma_0^{(r)}$ = background cross section defined in Sect. 7.3.2.1 for an infinite homogeneous medium of the absorber body mixture;

$\sigma_{esc}^{(r)}$ = microscopic escape cross section for nuclide *r*, defined as

$$\sigma_{esc}^{(r)} = \frac{\Sigma_{esc}}{N^{(r)}} \tag{7.3.20}$$

Σ_{esc} = macroscopic escape cross section for the absorber lump defined in BONAMI as

$$\Sigma_{esc} = \frac{(1 - c)A}{\bar{\ell}\ [1 + (A - 1)\, c]} \tag{7.3.21}$$

where

$\bar{\ell}$ = average chord length of the absorber body = $4 \times \frac{volume}{surface\ \ area}$;

A = Bell factor, used to improve the accuracy of the Wigner rational approximation;

c = lattice Dancoff factor, which is equal to the probability that a neutron escaping from one absorber body will reach another absorber body before colliding in the intervening moderator.

Values for the mean chord length $\bar{\ell}$ are computed in BONAMI for slab, sphere, and cylinder absorber bodies. In the most common mode of operation where BONAMI is executed through the XSProc module in SCALE, Dancoff factors for uniform lattices are computed automatically and provided as input to BONAMI. For nonuniform lattices-such as those containing water holes, control rods, etc.-it may be desirable for the user to run the SCALE module MCDancoff to compute Dancoff factors using Monte Carlo for an arbitrary 3D configuration. In this case the values are provided in the MORE DATA input block of XSProc. The Bell factor "A" is a correction factor to account for errors caused by use of the Wigner rational approximation to represent the escape probability from a lump. Two optional Bell factor corrections are included in BONAMI. The first uses expressions developed by Otter that essentially force the Wigner escape probability for an isolated absorber lump to agree with the exact escape probability for the particular geometry by determining a value of A as a function of $\Sigma_T \bar{\ell}$ for slab, cylindrical, or spherical geometries. Since the Otter expression was developed for isolated bodies, it does not account for errors in the Wigner rational approximation due to lattice effects. BONAMI also includes a Bell factor correction based on a modified formulation developed by Leslie [BONAMILHJ65] that is a function of the Dancoff factor.

7.3.3 INTERPOLATION SCHEME

After the background cross section for a system has been computed, BONAMI interpolates f-factors at the appropriate σ_0 and temperature from the tabulated values in the library. Fig. 7.3.1 shows a typical variation of the f-factor vs. background cross sections for the capture cross section of ^{238}U in the SCALE 252 group library.

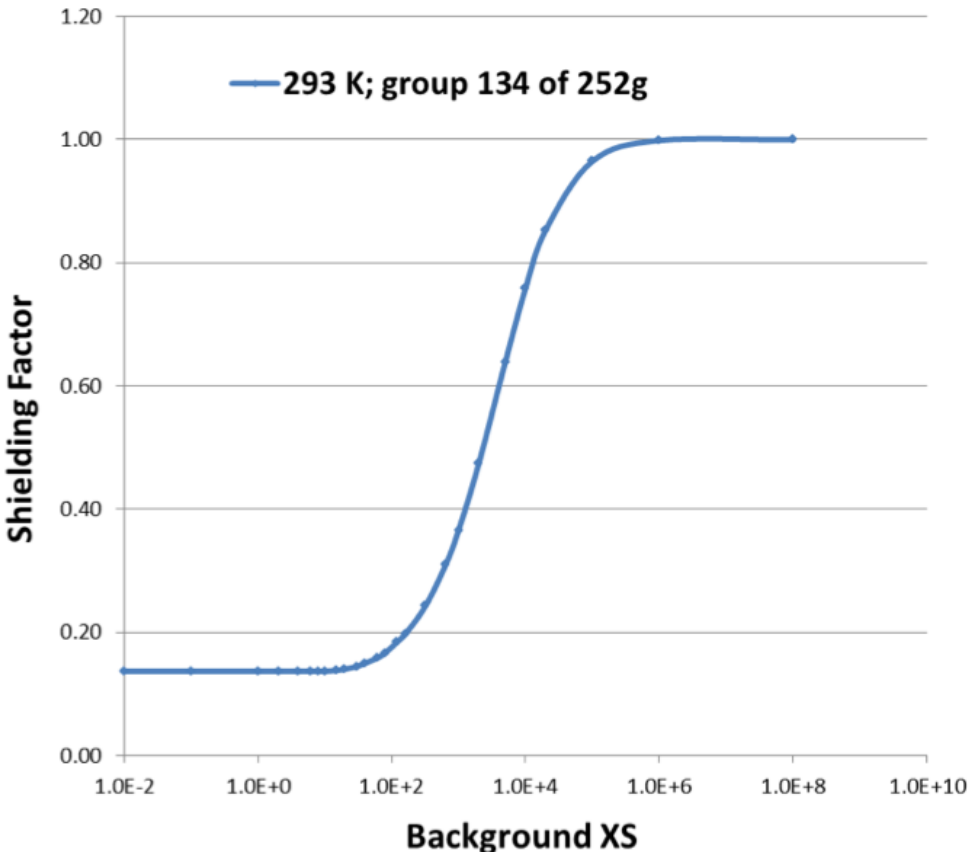


Fig. 7.3.1: Plot of f-factor variation for ^{238}U capture reaction.

Interpolation of the f-factors can be problematic, and several different schemes have been developed for this purpose. Some of the interpolation methods that have been used in other codes are constrained Lagrangian, [BONAMIDYB77] arc-tangent fitting, [BONAMIKid74] and an approach developed by Segev [BONAMISeg81]. All of these were tested and found to be inadequate for use with the SCALE libraries, which may have multiple energy groups within a single resonance. BONAMI uses a unique interpolation method developed by Greene, which is described in [BONAMIGre82]. Greene's interpolation method is essentially a polynomial approach in which the powers of the polynomial terms can vary within a panel, as shown in Eq. (7.3.25):

$$f(\sigma) = f(\sigma_1) + \frac{\sigma^{q(\sigma)} - \sigma_1^{q(\sigma)}}{\sigma_2^{q(\sigma)} - \sigma_1^{q(\sigma)}} (f(\sigma_2) - f(\sigma_1)) , \qquad (7.3.22)$$

where

$$q(\sigma) = q(\sigma_1) + \frac{\sigma - \sigma_1}{\sigma_2 - \sigma_1} (q(\sigma_2) - q(\sigma_1)) . \qquad (7.3.23)$$

Fig. 7.3.2 illustrates the expected behavior of Eq. (7.3.22) caused by varying the powers in a panel.

By allowing the power q to vary as a function of independent variable *sigma*, we can move between the various monotonic curves on the graph in a monotonic fashion. Note that when p crosses the $p = 1$ curve, the shape changes from concave to convex, or vice versa.

This shape change means that we can use the scheme to introduce an inflection point, which is exactly the situation needed for interpolating f-factors.

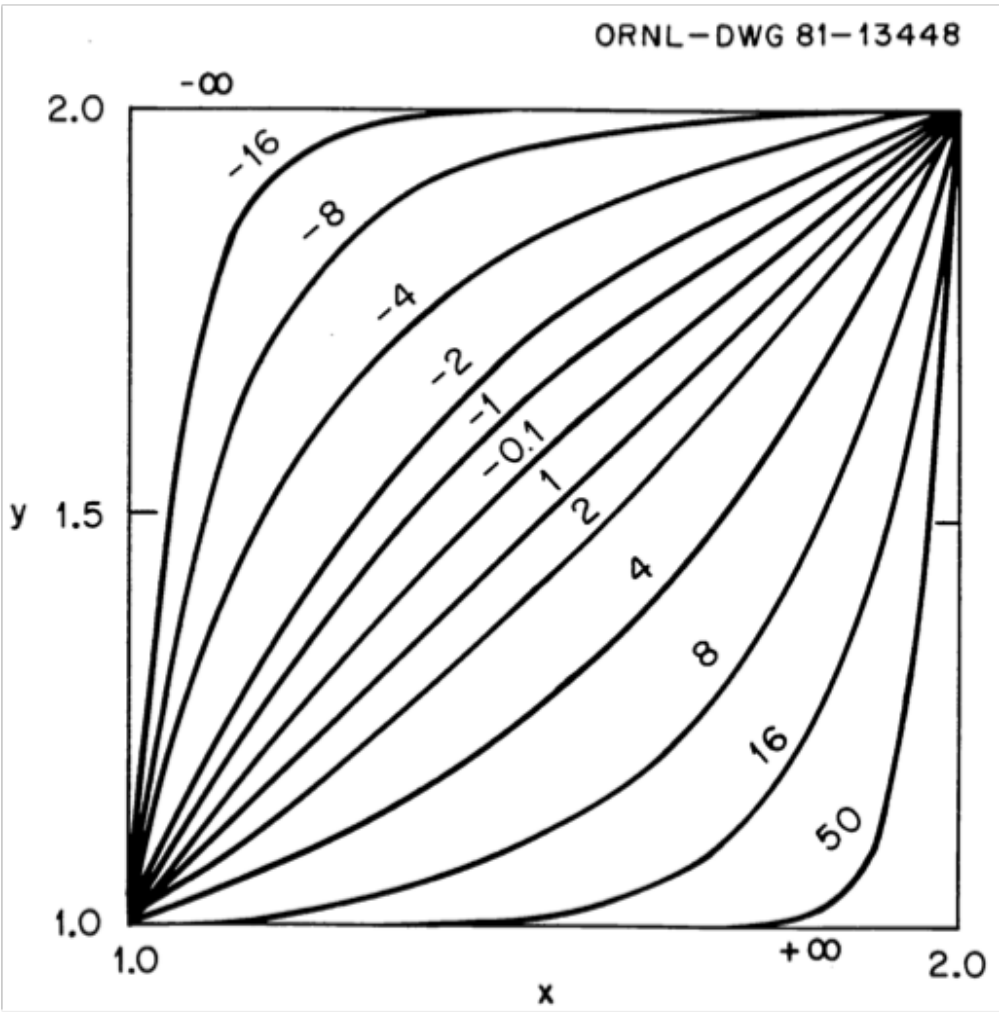


Fig. 7.3.2: Illustration of the effects of varying "powers" in the Greene interpolation method.

Fig. 7.3.3 and Fig. 7.3.3 show typical "fits" of the f-factors using the Greene interpolation scheme for two example cases. Note, in particular, that since this scheme has guaranteed monotonicity, it easily accommodates the end panels that have the smooth asymptotic variation. Even considering the extra task of having to determine the powers for temperature and σ_0 interpolations, the method is not significantly more time-consuming than the alternative schemes for most applications.

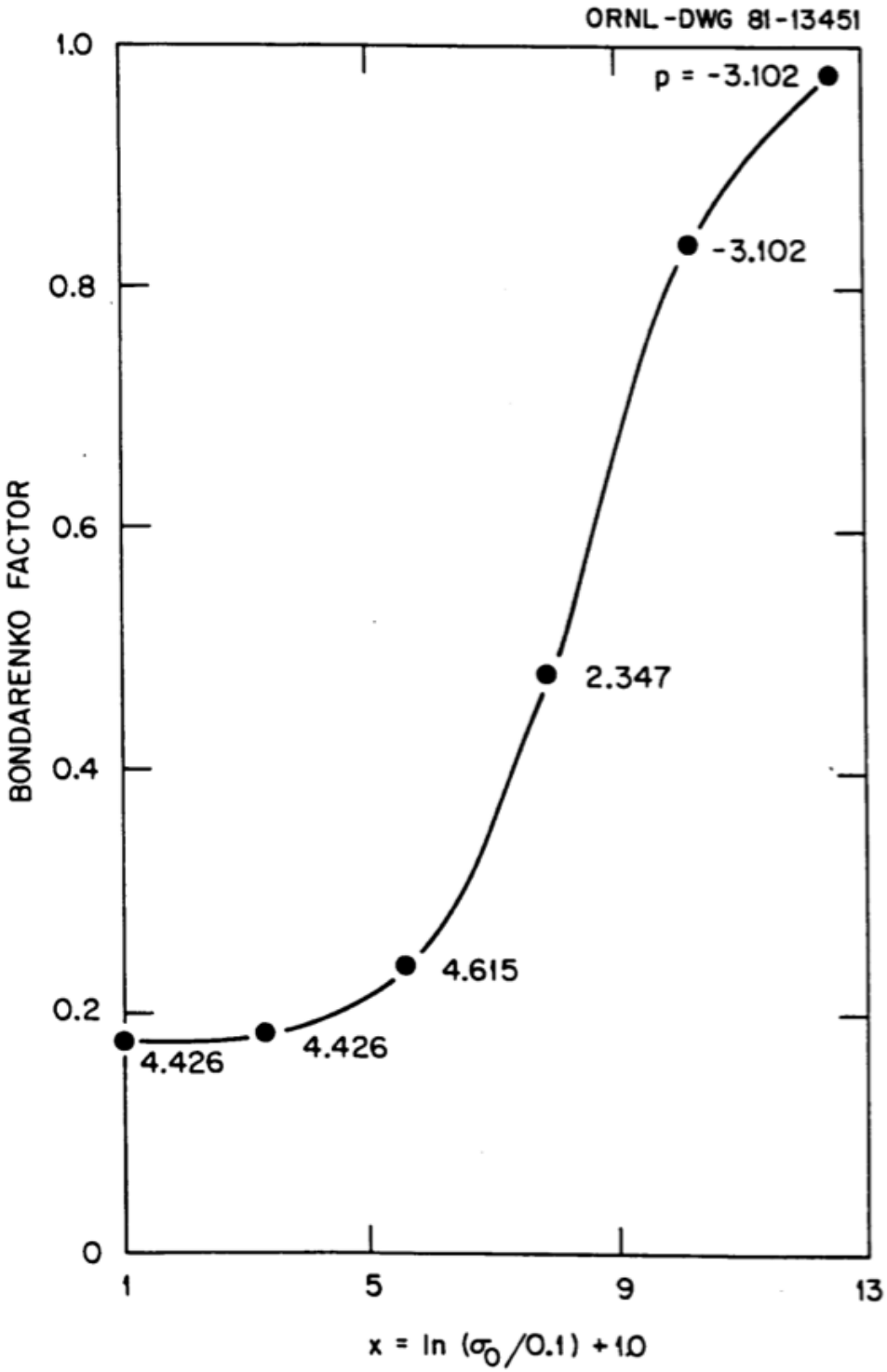


Fig. 7.3.3: Use of Greene's method to fit the σ_0 variation of Bondarenko factors for case 1.

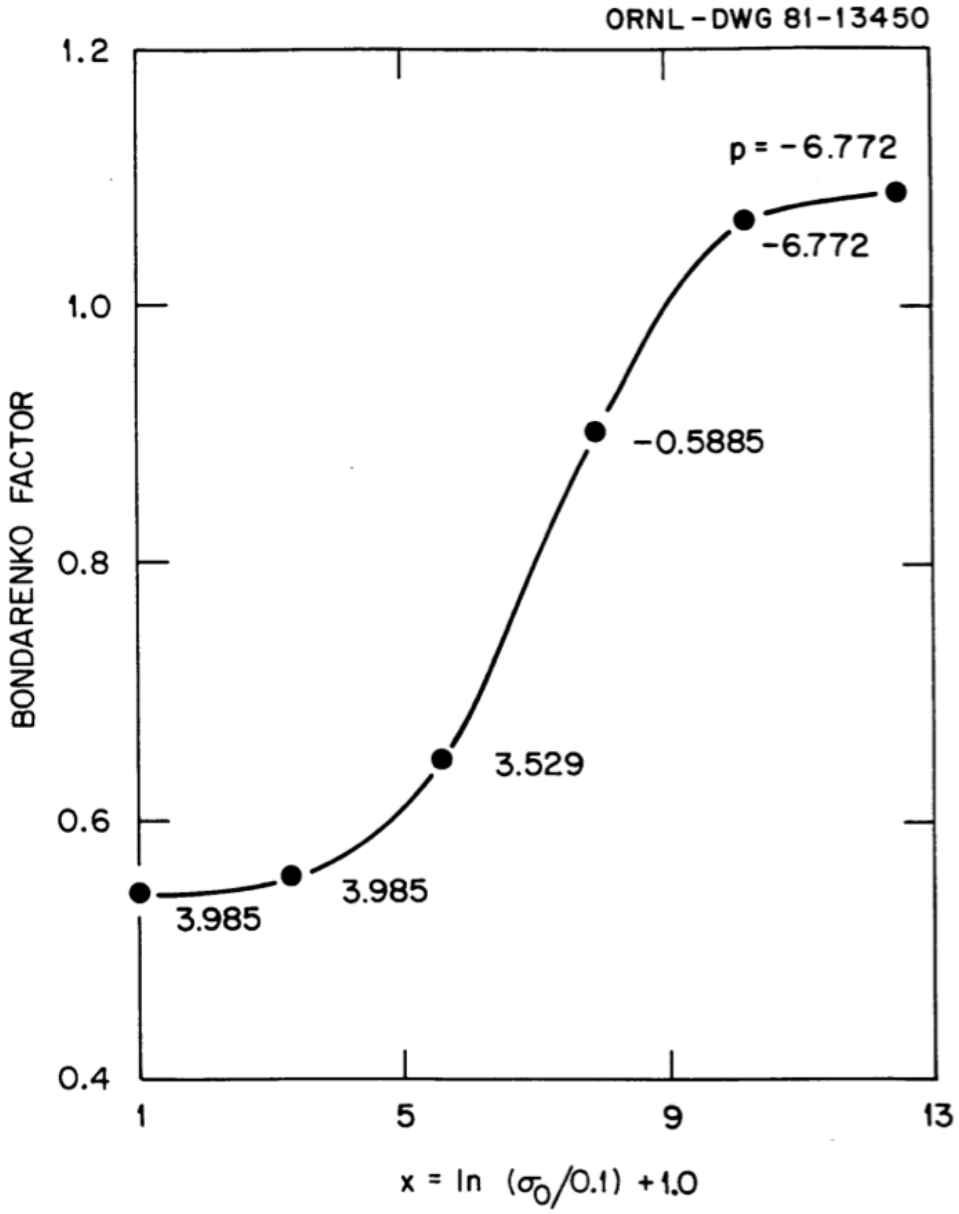


Fig. 7.3.4: Use of Greene's method to fit the σ_0 variation of Bondarenko factors for case 2.

7.3.4 INPUT INSTRUCTIONS

BONAMI is most commonly used as an integral component of XSProc through SCALE automated analysis sequences. XSProc automatically prepares all the input data for BONAMI and links it with the other self-shielding modules. During a SCALE sequence execution, the data are provided directly to BONAMI in memory through XSProc. Some of the input parameters can be modified in the MOREDATA block in XSProc.

However, the legacy interface to execute stand-alone BONAMI calculations has been preserved for expert users. The legacy input to BONAMI uses the FIDO schemes described in the FIDO chapter of the SCALE manual. The BONAMI input for standalone execution is given below, where the MOREDATA input keywords are marked in bold.

Data Block 1

0$ Logical Unit Assignments [4]

1. masterlib- input master library (Default = 23)
2. mwt-not used
3. msc-not used
4. newlib-output master library (Default = 22)

1$ Case Description [6]

1. cellgeometry-geometry description

   0 homogeneous

   1 slab

   2 cylinder

   3 sphere

2. numzones-number of zones or material regions

3. mixlength-mixing table length. This is the total number of entries needed to describe the concentrations of all constituents in all mixtures in the problem.

4. ib-not used
5. **crossedt**-output edit option

   0 no output (Default)

   1 input echo

   2 iteration list, timing

   3 background cross section calculation details

   4 shielded cross sections, Bondarenko factors

6. issopt-not used
7. **iropt-**resonance approximation option

   0 NR (Default) (Bondarenko iterations)

   1 IR with potential scattering

   2 IR with absorption and potential scattering (Bondarenko iterations)

   3 IR with absorption and elastic scattering (Bondarenko iterations)

8. **bellopt-**Bell factor calculation option

   0 Otter 1 Leslie (Default)

9. **escxsopt-**escape cross section calculation option

   0 consistent

   1 inconsistent (Default)

2* Floating-Point Constants [2]

1. **bonamieps**-convergence criteria for the Bondarenko iteration (Default = 0.001)

2. **bellfact**-geometrical escape probability adjustment factor. See notes below on this parameter (Default = 0.0).

T Terminate Data Block 1.

Data Block 2

3$ Mixture numbers in the mixing table [mixlength]

4$ Component (nuclide) identifiers in the mixing table [mixlength]

5* Concentrations (atoms/b-cm) in the mixing table [mixlength]

6$ Mixtures by zone [numzones]

7* Outer radii (cm) by zone [numzones]

8* Temperature (k) by zone [numzones]

9* Escape cross section (cm^{-1}) by zone [numzones]

10$ Not used

11$ Not used

12* Temperature (K) of the nuclide in a one-to-one correspondence with the mixing table arrays.

13* Dancoff factors by zone [numzones]

14* Lbar ($\bar{\ell}$) factors by zone [numzones]

T Terminate Data Block 2.

This concludes the input data required by BONAMI.

7.3.4.1 Notes on input

In the 1$ array, *cellgeometry* specifies the geometry. The geometry information is used in conjunction with the 7* array to calculate mean chord length *Lbar* if it is not provided by the user in the 14* array.

numzones, the number of zones, may or may not model a real situation. It may, for example, be used to specify *numzones* independent media to perform a cell calculation in parallel with one or more infinite medium calculations. The geometry description in 1$ array applies only to mean chord length calculations unless it is provided in 14*.

In the 2* array, *bonamieps* is used to specify the convergence expected on all macroscopic total values by zone, that is, each $\Sigma_t(g, j)$ in group g and zone j is converged so that

$$\frac{\left|\Sigma_t^i(g,j) \quad - \quad \Sigma_t^{i-1}(g,j)\right|}{\Sigma_t^i(g,j)} \quad \leq \quad bonamieps \quad . \tag{7.3.24}$$

The "Bell" factor in the 2* array is the parameter used to adjust the Wigner rational approximation for the escape probability to a more correct value. It has been suggested that if one wishes to use one constant value, the Bell factor should be 1.0 for slabs and 1.35 otherwise. In the ordinary case, BONAMI defaults the Bell factor to zero and uses a prescription by Otter [BONAMIOtt64] to determine a cross-section geometry-dependent value of the Bell factor for isolated absorber bodies. It uses a prescription by Leslie[6] to determine

the Dancoff factor–dependent values of the Bell factor for lattices, which are much more accurate than the single value. The user who wishes to determine the constant value can, however, use it by inputting a value other than zero.

The 3$, 4$, and 5* arrays are used to specify the concentrations of the constituents of all mixtures in the problem as follows:

Entry 3$ (Mixture Number) 4$ (Nuclide ID) 5* (Concentrations)

1

2

.

.

.

.

mixlength

.

Because of the manner in which BONAMI references the nuclides in a calculation, each nuclide in the problem must have a unique entry in the mixing table. Thus one cannot specify a mixture and subsequently load it into more than one zone, as can be the case with many modules requiring this type of data.

The 12* array is used to allow varying the temperatures by nuclide within a zone. In the event this array is omitted, the 12* array will default by nuclide to the temperature of the zone containing the nuclide.

The mixture numbers in each zone are specified in the 6$ array. Mixture numbers are arbitrary and need only match up with those used in the 3$ array.

The radii in the 7* array are referenced to a zero value at the left boundary of the system.

In the event the temperatures in the 8* array are not bounded by temperature values in the Bondarenko tables, BONAMI will extrapolate using the three temperature points closest to the value. For example, a request for 273 K for a nuclide with Bondarenko sets at 300, 900, and 2,100 K would use the polynomial fit from those three temperature points to extrapolate the 273 K value.

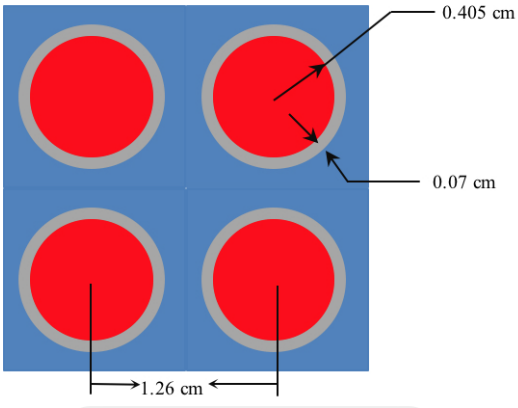
The escape cross sections in the 9* array allow a macro escape cross section (Σ_e^{input}) to be specified by zone. (This array can be ignored if Dancoff factors are provided.) If the Dancoff factor for a zone is specified as -1 in the input, then the user-specified escape cross section is used in calculating the background cross sections σ_0 as follows:

$$\sigma_0 = \frac{\sum_{n \neq i} N_n \sigma_t^n + \Sigma_e^{input}}{N_i} \tag{7.3.25}$$

7.3.5 SAMPLE PROBLEM

In most cases, the input data to BONAMI are simple and obvious because the complicated parameters are determined internally based on the options selected. The user describes his geometry, the materials contained therein, the temperatures, and a few options.

This problem is for a system of iron-clad uranium (U^{238} – U^{235}) fuel pins arranged in a square lattice in a water pool.



Our number densities are

Fuel:

$$N_{^{235}U} = 1.4987 \times 10^{-4}$$

$$N_{^{238}U} = 2.0664 \times 10^{-2}$$

Clad:

$$N_{^{56}Fe} = 9.5642 \times 10^{-5}$$

Water:

$$N_H = 6.6662 \times 10^{-2}$$

$$N_O = 3.3331 \times 10^{-2}$$

For the problem, we choose *iropt* = 1 (IR approximation with scattering approximated by math:*lambda*Σ_p) and *crossedt* = 4 for the most detailed output edits. An 8-group test library is used for fast execution and a short output file.

The XSProc/CSAS1X SCALE sequence input file, the corresponding i_bonami FIDO input file created by the sequence under the temporary working directory, and an abbreviated copy of the output from this case follows.

```
=csas1x
Assembly pin
test-8grp
read comp
' fuel
u-235  1 0 1.4987e-4 297.15 end
u-238  1 0 2.0664e-2 297.15 end
' clad
fe-56  2  0 9.5642e-5 297.15 end
' coolant
h      3 0 6.6662e-2 297.15 end
o      3 0 3.3331e-2 297.15 end
end comp
' ================================================================
read celldata
latticecell squarepitch  pitch=1.26 3  fuelr=0.405765 1
```

(continues on next page)

(continued from previous page)

```
                                    cladr=0.47498  2 end

moredata iropt=1 crossedt=4 end moredata
end celldata
' ===============================================================
end
```

FIDO input i_bonami

```
-1$$  a0001
    500000
e
 0$$  a0001
       11           0          18           1
e
 1$$  a0001
         1           3           5           0           4        1010
         1          -1          -1
e
2**  a0001
 1.00000E-03   0.00000E+00
 e
t
 3$$  a0001
         1           1           2           3           3
e
 4$$  a0001
     92235       92238       26056        1001        8016
e
5**  a0001
 1.49870E-04   2.06640E-02   9.56420E-05   6.66620E-02   3.33310E-02
 e
 6$$  a0001
         1           2           3
e
7**  a0001
 4.05765E-01   4.74980E-01   7.10879E-01
 e
8**  a0001
 2.97150E+02   2.97150E+02   2.97150E+02
 e
9**  a0001
 1.11870E+00   4.15813E+00   1.78119E-01
 e
10$$  a0001
     92235       92238       26056        1001        8016
e
11$$  a0001
         0           0           0
e
13**  a0001
 2.71260E-01   5.20852E-01   9.24912E-01
 e
14**  a0001
 8.11530E-01   1.38430E-01   4.71798E-01
 e
15**  a0001
 0.00000E+00   0.00000E+00   0.00000E+00   0.00000E+00
 e
16$$  a0001
         2           2           2
e
17$$  a0001
         0           0           0           0
e
```

(continues on next page)

(continued from previous page)

```
t
```

```
                                  program verification information

                               code system:  SCALE    version:  6.2




          program:  bonami

    creation date:   unknown

          library:  /home02/u2m/Workfolder/sampletmp


        test code:  bonami

          version:  6.2.0

          jobname:  u2m

     machine name:  node22.ornl.gov

date of execution:  04_dec_2013

time of execution:  21:43:54.38
```

```
1
                BONAMI CELL PARAMETERS
-----------------------------------------
Bonami Print Option          : 4
BellFactor                   : 0
Bondarenko Iteration eps     : 0.001
Resonance Option             : 1
Bell Factor  Option          : LESLIE
Escape CrossSection  Option  : INCONSISTENT
CellGeometry                 : 2
MasterLibrary                :
Number oF Neutron Groups     : 8
First Thermal Neutron Group  : 5
_________________________________________
Processing Zone               : 1
Mixture Number                : 1
Number Of Nuclides            : 2
Dancoff Factor                : 0.27126
Lbar                          : 0.81153
Escape Cross Section Input    : 1.1187
Material Temeprature          : 297.15

Processing Nuclide :  92235  Number Density : 0.00014987
Processing Nuclide :  92238  Number Density : 0.020664

Bondarenko Iterations
iteration    Nuclide Group   MaxChange   Selfsig0     Effsig0
     1        92235     0             0          0   0
     1        92238     0             0          0   0

Total number of Bondarenko Iterations  : 1
Max Change in Group                    : 0

Group  Eff Macro Sig0      Escape Xsec
   1      0.2351032         0.9075513
   2      0.2351032         0.9075513
```

(continues on next page)

(continued from previous page)

```
   3      0.2351032         0.9075513
   4      0.2351032         0.9075513
   5      0.2351032         0.9075513
   6      0.2351032         0.9075513
   7      0.2351032         0.9075513
   8      0.2351032         0.9075513

-----------------------------------------------------
```

```
Shielding Nuclide 92235

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
   1     1    7612.71875         7.19131      0.999998   7.19129
   1     2    7612.71875         10.2521      0.999616   10.2481
   1     3    7612.71875         24.9361       1.00241   24.9963
   1     4    7612.71875         75.1109       1.05902   79.5436
   1     5    7612.71875         56.0286       1.00205   56.1434
   1     6    7612.71875         198.645        1.0008   198.805
   1     7    7612.71875         347.945       1.00024   348.028
   1     8    7612.71875         761.257        1.0066   766.282

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
   2     1    7612.71875         3.71448      0.999999   3.71448
   2     2    7612.71875         7.63235       0.99935   7.62739
   2     3    7612.71875          11.841      0.999444   11.8345
   2     4    7612.71875         11.5408       1.00561   11.6055
   2     5    7612.71875         12.5449       1.00001   12.545
   2     6    7612.71875         14.2501       1.00007   14.2511
   2     7    7612.71875         14.8125       1.00003   14.8128
   2     8    7612.71875         15.1274       1.00015   15.1297

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
  18     1    7612.71875         1.21846      0.999996   1.21846
  18     2    7612.71875         1.40834        1.0002   1.40862
  18     3    7612.71875         8.92885       1.00132   8.94062
  18     4    7612.71875         39.2086       1.06274   41.6686
  18     5    7612.71875         32.7026       1.00105   32.737
  18     6    7612.71875         153.511       1.00089   153.647
  18     7    7612.71875         285.775       1.00026   285.848
  18     8    7612.71875         636.445       1.00655   640.611

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
 102     1    7612.71875        0.060296             1   0.0602962
 102     2    7612.71875        0.317627       1.00352   0.318746
 102     3    7612.71875         4.16593       1.01325   4.22113
 102     4    7612.71875         24.3615       1.07832   26.2695
 102     5    7612.71875          10.781       1.00749   10.8618
 102     6    7612.71875         30.8844       1.00074   30.9073
 102     7    7612.71875         47.3579        1.0002   47.3671
 102     8    7612.71875         109.685       1.00781   110.542

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
1007     1    7612.71875               0             0   0
1007     2    7612.71875               0             0   0
1007     3    7612.71875               0             0   0
1007     4    7612.71875               0             0   0
1007     5    7612.71875         12.5448       1.00001   12.5449
1007     6    7612.71875         14.2501       1.00007   14.2511
1007     7    7612.71875         14.8125       1.00003   14.8129
1007     8    7612.71875         15.1278       1.00015   15.13

-----------------------------------------------------
```

```
Shielding Nuclide 92238

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
   1     1    44.0034676         7.33815      0.999983   7.33803
   1     2    44.0034676         10.3566       1.00418   10.3999
   1     3    44.0034676         15.0517      0.976844   14.7032
   1     4    44.0034676          15.951      0.983793   15.6925
   1     5    44.0034676         9.43867       1.00002   9.43887
   1     6    44.0034676         10.1008       1.00008   10.1015
   1     7    44.0034676         10.7744       1.00004   10.7748
   1     8    44.0034676         12.2124       1.00145   12.2301

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
   2     1    44.0034676          4.0228      0.999974   4.0227
   2     2    44.0034676         9.05886       1.00575   9.11093
   2     3    44.0034676         14.0213      0.979923   13.7398
   2     4    44.0034676         11.9032       0.98795   11.7598
   2     5    44.0034676         8.86555      0.999984   8.86541
   2     6    44.0034676         9.24452       1.00002   9.24471
   2     7    44.0034676          9.2797       1.00002   9.27987
   2     8    44.0034676          9.3077       1.00009   9.30853

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
  18     1    44.0034676        0.376356       1.00001   0.376361
  18     2    44.0034676     0.000528746       1.00019   0.000528845
  18     3    44.0034676     0.000308061      0.966052   0.000297603
  18     4    44.0034676     4.75014e-06      0.967842   4.59738e-06
  18     5    44.0034676     2.60878e-06       1.00006   2.60893e-06
  18     6    44.0034676     5.27139e-06       1.00071   5.27512e-06
  18     7    44.0034676      9.3235e-06       1.00018   9.32514e-06
  18     8    44.0034676     1.81868e-05       1.00588   1.82937e-05

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
 102     1    44.0034676       0.0554327       1.00006   0.0554359
 102     2    44.0034676         0.17972      0.978628   0.175879
 102     3    44.0034676         1.03011      0.934934   0.963087
 102     4    44.0034676         4.04777      0.971568   3.93268
 102     5    44.0034676        0.573119        1.0006   0.573462
 102     6    44.0034676        0.856257       1.00068   0.856839
 102     7    44.0034676         1.49471       1.00017   1.49497
 102     8    44.0034676         2.90465       1.00586   2.92168

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
1007     1    44.0034676                0             0   0
1007     2    44.0034676                0             0   0
1007     3    44.0034676                0             0   0
1007     4    44.0034676                0             0   0
1007     5    44.0034676         8.86549      0.999984   8.86535
1007     6    44.0034676         9.24445       1.00002   9.24463
1007     7    44.0034676         9.27974       1.00002   9.27992
1007     8    44.0034676         9.30769       1.00009   9.30852
Zone Calculation is completed in 0 seconds
                BONAMI CELL PARAMETERS
-----------------------------------------------
Bonami Print Option          : 4
BellFactor                   : 0
Bondarenko Iteration eps     : 0.001
Resonance Option             : 1
Bell Factor  Option          : LESLIE
Escape CrossSection  Option  : INCONSISTENT
CellGeometry                 : 2
MasterLibrary                :
Number oF Neutron Groups     : 8
First Thermal Neutron Group  : 5
-----------------------------------------
Processing Zone               : 2
```

(continues on next page)

(continued from previous page)

```
Mixture Number               : 2
Number Of Nuclides           : 1
Dancoff Factor               : 0.520852
Lbar                         : 0.13843
Escape Cross Section Input   : 4.15813
Material Temeprature         : 297.15

Processing Nuclide :  26056  Number Density : 9.5642e-05

Bondarenko Iterations
iteration    Nuclide Group   MaxChange   Selfsig0     Effsig0
     1        26056     0             0          0   0

Total number of Bondarenko Iterations  : 1
Max Change in Group                    : 0

Group  Eff Macro Sig0      Escape Xsec
   1     0.0003553244          3.487286
   2     0.0003553244          3.487286
   3     0.0003553244          3.487286
   4     0.0003553244          3.487286
   5     0.0003553244          3.487286
   6     0.0003553244          3.487286
   7     0.0003553244          3.487286
   8     0.0003553244          3.487286

--------------------------------------------------
```

```
Shielding Nuclide 26056

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
   1     1    36461.8672         3.07957       1.00005   3.07972
   1     2    36461.8672         4.68958       1.00091   4.69382
   1     3    36461.8672         7.85712      0.999843   7.85589
   1     4    36461.8672         12.0029             1   12.0029
   1     5    36461.8672         12.3689       1.00001   12.369
   1     6    36461.8672         12.8598       1.00003   12.8602
   1     7    36461.8672         13.5237      0.999906   13.5224
   1     8    36461.8672         15.0714       0.99949   15.0637

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
   2     1    36461.8672         2.26476       1.00047   2.26583
   2     2    36461.8672          4.6817        1.0009   4.68592
   2     3    36461.8672         7.81457      0.999813   7.81311
   2     4    36461.8672         11.9143             1   11.9143
   2     5    36461.8672         12.0468       1.00001   12.0469
   2     6    36461.8672          12.065       1.00002   12.0653
   2     7    36461.8672         12.0887       1.00005   12.0893
   2     8    36461.8672         12.2042       1.00013   12.2057

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
 102     1    36461.8672      0.00206393        1.0015   0.00206702
 102     2    36461.8672      0.00787763        1.0035   0.00790524
 102     3    36461.8672       0.0425504       1.00623   0.0428155
 102     4    36461.8672       0.0885525             1   0.0885529
 102     5    36461.8672        0.322101       1.00002   0.322109
 102     6    36461.8672        0.794804        1.0002   0.79496
 102     7    36461.8672         1.43496      0.998734   1.43314
 102     8    36461.8672         2.86723      0.996792   2.85803

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
1007     1    36461.8672               0             0   0
1007     2    36461.8672               0             0   0
1007     3    36461.8672               0             0   0
```

(continues on next page)

(continued from previous page)

```
1007     4    36461.8672               0             0   0
1007     5    36461.8672         12.0468       1.00001   12.0469
1007     6    36461.8672          12.065       1.00002   12.0653
1007     7    36461.8672         12.0887       1.00005   12.0893
1007     8    36461.8672         12.2042       1.00013   12.2057
Zone Calculation is completed in 0 seconds
```

```
BONAMI CELL PARAMETERS
-----------------------------------------------
Bonami Print Option          : 4
BellFactor                   : 0
Bondarenko Iteration eps     : 0.001
Resonance Option             : 1
Bell Factor  Option          : LESLIE
Escape CrossSection  Option  : INCONSISTENT
CellGeometry                 : 2
MasterLibrary                :
Number oF Neutron Groups     : 8
First Thermal Neutron Group  : 5
-----------------------------------------
Processing Zone               : 3
Mixture Number                : 3
Number Of Nuclides            : 2
Dancoff Factor                : 0.924912
Lbar                          : 0.471798
Escape Cross Section Input    : 0.178119
Material Temeprature          : 297.15

Processing Nuclide :   1001  Number Density : 0.066662
Processing Nuclide :   8016  Number Density : 0.033331

Bondarenko Iterations
iteration    Nuclide Group   MaxChange   Selfsig0     Effsig0
     1         1001     0             0          0   0
     1         8016     0             0          0   0

Total number of Bondarenko Iterations  : 1
Max Change in Group                    : 0

Group  Eff Macro Sig0      Escape Xsec
   1       1.494705         0.1593803
   2       1.494705         0.1593803
   3       1.494705         0.1593803
   4       1.494705         0.1593803
   5       1.494705         0.1593803
   6       1.494705         0.1593803
   7       1.494705         0.1593803
   8       1.494705         0.1593803

-------------------------------------------------
```

```
Shielding Nuclide 1001

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
   1     1    4.33502197         2.98905      0.999485   2.98751
   1     2    4.33502197         9.87269      0.999169   9.86448
   1     3    4.33502197         19.9332      0.999972   19.9326
   1     4    4.33502197         20.4672      0.998926   20.4453
   1     5    4.33502197         21.1735       1.00001   21.1736
   1     6    4.33502197         26.1886       0.99995   26.1873
   1     7    4.33502197         35.0621      0.999821   35.0558
   1     8    4.33502197         54.9507      0.997361   54.8057

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
```

(continues on next page)

(continued from previous page)

```
   2     1    4.33502197         2.98901      0.999485   2.98747
   2     2    4.33502197          9.8726      0.999168   9.86439
   2     3    4.33502197         19.9315      0.999972   19.9309
   2     4    4.33502197         20.4556      0.998926   20.4336
   2     5    4.33502197         21.1321       1.00001   21.1322
   2     6    4.33502197         26.0865       0.99995   26.0852
   2     7    4.33502197         34.8778      0.999829   34.8718
   2     8    4.33502197         54.5786      0.997396   54.4365

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
 102     1    4.33502197     3.56422e-05       1.00003   3.56433e-05
 102     2    4.33502197     8.96827e-05      0.998165   8.9518e-05
 102     3    4.33502197      0.00171679      0.999794   0.00171643
 102     4    4.33502197       0.0116042       1.00002   0.0116044
 102     5    4.33502197       0.0413709       1.00001   0.0413714
 102     6    4.33502197        0.102043       1.00007   0.10205
 102     7    4.33502197        0.184322      0.998389   0.184025
 102     8    4.33502197        0.372079      0.992163   0.369163

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
1007     1    4.33502197               0             0   0
1007     2    4.33502197               0             0   0
1007     3    4.33502197               0             0   0
1007     4    4.33502197               0             0   0
1007     5    4.33502197         21.1312       1.00005   21.1322
1007     6    4.33502197         26.0871      0.999927   26.0852
1007     7    4.33502197         34.8779      0.999826   34.8718
1007     8    4.33502197         54.5802      0.997367   54.4365

--------------------------------------------------
```

```
Shielding Nuclide 8016

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
   1     1    45.7377472         2.36081      0.980071   2.31376
   1     2    45.7377472         3.95917      0.995755   3.94236
   1     3    45.7377472         3.84394             1   3.84394
   1     4    45.7377472         3.85289             1   3.85291
   1     5    45.7377472         3.85531       1.00002   3.85537
   1     6    45.7377472          3.8648       1.00006   3.86502
   1     7    45.7377472         3.89139       1.00014   3.89194
   1     8    45.7377472         4.01909       1.00036   4.02056

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
   2     1    45.7377472         2.34636      0.979939   2.29929
   2     2    45.7377472         3.95907      0.995756   3.94226
   2     3    45.7377472         3.84393             1   3.84393
   2     4    45.7377472         3.85288             1   3.8529
   2     5    45.7377472         3.85529       1.00002   3.85535
   2     6    45.7377472         3.86474       1.00006   3.86496
   2     7    45.7377472         3.89128       1.00014   3.89183
   2     8    45.7377472         4.01888       1.00037   4.02035

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
 102     1    45.7377472      0.00010111        1.0002   0.00010113
 102     2    45.7377472     0.000101635      0.991292   0.00010075
 102     3    45.7377472     7.10826e-06       1.00036   7.11084e-06
 102     4    45.7377472      7.2788e-06       1.00001   7.27885e-06
 102     5    45.7377472     2.38519e-05       1.00003   2.38526e-05
 102     6    45.7377472     5.83622e-05       1.00019   5.83734e-05
 102     7    45.7377472     0.000105195      0.998729   0.000105061
 102     8    45.7377472     0.000209981      0.996663   0.00020928

mt   Group      sig0      infDiluted Xsec    f-factor  shielded Xsec
```

(continues on next page)

(continued from previous page)

```
1007     1    45.7377472               0             0   0
1007     2    45.7377472               0             0   0
1007     3    45.7377472               0             0   0
1007     4    45.7377472               0             0   0
1007     5    45.7377472         3.85523       1.00003   3.85535
1007     6    45.7377472         3.86483       1.00003   3.86496
1007     7    45.7377472         3.89119       1.00016   3.89183
1007     8    45.7377472         4.01794        1.0006   4.02035
Zone Calculation is completed in 0 seconds
 module: BonamiM  has terminated after a cpu usage of  0.0100  seconds
```

7.4 CENTRM: A NEUTRON TRANSPORT CODE FOR COMPUTING CONTINUOUS-ENERGY SPECTRA IN GENERAL ONE-DIMENSIONAL GEOMETRIES AND TWO-DIMENSIONAL LATTICE CELLS

M. L. Williams and K. S. Kim

Abstract

CENTRM computes continuous-energy neutron spectra for infinite media, general one-dimensional (1D) systems, and two-dimensional (2D) unit cells in a lattice, by solving the Boltzmann transport equation using a combination of pointwise and multigroup nuclear data. Several calculational options are available, including a slowing-down computation for homogeneous infinite media, 1D discrete ordinates in slab, spherical, or cylindrical geometries; a simplified two-region solution; and 2D method of characteristics for a unit cell within a square-pitch lattice. In SCALE, CENTRM is used mainly to calculate problem-specific fluxes on a fine energy mesh (10,000–70,000 points), which may be used to generate self-shielded multigroup cross sections for subsequent radiation transport computations.

ACKNOWLEDGMENTS

Several current and former ORNL staff made valuable contributions to the CENTRM development. The author acknowledges the contributions of former ORNL staff members D. F. Hollenbach and N. M. Greene; as well as current staff member L. M. Petrie. Portions of the original code development were performed by M. Asgari as partial fulfillment of his PhD dissertation research at Louisiana State University (LSU); and Riyanto Raharjo from LSU made significant programming contributions for the inelastic scattering and thermal calculations.

7.4.1 INTRODUCTION

CENTRM (**C**ontinuous **EN**ergy **TR**ansport **M**odule) computes "continuous-energy" neutron spectra using various deterministic approximations to the Boltzmann transport equation. Computational methods are available for infinite media, general one-dimensional (1-D) geometries, and two-dimensional (2D) unit cells in a square-pitch lattice. The purpose of the code is to provide fluxes and flux moments for applications that require a high resolution of the fine-structure variation in the neutron energy spectrum. The major function of CENTRM is to determine problem-specific fluxes for processing multigroup (MG) data with the XSProc self-shielding module (Introduction in XSProc chapter), which is executed by all SCALE MG sequences. XSProc calls an application program interface (API) to perform a CENTRM calculation for a representative model (e.g., a unit cell in a lattice), and then utilizes the spectrum as a *problem-dependent* weight function for MG averaging. The MG data processing is done in XSProc by calling an API for the PMC code, which uses the CENTRM continuous-energy (CE) flux spectra and cross-section data to calculate group-averaged cross sections over some specified energy range. The resulting application-specific library is used for MG

neutron transport calculations within SCALE sequences. In this approach the CENTRM/PMC cross-section processing in XSProc becomes an active component in the overall transport analysis. CENTRM can also be executed as a standalone code, if the user provides all required input data and nuclear data libraries; but execution through XSProc is much simpler and less prone to error.

7.4.1.1 Description of problem solved

CENTRM uses a combination of MG and pointwise (PW) solution techniques to solve the neutron transport equation over the energy range ~0 to 20 MeV. The calculated CE spectrum consists of PW values for the flux per unit lethargy defined on a discrete energy mesh, for which a linear variation of the flux between energy points is assumed. Depending on the specified transport approximation, the flux spectrum may vary as a function of space and direction, in addition to energy. Spherical harmonic moments of the angular flux, which may be useful in processing MG matrices for higher order moments of the scattering cross section, can also be determined as a function of space and energy mesh.

CENTRM solves the fixed-source (inhomogeneous) form of the transport equation, with a user-specified fixed source term. The input source may correspond to MG histogram spectra for volumetric or surface sources or it may be a "fission source" which has a continuous-energy fission-spectrum distribution (computed internally) appropriate for each fissionable mixture. Note that eigenvalue calculations are *not* performed in CENTRM-these must be performed by downstream MG transport codes that utilize the self-shielded data processed with the CENTRM spectra.

7.4.1.2 Nuclear data required for CENTRM

A MG cross section library, a CE cross section library, and a CE thermal kernel [S(α, β)] library are required for the CENTRM transport calculation. During XSProc execution for a given unit cell in the CELL DATA block, the MG library specified in the input is processed by BONAMI prior to the CENTRM calculation, in order to provide self-shielded data based on the Bondarenko approximation for the MG component of the CENTRM solution. The shielded MG cross sections are also used in CENTRM to correct infinitely dilute CE data in the unresolved resoance range. The CRAWDAD module is executed by XSProc to generate the CENTRM CE cross section and thermal kernel libraries, respectively, by concatenating discrete PW data read from individual files for the nuclides in the unit cell mixtures. CE resonance profiles are based strictly on specifications in the nuclear data evaluations; e.g., Reich-Moore formalism is specified for most materials in ENDF/B-VII. PW data in the CENTRM library are processed such that values at any energy can be obtained by linear interpolation within some error tolerance specified during the library generation (usually ~0.1% or less). CRAWDAD also interpolates the CE cross section data and the Legendre moments of the thermal scattering kernels to the appropriate temperatures for the unit cell mixtures. The format of the CENTRM library is described in Sect. 7.4.6.1.

7.4.1.3 Code assumptions and features

As shown in Fig. 7.4.1, the energy range of interest is divided into three intervals called the Upper Multigroup Range (UMR), Pointwise Range (PW), and Lower Multigroup Range (LMR), respectively, which are defined by input. MG fluxes are computed using standard multigroup techniques for the UMR and LMR, and these values are then divided by the group lethargy width to obtain the average flux per lethargy within each group. This "pseudo-pointwise" flux is assigned to the midpoint lethargy of the group, so that there is one energy point per group in the UMR and LMR energy intervals. However, for each group in the PW range there are generally several, and possibly many, energy points for which CENTRM computes flux values. In this manner a problem-dependent spectrum is obtained over the entire energy range.

The default PW range goes from 0.001 eV to 20 keV, but the user can modify the PW limits. The energy range for the PW transport calculation is usually chosen to include the interval where the important absorber nuclides

have resolved resonances, while MG calculations are performed where the cross sections characteristically have a smoother variation or where shielding effects are less important. In the SCALE libraries the thermal range is defined to be energies less than 5.0 eV. Above thermal energies, scattering kinematics are based on the stationary nucleus model, while molecular motion and possible chemical binding effects are taken into account for thermal scattering, which can result in an incease in the neutron incident energy. The CENTRM thermal calculation uses Legendre coefficients from the CE kernel library that describes point-to-point energy transfers for incoherent and coherent scattering, as function of temperature, for all moderators that have thermal scattering law data provided in ENDF/B. Thermal kernels for all other materials are generated internally by CENTRM based on the free-gas model.

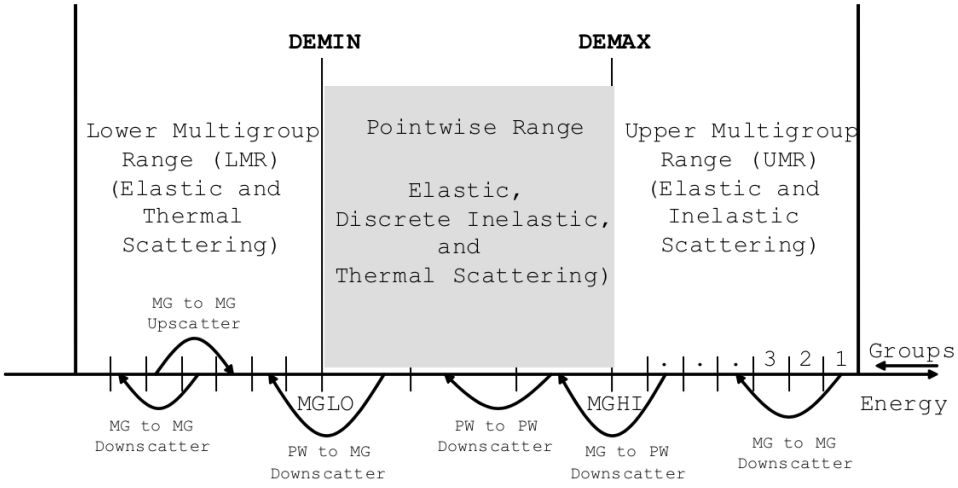


Fig. 7.4.1: Definition of UMR, PW, and LMR energy ranges.

Several transport computation methods are available for both MG and PW calculations. These include a space-independent slowing down calculation for infinite homogeneous media, 1D discrete ordinates or P1 methods for slab, spherical, and cylindrical geometries, and a 2D method of chracateristics (MoC) method for lattice unit cells. A simplified two-region collision-probablity method is also available for ther pointwise solution. In general the user may specify different transport methods for the UMR, PW, and LMR, respectively; however, if the 2D MoC method is specified for any range, it will be used for all.

The CENTRM 1D discrete ordinates calculation option has many of the same features as the SCALE MG code XSDRNPM. It represents the directional dependence of the angular flux with an arbitrary symmetric-quadrature order, and uses Legendre expansions up to P_5 to represent the scattering source. No restrictions are placed on the material arrangement or the number of spatial intervals in the calculation, and general boundary conditions (vacuum, reflected, periodic, albedo) can be applied on either boundary of the 1D geometry. Lattice cells are represented in the CENTRM discrete ordinates option by a 1D Wigner-Sitz cylindrical or spherical model with a white boundary condition on the outer surface.

Starting with SCALE-6.2, CENTRM also includes a 2D MoC solver for lattice cell geometries consisting of a cylindrical fuel rod (fuel/gap/clad) contained within a rectangular moderator region. The MoC calculation is presently limited to square lattices. The 2D unit cell uses a reflected boundary condition on the outer square surface, which provides a more rigorous treatment than the 1D Wigner-Seitz model; however the MoC option requires a longer execution time than the 1D discrete ordinates method. The MoC option has been found to improve results compared to the 1D Wigner-Seitz cell model for many cases, but in other cases the improvement is marginal.

A variable PW energy mesh is generated internally to accurately represent the fine-structure flux spectrum for the system of interest. This gives CENTRM the capability to rigorously account for resonance interference effects in systems with multiple resonance absorbers. Because CENTRM calculates the space-dependent PW flux spectrum, the spatial variation of the self-shielded cross sections within an absorber body can be obtained. A radial temperature distribution can also be specified, so that space-dependent Doppler broadening can be treated in the transport solution. Within the epithermal PW range, the slowing-down source due to elastic and discrete-level inelastic reactions is computed with the analytical scatter kernel based upon the neutron kinematic relations for *s*-wave scattering. Continuum inelastic scatter is approximated by an analytical evaporation spectrum, assumed isotropic in the laboratory system. For many thermal reactor and criticality safety problems, self-shielding of inelastic cross sections has a minor impact, and by default these options are turned off for faster execution. As previously discussed, the thermal scatter kernel is based on the ENDF/B scattering law data for bound moderators, and uses the free-gas model for other materials.

7.4.2 THEORY AND ANALYTICAL MODELS

This section describes the coupled MG and PW techniques used to solve the neutron transport equation.

7.4.2.1 Energy/lethargy ranges for MG and PW calculations

The combined MG/PW CENTRM calculation is performed over the energy range spanned by the group structure in the input MG library. The energy boundaries for the "IGM" neutron groups specified on the MG library divide the entire energy range into energy intervals. The lowest energy group contained in the UMR is defined to be "MGHI"; while the highest energy group in the LMR is designated "MGLO." The boundary between the PW and UMR energy intervals is set by the energy value "DEMAX," while "DEMIN" is the boundary between the PW and LMR. The default values of 0.001 eV and 20 keV for DEMIN and DEMAX, respectively, can be modified by user input, but the input values are altered by the code to correspond to the closest group boundaries. Hence, DEMAX is always equal to the lower energy boundary of group MGHI and DEMIN the upper energy boundary of MGLO. The PW calculation is performed in terms of lethargy (u), rather than energy (E). The origin (u=0) of the lethargy coordinate corresponds to the energy E=DEMAX, which is the top of the PW range. See Fig. 7.4.2.

The highest energy group of the thermal range is defined by the parameter "IFTG," obtained from the MG library. If DEMIN is less than the upper energy boundary of IFTG, the PW range extends into thermal. In this case, scattering in the PW region of the thermal range is based on the PW scattering kernel data; and the LMR calculation uses 2D transfer matrices for incoherent and coherent scattering on the MG library. Coupling between the MG and PW thermal calculations is treated, and outer iterations are required to address effects of upscattering.

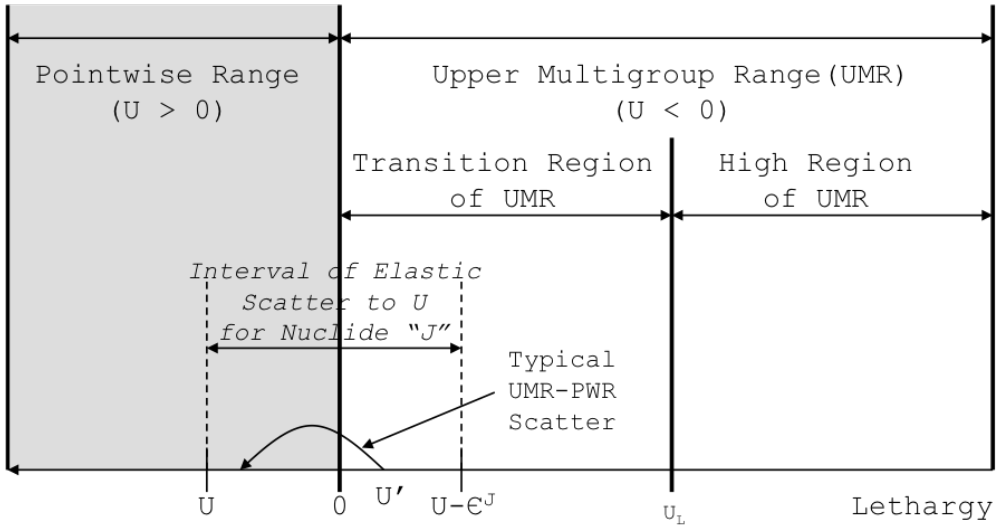


Fig. 7.4.2: Definition of *High* and *Transition* regions in upper multigroup region.

With the exception of hydrogen moderation, elastic down-scattering coupling the UMR and PW ranges, occurs only within a limited sub-range of the UMR called the "transition region". The highest energy group in the transition region is designated "MGTOP." The precise definition of the transition region is given in Sect. 7.4.2.6.1.

Energy boundaries of the group structure on the input MG library correspond to the IGM+1 values, { G_1, G_2, ... G_g, G_{g+1}, ..., G_{IGM+1}}. It is convenient to designate the number of groups in the UMR, PW, and LMR ranges equal to NG_U, NG_P, and NG_L, respectively, so that IGM = NG_U + NG_P + NG_L; or in terms of the parameters MGHI and MGLO introduced previously:

NG_U = MGHI; NG_P = MGLO - MGHI - 1; NG_L = IGM - MGLO + 1.

The flux per unit lethargy is calculated for a discrete energy (or lethargy) mesh spanning the MG structure. Groups in the UMR and LMR each contain a single energy mesh point, while groups in the PW range generally contain several points. The number of mesh *points* in the UMR, PW, and LMR is equal respectively to NG_U, N_P, and NG_L; and the total number of points in the entire energy mesh is designated as "N_T," which is equal to NG_U + N_P + NG_L. Thus the lethargy (u) mesh consists of the set of points: {u:sub:*1*,... u_{NGU}, u_{NGU+1},... u_{NGU+NP}, $u_{NGU+NP+1}$,... u_{NT}}. Based on the lethargy origin at E=DEMAX, the lethargy "u_n" associated with any energy point "E_n" is equal to,

$u_n = \ln(DEMAX/E_n)$.

Lethargy points are arranged in order of increasing value. The lethargy origin is at point NG_U+1, the lower energy boundary of group MGHI; i.e., $\mathbf{u}_{NGU+1}$=0. Note that the entire UMR (E>DEMAX) corresponds to negative lethargy values. Lethargy values for the first NG_U and the last NG_L points in the mesh are defined to be the midpoint lethargies of groups in the UMR and LMR ranges, respectively. For example, for the NG_U groups within the UMR,

$u_1 = 0.5[\ln(DEMAX/G_1) + \ln(DEMAX/G_2)]$;

$u_{NGU} = 0.5[\ln(DEMAX/G_{MGHI}) + \ln(DEMAX/G_{MGHI+1})]$;

and similarly for the NG_L groups in the LMR,

$u_{NGU+NP+1} = 0.5[\ln(DEMAX/G_{MGLO}) + \ln(DEMAX/G_{MGLO+1})]$

$u_{NT} = 0.5[\ln(DEMAX/G_{IGM}) + \ln(DEMAX/G_{IGM+1})]$

The remaining N_P points in the mesh (i.e., values u_{NGU+1} to u_{NGU+NP}) are contained within the NG_P groups that span the PW range. By definition the first point in the PW range is the lower energy boundary of group MGHI. The other mesh points are computed internally by CENTRM, based on the behavior of the macroscopic PW total cross sections and other criteria.

The neutron flux, as a function of space and direction, is calculated for each energy/lethargy point in the mesh by solving the Boltzmann transport equation. The transport equation at each lethargy point generally includes a source term representing the production rate due to elastic and inelastic scatter from other lethargies, which couples the solutions at different lethargy mesh points. Except in the thermal range, neutrons can only gain lethargy (lose energy) in a scattering reaction; thus the PW flux is computed by solving the transport equation at successive mesh points, sweeping from low to high lethargy values.

7.4.2.2 The Boltzmann equation for neutron transport

The steady state neutron transport equation shown below represents a particle balance-per unit phase space, at an arbitrary point :math:` ho` in phase space,

$$\Omega \cdot \nabla\Psi(\rho) + \sum_t (r, u)\Psi(\rho) = \int_0^{\infty} \int_0^{4\pi} \Sigma\left(u' \rightarrow u; \mu_0\right) \Psi\left(u', \Omega'\right) du' + Q_{ext}(\rho) \tag{7.4.1}$$

where:

$\psi(\rho)$ = angular flux (per lethargy) at phase space coordinate ρ;

$\rho = (r, u, \Omega)$ = phase space point defined by the six independent variables;

$r = (x_1, x_2, x_3)$ = space coordinates;

$u = \ln(E_{ref}/E)$ = lethargy at energy E, relative to an origin (u=0) at E_{ref};

$\Omega = (\mu, \zeta)$ = neutron direction defined by polar cosine μ and azimuthal angle ζ;

$\Sigma_t(r, u)$ = macroscopic total cross section;

$\Sigma(u' \rightarrow u; \mu_0)$ = double differential scatter cross section;

μ_0 = cosine of scatter angle, measured in laboratory coordinate system;

$Q_{ext}(\rho)$ = external source term, including fission source;

The left and right sides of Eq. (7.4.1) respectively, are equal to the neutron loss and production rates, per unit volume-direction-lethargy. In CENTRM the spatial distribution of the fission source is input as a component of the external source Q; hence, a fixed source rather than an eigenvalue calculation is required for the transport solution.

The angular dependence of the double-differential macroscopic scatter cross section of an arbitrary nuclide "j" is represented by a finite Legendre expansion of arbitrary order L:

$$\Sigma^{(j)}\left(u' \rightarrow u; \mu_0\right) = \sum_{\ell=0}^{L} \frac{2\ell + 1}{2} \Sigma_{\ell}^{(j)}\left(u' \rightarrow u\right) \quad P_{\ell}\left(\mu_0\right) \tag{7.4.2}$$

where $P_\ell(\mu_0)$ = Legendre polynomial evaluated at the laboratory scattering cosine mu_0; and

$\Sigma^{(j)}(u' \rightarrow u)$ = cross section moments of nuclide j, defined by the expression

$$\sum_{\ell}^{(\mathrm{j})}(\mathrm{u}' \rightarrow \mathrm{u}) = \int_{-1}^{1} \sum^{(\mathrm{j})}(\mathrm{u}' \rightarrow \mathrm{u}; \mu_0)\, \mathrm{P}_{\ell}(\mu_0)\, \mathrm{d}\mu_0 \tag{7.4.3}$$

After substitution of the above Legendre expansions for the scattering data of each nuclide, and applying the spherical harmonic addition theorem in the usual manner, the scattering source on the right side of Eq. (7.4.1) becomes [CENTRM-BG70]:

$$\mathrm{S}(\mathrm{r}, \mathrm{u}, \Omega) \equiv \int_0^{\infty} \int_0^{4\pi} \Sigma(\mathrm{u}' \rightarrow \mathrm{u}; \mu_0)\, \Psi(\mathrm{u}', \Omega')\, \mathrm{d}\Omega' \mathrm{d}u' = \sum_{\ell \mathrm{k}=1}^{\mathrm{LK}} \frac{2\ell + 1}{2} \mathrm{Y}_{\ell \mathrm{k}}(\Omega) \mathrm{S}_{\ell \mathrm{k}}(\mathrm{r}, \mathrm{u}) \tag{7.4.4}$$

wherein,

$\mathrm{Y}_{\ell \mathrm{k}}(\Omega) = \mathrm{Y}_{\ell \mathrm{k}}(\mu, \zeta)$ = the spherical harmonic function evaluated at direction Ω

$\mathrm{S}_{\ell \mathrm{k}}$ = spherical harmonic moments of the scatter source, per unit letharagy.

The summation index "ℓk" indicates a double sum over ℓ and k indices; in the most general case it is defined as:

$$\sum_{\ell \mathrm{k}=1}^{\mathrm{LK}} = \sum_{\ell=0}^{\mathrm{L}} \sum_{\mathrm{k}=0}^{\ell} \tag{7.4.5}$$

where "L" is the input value for the maximum order of scatter (input parameter "ISCT").

Due to symmetry conditions, some of the source moments may be zero. The parameter LK is defined to be the total number of non-zero moments (including scalar flux) for the particular geometry of interest, and is equal to,

LK = L + 1 for 1D slabs and spheres;

LK = L*(L+4)/4+1 for 1D cylinders, and

LK = L*(L+3)/2+1 for 2D MoC cells

More details concerning the 1-D Boltzmann equation can be found in the XSDRNPM chapter of the SCALE manual.

7.4.2.3 Legendre moments of the scattering source

The $S_{\ell k}$ moments in Eq. (7.4.4) correspond to expansion coefficients in a spherical harmonic expansion of the scatter source. These can be expressed in terms of the cross section and flux moments by

$$\mathrm{S}_{\ell \mathrm{k}}(\mathrm{u}) = \sum_{\mathrm{j}} \int_{\mathrm{u}'} \mathrm{S}_{\ell \mathrm{k}}^{(\mathrm{j})}(\mathrm{u}' \rightarrow \mathrm{u})\, \mathrm{d}u' = \sum_{\mathrm{j}} \int_{\mathrm{u}'} \Sigma_{\ell}^{(\mathrm{j})}(\mathrm{u}' \rightarrow \mathrm{u})\, \Psi_{\ell \mathrm{k}}(\mathrm{u}')\, \mathrm{d}u' \tag{7.4.6}$$

where $\psi_{\ell k} = (u)$ = spherical harmonic moments of the angular flux;

$$= \int_0^{4\pi} Y_{\ell \mathrm{k}}(\Omega) \quad \Psi(\Omega) \mathrm{d}\Omega \tag{7.4.7}$$

and $S_{\ell k}^{(j)}(u' \rightarrow u)$ = moments of the differential scatter rate from lethargy u′ to u, for nuclide "j";

$$= \Sigma_{\ell}^{(j)}(u' \rightarrow u)\, \Psi_{\ell k}(u') \tag{7.4.8}$$

The $\psi_{\ell k}$ flux moments are the well known coefficients appearing in a spherical harmonic expansion of the angular flux. These usually are the desired output from the transport calculation. In particular, the $\ell = 0, k = 0$ moment corresponds to the scalar flux [indicated here as $\phi\,(r, u)$],

$$\Psi_{0,0}(\mathrm{r,\,u}) = \Phi(\mathrm{r,\,u}) = \int_0^{4\pi} \Psi(\mathrm{r,\,u},\Omega) d\Omega \tag{7.4.9}$$

In general the epithermal component of the scatter source in Eq. (7.4.4) contains contributions from both elastic and inelastic scatter reactions; however, inelastic scatter is only possible above the threshold energy corresponding to the lowest inelastic level. The inelastic Q values for most materials are typically above 40 keV; therefore, elastic scatter is most important for slowing down calculations in the resolved resonance range of most absorber materials of interest. For example, the inelastic Q values of ^{238}U, iron, and oxygen are approximately 45 keV, 846 keV, and 6 MeV, respectively; while the upper energy of the ^{238}U resolved resonance range is 20 keV in ENDF/B-VII. The inelastic thresholds of some fissile materials like ^{235}U and ^{239}Pu are on the order of 10 keV; however, with the exception of highly enriched fast systems, these inelastic reactions usually contribute a negligible amount to the overall scattering source. CENTRM assumes that continuum inelastic scatter is isotropic in the laboratory system, while discrete level inelastic scatter is isotropic in the *center of mass* (CM) coordinate system.

Over a broad energy range, *elastic* scatter from most moderators can usually be assumed isotropic (*s*-wave) in the neutron-nucleus CM coordinate system. In the case of hydrogen, this is true up to approximately 13 MeV; for carbon up to 2 MeV; and for oxygen up to 100 keV. However, it is well known that isotropic CM scatter does not result in isotropic scattering in the laboratory system. For *s*-wave elastic scatter the average scatter-cosine in the laboratory system is given by: $\bar{\mu}_0 = 0.667/\ \mathrm{A}$;[3] where "A" is the mass number (in neutron mass units) of the scattering material. This relation indicates that *s*-wave, elastic scattering from low A materials tends to be more anisotropic in the laboratory, and that the laboratory scattering distribution approaches isotropic ($\bar{\mu}_0 = 0; \theta_0 = 90$) as A becomes large. For example, the $\bar{\mu}_0$ of hydrogen is 0.667 (48.2°); while it is about 0.042 (87.6°) for oxygen. Because *s*-wave scattering from heavy materials is nearly isotropic in the laboratory system, the differential scattering cross section (and thus the scattering source) can usually be expressed accurately by a low order Legendre expansion. On the other hand light moderators like hydrogen may require more terms-depending on the flux anisotropy-to accurately represent the elastic scatter source in the laboratory system. The default settings in CENTRM are to use P0 (isotropic lab scatter) for mass numbers greater than A=100, and P1 for lighter masses.

An analytical expression can be derived for the cross-section moments in the case of two-body reactions, such as elastic and discrete-level inelastic scattering from "stationary" nuclei. Stationary here implies that the effect of nuclear motion on neutron scattering kinematics is neglected.

Note: The stationary nucleus approximation for treating scattering kinematics does not imply that the effect of nuclear motion on Doppler broadening of resonance cross sections is ignored, since this effect is included in the PW cross-section data.

In CENTRM the stationary nucleus approximation is applied above the thermal cutoff, typically around 3-5 eV, but is not valid for low energy neutrons. CENTRM has the capability to perform a PW transport calculation in the thermal energy range using tabulated thermal scattering law data for bound molecules, combined with the analytical free-gas kernel for other materials. In this case the cross-section moments appearing in Eq. (7.4.3) include upscattering effects. The expressions used in CENTRM to compute the PW scatter source moments in the thermal range are given in Sect. 7.4.2.6.

The following two sections discuss the evaluation of the scatter source moments for epithermal elastic and inelastic reactions, respectively.

Epithermal Elastic Scatter

Consider a neutron with energy E′, traveling in a direction Ω′, that scatters elastically from an arbitrary material "j," having a mass $A^{(j)}$ in neutron-mass units. Conservation of kinetic energy and momentum requires that there be a unique relation between the angle that the neutron scatters (relative to the initial direction) and its final energy E after the collision. If the nucleus is assumed to be stationary in the laboratory coordinate system, then the cosine (μ_0) of the scatter angle (θ_0) measured in the laboratory system, as a function of the initial and final energies, is found to be

$$\mu_0 \equiv \Omega' \cdot \Omega = G^{(j)}(E', E), \tag{7.4.10}$$

where the kinematic correlation function G relating E′, E, and μ_0 for elastic scatter is equal to

$$\begin{gathered} G^{(j)}(E', E) = a_1^{(j)} \times [E/E']^{\frac{1}{2}} - a_2^{(j)} \times [E'/E]^{\frac{1}{2}} \\ \text{and } a_1^{(j)} = \left(A^{(j)} + 1\right)/2 \quad ; \quad a_2^{(j)} = \left(A^{(j)} - 1\right)/2 \end{gathered} . \tag{7.4.11}$$

The final energy E of an elastically scattered neutron is restricted to the range,

$$\alpha^{(j)} E' \leq E \leq E' \tag{7.4.12}$$

where $\alpha^{(j)} =$ maximum fractional energy lost by elastic scatter

$$= \left[a_2^{(j)}/a_1^{(j)}\right]^2 \tag{7.4.13}$$

The corresponding range of scattered neutrons in terms of lethargy is equal to

$$u' \leq u \leq u' + \varepsilon^{(j)} \tag{7.4.14}$$

where

$$\begin{aligned} u, u' = u(E), u'(E') &= \text{ lethargies corresponding to E and E}', \text{ respectively; and} \\ \varepsilon^{(j)} &= \text{ maximum increase in lethargy, per elastic scatter } = \ln\left[1/\alpha^{(j)}\right] \end{aligned} \tag{7.4.15}$$

The double-differential scatter kernel of nuclide j (per unit lethargy and solid angle) for *s*-wave elastic scatter of neutrons from stationary nuclei, is equal to

$$\begin{aligned} \Sigma^{(j)}(u' \to u; \mu_0) &= \frac{E\Sigma^{(i)}(u')}{E'\left(1 - \alpha^{(j)}\right)} \delta\left[\mu_0 - G^{(j)}(E', E)\right], \text{ for } u' \leq u \leq u' + \varepsilon^{(j)} \\ &= 0 \quad u < u' \text{ or } u > u' + \varepsilon^{(j)} \end{aligned} \tag{7.4.16}$$

The presence of the Dirac delta function completely correlates the angle of scatter and the values of the initial and final energies. Substituting the double differential cross-section expression from Eq. (7.4.16) into Eq. (7.4.3) gives the single-differential Legendre moments of the cross section, per final lethargy:

$$\begin{aligned} \Sigma_\ell^{(j)}(u' \to u) &= \frac{EP_\ell\left[G^{(j)}\right]\Sigma^{(j)}(u')}{E'\left(1 - \alpha^{(j)}\right)}, \text{ for } u' \leq u \leq u' + \varepsilon^{(j)} \\ &= 0 \quad u' \text{ or } u > u' + \varepsilon^{(j)} \end{aligned} \tag{7.4.17}$$

where P_ℓ = Legendre polynomial evaluated at argument $G^{(j)}$ equal to the scatter cosine.

When the above expressions are used in Eq. (7.4.6), the following is obtained for the ℓk moment of the epithermal elastic scattering source at lethargy u:

$$S_{\ell k}{}^{(j)}(u) = \sum_j \int_{u-\varepsilon^{(j)}}^{u} S_{\ell k}(u' \to u)\, du' = \sum_j \int_{u-\varepsilon^{(j)}}^{u} \frac{E\Sigma^{(j)}(u') P_\ell\left[G^{(j)}\right]}{E'\left(1 - \alpha^{(j)}\right)} \Psi_{\ell k}(u')\, du'. \tag{7.4.18}$$

Epithermal Inelastic Scatter

If the input value of DEMAX is set above the inelastic threshold of some materials in the problem, then inelastic scattering can occur in the PW range. The pointwise transport calculation may optionally include discrete-level and continuum inelastic reactions in computing the PW scatter source moments. The multigroup calculations always consider inelastic reactions.

Discrete-level inelastic reactions are two-body interactions, so that kinematic relations can be derived relating the initial and final energies and the angle of scatter. It can be shown that the kinematic correlation function for discrete-level inelastic scatter can be written in a form identical to that for elastic scatter by redefining the parameter a_1 in Eq. (7.4.11) to be the energy dependent function [CENTRM-Wil00],

$$a_1^{(\,m,j)} = \frac{\left(A^{(j)} + 1\right)}{2} + \frac{\left(-Q^{(m,j)}\right) A^{(j)}}{2E} \tag{7.4.19}$$

The parameter $Q^{(m,\,j)}$ is the Q-value for the m_{th} level of nuclide "j". The Q value is negative for inelastic scattering, while it is zero for elastic scatter. The threshold energy in the laboratory coordinate system is proportional to the Q-value of the inelastic level, and is given by:

$$E_T^{(m,j)} = \frac{A^{(j)} + 1}{A^{(j)}} \times \left(-Q^{(m,j)}\right) \tag{7.4.20}$$

The range of energies that can contribute to the scatter source at E, due to inelastic scatter from the m_{th} level of nuclide j is defined to be [E:sub:*L* , E:sub:*H*], where E_H >E:sub:*L* >E:sub:*T*. This energy range has a corresponding lethargy range of [u:sub:*LO* , u:sub:*HI*] which is equal to,

$$\begin{aligned} u_{LO} &= u - \ln\left(\frac{1}{\alpha_1^{(j)}(E_H)}\right) = u - \varepsilon_1^{(j)} \\ u_{HI} &= u - \ln\left(\frac{1}{\alpha_2^{(j)}(E_L)}\right) = u - \varepsilon_2^{(j)} \end{aligned} \tag{7.4.21}$$

The energy-dependent alpha parameters in the above expressions are defined as,

$$\begin{aligned} \alpha_1(E) &= \left(\frac{A\Delta^{(m,j)}(E)-1}{A+1}\right)^2 \\ \alpha_2(E) &= \left(\frac{A\Delta^{(m,j)}(E)+1}{A+1}\right)^2 \end{aligned} \tag{7.4.22}$$

where

$$\Delta^{(m,j)}(E) = \sqrt{1 - \frac{E_T^{(m,j)}}{E}} \tag{7.4.23}$$

Modifying the epithermal elastic scatter source in Eq. (7.4.18) to include discrete-level inelastic scatter gives the following expression

$$S_{\ell k}(u) = \sum_{m,j} \int_{u_{LO}}^{u_{HI}} \frac{E}{E'} \frac{\Sigma^{(m,j)}(E')\, P_\ell\left[G^{(m,j)}\right]}{(1-\alpha^j)\,\Delta^{(m,j)}(E')} \Psi_{\ell k}(u')\, du' \tag{7.4.24}$$

Detailed expressions for the lethargy limits are given in [CENTRM-Wil00]. Since $\Delta^{(m,j)}$ is equal to unity for elastic scatter, the above equation reduces to Eq. (7.4.15) if there is no discrete-level inelastic contribution.

At high energies, the inelastic levels of the nucleus become a continuum. In this case CENTRM represents the energy distribution of the scattered neutrons by an evaporation spectrum with an isotropic angular distribution in the lab system; thus, only the P_0 moment appears in the continuum inelastic scattering source. Including continuum inelastic reactions in the PW calculation usually has a small impact on the spectrum used for resonance self-shielding, and may adversely impact the computer memory requirements and execution time. Therefore, by default, CENTRM does not include continuum inelastic reactions in the pointwise solution; however, it is always included in the UMR solution.

Thermal Scatter

Since thermal neutrons have energies comparable to the mean kinetic energy of molecules in thermal equilibrium, the scattering kernels must account for molecular motion. The scatter moments include both downscatter as well as upscatter contributions; hence, the integration limits appearing in Eq. (7.4.18) must be extended from the lowest to the highest energy of the thermal range. Furthermore the cross-section moments correspond to the Legendre expansion coefficients of the thermal scatter kernel, which has a substantially different form than the epithermal kernel discussed in the previous two sections. In general the ℓ^{th} Legendre moment of the thermal scattering kernel at temperature T, describing scattering from E to E', is given by

$$\sigma_\ell(\mathrm{E}' \rightarrow \mathrm{E}; \mathrm{T}) = \frac{\sigma_b}{T}\sqrt{\frac{\mathrm{E}}{\mathrm{E}'}}e^{-\frac{\beta(\mathrm{E}'\rightarrow\mathrm{E})}{2}}\int \mathrm{P}_\ell(\mu_0)\,\mathrm{S}[\alpha,\beta;T]d\mu_0 \tag{7.4.25}$$

where $\beta(E' \rightarrow E)$ and $\alpha(E', E, \mu_0)$ are dimensionless variables (functions of temperature) defining the energy and momentum exchange, respectively, of the collision [CENTRM-BG70]; σ_b is the rigidly bound scatter cross section, which is proportional to the free atom cross section; and S(α, β; T) describes the temperature-dependent thermal scattering law.

If atomic bonding effects are neglected, the atoms of a material behave like a gas in thermal equilibrium at the temperature of the medium. In this case S(α, β) can be expressed by an analytical function. CENTRM uses the free gas model for all nuclides except those materials that have thermal scattering laws available in the ENDF/B nuclear data files. The ENDF/B scattering law data account for the effects of molecular bonding and possible polyatomic crystalline structure. While free-gas kernels are computed internally in CENTRM, the kernel moments describing bound thermal scatterers are stored in a data file that can be accessed by CENTRM.

Bound thermal kernels

Thermal scattering from bound atoms is classified either as an "inelastic reaction," in which the neutron energy may change, or an "elastic reaction," in which the neutron changes direction, but does not change energy. In ENDF/B the former reactions are treated as incoherent inelastic scattering with a doubly differential kernel describing the secondary neutron energy and angle distribution. The latter reactions are usual treated as coherent elastic scatter characterized by diffractive interference of the scattered deBroglie waves, although a few materials are modeled by the incoherent elastic approximation. Legendre moments for thermal elastic kernels describe the secondary angular distribution with no energy exchange, at a given neutron energy. Bound scatter kernels have been processed by the AMPX code system for most of the ~25 compounds with thermal scatter laws in ENDF/B, and are stored in individual kinematics files distributed with the SCALE code system. These include materials such as: H in water, H and C in polyethylene, H and Zr in ZrH, C in graphite, deuterium in heavy water, Be metal, Be in BeO, etc. The CRAWDAD module processes scattering kernel data for individual nuclides into a combined library used in CENTRM, and also interpolates the kernels to the appropriate temperatures.

The bound scatter kernels are tabulated at different energy points from the flux solution mesh; therefore it is necessary to map the data onto the desired energy mesh in the CENTRM calculation. Because thermal

elastic scattering results in no energy loss, the elastic moments only appear in the within-point term of the scattering source in the CENTRM thermal calculation. Thus the coherent elastic data is easily interpolated since it only involves a single energy index and temperature. However, the incoherent inelastic moments are 2-D arrays in terms of the initial and final energies, so that a 2-D interpolation must be done for each temperature. CENTRM uses a simple type of "unit-base transform" method to interpolate incoherent inelastic kernels onto the flux solution mesh. The method attempts to preserve the absolute peak of the secondary energy distribution, at given initial energy. For water-bound hydrogen and several other moderators, this is quite adequate, since the kernel generally has only a single maximum. However, if more than one local extrema is present, such as for graphite, the other local peaks are not explicitly preserved in the interpolation method. For this reason it is necessary to include a fairly dense set of initial energies in the tabulated kernels of graphite and similar materials, to avoid gross changes in the kernel shape at adjacent initial energy panels.

Free gas thermal kernels

CENTRM computes free-gas kernels using the approach proposed by Robinson [CENTRM-Rob81] as a modification to the original FLANGE [CENTRM-HF71] methodology. Legendre moments of the free-gas scatter kernel per unit lethargy are expressed as,

$$\Sigma_{\ell}^{(j)}\left(u' \rightarrow u\right) = A^{(j)}\Sigma_{free}^{(j)} \quad \frac{E}{E'}e^{-\beta/2}\sum_{n=0}^{\ell} W_{\ell n}H_n\left(E', E\right) \tag{7.4.26}$$

where $W_{\ell n}$ are constant coefficients associated with the Legendre polynomial of order ℓ; Σ_{free} is the constant free-atom cross section for the material; and H_n are the α-moments of the free-gas scatter law, given as

$$H_n\left(E', E\right) = \frac{1}{\sqrt{\pi}}\int_{\alpha_L}^{\alpha_H} \alpha^n \times \left(\frac{e^{-\frac{\alpha^2+\beta^2}{4\alpha}}}{2\sqrt{\alpha}}\right) d\alpha \tag{7.4.27}$$

The limits on the above integral correspond to:

$$\alpha_L\left(E', E\right) = \alpha\left(E', E, \mu_0 = -1\right) \quad \text{and} \quad \alpha_H\left(E', E\right) = \alpha\left(E', E, \mu_0 = 1\right). \tag{7.4.28}$$

The alpha moments for $n > 0$ can be evaluated very efficiently using a recursive relation [CENTRM-Wil00]:

$$H_n\left(E', E\right) = 2(2n-1)H_{n-1} + \beta^2 H_{n-2} - \left[F_n\left(\sqrt{\alpha_H}, \beta\right) - F_n\left(\sqrt{\alpha_L}, \beta\right)\right] \tag{7.4.29}$$

where F_n is the function,

$$F_n(t, \beta) = \frac{t^{2n-1}e^{-\frac{1}{4}\left(\frac{\beta^2}{t^2}+t^2\right)}}{\sqrt{\pi}/2} \tag{7.4.30}$$

Analytical expressions for the initial two moments, H_0 and H:sub:` -1`, are given in [CENTRM-Rob81].

The standard free-gas kernel is based on the assumption of a constant free atom cross section. When averaged over the molecular velocity distribution, this gives a 1/v variation in the effective free-gas cross section at low energies. To approximately account for nuclear structure effects on the energy dependence of the thermal cross section (e.g., low energy resonances), the free-gas moments are multiplied by the ratio $\sigma_s(E)/\sigma_{FG}(E)$, where σ_s is the Doppler broadened scatter cross section processed from ENDF/B data; and σ_{FG} is the effective free-gas cross section,

$$\sigma_{FG}\left(E'\right) = \frac{\sigma_{free}}{y^2}\left[\left(y^2 + 1/2\right)\mathrm{erf}(y) + \frac{ye^{-y^2}}{\sqrt{\pi}}\right] \tag{7.4.31}$$

where $y^2 = A\frac{E}{kT}$.

7.4.2.4 Sub-moment expansion of the epithermal scattering source

One difficulty in computing the epithermal scatter source moments is that the Legendre polynomial in the integrand of Eq. (7.4.18) and Eq. (7.4.24) is a function of both the initial and final lethargy (or energy) of the scattered neutrons, due to the correlation function $G^{(j)}(E, E')$). At each lethargy u this requires that the u'-integral be recomputed over all lower lethargies, for every nuclide and moment. A more efficient algorithm would be possible if the differential scattering moments appearing in the integrand could be factored into a product of a function of u multiplied by a function of u' such as

$$S_{\ell k}{}^{(j)}(u' \to u) = F^{(j)}(u') \quad H^{(j)}(u) \tag{7.4.32}$$

where F and H are the two factors (to be specified later).

If this is done, the u-function can be factored from the scatter source integrals, leaving only integrals over the u$\prime$-function as shown below:

$$S^{(j)}(u) = \int_{u'} S^{(j)}_{\ell k}(u' \to u)\, du' = H^{(j)}(u) \int_{u'} F^{(j)}(u')\, du' \tag{7.4.33}$$

Because the factored integrand does not depend on the variable u, a running summation over all u' points can be accumulated and saved as the calculation sweeps from low to high lethargy. For example, note that the $\ell = 0$ moment in Eq. (7.4.18) is already separable into a product of u times u' because P_0 is equal to one at all values of G. Thus the isotropic component of the elastic differential scatter rate (per unit lethargy) from u' to u is proportional to *E/E^{prime}*, where

$$E = E(u) = E_{ref} e^{-u}, \quad \text{and} \quad E' = E'(u') = E_{ref} e^{-u'} \tag{7.4.34}$$

Therefore, the two separable factors in the lowest moment, $S^{(j)}_{0.0}(u' \to u)$ are identified as,

$$\begin{aligned} H(u) &= E/\left(1 - \alpha^{(j)}\right), \quad \text{and} \\ F(u') &= \Sigma^{(j)}(u')\, \Psi_{00}(u')\, /E' \end{aligned} \tag{7.4.35}$$

However, the higher order Legendre moments contain the term $P_\ell(G)$ in the integrand; and the expression for $G(E', E)$ is a difference of two terms that depend on both E and E'. A new method called a "sub-moment expansion" has been developed for CENTRM that allows the Legendre polynomials appearing in the differential scatter moments to be factored into the desired separable form. Each spherical harmonic moment of the scattering source appears expanded in a series of factored "sub-moments."

The Legendre polynomial of order ℓ is a polynomial containing terms up to the *ell*$^{\text{th}}$ power. Applying the binomial expansion theorem and some algebraic manipulation, the standard expression for P_ℓ evaluated at "G" can be expressed as

$$P_\ell(G) = \frac{E'}{E} \times a_1^\ell \sum_{K=-\ell}^{\ell} \tilde{g}_{\ell,K}(E) \quad h_K(E) h_K^{-1}(E') \tag{7.4.36}$$

where $h_k(E) = E^{1+K/2}$; and the expansion coefficients $\tilde{g}_{\ell,k}$ are equal to, $\tilde{g}_{\ell K} = \frac{g_{\ell K}}{N_\ell \times \alpha_1^\ell}$ where the $g_{\ell,K}$ (no tilde) parameters were defined in [CENTRM-WA95] to be:

$$\begin{aligned} g_{\ell,K} &= \frac{\left(1 + (-1)^{\ell+K}\right)}{2} \sum_{K'=0}^{\frac{\ell-K}{2}} (-1)^{K'} b_{2K'+K,\ell} \begin{pmatrix} 2K'+K \\ K' \end{pmatrix} \quad a_1^{K+K'} a_2^{K'}; \quad \text{for} \quad K \geq 0 \\ &\text{and} \\ &= (-a_2/a_1)^{|K|} \quad g_{\ell,|K|} \quad ; \text{ for } K < 0. \end{aligned} \tag{7.4.37}$$

In Eq. (7.4.36)-Eq. (7.4.37), the constants $b_{m,\ell}$ and N_ℓ are the standard Legendre constants and normalization factors, respectively, which are tabulated in Table 7.4.1 for orders through P_7; and $\begin{pmatrix} m \\ i \end{pmatrix}$ = the binomial expansion coefficients[20] = $\frac{m!}{(m-i)!\quad i!}$

continues on next page

Table 7.4.1 – continued from previous page

Table 7.4.1: Constants appearing in Legendre polynomials.

Subroutine Name	Calling Subroutine	Called Subroutine
add2d	xscal	pint
addpxs	xscal	
boss	pmc	flxrec process replace set zonefl
capy	replace	cellav2 factor normal repl2d
cellav2	capy	
cellxs	process	
dilutxs	xscal	
factor	capy	iset
flxrec	boss	
normal	capy	iset
pgen	pint	
pint	add2d	pgen pinte pinto
pinte	pint	
pinto	pint	
pmc		boss setup
print	process	prt1d
process	boss	cellxs print sumtot xscal
reorder	xscal	
repl2d	capy	
replace	boss	capy
setup	pmc	mesage
xscal	process	add2d addpxs dilutxs reorder
zonefl	boss	

continues on next page

The explicit dependence of the constants a_1 and a_2 on the nuclide index j [see Eq. (7.4.11)] has been suppressed to simplify notation. For discrete-level inelastic scatter the parameter a_1 is an energy dependent function given by Eq. (7.4.19), but for elastic scatter this reduces to the constant in Eq. (7.4.11). Note that the $g_{\ell,K}$ value is zero unless ℓ and K are both even or both odd, respectively, so that about half the terms appearing in the summation of Eq. (7.4.36) vanish. Table 7.4.2 through Table 7.4.4 give values for the submoment expansion coefficients for several nuclides.

The sub-moment expansion of the scattering source, including both elastic and discrete-level inelastic reactions, is obtained by substituting the expansion of the Legendre polynomial from Eq. (7.4.36) into Eq. (7.4.26), giving

$$S_{\ell k}(u) = \sum_{m,j} \sum_{K=-\ell}^{\ell} Z_{\ell,K}^{(m,j)}(E) h_K(E) \int_{u_{LO}^{(m,j)}}^{u_{HI}^{(m,j)}} \psi_{\ell k}(u') \Sigma^{(m,j)}(E') \frac{h_K^{-1}(E')}{\Delta^{(m,j)}(E')} du' \tag{7.4.38}$$

where $Z_{\ell K}^{(m,j)}(E) = a_1^{\ell}(E) \frac{\tilde{g}_{\ell,K}^{(m,j)}(E)}{(1-\alpha^{(j)})}$. For elastic scatter, the Z coefficients are independent of energy.

With this approach the scatter source moments in Eq. (7.4.39) have been further expanded into a summation of "submoments" identified by index K (although some of these terms are equal to zero, due to the behavior of the $g_{\ell,K}$ coefficients). Each term has the desired factored form expressed in Eq. (7.4.32); i.e., separable in terms of the variables u and u' with

$$H_{\ell,K}^{(i)}(u) = Z_{\ell,K}^{(j)}(E) h_K(E), \quad \text{and} \quad F_{\ell k,K}^{(j)}\left(u^j\right) = \frac{h_K^{-1}(E') \Psi_{\ell k}(u') \Sigma^{(j)}(u')}{\Delta^{(m,j)}(E')} \tag{7.4.39}$$

so that the lk_{th} moment of the scatter source can be written as

$$S_{\ell k}(u) = \sum_{m,j} \sum_{K=-1}^{\ell} H_{\ell,K}^{(j)}(u) \int_{u_{LO}^{(m,j)}}^{u_{HI}^{(m,j)}} F_{\ell k,K}^{(j)}(u') du' \tag{7.4.40}$$

Characteristics and Properties of the Sub-Moment Expansion

The expansion in Eq. (7.4.36) becomes numerically unstable for heavy nuclides (large A), with high Legendre orders. Using double precision arithmetic, it was found that the accuracy of the expansion begins to break down for heavy nuclides (A100) if the order of scatter exceeds P_5; although the expansion for lighter nuclides (viz, moderators) is very accurate even for scattering orders as high as P_7 or more. For this reason CENTRM has an option to restrict the Legendre expansion to lower orders for heavy masses, while using the input value of "ISCT" for lighter nuclides. The restricted Legendre order and mass cut-off value can be controlled by user input, but the default is P_0 (i.e., isotropic lab scattering) for A>100. Table 7.4.5 shows the maximum error observed in the series representation of Legendre polynomials up to P_5, for selected mass numbers. These values were obtained by evaluating the series expansion for $P_\ell(G(x))$ in Eq. (7.4.36), and comparing to the exact value computed at 11 equally spaced values for E/E'. The observed error in the P_5 polynomial expansion is < 1% even for heavy materials such as ^{238}U, while nuclides whose mass is < 100 are computed nearly exactly by the expansion.

Although the accuracy of the submoment expansion is good through P_7 scattering in moderators, Legendre expansions above P3 are not recommended because the number of terms in the expansion increases rapidly with increasing scattering order, especially for 2D MoC and 1D cylindrical systems. The number of spherical

harmonic moments appearing in the scattering source depends on the order (L=ISCAT) of the Legendre expansion used to represent the differential scatter cross section, as well as on the type of geometry (slab, spherical, cylindrical, or 2D MoC) used in the transport calculation. The submoment method further expands each source moment. Table 7.4.6 shows the number of moments in the cross-section expansion, and the number of moments and submoments in the scatter source expansion, as a function of scatter order and geometry type. Although the use of cumulative integrals discussed below allows the sub-moments to be evaluated rapidly, the large number of terms becomes prohibative for high scattering orders. Fortunately a P_1 Legendre order is sufficient for most self-shielding calculations, and orders beyond P_2 should seldom be required for reactor physics and criticality applications.

Table 7.4.2: Coefficients in expansion of $P_\ell\left[G\left(x\right)\right]*$ for hydrogen (A = 1).

$\tilde{g}_{1,K}$, K = −1, . . . , 1							
Legendre order (ℓ) K:	-3	-2	-1	0	1	2	3
0				1			
1			0	0	1		
2		0	0	-0.5	0	1.5	
3		0	0	0	-1.5	0	2.5
*For A=1, G(x) = x; $\left\lfloor x = (E/E')^{1/2}\right.$; where E',E=initial and final energies.							

Table 7.4.3: Coefficients in expansion of $P_\ell\left[G\left(x\right)\right]*$ for oxygen (A = 16).

$\tilde{g}_{l,K}$, K = −1, . . . , 1								
Legendre order (ℓ)	K	-3	-2	-1	0	1	2	3
0					1			
1				-0.88235294	0	1		
2			1.16782007	0	-2.6539724	0	1.5	
3		-1.7173824	0	5.85741909	0	-6.6384083	0	2.5
*For A=16, G(x) = $\frac{8.5}{x} - 7.5x$: $\left\lfloor x = (E/E')^{1/2}\right.$; where E',E=initial and final energies.								

Table 7.4.4: Coefficients in expansion of $P_\ell\left[G\left(x\right)\right]*$ for U-238 (A = 236).

$\tilde{g}_{1,K}$, K = −1, . . . , 1								
Legendre order (ℓ)	K	-3	-2	-1	0	1	2	3
0					1			
1				-0.99156118	0	1		
2			1.47479036	0	-2.97471915	0	1.5	
3		-2.43724146	0	7.37405774	0	-7.43681568	0	2.5

continues on next page

Table 7.4.4 – continued from previous page

$\tilde{g}_{1,K}$, K = −1, . . . , 1
*For A=236, $G(x) = \frac{118.5}{x} - 117.5x$: $\lfloor x = (E/E')^{1/2}$; where E',E=initial and final energies.

Table 7.4.5: Fractional error (1) in series expansion of Legendre polynomials.

Legendre order (*ell*)	Mass number in neutron mass units (A)					
	1	12	56	100	150	236
1	0(*)	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	1.2×10^{-5}
5	0	0	0	2.1×10^{-5}	3.9×10^{-3}	8.4×10^{-3}

(1) Fractional error = | 1.0 − (approximate value/exact value) |
(*) 0 error indicates $< 10^{-6}$.

Table 7.4.6: Number of moments and submoments as function of scattering order.

Legendre order (ℓ) of scattering	Geometry type	Number of scattering x-section moments	Number of scattering source moments	Number of scattering source submoments
0	sphere, slab	1	1	1
	cylinder	1	1	1
	2D MoC	1	1	1
1	sphere, slab	2	2	3
	cylinder	2	2	3
	2D MoC	2	3	5
2	sphere, slab	3	3	6
	1D cylinder	3	4	9
	2D MoC	3	6	14
3	sphere, slab	4	4	10
	1D cylinder	4	6	17
	2D MoC	4	10	30
4	sphere, slab	5	5	15
	1D cylinder	5	9	36
	2D MoC	5	15	55
5	sphere, slab	6	6	21
	1D cylinder	6	12	50
	2D MoC	6	21	91

Scattering moments expressed with cumulative integral operator

It will be convenient to express the scatter source moments in terms of an integral operator $\mathbb{C}$, designated here as the "cumulative integral." The domain of this operator is the vector space of all integrable lethargy functions. The operator is defined for an arbitrary domain element $f(u')$, at an arbitrary lethargy limit U, to be:

$$\mathbb{C}(\mathrm{f};\mathrm{U}) = \int_{\mathrm{u_0}}^{\mathrm{U}} \mathrm{f}(\mathrm{u}')\,\mathrm{du}' \tag{7.4.41}$$

where u_0 is an arbitrary reference point. In implementing this method in CENTRM, it is convenient to set u_0=u_L; i.e., the negative lethargy value corresponding to highest energy of the transition range.

The cumulative integral at some lethargy mesh point u_n is related to the value at the previous lethargy mesh point u_{n-1} by the expression

$$£(f;u_n) = £(f;u_{n-1}) + \int_{u_{n-1}}^{u_n} f(u')\,du' \tag{7.4.42}$$

where $u_n > u_{n-1}$.

Note that only *a single panel of integration* over the interval $[u_{n-1}, u_n]$ must be performed to update the cumulative integrals.

The sub-moment expansion of the scatter source in Eq. (7.4.38) can be expressed in terms of the cumulative integral operator as follows:

$$\mathrm{S}^{(\mathrm{j})}_{\ell\mathrm{k}}(\mathrm{u}) = \sum_j \sum_{K=-\ell}^{\ell} \mathrm{H}^{(\mathrm{j})}_{\ell,\mathrm{K}}(\mathrm{u}) \times \left[£\left(\mathrm{F}^{(\mathrm{j})}_{\ell\mathrm{k},\mathrm{K}}; u^{(m,j)}_{HI} \right) - £\left(\mathrm{F}^{(\mathrm{j})}_{\ell\mathrm{k},\mathrm{K}}; u^{(m,j)}_{LO} \right) \right] \tag{7.4.43}$$

For elastic scatter the value of $\mathrm{u}^{(\mathrm{m,j})}_{\mathrm{LO}}$ is equal to $u - \varepsilon^{(j)}$, and $\mathrm{u}^{(\mathrm{m,j})}_{\mathrm{HI}}$ is equal to u.

7.4.2.5 Multigroup Boltzmann equation

The MG form of the transport equation used in the UMR and LMR is derived by integrating Eq. (7.4.1) over the energy intervals defined by the group structure in the MG library. Details concerning the MG transport equation, including its solution using the discrete ordinates method, can be found in the SCALE documentation of XSDRNPM. The CENTRM MG solution is similar to the XSDRNPM method: however; the outer iteration loop in CENTRM is limited to the thermal groups, since no eigenvalue calculation is performed in CENTRM. The MG scatter source in the thermal range has upscatter contributions that depend on group fluxes from lower energy groups in the LMR, so that outer iterations are performed over thermal groups in the LMR until the upscatter portion of the MG scatter source converges.

Multigroup data for CENTRM calculation

Group cross-section data for the MG calculations are taken from the input MG library which should include a combined 2D transfer matrix representing all pertinent scatter reactions (viz, elastic, inelastic, coherent and incoherent thermal reactions, n-2n, etc). MG cross sections also should be problem-dependent values. This is done by processing the data with BONAMI prior to the CENTRM calculation. BONAMI converts the problem-independent cross-sections into problem-dependent values by using the Bondarenko factors on the MG library.

Conversion of multigroup fluxes to pseudo-pointwise values

The MG flux solution provides the integrated flux over lethargy, for each group interval. The average flux within a group is assumed to approximate the value of the flux per unit lethargy at the midpoint lethargy of the group; thus a set of "pseudo-pointwise" angular fluxes and moments can be obtained for the NG_U and NG_L mesh points in the UMR and LMR, respectively. For lethargy point u_n , corresponding to the midpoint lethargy of group g contained within the LMR and UMR, a PW flux value is computed from the expression,

$$\Psi(u_n) = \Psi_g/\Delta u_g \tag{7.4.44}$$

where Δu_g is the lethargy width of group g. Eq. (7.4.44) provides PW flux values for lethargy mesh points,

$$u_1 \ldots u_{NGU,} \quad \text{and} \quad u_{NGU+NP+1} \cdots \ldots u_{NT} \tag{7.4.45}$$

A linear variation of the flux per unit lethargy is assumed between lethargy points to obtain a continuous representation in the UMR and LMR.

7.4.2.6 The Boltzmann equation within the PW range

In contrast to the "pseudo-pointwise" fluxes obtained from the MG transport calculation, a true PW solution is performed for the N_P lethargy points between DEMAX and DEMIN. The PW solution is performed within a loop over energy groups: i.e., for each of the NG_P groups in the PW range there is an additional loop over all lethargy mesh points contained inside the group. This approach facilitates coupling of the scatter source from the UMR to the PW range and from the PW and LMR.

Evaluating Eq. (7.4.1) at each of the N_P energy mesh-points in the PW range gives a system of integro-differential equations that can be solved to obtain the PW flux moments, per lethargy, for the N_P energy mesh points in the range DEMAX to DEMIN-which correspond to the lethargy points, $U_{NGU+1}, \ldots U_{NGU+NP}$. Again linear variation of the flux between lethargy points is assumed, to obtain a continuous spectrum. Substituting Eq. (7.4.4) into Eq. (7.4.1), the PW transport equation at mesh point n is found to be,

$$\Omega \bullet \nabla\Psi_n(r,\Omega) + \Sigma_{t,n}(r)\Psi_n(r,\Omega) = \sum_{\ell k} \frac{2\ell+1}{2} Y_{\ell k}(\Omega) S_{\ell k,n}(r) + Q_n(r,\Omega) \tag{7.4.46}$$

for $n = (NG_U + 1), \ldots., (NG_U + N_P)$

where

$$\begin{aligned} \sum_{t,n}(\mathbf{r}) &= \sum_{t}(\mathbf{r}, u_n) \\ \Psi_n(\mathbf{r},\Omega) &= \Psi(\mathbf{r},\Omega,u_n) \\ S_{\ell k,n}(\mathbf{r}) &= S_{\ell k}(\mathbf{r},u_n) \end{aligned} \tag{7.4.47}$$

Aside from the definition of the cross-section data, the above equation appears identical in form to the MG transport equation, and can be solved with virtually the same algorithm as the MG solution, once the scatter source moments are determined. The same computer routines in CENTRM calculate both the MG and PW fluxes. However, a major conceptual difference between the PW and MG transport equations is that the PW equation describes a differential neutron balance per unit lethargy *at an energy point*, while the MG equation represents an integral balance over an interval of lethargy points. Although this type of point solution is not inherently conservative over the intervals defined by the energy mesh, the particle balance for each interval has been found to be very good. It should also be noted that exact particle conservation is not a

strict requirement for this type of application where flux spectra rather than particle balances are primarily of interest.

In the PW range the scatter source is composed of (a) MG-to-PW scatter from the UMR and possibly upscatter from the LMR if the PW range extends into thermal, and (b) PW-to-PW scatter from points in the PW range. The submoment expansion method described previously is used in CENTRM to provide an efficient method of evaluating the PW-to-PW downscatter source for the epithermal range, which includes most of the resolved resonances.

Scattering sources for the PW range

In the case of elastic scatter from nuclide "j," only the lethargy interval below $u_n - \varepsilon^{(j)}$ can scatter to a lethargy point u_n in the PW range. If $u_n - \varepsilon^{(j)}$ is negative, then some portion of the source at u_n is due to UMR-to-PW from energies above DEMAX, since zero-lethargy is equal to the top energy of the PW range. Otherwise, the elastic source is entirely PW-to-PW.

For any given nuclide j, the lowest lethargy in the UMR range that contributes to the elastic scatter source in the PW range is equal to $-\epsilon^{(j)}$. Let "jL" represent the lightest *non-hydrogen* nuclide (i.e., having the smallest A value greater than unity) in the system. The associated fractional energy loss for this material is indicated as α_L, so that the highest energy neutron in the UMR range that can scatter into the PW range from an elastic collision with any non-hydrogenous moderator will have an energy equal to DEMAX/α_L. The corresponding lethargy is equal to be the negative value $-\epsilon^{(L)}$, or -ln(α_L). The value of $-\epsilon^{(L)}$ is actually adjusted in CENTRM to coincide with the immediately preceding multigroup boundary, which has a lethargy value designated as u_L. The interval of negative lethargy in the UMR between u_L and 0 has been defined previously to be transition range, because the elastic slowing-down source from this interval provides a transition between the UMR and PW solutions, respectively. The transition range always contains an integer number of groups, corresponding to MGTOP to MGHI. The total downscatter source from the UMR to lethargy u_n is composed of elastic and inelastic contributions from the transition range between [u_L,0]; and contributions from the "*high*" energy range from lethargies below u_L. The high contribution comes from inelastic and hydrogen elastic reactions in the energy interval above the transition range.

The downscatter source at u_n in the PW range can thus be expressed as the sum of three distinct contributions - S_{HI}, S_{Tr}, and S_{PW} -, that correspond to scatter from the high region of the UMR, the transition region of the UMR, and the PW ranges, respectively. The source moments appearing in Eq. (7.4.46) can thus be expressed as:

$$\begin{aligned} S_{\ell k,n}(r) &= S_{\ell k,HI}(r, u_n) + S_{\ell k,Tr}(r, u_n) + S_{\ell k,PR}(r, u_n) \\ &= \int_{-\infty}^{u_L} S_{\ell k}(u' \to u)\, du' + \int_{u_L}^{0} S_{\ell k}(u' \to u)\, du' + \int_{0}^{u_n} S_{\ell k}(u' \to u)\, du' \end{aligned} \tag{7.4.48}$$

Downscatter source from high region of the UMR to the PW range (SHI)

The high region of the UMR corresponds to groups 1 through MGTOP-1. The MG-to-PW scattering source (S_{HI}) from high energy region originates in the energy range above DEMAX/α_L; i.e., lethargies below u_L (see Fig. 7.4.2). In this region, inelastic reactions may scatter neutrons to the PW range; but due to the definition of u_L, the only elastic reactions that scatter to the PW range are due to hydrogen. Therefore in general, the MG matrices describing scatter from groups in high region to groups in the PW range correspond to discrete and continuum inelastic reactions, and elastic scatter from hydrogen. If g' is an arbitrary group in the UMR range above the transition interval and g is a fixed group interval in the PW range, then the rate that neutrons

scatter from all groups g' in the high region to all energy points in g, for a given direction Ω, is obtained from the usual expression for MG-to-MG transfers, and is equal to

$$S_g(r,\Omega) = \sum_{\ell k} \frac{2\ell+1}{2} Y_{\ell k}(\Omega) S_{\ell k,g} \tag{7.4.49}$$

where MGLO > g > MGHI, and the MG source moments are,

$$S_{\ell k,g} = \sum_{g'=1}^{MGTOP-1} \Sigma_{\ell,g'\to g} \Psi_{\ell k,g'} \tag{7.4.50}$$

while Eq. (7.4.50) gives the moments of the overall scatter rate from all groups in the high range into the *entire* PW group g, it is necessary to determine how the group source should be distributed over the PW energy mesh contained within the group; i.e., it is desired to extract the PW source moments, from the group moments by applying some "intra-group" distribution $H_{\ell k,g}(E)$ such that,

$$S_{\ell k,HI}(u) = S_{\ell k,g} H_{\ell k,g}(E), \qquad \text{for } u(E)\varepsilon \text{ group } g \tag{7.4.51}$$

The intra-group distribution has units of "per unit lethargy," and its integral over the group is normalized to unity. This form of the scatter source preserves the MG moments $S_{\ell k,g}$, whenever $S_{\ell k,HI}(u)$ is integrated over group g, ensuring that the correct number of neutrons (as determined from the UMR calculation) will always be transferred from the high range into the PW group. Only the distribution within the group is approximate.

Recall that the scatter source of concern here is due only to elastic scatter from hydrogen and inelastic scatter from all other materials. In the case of *s*-wave elastic scatter from hydrogen, the P_0 and P_1 moments per unit lethargy, respectively, can be rigorously expressed in the form of Eq. (7.4.51) with

$$H_0 \propto E \quad , \quad \text{and} \quad H_1 \propto E^{3/2} \tag{7.4.52}$$

These expressions can be inferred directly from the moments of the scatter kernel in Eq. (7.4.17). The higher order scatter moments for hydrogen have a somewhat more complicated form containing sums of energy functions; but since these moments are usually less important than the first two moments, a less rigorous treatment of their intra-group distribution is used. The intra-group distribution due to inelastic scatter depends on the Q values for the individual levels, and these are not available on the multigroup libraries. Fortunately, the scatter source in the PW range is not very sensitive to the assumed intra-group distribution for inelastic scatter, as long as the total inelastic source for the group is computed correctly. As a reasonable trade-off between rigor and complexity, the high energy component of the UMR-to-PW scatter source is approximated using H_0 for the intra-group distribution of all P_0 moments, and H_1 for all higher order moments. This approximation produces the correct intra-group variation for the lowest two moments of the hydrogen scatter source, but the higher order moments of hydrogen and the inelastic scatter source are not distributed exactly throughout the group. However, the integrated source moments are correct in all cases. Again, it should be emphasized that the approximations discussed here only apply to the UMR-to-PW component designated as S_{HI}, which comes from reactions above the transition range (energies above E_{HI}/α_L). This is often a small contribution to the overall PW source term.

Scattering sources from UMR transition region and epithermal PW range

Most coupling between the UMR and the PW range is due usually to elastic scatter from energies immediately above DEMAX. The contribution to the PW source due to downscatter source from this transition range has been designated $S_{Tr}(u_n)$. The other component of the PW source, $S_{PW}(u_n)$, accounts for the scattering source coming from all lethargies lower than u_n in the PW range. It is convenient to combine the two sources together as the PW epithermal source called "S_{Ep}," which has an lk_{th} moment given by Eq. (7.4.32),

$$\begin{aligned} S_{\ell k,Ep} &= \int_{u_L}^{u_n} S_{\ell k}(r, u' \rightarrow u)\, du' \\ &= \sum_j \sum_{K=-\ell}^{\ell} Z_{\ell,k}^{(j)} h_K(E) \int_{u_n-\varepsilon^{(j)}}^{u_n} \Psi_{\ell k}(u') \quad \Sigma^{(j)}(u') \quad h_K^{-1}(E')\, du'. \end{aligned} \tag{7.4.53}$$

This is done because CENTRM uses the submoment expansion technique to compute both the PW-to-PW epithermal source from the PW range as well as the MG-to-PW source from the transition range of the UMR. Note that elastic scattering from the transition range only impacts the PW scatter source at the initial mesh points in the PW range; i.e., those contained in the interval $0 < u_n < \epsilon^{(j)}$, for nuclide j. Beyond these mesh points the elastic scatter source is due only to PW-to-PW scatter, as illustrated in Fig. 7.4.2.

The epithermal elastic source at u_n, coming from the range u_L to u_n, is expressed as an integral over the immediately preceding lethargy mesh interval from u_{n-1} to u_n plus the integral over the remaining lethargy interval, as illustrated in Fig. 7.4.3. The former integral is designated as $I(u_{n-1},u_n)$ and the latter as $I(u_L,u_{n-1})$, so that

$$S_{\ell k,E\ell}(u_n) = I(u_{n-1}, u_n) + I(u_L, u_{n-1}) \tag{7.4.54}$$

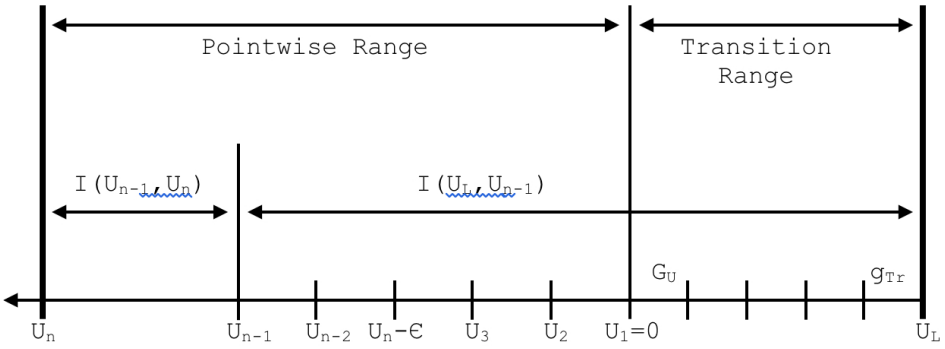


Fig. 7.4.3: Definition of cumulative integral elements.

The lethargy mesh in CENTRM is constrained such that the maximum lethargy gain in an elastic reaction ($\epsilon^{(j)}$) is always greater than the maximum mesh interval size, which insures that $I(u_{n-1},u_n)$ always includes the full panel from u_{n-1} to u_n. In the above and subsequent equations the explicit dependence of S_{Ep} on independent variables other than lethargy is not shown for notational convenience. The integral $I(u_{n-1},u_n)$ is evaluated approximately by applying the trapezoidal rule, which leads to,

$$I(u_n, u_{n-1}) = \int_{u_{n-1}}^{u_n} S_{\ell k}(u' \rightarrow u_n)\, du' \sim \frac{[S_{\ell k}(u_n \rightarrow u_n) + S_{\ell k}(u_{n-1} \rightarrow u_n)]}{2} \times (u_n - u_{n-1}) \tag{7.4.55}$$

Using the submoment expansion from Eq. (7.4.38), Eq. (7.4.55) can be written for elastic scatter as

$$I(u_{n-1}, u_n) = \Sigma_{n\to n}\Psi_{\ell k,n} + \sum_K Z^{(j)}_{\ell K} h_K(E_n)\, h_K^{-1}(E_{n-1})\, \Sigma^{(j)}(u_{n-1})\, \Psi_{\ell k,n-1}\frac{\Delta u_{n-1}}{2}. \tag{7.4.56}$$

The first term on the right side of Eq. (7.4.56) corresponds to the "within-point" component of elastic scatter from u_n to u_n, which only occurs for straight ahead scatter ($\mu_0 = 1$). The within-point cross section is defined as,

$$\Sigma_{n\to n} = \frac{\Delta u_{n-1}}{2}\sum_j \frac{\Sigma^{(j)}_n}{(1-\alpha^{(j)})}. \tag{7.4.57}$$

In deriving this term the following relation has been used for each nuclide:

$$\sum_K Z_K = \frac{1}{1-\alpha}. \tag{7.4.58}$$

The $I(u_L,u_{n-1})$ portion of the integral in Eq. (7.4.53) is equal to

$$I(u_L, u_{n-1}) = \sum_j \sum_{K=-\ell}^{\ell} Z^{(j)}_{\ell,K} \quad h_K(E_n) \int_{u_n-\varepsilon^{(j)}}^{u_{n-1}} \Psi_{\ell k}(u') \quad \Sigma^{(j)}(u')\, h_K^{-1}(E')\, du'. \tag{7.4.59}$$

Note that the lower lethargy limit of the integral is restricted to $u_n - \epsilon^{(j)}$, since this is the maximum limit of lethargy that can scatter to u_n in an elastic reaction. In terms of the cumulative integral operator, the integral in Eq. (7.4.59) over the interval $\left[u_n - \epsilon^{(j)}, u_{n-1}\right]$ is equal to:

$$\int_{u_n-\varepsilon(j)}^{u_n-1} \Psi_{\ell k}(u')\,\Sigma^{(j)}(u')\, h_K(E')^{-1}\, du' = \left[£\left(F^{(j)}_{\ell k,K}; u_{n-1}\right) - £\left(F^{(j)}_{\ell k,K}; u_n - \varepsilon^{(j)}\right)\right] \tag{7.4.60}$$

where $F_{\ell k,K}$ has been defined in Eq. (7.4.39). In order to evaluate Eq. (7.4.60) it is necessary to determine the cumulative integral values at u_{n-1} and $u_n - \varepsilon^{(j)}$. The lethargy u_{n-1} will always correspond to a mesh point value, but in general $u_n - \varepsilon^{(j)}$ can fall between mesh points. Evaluation of the cumulative integrals at an arbitrary limit such as $u_n - \varepsilon^{(j)}$ is performed in CENTRM by interpolation of previously calculated values stored for all the mesh points below u_n during the transport calculation at lower lethargies. The interpolated value of the cumulative integral at $u_n - \varepsilon^{(j)}$ that is subtracted in Eq. (7.4.60) is called the "*excess integral*" in CENTRM. At each lethargy point, excess integrals must be found as a function of space, nuclide, moment, and submoment. Also note that for some initial mesh points (i.e., $u_n < \varepsilon^{(j)}$) the value $u_n - \varepsilon^{(j)}$ can be negative, indicating that a portion of the PW scatter source at u_n is due to elastic scattering from the negative lethargy range above DEMAX. This means that cumulative integrals must be known for mesh intervals in the transition as well as in the PW range. Values of the cumulative integrals at all points within the transition range are first computed from the results from the UMR calculation, prior to the PW transport calculation (but after the UMR calculation). Additional cumulative integrals are then calculated successively during the PW transport solution at all mesh points and are stored as the calculation proceeds from low to high lethargy. Thus in evaluating $S_{\ell k,Ep}(u_n)$, the cumulative integrals at every space interval already will have been stored at all energy points up to (n-1), in an array called $CUM^j_{\ell k,K}$, for each nuclide j, moment ℓk, and submoment K:

$$CUM^{(j)}{}_{\ell k,K} = \left\{£\left(F^{(j)}_{\ell k,K}; u_{n'}\right), n' = 1, n-1\right\}, \tag{7.4.61}$$

so that the excess integral values can be interpolated from the above array. The first N_{Tr} elements of the array $CUM^j_{\ell k,K}$, correspond to lethargy points in the transition range, and the remainder are in the PW range, where

$\mathbf{N}_{\mathrm{Tr}} = \mathrm{G_U} - \mathrm{g_{Tr}} + 1$;

$\mathbf{g}_{\mathrm{Tr}}$ = MGTOP, the highest energy group in the transition range; (i.e., the group whose high energy boundary corresponds to u_L

$\mathrm{G_U}$ = Lowest energy group in the transition range.

Elastic cumulative integrals contained in array $\mathrm{CUM}^{j}_{\ell k,K}$, are calculated at each lethargy point $\mathrm{u_n}$ with the expression:

$$\begin{aligned} \pounds\left(\mathrm{F}^{(\mathrm{j})}_{\ell \mathrm{k,K}}; \mathrm{u_n}\right) &= \pounds\left(\mathrm{F}^{(\mathrm{j})}_{\ell \mathrm{k,K}}; \mathrm{u_{n-1}}\right) + \int_{u_{\mathrm{n-1}}}^{\mathrm{u_n}} \mathrm{F}^{(\mathrm{j})}_{\ell \mathrm{k,K}}(\mathrm{u}')\,\mathrm{du}' \\ &= \pounds\left(\mathrm{F}^{(\mathrm{j})}_{\ell \mathrm{k,K}}; \mathrm{u_{n-1}}\right) + \int_{\mathrm{u_n}-1}^{\mathrm{u_n}} \Psi_{\ell \mathrm{k}}(\mathrm{u}') \quad \Sigma^{(\mathrm{j})}(\mathrm{u}') \quad \mathrm{h_K}(\mathrm{E}')^{-1}\,\mathrm{du}' \end{aligned} \tag{7.4.62}$$

After completing the calculation of PW angular fluxes and moments at $\mathrm{u_n}$ the integral over the most current lethargy panel [$\mathrm{u_{n-1}}$,$\mathrm{u_n}$] is evaluated with the trapezoidal approximation, resulting in an updated cumulative integral array containing the value at lethargy $\mathrm{u_n}$:

$$\pounds\left(\mathrm{F}^{(\mathrm{j})}_{\ell \mathrm{k,K}}; \mathrm{u_n}\right) \approx \quad \pounds\left(\mathrm{F}^{(\mathrm{j}}_{\ell \mathrm{k,K}}; \mathrm{u_{n-1}}\right) + \Delta \mathrm{u_{n-1}} \frac{\left[\mathrm{h_K}^{-1}(\mathrm{E_{n-1}})\,\Sigma^{(\mathrm{j})}_{\mathrm{n-1}}\Psi_{\ell \mathrm{k,n-1}} + \mathrm{h_K^{-1}}(\mathrm{E_n})\,\Sigma^{(\mathrm{j})}_{\mathrm{n}}\Psi_{\ell \mathrm{k,n}}\right]}{2}, \tag{7.4.63}$$

where the cumulative integrals at the preceding mesh point are known from the previous calculation, and the flux moments $\psi_{\ell k}(u_n)$ are determined from the transport calculation at the current lethargy point. Only a single panel of integration is required to update the cumulative integrals, significantly reducing the amount of computation compared to recomputing the entire summation again at each new energy point. The integration is performed rapidly with the trapezoidal approximation, which should be accurate since the energy mesh is defined to reproduce the macroscopic cross sections linearly between mesh points. In order to avoid loss of numerical significance, the set of stored cumulative integrals is periodically "renormalized," by translating to a new reference lethargy point (recall that only the *differences* of cumulative integrals is needed).

Elastic cumulative integrals for the transition range are calculated with a slightly different expression, using MG flux moments obtained in the UMR calculation. Because the transition interval is part of the UMR, it is convenient to evaluate cumulative integrals at lethargy values corresponding to group boundaries. This requires approximating the energy distribution of the flux spectrum within each group in the transition range. To evaluate the cumulative interval in the transition range of some nuclide j, the scalar flux per energy (at a given space location) within a transition group is approximated as: $\Phi(\mathrm{E}) = \mathrm{M}^{(\mathrm{j})}/\mathrm{E}$, where $\mathrm{M}^{(\mathrm{j})}$ is a normalization constant defined so that the MG outscatter rate (i.e., slowing-down density) from the group is preserved. It can be shown that this normalization condition requires that

$$\mathbf{M}^{(j)} = \frac{\left[\Sigma^{(j)}_{t,g'} - \Sigma^{(j)}_{a,g'} - \Sigma^{(j)}_{g'},\right]\Delta u_{g'}}{\xi^{(j)}\Sigma^{(j)}_{s,g'}} \times \left[\phi_{g'}/\Delta u_{g'}\right] \tag{7.4.64}$$

where ξ is the average lethargy gain in an elastic reaction and $\Sigma_{g'g'}$ is the within-group MG scatter cross section. Thus the scalar flux per unit lethargy used to evaluate cumulative integrals of nuclide j is:

$$\phi(\mathrm{u}') = \mathrm{M}^{(\mathrm{j})}; \quad \text{for } \mathrm{u}' \in \mathrm{g}', \text{ and } \mathrm{g}' \in \text{ Transition region of UMR} \tag{7.4.65}$$

Within-group energy spectra for the higher order flux-moments could be approximated in similar manner by preserving the higher order Legendre moments of the slowing-down density, but CENTRM simply uses the

same form in Eq. (7.4.64) for all flux moments, so that in general the within-group energy distribution for any ℓk_{th} moment in the transition range is approximated as,

$$\Psi_{\ell k}(u') = \frac{\left[\sum_{t,g'}^{(j)} - \Sigma_{a,g'}^{(j)} - \Sigma_{g'g'}^{(j)}\right] \Delta u_{g'}}{\xi^{(j)} \sum_{s,g'}^{(j)}} \times \left[\Psi_{\ell k,g'} / \Delta u_{g'}\right] \tag{7.4.66}$$

for $u' \in g'$, and$g' \in$ transition region of UMR. Therefore the following integrals can be evaluated:

$$\int_{u_{g'}}^{u_{g'+1}} h_K^{-1}(E') \Sigma^{(\mathrm{j})}(\mathrm{u}') \Psi_{\ell \mathrm{k}}(\mathrm{u}')\, \mathrm{du}' = \frac{\sum_{\mathrm{r,g'}}^{(\mathrm{j})} \Delta \mathrm{u_{g'}}}{\xi^{(\mathrm{j})}} \frac{\Psi_{\ell \mathrm{k,g'}}}{\Delta \mathrm{u_{g'}}} \int_{\mathrm{u_{g'}}}^{\mathrm{u_{g'+1}}} \mathrm{h_K^{-1}}(\mathrm{E}')\, \mathrm{du}'. \tag{7.4.67}$$

Integration of the $\mathrm{h_k}^{-1}$ function is performed analytically to give the cumulative integral at any group boundary $\mathrm{u_g}$ in the transition range:

$$\begin{aligned} &£\left(\mathrm{F}_{\ell \mathrm{k,K}}^{(\mathrm{j})}; \mathrm{u_g}\right); \sum_{\mathrm{g'=g}}^{\mathrm{G_U}} \frac{\Sigma_{\mathrm{r,g'}}^{(\mathrm{j})} \Delta \mathrm{u_{g'}}}{\xi^{(\mathrm{j})}} \frac{\Psi_{\ell \mathrm{k,g'}}}{\Delta \mathrm{u_{g'}}} \times \left[\frac{2}{\mathrm{K}+2}\right] \left[\mathrm{h_K^{-1}}\left(\mathrm{E_{g'+1}}\right) - \mathrm{h_K^{-1}}\left(\mathrm{E_{g'}}\right)\right] \\ &\mathrm{g} = \mathrm{g_{Tr}}, \quad \mathrm{g_{Tr+1}}, \ldots\ldots \mathrm{G_U} \end{aligned} \tag{7.4.68}$$

Eq. (7.4.68) is used to obtain the initial $\mathrm{N_{Tr}}$ values of the cumulative integrals, corresponding to the transition range. If the lower limit of the integral in Eq. (7.4.60) is negative, then the cumulative integral at $u_n - \varepsilon^{(j)}$ is interpolated from among the set of $\mathrm{N_{Tr}}$ tabulated values generated by Eq. (7.4.68); otherwise it is interpolated from the values that were computed with Eq. (7.4.63). The following algorithm is used to interpolate cumulative integrals for negative lethargy arguments (i.e., in the transition range):

$$\begin{aligned} £\left(F_{\ell k,K}^{(j)}; u\right) &= £\left(F_{\ell k,K}^{(j)}; u_g\right) + \frac{h_K^{-1}(E) - h_K^{-1}\left(E_g\right)}{h_K^{-1}\left(E_{g+1}\right) - h_K^{-1}\left(E_g\right)} \\ &\times \left[£\left(F_{\ell k,K}^{(j)}; u_{g+1}\right) - £\left(F_{\ell k,K}^{(j)}; u_g\right)\right] \end{aligned} \tag{7.4.69}$$

or $u(E) \in g$; and g $\in$ transition range of UMR.

Because the energy mesh in the PW range is very fine, simple linear interpolation of the cumulative integrals is used for positive lethargy arguments.

The complete epithermal elastic scatter source $S(r, \Omega, u_n)$ appearing in Eq. (7.4.46) at any mesh point $\mathrm{u_n}$ corresponds to a spherical harmonic expansion using the previously derived moments of $\mathrm{S_{HI}}$ and $\mathrm{S_{Ep}}$. This angular scatter source is equal to,

$$\begin{aligned} &\mathrm{S}(\mathbf{r}, \Omega, \mathrm{u_n}) = \Sigma_{\mathrm{n}\to\mathrm{n}} \Psi_{\mathrm{n}}(\mathbf{r}, \Omega) \\ &+ \sum_{\ell \mathrm{k}} \frac{2\ell+1}{2} \mathrm{Y}_{\ell \mathrm{k}}(\Omega) \left\{ \mathrm{H}_\ell(\mathrm{E_n}) \sum_{\mathrm{g'=1}}^{\mathrm{g_{Tr}-1}} \sum_{\ell, \mathrm{g'} \to \mathrm{g}} \psi_{\ell \mathrm{k,g'}} \right. \\ &\left. + \sum_{\mathrm{j}} \sum_{\mathrm{K}} \mathrm{Z}_{\ell \mathrm{K}}^{(\mathrm{j})} \mathrm{h_K}(\mathrm{E_n}) \left[0.5 \Delta \mathrm{u_{n-1}}\, \mathrm{F}_{\ell \mathrm{k,K}}^{(\mathrm{j})}(\mathrm{u_{n-1}}) + £\left(\mathrm{F}_{\ell \mathrm{k,K}}^{(\mathrm{j})}; \mathrm{u_{n-1}}\right) - £\left(\mathrm{F}_{\ell \mathrm{k,K}}^{(\mathrm{j})}; \mathrm{u_n} - \varepsilon^{(\mathrm{j})}\right)\right] \right\} \end{aligned} \tag{7.4.70}$$

The above expression was written explicitly for the case of elastic scatter; however, the discrete level inelastic PW source can be incorporated with little modification. The only changes are that additional cumulative integral terms corresponding to each inelastic level will appear in Eq. (7.4.70); the cumulative integrals

for the inelastic levels must be computed by integrating the more general expression in Eq. (7.4.39); and the lethargy arguments for the inelastic cumulative integrals are the generalized lethargy limits u_{LO} and u_{HI} defined in Sect. 7.4.2.4 and [CENTRM-Wil00].

Note that Eq. (7.4.70) contains the term $\Sigma_{n\to n} \quad \Psi_n(r,\Omega)$ which can be subtracted from both sides of the transport equation in Eq. to give a slightly altered form of the PW transport equation that contains a modified scatter source and a modified total cross section. The modified source component is identical to the expression in Eq. (7.4.70) with the within-point term $\Sigma_{n\to n}\Psi_n(r,\Omega)$ removed. The modified total cross section, represented by $\tilde{\Sigma}_{t,n}$ has the appearance of a "transport-corrected" cross section given below:

$$\tilde{\Sigma}_{t,n} = \Sigma_{t,n} - \Sigma_{n\to n} \tag{7.4.71}$$

An interesting and significant consequence of this operation is that the right side of Eq. (7.4.70) no longer contains the unknown flux $\psi_n(r,\Omega)$ since the within-point term is eliminated. The resulting modified transport equation has the same form as a purely absorbing medium with a known source term; and thus can be solved without requiring scatter-source iterations in the epithermal range. However, iterations may still be required for cell cases with two reflected or albedo boundary conditions.

PW thermal scatter source

There are significant differences in the CENTRM epithermal and thermal PW transport solutions. In the epithermal range neutrons can only lose energy in scattering reactions, so that a single sweep from high to low energy (i.e., low to high lethargy) is required in the solution. On the other hand, since low energy neutrons may gain as well as lose energy in scattering reactions, outer iterations are required to converge the thermal scattering source. Furthermore, the PW scatter kernels $\Sigma_\ell(u'\to u)$ in the epithermal range represent two-body interactions (such as elastic and discrete-level inelastic reactions) between a neutron and a stationary nucleus. The simple kinematic relations for these cases allow the efficient sub-moment expansion method to be utilized in computing scattering source moments. Thermal scattering reactions are not two body reactions, but rather represent an effective average over the molecular velocity distribution; thus, there is no simple kinematic relationship between neutron energy loss and the angle of scatter relative to its initial direction. In solving the transport equation for thermal neutrons, the scatter source at lethargy u_n is approximated as a summation over the "N" mesh points in the thermal range,

$$\int_{\text{thermal}} \Sigma_\ell^{(j)}(u'\to u_n)\,\Psi_{\ell k}(u')\,du' = \sum_{m=1}^{N} W_m \Sigma_\ell^{(j)}(u_m\to u_n)\,\Psi_{\ell k}(u_m) \tag{7.4.72}$$

where

m = 1 is the thermal/epithermal boundary point;

m = N is the lowest thermal energy point; and

W_m are standard quadrature weights for trapezoidal integration with N-1 lethargy panels:

$$\begin{aligned} W_m =& 0.5\times(\Delta u_m + \Delta u_{m+1}) \quad ; \quad \text{for m} = 2,3,\ldots N-1 \\ & 0.5\times\Delta u_m \quad ; \quad \text{for m} = 1 \quad \text{or} \quad N \end{aligned} \tag{7.4.73}$$

Point-to-point cross-section moments in the thermal range are computed from the free-gas or bound kernels evaluated at the desired initial (u_m) and final (u_n) lethargy mesh points. For a given outer iteration, the summation in Eq. (7.4.72) is evaluated using the most recently computed flux moments. In many instances the main purpose of the CENTRM calculation will be to obtain a PW spectrum for resonance self-shielding calculations. In these cases the thermal flux does not have to be converged very tightly to obtain a reasonable

thermal spectrum for self-shielding low energy resonances, so that only a few outer iterations are typically employed.

An additional complication in the thermal calculation is that inner iterations are necessary to converge the "within-point" (no energy loss) contribution of the thermal scattering source, due to the presence of PW flux moments at lethargy point m = n. No inner iterations are required to converge the within-point elastic scatter term in the epithermal PW calculation because there can be no change in the neutron direction if there is no energy loss, unlike the thermal range.

A space-dependent rebalance calculation for the entire thermal energy band is performed between outer iterations in order to speed up convergence of the solution. Reaction rates and leakage values appearing in the thermal-band rebalance equation are obtained by integrating PW values over the thermal range. Other acceleration techniques, such as over-relaxation, extrapolation, and renormalization, are also employed.

Downscatter source from the epithermal PW range to the LMR

MG transport calculations performed in the energy range below DEMIN, which includes the thermal energy range, are coupled to the epithermal PW range transport calculations by the slowing down source. The epithermal PW-to-LMR scatter source represents the contribution to the multigroup source in some fixed group g contained in the LMR, from scatter reactions in the epithermal range above DEMIN. The lethargy value corresponding to the energy DEMIN (i.e., the bottom energy of the PW range) will be indicated as u_{PW}, thus u_{PW} = ln(DEMAX/DEMIN); while the lethargy corresponding to the thermal energy boundary will be designated as u_{TH}. The cut-off lethargy for the epithermal PW range will correspond to: $u_{cut} = \min(u_{PW}, u_{TH})$. If there is no PW thermal calculation in CENTRM, then $u_{cut} = u_{PW}$; otherwise, u_{cut} = u:sub:*TH*. For a given nuclide j, the lowest lethargy in the epithermal PW range from which a neutron can scatter elastically into the LMR is equal to $(u_{cut} - \varepsilon^{(j)})$ If the value of $(u - \varepsilon^{(j)})$ is greater than u_{cut}, then an elastic collision with nuclide j cannot moderate an epithermal neutron from the PW range to u. Therefore in general for a given material zone, only a limited number of nuclides (possibly none) and a limited portion of the epithermal PW energy range may be able to scatter neutrons elastically to any particular group in the LMR. Utilizing the elastic scatter kernel and applying a sub-moment expansion to the resulting expression, the source moment describing scatter from the PW epithermal range to a lethargy u in the LMR is found to be

$$S_{\ell k}^{(j)}(u) = \sum_{K} Z_{\ell K}^{(j)} \quad h_K(E) \int_{u-\varepsilon^{(j)}}^{u_{\text{cut}}} F_{\ell k,K}^{(j)}(u')\, du' \tag{7.4.74}$$

where u is in group g; and $g \in$ LMR.

The integral in the above expression can be evaluated from cumulative integrals stored during the epithermal PW transport calculation. Thus the source moment per unit lethargy at u in the LMR range, due to epithermal scattering from nuclide j, can be written as,

$$S_{\ell k}{}^{(j)}(u) = \sum_{K} Z_{\ell K}^{(j)} h_K(E) \left[£\left(F_{\ell k,K}^{(j}; u_{cut} \right) - £\left(F_{\ell k,K}^{(j)}; u - \varepsilon^{(j)} \right) \right], \tag{7.4.75}$$

for u in group g and $u - \varepsilon(j) < u_{PW}$.

The source per unit lethargy in Eq. (7.4.75) is integrated over the "sink group" g in the LMR to determine the desired MG scatter source moment due to reactions in the epithermal PW range. The actual integral over group g is performed numerically by introducing a three-point (two panel) integration mesh within the group,

as follows:

$$\begin{aligned} u_I &= \text{initial integration point in group g} = \text{lethargy at top energy of group g} = u_g \\ u_F^{(j)} &= \text{final integration point in group g} \\ &= \text{MIN}\left\{u_{g+1}; u_{cut} + \varepsilon^{(j)}\right\}, \quad \text{where } u_{g+1} = \text{lethargy at bottom of energy of group g} \\ u_A^{(j)} &= \text{middle integration point in group g} = 0.5\left(u_F^{(j)} + u_I\right) \end{aligned} \tag{7.4.76}$$

Note that the final and middle points of integration (i.e., $u_F^{(j)}$ and $u_A^{(j)}$) may be nuclide dependent; and if $u_I - \varepsilon^{(j)} > u_{cut}$, then nuclide j does not contribute to the pointwise-to-LMR scatter source in g. Applying the two-panel Simpson's approximation for integration over group g results in

$$S_{\ell k,g}^{(j)} = \Delta^{(j)}/3\left[\, S_{\ell k}^{(j)}(u_I) + 4\, S_{\ell k}^{(j)}\left(u_A^{(j)}\right) + S_{\ell k}^{(j)}\left(u_F^{(j)}\right)\right] \tag{7.4.77}$$

where $\Delta^{(j)} = 0.5\left(u_F^{(j)} - u_I\right)$.

The values for

$S_{\ell k}^{(j)}(u_I)$, $S_{\ell k}^{(j)}\left(u_A^{(j)}\right)$, and $S_{\ell k}^{(j)}\left(u_F^{(j)}\right)$ in Eq. (7.4.77) are obtained by evaluating Eq. (7.4.75) at the lethargy values u_I, $u_A^{(j)}$, and $u_F^{(j)}$, respectively. Use of more than two panels for the group integration was found to have an insignificant impact.

The complete epithermal PW-to-LMR source in group g is finally obtained by summing Eq. (7.4.75) over all nuclides and then substituting the spherical harmonic moments into the Legendre expansion of the MG scatter source, resulting in

$$S_{PW\to g} = \sum_{\ell k} \frac{2\ell+1}{2} Y_{\ell k}(\Omega) \sum_j \Delta^{(j)}/3 \quad \left[\, S_{\ell k}^{(j)}(u_I) + 4\, S_{\ell k}^{(j)}\left(u_A^{(j)}\right) + S_{\ell k}^{(j)}\left(u_F^{(j)}\right)\right] \tag{7.4.78}$$

Thermal scatter sources from LMR and PW range

If the value of DEMIN is specified to be below the thermal energy boundary, the portion of the PW range between DEMIN and the thermal cutoff, as well as the entire LMR, will be contained in the thermal range. In this case thermal neutrons will downscatter from the thermal PW range to the LMR, and upscatter from the LMR to the thermal PW range.

The latter thermal source (LMR-to-PW) is computed in exactly the same manner as used to compute the UMR-to-PW source S_{HI}, described in Sect. 7.4.2.6.2. On the other hand, the scatter source from the thermal PW to the LMR is computed with a similar approach as given in the previous section for epithermal PW-to-LMR scatter. In this case Eq. (7.4.78) is used as before, except the source moments are not obtained from the submoment expansion in Eq. (7.4.75), but rather by evaluating the PW thermal scatter expression in Eq. (7.4.72).

In performing the transport calculation for any group g in the LMR range, the PW-to-MG source component in Eq. (7.4.78) is added to the MG-to-MG scattering into g from all groups in the UMR and LMR ranges, respectively, to obtain the total scatter source.

7.4.2.7 Determination of energy mesh for PW flux calculation

The energy mesh for the PW flux computation is determined for a specific problem as follows: (a) for each zone-composition, microscopic cross-section data are interpolated (if necessary) to the desired zone-temperature, and a union energy mesh is formed from the energy meshes of PW total cross sections of all materials in that zone, plus the MG boundaries; (b) macroscopic total cross sections are computed for the union meshes in each zone; (c) union meshes for each zone are thinned (i.e., some energy points eliminated) in a manner that allows the zone macroscopic cross section to be interpolated linearly, within some input error tolerance; (d) a union mesh is created from the thinned energy meshes for each zone thus producing a "global" energy mesh; (e) the global mesh is checked to insure that it still contains group boundaries and midpoint-energies of the input MG library, and finally, (f) still more points may be added to constrain the maximum interval width between successive lethargy points to be less than some fraction of the maximum lethargy gained by elastic scatter from a fictitious nuclide having a mass of approximately 400. The fraction used in limiting the maximum size of any lethargy interval can be set by the input value of "FLET," but is defaulted to a value of 1/3.

The mesh thinning procedure is effective in reducing the number of energy points in the PW transport calculation, while preserving essential features of the macroscopic cross-section data that affect the flux spectrum; viz, the mesh is typically fine in energy regions corresponding to important resonances, but coarser where there is little variation in the macroscopic cross-section data. The default thinning tolerance is 0.1%. A less stringent thinning tolerance may give a large reduction in computation time, but also can affect the accuracy.

7.4.2.8 CENTRM cross sections and fixed sources

CENTRM PW cross-section libraries

SCALE includes CE nuclear data for all materials and all reaction types available in ENDF/B, processed for several different temperatures. The CE data, spanning the energy range from 10^{-5} eV to 20 MeV, are stored in separate files for individual nuclides, which can be used for CENTRM as well as CE Monte Carlo calculations. The CRAWDAD module reads these files and merges the data to form a single CENTRM formatted library containing only the particular materials, cross section types, temperatures, and energy range needed for a given calculation (see Sect. 8.1.6). In general the CENTRM library includes CE data for the unresolved, as well as the resolved, resonance range. Unresolved resonance data typically have rather smooth variations, but in reality the cross sections represent average values for very closely spaced resonances that can not be measured individually.

Linearization of MG cross sections and fixed sources

Shielded group cross sections from the input MG library are always required for the UMR and LMR portions of the CENTRM calculation. Two approaches are available to translate multigroup cross sections into pseudo-PW data at energy points within a group. The first is a "step" approximation in which $\sigma) = \sigma_g$, where E_n is any energy point contained in group g. This leads to a histogram representation of σ(E) that is discontinuous at the group boundaries. If the multigroup data show significant variation between adjacent groups, then the histogram approach can introduce discontinuities and oscillations into the pointwise flux. An alternative approach is to "linearize" the multigroup cross sections, using a linear representation that preserves the group-average values and is continuous at the group boundaries. Although the resulting cross section is continuous and does not cause distortions in the flux spectrum, the data does not necessarily represent the actual energy variation of the cross section.

Input fixed source terms are treated in a similar manner. The multigroup spectra that are input by the user may be converted either to a discontinuous histogram function in lethargy; or may be linearized by group. In

the latter case the resulting groupwise-linear function is evaluated at the energy mesh points to obtain the pointwise source term.

7.4.3 AVAILABLE METHODS FOR SOLVING TRANSPORT EQUATION

CENTRM offers several calculation options for solving the Boltzmann equation. Some of these are only available for either the MG or PW calculations, respectively. In the case of the MG methods, the calculation procedures are similar to those described in the XSDRNPM documentation. The following sections briefly describe the PW transport approximations available in CENTRM.

7.4.3.1 Discrete ordinates

The discrete ordinates method can be used for both MG and PW solutions. The main difference in the solution is the computation of the scattering sources: the multigroup method uses group-to-group scatter matrices, while the PW method uses the submoment expansion technique described earlier. Also, as previously discussed, the pointwise discrete ordinates equation has the same form as the transport equation for a purely absorbing medium; so that inner iterations are not required to converge the pointwise scattering source. The XSDRNPM documentation shows the finite-difference form of the discrete ordinate equations, and includes a discussion of S_N quadratures, the weighted-difference model, angular streaming coefficients, treatment of boundary conditions, and other standard procedures used in the CENTRM 1-D discrete ordinates solution.

Homogenized infinite medium

A homogenized infinite medium calculation can be performed for either the MG or the PW energy ranges. This method is essentially a "zero-dimensional" model that has no spatial or angular variation in the flux (only energy dependence). The materials contained in all zones are "smeared" into a single homogenized mixture using volume weighting of the number densities, and the effective external source is defined to be the volume-weighted source density. The resulting homogenized composition is then solved as an infinite medium, so that the PW scalar flux is equal to,

$$\Phi(\mathrm{u_n}) = \frac{\int \tilde{\Sigma}(u' \rightarrow \mathrm{u_n})\,\Phi(u')\,du' + \tilde{Q}_{ext}(E)}{\tilde{\Sigma}_t(\mathrm{u_n})} = \frac{\tilde{S}(\mathrm{u_n}) + \tilde{Q}_{\mathrm{ext}}(\mathrm{u_n})}{\tilde{\Sigma}_t(\mathrm{u_n})} \tag{7.4.79}$$

where $\tilde{\Sigma}_t, \tilde{S}, \tilde{Q}_{ext}$ = homogenized cross section, scatter source, and external source, respectively.

The total cross section in the PW calculation is reduced by the value of the "within-point" cross section, as discussed in Sect. 7.4.2.6. The PW scattering source is computed from a P_0 submoment expansion using cumulative integrals, as described in Sect. 7.4.2.4. The scalar flux at all space-intervals and the angular flux in all directions are set to the above value, while higher order flux moments are equal to zero.

7.4.3.2 Zonewise infinite medium

The zonewise infinite medium solution is an option for the both PW and MG calculations. This method is similar to the homogenized infinite medium option, except each zone is treated independently as an infinite medium, so that no spatial homogenization is required; i.e., Eq. is solved for each individual zone. The zonewise source corresponds to the *input* external source within the respective zone.

Note: *THERE IS NO COUPLING BETWEEN THE CALCULATIONS FOR EACH ZONE.*

Hence, any zones that do not contain an input external source will have zero fluxes, since there is no leakage between zones in the zonewise infinite medium model. To avoid this problem the user must either input a volumetric source, or specify a fission source and add a minute amount of fissionable material to generate a fission spectrum source.

7.4.3.3 Collision probability method

A collision probability (CP) solution of the 1-D integral transport equation is available in CENTRM as an alternative to the pointwise discrete ordinates approach. This option is only available for the pointwise solution, and has the following additional restrictions compared to the pointwise S_N method:

(a) only 1-D cylindrical or slab geometry is treated (no spherical geometry);

(b) periodic boundary conditions have not been implemented;

(c) interior surface sources cannot be treated;

(d) P_0 scattering (isotropic *laboratory* scatter) is assumed; no transport correction is applied to the pointwise cross sections in CENTRM.

(e) due to assumption (d), computation of the angular flux is not required in order to obtain the scalar flux; therefore, only the P_0 flux moment (scalar flux) of the angular flux is calculated.

(f) PW thermal calculations are not supported for the CP method.

This method has only had a limited amount of testing, and has not been validated as extensively as the discrete ordinates and MoC options.

The PW scattering source for the integral transport equation is similar to that for the discrete ordinates method, except that only the P_0 source moment is considered, and the "within point" correction is not applied to the total cross section for the collision probability method. Therefore only the spatial and directional treatment is substantially different in the two transport approaches. In the collision probability method the total interaction rate within a space interval is expressed in terms of collision probabilities $P_{i\prime\rightarrow i}$ corresponding to the probability that neutrons born uniformly in volume $V_{i\prime}$, at lethargy u_n will collide in interval i with volume V_i. The scalar flux in space interval "i," at lethargy point u_n obeys the integral transport equation,

$$\Sigma_i(u_n)\,\phi_i(u_n)\,V_i = \sum_{i'} P_{i'\rightarrow i}(u_n)\left[\Sigma_{i';n\rightarrow n} \quad \Phi_i(u_n) + Q_{\text{eff},i'}(u_n)\right] V_{i'} \tag{7.4.80}$$

where

$\sum_{i',n\rightarrow n}$ is the within-point cross section at i$\prime$, defined in Eq. (7.4.57); and

$Q_{\text{eff},i'}$ is the external source (Q_n) plus the P_0 downscatter source in i$\prime$ and at u_n, which is computed in a similar manner as for the discrete ordinates solution.

The boundary condition at the center of a cylinder is assumed to be reflected; while the outer boundary condition can either be vacuum or albedo. An albedo boundary condition returns a fraction of the outward leakage, using a cosine distribution for the return-current. An albedo of unity corresponds to a "white" boundary condition.

Unlike the integro-differential transport equation used by the discrete ordinates method, the integral equation at space interval i, contains the unknown fluxes at *all other* space intervals i$\prime$. This requires that inner iterations be performed for the PW solution using the CP option.

Collision Probabilities (CP) in 1-D cylindrical geometry are computed using the method developed by Carlvik [CENTRM-Car64]. The CPs for a vacuum outer boundary condition are first computed. In this case the probability that a neutron born in annular interval i with outer radius R_i, will have its next collision in annular interval j with outer radius R_j , is computed from the expression:

$$P_{i\rightarrow j} = \delta_{ij}\Sigma_j V_j + 2\left(S_{i-1,j-1} - S_{i-1,j} - S_{i,j-1} - S_{i,j}\right) \tag{7.4.81}$$

where δ_{ij} represents the Kronecker delta function and

$$S_{ij} = \int_0^{R_t} \left[Ki_3\left(\tau_{ij}^{+}\right) - Ki_3\left(\tau_{ij}^{-}\right) \right] dy \tag{7.4.82}$$

In the above expression Ki_3 corresponds to a 3rd order Bickley function, and $\tau_{ij}^{+}$and τ_{ij}^{-} are,

$$\tau_{if}^{\pm} = \Sigma \times \left(\sqrt{R_j^2 - y^2} \pm \sqrt{R_i^2 - y^2} \right) \tag{7.4.83}$$

Integration of the Bickley functions is performed numerically using Gauss-Jacobi quadrature.

The probability that a neutron born in interval "i" will escape uncollided from a cell with a vacuum outer boundary is equal to one minus the probability that it has a collision in any interval of the cell, so that

$$P_{esc,i} = 1 - \sum_j P_{i \to j} \tag{7.4.84}$$

The "blackness" (γ_i) of interval i is defined to be the probability that neutrons entering the cell with a cosine-current angular distribution on the surface will collide within interval i. The total blackness (γ) is the probability that a neutron entering the outer boundary will collide anywhere in the cell. The blackness is computed from the escape probability:

$$\Gamma_i = \frac{4V_i\Sigma_i}{S} \times P_{esc,i} \quad ; \quad \text{and} \quad \Gamma = \sum_i \Gamma_i \tag{7.4.85}$$

where S is the outer surface area of the cell.

The vacuum CPs can be modified to account for the effect of an albedo condition on the outer surface. In these cases a fraction of the neutrons reaching the outer boundary are returned isotropically back into the cell, and "bounce" back and forth between the two boundaries, with each traverse reducing the number of uncollided neutrons. Mathematically this corresponds to a converging geometric series that accounts for the cumulative effect of the "infinite" number of traverses through the cell. Evaluating the geometric sum, the collision probability (P^A) for an albedo of α can be expressed as

$$P_{i \to j}^{A} = P_{i \to j} + \frac{S}{4V_i\Sigma_i} \times \frac{\alpha\Gamma_i\Gamma_j}{1 - \alpha(1 - \Gamma)} \tag{7.4.86}$$

In the case of a slab with two vacuum boundaries, the CP's are expressed in terms of transmission probabilities $T_{i \to j}$, corresponding to the probability that a neutron born in volume i will reach surface j without a collision. The transmission probability is equal to,

$$T_{i \to j} = \frac{E_3\left(\tau_{ij}\right) - E_3\left(\tau_{ij} + \tau_i\right)}{2\tau_i} \tag{7.4.87}$$

where:

τ_{ij} is the optical distance between surface S_j and the surface volume V_i that is closest to S_j;

τ_i is the optical thickness of volume V_i; and

E_3 is the third order Exponential Function.

In a slab with vacuum boundaries, the CP for a neutron born in interval i to have its next collision in some interval j bounded by surfaces S_j and S_{j+1}, is given by the expression,

$$P_{i\to j} = T_{i\to j} - T_{i\to j+1} \tag{7.4.88}$$

A slab with a specular-reflected boundary condition on one side and a vacuum on the other can be represented as an "expanded" geometry with two vacuum boundaries, simply by adding the cell's mirror image at the reflected boundary. Hence, collision probabilities for these cases can also be obtained from the expressions for two vacuum boundaries. For slabs with an albedo (or white) boundary on one side and a reflected boundary on the other, the same expression in Eq. (7.4.86) presented earlier for cylinders can also be used to obtain albedo-corrected collision probabilities from the vacuum values. However, unlike cylindrical geometry, a slab may have specular-reflected boundary conditions on both sides, corresponding to an infinite array of repeating cells. In CENTRM the collision probabilities for a doubly-reflected slab geometry are obtained by explicitly including two reflected cells in addition to the "primary" cell, and then applying a white boundary condition on the outer surface of the expanded geometry. The resulting modified geometry, consisting of three cells (i.e., the primary plus two reflected cells) with a reflected center and outer albedo boundary, can be treated as described previously.

7.4.3.4 P_1 ("Diffusion Theory") method

The CENTRM MG calculation has an option for a P_1 calculation; however, this option is not available for the PW calculation. The method is essentially identical to that used in XSDRNPM. The P_1 option is called diffusion theory in both XSDRNPM and CENTRM input descriptions; but since the P_1 component of the scatter source is explicitly treated and since Fick's Law is not assumed, the method is actually more closely related to the P_1 spherical harmonic solution approach. The P_1 method is also used in CENTRM to generate a flux guess for the thermal flux distribution in an S_N calculation. Several outer iterations over the thermal groups are always performed with P_1 theory, prior to beginning the discrete ordinates calculation for the thermal range. In the case of PW thermal calculations, the multigroup thermal flux guess is converted in PW flux values by multiplying by the ratio of MG to PW cross-section values at each lethargy point.

7.4.3.5 Two-region (2R) method

A two-region (2R) calculation similar to the Nordheim method is also available as a PW option in CENTRM. In general the PW S_N or MoC solutions provide a more rigorous approach to compute self-shielded cross sections than the 2R method; however the 2R approximation gives accurate results for a wide range of "conventional cases." Verification studies have shown that the 2R option produces eigenvalues comparable to those obtained with the S_N and MoC options for numerous cases of interest, and the execution time is often significantly faster since only the zone-averaged PW scalar flux is computed. Thus the CENTRM 2R option is an adequate and attractive alternative for many applications.

The CENTRM 2R solution provides several advantages over the original Nordheim method. CENTRM uses pre-processed PW nuclear data, rather than the Nordheim approach of using input resonance parameters for a built-in resonance formula (Breit-Wigner). This allows the CENTRM-2R calculation to utilize PW cross sections processed from Reich-Moore resonance data in ENDF/B, while the conventional Nordheim method is limited to Breit-Wigner resonance formulae that do not treat level-level interference. Other advantages of the CENTRM-2R calculation include a rigorous treatment of resonance overlap from mixed absorbers, ability to address mixtures of arbitrary moderators with energy dependent cross sections, and capability to include inelastic and thermal scattering effects in the flux calculation.

The 2R approximation, which is a simplified version of the general collision probability method, represents the system by an interior region containing the mixture of materials to be self-shielded, and an exterior

moderator region where the asympototic flux per lethargy is approximated by a simple analytical expression (e.g., constant for epithermal energies).

CENTRM performs a separate 2R calculation for each zone. For example, if a problem consists of fuel and moderator zones, then two 2R calculations are done: one has fuel as the interior region and moderator as the exterior, and the other has moderator as interior and fuel as exterior. In the case of lattices, multiple bodies composed of the same mixture are addressed by introducing a Dancoff factor.

The 2R equation solved by CENTRM for interior region "I" is equal to,

$$\Sigma_I(u_n)\,\phi_I(u_n) = \left(1 - P_I^{(esc)}(u_n)\right)\left(S_I(u_n) + Q_I^{ext}(u_n)\right) + \Sigma_I(u_n)\,P_I^{(esc)}(u_n)\,\varphi_{asy}(u_n) \tag{7.4.89}$$

In the above equation, S_I and Q_I are the scatter and fixed sources, respectively, for interior region "I"; and $P_I^{(esc)}$ is the probability that a neutron born in the interior region will escape and have its next collision in the external region. For an interior region consisting of multiple bodies of the same composition separated by an exterior region, the escape probability is equal to

$$P_I^{(esc)}(u_n) = \frac{P_0^{(esc)}(u_n)\,[1 - C_I]}{[1 - C_I] + C_I\bar{\ell}_I\Sigma_I(u_n)\,P_0^{(esc)}(u_n)} \tag{7.4.90}$$

where $P_0^{(esc)}$ is the escape probability from a single, isolated body in the interior region; $\bar{\ell}_I$ is the average chord length of bodies in the interior region; and C_I is the Dancoff factor, corresponding to the probability that a neutron escaping one interior body will pass through the exterior region and have its next collision in another body of the interior region. Values for $P_0^{(esc)}$ are computed internally by the code at each energy mesh point, but Dancoff factors must be provided by input for each zone. In the standard XSProc computational sequence, Dancoff values are automatically computed and provided to CENTRM.

The asymptotic flux φ_{asy} for the exterior region is represented by analytical expressions in the fast, epithermal, and thermal energy ranges, respectively, as summarized in Table 7.4.7. The constants C_1, C_2 and C_3 are defined to impose continuity at the energy boundaries, and also include the overall normalization condition that the PW asymptotic flux at DEMAX is equal to the MG flux for group MGHI obtained from the UMR calculation.

Table 7.4.7: Asymptotic flux representation.

Upper energy	**Description**	**Distribution (per unit energy)**	**Distribution (per unit lethargy)**
20 MeV	Fission spectrum	$C_3E^{1/2}\,e^{-E/\theta}$ (*)	$C_3E^{3/2}\,e^{-E/\theta}$ (*)
200 keV	Slowing-down	C_2/E	C_2
5kT	Maxwellian	$C_1E\,e^{-E/kT}$	$C_1E^2\,e^{-E/kT}$
(*) θ = fission spectrum "temperature" = 1 MeV.			

Equation Eq. (7.4.89) is solved similarly to the zonewise infinite medium equation; the only difference being that the interior sources are multiplied by (1-$P_0^{(esc)}$), and the presence of an inhomogeneous source term coming from the exterior region. Like all CENTRM transport options, the PW scattering source is computed using cumulative integrals as given in Eq. (7.4.43) In the 2R option, only the P_0 scatter-source moment is needed.

7.4.3.6 Method of characteristics for a 2D unit cell

The 2D MoC option is available for square-lattice unit cell cases, with both multigroup and pointwisecross sections. The geometry for the MoC cell calculation consists of a number of concentric cylindrical zones contained within a square outer surface with reflected boundary conditions. The MoC calculation is not currently functional for triangular lattices. A multiregion cell may be used in the MoC option, but it must correspond to a similar geometry as a square lattice cell. Because the 2D MoC solution is limited to 2D rectangular lattice cell geometries, the code performs a number of internal checks to verify the input geometry is permitted. The energy mesh generation, scattering source calculation, etc. are performed in the same manner as for the discrete ordinates option, except that only P_0 scatter is treated. Unlike the CENTRM discrete ordinates method which solves the 1D transport equation, the MoC option solves the 2D transport equation for planar XY slice through a square-pitch lattice of cylindrical pins. Because the the MoC calculation treats the outer rectangular boundary of the unit cell correctly, it provides a more rigorous calculation than using an equivalent 1D cylindrical Wigner-Seitz cell. The geometry input for the MoC calculation is identical to the input for the 1D cylindrical model— the code internally converts the input value for the equivalent cylindrical radius into the outer rectangular cell dimension (pitch) by preserving the total cell area. Thus the length (L) of a side of the rectangular cell in the MoC calculation is computed form the relation $L = \sqrt{\pi R_{eq}^2}$, where Req is the equivalent outer radius of the Wigner-Seitz cell, given in the CENTRM input array of radial dimensions. Due to symmetry conditions, the MoC calculation is performed for only 1/8 of the rectangular cell (i.e., a 45 degree sector) and for the upward directions.

MoC uses a integrating factor to convert the divergence term in the transport equation into a directional derivative along the direction of neutron transport (Ω). For a given direction in the angular quadrature set, the spatial domain is spanned by a set of parallel characteristic rays originating at one boundary and terminating at the opposite boundary. The default separation distance between the parallel rays is 0.02 cm. in CENTRM. The angular flux in given direction is computed by integrating the directional derivative along the characteristic direction. A reflected boundary condition is normally applied along the rectangular outer surface of the unit cell. The default directional quadrature, which defines the characterstic directions and is used for integration, consists of 8 azimuthal angles for the 45 degree sector in the X-Y plane and 3 postive polar angles relative to the Z-axis.

The spatial domain of the unit cell is defined by zones of uniform mixtures (e.g., fuel, clad, moderator) in which the cross section at a fixed energy point or group is constant. These uniform composition regions may be further divided into sufficiently small "flat-source" sub-divisons in which the sources (scattering and external) can be approximated as being constant. The MoC computes the average flux in the flat-source region by summing over all characteristics in the volume.

[CENTRM-KW12] gives more information about the CENTRM MoC solution method.

7.4.4 CENTRM INPUT DATA

The standard mode for executing CENTRM is through the XSProc self-shielding module which automatically defines the CENTRM options, mixing table, and geometry, and also prepares the necessary MG and PW nuclear data files and passes the CENTRM PW flux results to the downstream code PMC [see section describing XSProc]. However CENTRM can also be executed in standalone using the FIDO input provided in this section. When executed through XSProc, the user can set values for most CENTRM input parameters given in this section by using keywords in the CENTRM DATA block [see XSProc section]. Note that if CENTRM is run as a standalone module, the input data files defined in Table 7.4.8 must be provided, and the default values for some input parameters are different.

FIDO INPUT FOR STANDALONE CENTRM CALCULATIONS

***** Title Card - (72 Characters)

DATA BLOCK 1 : GENERAL PROBLEM DATA

1$ $ GENERAL PROBLEM DESCRIPTION (18 entries. Defaults shown in parenthesis)

1. IGE = Problem Geometry:

   0/1/2/3 = inf. hom. medium /plane/ cylinder/ sphere

2. IZM = Number of Zones
3. IM = Number of Spatial Intervals
4. IBL = Left Boundary Condition:

0/1/2/3 = vacuum/reflected/periodic/albedo

5. IBR = Right Boundary Condition:

   0/1/2/3 = vacuum/reflected/periodic/albedo

6. MXX = Number of Mixtures
7. MS = Mixing Table Length
8. ISN = S_N Quadrature Order
9. ISCT = Order of Elastic Scattering
10. ISRC = Problem Type: 0/1/2 = fixed source/input fission source/both (0)

    [see Note 1]

11. IIM = Inner Iteration Maximum (10)
12. IUP = Upscatter Outer Iterations in Thermal Range (3) [see Note 2]
13. NFST = Multigroup Calculation Option in Upper MG Range [E>DEMAX] (3)

    0/1/2/3/4/5/6 = S_N / diffusion / homogenized infinite medium /

    zonewise infinite medium/(option 5 deprecated)/2D MoC cell

    [see Note 3]

14. NTHR = Multigroup Calculation Option in Lower MG Range [E<DEMIN] (3)

    0/1/2/3/4/5/6 = S_N / diffusion / homogenized infinite medium /

    zonewise infinite medium/(option 5 deprecated)/2D MoC cell

    [see Note 3]

15. NPXS = Calculation Option in PW Range [DEMIN<E<DEMAX] (4)

    0/1/2/3/4/5/6 = none (do NFST multigroup calculation) / S_N / collision-probability/homogenized infinite medium /zonewise infinite medium/ 2-region/

    2D MoC cell method of characteristics (only for square lattice cells)

    [see Note 3]

16. ISVAR = Multigroup Source & Cross-Section Linearization Option (3)

    0/1/2/3 = none /linearize MG source /linearize MG XS's /both [see Note 4]

17. mocMesh = Mesh Options for MoC Calculation; npxs=6 only:

    0/1/2 = coarse/regular/fine mesh/input by zone (0)

18. mocPol = Number of Polar Angles for MoC Quadrature; npxs=6 only:

    2/3/4 = allowable values (3)

19. mocAzi = Number of Azimuthal Angles for MoC Quadrature; npxs=6 only:

    1 -> 16 = allowable values (8)

20. kern = Thermal Neutron Scattering Treatment:

    0/1 = all free-gas kernels/use bound S(alpha, beta) data if available (1)

21. ISCTI = P_N Order of Scattering for PW inelastic [<= ISCT] (1)

22. NMF6 = PW Inelastic Scatter Option (-1)

    -1/0/1 = no inelastic/ discrete level inelastic/ discrete and continuum

2$ $ EDITING AND OTHER OPTIONS (12 entries. Defaults shown in parenthesis)

1. IPRT = Mixture Cross-Section Print Option: (-3)

   -3/-2/-1/N = none / write PW macro cross sections to output file, in tab1 format/

   print 1-D MG cross sections /print $P_0 \rightarrow P_N$ MG scatter matrices

2. ID1 = Flux Print/Punch Options (-1)

   -1/0/1/2/ = none / print MG flux / also print M.G moments

   /save PW zone-average flux in ascii file

3. IPBT = Balance Tables Print Option (PW thermal B.T. edit not functional) (0)

   0/1 = none / print balance tables

4. IQM = Use Volumetric Sources: 0/N = No / Yes (0)
5. IPM = Use Boundary Source: 0/N = No / Yes (0)
6. IPN = Group Diffusion Coefficient Option (2)

   0/1/2 = [*see XSDRNPM input*]

7. IDFM = Use Density Factors 0/1 = No / Yes (0)
8. IXPRT = Extra Print Option:

0/1 = minimum print /regular print (0)

9. MLIM = Mass Value Restriction on Order of Scattering (100)

   0/M = no effect / restrict nuclides with mass *geq* M to have scatter

   order *leq* NLIM

10. NLIM = Restrictive Scatter Order (0)

3** CONVERGENCE CRITERIA AND OTHER CONSTANTS (9 entries. Defaults in parenthesis)

1. EPS = Upscatter Integral Convergence Criterion (0.001)
2. PTC = Point Convergence Criterion (0.001) [see Note 5]
3. XNF = Source Normalization Factor (1.0) [see Note 1]
4. B2 = Material Buckling Value [units of cm^{-2}] (0.0)
5. DEMIN = Lowest Energy of Pointwise Flux Calculation, in eV (0.001)
6. DEMAX = Highest Energy of Pointwise Flux Calculation, in eV (2.0E4)
7. TOLE = Tolerance Used in Thinning Pointwise Cross Sections (0.001) [see Note 7]
8. MOCRAY = Distance (cm) Between MoC Rays; only for npxs=6 (0.02)
9. FLET = Lethargy-Gain Fraction For Determining Energy Mesh (0.1) [see Note 7]
10. ALUMP = 0.0→1.0, Criterion for Lumping of Materials by Mass (0.0) [see Notes]

T *[TERMINATE DATA BLOCK 1]*

DATA BLOCK 2 : MIXING TABLE

12$ $ COMPOSITION NUMBERS [AS IN MG LIBRARY] (MS entries)

13$ $ MIXTURE NUMBERS (MS entries)

14$ $ NUCLIDE IDENTIFIERS [AS IN MG LIBRARY] (MS entries)

[*Note on 14$ $* : Negative entry excludes material from PW treatment; i.e., MG data used]

15** NUCLIDE CONCENTRATIONS (MS entries)

T *[TERMINATE DATA BLOCK 2]*

DATA BLOCK 3: SOURCE DATA

30$ $ SOURCE NO. BY INTERVAL (IM entries, if IQM or IPM > 0)

31** VOLUMETRIC MULTIGROUP SOURCE SPECTRA (IQM*IGM entries, if IQM >0)

32** MULTIGROUP BOUNDARY ANGULAR SOURCE SPECTRA (IPM*IGM*MM, if IPM >0)

34** SPACE-DEPENDENT FISSION SOURCE (IM entries, if ISRC > 0)

T *[TERMINATE DATA BLOCK 3]*

DATA BLOCK 4: OTHER INPUT ARRAYS

35** INTERVAL BOUNDARIES (IM+1 entries)

36$ $ ZONE NUMBER BY INTERVAL (IM entries)

38** DENSITY FACTORS (IM entries, if IDFM>0)

39$ $ MIXTURE NUMBER BY ZONE (IZM entries)

41** TEMPERATURE [kelvin] BY ZONE (IZM entries: Default = F300.0)

47** RIGHT BOUNDARY ALBEDOS (IGM entries if IBR=3: Default = F1.0)

48** LEFT BOUNDARY ALBEDOS (IGM entries if IBL=3: Default = F1.0)

49** DANCOFF FACTOR BY ZONE (IZM entries: Default = 0.0)

T *[TERMINATE DATA BLOCK 4]*

******* END OF CENTRM INPUT DATA *******

7.4.4.1 CENTRM data notes

1. If ISRC = 0, an external volumetric or boundary source is used, as specified in the 30$$, 31**, and 32** arrays (these are similar to same arrays in XSDRNPM). This option is only available for standalone CENTRM calculations; it is not an option for XSProc execution. If ISRC=1, the magnitude of the fission source density (neutrons/cm^3-s) is input in the 34** array, and the source energy spectrum is assumed to be χchi`=0 regardless of value in 34** array). Both external and fission sources may be used if ISRC = 2. The source normalization parameter XNF in the 3* array applies to the combined sources specified in data block 3.

2. If the value of DEMIN in the 3* array is set lower than the thermal cutoff energy of the AMPX MG library, a PW thermal calculation is performed over the energy range between DEMIN and the thermal cutoff. A MG thermal calculation is performed over the remainder of the thermal range. PW thermal calculations always start with ten outer iterations of MG theory to obtain an initial flux guess. The value "IUP" indicates how many *additional* outer iterations are to be performed.

3. The solutions in the upper and lower MG ranges, as well as the PW solution, provide several calculational options in addition to the default method. These are described in Sect. 7.4.3.

4. The parameter ISVAR indicates how MG values for sources and cross sections are mapped onto the PW energy mesh. See Sect. 7.4.2.8.

5. "Point Convergence" refers to the worst convergence over all space intervals, between successive iterations. It is applied to (a) inner iterations for MG S_N solution; (b) inner iterations of doubly-reflected boundary conditions in PW S_N solution; and (c) outer iterations of MG and PW solution in thermal range.

6. The values for DEMAX and DEMIN determine the energy range of the PW calculation. If the purpose of the CENTRM calculation is to obtain PW fluxes for resonance self-shielding calculations with PMC, then the PW energy range should at least span the resolved resonance ranges of all materials for which self shielding is significant.

7. The energy mesh for the PW flux calculation is based on several factors, as described in Sect. 7.4.2.7.

8. If parameter alump > 0, epithermal and thermal scattering sources for individual nuclides with similar masses are combined into macroscopic "lumps," based on a fractional mass deviation of "alump." For example, alump=0.2 means that materials are combined into one or more lumps such that their masses are within +/- 20% of the effective mass of the lump. The effective mass of the lump is defined to preserve the macroscopic slowing down power. Mass lumping up to 0.2 will often reduce execution time with little impact on the results.

7.4.4.2 CENTRM I/O files

Table 7.4.8 shows filenames used by CENTRM. Some files are not required for some calculations.

Table 7.4.8: Default File Names

File Name	Description
ft04f001	Input MG library (only for standalone execution)
ft81f001	Input PW cross-section library from Crawdad PW
lib_cen_kern	Input PW thermal scatter kernels from Crawdad
_centrm.pw.flux	Output PW flux by zone (only for standalone)
_centrm.pw_macro_xs	Output PW macro cross-section (optional)

7.4.4.3 Description of the CENTRM CE cross section file

The CENTRM CE cross section library is typically created using the CRAWDAD module. CRAWDAD is executed automatically during XSProc cases; but when CENTRM run standalone, the CE library must be pre-generated prior to the CENTRM calculation (e.g., by running CRAWDAD). CRAWDAD reads the SCALE CE data files for individual nuclides, and creates a combined CENTRM library in the binary format shown below.

Table 7.4.9: Header Records.

Record No.	Parameters	Description
1	IDTAPE	Tape Identifier.
1	NNUC	Number of Nuclides on pointwise file.
1	MTMP	Maximum Number of Temperatures for any Nuclide on file.
1	MAXMT	Maximum Number of Reactions for any Nuclide on file.
1	MAXPT	Maximum Number of Energy Points for any Nuclide (any MT) on File.
1	IX(10)	Unused for Now.
1	TITLE	Title (Character*72)
2	NZAPT(NNUC)	ZA's for Each Nuclide.
3	IDPT(NNUC)	MAT ID's for Each Nuclide.
4	NTEM(NNUC)	Number of Temperatures for Each Nuclide.
5	EMAX(NNUC)$^{(DP)}$	Highest Energy of Pointwise Data for Each Nuclide.
6	EMIN(NNUC)$^{(DP)}$	Lowest Energy of Pointwise Data for Each Nuclide.
7	EUPUR(NNUC)$^{(DP)}$	Upper Energy of Unresolved range.
8	EUPRR(NNUC)$^{(DP)}$	Upper Energy of Resolved range
9	ELORR(NNUC)$^{(DP)}$	Lower Energy of Resolved range.
10	MPT(NNUC)	Number of Energy Points for Total Cross Sections of Each Nuclide.
11	MPS(NNUC)	Number of Energy Points for Elastic Scatter Cross Sections of Each Nuclide.
12	NFIRST(NNUC)	Starting Record Number of Data for Each Nuclide.
13	MPIN(NNUC)	Number of inelastic levels for Each Nuclide.
14	POTXS(NNUC)	Potential Cross Sections for Each Nuclide.
15	MAX_INEL(NNUC)	Maximum Number of Energy Points for any Inelastic Discrete or Continuum Level.
16	NTHERM(NNUC)	Thermal Scattering Kernel IDs for Each Nuclide (0 = free gas).
17	Free	Not Used

Table 7.4.10: Nuclide Dependent Records (repeat for each nuclide)

Parameters	Description
DTEXT	Descriptive Text (Character*72).
ID	Nuclide ID.
IZA	Nuclide ZA.
NMT	Total Number of MT's.
(NTOMT(K),K=1,NMT)	Number of Temperatures.
NRRP	Number of Resolved Resonances Processed.(not used)
NTOTR	Total Number of Records for This Nuclide.
IX(6)	Unused for Now.
(MT(N),N=1,NMT)	MT's for This Nuclide.
((TEMP(M,MN), →	Temperatures for This Nuclide.
M=1, NTOMT(MN)), MN=1,NMT))	
(NPT(J),J=1,NMT)	Number of Energy Points for Each MT.
[ENERGY POINTS : NMT Records]	
DO 1 N = 1 , NMT	
1 (E(L)$^{(DP)}$, L = 1 , NPT(N))	
*[POINT XS VALUES : NMT * NT Records]*	
DO 2 N = 1 , NMT	
DO 2 M = 1 , NTOMT(N)	
NUMBER = NPT(N)	
2 (XS(L)$^{(DP)}$, L = 1, NPT(N))	
$^{(DP)}$ *DOUBLE PRECISION ARRAYS*	

7.4.4.4 Description of the CENTRM output PW flux file

Table 7.4.11: 3 Header Records Described Below.

Record No.	Parameters	Description
1	IGE	Type of Geometry
1	MMT	Number of Neutron Groups
1	JT	Number of Flux Moments
1	IZM	Number of Zones
1	IZMT	Total Number of Nuclides in all Zones
1	IM	Number of Space Intervals
1	IBR	Right Boundary Condition
1	IBL	Left Boundary Condition
1	MGHI	Lowest Energy Group in UMR
1	MGLO	Highest Energy Group in LMR
1	MT	Number of Materials in Problem
1	IFTG	First Thermal Group

continues on next page

Table 7.4.11 – continued from previous page

Record No.	Parameters	Description
1	NTOT	Number of Points in PW Energy Mesh
1	NTOTP	Number of Points in Full Energy Mesh (UMR+PW+LMR)
1	MULT	Flag for Double or Single Precision Word Length
1	IX (10)	Dummy array of 10 integers
1	DEMAX$^{(DP)}$	Upper Energy of PW Calculation
1	DEMIN$^{(DP)}$	Lowest Energy of PW Calculation
1	ETHRM$^{(DP)}$	Energy Corresponding to Thermal Cutoff
1	RX (10)	Dummy Array of 10 Single-Precision Real Numbers
2	NIDS(MT)	Nuclide IDs
2	NZA(MT)	Nuclide ZA Numbers
2	IR-CUM(IZM)	Cumulative Number of Nuclides in Each Zone
2	MBYZ(IZMT)	List of All Nuclides in All Zones
2	ZTEMP(IZM)	Zone Temperatures
2	RD(IZMT)	Number Densities of all Nuclides in All Zones
2	R(IM)	Interval Volumes
2	MA(IM)	Zone Number by Interval
2	DF(IM)	Density Factor
3	DEN(IGP)$^{(DP)}$	Group Energy Boundaries
3	E(NTOTP)$^{(DP)}$	Energies Corresponding to PW Energy Mesh
3	U(NTOTP)$^{(DP)}$	Lethargies Corresponding to PW Energy Mesh
[IZM Records Containing Zone-Averaged PW Fluxes and Moments]		
DO 1 N = 1 , IZM		
1 PXJ(NTOTP , M), M=1, JT+1)		
$^{(DP)}$ DOUBLE PRECISION ARRAYS		

7.4.5 EXAMPLE CASE

In this section an example standalone CENTRM calculation is demonstrated for a 1-D slab geometry model of a highly enriched uranium solution with an iron-56 reflector. To execute CENTRM in standalone mode, nuclear data libraries must be provided for MG cross sections (file ft04f001), PW cross sections (file ft81f001), and PW thermal scatter kernels (lib_cen_kernel). The shell script (=shell) in front of the CENTRM input is used to link the necessary nuclear data libraries to the CENTRM calculation. Here it is assumed that the Crawdad module has been previously run to produce the PW cross section and thermal kernel libraries, which are located in the same directory from which that the job is submitted, while the MG library is the 8 group test library from the SCALE data directory.

7.4.5.1 CENTRM input for example case

```
=shell
  ln -sf /scale/scale_data/test8g_v7.1 ft04f001
  ln -sf  $RTNDIR/ft81f001 ft81f001
  ln -sf  $RTNDIR/lib_cen_kernel lib_cen_kernel
end

=centrm
centrm standalone example
1$$ 1 2 15 1 0 2 5 e  2$$  -3 0 a8 1 e
3**  a5  0.0001 4000.0 e
t
12$$  1 1 1 1 2   13$$  1 1 1 1 2
14$$  92235 1001 8016 7014 26056
15**  0.003 0.06 0.04 0.018 0.08
t
34**  f1.0    t
35** 9i 0.0  4i 20.0 30.0  36$$ 10r1 5r2  39$$ 1 2  41** f300.0
t
end
```

7.4.5.2 CENTRM output for example case

```
*****************************************************************


    module shell will be called on Tue Mar 15 16:52:40 2016.
    sequence specification record:=shell

Input Data:
      ln -sf /scale/scale_dev_data/test8g_v7.1 ft04f001
      ln -sf $RTNDIR/ft81f001 ft81f001
      ln -sf $RTNDIR/lib_cen_kernel lib_cen_kernel
end

module shell is finished. completion code 0

    module centrm will be called on Tue Mar 15 16:52:40 2016.
    sequence specification record:=centrm

Input Data:
      centrm standalone example
      1$$ 1 2 15 1 0 2 5 e  2$$  -3 0 a8 1 e
      3**  a5  0.0001 4000.0 e
      t
      12$$  f0
      13$$  1 1 1 1 2
      14$$  92235 1001 8016 7014 26056
      15**  0.003 0.06 0.04 0.018 0.08
      t
      34**  f1.0 t
      35** 9i 0.0  4i 20.0 30.0
      36$$ 10r1 5r2  39$$ 1 2  41** f300.0
      t
end
```

```
1$ array     18 entries read

     2$ array     12 entries read

     3* array      9 entries read

     0t
```

(continues on next page)

(continued from previous page)

```
   12$ array      5 entries read

   13$ array      5 entries read

   14$ array      5 entries read

   15* array      5 entries read

    0t

   34* array     15 entries read

    0t

   35* array     16 entries read

   36$ array     15 entries read

   39$ array      2 entries read

   41* array      2 entries read

    0t

    CENTRM MATERIALS

       nuclides on                                                  Mixing Table
       multi-grp lib.  composition              mixture   component   atom density      small
    1     92235           0                        1        92235     3.00000E-03         1
    2      1001           0                        1         1001     6.00000E-02         1
    3      8016           0                        1         8016     4.00000E-02         1
    4      7014           0                        1         7014     1.80000E-02         1
    5     26056           0                        2        26056     8.00000E-02         1

  elapsed time .00 min.
  =time after return from setup_centrm.

    ft81f001           =   pointwise XS library from Crawdad
```

```
MULTIGROUP STRUCTURE

igm      =        8      number of energy groups
mmt      =        8      number of neutron groups
mcr      =        0      number of gamma groups
iftg     =        5      first thermal group number


GENERAL PROBLEM DATA

ige      =        1     problem geometry
                         0/1/2/3 = inf. hom. medium/plane/cylinder/sphere
izm      =        2     number of zones
im       =       15     number of spatial intervals
ibl      =        1     left boundary condition:
                         0/1/2/3 = vacuum/reflected/periodic/albedo
ibr      =        0     right boundary condition:
                         0/1/2/3 = vacuum/reflected/periodic/albedo
mxx      =        2     number of mixtures
ms       =        5     mixing table length
isn      =        8     SN quadrature order (not used for npxs=6)
isct     =        3     order of elastic scattering
isrc     =        1     type of source spectrum:
                         0/1/2 = multigroup input spectrum/fission spectrum/both (1)
iim      =       20     inner iterations maximum (10)
```

(continues on next page)

(continued from previous page)

```
iup      =        3     upscatter outer iterations in thermal range (3)
nfst     =        0     multigroup calculation option in upper energy range [option nfst=5 is deprecated]:
                         0/1/2/3/4/6 = sn/diffusion/homogeneous/zonewise homogeneous/BN/2D MoC cell (0)
nthr     =        0     multigroup calculation option in lower energy range [option nthr=5 is deprecated]:
                         0/1/2/3/4/6 = sn/diffusion/homogeneous/zonewise homogeneous/BN/2D MoC cell (0)
npxs     =        1     pointwise calculation option:
                         <=0/1/2/3/4/5/6 = no pointwise: do multigroup as in nfst/sn/coll. prob./
                           homogeneous/zonewise homogeneous/zonewise two-region/2D MoC cell (1)
isvar    =        3     multigroup source & cross section linearization option :
                         0/1/2/3 = none/linearize source/linearize group cross sections/both (3)
mocMesh  =    *****     meshing options for MoC calculation; npxs=6 only:
                         0/1/2   = coarse/regular/fine mesh/input by zone (0)
mocPol   =        0     number of polar angles for MoC quadrature; npxs=6 only:
                         2/3/4 = allowable values        (3)
mocAzi   =        0     number of azimuthal angles for MoC quadrature; npxs=6 only:
                         1 -> 16 = allowable values      (8)
kern     =        0     thermal neutron scattering treatment:
                         0/1 = all free-gas thermal kernels/use bound S(alpha, beta) data if available (0)
iscti    =        1     inelastic scattering order (0)
nmf6     =        0     inelastic scattering option:
                         -1/0/1 = no inelastic/discrete level only/discrete+continuum (0)
```

```
EDITTING AND OTHER OPTIONS

iprt     =       -3     -3/-2/-1/n = mixture cross sections print options:
                          none/write P.W. energy & macro x-sections to output file/print 1-D M.G. x-sections
                          /print nth p (p0,p1..) M.G. x-section matrix (-3)
id1      =        0     -1/0/1/2 = flux print options:
                           none/print M.G flux/also print M.G moments/save PW zone flux in ascii file
ipbt     =        0     0/1 = none/group summary table print (0)
iqm      =        0     input multigrp volumetric sources (0/n=no/yes)
ipm      =        0     input multigrp boundary angular sources (0/n=no/yes)
ipn      =        2     0/1/2 diff. coef. param (2)
idfm     =        0     0/1 = none/density factors (0)
ixprt    =        1     0/1 = minimum print/regular print (0)
mlim     =      100     0/m = no effect/restrict nuclides with mass >= m to have PW scatter order <= nlim ␣
↪(100)
nlim     =        3     restrictive scatter order (1)


FLOATING POINT VALUES

eps      =     0.10000E-03    upscatter integral convergence (0.001)
ptc      =     0.10000E-03    inner iteration convergence (0.001)
xnf      =     0.10000E+01    source normalization factor (1.0)
b2       =     0.00000E+00    buckling value ( cm**(-2) )
demin    =     0.10000E-03    lowest energy of pointwise calculation in (eV)
demax    =     0.40000E+04    highest energy of pointwise calculation in (eV)
tole     =     0.10000E-02    tolerance used in thinning the pointwise cross sections (0.001)
mocRay   =     0.00000E+00    distance between MoC rays; npxs=6 only. (0.02)
flet     =     0.10000E+00    fractional lethargy used in construction of flux energy mesh (0.1)
alump    =     0.00000E+00    mass-lumping criterion (0)
```

```
.....................................................................................................

  Input DEMIN and DEMAX values may be modified to lie on multigroup boundaries.
    Min and Max energies (eV) defining range for pointwise flux calculation  =     4.000E-02    2.000E+04
 .....................................................................................................
1
                            POINTWISE MATERIALS USED IN THIS PROBLEM

         **** ZONE NO.   1
              Nuclide ID     No. of Micro XS Energy Points in Energy Range           0.040 to        20000.000
                   1001              260
```

(continues on next page)

(continued from previous page)

```
                7014             604
                8016             448
               92235          118346
                          No. of Macro XS Energy Points for Zone (After Thinning):    7133

       **** ZONE NO.   2
            Nuclide ID     No. of Micro XS Energy Points in Energy Range          0.040 to        20000.000
                26056             1379
                          No. of Macro XS Energy Points for Zone (After Thinning):     363


                          NUMBER OF POINTS IN FINAL ENERGY MESH FOR FLUX CALCULATION     12687
                          NUMBER OF POINTS IN PW THERMAL RANGE                             458


         No unit supplied for PW kinematics data.
         All PW thermal scatter kernels are treated as Free Gas
```

```
NEUTRON MULTIGROUP PARAMETERS


   gp    energy      lethargy      mid pt      pointwise      calc                      right        left
       boundaries   boundaries   velocities     flx pts       type                     albedo       albedo
    1  2.00000E+07 -6.93147E-01  2.51321E+09          0           0
    2  8.20000E+05  2.50104E+00  4.31276E+08          0           0
    3  2.00000E+04  6.21461E+00  4.01867E+07       8332           1
    4  1.05000E+02  1.14641E+01  6.03421E+06       3893           1
    5  5.00000E+00  1.45087E+01  1.78251E+06        238           1
    6  6.50000E-01  1.65489E+01  5.78908E+05        134           1
    7  1.50000E-01  1.80152E+01  3.45722E+05         85           1
    8  4.00000E-02  1.93370E+01  1.55701E+05          0           0
    9  1.00000E-05  2.76310E+01


             DESCRIPTION OF ZONES

         mixture   temperature  Dancoff factr
         by zone     by zone       by zone
    1         1     3.00000E+02       0
    2         2     3.00000E+02       0


                  SN QUADRATURE CONSTANTS

         weights    directions   refl direc    wt x cos
    1       0      -1.00000E+00         9          0
    2  8.69637E-02 -9.30568E-01         9    -8.09257E-02
    3  1.63036E-01 -6.69991E-01         8    -1.09233E-01
    4  1.63036E-01 -3.30009E-01         7    -5.38035E-02
    5  8.69637E-02 -6.94318E-02         6    -6.03805E-03
    6  8.69637E-02  6.94318E-02         5     6.03805E-03
    7  1.63036E-01  3.30009E-01         4     5.38035E-02
    8  1.63036E-01  6.69991E-01         3     1.09233E-01
    9  8.69637E-02  9.30568E-01         2     8.09257E-02

 CONSTANTS FOR P( 3) SCATTERING IN SN CALCULATION

 angl    set  1       set  2       set  3
     1   -1.000E+00    1.000E+00   -1.000E+00
     2   -9.306E-01    7.989E-01   -6.187E-01
     3   -6.700E-01    1.733E-01    2.531E-01
     4   -3.300E-01   -3.366E-01    4.052E-01
     5   -6.943E-02   -4.928E-01    1.033E-01
     6    6.943E-02   -4.928E-01   -1.033E-01
```

(continues on next page)

(continued from previous page)

```
     7    3.300E-01   -3.366E-01   -4.052E-01
     8    6.700E-01    1.733E-01   -2.531E-01
     9    9.306E-01    7.989E-01    6.187E-01

   elapsed time .02 min.
   =time at after return from drtran... calling calc.
```

```
GEOMETRY DESCRIPTION

1  int     radii       mid pts     zone no.       areas       volumes     dens fact
1       0       1.00000E+00         1     1.00000E+00  2.00000E+00
2  2.00000E+00  3.00000E+00         1     1.00000E+00  2.00000E+00
3  4.00000E+00  5.00000E+00         1     1.00000E+00  2.00000E+00
4  6.00000E+00  7.00000E+00         1     1.00000E+00  2.00000E+00
5  8.00000E+00  9.00000E+00         1     1.00000E+00  2.00000E+00
6  1.00000E+01  1.10000E+01         1     1.00000E+00  2.00000E+00
7  1.20000E+01  1.30000E+01         1     1.00000E+00  2.00000E+00
8  1.40000E+01  1.50000E+01         1     1.00000E+00  2.00000E+00
9  1.60000E+01  1.70000E+01         1     1.00000E+00  2.00000E+00
10  1.80000E+01  1.90000E+01         1     1.00000E+00  2.00000E+00
11  2.00000E+01  2.10000E+01         2     1.00000E+00  2.00000E+00
12  2.20000E+01  2.30000E+01         2     1.00000E+00  2.00000E+00
13  2.40000E+01  2.50000E+01         2     1.00000E+00  2.00000E+00
14  2.60000E+01  2.70000E+01         2     1.00000E+00  2.00000E+00
15  2.80000E+01  2.90000E+01         2     1.00000E+00  2.00000E+00
16  3.00000E+01                           1.00000E+00
1

EXTERNAL AND FISSION SOURCE DATA

int     radii       mid pts   fixd src spec          normalized fiss src
1       0       1.00000E+00                          5.00000E-02
2  2.00000E+00  3.00000E+00                          5.00000E-02
3  4.00000E+00  5.00000E+00                          5.00000E-02
4  6.00000E+00  7.00000E+00                          5.00000E-02
5  8.00000E+00  9.00000E+00                          5.00000E-02
6  1.00000E+01  1.10000E+01                          5.00000E-02
7  1.20000E+01  1.30000E+01                          5.00000E-02
8  1.40000E+01  1.50000E+01                          5.00000E-02
9  1.60000E+01  1.70000E+01                          5.00000E-02
10  1.80000E+01  1.90000E+01                           5.00000E-02
11  2.00000E+01  2.10000E+01                                0
12  2.20000E+01  2.30000E+01                                0
13  2.40000E+01  2.50000E+01                                0
14  2.60000E+01  2.70000E+01                                0
15  2.80000E+01  2.90000E+01                                0
16  3.00000E+01
1
```

```
MULTIGROUP SCALAR FLUX (P0 MOMENT)

  int.    grp.  1      grp.  2      grp.  3      grp.  4      grp.  5      grp.  6      grp.  7      grp.  8
     1  1.49464E-01  7.16136E-02  1.97417E-01  7.40893E-02  3.79816E-02  1.46041E-02  5.93163E-03  4.53285E-03
     2  1.49408E-01  7.16093E-02  1.97385E-01  7.40770E-02  3.79734E-02  1.46035E-02  5.98650E-03  4.79191E-03
     3  1.49422E-01  7.15720E-02  1.97300E-01  7.40356E-02  3.79569E-02  1.45962E-02  5.95732E-03  4.61562E-03
     4  1.49164E-01  7.15343E-02  1.97118E-01  7.39612E-02  3.79044E-02  1.45759E-02  5.95761E-03  4.72210E-03
     5  1.49106E-01  7.13917E-02  1.96726E-01  7.37835E-02  3.78199E-02  1.45410E-02  5.94751E-03  4.64476E-03
     6  1.48182E-01  7.11713E-02  1.95901E-01  7.33956E-02  3.75685E-02  1.44420E-02  5.89064E-03  4.64109E-03
     7  1.47512E-01  7.05913E-02  1.93957E-01  7.24791E-02  3.71327E-02  1.42704E-02  5.84992E-03  4.59489E-03
     8  1.43892E-01  6.93993E-02  1.89432E-01  7.09230E-02  3.61956E-02  1.39005E-02  5.64907E-03  4.43077E-03
     9  1.38788E-01  6.61071E-02  1.82365E-01  6.80018E-02  3.47040E-02  1.33102E-02  5.46977E-03  4.33076E-03
    10  1.14667E-01  5.81119E-02  1.75350E-01  6.17556E-02  3.03161E-02  1.13955E-02  4.57365E-03  3.58528E-03
    11  7.10033E-02  4.72163E-02  1.70191E-01  5.14605E-02  2.15414E-02  7.71994E-03  2.57498E-03  1.07163E-03
    12  4.41942E-02  3.58651E-02  1.47094E-01  3.99690E-02  1.38534E-02  4.46652E-03  1.23499E-03  1.02896E-04
```

(continues on next page)

(continued from previous page)

```
    13  3.07018E-02  2.56932E-02  1.17584E-01  2.91750E-02  9.12693E-03  2.42263E-03  6.13347E-04  4.02885E-05
    14  2.00333E-02  1.74870E-02  8.62759E-02  1.88824E-02  5.53088E-03  1.18988E-03  2.76717E-04  1.54836E-05
    15  1.25992E-02  9.76450E-03  5.01845E-02  8.56130E-03  2.43234E-03  4.58629E-04  1.02719E-04  5.89663E-06

   elapsed time .03 min.
   =time at end of centrm_Execute.

    module centrm used 2.34 seconds cpu time for the current pass.

    module centrm is finished. completion code    0. total cpu time used 0 seconds.

    SCALE is finished on Tue Mar 15 16:52:42 2016.

------------------------- Summary -------------------------
shell finished. used 0.02 seconds.
centrm finished. used 2.34 seconds.
----------------------- End Summary -----------------------
```

7.4.6 CENTRM PW LIBRARY AND FLUX FILE FORMATS

Sect. 7.4.6.1 below describes the format for the input CENTRM PW data library (binary). The subsequent Sect. 7.4.6.2, describes the format of the output PW flux file (binary) produced by CENTRM, which is input to the PMC code.

7.4.6.1 CENTRM CE cross section file

The CENTRM CE cross section library is typically created using the CRAWDAD module in SCALE. CRAWDAD reads the SCALE CE data files for individual nuclides, and creates the combined CENTRM library.

Table 7.4.12: 17 Header Records described below.

Record No.	Parameters	Description
1	IDTAPE	Tape Identifier.
1	NNUC	Number of Nuclides on PW file.
1	MTMP	Maximum Number of Temperatures for any Nuclide on file.
1	MAXMT	Maximum Number of Reactions for any Nuclide on file.
1	MAXPT	Maximum Number of Energy Points for any Nuclide (any MT) on File.
1	IX(10)	Unused for Now.
1	TITLE	Title (Character*72)
2	NZAPT(NNUC)	ZA's for Each Nuclide.
3	IDPT(NNUC)	MAT ID's for Each Nuclide.
4	NTEM(NNUC)	Number of Temperatures for Each Nuclide.
5	EMAX(NNUC)$^{(DP)}$	Highest Energy of Pointwise Data for Each Nuclide.
6	EMIN(NNUC)$^{(DP)}$	Lowest Energy of Pointwise Data for Each Nuclide.
7	EUPUR(NNUC)$^{(DP)}$	Upper Energy of Unresolved range.
8	EUPRR(NNUC)$^{(DP)}$	Upper Energy of Resolved range
9	ELORR(NNUC)$^{(DP)}$	Lower Energy of Resolved range.
10	MPT(NNUC)	Number of Energy Points for Total Cross Sections of Each Nuclide.
11	MPS(NNUC)	Number of Energy Points for Elastic Scatter Cross Sections of Each Nuclide.
12	NFIRST(NNUC)	Starting Record Number of Data for Each Nuclide.
13	MPIN(NNUC)	Number of inelastic levels for Each Nuclide.
14	POTXS(NNUC)	Potential Cross Sections for Each Nuclide.
15	MAX_INEL(NNUC)	Maximum Number of Energy Points for any Inelastic Discrete or Continuum Level.
16	NTHERM(NNUC)	Thermal Scattering Kernel IDs for Each Nuclide (0 = free gas).
17	Free	Not Used

Table 7.4.13: [NUCLIDE DIRECTORY : 1 RECORD/NUCLIDE].

Parameters	Description
DTEXT	Descriptive Text (Character*72).
ID	Nuclide ID.
IZA	Nuclide ZA.
NMT	Total Number of MT's.
(NTOMT(K),K=1,NMT)	Number of Temperatures.
NRRP	Number of Resolved Resonances Processed.(not used)
NTOTR	Total Number of Records for This Nuclide.
IX(6)	Unused for Now.
(MT(N),N=1,NMT)	MT's for This Nuclide.
((TEMP(M,MN), →	Temperatures for This Nuclide.
M=1, NTOMT(MN)), MN=1,NMT))	
(NPT(J),J=1,NMT)	Number of Energy Points for Each MT.
[ENERGY POINTS : NMT Records]	
DO 1 N = 1 , NMT	
1 (E(L)$^{(DP)}$, L = 1 , NPT(N))	
*[POINT XS VALUES : NMT * NT Records]*	
DO 2 N = 1 , NMT	
DO 2 M = 1 , NTOMT(N)	
NUMBER = NPT(N)	
2 (XS(L)$^{(DP)}$, L = 1, NPT(N))	

$^{(DP)}$ *DOUBLE PRECISION ARRAYS*

7.4.6.2 CENTRM output PW flux file

continues on next page

Table 7.4.14 – continued from previous page

Table 7.4.14: [3 Header Records Described Below].

Record No.	Parameters	Description
1	IGE	Type of Geometry
1	MMT	Number of Neutron Groups
1	JT	Number of Flux Moments
1	IZM	Number of Zones
1	IZMT	Total Number of Nuclides in all Zones
1	IM	Number of Space Intervals
1	IBR	Right Boundary Condition
1	IBL	Left Boundary Condition
1	MGHI	Lowest Energy Group in UMR
1	MGLO	Highest Energy Group in LMR
1	MT	Number of Materials in Problem
1	IFTG	First Thermal Group
1	NTOT	Number of Points in PW Energy Mesh
1	NTOTP	Number of Points in Full Energy Mesh (UMR+PW+LMR)
1	MULT	Flag for Double or Single Precision Word Length
1	IX (10)	Dummy array of 10 integers
1	DEMAX$^{(DP)}$	Upper Energy of PW Calculation
1	DEMIN$^{(DP)}$	Lowest Energy of PW Calculation
1	ETHRM$^{(DP)}$	Energy Corresponding to Thermal Cutoff
1	RX (10)	Dummy Array of 10 Single-Precision Real Numbers
2	NIDS(MT)	Nuclide IDs
2	NZA(MT)	Nuclide ZA Numbers
2	IRCUM(IZM)	Cumulative Number of Nuclides in Each Zone
2	MBYZ(IZMT)	List of All Nuclides in All Zones
2	ZTEMP(IZM)	Zone Temperatures
2	RD(IZMT)	Number Densities of all Nuclides in All Zones
2	R(IM)	Interval Volumes
2	MA(IM)	Zone Number by Interval
2	DF(IM)	Density Factor
3	DEN(IGP)$^{(DP)}$	Group Energy Boundaries
3	E(NTOTP)$^{(DP)}$	Energies Corresponding to PW Energy Mesh
3	U(NTOTP)$^{(DP)}$	Lethargies Corresponding to PW Energy Mesh

continues on next page

Table 7.4.14 – continued from previous page

[IZM Records Containing Zone-Averaged P.W. Fluxes and Moments, Described Below]

DO 1 N = 1 , IZM
1 PXJ(NTOTP , M), M=1, JT+1)
(DP) DOUBLE PRECISION ARRAYS

7.4.7 CENTRM ERROR MESSAGES

CENTRM prints several of the same error messages as the XSDRNPM code. Users should refer to the XSDRNPM chapter for these. CENTRM also prints the additional error messages shown below.

STATEMENT IN PRINTOUT	USER ACTION
ERROR. DEMIN exceeds maximum energy in working library STOP 400	Reduce input value of DEMIN
ERROR. DEMAX exceeds DEMIN STOP 401	Check input values for PW energy limits
***CALCULATION TERMINATED. No neutron groups in working library	Check AMPX working library
WARNING: Thermal Source is 0 in ACCEL **–OR-** WARNING: Thermal Absorption + Leakage is 0 in ACCEL	Check input source and thermal cross-section data (often caused by using Zonewise Infinite Medium option with no source in some zone)
Requests nuclide *XXX* which is not on your working library.	Check nuclide ID's and composition numbers in mixing table to make sure they are on MG library; Verify that PW and MG libraries are consistent

7.5 PMC: A PROGRAM TO PRODUCE MULTIGROUP CROSS SECTIONS USING POINTWISE ENERGY SPECTRA FROM CENTRM

M. L. Williams, D. F. Hollenbach, U. Merteryuk, and K. S. Kim

ABSTRACT

PMC generates problem-dependent multigroup cross sections from an existing multigroup cross-section library, a pointwise nuclear data library, and a pointwise neutron flux file produced by the CENTRM continuous-energy transport code. In the SCALE sequences, PMC is a computational module called from XSProc to produce self-shielded multigroup (MG) cross-sections over a specified energy range (e.g., resolved resonance range) of individual nuclides in the system of interest. The self-shielded cross sections are obtained by integrating the pointwise (PW) nuclear data using the CENTRM problem-specific, continuous-energy flux as a weight function for each spatial mixture in the system. Several options are available in PMC to specify various types of weighting methods for the one-dimensional and two-dimensional MG data. PMC outputs problem-dependent self-shielded cross sections that can be used in XSDRNPM, KENO, NEWT or other MG transport codes.

ACKNOWLEDGMENTS

The authors acknowledge the suggestions and direct contributions of L. M. Petrie of Oak Ridge National Laboratory, and former ORNL staff N. M. Greene, and R. M. Westfall.

7.5.1 INTRODUCTION

PMC is a computational module used for the CENTRM/PMC self-shielding method performed by the XSProc driver module. It can also be run in standalone mode. PMC (**P**roduce **M**ultigroup **C**ross sections) computes multigroup (MG) cross sections by utilizing the pointwise (PW) neutron spectra calculated in CENTRM [PMCWA95] to weight cross sections in a continuous-energy (CE) library file. This provides problem-dependent, self-shielded MG data representative of the fine-structure variation in the neutron energy spectrum for the system of interest. PMC only computes shielded cross sections within the energy interval of the CENTRM PW flux calculation, defined by the energy limits DEMIN and DEMAX. By default the lower limit is DEMIN=0.001 eV, and the upper energy is DEMAX=20,000 eV; however these parameters can be modified by the user in the CENTRM DATA input block. Outside of this energy interval, the shielded cross sections previously computed with the Bondarenko method in BONAMI are retained. PMC is automatically called from the XSProc driver module during execution of a SCALE sequence, and the resulting zone-averaged, problem-dependent cross sections can be passed to MG transport solvers (e.g., KENO, NEWT, XSDRNPM, etc.) called by the sequence.

7.5.1.1 Description of PMC input nuclear data

The nuclear data input to PMC consists of both MG and CE cross sections. Input MG data are obtained from the MG library specified in the sequence input. During SCALE execution, CE data used by both CENTRM and PMC are prepared by the code CRAWDAD, which reads CE files for individual nuclides, interpolates the data to the appropriate temperatures for the specified mixtures, and concatenates the data into a one problem-specific, multiple-nuclide CENTRM PW library. In general each nuclide has its own unique energy mesh defined such that the cross section at any energy value can be interpolated linearly from the library point data to accuracy better than 0.1%. Although cross sections in the original CE data files include values over the full energy range of 0-20 MeV, CRAWDAD reduces the energy range to interval of the CENTRM PW calculation (i.e., DEMIN→DEMAX). It is this combined PW library that is accessed by PMC. The format of the CENTRM PW library is described in CENTRM: A Neutron Transport Code for Computing Continuous-Energy Spectra in General One-Dimensional Geometries and Two-Dimensional Lattice Cells.

7.5.1.2 Description of PMC input pointwise flux data

In addition to the input nuclear data, PMC also requires PW flux values calculated in CENTRM to be provided. Depending on the CENTRM transport approximation, the flux data includes the PW scalar flux spectrum as a function of energy and spatial-zone, and also may include PW spherical harmonic moments of the angular flux (e.g., the current), which can be used in processing MG scattering matrices for higher-order Legendre moments. The non-uniform energy-mesh of the PW flux is determined during the CENTRM calculation in order to represent the spectrum variation with a minimum number of energy points. Like the CE cross section data, the flux spectrum at any energy value can be obtained within a specified tolerance by linear interpolation of the PW flux values.

7.5.2 CODE FEATURES

Two types of MG data are processed by PMC: 1-D cross sections and 2-D scatter matrices. The 1-D cross sections are weighted-average values over each energy group, by nuclide and reaction type. If there are "G" energy groups on the input library, then the 1-D cross section for each reaction type can be viewed as a 1-D vector with G values (of course some may be zero). Depending on the options and PW energy range specified, PMC will generally only re-compute and replace some of the G-group data. The 2-D cross sections correspond to group-to-group transfers (and corresponding Legendre moments) associated with various types of scatter reactions. These data can be arranged into a 2-D G by G matrix. For most materials this matrix is quite sparse. The 2-D data depend not only on the cross-section data, but also on the

energy/angular distributions of the secondary neutrons, which are represented by Legendre moments. PMC always re-normalizes the 2-D elastic and inelastic scattering matrices (including moments) to be consistent with the respective self-shielded 1-D data. In the case of elastic scattering, PMC also has rigorous options that can be used to modify the secondary energy distribution to account for self-shielding effects, such as by correcting the group removal cross section.

7.5.2.1 Options for treatment of 1-D cross sections

PMC computes new MG data for each reaction type (MT) and each nuclide on the input MG library, which also has CE data on the CENTRM PW library. Cross sections for reactions on the input MG library which do not have corresponding PW reaction data are not replaced; i.e., the original MG values are retained. SCALE CE library files for individual nuclides contain all reaction types included in the ENDF/B data; however the CRAWDAD module, executed prior to PMC, only includes certain ones when it produces the problem-specific CENTRM library. By default the CENTRM PW nuclear data library always includes cross sections for the total (MT-1), radiative capture (MT-102), and elastic scattering reactions (MT-2) of all nuclides; as well as fission (MT-18), and prompt, delayed, and total-nubar values (MTs-456, 455, 452, respectively) for fissionable nuclides. The (n,alpha) cross sections (MT-107) for B-10 and Li-6 are also always included if these nuclides are present in a mixture. If the CENTRM PW transport calculation includes the inelastic scattering option, indicated by CENTRM input parameter nmf6 >= 0, the discrete-level PW inelastic (MTs 50-90) and continuum inelastic (MT-91) data are also included in the CENTRM PW library.

PW data for the unresolved resonance range are infinitely dilute on the CENTRM library; therefore PMC does not use PW cross sections to compute self-shielded data for the unresolved range. Instead, self-shielded cross sections in the unresolved range are calculated using the Bondarenko method in BONAMI prior to the CENTRM and PMC calculations. This step is automatically performed by XSProc in the SCALE calculation sequences.

PMC offers two methods to compute the total cross section. In the first method the MG value for the total cross section (MT=1) is processed directly from the PW MT-1 data on the CENTRM library. Total cross sections are generally considered the most accurate type of evaluated reaction data (due to measurement techniques); however if PW data for MT-1 are processed as an independent cross section, there is no guarantee that the sum of the partial cross sections will sum to the total. These small imbalances in cross sections affect the neutron balance, and may impact eigenvalue calculations. For this reason the PMC default option does not compute the total cross section by weighting the MT-1 PW data, but rather by summing the MG partial cross sections (including the original MG data not re-processed in PMC).

The 1-D cross sections can be weighted using either the P_0 (scalar flux) or P_1 (current) PW Legendre moment. In almost all cases flux weighting is more desirable, since resonance reaction rates are usually the dominant factor in the PW range. However, current-weighting may be more accurate for certain problems where spatial transport and leakage strongly influence the spectrum in the resonance range, such as when the leakage spectrum is greatly impacted by cross section interference minima such as occur in iron media. The current-weighting option has been successfully applied for criticality calculations involving mixtures of highly-enriched uranium and iron. An alternative approach to using the current-weighted total cross section is to include a Legendre expansion of the angular-flux-weighted total cross section, which modifies the diagonal elements of the 2D elastic scattering moments.[7] This option is specified by setting PMC input parameter n2d=±2, as discussed in Sect. 7.5.2.4.

7.5.2.2 Spatial averaging of 1D cross sections

PMC computes MG microscopic cross sections for each material mixture in a given CENTRM calculation, using the spatially averaged PW spectrum within the mixture. In SCALE this method is called "zone-weighting", and it is the default for PMC. Zone-weighted cross sections are generated for every mixture zone in the unit cell. In configurations containing fuel/absorber mixtures (e.g., lattices) in multiple unit cells, CENTRM/PMC calculations may be performed for each mixture, resulting in multiple mixture-weighted cross sections for the same nuclide ID. For this reason, both the nuclide ID and a mixture number are generally required to uniquely identify any specific cross section data generated by PMC.

PMC also has an option to calculate "cell-weighted" (i.e., homogenized) MG data, which applies disadvantage factors to preserve the cell-averaged reaction rates for the entire unit cell. This is not typically done, except for treating doubly-heterogeneous cells with SCALE. In this case the PMC cell-weighting option is performed to produce homogenized MG cross sections for the low level heterogeneity (e.g., fuel grain in a fuel pebble). The XSProc control module automatically sets the correct PMC weighing flag based on the type of unit cell.

7.5.2.3 Energy ranges for multigroup weighting

The energy range of the MG and CE libraries in SCALE typically spans 10^{-5} to $2*10^{7}$ eV. In general this encompasses the (a) thermal region where upscatter is treated, (b) resolved and unresolved resonance ranges, and (c) high energy region above the resonance ranges. The thermal range for the current SCALE libraries is defined to be below 5 eV. Energy limits for the resolved and unresolved resonance ranges are defined by the individual ENDF/B evaluations for each nuclide, and these limits are included in the CENTRM PW library.

As discussed in section 8.3, the CENTRM PW flux file contains values of the zone-flux (and moments) per unit lethargy, calculated over the entire energy range 10^{-5} eV to 20 MeV; however, only the fluxes in the energy range from DEMAX to DEMIN are computed from the PW transport solution and exhibit the spectral fine-structure due to resonance reactions. The flux outside interval [DEMAX, DEMIN] is represented by the smoother "pseudo-pointwise" values obtained from CENTRM's MG solution. PMC provides two options to define the nuclide-specific energy range for computing problem-dependent MG data:

Option (1). Compute MG cross sections of a given nuclide only over the resolved resonance range of the nuclide. If the CENTRM PW calculation does not encompass the entire resolved resonance range for the nuclide, pseudo-point fluxes are be used in the self-shielding calculations for some groups in the resolved regions. The pseudo-point fluxes are generally a good representation for the gross spectrum shape, but do not reflect fine-structure effects caused by resonance absorption; therefore with this option, the user should take care that the CENTRM PW limits are appropriate for the resonance nuclides of interest.

Option (2). Compute MG cross sections for a given nuclide over the entire energy range for which PW flux values are calculated in the CENTRM. In this case PMC computes MG cross sections only over the portion of the PW data that is contained within the PW flux range; i.e., the pseudo PW spectrum is not used to process any data. Shielded cross sections for groups not included in the PW calculation are based on the BONAMI self-shielding method.

Option (2) above is default in PMC. SCALE-6.2 has DEMIN and DEMAX default values of 0.001 eV and 20 keV. This is sufficient for resonance self-shielding of essentially all actinide and important fission product nuclides; but some structural materials such as iron have resonances above 20 keV which would be shielded by BONAMI (Sect. 7.3).

7.5.2.4 Options for treatment of 2-D cross sections

The input parameter N2D defines five PMC options for processing problem-dependent, 2-D elastic scattering matrices. The first approach, N2D=0, simply multiplies the elastic scattering matrices by the ratio of the new to old 1-D elastic cross sections for the specified reaction process, where the "old" data are the 1-D values in the original MG library, and the "new" data are the problem-dependent MG cross sections processed using the PW flux as described above. The P_{ell} Legendre moments as well as the P_0 matrix are scaled by the same ratio for a given group. This method is also always used for discrete-level and continuum inelastic cross sections, as well as any other 2-D data other than elastic. The basic assumption is that the relative group-to-group scattering distribution does not change from the distribution in the original MG library, which is processed with an infinitely dilute spectrum- i.e., self-shielding only affects the total scatter rate. This approach gives good results for many applications, and is very efficient computationally. However, for intermediate and high mass materials, the elastic removal rate from a group may be sensitive to the problem-dependent CE spectrum. In these cases the scaling approximation may not give the correct elastic removal rate from the group, because the within-group elastic cross section will be in error. In these cases the alternate approaches described below can be used.

The option N2D= -1 corrects for the impact of resonance self-shielding on the elastic removal from an energy group. This option recomputes a new value for the within-group cross section by applying a correction factor based on the ratio of shielded versus unshielded removal probabilities for *s*-wave scatter (isotropic center-of-mass scatter). The P_0 out-scattering cross sections are then renormalized to give the correct 1D shielded cross section for the group. This approach provides a reasonable and computationally efficient approximation to process 2D elastic matrices in the resolved resonance range of actinide nuclides. However the assumption of s-wave scatter may not be valid in the resolved resonance range of a structural material such as iron; therefore users should beware when applying the approximation if the PW range is extended above 50 keV, for systems with large sensitivity to structural materials.

Option N2D=1 uses the CENTRM PW flux to recompute the entire set of group-to-group scatter data (including Legendre moments) using PW thermal scattering kernel data for the thermal energy range and assuming *s*-wave kinematics for the epithermal energy range. Since the CENTRM PW flux is used as the weighting function, this approach is sometimes more accurate for groups with large spectral gradients as discussed above. As with the N2D=-1 option, the main limitation is the *s*-wave scattering approximation for the secondary energy distribution. This option requires more computation time than the N2D methods discussed previously, and usually gives similar results as N2D=-1.

A rigorous derivation of the MG transport equation from the CE equation results in a directionally dependent total cross section. PMC option N2D=2 uses the method in [PMCBG70] to address this effect by modifying the Legendre moments of the 2D elastic matrix. For cross section moment "n", the diagonal term (i.e., within-group scatter) is modified by adding a term equal to the difference in the MG total cross section weighted with the PW scalar flux and the MG total cross section weighted with the n_{th} Legendre moment of the PW flux. When using MoC (NPXS=6), since default ISCT is 0, there are no Pn flux moments on CENTRM PW flux file. Therefore, no diagonal P_n correction will be applied to cross section moments.

Option N2D=-2 is essentially a combination of options N2D=2 and N2D=-1. This option applies the elastic removal correction to the diagonal term of the P_0 moment of the elastic 2D matrix, and applies the PL correction described above to the diagonal term of the PL Legendre moment of the elastic matrix.

The thermal energy range presents a particularly difficult challenge for processing problem-dependent 2-D scattering data, due to the complicated kinematics associated with molecular motion, chemical binding, and coherent scattering effects. PMC currently uses the scaling approximation (N2D=0 option) for the thermal energy range with any input value of N2D except for N2D=1.

7.5.3 CALCULATION OF PROBLEM-DEPENDENT MULTIGROUP CROSS SECTIONS

7.5.3.1 1-D cross sections

$$\sigma^{j}_{z,r,g} = \frac{\int_{\Delta E_g} \sigma^{j}_{z,r}(E)\Phi_z(E)dE}{\int_{\Delta E_g} \Phi_z(E)dE} = \frac{\int_{\Delta E_g} \sigma^{j}_{z,r}(E)\Phi_z(E)dE}{\Phi_{z,g}} \tag{7.5.1}$$

where

$\Phi_{z,g}$ is the multigroup zone flux,

$\sigma^{j}_{z,r,g}$ is the zone-average, group cross section, and

ΔE_g is the energy interval of group g.

The integration in Eq. (7.5.1) is performed by summing over a discrete energy mesh within the group boundaries. Since the CE cross section and the PW flux generally have different energy grids, the integration mesh for the numerator is formed by taking the union of the two. The CE cross sections and the PW flux are mapped onto the union mesh, and the integral is evaluated using the trapezoidal method. Eq. (7.5.1) is used to compute weighted group data for all MT's for which CE data are available on the CENTRM library, except in the case of the fission neutron yield ν. Instead of using the PW scalar flux as the weighting function, the MG value for ν is weighted by the product of the PW flux and the PW fission cross section for the material.

7.5.3.2 2-D scattering cross sections

The 2-D MG cross section moments are defined as the weighted group-average of terms appearing in a Legendre (PL) expansion of the CE double-differential scatter cross section, which describes the transfer of neutrons from one energy to another, for a given angle of scatter. The PL Legendre moments on the original MG library are fully consistent with the ENDF/B kinematic specifications. Thus the specified anisotropy in elastic or inelastic data in the center-of-mass (CM) system is reflected in the PL scattering matrices; however the library MG data are processed with an infinitely dilute flux spectrum. PMC provides several options for modifying these data to correct for problem-specific spectral effects, such as self-shielding. First, consider the scaling method (N2D=0) in which all the elements of the original scatter matrix (i.e., on the input Master library) for a given initial group are multiplied by the ratio of 1-D scatter cross sections. This has the effect of normalizing the original scatter matrix to the problem-dependent value calculated for the 1-D scatter data. In this case the l_{th} Legendre moment of the 2-D multigroup cross section for reaction type "s" of nuclide "j" in zone "z" (at a specified temperature), for scatter from initial group g′ to final group g, is computed by:

$$\sigma^{j}_{l,z,s,g'\to g} = \frac{\left(\sigma^{j}_{z,s,g'}\right)_{new}}{\left(\sigma^{j}_{s,g'}\right)_{orig}} \times \left(\sigma^{j}_{l,s,g'\to g}\right)_{orig} \tag{7.5.2}$$

where the subscripts "*orig*" and "*new,*" respectively, refer to the original MG data on the Master library, and the new problem-dependent data computed by PMC. The types of reactions for which problem-dependent 2-D cross sections may be processed using the scaling method are elastic (MT=2), discrete-level inelastic (MT's 50–89), continuum inelastic (MT=90), and (n,2n) (MT=16). This approach is also applied to obtain problem-dependent thermal scatter matrices, which contain upscatter as well as down-scatter reactions. The CENTRM nuclear data libraries include PW cross sections for incoherent (MT=1007) and coherent (MT=1008, if available) thermal scattering reactions, which can be processed into 1-D MG data by PMC in the same manner as other reaction types. The 1-D weighted thermal scattering data are then used to normalize

the 2-D thermal matrices on the input Master library. For materials with both coherent and incoherent thermal scatter data, each matrix is scaled by the corresponding type of 1-D data. The coherent scattering matrix only contains within-group terms.

The option N2D= -1 recomputes the P_0 within-group elastic cross section based on the assumption of s-wave scatter kinematics, and scales the other terms of the original P0 elastic matrix by the modified removal rate. This procedure approximately corrects for effects of resonance self-shielding on the group removal probability, without having to recompute the entire matrix assuming *s*-wave scatter, as done for N2D=1. Suppressing the zone index for simplicity, the P_0 within-group XS is defined as:

$$\sigma_{\mathrm{g,g}} \equiv \frac{\int_{\mathrm{g}} \sigma_{\mathrm{s}}(\mathrm{E}) \left[1 - \mathrm{p_r(E)}\right] \Phi(\mathrm{E}) \mathrm{dE}}{\int_{\mathrm{g}} \Phi(\mathrm{E}) \mathrm{dE}} \tag{7.5.3}$$

where $p_r(E)$ is the probability that a neutron at energy E, within group g, will scatter to an energy below the lower boundary of the group. For *s*-wave scattering this equation becomes,

$$\sigma_{\mathrm{g,g}} = \frac{\int_{\mathrm{E_{Lo}}}^{\min\left(\mathrm{E_{Hi}}, \frac{\mathrm{E_{Lo}}}{\alpha}\right)} \sigma_{\mathrm{s}}(\mathrm{E}) \left[\frac{\mathrm{E}-\mathrm{E_L}}{\mathrm{E}(1-\alpha)}\right] \Phi(\mathrm{E}) \mathrm{dE}}{\int_{\mathrm{g}} \Phi(\mathrm{E}) \mathrm{dE}} \tag{7.5.4}$$

The N2D= -1 option recomputes a modified P_0 within-group cross section from the expression,

$$\left(\sigma_{\mathrm{g,g}}\right)_{\mathrm{new}} = \frac{\widetilde{\sigma}^{(\varphi)}_{\mathrm{g,g}}}{\widetilde{\sigma}^{\infty}_{\mathrm{g,g}}} \left(\sigma_{\mathrm{g,g}}\right)_{\mathrm{orig}} \tag{7.5.5}$$

where

$(\sigma_{\mathrm{g,g}})_{\mathrm{orig}}$ is the original within-group cross section on the MG library, based on actual kinematics and weighted with an infinitely dilute spectrum;

$\widetilde{\sigma}^{(\infty)}_{\mathrm{g,g}}$ is the infinitely dilute within-group cross section based on s-wave kinematics, which is computed from Eq. (7.5.4) using an infinitely dilute spectrum

$\widetilde{\sigma}^{(\varphi)}_{\mathrm{g,g}}$ is the self-shielded within-group based on s-wave kinematics, computed from Eq. (7.5.4) using $\Phi(\mathrm{E}) \rightarrow$ CENTRM PW flux.

If the effects of resonance self-shielding are small, then there will be little change in the original within-group value, since in this case $\widetilde{\sigma}^{(\varphi)}_{\mathrm{g,g}} \sim \widetilde{\mathrm{O}}^{(\infty)}_{\mathrm{g,g}}$.

The P_0 group-to-group out-scatter terms for N2D=-1 are scaled as follows:

$$\sigma_{g \rightarrow g'} = \frac{\left(\sigma_{\mathrm{s,g}}\right)_{\mathrm{new}} - \widetilde{\sigma}^{(\varphi)}_{\mathrm{g,g}}}{\left(\sigma^{\infty}_{\mathrm{s,g}}\right)_{\mathrm{new}} - \widetilde{\sigma}^{\infty}_{\mathrm{g,g}}} \times \left(\sigma_{\mathrm{g \rightarrow g'}}\right)_{\mathrm{orig}} \tag{7.5.6}$$

Again if there is little self-shielding, the change in off-diagonal matrix elements is small, so that the original secondary energy distribution is preserved. Finally the entire modified P_0 scatter matrix is renormalized to correspond to the self-shielded 1-D scatter cross section.

For the option N2D=1, an entirely new PL elastic scattering matrix is computed. The l_{th} Legendre moment of the 2-D MG elastic cross section of nuclide "j" in zone "z" (at a specified temperature), for scattering from initial group g′ to final group g is rigorously defined as, [PMCBG70]

$$\sigma^{j}_{l,g' \rightarrow g} = \frac{\int_{\Delta E_g} \int_{\Delta E_{g'}} \sigma^{j}_{l}(E' \rightarrow E) \Phi_{l,z}(E') \, dE' dE}{\int_{\Delta E_{g'}} \Phi_{l,z}(E') \, dE'} = \frac{\int_{\Delta E_g} \int_{\Delta E_{g'}} \sigma^{j}(E') f^{j}_{l}(E' \rightarrow E) \Phi_{l,z}(E') \, dE' dE}{\int_{\Delta E_{g'}} \Phi_{l,z}(E') \, dE'} \tag{7.5.7}$$

where σ_z(E) is the CE elastic cross-section data from the CENTRM nuclear data file, evaluated at the appropriate temperature for zone z; f_l^j (E′ → E) is the secondary neutron energy distribution from elastic scattering; and $\Phi_{l,z}$(E) is the lth PW flux moment averaged over zone Z. PMC assumes *s*-wave scattering from stationary nuclei to evaluate the scattering distribution, and uses the P_0 flux moment (i.e., scalar flux) as for the weighting function for all PL matrices; therefore the expression evaluated by PMC for N2D=1 is:

$$\sigma^j_{l,z,g'\to g} = \frac{\int_{g'}\int_g \frac{\sigma^j_z(\mathrm{E})\Phi_z(E')P_l(G^j)}{(1-\alpha^j)E'}dE'dE}{\int_g \Phi_z(E')\,dE'} \tag{7.5.8}$$

here Pl is the l_{th} order Legendre polynomial; and G^j is the kinematics relation expressing the cosine of the scattering angle as a function of E and E', for elastic scattering from nuclear mass A^j. The kinematics function for nuclide j is defined as,

$$\mathrm{G^j}(\mathrm{E'},\mathrm{E}) = \frac{\mathrm{A^j}+1}{2}\sqrt{\frac{\mathrm{E}}{\mathrm{E'}}} - \frac{\mathrm{A^j}-1}{2}\sqrt{\frac{\mathrm{E'}}{\mathrm{E}}}, \tag{7.5.9}$$

where G^j(E′,E) is equal to the cosine of the angle of scatter between the initial and final directions. The integral over the final group (g) is evaluated analytically using routines developed by J. A. Bucholz [PMCBuc78]. Integration over the initial group (g′) is then performed numerically using the same method as for evaluating the problem-dependent 1-D cross sections.

Option N2D=2 adds the following term to the diagonal of the l_{th} moment of the PL elastic scatter matrix,

$$\left(\sigma^j_{l;g,g}\right)_{new} = \left(\sigma^j_{l;g,g}\right)_{orig} + \sigma^j_{t;g} - \sigma^j_{t,l;g}; \quad 0 < l < isct \tag{7.5.10}$$

where isct is the order of scatter specified in CENTRM calculation; $\sigma^j_{t;g}$ is the standard 1D total cross section weighted with the scalar flux, and $\sigma^j_{t,l;g}$ is the total cross section weighted with the l^{th} Legendre moment of the angular flux; i.e.,

$$\sigma^j_{t,l;g} = \frac{\int_g \sigma^j_t(E)\Phi_l(E)dE}{\int_g \Phi_l(E)dE} \tag{7.5.11}$$

7.5.3.3 Problem-dependent fission spectra

Fission spectra (chi) describing the energy distribution of secondary neutrons produced by fission depend upon the energy of the neutron causing the fission, thus the MG chi data should be a 2-D matrix, $\chi_{g\to g'}$. However, neutron transport codes in SCALE expect a 1-D distribution, $\chi_{g'}$; therefore the production of fission neutrons in group g′ by neutrons in group g is approximated as,

$$\mathrm{P_{g\to g'}} = \chi_{\mathrm{g'}}\nu_{\mathrm{g}}\sigma_{\mathrm{f,g}}\Phi_{\mathrm{g}} \tag{7.5.12}$$

and the total number of secondary neutrons generated in group g' is,

$$\mathrm{P_{g'}} = \chi_{\mathrm{g'}}\sum_{\mathrm{g}}\nu_{\mathrm{g}}\sigma_{\mathrm{f,g}}\Phi_{\mathrm{g}} \tag{7.5.13}$$

SCALE MG libraries contain "generic" 1-D chi distributions for each fissionable nuclide. These are processed from the evaluated ENDF/B fission data, weighted by the standard weighting function used to process the SCALE MG libraries (i.e., Maxwellian in thermal energy range, 1/E in epithermal range, fission spectrum in

fast range). The SCALE MG libraries also contain 2-D chi distributions processed from ENDF/B fission data, can be processed with a problem-dependent weighting function to create a more representative 1-D chi. This procedure is done in PMC for each fissionable nuclide, using the following equation that preserves the secondary neutron energy distribution:

$$\chi_{g'} = \frac{\sum_g \chi_{g\to g'} \nu_g \sigma_{f,g} \Phi_g}{\sum_g \nu_g \sigma_{f,g} \Phi_g} \tag{7.5.14}$$

In the above equation, $\nu_g, \sigma_{f,g},$ and Φ_g are problem-dependent 1-D data computed by PMC using the PW fluxes calculated by CENTRM, and $\chi_{g\to g'}$, are the 2-D MG fission spectra data on the AMPX multigroup Master library. The 1D prompt chi computed by PMC includes all fission components (first-chance-fission, second-chance-fission, etc) given in the ENDF/B files, weighted by the relative fission-source fraction associated with each channel. PMC also computes an effective delayed neutron fission spectra, and this is combined with the prompt chi, using the appropriate delayed neutron fraction, to obtain the final 1-D fission spectra. The 1-D chi computed by PMC replaces the generic 1-D values for MT-1018 that were originally in the Master library.

7.5.3.4 Definition of background cross sections

The value of the "background cross section (σ_0)" may be used in PMC to determine which materials are considered to be infinitely dilute, in which case no cross section processing is done for the material. No processing is performed for material "j" if its background cross section exceeds the value of input parameter *XS_dilute* ; i.e., if $\sigma_0^{(j)}$ > XS_dilute. The expression used in PMC to compute the background cross section $\sigma_0^{(j)}$ is given in the BONAMI section.

7.5.4 PMC INPUT DATA

The Fido input blocks shown in this section are only required when executing PMC as a standalone module. In the more typical case where PMC is executed through the XSProc module during a SCALE sequence calculation, the default parameter values are automatically defined within XSProc. Default values for XSProc execution can be overridden using keyword input in the CENTRM DATA block (see Sect. 7.4.4). The keyword input names correspond to the variable names given in this section.

DATA BLOCK 1

0$ $ LOGICAL UNIT ASSIGNMENTS (8 entries. Default values given in parenthesis)*

1. LIBM = Input AMPX Master nuclear data library (22)
2. LIBX = Input CENTRM pointwise nuclear data library (90)
3. LIBF = Pointwise flux file produced by CENTRM (91)
4. LIBNM = Output problem-dependent Master library created by PMC (92)
5. LIBSC = Scratch unit (18)
6. LIBSX = Scratch unit (24)

(*) Parameters in the 0$ $ array cannot be modified for XSProc execution.

1$ $ INTEGER PARAMETERS (10 entries)

1. MRANGE

= 0, obsolete option

= 1, Compute new group cross sections over resolved resonance range of pointwise nuclides [from EUPR to ELOR given in CENTRM data library]

= 2, Compute new group cross sections over pointwise flux range [from DEMAX to DEMIN in CENTRM flux calculation] (2).

2. N2D

   = -2, Apply removal correction to P0 elastic scatter matrix AND apply consistent PN correction to higher order Legendre components; normalize to 1D.

   -1, Apply elastic removal correction to P0 elastic scatter matrix; normalize to 1D.

   = 0, Normalize P_N components of original elastic scattering matrix to new 1-D elastic value.

   = 1, Compute new P_N components of elastic matrix, using scalar flux as weighting function.

   = 2, Modify diagonal elements of the PN moments of the elastic matrix using the consistent PN method (-1).

3. NTHRM

   = 0 Treatment of thermal scatter kernels [not functional] (0)

4. NPRT

   = -1, Minimum printed output;

   = 0, Standard print out;

   = 1, Also print new weighted cross sections for MT's 1, 2, 18, and 102.

   = 2, Maximum amount of printed output includes 2D matrices (-1).

5. NWT

   = 0, Generate zone-weighted multigroup data;

   = 1, Generate cell-weighted multigroup data (0).

6. MTT

   = 0, Process all MT's included in LIBX. [**NOTE:** With this option, total cross section may not equal to sum of partials];

   = 1, Process all MT's except 1, 27, 101; then compute:

      MT 101 = sum of MT's 102-114,

      MT 27 = sum of MT's 18 and 101,

      MT 1 = sum of MT's 2, 4, 16, 17, and 27 (1).

7. PMC_OMIT

   = 0, Process all pointwise nuclides used in CENTRM calculation;

   = 1, Process only nuclides in fuel zones.

   > 1, Process all materials except those in 2$$ array

8. IXTR2

   = 0, PMC run in CSAS standard sequence;

   = 1, PMC run in stand-alone mode (1);

   = 2 PMC run in CSAS double-heterogeneous cell sequence

9. IXTR3

   = -1, Process new data for all Legendre components on the input AMPX master library up to P_7.

   = N, Process new data through P_N moments. [N=Scattering Order+1] (-1).

10. N1D

    = 0 Use CENTRM scalar flux for weighting function;

    = 1, Use the absolute value of CENTRM current for weighting function (0).

*1** REAL PARAMETERS** (10 entries)

1. XS_DILUTE = background cross section (barns) considered to be infinitely dilute (10^{10})

2-10. Fill with 0.0

T [TERMINATE DATA BLOCK 1]

DATA BLOCK 2 : INDIVIDUAL NUCLIDES OMITTED FROM PROCESSING

Note: This data cannot be entered for XSProc execution.

2$ $ ISOTOPE IDENTIFIERS (PMC_OMIT entries). Only enter PMC_OMIT > 1

[IDs of nuclides to be omitted from pointwise processing]

T [TERMINATE DATA BLOCK

END OF PMC INPUT DATA

7.5.4.1 Notes for PMC users

1. N2D specifies the method used to process the P_N components of the 2-D elastic scattering matrices. In the option N2D=0, the P_N components of the original elastic scattering matrix are simply re-normalized using the new, problem-dependent 1-D elastic values. This simple scaling approach often works well, but it does not account for the impact of resonance self-shielding on the group removal probability. The default option N2D= -1 approximately corrects the P0 elastic matrix for removal self-shielding effects on and is usually preferred to N2D=0, except for fast systems. Option N2D=1 re-computes all the P_N components of 2-D elastic cross sections using the scalar flux as a weighting function, along with the the use of PW thermal scattering kernel data and assumption of *s*-wave scattering within the epithermal PW energy range. This approach takes significantly more execution time than N2D=-1, and usually is not necessary. Option N2D=2 corrects the diagonal terms of the Legendre moments, using the consistent PN expression. Option N2D=-2 is similar to N2D=2, except the elastic removal correction is applied to the P0 moment (Like for N2D=-1). Option N2D=-2 has been found to improve results for many infinite lattice cases.

2. NWT specifies whether the new multigroup cross sections are zone-weighted or cell-weighted. When PMC is executed through XSProc, nuclides are always zone-weighted unless the double-heterogeneous

option is specified in the CELLDATA block of the sequence input. Except for double-heterogeneous cells, cell-weighting of the MG cross sections should be done by the multigroup XSDRNPM calculation.

3. PMC_OMIT is used to indicate which pointwise nuclides are processed when computing new group cross sections. If PMC_OMIT=1, only nuclides in fuel mixtures are processed. Fuel mixtures are defined as having at least one material with Z ≥ 90. Option PMC_OMIT>1 only works for PMC standalone runs, since there is no mechanism for inputting the 2$$ array in sequences.

4. IXTR3 is used to indicate through what Legendre order the scattering matrices are to be processed. By default, in stand-alone mode all P_N moments on the Master library are processed, where as in a SCALE sequence only through order N=5 are processed. With few exceptions, the SCALE multigroup libraries contain scattering data through P_5.

5. If input parameter XS_DILUTE > 0.0, PMC computes background cross sections (σ_0) for each material, and bypasses processing materials with σ_0 > XS_DILUTE. The default of XS_DILUTE =10^{10} barns causes essentially all materials to be processed regardless of dilution. Smaller XS_DILUTE values may reduce the number of materials being processed, and hence reduce the execution time; however, XS_DILUTE should not be so low that important absorbers are not shielded.

7.5.5 EXAMPLE CASE

Usually PMC is executed through one of the automated SCALE sequences such as CSAS or TRITON where it is called by XSProc in conjunction with other SCALE modules, such as CRAWDAD which provides the pointwise nuclear data library and CENTRM which provides pointwise fluxes. In such cases the user does not have to prepare input directly for PMC.

7.5.5.1 PMC input for example case

An example of PMC stand-alone execution is given below, but it should be noted that this PMC case cannot be executed unless it is linked to the output data files produced by other modules. The example problem given in the CENTRM chapter shows the coupled execution of several stand-alone modules, including PMC, which mimics the function of XSProc.

```
=pmc
0$$      -42      81      15      -42      18      19      17
1$$     2   -1    0   0    0    1    0    0    5    0
 1t
end
```

7.5.5.2 PMC output for example case

Only the printed output produced by PMC for the example problem is shown here. In this case the "standard" PMC editing option (NPRT=0) was specified. The XSProc default of "minimum" print in the SCALE sequences produces considerably less output.

```
                program verification information

           code system:    scale  version:    6.0



       program:  pmc

  creation date:  18_nov_2008
```

(continues on next page)

(continued from previous page)

```
        library:  /scale/scale6/Linux_x86_64/bin

  production code:  pmc

        version:  6.0.9

        jobname:  xmw

    machine name:  node12.ornl.gov

date of execution:  05_dec_2008

time of execution:  13:22:19.23
```

```
1

     0$ array      7 entries read

     1$ array     10 entries read

     1t

          **** LOGICAL UNITS ****

     nin   =   5   Card Image Input Unit
     nout  =   6   Print Output Unit
     libm  = -42   Input Master Library
     libx  =  81   Input Pointwise XS Library
     libf  =  15   Input Pointwise Flux File
     libnm = -42   Output Master Library
     libsc =  18   Scratch Unit 1
     libsx =  19   Scratch Unit 2
     libsm =  17   scratch unit (master library)



          **** INPUT PARAMETERS ****

     mrange =  2   Option for choosing energy range     0   Averaging over pointwise xs limits
                                                        1   Averaging over resolved resonance range
                                                        2   Averaging over pointwise flux limits

     n2d    = -1   Option for 2-D scat. calculation    -1  Recompute self-scatter, then normalize 2-D elastic
                                                           data to shielded 1-D value
                                                        0   Normalize 2-D elastic data to shielded 1-D value
                                                        1   Recompute 2-D elastic using flux and s-wave kernel
                                                        2   Recompute 2-D moments with flux-moments weighting

     nthrm  =  0   Option for thermal scatter kernal
                      (NOT FUNCTIONAL)

     nprt   =  0   Option for PMC print output         -1   Minimum data printed
                                                        0   Standard printed output
                                                        1   Print 1-D XSs
                                                        2   Print both 1-D and 2-D XSs

     nwt    =  0   Option for XS averaging              0   Zone average
                                                        1   Cell average

     mtt    =  1   Option for total XS calculation      0   Average independently
                                                        1   As sum of partial XS

     ixtr(1)=  0   Option for Processing PW Materials   0   Process all Pointwise Materials Used in CENTRM
                                                        N   Omit N Materials
```

(continues on next page)

(continued from previous page)

```
    ixtr(2)=  0   Option for calculation sequence      0   CSAS Standard Sequence
                                                       1   Independant (stand-alone) Execution
                                                       2   CSAS Doubly-Heterogeneous Cell Sequence

    ixtr(3)=  5   Legendre expansion order            -1   Process all Legendre expansion moments found on␣
↪AMPX LIB.
                                                 =0,...N   Process only up through PN moments

    n1d    =  0   Option for 1-D cross-sections        0   Weight using using scalar flux
                                                       1   Weight using using abs value of current (1st␣
↪moment)
```

```
**** POINTWISE CROSS SECTION LIBRARY ****

    tape identifier                   66666
    No. of nuclides                       9
    Max no. of temperatures               2
    Max no. of processes                  9
    Max no. of energy points         174194



         **** POINTWISE FLUX FILE ****

    No. of nuclides                      10
    No. of flux moments                   1
    No. of zones                          3
    No. of energy points              48313
    Upper energy limit,demax    0.25000E+05
    Lower energy limit,demin    0.10000E-02



         **** AMPX INPUT MASTER LIBRARY ****

    ID of the tape               238000
    No. of nuclides                  10
    No. of neutron groups           238
    No. of gamma groups               0


         **** POINTWISE CROSS SECTION DIRECTORY ****

   ZA     Pointwise   Pointwise  Unresolved   Resolved    Resolved
            EMAX        EMIN        EMAX        EMAX        EMIN
   8016  0.2500E+05  0.1000E-02  0.0000E+00  0.0000E+00  0.0000E+00
  40090  0.2500E+05  0.1000E-02  0.4000E+06  0.6000E+05  0.0000E+00
  40091  0.2500E+05  0.1000E-02  0.1000E+06  0.2000E+05  0.0000E+00
  40092  0.2500E+05  0.1000E-02  0.1000E+06  0.7100E+05  0.0000E+00
  40094  0.2500E+05  0.1000E-02  0.1000E+06  0.9000E+05  0.0000E+00
  40096  0.2500E+05  0.1000E-02  0.1000E+06  0.1000E+06  0.0000E+00
  92235  0.2500E+05  0.1000E-02  0.2500E+05  0.2250E+04  0.0000E+00
  92238  0.2500E+05  0.1000E-02  0.1490E+06  0.2000E+05  0.0000E+00
   1001  0.2500E+05  0.1000E-02  0.0000E+00  0.0000E+00  0.0000E+00
```

```
**** NUCLIDES IN POINTWISE FLUX CALCULATION ****
      Zone   IR(# of nuclides)     Temperature
        1           3                 900.0
        2           5                 600.0
        3           2                 600.0

              -- Nuclide  by Zone --
               0 -- no;  1 -- yes
```

(continues on next page)

(continued from previous page)

```
   ID::       1008016     3008016     2040090     2040091     2040092     2040094
  ZONE::
    1            1           0           0           0           0           0
    2            0           0           1           1           1           1
    3            0           1           0           0           0           0
   ID::       2040096     1092235     1092238     3001001
  ZONE::
    1            0           1           1           0
    2            1           0           0           0
    3            0           0           0           1

           -- Atom Density by Zone --
   ID::       1008016     3008016     2040090     2040091     2040092     2040094
  ZONE::
    1      4.5968E-02  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
    2      0.0000E+00  0.0000E+00  2.5714E-02  5.6076E-03  8.5714E-03  8.6863E-03
    3      0.0000E+00  2.3831E-02  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00

           -- Averaged Cell Atom Density    --
          2.2461E-02  1.0506E-02  1.8133E-03  3.9544E-04  6.0445E-04  6.1255E-04


   ID::       2040096     1092235     1092238     3001001
  ZONE::
    1      0.0000E+00  4.8838E-04  2.2480E-02  0.0000E+00
    2      1.3994E-03  0.0000E+00  0.0000E+00  0.0000E+00
    3      0.0000E+00  0.0000E+00  0.0000E+00  4.7662E-02

           -- Averaged Cell Atom Density    --
          9.8685E-05  2.3863E-04  1.0984E-02  2.1012E-02



        **** INPUT MASTER LIB. DIRECTORY ****

    nmt:   No. of 1-D Neutron Processes
    nbond: No. of Sets of Bondarenko Data
    nrec:  No. of Records for this Nuclide

       id          za          nmt        nbond     nrec

    1008016      8016.0         49          0          3
    3008016      8016.0         49          0          3
    2040090     40090.0         86          0          3
    2040091     40091.0         45          0          3
    2040092     40092.0         47          0          3
    2040094     40094.0         39          0          3
    2040096     40096.0         32          0          3
    1092235     92235.0         77          0          3
    1092238     92238.0         77          0          3
    3001001      1001.0         10          0          3
```

```
P r o c e s s i n g   N u c l i d e  1008016

Energy Range for Multigroup Averaging of this Data
EH = 2.50000E+04  EL = 1.00000E-03

      INFORMATION ON CENTRM POINTWISE XS LIB:

   MT    ENERGY POINTS      TEMPERATURE (K)
    1        459            600.0    900.0
    2        459            600.0    900.0
  102        459            600.0    900.0
```

(continues on next page)

(continued from previous page)

```
          <<<<< ZONE:    1 >>>>>

PROCESSING MT =     1
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =     2
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =   102
     Generating new multigroup data from group   55 through group  234

====>> Done Processing Shielded Zone-Averaged Cross Section  1008016


P r o c e s s i n g   N u c l i d e  3008016

Energy Range for Multigroup Averaging of this Data
EH = 2.50000E+04  EL = 1.00000E-03

      INFORMATION ON CENTRM POINTWISE XS LIB:

   MT    ENERGY POINTS      TEMPERATURE (K)
    1        459            600.0    900.0
    2        459            600.0    900.0
  102        459            600.0    900.0

          <<<<< ZONE:    3 >>>>>

PROCESSING MT =     1
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =     2
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =   102
     Generating new multigroup data from group   55 through group  234

====>> Done Processing Shielded Zone-Averaged Cross Section  3008016
```

```
P r o c e s s i n g   N u c l i d e  2040090

  Energy Range for Multigroup Averaging of this Data
  EH = 2.50000E+04  EL = 1.00000E-03

         INFORMATION ON CENTRM POINTWISE XS LIB:

       MT    ENERGY POINTS      TEMPERATURE (K)
        1       4488            600.0
        2       4488            600.0
      102       4488            600.0

             <<<<< ZONE:    2 >>>>>

    PROCESSING MT =     1
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =     2
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =   102
         Generating new multigroup data from group   55 through group  234

  ====>> Done Processing Shielded Zone-Averaged Cross Section  2040090


    P r o c e s s i n g   N u c l i d e  2040091

  Energy Range for Multigroup Averaging of this Data
  EH = 2.50000E+04  EL = 1.00000E-03
```

(continues on next page)

(continued from previous page)

```
         INFORMATION ON CENTRM POINTWISE XS LIB:

       MT    ENERGY POINTS      TEMPERATURE (K)
        1      24295            600.0
        2      24295            600.0
      102      24295            600.0

               <<<<< ZONE:    2 >>>>>

    PROCESSING MT =     1
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =     2
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =   102
         Generating new multigroup data from group   55 through group  234

  ====>> Done Processing Shielded Zone-Averaged Cross Section  2040091
```

```
  P r o c e s s i n g   N u c l i d e  2040092

Energy Range for Multigroup Averaging of this Data
EH = 2.50000E+04  EL = 1.00000E-03

        INFORMATION ON CENTRM POINTWISE XS LIB:

     MT    ENERGY POINTS      TEMPERATURE (K)
      1       8142            600.0
      2       8142            600.0
    102       8142            600.0

            <<<<< ZONE:    2 >>>>>

  PROCESSING MT =     1
       Generating new multigroup data from group   55 through group  234
  PROCESSING MT =     2
       Generating new multigroup data from group   55 through group  234
  PROCESSING MT =   102
       Generating new multigroup data from group   55 through group  234

====>> Done Processing Shielded Zone-Averaged Cross Section  2040092


  P r o c e s s i n g   N u c l i d e  2040094

Energy Range for Multigroup Averaging of this Data
EH = 2.50000E+04  EL = 1.00000E-03

        INFORMATION ON CENTRM POINTWISE XS LIB:

     MT    ENERGY POINTS      TEMPERATURE (K)
      1       8068            600.0
      2       8068            600.0
    102       8068            600.0

            <<<<< ZONE:    2 >>>>>

  PROCESSING MT =     1
       Generating new multigroup data from group   55 through group  234
  PROCESSING MT =     2
       Generating new multigroup data from group   55 through group  234
  PROCESSING MT =   102
       Generating new multigroup data from group   55 through group  234
```

(continues on next page)

(continued from previous page)

```
====>> Done Processing Shielded Zone-Averaged Cross Section  2040094


  P r o c e s s i n g   N u c l i d e  2040096

Energy Range for Multigroup Averaging of this Data
EH = 2.50000E+04  EL = 1.00000E-03

        INFORMATION ON CENTRM POINTWISE XS LIB:

     MT    ENERGY POINTS      TEMPERATURE (K)
      1       4944            600.0
      2       4944            600.0
    102       4944            600.0

            <<<<< ZONE:    2 >>>>>

  PROCESSING MT =     1
       Generating new multigroup data from group   55 through group  234
  PROCESSING MT =     2
       Generating new multigroup data from group   55 through group  234
  PROCESSING MT =   102
       Generating new multigroup data from group   55 through group  234

====>> Done Processing Shielded Zone-Averaged Cross Section  2040096
```

```
P r o c e s s i n g   N u c l i d e  1092235

  Energy Range for Multigroup Averaging of this Data
  EH = 2.50000E+04  EL = 1.00000E-03

          INFORMATION ON CENTRM POINTWISE XS LIB:

       MT    ENERGY POINTS      TEMPERATURE (K)
        1      59851            900.0
        2      59851            900.0
       18      59851            900.0
      102      59851            900.0
       51         94              0.0
       52         74              0.0
      452         48              0.0
      455          6              0.0
      456         48              0.0

              <<<<< ZONE:    1 >>>>>

    PROCESSING MT =     1
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =     2
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =    18
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =   102
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =    51
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =    52
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =   452
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =   455
         Generating new multigroup data from group   55 through group  234
    PROCESSING MT =   456
         Generating new multigroup data from group   55 through group  234
```

(continues on next page)

(continued from previous page)

```
    PROCESSING MT =  1018
     Collapsing 2D chi to effective 1D

  ====>> Done Processing Shielded Zone-Averaged Cross Section  1092235
```

```
P r o c e s s i n g   N u c l i d e  1092238

Energy Range for Multigroup Averaging of this Data
EH = 2.50000E+04  EL = 1.00000E-03

      INFORMATION ON CENTRM POINTWISE XS LIB:

   MT    ENERGY POINTS      TEMPERATURE (K)
    1     174194            900.0
    2     174194            900.0
   18     174194            900.0
  102     174194            900.0
  452         10              0.0
  455          4              0.0
  456         10              0.0

           <<<<< ZONE:    1 >>>>>

PROCESSING MT =     1
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =     2
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =    18
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =   102
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =   452
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =   455
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =   456
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =  1018
 Collapsing 2D chi to effective 1D

====>> Done Processing Shielded Zone-Averaged Cross Section  1092238


P r o c e s s i n g   N u c l i d e  3001001

Energy Range for Multigroup Averaging of this Data
EH = 2.50000E+04  EL = 1.00000E-03

      INFORMATION ON CENTRM POINTWISE XS LIB:

   MT    ENERGY POINTS      TEMPERATURE (K)
    1        324            600.0
    2        324            600.0
  102        324            600.0

           <<<<< ZONE:    3 >>>>>

PROCESSING MT =     1
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =     2
     Generating new multigroup data from group   55 through group  234
PROCESSING MT =   102
     Generating new multigroup data from group   55 through group  234
```

(continues on next page)

(continued from previous page)

```
====>> Done Processing Shielded Zone-Averaged Cross Section  3001001


elapsed time   0.01 min.
```

```
Number of nuclides in new master library  10

  The Output AMPX Master Library Produced by PMC

        Logical Unit No.                          -42
        Tape ID No.                            238000
        No. of Weighted Cross Section Sets          10
        No. of Neutron Groups                      238
        No. of Gamma Groups                          0
        First Thermal Neutron Group                149

  Contents of Output Master Library

        o16 825 endfb7 rel8 rev7 mod3                  08/13/08                        ID    1008016
        o16 825 endfb7 rel8 rev7 mod3                  08/13/08                        ID    3008016
        zr90 4025 endfb7 rel0 rev7 mod1                08/13/08                        ID    2040090
        zr91 4028 endfb7 rel0 rev7 mod1                08/13/08                        ID    2040091
        zr92 4031 endfb7 rel3 rev7 mod4                08/13/08                        ID    2040092
        zr94 4037 endfb7 rel3 rev7 mod1                08/13/08                        ID    2040094
        zr96 4043 endfb7 rel0 rev7 mod1                08/13/08                        ID    2040096
        u235 9228 endfb7 rel0 rev7 mod7                08/13/08                        ID    1092235
        u238 9237 endfb7 rel6 rev7 mod5                08/13/08                        ID    1092238
        h_h2o 1 endfbv7 rel0 rev7 mod0                 09/29/08                        ID    3001001

   elapsed time   0.02 min.

          **** PMC CALCULATION COMPLETED ****
```

7.5.6 FORMATS OF DATA FILES

The CENTRM chapter of the SCALE manual describes the format for the CENTRM PW nuclear data library and the format of the output PW flux file produced by CENTRM, which is input to the PMC code.

7.5.7 APPENDICES

7.5.7.1 Alphabetical Index of Subroutines

This section provides a convenient alphabetical index of the subroutines used in PMC, the subroutines that call them and the subroutines they call.

Subroutine Name	Calling Subroutine	Called Subroutine
add2d	xscal	pint
addpxs	xscal	
boss	pmc	flxrec
		process
		replace
		set
		zonefl
capy	replace	cellav2
		factor
		normal
		repl2d
cellav2	capy	
cellxs	process	
dilutxs	xscal	
factor	capy	iset
flxrec	boss	
normal	capy	iset
pgen	pint	
pint	add2d	pgen
		pinte
		pinto
pinte	pint	
pinto	pint	
pmc		boss
		setup
print	process	prtld
process	boss	cellxs
		print
		sumtot
		xscal
reorder	xscal	
repl2d	capy	
replace	boss	capy
setup	pmc	mesage
xscal	process	add2d
		addpxs
		dilutxs
		reorder
zonefl	boss	

7.5.7.2 Alphabetical Index of Modules

This section provides a list of the modules used in PMC and the subroutines that reference them.

ARRAY Format: READ ARRAY array parameters data type orientation data END ARRAY See Sections on ARRAY Data, Geometry, and alternative sample problem mockups

Repeat the sequence ARRAY PARAMETERS DATA TYPE ORIENTATION DATA for each array used in the problem.

ARRAY PARAMETERS

KEYWORD	DEFAULT	DEFINITION
ARA=	none[a]	no. defining the array
TYP=	Cuboidal[a]	array type (cuboidal or square, hexagonal or triangular, rhexagonal, rhexagonal, dodecahedral)
NUX=	none[a]	no. of units in X direction
NUY=	none[a]	no. of units in Y direction
NUZ=	none[a]	no. of units in Z direction
GBL=	none[b]	global or overall array number
PRT=	YES	print any label
COM=	none	delim comment delim optional, maximum of 132 characters

ORIENTATION DATA FOR FILL

Enter unit numbers to define every position in the array. When entering data using the options in this table, the count and option fields must be adjacent with no imbedded blanks. The operand field may be separated from the option field by one or more blanks. Orientation data for FILL is terminated by entering END FILL.

COUNT FIELD	OPTION FIELD	OPERAND FIELD	COMMENTS
		j	stores j at the current position in the array
i	R, *, $	j	stores j in the next i positions in the array
	F	j	fills remainder of the array with unit no. j starting with the current array position
	A	j	sets the current position in the array to j
i	S		increments current position in the array by i (This allows skipping i positions. The value of i may be positive or negative.)
i	Q	j	repeats the previous j entries i times. The default value of i is 1
i	N	j	repeats the previous j entries i times, inverting the sequence each time. The default value of i is 1.
i	B	j	starting with the entry at -i from the current position, store entries in inverse order until position -(i+j) is reached. Default value of i=1.
i	I	j k	provides the end points (j, k) with i entries linearly interpolated between them (i.e., a total of i+2 points). At least one blank must separate j and k. When used for an integer array, the i option should only be used to generate integer steps (i.e., (k-j)/(i+1) should be a whole number).

ORIENTATION DATA FOR LOOP

Enter the unit number and nine numbers that define the position(s) of that unit. Data for each of these ten entries are repeated until every position in the array has been defined. Orientation data for LOOP are terminated by entering END LOOP.
ENTER DATA IN THE FORM:

DATA ENTRY	COMMENTS
LTYPE	The unit type. LTYPE must be greater than 0.
IX1	Starting position in the X direction. IX1 must be at least 1 and no larger than the value entered for NUX.
IX2	Ending position in the X direction. IX2 must be at least 1 and no larger than the value of NUX.
INCX	The number of units by which increments are made in the X direction.
IY1	The starting position in the Y direction. IY1 must be at least 1 and less than the value entered for NUY.
IY2	Ending position in the Y direction. IY2 must be at least 1 and no larger than the value of NUY.
INCY	The number of units by which increments are made in the positive Y direction.
IZ1	Starting position in the Z direction. IZ1 must be at least 1 and no larger than NUZ.
IZ2	Ending position in the Z direction. IZ2 must be at least 1 and no larger than NUZ.
INCZ	The number of units by which increments are made in the positive Z direction.

[a] In KENO Va the default is 1

[b] In KENO Va the default is the largest array number, *nuxnuy*

7.6 CHOPS: MODULE TO COMPUTE POINTWISE DISADVANTAGE FACTORS AND PRODUCE A CELL-HOMOGENIZED CENTRM LIBRARY

M. L. Williams and L. M. Petrie

7.6.1 INTRODUCTION

CHOPS (**C**ompute **HO**mogenized **P**ointwise **S**tuff) computes pointwise (PW) disadvantage factors from the PW zone fluxes on a CENTRM output file, and then multiples the disadvantage factors by continuous-energy (CE) cross section data in a CENTRM library to generate a new cell-homogenized CENTRM CE library. The PW disadvantage factor for zone "Z", as a function of energy E, is calculated from the expression,

$$D_Z(E) = \frac{\Phi_Z(E)}{\Phi_C(E)} \tag{7.6.1}$$

where $\Phi_Z(E)$ is the CENTRM PW flux spectrum averaged over the volume of zone Z in the cell, and $\Phi_C(E)$ is the PW flux averaged over the entire cell volume. The cell-homogenized CE cross section for a nuclide "j"

is equal to

$$\sigma_C^{(j)}(E) = \sum_Z F_Z^{(j)} D_Z(E) \sigma_Z^{(j)}(E) \tag{7.6.2}$$

where $F_Z^{(j)}$ is the fraction of all nuclide–j atoms contained in zone Z, and $\sigma_Z^{(j)}(E)$ is the CE cross section for nuclide-j at the temperature of zone Z. When multiplied by the cell-homogenized number density of nuclide-j and by the cell-average flux, the cross section expression in Eq. (7.6.2) gives the correct average reaction rate at energy E.

CHOPS is used in the automated double heterogeneity sequence in SCALE, in which a low-level heterogeneity, such as microspheres in a granular fuel element, are smeared into a homogenized absorber region appearing in the second level heterogeneity, such as fuel pellet or pebble appearing in a lattice. The disadvantage factors provide for flux weighting of the PW XS data so that the spatial self-shielding is treated correctly in the homogenized geometry. A second CENTRM PW transport calculation is performed with the cell-averaged PW library output by CHOPS in order to account for the additional self-shielding of the absorber pellets/pebbles in the lattice. CHOPS is called automatically by the XSProc module for double-heterogeneous unit cells, or it can run as a standalone code.

7.6.2 CHOPS INPUT DATA

DATA BLOCK 1

0$ $ LOGICAL UNIT ASSIGNMENTS (10 entries. Default values given in parentheses)

1. lold – logical unit number of input CENTRM XS library (1)

2. lnew – logical unit number of output CENTRM homogenized XS library (2)

3. lflx – logical unit number of input CENTRM PW flux library (3)

4. ldis – logical unit number for edit of PW disadvantage factors (0)

5. n15 – logical unit number for scratch (15)

6. n16 – logical unit number for scratch (16)

7. n17 – logical unit number for scratch (17)

8. n18 – logical unit number for scratch (18)

9. n19 – logical unit number for scratch (19)

10. nsq – sequence number used in filename on unit “lnew” (1)

[Example: if *lnew=11* and *nsq=3*: output filename of homogenized library= *ft11f003]*

1$ $ INTEGER PARAMETERS (5 entries)

1. idtap – identifier for the new library (55555)

[for macro library, the value of *idtap* is made negative]

2. nprt – output print option: 0 = > min print; 1 = > normal; 2 = > max print (0)

3. iden – if=0 = > define homogenized XS id = id on CENTRM flux file (0)

if>0 = > define homogenized XS id to be, ($iden$*10^6 + ZA)

4. macr – type of XS output: 0 = > microscopic ; 1 = > macroscopic (0)

5. icorr – not used (0)

2 REAL PARAMETERS** (3 entries)

1. tole – tolerance used to thin pointwise cross-sections (0.0025)

(0.0 means no thinning is done)

2. cleth – maximum lethargy between thinned pointwise cross-sections

points that allow a point to be discarded (0.25)

3. vfrac – multiplier applied to all output XS's [eg, grain fraction] (1.0)

T [TERMINATE DATA BLOCK 1]

7.6.3 CHOPS I/O UNITS

Table 7.6.1 shows default logical unit numbers used by CHOPS. These values may be changed in the 0$$ array of input.

Table 7.6.1: Default I/O unit assignments for CHOPS.

Unit number	Description
1	Input CENTRM CE data library
2	Output homogenized CENTRM CE data library
3	Input pointwise CENTRM flux file
15	Scratch file
16	Scratch file
17	Scratch file
18	Scratch file
19	Scratch file

7.6.4 CHOPS SAMPLE INPUT

The sample case in Example 7.6.1 first executes a CENTRM unit cell geometry calculation using the CSAS-MG sequence, which by default generates the PW flux file on unit 15, as well as the CE nuclear data library on unit 81 for input to CHOPS. The standalone CHOPS code then computes a cell-homogenized CE library for the unit cell. The new homogenized CENTRM CE library is output on unit 91with filename: *ft91f001*

Example 7.6.1: CHOPS sample input.

```
  =CSAS-MG      parm=centrm
 test case for CHOPS
v7-252n
READ COMP
' Fuel pellet
o          1 0 4.59675e-2 900.0 end
u-235   1 0 4.88385e-4 900.0 end
u-238   1 0 2.24804e-2 900.0 end
' Clad
zr         2 0 4.99789e-2 600.0 end
' Coolant
h          3 0 4.76619e-2 600.0 end
o          3 0 2.38310e-2 600.0 end
END COMP
READ CELLDATA
```

(continues on next page)

(continued from previous page)

```
   latticecell squarepitch pitch=1.6  3
     fuel=1.262 1 cladd=1.350 2   end
END CELLDATA
END
=CHOPS
0$$  81 91 15 93 92  e
1$$  a2 1 e
2**  a3 1.0 e
 t
END
```

7.7 CRAWDAD: MODULE TO PRODUCE CENTRM-FORMATTED CONTINUOUS-ENERGY NUCLEAR DATA LIBRARIES

M. L. Williams, D. Wiarda, and S. W. D. Hart

ACKNOWLEDGMENTS

The authors would like to acknowledge the important contributions to CRAWDAD made by former ORNL staff N. M. Greene and D. F. Hollenbach.

7.7.1 INTRODUCTION

SCALE uses the code CRAWDAD (Code to Read And Write DAta for Discretized solution) to read nuclear data from the SCALE-6 continuous (CE) library files, and write it to an output file in the particular format needed for the discretized energy solution in CENTRM. Prior to SCALE-6, the CE data used by the CENTRM and PMC modules were distributed directly in the CENTRM library format. However beginning with SCALE-6, the same CE data are used by both CENTRM/PMC and by the CE versions of the KENO and Monaco Monte Carlo codes. CE nuclear data for each nuclide are stored in individual files contained in the SCALE permanent data directory. CRAWDAD reads the files for each material appearing in a problem and combines all data into a single problem-dependent CENTRM library file stored in the temporary directory for execution.

All SCALE-6 calculations that use modules CENTRM and PMC for self-shielding multigroup (MG) cross sections must first execute the CRAWDAD computational module. During execution of SCALE sequences, the XSProc self-shielding module automatically executes CRAWDAD whenever the CENTRM/PMC method is specified. CRAWDAD also can be run in stand-alone mode to process and save a CENTRM-formatted library for subsequent CENTRM/PMC calculations.

PMC allows the energy range of the CE data to be selected, as well as which reactions are placed on the output CENTRM library. The output CENTRM library always contains the following "standard" nuclear data types for all materials: total (1); elastic (2); complete inelastic (4); radiative capture (102); fission (18); total/prompt/delayed nubars (452, 455, 456); and (n,:math:*alpha*) cross section for 10B and 7Li. In this list, the number shown in parenthesis corresponds to the ENDF/B "mt numbers."

CE data are obtained for arbitrary energies by linear interpolation of discrete cross sections defined on a pointwise (PW) energy mesh. The PW energy mesh for a given nuclide is sufficiently fine that error introduced by linear interpolation between any two points is less than 0.1%. CRAWDAD also interpolates the CE data to the specific temperatures needed for the problem. The default temperature interpolation method uses square-root of temperature below 1200 Kelvin and a finite difference procedure above this temperature [CRAWDADHCML16].

7.7.2 CRAWDAD INPUT DATA

For standalone CRAWDAD execution, the user prepares the FIDO input deck as described below. However during a SCALE sequence computation, the XSProc module always executes CRAWDAD for CENTRM/PMC self-shielding calculations, and defines appropriate CRAWDAD parameter values based on specified CENTRM and PMC options. This is the recommended mode of operation. Some XSProc default values for CRAWDAD can be changed using keywords in the CENTRM DATA block; e.g., see parameters *mtout*= and *kernel*= in Sect. 7.4.4. Several options available for stand-alone execution cannot be controlled by keywords in the sequence runs, as these are set automatically

```
CRAWDAD STANDALONE INPUT

**************     DATA BLOCK 1

0$$   LOGICAL UNIT ASSIGNMENTS [4 entries.  Default values given in parentheses]

Entry Number  Variable Name   Description     Default Value
1     lcen    logical unit number of output CENTRM library    (81)
2     n17     logical unit for scratch        (17)
3     n18     logical unit for scratch        (18)
4     n19     logical unit for reading CE-KENO libraries      (88)

1$$   INTEGER PARAMETERS  [10 entries ]

1     num_nucs        number of PW nuclides to process        (1)
2     idtap   identifier placed on header of output CENTRM library    (66666)
3     iprt    print out option  (1)
              -1  no print out AT ALL
              0  hardly any print
              1  normal print
              2  debug print
4     obsolete feature
5     iterp           temperature interpolation method for PW cross sections (0)
              0  square-root-T interpolation for T<1200 K and finite difference for T>1200 K
              1  square-root-T interpolation for all temperatures
              2  finite difference interpolation for all temperatures
6     libth           create CENTRM thermal kernel library for bound moderators (1)
              0  no
              1  yes  (output kernel file is named lib_cen_kern)
7-10  N/A     extra integer parameters (not used)     (0)


1**   REAL PARAMETERS  [10 entries]

1     teps    tolerance on temperature differences  (5.0)
      ( temperatures within +/- "teps" are assumed equal)
2     tole    not implemented
3-10  N/A     extra real parameters (not used)        (0.0)

T       terminate data block 1
```

```
**************     DATA BLOCK 2
***** Repeat data block(s) 2 and 3, stacked "num_nucs" times to create a new
      CENTRM library containing specified temperatures and reaction types

2$$  NUCLIDE INFORMATION [5 entries]

Entry Number  Variable Name   Description     Default Value

1     za      zaid for this nuclide in PW XS library
2     lver    version number of evaluated nuclear data (e.g, 7 for ENDF/B-VII)
3     mod     desired mod number of evaluated nuclear  (-1)
              -1 => use latest mod
```

(continues on next page)

(continued from previous page)

```
4     inum    desired number of temperatures for this nuclide (0)
              0 - put all available temperatures on output CENTRM library
              n - include data at the "n" temperatures in 4** array
5     mtout   MTS to be included on output CENTRM PW library (2)
              0 - output PW data for all available MTs
              1 - output PW data only for default standard MTs:
              1, 2, 4, 102, 18, 452, 455, 456 for all materials; and  107 for 10B and 7Li
              2 - output standard MTs, plus inelastic levels and (n,2n)
              3 - standard MTs plus those listed in 5$$ array
              -3 - out all MTs EXCEPT those listed in 5$$
6     kmod    mod number for ENDF thermal scattering law data (-1)
              :math:`\geq` 0 - use cross section data with this thermal mod number
              -1 - use cross section data with latest thermal mod and kernel (if available)
              -2 - do not include bound kernel data (i.e., free-gas scattering will be used in CENTRM)
7     lsrc    Source of nuclear data  (0 only allowed at present)
              0/1/2/3/4 => ENDF/JEF/JENDL/BROND/CENDL

3** ENERGY LIMITS [2 entries]

1     pemin   minimum energy for PW data      (0.0001 eV)
2     pemax   maximum energy for PW data      (20 MeV)

T  terminate data block 2
```

```
**************     DATA BLOCK 3
***** Only enter if inum >0, or mtout= +/- 3 )

4**   DESIRED TEMPERATURES for this nuclide  [inum entries]
5$$   MT VALUES (if mtout = +/-3)   [always end with an "E"]

T    terminate data block 3

Optional  72 character title for the CENTRM library
```

7.7.3 CRAWDAD SAMPLE INPUT

Example 7.7.1 shows an example input file for standalone execution of CRAWDAD. The CRAWDAD output for this case is shown in Example 7.7.2. In more typical cases where CRAWDAD is executed automatically by the XSProc module as part of a SCALE sequence calculation, no CRAWDAD input is needed, but similar CRAWDAD output will be printed.

Example 7.7.1: CRAWDAD input generated by CSAS1 sample.

```
=crawdad
0$$   81   17   18    77         e
1$$   5    66666  0   0   2   1  e
1**   5.00E+00                   e
 t
2$$  8016  7   3   2   2   -1   0
  3**   1.00-03   1.30+04         2t
  4**   6.00+02   9.00+02         3t
2$$  13027  7   1   1   2  -1   0
  3**   1.00E-03   1.30E+04       2t
  4**   6.50E+02                  3t
2$$  92235  7   7   1   2   -1   0
  3**   1.00E-03   1.30E+04       2t
  4**   9.00E+02                  3t
2$$  92238  7   5   1   2   -1   0
  3**   1.00E-03   1.30E+04       2t
  4**   9.00E+02                  3t
2$$  1001   7   5   1   2    0   0
```

(continues on next page)

(continued from previous page)

```
  3**   1.00E-03   1.30E+04       2t
  4**   6.00E+02                  3t
end
, move the generated PW CENTRM library to execution directory
=shell
  mv ft81f001 $RTNDIR
end shell
' ..........................................................................
```

Example 7.7.2: Sample output edit from CRAWDAD.

```
A new centrm library has been written on unit number:   81
  The number of input nuclides was:                       5
  Number of Nuclides on output PW library:                5
  Directory containing input PW library files:    /scale/scale6.dev/data/cekenolib_7.0


                  Description of Output CENTRM Library

 Entry    ZA   Data Src  Vers No.  Mod No.  MT-Optn  Thermal ID    XS temperatures
 -----  -----  --------  --------  -------  -------  ----------    ---------------
   1     8016    endf       7         3        2            0        600.00
                                                                    900.00
   2    13027    endf       7         1        2            0        650.00
   3    92235    endf       7         7        2            0        900.00
   4    92238    endf       7         5        2            0        900.00
   5     1001    endf       7         5        2      7000001        600.00



   Nuclides in Problem-Dependent Thermal Kernel Library

            Library Identifier:     901
             Number of kernels:       1
   Maximum Order of Scattering:       6
 Maximum Number of Temperatures:      9

 Library Directory
 Nuclide     Identifier  Sigfree  File
 ----------  ----------  -------  ------------------------------------------
 h(h2o)         7000001    20.48  endf_b/vers7/1-0

 ============================================================================
 logical 18 (problem dependent centrm thermal kernel library)
 dataset name: /usr/tmp/xmw.9890/lib_cen_kernel
       volume:
 ============================================================================

 CRAWDAD has terminated normally
```

7.8 MCDANCOFF: MONTE-CARLO BASED DANCOFF FACTOR CALCULATION

L. M. Petrie, B. T. Rearden, A. M. Holcomb, and K. S. Kim

ABSTRACT

The MCDancoff program is used to calculate Dancoff factors in complicated, three-dimensional (3-D) geometries using Monte Carlo integrations. The geometries are standard SCALE geometry descriptions, with the current restriction that Dancoff factors can only be calculated for regions bounded by cuboids, spheres, or cylinders. Multiple Dancoff factors can be calculated with one input file. New AMPX one group library for MCDancoff was developed for SCALE 6.3 based on ENDF/B-VII.1.

ACKNOWLEDGMENT

This work was sponsored in part by Atomic Energy of Canada, Ltd. Development of new ENDF/B-VII.1 based AMPX one group library for SCALE 6.3 was sponsored by the US Nuclear Regulatory Commision (NRC). The contribution of S. J. Poarch in preparing this document is gratefully acknowledged.

7.8.1 INTRODUCTION

MCDancoff (Monte Carlo Dancoff) is a program that calculates Dancoff factors for complicated, three-dimensional geometries. Its input is a slight modification of a CSAS6 input file which uses the standard SCALE geometry as detailed for KENO-VI. The modifications to the input involve different input in the START data block describing which Dancoff factors are to be calculated. The calculation involves starting histories isotropically on the surface of the region for which the Dancoff factor is to be calculated and following the path of each history until it has encountered all the elements of the material in the region, or until it has exited the system. A one group cross-section library is used to determine the total cross sections of the mixtures in the problem. The one group library has been developed based on ENDF/B-VII.1 for SCALE 6.3 for which various reaction cross sections were obtained by energy group collapsing and elastic cross sections (MT=2) were replaced by potential cross sections.

A current restriction of MCDancoff is that it can only calculate Dancoff factors for regions bounded by cylinders, spheres, or cuboids. Other simple bodies could be added in the future, but a general bounding surface would be impractical.

The Dancoff factors are used in SCALE to correctly self-shield multigroup cross sections for a given problem; either as input to BONAMI or to determine an equivalent cell for CENTRM. This is most typically accomplished through the MORE DATA and CENTRM DATA blocks.

The Dancoff factors are actually calculated by a modified version of the KENO-VI code called KENO_Dancoff. All printed output from these calculations is suppressed by default. If there is a need to see this output (for example, to find an error message), it can be turned on by setting an environment variable **print_dancoff=yes**.

7.8.2 INPUT DATA DESCRIPTION

MCDancoff input data is the same as CSAS6 input data with the following exceptions. A special one group cross-section library will be used. It can be specified as **xn01** in the input but will be set to this if anything else is entered for the library. Because MCDancoff is running a fixed source problem, and the Dancoff factor doesn't need to be calculated with the same accuracy as an eigenvalue, there are useful changes that can be made to the parameters in the PARAMETER data block. Sect. 7.8.3 discusses this in more detail. Finally, the START data block is used to define which Dancoff factors will be calculated. This data block is defined below.

```
READ START Begins the data block

  1. dancoff

  begins defining a new Dancoff factor. Always start
  relative to the global unit in the geometry.

  2. array

  step into an array contained in the current unit - followed
  by karray, nbx, nby, nbz where karray is the
  region containing the array in the current unit, nbx is the x
  position in the array of the next unit, nby is the y position
  in the array of the next unit, and **nbz** is the z position in
  the array of the next unit.

  3. hole

  step into a hole contained in the current unit - followed by
  nhole, the hole number relative to the current unit.

  4. unit

  final unit in the nesting chain - followed by **nn**, the
  unit number

  5. region

  region to calculate the Dancoff factor for - followed by
  k, the relative geometry word in unit nn defining the
  outer bound of the region.

  6. nst

if input, must be 0 (defaults to 0).
Repeat 2 and 3 to get from the global unit to the final unit nn.
Repeat 1-5 for each Dancoff factor to be calculated.

END START Ends the data block
```

7.8.3 CALCULATION AND USE OF 3D DANCOFF FACTORS

1. The 3-D Dancoff factors are computed with KENO-VI geometry. If beginning with CSAS5 model, use C5TOC6 to convert to CSAS6.
2. Change sequence name from CSAS6 to MCDancoff and change cross-section library to **xn01**.
3. Input appropriate parameter data.

   Since the Dancoff calculation is fixed source integration, there is no need to skip generations, and **nsk** should be set to 0. Since small changes to the Dancoff have very minor effects on the cross sections, fewer histories are probably needed for calculating the Dancoff than for calculating *keff*. Thirty thousand histories divided as 100 generations of 300 histories per generation has produced Dancoff factors with deviations of less than 1 percent. It may be advantageous to turn off plots at this point. Since the same parameters can be entered more than once, with the final entry being the one used, adding a separate record with these values immediately before the **end parameter** keywords would override the original KENO-VI parameters.

   Example:

```
read param
        .........
  nsk=0 npg=300 gen=100 nub=no fdn=no flx=yes plt=no
end param
```

4. Identify the region for which Dancoff factors are desired in START data.

   The start type needs to be set to **0** for the Dancoff calculation (this is the default). All KENO-VI START data should be removed or commented out by placing an apostrophe in column 1. Each region for which a Dancoff calculation is desired then starts with the keyword **dancoff**. This is followed by data that specify the relationship of the global unit to the specific geometry description of the region. If the region is nested inside an array, then the keyword, **array**, is specified, followed by four integers. The first integer is the indices of the media record specifying the array relative to the current unit. The next 3 integers are the X, Y, and Z indexes of the position of the next unit in the array. If the region is nested in a hole, then the keyword, **hole**, is specified, followed by the relative count of the correct hole in the unit. The preceding data are repeated (in the correct nesting order starting with the global unit) until reaching the unit where the region is located. Then the keyword, **unit**, followed by the unit number is given, followed by the keyword, **region**, followed by the relative index of the geometry keyword describing the desired region with respect to that unit. Currently, only cylinders, spheres, and cuboids are programmed for calculating Dancoff factors.

   Examples:

```
read start
  nst=0
  dancoff    hole 1    unit=1  reg=1
end start

read start
   dancoff  array  1 1 1 1  array  1 17 17 2  unit 10  region 1
end start
```

5. Execute MCDANCOFF *filename*.input file like any other SCALE input file.

6. Examine *filename*.dancoff file, which will contain Dancoff factors for each nuclide in the specified region

```
index        nuclide        dancoff      deviation
                  1          92234    3.36340E-01    1.81134E-03
                  2          92235    3.36340E-01    1.81134E-03
                  3          92236    3.36340E-01    1.81134E-03
                  4          92238    3.36340E-01    1.81134E-03
                  5           8016    1.00000E+00    0.00000E+00
```

7. Once all desired Dancoff factors are obtained, return to original model and enter CENTRM DATA for each cell with dan2pitch(mix) specified.

```
read celldata
  latticecell triangpitch fuelr=0.633  1 gapr=0.637 0 cladr=0.675 10 hpitch=0.867  14 end
centrm data
  dan2pitch(1)=0.336
end centrm
```

8. If executing TSUNAMI-3D, additional steps are necessary because TSUNAMI-3D does not treat the dan2pitch input parameter.

Return to the original TSUNAMI-3D input file and replace the sequence name to "CSAS-MG PARM=CHECK" and delete all data after the unit cell data to quickly obtain revised pitch values. (Note: CSAS will not modify cell dimensions to more than 20 cm, so a revised moderator density may need to be entered to obtain the desired Dancoff factor.) Search for the word "*desired*" in output file to find new pitch values for each cell.

```
unit cell  =    1
  original pitch               = 1.7340E+00
  Dancoff for orig pitch       = 2.9728E-01
  desired Dancoff              = 3.3600E-01

pitch to produce desired Dancoff= 1.6845E+00
```

9. Enter revised pitch and revised moderator density (for cell calculation only, not for geometry model) in TSUNAMI model.

7.8.4 EXAMPLE CASE

The following is a contrived case to illustrate an input file using both holes, arrays, and multiple sets of Dancoff factors (although both factors apply to the same pin, so only one set can be used). The case represents two fuel assemblies in a cylindrical tank, each assembly having a poisoned central pin, and four water holes. The Dancoff factors are calculated for each central pin. The input file is listed in Example 7.8.1.

Example 7.8.1: Example input file.

```
=mcdancoff
sample case demonstrating calculating Dancoff factors
xn01
read composition
  uo2     1  den=10.38  1  294  92234 .0303  92235 4.7378  92236 .1364  92238 95.0955  end uo2
  zirc4   2             1  294  end zirc4
  h2o     3             1  294  end h2o
  uo2     4  den=10.08  1  294  92234 .0303  92235 4.7378  92236 .1364  92238 95.0955  end uo2
  gd      4  den= 0.3   1  294  end gd
end composition
read param
  nsk=0 gen=100 npg=300
end param
read geometry
  unit 1
    com=!fuel pin!
    cylinder   10  0.395  40.0  -40.0
    cylinder   20  0.410  40.0  -40.0
    cylinder   30  0.470  40.0  -40.0
    cuboid     40  4p0.65  2p40.0
    media      1  1  10
    media      0  1  20 -10
    media      2  1  30 -20
    media      3  1  40 -30
    boundary   40
  unit 2
    com=!water hole!
    cuboid     40  4p0.65  2p40.0
    media      3  1  40
    boundary   40
  unit 3
    com=!unit containing a 2x2 array of fuel pins!
    cuboid     10  4p1.30  2p40.0
    array      1  10  place 1 1 1 -0.65 -0.65 0.0
    boundary   10
  unit 4
```

(continues on next page)

(continued from previous page)

```
  com=!unit containing a 1x2 array of fuel pins!
  cuboid     10  2p0.65  2p1.30  2p40.0
  array      2  10  place 1 1 1 0.0 -0.65 0.0
  boundary   10
unit 5
  com=!unit containing a 2x1 array of fuel pins!
  cuboid     10  2p1.30  2p0.65  2p40.0
  array      3  10  place 1 1 1 -0.65 0.0 0.0
  boundary   10
unit 6
  com=!unit containing a 5x5 array of fuel pins!
  cuboid     10  4p3.25  2p40.0
  array      4  10  place 2 2 1 0.0 0.0 0.0
  boundary   10
unit 7
  com=!unit containing a 5x5 array of fuel pins - water hole in the middle!
  cuboid     10  4p3.25  2p40.0
  array      5  10  place 2 2 1 0.0 0.0 0.0
  boundary   10
unit 8
  com=!unit containing a 5x5 array of fuel pins - poisoned pin in the middle!
  cuboid     10  4p3.25  2p40.0
  array      6  10  place 2 2 1 0.0 0.0 0.0
  boundary   10
unit 9
  com=!poisoned fuel pin!
  cylinder   10  0.395  40.0  -40.0
  cylinder   20  0.410  40.0  -40.0
  cylinder   30  0.470  40.0  -40.0
  cuboid     40  4p0.65  2p40.0
  media      4  1  10
  media      0  1  20 -10
  media      2  1  30 -20
  media      3  1  40 -30
  boundary   40
unit 10
  com=!unit containing a 15x15 fuel assembly!
  cuboid     10  4p9.75  2p40.0
  array      7  10  place  2 2 1 0.0 0.0 0.0
  boundary   10
global
unit 11
  com=!global unit with 2 fuel assemblies!
  cylinder   10  25.0  60.0  -60.0
  hole       10  origin x=-10.0
  hole       10  origin x= 10.0
  media      3  1  10
  boundary   10
end geometry
read array
ara=1 typ=square nux=2 nuy=2 nuz=1  fill f1 end fill
ara=2 typ=square nux=1 nuy=2 nuz=1  fill f1 end fill
ara=3 typ=square nux=2 nuy=1 nuz=1  fill f1 end fill
ara=4 typ=square nux=3 nuy=3 nuz=1  fill 3 4 3 5 1 5 3 4 3 end fill
ara=5 typ=square nux=3 nuy=3 nuz=1  fill 3 4 3 5 2 5 3 4 3 end fill
ara=6 typ=square nux=3 nuy=3 nuz=1  fill 3 4 3 5 9 5 3 4 3 end fill
ara=7 typ=square nux=3 nuy=3 nuz=1  fill 7 6 7 6 8 6 7 6 7 end fill
end array
read start
' first Dancoff - calculate for the poisoned fuel pin in unit 9 for the x=-10 assembly
dancoff
' hole 1 is unit 10 at x=-10
  hole 1
' array in first region of unit 10 is array 7 - 2 2 1 position is unit 8
  array 1 2 2 1
```

(continues on next page)

(continued from previous page)

```
  ' array in first region of unit 8 is array 6 - 2 2 1 position is unit 9
    array 1 2 2 1
  '  cylinder labeled 10 in unit 9 is the first region
    unit 9   region 1
  ' second Dancoff - calculate for the poisoned fuel pin in unit 9 for the x=+10 assembly
  dancoff
  ' hole 2 is unit 10 at x=+10
    hole 2
  ' array in first region of unit 10 is array 7 - 2 2 1 position is unit 8
    array 1 2 2 1
  ' array in first region of unit 8 is array 6 - 2 2 1 position is unit 9
    array 1 2 2 1
  '  cylinder labeled 10 in unit 9 is the first region
    unit 9   region 1
  end start
  end data
  end
```

This input file creates two files of Dancoff factors. The first such file is listed in Example 7.8.2.

Example 7.8.2: Output file of Dancoff factors.

```
Unit 9 at global x -1.00000E+01  y  0.00000E+00  z  0.00000E+00
          index        nuclide        dancoff      deviation
              1          92234    2.20873E-01    1.03436E-03
              2          92235    2.20873E-01    1.03436E-03
              3          92236    2.20873E-01    1.03436E-03
              4          92238    2.20873E-01    1.03436E-03
              5           8016    9.64748E-01    4.28121E-04
              6          64000    2.82254E-10    3.61320E-11
```

The second file is statistically the same, as it solved for the mirror image Dancoff factor.

7.9 CAJUN: MODULE FOR COMBINING AND MANIPULATING CENTRM CONTINUOUS-ENERGY LIBRARIES

L. M. Petrie and N. M. Greene

7.9.1 INTRODUCTION

CAJUN is a program used to combine continuous-energy (CE) cross section libraries for use in the cross-section processing codes CENTRM and PMC. It is used primarily by SCALE sequences when processing pointwise cross section data for DOUBLEHET unit cells with the XSProc module (see Sect. 7.1.3), although it can also be run standalone in conjunction with CRAWDAD. CAJUN combines multiple CE libraries into a single library, adds selected nuclides from one library into another library, deletes nuclides from a specified library, or renames nuclides in a library. CAJUN performs an analogous function for CE libraries as AJAX does for multigroup libraries.

In order for input CE libraries to be read by CAJUN, they must be assigned a unit number from 1 to 99. The SHELL module can be used to link individual CE libraries to appropriate file names that are accessible by unit number.

7.9.2 CAJUN INPUT DATA

Input data for CAJUN is read into the program using FIDO type input. The data is divided into three data blocks. The first data block provides the output file number and number of libraries processed. The second data block provides input library numbers, number of nuclides in each library, and whether nuclides are selected by MAT or ZA number. The third data block provides current nuclide MAT or ZA numbers and new nuclide MAT or ZA numbers. Detailed description of the CAJUN input data is provided below.

```
Data Block 1

  0$$         unit assignments (1)

      1.      lcen - logical unit number of output CE CENTRM library (1)

  1$$         number of files to process (2)

      1.      nfile - the number of CENTRM CE libraries to process (1)
      2.      idtap - identifier for the new library (0)

      T       terminate data block 1

**************************************************************************
 *** repeat data block(s) 2 and 3 "nfile" times to create new library
**************************************************************************

Data Block 2

      2$$     file selection and treatment option (4)

      1.      log - logical unit number of input CENTRM CE library (77)
      2.      inum - number of nuclides selected from this library (0)
              0 - select all nuclides on the library as is.
              -n - select all nuclides on the library as is except for those indicated in the 3$$ array.
              n - select the "n" nuclides on the library listed in the 3$$ array.
      3.      iopt - select nuclides by 'mat' or 'nza' number   (1)
              0 - mat
              1 - nza (default)
      4.      nsq - sequence number of file opened on unit LOG (1)

      T       terminate data block 2

Data Block 3 (enter if inum is non-zero)

      3$$     nuclide selection list (inum)
              enter a positive identifier to select a nuclide.
              enter a negative identifier to exclude a nuclide.

      4$$     new nuclide identifiers (inum)
              enter the new nuclide identifiers in the locations corresponding
              to the positive identifier entry in the 3$ array.

      5$$     version of the data (-1,0,1,2,3,4/unk,ENDF,JEF,JENDL,BROND,CENDL)  (inum)

      6$$     8-Character Identifier for Thermal Kinematics Data  (inum)

      7$$     ZA-override Values.  Non-zero values will replace the ZA values in the
              Header Record.  The ZA values in the Data Directory records are not changed.  (inum)

      T       terminate data block 3
```

7.9.3 CAJUN I/O UNITS

CAJUN requires the following I/O devices.

Unit No.	Purpose
5	Standard definition input
6	Output
18	Scratch file

8. MONTE CARLO TRANSPORT

Introduction by B. T. Rearden

SCALE Monte Carlo transport capabilities enable criticality safety, shielding, depletion, and sensitivity and uncertainty analysis [Monte-CarloRPP+14]. SCALE provides separate Monte Carlo capabilities for eigenvalue neutronics and fixed-source coupled neutron-gamma calculations, in the KENO code [Monte-CarloGPJD+11] and fixed-source coupled neutron-gamma calculations in the Monaco code [Monte-CarloPep11] Although the eigenvalue and fixed-source capabilities are provided in separate codes, many capabilities are shared between them, including physics and geometry packages. The foundational features shared between the codes are described below, with specific implementations provided in subsequent sections. Generally, the use of the Monte Carlo transport solvers in SCALE are best accessed through the capability-specific sequences: CSAS for criticality safety, MAVRIC for shielding, TRITON for depletion, TSUNAMI-3D for sensitivity and uncertainty analysis, and MCDancoff for three-dimensional Dancoff factor calculations.

Multigroup Physics

The multigroup treatment implemented in SCALE has been in use since the 1960s and provides efficient, effective solutions with superior runtime performance. Problem-dependent multigroup cross section data are temperature interpolated and resonance self-shielded by other SCALE modules before they are used in each Monte Carlo calculation. Without proper resonance self-shielding, accurate multigroup calculations would not be possible for thermal or intermediate energy spectrum systems. After self-shielding has been accomplished and the two-dimensional expansions have been summed into a Legendre expansion of the total group-to-group transfer arrays, individual nuclide cross sections are multiplied by their densities and summed into mixtures. These mixture cross sections can then be used by the deterministic transport codes for their calculations. The Monte Carlo codes convert the Legendre expansion of the transfer arrays into probability distributions for the group-to-group transfers and for the discrete scattering angles and probabilities that preserve the moments of the Legendre expansion of each group-to-group transfer. These transfer probabilities, angles, and angle probabilities are then transformed so that the new group and angle of scatter are efficiently selected through two random numbers with only one multiplication and one addition operation. If the selected new group is negative, it is reset to positive, and the new direction is chosen isotropically. If the problem is run with P_1 scattering, the scattering angle is chosen from a continuous distribution. For higher order scattering, the polar scatter angle is discrete, and the azimuthal angle is randomly selected from a uniform distribution. Multigroup physics is implemented for neutron, photon, and neutron-photon coupled particle transport modes.

Continuous-energy Physics

The continuous energy treatment in SCALE provides high resolution solution strategies with explicit physics representation. The continuous energy data represent thermal scattering using free gas and $s(\alpha, \beta)$, with explicit point-to-point data provided through the thermal region. The resolved resonance region is represented by pointwise data where the energy point density is optimized for each reaction of each nuclide. Data in the unresolved resonance region are represented by probability tables, and data above the unresolved region implement pointwise data with explicit point-to-point representation for secondary particles. Photon yield data represent each discrete photon. Continuous energy physics contains non-transport data handling to support various flux, reaction rate, point detector tallies, and sensitivity analysis. In addition, continuous energy data are converted from a double differential data format to a lab format in a process where fast look-up tables are provided during library generation. In SCALE 6.0–6.1, calculations are performed only at temperatures available on the data libraries by selecting the library temperature nearest to the desired temperature for

the calculation. Resonance upscattering techniques are implemented via the Doppler Broadened Rejection Correction method [Monte-CarloHMGR13]. With SCALE 6.2, problem-dependent continuous energy cross sections at the user specified temperature are generated at the beginning of the calculation. Continuous energy physics is implemented for neutron, photon, and neutron-photon coupled particle transport modes.

Geometry Packages

Two variants of KENO provide identical solution capabilities with different geometry packages. KENO V.a implements a simple and efficient geometry package sufficient for modeling many systems of interest to criticality safety and reactor physics analysts. KENO-VI implements the SCALE Generalized Geometry Package (SGGP), which provides a quadratic-based geometry system with much greater flexibility in solution modeling. Monaco implements only the SGGP geometry package. Both packages are based on solid bodies organized into reusable objects called *units* that are constructed of material *regions*. Units can be conveniently arranged in rectangular or hexagonal *arrays* of repeating units. Additionally, nesting is available so that one unit can contain another unit as a *hole*, or an array can be nested inside of a unit, which itself can be repeated in another array. There is no limit to the number of nesting levels available, so very complex systems can be quickly generated.

KENO V.a models are constructed from regions of specific shapes following strict rules which provide great efficiency in geometry tracking. Allowed shapes are cubes, cuboids (rectangular parallelepipeds), spheres, cylinders, hemispheres, and hemicylinders. These shapes must be oriented along orthogonal axes, and they can be translated, but they cannot be rotated. A major restriction applied to KENO V.a geometry is that intersections are not allowed, and each region of a unit must fully enclose the preceding region. An exception to this rule is in the use of *holes* through which many units can be placed within an enclosing unit. However, there is a runtime penalty in geometry tracking for this flexibility, so this feature should be used judiciously. KENO V.a provides *rectangular* arrays where the outer body of each unit contained in the array must have a cuboidal shape, and adjacent faces must have the same dimensions. The entire array must be fully enclosed by the region in which it is placed.

SGGP is a quadratic-based geometry system that provides predefined bodies including cone, cuboid, cylinder, dodecahedron, ecylinder (elliptical cylinders), ellipsoid, hexprism, hopper (truncated pyramid), parallelepiped, planes, rhombohedron, rhexprism (rotated hexprisms), sphere, and wedge. Bodies not directly provided with SGGP can be constructed from quadratic surfaces defined with coefficients entered by the user. All bodies and surfaces can be rotated and translated to any orientation and position within their respective unit. SGGP also provides intersecting regions.

SGGP arrays may be composed of cuboids, hexprisms, rhexprisms, or dodecahedrons. Like KENO V.a, the faces of adjacent units in an array must have the same dimensions. An array boundary must be specified for each array, and only the portion of the array within the boundary is considered a part of the system. Also, the specified array must fill the entire volume in the specified array boundary. The array boundary may be any shape that can be specified using quadratic equations.

The use of holes is more flexible in SGGP than in KENO V.a. Within a unit, holes cannot intersect other holes or the unit boundary, but they can intersect region boundaries. The use of holes is not necessary to build complex geometries; they are used primarily to more efficiently build complex geometries and improve the tracking efficiency of the simulation. In SGGP the distance to each surface in the unit must be calculated after each collision. By moving some of the surfaces in a unit into another unit that is included as a hole, all the surfaces in the hole unit except the outer boundary are removed from the containing unit. The judicious use of holes in SGGP can significantly speed up the calculation.

Eigenvalue Analysis

KENO performs eigenvalue calculations for neutron transport primarily to calculate multiplication factors and flux distributions of fissile systems in continuous energy and multigroup modes. Both codes allow explicit geometric representation with their respective geometry packages. KENO provides a multigroup adjoint capability which is especially useful for sensitivity analysis. KENO implements standard variance reduction techniques such as implicit capture, splitting, and Russian roulette.

The initial fission source distribution in KENO can be specified with nine options. These options include the default option of a uniform distribution throughout the fissile material; an axially varying distribution input by the user or defined as cos(Z) or (1-cos(Z)):sup:2, where Z is the axial position; several options to initialize the source at a given position (within a given volume, a given unit, or a unit at a specified array index); or to specifically provide the coordinates of each starting point.

KENO approximates the real *keff* variance using an iterative approach and lagging covariance data between generations [Monte-CarloUMN97]. KENO provides a χ^2 test for the normality of *keff* and provides plots of k_{eff} by active and inactive generations. KENO reports a *best estimate* of k_{eff} that is computed as the minimum variance of k_{eff} based on generations skipped and generations run.

KENO provides track-length tallies for scalar flux and angular flux moments needed for sensitivity analysis. Additionally, tallies are provided for reaction rates, with isotopic tallies available only in CE calculations. KENO also provides mesh tallies based on a user-input orthogonal grid.

Matrix k_{eff} calculations provide an additional method of calculating the k_{eff} of the system. Cofactor k_{eff} and source vectors, which describe the contribution to the system $k_{eff}$from each unit, hole, or array, are available.

KENO provides plots of *keff*by generation and average k_{eff} for visual inspection of source convergence, followed by a χ^2 statistical assessment of convergence. Fission source convergence diagnostic techniques are implemented in KENO to provide improved confidence in the computed results and to reduce the simulation time for some cases. Confirming the convergence of the fission source distribution is especially useful to avoid the false convergence of k_{eff} that can be caused by insufficient sampling of important portions of the system [Monte-CarloUB05] KENO source convergence diagnostics rely on Shannon entropy statistics of the mesh-based fission source data.

Parallel computation capabilities are available in both versions of KENO to provide reductions in wall clock time, especially for sensitivity analysis or Monte Carlo depletion on computer clusters. By introducing a simple master-slave approach via message passing interface (MPI), KENO runs different random walks concurrently on the replicated geometry within the same generation. The fission source and other tallied quantities are gathered at the end of each generation by the master process, and then they are processed either for final edits or next generations.

Shielding Analysis

Monaco is a fixed-source Monte Carlo shielding code that calculates neutron and photon fluxes and response functions for specific geometry regions, point detectors, and mesh tallies. Monaco has variance reduction capabilities, such as source biasing and weight windows, which can be automated via the MAVRIC sequence. MAVRIC performs radiation transport on problems that are too challenging for standard, unbiased Monte Carlo methods. Monaco provides multiple methods to enter the radioactive source descriptions. Spatial distribution options include volumetric sources and mesh sources which can be generated by other codes such as KENO. Energy distributions can be entered by the user or imported directly from emission data provided by ORIGEN. Spent fuel analysis is simplified through direct coupling with the ORIGEN binary concentration files.

8.1 KENO: A MONTE CARLO CRITICALITY PROGRAM

K. B. Bekar, C. Celik, M. E. Dunn,[1] *S. Goluoglu,*[1] *D. F. Hollenbach,*[1] *N. F. Landers,*[1] *C. M.Perfetti,*[1] *L. M. Petrie,*[1] *B. T. Rearden,*[1]

KENO is a three-dimensional (3D) Monte Carlo criticality transport program developed and maintained for use as part of the SCALE Code System. It can be used as part of a sequence or as a standalone program. There are two versions of the code currently supported in SCALE. KENO V.a is the older of the two. KENO-VI contains all current KENO V.a features plus a more flexible geometry package known as the SCALE Generalized Geometry Package. The geometry package in KENO-VI is capable of modeling any volume that can be constructed using quadratic equations. In addition, such features as geometry intersections, body rotations, hexagonal and dodecahedral arrays, and array boundaries have been included to make the code more flexible.

The simpler geometry features supported by KENO V.a allow for significantly shorter execution times than KENO-VI, while the additional geometry features supported in KENO-VI make the code appropriate for cases where geometry modeling is not possible with KENO V.a. In particular, KENO-VI allows intersections, body truncations with planes, and a much wider variety of geometrical bodies. KENO-VI also has the ability to rotate bodies so that volumes no longer must be positioned parallel to a major axis. Hexagonal arrays are available in KENO-VI and dohecahedral arrays enable the code to model pebble bed reactors and other systems composed of close packed spheres. The use of array boundaries makes it possible to fill a non-cuboidal volume with an array, specifying the boundary where a particle leaves and enters the array.

Except for geometry capabilities, the two versions of KENO share most of the computational capabilities and the input flexibility specific to most SCALE modules. They can both operate in multigroup or continuous energy mode, run as standalone codes, or integrated in computational sequences such as CSAS, TSUNAMI-3D, or TRITON. Both versions of the code are continually updated and are written in FORTRAN 90.

Computational capabilities shared by the two versions of KENO include the determination of k-effective, neutron lifetime, generation time, energy-dependent leakages, energy- and region-dependent absorptions, fissions, the system mean-free-path, the region-dependent mean-free-path, average neutron energy, flux densities, fission densities, reaction rate tallies, mesh tallies, source convergence diagnostics, problem-dependent continuous energy temperature treatments, parallel calculations, restart capabilities, and many more.

[1]Formerly with Oak Ridge National Laboratory

8.1.1 ACKNOWLEDGMENTS

Many individuals have contributed significantly to the development of KENO. Special recognition is given to G. E. Whitesides, former Director of the Computing Applications Division, who was responsible for the concept and development of the original KENO code. He has also contributed significantly to some of the techniques used in both KENO versions. The late J. T. Thomas offered many ideas that have been implemented in the code. R. M. Westfall, retired from ORNL, provided early consultation, encouragement, and benchmarks for validating the code. The special abilities of J. R. Knight, retired from ORNL, contributed substantially to debugging early versions of the code. S. W. D. Hart was instrumental in implementing continuous energy temperature treatments. W. J. Marshall has provided substantial validation and quality assurance reviews. Appreciation is expressed to C. V. Parks and S. M. Bowman for their support of KENO and the KENO3D visualization tool. The late P. B. Fox provided many of the figures in this document. D. Ilas, B. J. Marshall, and D. E. Mueller consolidated the previous KENO V.a and KENO-VI manuals into this present form. The efforts of L. F. Norris (retired), W. C. Carter (retired), S. J. Poarch, D. J. Weaver (retired), S. Y. Walker and R. B. Raney in preparing this document are gratefully acknowledged.

The authors thank the U. S. Nuclear Regulatory Commission and the DOE Nuclear Criticality Safety Program for sponsorship of the continuous energy, source convergence diagnostics, and grid geometry features in the current version.

8.1.2 INTRODUCTION TO KENO

KENO, a functional module in the SCALE system, is a Monte Carlo criticality program used to calculate k_{eff}, fluxes, reaction rates, and other data for three-dimensional (3-D) systems. Special features include multigroup or continuous energy mode, simplified data input, the ability to specify origins for spherical and cylindrical geometry regions, a P_n scattering treatment, and restart capability.

The KENO data input features flexibility in the order of input. The only restrictions are that the sequence identifier, title, and cross section library must be entered first. A large portion of the data has been assigned default values found to be adequate for many problems. This feature enables the user to run a problem with a minimum of input data.

In addition to the features listed above, KENO-VI uses the SCALE Generalized Geometry Package (SGGP), which contains a much larger set of geometrical bodies, including cuboids, cylinders, spheres, cones, dodecahedrons, elliptical cylinders, ellipsoids, hoppers, parallelepipeds, planes, rhomboids, and wedges. The code's flexibility is increased by allowing: intersecting geometry regions; hexagonal, dodecahedral, and cuboidal arrays; bodies and holes rotated to any angle and translated to any position; and a specified array boundary that contains only that portion of the array located inside the boundary. Users should be aware that the added geometry features in KENO-VI can result in significantly longer run times than KENO V.a. A KENO-VI problem that can be modeled in KENO V.a will typically run about four times as long with KENO-VI as it does with KENO V.a. Therefore KENO-VI is not a replacement for KENO V.a, but rather an additional version for more complex geometries that could not be modeled previously.

Blocks of input data are entered in the form

```
READ XXXX** *input_data* ``END XXXX
```

where `XXXX` is the keyword for the type of data being entered. The types of data entered include parameters, geometry region data, array definition data, biasing or weighting data, albedo boundary conditions, starting distribution information, the cross section mixing table, extra one-dimensional (1-D) (reaction rate) cross section IDs for special applications, energy group boundaries for tallying in the continuous energy mode, a mesh grid for collecting flux moments, and printer plot information.

A block of data can be omitted unless it is needed or desired for the problem. Within the blocks of data, most of the input is activated by using keywords to override default values.

The treatment of the energy variable can be either multigroup or continuous. Changing the calculation mode from multigroup to continuous energy or vice versa is established by simply changing the cross section library used. All available calculated entities in the multigroup mode can also be calculated in the continuous energy mode. If the calculated entity is energy or group dependent, it is automatically tallied into the appropriate group structure in the continuous energy mode.

The KENO V.a geometry input consists of spheres, hemispheres, cylinders, hemicylinders, and cuboids. Although the origin of the cylinders, hemicylinders, spheres, and hemispheres is zero by default, they may be specified to any value that will allow the geometry to fit in the problem. This feature allows the use of nonconcentric cylindrical and spherical shapes and provides a great deal of freedom in positioning them. Another feature that expands the generality of the code is the ability to place the cut surface of the hemicylinders and hemispheres at any distance between the radius and the origin.

An additional convenience is the availability of an alternative method for specifying the array definition unit-location data. This method uses FIDO-like options for filling the array.

As mentioned above, KENO-VI uses the SGGP, which contains a much more flexible geometry package than the one in KENO V.a. In KENO-VI, geometry regions are constructed and processed as sets of quadratic equations. A set of geometric shapes (including all of those used in KENO V.a plus others) is available in KENO-VI, as well as the ability to build more complex geometric shapes using sets of quadratic equations. Unlike KENO V.a, KENO-VI allows intersections between geometry regions within a unit, and it provides the ability to specify an array boundary that intersects the array.

The most flexible KENO V.a geometry features are the "ARRAY-of-ARRAYs" and "HOLEs" capabilities. The ARRAY-of-ARRAYs option allows the construction of ARRAYs from other ARRAYs. The depth of nesting is limited only by computer space restrictions. This option greatly simplifies the setup for ARRAYs involving different UNITs at different spacings. The HOLE option allows a UNIT or an ARRAY to be placed at any desired location within a geometry region. The emplaced UNIT or ARRAY cannot intersect any geometry region and must be wholly contained within a region. As many HOLEs as will snugly fit without intersecting can be placed in a region. This option is especially useful for describing shipping casks and reflectors that have gaps or other geometrical features. Any number of HOLEs can be described in a problem, and HOLEs can be nested to any depth.

The primary difference between the KENO V.a and KENO-VI geometry input is the methodology used to represent the geometry/material regions in a unit. KENO-VI uses two geometry records (cards) to describe a region. The first record, called the GEOMETRY record, contains the geometry (shape) keyword, region boundary definitions, and any geometry modification data. Using geometry modification data, regions can be rotated and translated to any angle and position within a unit. The second record, the CONTENT record, contains the MEDIA keyword; the material, HOLE, or ARRAY ID number; the bias ID number; and the region definition vector. KENO-VI requires that a GLOBAL UNIT be specified in all problems, including single unit problems.

In addition to the *cuboidal* ARRAYs available in KENO V.a, *hexagonal* ARRAYs and *dodecahedral* ARRAYs can be directly constructed in KENO-VI. Also, the ability to specify an ARRAY boundary that intersects the ARRAY makes it possible to construct a lattice in a cylinder using one ARRAY in KENO-VI instead of multiple ARRAYs and HOLEs as would be required in KENO V.a.

Anisotropic scattering is treated by using discrete scattering angles. The angles and associated probabilities are generated in a manner that preserves the moments of the angular scattering distribution for the selected group-to-group transfer. These moments can be derived from the coefficients of a P_n Legendre polynomial expansion. All moments through the 2n - 1 moment are preserved for n discrete scattering angles. A one-to-one correspondence exists such that n Legendre coefficients yield n moments. The cases of zero and one scattering angle are treated in a special manner. Even when the user specifies multiple scattering angles, KENO can recognize that the distribution is isotropic, and therefore KENO selects from a continuous isotropic distribution. If the user specifies one scattering angle, the code selects the scattering angle from a linear function if it is positive between -1 and +1, and otherwise it performs semicontinuous scattering by picking scattering angle cosines uniformly over some range between -1 and +1. The probability is zero over the rest of the range.

The KENO restart option is easy to activate. Certain changes can be made when a problem is restarted, including using a different random sequence or turning off certain print options such as fluxes or the fissions and absorptions by region.

KENO can also compute angular fluxes and flux moments in multigroup calculations, which are required to compute scattering terms for generation of sensitivity coefficients with the SAMS module or the TSUNAMI-

3D sequence. Fluxes can also be accumulated in a Cartesian mesh that is superimposed over the user-defined geometry in an automated manner.

KENO can perform Monte Carlo transport calculations concurrently on a number of computational nodes. By introducing a simple master-slave approach via MPI, KENO runs different random walks concurrently on the replicated geometry within the same generation. Fission source and other tallied quantities are gathered at the end of each generation by the master process and are then processed either for final edits or subsequent generations. Code parallel performance is strongly dependent on the size of the problem simulated and the size of the tallied quantities.

8.1.3 KENO DATA GUIDE

KENO may be run stand alone or as part of a SCALE criticality safety or sensitivity and uncertainty analysis sequence. If KENO is run stand alone in the multigroup mode, cross section data can be used from an AMPX [DG02] working format library or from a Monte Carlo format cross section library. If KENO uses an AMPX working format library, a mixing table data block must be entered. If a Monte Carlo format library is used, a mixing table data block is not entered, and the mixtures specified in the KENO geometry description must be consistent with the mixtures created on the Monte Carlo format library file.

If KENO is run stand alone in the continuous energy mode, a mixing table data block must be provided unless the restart option is used.

If KENO is run as part of a SCALE sequence, the mixtures are defined in the sequence input; therefore, a mixing table data block cannot be entered in KENO. Furthermore, the mixture numbers used in the KENO geometry description must correspond to those defined in the composition data block of the sequence input. To use a cell-weighted mixture in KENO, the keyword `CELLMIX=`, followed by a unique mixture number, must be specified in the cell data input block of the sequence. Note that cell data are applicable only in the multigroup mode. The mixture number used in the KENO input is the unique mixture number immediately following the keyword `CELLMIX`. A cell-weighted mixture is available only in SCALE sequences that use XSDRN to perform a cell-weighting calculation using a multigroup cross section library. Table 8.1.1 through Table 8.1.16 summarize the KENO input data blocks. These input data blocks are discussed in detail in the following sections. See CSAS, TRITON, and TSUNAMI manuals for more details and tips about how KENO is used as part of these sequences.

To run KENO in parallel (standalone execution), the user must provide the module name with the `%` prefix in the input file (e.g., `=%kenovi`), and provide the required arguments in the command line for parallel execution. The `%` prefix is not required if KENO is run as part of a SCALE sequence. Sequences such as CSAS, TRITON, and TSUNAMI-3D automatically initiate parallel KENO execution if the user provides the required arguments in the command line while running this code.

Warning: KENO can be run in parallel if SCALE has been built with MPI. SCALE pre-built binaries disseminated with each SCALE release are usually not MPI-enabled binaries.

Table 8.1.1: Summary of KENO parameter data

PARAMETERS:		Format: `READ PARAM` *parameter_data* `END PARAM` If parameters are entered, they must follow the sequence ID, title, and cross section library name See Sect. 8.1.3.3, Sect. 8.1.4.2, and Sect. 8.1.4.3 for further details.			
KEY	**DEFAULT**	**DEFINITION**	**KEY**	**DEFAULT**	**DEFINITION**
`RND=`	given	random number	`WTL=`	1/WTH	Russian Roulette weight
`TME=`	no limit	execution time (min)	`TTL=`	-1.0	CE temperature tolerance
`TBA=`	10 min	batch time (min)	`THC=`	10.0	thermal energy cutoff (eV)
`SIG=`	0.0	deviation limit	`DBH=`	210	DBRC upper energy cutoff (eV)
`WTA=`	0.5	average weight	`DBL=`	0.4	DBRC lower energy cutoff (eV)
`WTH=`	3.0	weight for splitting	`MSH=`	0.0	mesh size of the cubic mesh
`GEN=`	203	number of generations	`CET=`	0	CE TSUNAMI calculation mode
`NPG=`	1000	number per generation	`CFP=`	-1	number of latent generations for CE-TSUNAMI
`NSK=`	3	generations skipped	`X1D=`	0 or 1	number of of extra 1-Ds
`RES=`	0	generations between restart	`NB8=`	1000	blocks for direct access unit
`BEG=`	1	restart at this generation	`NL8=`	512	length of direct access block

continues on next page

Table 8.1.1 – continued from previous page

PARAMETERS:		Format: `READ PARAM` *parameter_data* `END PARAM` If parameters are entered, they must follow the sequence ID, title, and cross section library name See Sect. 8.1.3.3, Sect. 8.1.4.2, and Sect. 8.1.4.3 for further details.			
`NGP=`	252	number of energy groups for tallying	`NFB=`	NPG	fission bank positions
`DBR=`	0	use DBRC for scattering	`NBK=`	NPG+25	neutron bank positions
`DBX=`	2	Doppler Broadening method	`XNB=`	0	extra bank entries
`NQD=`	0	quadrature order for angular flux moments	`XFB=`	0	extra bank entries
`PNM=`	0	order of flux moments	`AMX=`	NO	all mixture xsecs
`CEP=`	NO	continuous energy directory file	`XAP=`	NO	xsec angles & probabilities
`XS1=`	NO	1-D xsecs	`XS2=`	NO	2-D xsecs
`APP=`	NO	append restart data	`XSL=`	NO	2-D Pl xsecs
`ADJ=`	NO	adjoint calculation	`PKI=`	NO	fission spectrum

continues on next page

Table 8.1.1 – continued from previous page

PARAMETERS:		Format: `READ PARAM` *parameter_data* `END PARAM` If parameters are entered, they must follow the sequence ID, title, and cross section library name See Sect. 8.1.3.3, Sect. 8.1.4.2, and Sect. 8.1.4.3 for further details.			
`PTB=`	YES	use probability tables	`P1D=`	NO	extra 1-D xsecs
`PNU=`	NO	use prompt neutron spectrum only	`PWT=`	NO	print average weight
`FRE=`	YES	use analytic free gas kernels	`PGM=`	NO	print unprocessed geometry
`UUM=`	NO	use unionized mixture xsec	`PAX=`	NO	albedo-xsec array
`M2U=`	NO	use unionized nuclide xsec	`PMF=`	NO	print angular fluxes
`CFX=`	NO	collect fluxes	`PMS=`	NO	print mesh fluxes
`FLX=`	NO	collect and print region fluxes	`PMM=`	NO	print mesh flux angular moments
`FDN=`	YES	fision densities	`PMV=`	NO	print mesh volumes
`FAR=`	NO	fission and absorption per region	`TFM=`	NO	coordinate transform
`GAS=`	FAR	FAR by energy	`FST=`	NO	print F*(r) 3dmap

continues on next page

Table 8.1.1 – continued from previous page

PARAMETERS:		Format: `READ PARAM` *parameter_data* `END PARAM` If parameters are entered, they must follow the sequence ID, title, and cross section library name See Sect. 8.1.3.3, Sect. 8.1.4.2, and Sect. 8.1.4.3 for further details.			
`NUB=`	YES	neutrons per fission	`SCX=`	NO	save CE-xsec to restart file
`MFP=`	NO	compute and print mean free paths	`SCD=`	NO	fission source convergence diag.
`SMU=`	NO	self-multiplication	`CDS=`	NO	accumulate neutron production
`MKP=`	NO	matrix keff by location in array	`FIS=`	NO	fission rate mesh tally
`CKP=`	NO	cofactor keff by location	`GFX=`	NO	compute grid fluxes
`FMP=`	NO	fission production by location	`MFX=`	NO	compute mesh fluxes
`MKU=`	NO	matrix keff by unit	`CGD=`	NO	use mesh for CLUTCH F*(r) calc.
`CKU=`	NO	cofactor keff by unit	`HTM=`	YES	produce HTML
`FMU=`	NO	fission production by unit	`RUN=`	YES	execute problem
`MKA=`	NO	matrix keff by array	`PLT=`	YES	print plots

continues on next page

Table 8.1.1 – continued from previous page

PARAMETERS:		Format: `READ PARAM` *parameter_data* `END PARAM` If parameters are entered, they must follow the sequence ID, title, and cross section library name See Sect. 8.1.3.3, Sect. 8.1.4.2, and Sect. 8.1.4.3 for further details.			
`CKA=`	NO	co-factor keff by array	`BUG=`	NO	debug print
`FMA=`	NO	fission production by array	`TRK=`	NO	print neutron tracks
`MKH=`	NO	matrix keff by hole	`GRP=`	0	group structure library
`CKH=`	NO	cofactor keff by hole	`RST=`	0	read restart
`FMH=`	NO	fission production by hole	`WRS=`	0	write restart
`HAL=`	NO	MKA at highest level	`SKT=`	16	scratch
`HHL=`	NO	MKH at highest level	`LIB=`	0	working xsecs
`XSC=`	14	mixed xsecs	`FNI=`		input restart file identifier
`ALB=`	79	albedo	`FNO=`		output restart file identifier
`WTS=`	80	weights			

Table 8.1.2: Summary of array data.

ARRAY Format: READ ARRAY array parameters data type orientation data END ARRAY See Sections on ARRAY Data, Geometry, and alternative sample problem mockups

Repeat the sequence ARRAY PARAMETERS DATA TYPE ORIENTATION DATA for each array used in the problem.

ARRAY PARAMETERS

KEYWORD	DEFAULT	DEFINITION
ARA=	none[a]	no. defining the array
TYP=	Cuboidal[c]	array type (cuboidal or square, hexagonal or triangular, shexagonal, rhexagonal, dodecahedral)
NUX=	none[a]	no. of units in X direction
NUY=	none[a]	no. of units in Y direction
NUZ=	none[a]	no. of units in Z direction
GBL=	none[b]	global or overall array number
PRT=	YES	print any label
COM=	none	delim comment delim optional, maximum of 132 characters

ORIENTATION DATA FOR FILL

Enter unit numbers to define every position in the array. When entering data using the options in this table, the count and option fields must be adjacent with no imbedded blanks. The operand field may be separated from the option field by one or more blanks. Orientation data for FILL is terminated by entering END FILL.

COUNT FIELD	OPTION FIELD	OPERAND FIELD	COMMENTS
		j	stores j at the current position in the array
i	R, *, $	j	stores j in the next i positions in the array
	F	j	fills remainder of the array with unit no. j starting with the current array position
	A	j	sets the current position in the array to j
i	S		increments current position in the array by i (This allows skipping i positions. The value of i may be positive or negative.)
i	Q	j	repeats the previous j entries i times. The default value of i is 1
i	N	j	repeats the previous j entries i times, inverting the sequence each time. The default value of i is 1.
i	B	j	starting with the entry at -i from the current position, store entries in inverse order until position -(i+1) is reached. Default value of i=1.
i	I	j k	provides the end points (j, k) with i entries linearly interpolated between them (i.e., a total of i+2 points). At least one blank must separate j and k. When used for an integer array, the i option should only be used to generate integer steps (i.e., (k-j)/(i+1) should be a whole number).

ORIENTATION DATA FOR LOOP

Enter the unit number and nine numbers that define the position(s) of that unit. Data for each of these ten entries are repeated until every position in the array has been defined. Orientation data for LOOP are terminated by entering END LOOP.
ENTER DATA IN THE FORM:

DATA ENTRY	COMMENTS
LTYPE	The unit type. LTYPE must be greater than 0.
IX1	Starting position in the X direction. IX1 must be at least 1 and no larger than the value entered for NUX.
IX2	Ending position in the X direction. IX2 must be at least 1 and no larger than the value of NUX.
INCX	The number of units by which increments are made in the X direction.
IY1	The starting position in the Y direction. IY1 must be at least 1 and less than the value entered for NUY.
IY2	Ending position in the Y direction. IY2 must be at least 1 and no larger than the value of NUY.
INCY	The number of units by which increments are made in the positive Y direction.
IZ1	Starting position in the Z direction. IZ1 must be at least 1 and no larger than NUZ.
IZ2	Ending position in the Z direction. IZ2 must be at least 1 and no larger than NUZ.
INCZ	The number of units by which increments are made in the positive Z direction.

[a] In KENO Va the default is 1

[b] In KENO Va the default is the largest array number, *maxarr*

Table 8.1.3: Summary of biasing data.

BIAS (weighting)

Format: **READ BIAS** keyword correlation data auxiliary data **END BIAS**

See sections on Biasing or weighting data

KEYWORD	DESCRIPTION	MATERIAL	ID	ENERGY GROUPS	THICKNESS/ INCREMENT

ID= CORRELATION DATA will be read next.

CORRELATION DATA

id	material ID. Enter ID from table at right to use	concrete	301	28,56,200,238,252	5 cm
	weighting data from the library	paraffin	400	28,56,200,238,252	3 cm
ibgn	beginning bias ID	water	500	28,56,200,238,252	3 cm
iend	ending bias ID	graphite	6100	28,56,200,238,252	20 cm

WT= AUXILIARY DATA will be read next.

or

WTS= AUXILIARY DATA will be read next.

AUXILIARY DATA

wttitl	material title (12-character maximum)
id	material ID
nsets	number of sets of group structures
REPEAT	(THKINC, NUMINC, NGPWT, WTAVG) NSETS TIMES
thkinc	thickness per increment
numinc	number of increments
ngpwt	number of energy groups for this set of wts
wtavg	enter numinc x ngpwt values of wtavg

For CORRELATION DATA, the material ID is chosen from the material **ID** column above (the keyword is **ID=**).
For AUXILIARY DATA, the material ID is chosen by the user, and the keyword is **WT=** or **WTS=**. When AUXILIARY DATA are entered, CORRELATION DATA must also be entered to use the data.

Beginning and ending bias IDs are defined by the user. The geometry specification that has the bias ID equal to the beginning bias ID uses the wtavg's from the first interval of material ID.

Table 8.1.4: Summary of boundary condition data.

<table>
<tr><td>BNDS
(albedo or boundary conditions)</td><td colspan="8">Format:
KENO V.a: READ BNDS face code albedo name END BNDS KENO-VI : READ BNDS [BODY=label] face code albedo name END BNDS
See Sect. 8.1.3.6</td></tr>
<tr><td></td><td colspan="8">The sequence FACE CODE ALBEDO NAME is entered as many times as necessary to define the appropriate albedo boundary conditions. KENO-VI's BODY parameter must be used to define the appropriate albedo boundary conditions when the boundary definition vector of the global unit has more than one body label. The default for all faces is vacuum.</td></tr>
<tr><td></td><td colspan="8">FACE CODES FOR ENTERING BOUNDARY (ALBEDO) CONDITIONS</td></tr>
<tr><td></td><td>FACE CODE</td><td>DEFINITION</td><td>FACE CODE</td><td>DEFINITION</td><td>FACE CODE</td><td>DEFINITION</td><td>FACE CODE</td><td>DEFINITION</td></tr>
<tr><td></td><td>SURFACE(k)</td><td>albedo surface enumeration indicates any k^{th} face of the boundary shape. Used for both cuboidal and non-cuboidal boundary shapes. shape-specific albedo numbers for both KENO V.a and. KENO-VI are listed in Table 8.1.6</td><td>+XB=</td><td>positive X face</td><td>&FC=</td><td>all positive faces</td><td>+YZ=</td><td>positive Y and Z faces</td></tr>
<tr><td></td><td></td><td></td><td>&XB=</td><td>positive X face</td><td>-FC=</td><td>all negative faces</td><td>+ZY=</td><td>positive Y and Z faces</td></tr>
<tr><td></td><td></td><td></td><td>-XB=</td><td>negative X face</td><td>XYF=</td><td>all X and Y faces</td><td>&YZ=</td><td>positive Y and Z faces</td></tr>
<tr><td></td><td></td><td></td><td>+YB=</td><td>positive Y face</td><td>XZF=</td><td>all X and Z faces</td><td>&ZY=</td><td>positive Y and Z faces</td></tr>
<tr><td></td><td></td><td></td><td>&YB=</td><td>positive Y face</td><td>YZF=</td><td>all Y and Z faces</td><td>-XY=</td><td>negative X and Y faces</td></tr>
<tr><td></td><td></td><td></td><td>-YB=</td><td>negative Y face</td><td>+XY=</td><td>positive X and Y faces</td><td>-XZ=</td><td>negative X and Z faces</td></tr>
</table>

continues on next page

Table 8.1.4 – continued from previous page

			+ZB=	positive Z face	**+YX=**	positive X and Y faces	**-YZ=**	negative Y and Z faces
			&ZB=	positive Z face	**&YX=**	positive X and Y faces	**YXF=**	all X and Y faces
			-ZB=	negative Z face	**&XY=**	positive X and Y faces	**ZXF=**	all X and Z faces
			XFC=	both X faces	**+XZ=**	positive X and Z faces	**ZYF=**	all Y and Z faces
			YFC=	both Y faces	**+ZX=**	positive X and Z faces	**-YX=**	negative X and Y faces
	ALL=	all faces of a single or multiple boundary shape(s)	ZFC=	both Z faces	**&XZ=**	positive X and Z faces	**-ZX=**	negative X and Z faces
			+FC=	all positive faces	**&ZX=**	positive X and Z faces	**-ZY=**	negative Y and Z faces
			Above face codes are applicable for **only cuboidal boundary shapes** (cube or cuboid).					

Table 8.1.5: Summary of boundary condition data (continued).

ALBEDO NAMES AVAILABLE ON THE KENO ALBEDO LIBRARY, FOR USE WITH THE FACE CODES								
ALBEDO NAME	DESC.	MODE	ALBEDO NAME	DESC.	MODE	ALBEDO NAME	DESC.	MODE
DP0H2O DP0H2O DP0 DP0	12 in. double P0 water differential albedo with 4 incident angles	MG	CONC-4 CON4 CONC4	4 in. concrete differential albedo with 4 incident angles	MG	VAC-UUM VOID VACU VAC	vacuum condition	MG and CE
			CONC-8 CON8 CONC8	8 in. concrete differential albedo with 4 incident angles	MG	SPECU-LAR MIR-ROR MIRR SPEC SPE MIR	mirror image reflection	MG and CE
H2O WATER	12 in. water differential albedo with 4 incident angles	MG						
PARAF-FIN PARA WAX	12 in. paraffin differential albedo with 4 incident angles	MG	CONC-12 CON12 CONC12	12 in. concrete differential albedo with 4 incident angles	MG			

continues on next page

Table 8.1.5 – continued from previous page

ALBEDO NAMES AVAILABLE ON THE KENO ALBEDO LIBRARY, FOR USE WITH THE FACE CODES								
ALBEDO NAME	DESC.	MODE	ALBEDO NAME	DESC.	MODE	ALBEDO NAME	DESC.	MODE
CARBON GRAPHITE C	200 cm carbon differential albedo with 4 incident angles	MG	CONC-16 CON16 CONC16	16 in. concrete differential albedo with 4 incident angles	MG	PERIODIC PERI PER	periodic boundary condition	MG and CE
ETHYLENE POLY CH2	12 in. polyethylene differential albedo with 4 incident angles	MG	CONC-24 CONC CONC24	24 in. concrete differential albedo with 4 incident angles	MG	WHITE	White boundary condition	MG and CE

Table 8.1.6: Summary of boundary condition data specific to KENO V.a

ALBEDO SURFACE NUMBERS RELATED TO KENO V.a GEOMETRY SHAPES						
GEOMETRY SHAPE	1	2	3	4	5	6
CUBE	+X	-X				
CUBOID	+X	-X	+Y	-Y	+Z	-Z
CYLINDER	Radial	+Z	-Z			
HEMISPHERE	Radial	Cut surface				
HEMICYLINDER	Radial	Top	Bottom	Cut surface		
SPHERE	Radial					
XCYLINDER	Radial	+X	-X			
YCYLINDER	Radial	+Y	-Y			
ZCYLINDER	Radial	+Z	-Z			

Table 8.1.7: Summary of boundary condition data specific to KENO-VI.

ALBEDO SURFACE NUMBERS RELATED TO KENO-VI GEOMETRY BODIES												
GEOMETRY BODY	1	2	3	4	5	6	7	8	9	10	11	12
CONE	Radial	+Z	-Z									
CUBOID	+X	-X	+Y	-Y	+Z	-Z						
CYLINDER	Radial	+Z	-Z									
DODECAHEDRON	+X	-X	+Y	-Y	+X +Y +Z	-X -Y -Z	-X +Y +Z	+X -Y -Z	-X -Y +Z	+X +Y -Z	+X -Y +Z	-X +Y -Z
ECYLINDER	Radial	+Z	-Z									
ELLIPSOID	Radial											
HEXPRISM	+X	-X	+X +Y	-X -Y	-X +Y	+X -Y	+Z	-Z				
HOPPER	+X	-X	+Y	-Y	+Z	-Z						
PENTAGON	-Y	+X -Y	+X +Y	-X +Y	-X -Y	+Z	-Z					
PLANE	Surface											
QUADRATIC	Surface											
RHEXPRISM	+Y	-Y	-X +Y	+X -Y	+X +Y	-X -Y	+Z	-Z				

continues on next page

Table 8.1.7 – continued from previous page

ALBEDO SURFACE NUMBERS RELATED TO KENO-VI GEOMETRY BODIES												
RING	Inner Radius	Outer Radius	+Z	-Z								
SPHERE	Radial											
WEDGE	-Y	-X +Y	+X +Y	+Z	-Z							
XCYLINDER	Radial	+X	-X									
XPPLANE	+X	-X										
YCYLINDER	Radial	+Y	-Y									
YPPLANE	+Y	-Y										
ZCYLINDER	Radial	+Z	-Z									
ZPPLANE	+Z	-Z										
Surfaces refer to the pre-rotation surface of the body that occurs in the indicated quadrant. Refer to Fig. 8.1.1 through Fig. 8.1.28 for illustrations of each geometry body.												

Table 8.1.8: Summary of geometry data in KENO V.a.

GEOMETRY (region)

Format: READ GEOM enter geometry region data here END GEOM

See sections on Geometry, Geometry data, Multiple entries in geometry data, and Alternative Sample problem mockups.

GEOMETRY REGION DATA consist of SIMPLE GEOMETRY REGION DATA and EXTENDED GEOMETRY REGION DATA.

ENTER GEOMETRY DATA IN THE FOLLOWING FORM:

OPTIONAL GLOBAL SPECIFICATION

UNIT n

OPTIONAL GEOMETRY COMMENT

GEOMETRY REGION DATA and/or EXTENDED GEOMETRY REGION DATA

****

ENTER SIMPLE REGION DATA IN THE FOLLOWING FORM:

GLOBAL Enter only to specify this unit as the global unit.

UNIT n

COM=delim comment delim This optional comment can be up to 132 characters long. It must begin and end with a delimiter (i.e. " , ' , * , ^ , or !)

fgeom mix no. bias ID dimensions optional origin data (ORIGIN coordinates) optional chord data (CHORD distance) optional center data (CENTER type X-offset Y-offset Z-offset)

Enter as many geometry description specifications as necessary to describe the unit and as many units as necessary to describe the system.

SIMPLE GEOMETRY REGION INPUT DATA REQUIREMENTS

TYPE OF DATA	TYPE 1 DATA	TYPE 2 DATA	TYPE 3 DATA	TYPE 4 DATA	TYPE 5 DATA	TYPE 6 DATA
fgeom	SPHERE HEMISPHERE HEMISPHE+X HEMISPHE−X HEMISPHE+Y HEMISPHE−Y HEMISPHE+Z HEMISPHE−Z	XCYLINDER XHEMICYL+Y XHEMICYL−Y XHEMICYL+Z XHEMICYL−Z	YCYLINDER YHEMICYL+X YHEMICYL−X YHEMICYL+Z YHEMICYL−Z	CYLINDER ZCYLINDER ZHEMICYL+X ZHEMICYL−X ZHEMICYL+Y ZHEMICYL−Y	CUBE	CUBOID
dimensions	R (radius)	R +X −X	R +Y −Y	R +Z −Z	+X −X	+X −X +Y −Y +Z −Z
optional origin coordinates*	Enter the X Y Z coordinates of origin	Enter the Y Z coordinates of centerline	Enter the X Z coordinates of centerline	Enter the X Y coordinates of centerline	omit	omit
optional chord data**	Enter the dist. to plane	Enter the dist. to plane	Enter the dist. to plane	Enter the dist. to plane	omit	omit
optional center data***	Center type	X-offset	X-offset	Z-offset	omit	omit

*Enter ORIG or ORIGIN for *fgeom*.

**Enter CHORD for *fgeom*.

***Center data are only applicable when flux moments (PNM>0) or angular fluxes (NQD>0) are requested with the coordinate transform (TFM) set to YES.

NOTE: Chord data are not applicable for SPHERE, XCYLINDER, YCYLINDER, CYLINDER, ZCYLINDER, CUBE, or CUBOID.
Origin data are not applicable for a CUBE or CUBOID.

GEOMETRY (region) (continued)

ENTER GEOMETRY DATA IN THE FOLLOWING FORM:

fgeom ref. ID bias ID thickness per region origin coordinates nreg

EXTENDED GEOMETRY REGION INPUT DATA REQUIREMENTS

TYPE OF DATA	TYPE 1 DATA	TYPE 2 DATA	TYPE 3 DATA
fgeom	ARRAY	HOLE	REPLICATE REFLECTOR
ref. ID	array no.	emplaced unit number	mixture number in generated regions
bias ID	omit for ARRAY	omit	first bias ID
thickness / region	omit	omit	variable▫
origin coordinates	Enter the X Y Z coordinates of the most negative point of the array	Enter the X Y Z coordinates of the origin	omit
nreg	omit	omit	number of regions to be generated

▫The number of dimensions to be entered is the same as the region preceding the replicate or reflector specification because the g enerated regions have that shape. The value of the dimensions is the thickness of each generated region of material on that surface and must therefore be a positive value.

Table 8.1.9: Summary of geometry data in KENO-VI.

GEOMETRY (region)

Format: READ GEOM enter geometry region data here END GEOM.. See Appendix B.
GEOMETRY REGION DATA consist of SIMPLE GEOMETRY REGION DATA and EXTENDED GEOMETRY REGION DATA.
* *
ENTER DATA IN THE FOLLOWING FORM:
GLOBAL Enter once to specify this unit as the global unit.
UNIT n
COM=delim comment delim This optional comment can be up to 132 characters. It must begin and end with a delimiter (i.e. " , ' , * , ^ , or !).
Geometry Keyword Label Input Parameters Optional origin data (ORIGIN) Optional chord data (CHORD) Optional rotation data (ROTATE degrees)
Media Keyword Input Parameters
Boundary Keyword Input Parameters Each unit must have one boundary record.
Enter as many geometry and content descriptions as necessary to describe the unit and as many units as necessary to describe the system.

GEOMETRY REGION INPUT DATA REQUIREMENTS

TYPE OF DATA	KEYWORD	INPUT PARAMETERS	NOTES
GEOMETRY	CONE	Lbl Rt Zt Rb Zb	
	CUBOID	Lbl +X −X +Y −Y +Z −Z	
	CYLINDER	Lbl R Zt Zb	
	DODECAHEDRON	Lbl R	
	ECYLINDER	Lbl Rx Ry Zt Zb	
	ELLIPSOID	Lbl Rx Ry Rz	
	HEXPRISM	Lbl R Zt Zb	
	HOPPER	Lbl Lxt Lyt Zt Lxb Lyb Zb	
	PARALLELPIPED	Lbl Xdist Ydist Zdist Psi Theta Phi	
	PENTAGON	Lbl R Zt Zb	
	PLANE	Lbl XPL=a YPL=b ZPL=c CON=d	For geometry keywords, PLANE and QUADRATIC data are entered after the = following the respective subordinate keyword. Only nonzero data are required.
	QUADRATIC	Lbl AQU=a BQU=b CQU=c DQU=d EQU=e FQU=f GQU=g HQU=h IQU=i JQU=j	
	RHEXPRISM	Lbl R Zt Zb	
	RHOMBOID	Lbl Dx Psi	
	RING	Lbl Ri Ro Zt Zb	
	SPHERE	Lbl R	
	WEDGE	Lbl Xbase Xpt Ypt Zlng	
	XCYLINDER	Lbl R Xt Xb	
	XPPLANE	Lbl Xt Xb	
	YCYLINDER	Lbl R Yt Yb	
	YPPLANE	Lbl Yt Yb	
	ZCYLINDER	Lbl R Zt Zb	
	ZPPLANE	Lbl Zt Zb	
MODIFICATION*	CHORD	+X= −X= +Y= −Y= +Z= −Z=	Only nonzero data are required.
	ORIGIN	X= Y= Z=	Only nonzero data are required.
	ROTATE	A1= A2= A3=	The body is rotated about the Origin using the Euler angle X-convention. Only nonzero data are required; input is in degrees.
MEDIA**	MEDIA	Mixture ID Bias ID i1 i2 . . .	
	HOLE	Hole unit number	
	ARRAY	Array ID i1 i2 . . . PLACE nx ny nz xx yy zz	Place positions the array by placing the origin of the unit in array position nx ny nz at xx yy zz. The region boundary containing the array must be enclosed in or on the array boundary.
BOUNDARY	BOUNDARY	i1 i2 ...	Defines the overall volume of the unit.

*Modification data can be entered in any order. Any combination of modification data can follow any geometry record.
**The Region definition vector, i1 i2 . . ., should contain references to all geometry records which share a boundary with t he region being defined.

Table 8.1.10: Summary of mixing table data.

MIXTURES

Format: READ MIXT xsec parameters mixing table data. END MIXT
Multigroup mode: These data are entered only if an AMPX working format library is being used. (LIB=) in the parameter data, s ee section on Title and parameter data. Do not enter if a Monte Carlo mixed library is used, (XSC=) in the parameter data.
Continuous energy mode: Only the mixing table data are entered. See sections on Mixing table and Mixing table data.

XSEC PARAMETERS — consists of keywords and associated values.
These parameters, if entered, only need to be entered once.

KEYWORD	DEFAULT	DEFINITION
SCT=	1	number of discrete scattering angles 0 is isotropic 1 is P1 2 is P3 3 is P5
EPS=	0.1	cross section message cutoff value used to suppress message numbers K5-060 and K6-060.

MIXING TABLE DATA consists of
(1) a keyword and mixture ID for the mixture
The keyword is MIX=
The desired mixture number follows the keyword
(1a) optionally, a keyword and nuclide mixture ID for the rest of the current mixture
The keyword is NCM=
The desired nuclide mixture ID follows the keyword
(1b) optionally, a keyword and temperature for the mixture (used for continuous energy problems)
The keyword is TMP= or TEM=
The desired temperature in Kelvin follows the keyword
(2) nuclide ID**
(2a) optionally a keyword and nuclide mixture ID for this nuclide
The keyword is NCM= and the desired nuclide mixture ID follows the keyword
(3) number density**
(4) continuous energy cross section filename**
** The sequence (2) (3) (4) is repeated for each nuclide in the mixture. Entry (4) is optional and entered with XS= name where name is the filename.

REPEAT the sequence (1) (2)s (3)s (4)s until all the mixtures have been described.

Table 8.1.11: Summary of plot data.

PLOT — Format: READ PLOT plot parameters END PLOT — plot parameters must be entered for each plot that is to be made.
See sections Plot data and Color plots.

KEYWORD	DEFAULT	DEFINITION
TTL=	prob. title	delim ptitl delim — delim is a one-character delimiter (i.e. ' , " , * , / , or !) that signals the beginning and end of the title.
		ptitl is the plot title (max. 132 char.)
PIC=	MAT	Type of plot: MIXTURE NO., UNIT NO. or BIAS ID NO.
		MIXTURE ----------MAT MIX MIXT MIXTURE MEDI MEDIA
		UNIT NO. ----------UNT UNIT UNITTYPE
		BIAS ID NO. -------IMP BIAS BIASID WTS WEIG WEIGHTS WGT WGTS
XUL=	prev. plot	X coord. of upper left corner of plot
YUL=	prev. plot	Y coord. of upper left corner of plot
ZUL=	prev. plot	Z coord. of upper left corner of plot
XLR=	prev. plot	X coord. of lower right corner of plot
YLR=	prev. plot	Y coord. of lower right corner of plot
ZLR=	prev. plot	Z coord. of lower right corner of plot

KEYWORD	DEFAULT	DEFINITION
UAX=	prev. plot 0 IF VAX OR WAX is read	X component of direction cosine for the AX axis of the plot (across)
VAX=	prev. plot 0 IF UAX OR WAX is read	Y component of direction cosine for the AX axis of the plot (across)
WAX=	prev. plot 0 IF UAX OR VAX is read	Z component of direction cosine for the AX axis of the plot (across)
UDN=	prev. plot 0 IF VDN OR WDN is read	X component of direction cosine for the DN axis of the plot (down)
VDN=	prev. plot 0 IF UDN OR WDN is read	Y component of direction cosine for the DN axis of the plot (down)
WDN=	prev. plot 0 IF UDN OR VDN is read	Z component of direction cosine for the DN axis of the plot (down)
DLX=		Horizontal spacing between points on plot
DLD=		Vertical spacing between points on plot
NAX=		No. of intervals to be printed across page
NDN=		No. of intervals to be printed down page
LPI=	8.0 (character plots) 10 (color plots)	Vertical to horizontal scaling factor for plot proportionality.
RUN=	YES	YES allows the problem to execute NO terminates problem after data checking
PLT=	YES	YES allows the plot(s) to be made NO allows reading the plot data without making a plot
SCR=	YES	Display plot method SCR=YES utilizes color plot display SCR=NO utilizes printer plot display
NCH=	CHRS[c]	delim CHRS delim a one character delimiter[b] signals the beginning and end of the character string
CLR=	See Table on default color specifications for the color plot display method	num(i) red(num(i)) green (num(i)) blue (num(i)) num(i) defines mix. no., unit no., or bias ID next 3 entries define red, green, and blue components of the color representing num(i).

PLOT ORIGIN:
(1) SINGLE UNIT - coincides with origin of geometry description.
(2) BASE ARRAY - at the most negative point of the global array
(3) REFLECTED ARRAY - coincides with the origin of the CORE or ARRAY description of the global array.

[c] Default values of CHRS are given below:

MEDIA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
CHRS		1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	I	J	K	L	M
MEDIA	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42			
CHRS		N	O	P	Q	R	S	T	U	V	W	X	Y	Z	#		$	<	=	)			
MEDIA	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58							
CHRS	&	>	:	;	.	-	%	*	"	=	!	(	@	<	/	0							

[b] Acceptable delimiters are ',-,*,^, or !.

Table 8.1.12: Summary of starting data.

START Format: READ START enter start data here END START
The default value of start type is zero. See Sect. **8.1.2.8.**

START TYPE	REQUIRED DATA	OPTIONAL DATA	STARTING DISTRIBUTION
0	none	NST	uniform
		XSM	
		XSP	
		YSM	
		YSP	
		ZSM	
		ZSP	
		RFL[1]	
		PSP	
		WS6	
1	NST	XSM	cosine
		XSP	
		YSM	
		YSP	
		ZSM	
		ZSP	
		RFL	
		PSP	
		WS6	
2	NST	XSM	cosine with
	NXS	XSP	fraction in
	NYS	YSM	specified unit
	NZS	YSP	
	FCT	ZSM	
		ZSP	
		RFL	
		PSP	
		WS6	

START TYPE	REQUIRED DATA	OPTIONAL DATA	STARTING DISTRIBUTION
3	NST	KFS	spike
	TFX	PSP	
	TFY	WS6	
	TFZ		
	NXS		
	NYS		
	NZS		
4	NST	KFS	multiple
	TFX	PSP	spikes
	TFY	WS6	
	TFZ		
	NBX		
5	NST	PSP	in specified
	NBX	WS6	units
6	NST	NXS	arbitrary
	TFX	NYS	points
	TFY	NZS	
	TFZ	KFS	
	LNU[2]	PS6	
		PSP	
		RDU	
		WS6	
7	NST	XSM	uniform in X-
		XSP	and Y-dimension
		YSM	and $(1\text{-}\cos)^2$ in
		YSP	Z-dimension
		ZSM	
		ZSP	
		PSP	
		WS6	
8	NST		flat in X- and
	ZSM	XSM	Y-dimension and
	ZSP	XSP	segmented in Z
	FCT	YSM	
		YSP	
		PSP	
		WS6	

KEYWORD	DEFAULT	DEFINITION
NST=	0	start type
TFX=	0.0	X coordinate
TFY=	0.0	Y coordinate
TFZ=	0.0	Z coordinate
NXS=	0	X index of unit pos.
NYS=	0	Y index of unit pos.
NZS=	0	Z index of unit pos.
KFS=		fissile mixture no.
LNU=	0	number of last neutron
NBX=	0	source unit number
FCT=	0	fraction
XSM=	−X	−X of source cuboid
XSP=	+X	+X of source cuboid
YSM=	−Y	−Y of source cuboid
YSP=	+Y	+Y of source cuboid
ZSM=	−Z	−Z of source cuboid
ZSP=	+Z	+Z of source cuboid
RFL=	NO	start in reflector
PS6=	NO	print start 6 input
PSP=	NO	print starting points
RDU=	0	unit containing start data
WS6=	0	unit to write start data

[1] RFL is no longer supported (obsolete parameter)

[2] LNU must be the last entry for each set of start 6 data. The LNU of each successive set of data must be larger than the last.

Table 8.1.13: Summary of volume data (KENO-VI).

KEYWORD — Format: READ VOLUME enter volume data here END VOLUME
The default type for volume calculation type is NONE. See section Volume Data.

	TYPE	REQUIRED DATA	OPTIONAL DATA	DEFAULT
TYPE=	NONE	none	none	
	RANDOM	none	BATCHES=	500
			POINTS=	1000
			XP=	global unit boundary
			XM=	global unit boundary
			YP=	global unit boundary
			YM=	global unit boundary
			ZP=	global unit boundary
			ZM=	global unit boundary
			SAMPLE_DEN=	points sampled per cubic cm
	TRACE	none	NRAYS=	100,000
			XP=	global unit boundary
			XM=	global unit boundary
			YP=	global unit boundary
			YM=	global unit boundary
			ZP=	global unit boundary
			ZM=	global unit boundary
			IFACE=	smallest face
READVOL=		File name	none	

Table 8.1.14: Summary of grid geometry data.

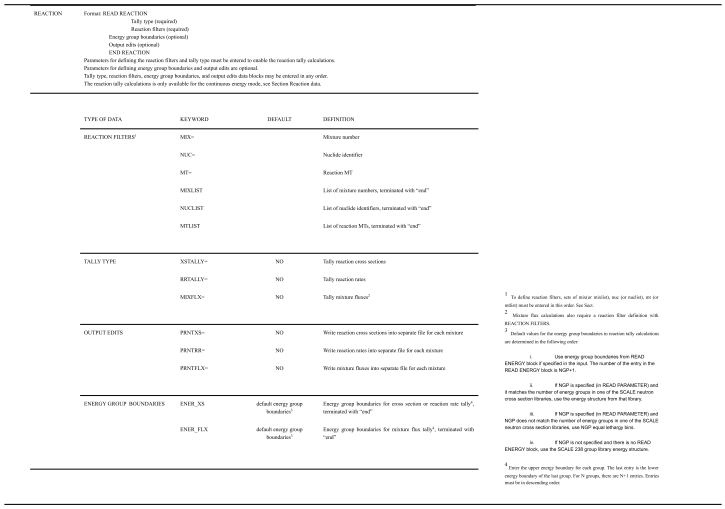
GRID	Format: READ GRID enter grid data here END GRID See Section Grid Geometry Data.		
	KEYWORD	DEFAULT	DEFINITION
	None	-1	grid ID number, required if multiple mesh is defined
	NXCELLS= NUMXCELLS=	1	number of cells in the the X direction
	XMIN=	0	minimum cell boundary in the X direction
	XMAX=	1	maximum cell boundary in the X direction
	or		
	XPLANES	[0,1]	the cell boundaries in the X direction
	XLINEAR		*ncells; xmin; xmax* Number of cells in the X direction; X minimum; X maximum
	NYCELLS= NUMYCELLS=	1	number of cells in the Y direction
	YMIN=	0	minimum cell boundary in the Y direction
	YMAX=	1	maximum cell boundary in the Y direction
	or		
	YPLANES=	[0,1]	the cell boundaries in the Y direction
	YLINEAR		*ncells; ymin; ymax* Number of cells in the Y direction; Y minimum; Y maximum
	NZCELLS= NUMZCELLS=	1	number of cells in the Z direction
	ZMIN=	0	minimum cell boundary in the Z direction
	ZMAX=	1	maximum cell boundary in the Z direction
	or		
	ZPLANES=	[0,1]	the cell boundaries in the Z direction
	ZLINEAR		*ncells; zmin; zmax* Number of cells in the Z direction; Z minimum; Z maximum
	TITLE=	' '	a title for this grid mesh
	TOLERANCE=	1.0E-6	For removing duplicate planes from *PLANES or *LINEAR

Note: The grid must be defined for MG TSUNAMI-3D calculations so that the entire system lies inside the boundaries of the geometry grid. Multiple grids are defined by entering this block multiple times.

Table 8.1.15: Summary of energy group boundaries data.

ENERGY	Format: READ ENERGY energy group boundaries END ENERGY Enter upper energy boundary for each group in eV. The last entry is the lower energy boundary of the last group. For N groups, there are N+1 entries. Entries must be in descending order and in units of eV.

Table 8.1.16: Summary of reaction data.



REACTION

Format: READ REACTION
Tally type (required)
Reaction filters (required)
Energy group boundaries (optional)
Output edits (optional)
END REACTION
Parameters for defining the reaction filters and tally type must be entered to enable the reaction tally calculations.
Parameters for defining energy group boundaries and output edits are optional.
Tally type, reaction filters, energy group boundaries, and output edits data blocks may be entered in any order.
The reaction tally calculations is only available for the continuous energy mode, see Section Reaction data.

TYPE OF DATA	KEYWORD	DEFAULT	DEFINITION
REACTION FILTERS[1]	MIX=		Mixture number
	NUC=		Nuclide identifier
	MT=		Reaction MT
	MIXLIST		List of mixture numbers, terminated with "end"
	NUCLIST		List of nuclide identifiers, terminated with "end"
	MTLIST		List of reaction MTs, terminated with "end"
TALLY TYPE	XSTALLY=	NO	Tally reaction cross sections
	RRTALLY=	NO	Tally reaction rates
	MIXFLX=	NO	Tally mixture fluxes[2]
OUTPUT EDITS	PRNTXS=	NO	Write reaction cross sections into separate file for each mixture
	PRNTRR=	NO	Write reaction rates into separate file for each mixture
	PRNTFLX=	NO	Write mixture fluxes into separate file for each mixture
ENERGY GROUP BOUNDARIES	ENER_XS	default energy group boundaries[3]	Energy group boundaries for cross section or reaction rate tally[4], terminated with "end"
	ENER_FLX	default energy group boundaries[3]	Energy group boundaries for mixture flux tally[4], terminated with "end"

[1] To define reaction filters, sets of mixture number, nuc (or nuclide), mt (or nuclide) must be entered in this order. See Sect.

[2] Mixture flux calculations also require a reaction filter definition with REACTION FILTERS.

[3] Default values for the energy group boundaries in reaction tally calculations are determined in the following order:

i. Use energy group boundaries from READ ENERGY block if specified in the input. The number of the entry in the READ ENERGY block is NGP+1.

ii. If NGP is specified (in READ PARAMETER) and it matches the number of energy groups in one of the SCALE neutron cross section libraries, use the energy structure from that library.

iii. If NGP is specified (in READ PARAMETER) and NGP does not match the number of energy groups in one of the SCALE neutron cross section libraries, use NGP equal-lethargy bins.

iv. If NGP is not specified and there is no READ ENERGY block, use the SCALE 238 group library energy structure.

[4] Enter the upper energy boundary for each group. The last entry is the lower energy boundary of the last group. For N groups, there are N+1 entries. Entries must be in descending order.

8.1.3.1 Keno input outline

The data input for KENO is outlined below. Default data for KENO have been found to be adequate for many problems. These values should be carefully considered when entering data.

Blocks of input data are entered in the form:

`READ XXXX` *input_data* `END XXXX`

where `XXXX` is the keyword for the type of data being entered. The keywords that can be used are listed in Table 8.1.17. A minimum of four characters is required for a keyword, and some keyword names may be as long as twelve characters (`READ PARAMETER`, `READ GEOMETRY`, etc.). Keyword inputs are not case sensitive. Data input is activated by entering the words `READ XXXX` followed by one or more blanks. All input data pertinent to `XXXX` are then entered. Data for `XXXX` are terminated by entering `END XXXX` followed by two or more blanks. Note that multiple `READ GRID` blocks are used if multiple grid definitions are needed.

Table 8.1.17: Types of input data.

Type of data	**Keyword**
Parameters	`PARAMETER`, `PARA` or `PARM`
Geometry	`GEOMETRY`, `GEOM`
Biasing	`BIAS`
Boundary conditions	`BOUNDS`, `BOUN` or `BNDS`
Start	`START`, `STAR` or `STRT`
Energy	`ENERGY` or `ENER`
Array (unit orientation)	`ARRAY` or `ARRA`
Extra 1-D cross sections	`X1DS`
Cross section mixing table[a]	`MIXT` or `MIX`
Plot[a]	`PLOT` or `PLT` or `PICT`
Volumes	`VOLUME`
Grid geometry	`GRIDGEOMETRY`, `GRIDGEOM`, or `GRID`
Reactions	`REACTION`, `REAC`
[a] MIX and PLT must include a trailing blank, which is considered part of the keyword.	

Three data records must be entered for every problem:

(1) the SCALE sequence identifier,

(2) the problem title,

(3) and the `END DATA` to terminate the problem.

KENO V.a or KENO-VI are typically run as part of CSAS, TRITON, or TSUNAMI sequences, but it may also be run standalone. For standalone KENO execution, the sequence identifier is specified using one line similar to:

```
=kenovi
```

A problem title **must be entered** and must immediately follow the sequence identifier (limit 80 characters, including blanks; extra characters will be discarded). See Sect. 8.1.3.3.

The following guidance generally assumes the user is running KENO stand alone. If KENO is to be run using of the other sequences (e.g., CSAS5), see the appropriate manual section for additional guidance.

`READ PARA` *parameter_data* `END PARA`

Enter parameter input as needed to describe a problem. If parameter data are desired in standalone KENO calculations (i.e., non-CSAS), they must immediately follow the problem title. Default values are assigned to all parameters. A problem **can** be run without entering any parameter data if the default values are acceptable.

Parameter data must begin with the words `READ PARA`, `READ PARM`, or `READ PARAMETER`. Parameter data may be entered in any order. If a parameter is entered more than once, the last value is used. The words `END PARA` or `END PARM`, or `END PARAMETER` terminate the parameter data. See Sect. 8.1.3.3.

(n_1)...(n_{13}) The following data may be entered in any order. Data not needed to describe the problem may be omitted.

(n_1) `READ GEOM` *all_geometry_region_data* `END GEOM`

Geometry region data must be entered for every problem that is not a restart problem. Geometry data must begin with the words `READ GEOM` or `READ GEOMETRY`. The words `END GEOM` or `END GEOMETRY` terminate the geometry region data. See Sect. 8.1.3.4.

(n_2) `READ ARRA` *array_definition_data* `END ARRA`

Enter array definition data as needed to describe the problem. Array definition data define the array size and position units (defined in the geometry data) in a 3-D lattice that represents the physical problem being analyzed. Array data must begin with the words `READ ARRA` or `READ ARRAY` and must terminate with the words `END ARRA` or `END ARRAY`. See Sect. 8.1.3.5.

(n_4) `READ BOUN` *albedo_boundary_conditions* `END BOUN`

Enter albedo boundary conditions as needed to describe the problem. Albedo data must begin with the words `READ BOUN`, `READ BNDS`, `READ BOUND`, or `READ BOUNDS`, and it must terminate with the words `END BOUN`, `ENDS BNDS`, `END BOUND`, or `END BOUNDS`. See Sect. 8.1.3.6.

(n_3) `READ BIAS` *biasing_information* `END BIAS`

The *biasing_information* is used to define the weight given to a neutron surviving Russian roulette. Biasing data must begin with the words READ BIAS. The words END BIAS terminate the biasing data. See Sect. 8.1.3.7.

(n_5) READ STAR *starting_distribution_information* END STAR

Enter starting information data for starting the initial source neutrons only if a uniform starting distribution is undesirable. Start data must begin with the words READ STAR, READ STRT or READ START, and it must terminate with the words END STAR, END STRT or END START. See Sect. 8.1.3.8.

(n_6) READ ENER *energy_group_boundaries* END ENER

Enter upper energy boundaries for each neutron energy group to be used for tallying in the continuous energy mode. Energy bin data begin with the words READ ENER or READ ENERGY and terminate with the words END ENER or END ENERGY. The last entry is the lower energy boundary of the last group. The values must be in descending order. This block is only applicable to continuous energy KENO calculations. See Sect. 8.1.3.12.

(n_7) READ MIXT *cross_section_mixing_table* END MIXT

Enter a mixing table to define all the mixtures to be used in the problem. The mixing table must begin with the words READ MIXT or READ MIX and must end with the words END MIXT or END MIX. Do not enter mixing table data if KENO is being executed as a part of a SCALE sequence. See Sect. 8.1.3.10.

(n_8) READ X1DS *extra_1-D_cross_section_IDs* END X1DS

Enter the IDs of any extra 1-D cross sections to be used in the problem. These must be available on the mixture cross section library. Extra 1-D cross section data must begin with the words READ X1DS and terminate with the words END X1DS. See Sect. 8.1.3.9.

(n_9) READ PLOT *plot_data* END PLOT

Enter the data needed to provide a 2-D character or color plot of a slice through a specified portion of the 3-D geometrical representation of the problem. Plot data must begin with the words READ PLOT, READ PLT, or READ PICT and terminate with the words END PLOT, END PLT, or END PICT. See Sect. 8.1.3.11.

(n_{10}) READ VOLUME *volume_data* END VOLUME

Enter the data needed to specify the volumes of the geometry data. Volume data must begin with the words READ VOLUME and end with the words END VOLUME. See Sect. 8.1.3.13.

(n_{11}) READ GRID *mesh_grid_data* END GRID

Enter the data needed to specify a simple Cartesian grid over either the entire problem or part of the problem geometry for tallying fluxes, moments, fission sources, etc. Grid data may be entered using the keywords READ GRID, READ GRIDGEOM, or READ GRIDGEOMETRY, and they are terminated with either END GRID, END GRIDGEOM, or END GRIDGEOMETRY. Multiple grids may be defined by repeating the READ GRID block several times, specifying a different mesh grid identification number for each so defined grid. See Sect. 8.1.3.14 for further information.

(n_{12}) READ REAC *reaction_data* END REAC

Enter the data needed to specify filters for the reaction tally calculations. Reaction data must begin with the words READ REAC and terminate with END REAC. This block is only applicable to calculations in the continuous energy mode. See Sect. 8.1.3.15.

(n_{13}) END DATA must be entered

Terminate the data for the problem.

8.1.3.2 Procedure for data input

For a standalone KENO problem, the first data records **must** be the sequence identifier (e.g., =kenovi or =kenova) and the title. The next block of data **must** be the parameters if they are to be entered. A problem can be run without entering the parameters, which causes KENO to use default values for input parameters. The remaining blocks of data can be entered in any order.

Keywords are deonated using FIXED-WIDTH. A keyword is used to identify the data that follow it. When a keyword is used, it must be entered exactly as shown in the data guide. All keywords except those ending with an equal sign must be followed by at least one blank.

small_italics correlate data with a program variable name. The actual values are entered in place of the program variable name and are terminated by a blank or a comma.

CAPITAL ITALICS identify general data items. General data items are general classes of data including:

(1) geometry data such as *UNIT INITIALIZATION* and *UNIT NUMBER DEFINITION*, *GEOMETRY REGION DESCRIPTION*, *GEOMETRY WORD*, *MIXTURE NUMBER*, *BIAS ID*, and *REGION DIMENSIONS*,

(2) albedo data such as *FACE CODES* and *ALBEDO NAMES*,

(3) weighting data such as *BIAS ID NUMBERS*, etc.

The square brackets, [and], are used to show that an entry is optional.

The broken line, |, is used as a logical "or" symbol to show that the entries to its left and right are alternatives that cannot be used simultaneously.

8.1.3.3 Title and parameter data

A *title*, a character string, must be entered at the top of the input file. The syntax is:

title a string of characters with a length of up to 80 characters, including blanks.

The PARAMETER block may contain parameter initializations for those parameters that need to be changed from their default value. The syntax for the PARAMETER block is:

READ PARA[METER] *p1* ... *pN* END PARA[METER]

or

READ PARM *p1* ... *pN* END PARM

p1 ... *pN* are *N* (*N* greater than or equal to zero) keyworded parameters that together make up the *PARAMETER DATA*

The commonly changed parameters are TME, GEN, NSK, and NPG. Seldom-changed parameters are NBK, NFB, XNB, XFB, WTH, WTL, TBA, BUG, TRK, and LNG.

The *PARAMETER DATA*, *p1 . . . pN*, consists of one or more of the parameters described below. Some of the parameters are valid either for only multigroup or continuous-energy mode. All below parameters and their values are printed in the Numeric Parameters or Logical Parameters output edits, regardless of whether the parameter is valid in the current transport mode (either multigroup or continous energy).

Floating point parameters

`RND` = *rndnum* input hexadecimal random number, a default value is provided.

`TME` = *tmax* execution time (in minutes) for the problem, default = 0.0 (no limit).

> **Caution:** Note that it is only tested at the end of each generation whether the given time limit has been exceeded. The job may be terminated without completing all generations or finalizing all results for output editing if *tmax* has not been entered carefully.

`TBA` = *tbtch* time allotted for each generation (in minutes), default = 10 minutes. If *tbtch* is exceeded in any generation, the problem is assumed to be looping. Execution is terminated, and final edits are performed. The problem can loop indefinitely on a computer if the system-dependent routine to interrupt the problem (PULL) is not functional. `TBA=` is also used to set the amount of time available for generating the initial starting points.

`SIG` = *tsigma* if entered and > 0.0, this is the standard deviation at which the problem will terminate, default = 0.0, which means do not check sigma.

`WTA` = *dwtav* the default average weight given a neutron that survives Russian roulette, *dwtav* default = 0.5.

`WTH` = *wthigh* the default value of *wthigh* is 3.0 and should be changed only if the user has a valid reason to do so. The weight at which splitting occurs is defined to be *wthigh x wtavg*, where *wtavg* is the weight given to a neutron that survives Russian roulette.

`WTL` = *wtlow* Russian roulette is played when the weight of a neutron is less than *wtlow x wtavg*. The *wtlow* default = 1.0/*wthigh*.

Note: The default values of *wthigh* and *wtlow* have been determined to minimize the deviation per unit running time for many problems.

`TTL` = *temperature_tolerance* the continuous energy cross sections must be within the *temperature_tolerance* (in degrees Kelvin) of the requested temperature for the problem to run. A negative value specifies the use of the closest temperature to that requested. TTL is ignored when `DBX` is nonzero. The default = -1.0.

Note:

If a parameter entered is not valid for the current transport mode (either multigroup or continuous energy mode), KENO usually ignores this parameter without a warning. Although a parameter is ignored, its user-defined value may appear in Numeric Parameters or Logical Parameters output edits.

THC = *ethermal_cutoff* the thermal cutoff energy for bound and free-gas moderators in continuous-energy transport. The cutoff energy for the thermal neutron transport treatments is represented by a single energy for all nuclides. If the incident energy is below THC, then thermal scattering kinematics are governed by $S(\alpha,\beta)$ data or the free-gas treatment. If incident energy is greater than THC, then the energy of the motion of the nuclei is considered negligible compared to the neutron energy. See Sect. 8.1.7.2.17 for more details. The default = 10.0 eV.

DBH = *dbrc_high* the energy cutoff (in eV) up to which the Doppler Broadening Rejection Correction (DBRC) method will be used on nuclides for which DBRC is enabled, and cross section libraries are available. DBH is used only in continuous-energy mode. Default = 210.0 eV.

DBL = *dbrc_low* the energy cutoff (in eV) down to which DBRC will be used on nuclides for which DBRC is enabled and cross section libraries are available. Only used in continuous-energy mode. Default = 0.4 eV.

MSH = *mesh_size* length (cm) of one side of a cubic mesh for tallying fluxes, fission source or fission densities. Default = 0.0. A positive non-zero value must be entered if one of MFX, CDS, FIS, or GFX parameters is defined as *YES* and **KENO grid data** input is not entered. See Sect. 8.1.4.11 for more details.

Integer parameters

GEN = *nba* number of generations to be run, default = 203.

NPG = *npb* number of neutrons per generation, default = 1000.

NSK = *nskip* number of generations (1 through *nskip*) to be omitted when collecting results, default = 3.

RES = *nrstrt* number of generations between writing restart data, default = 0. If RES is zero, restart data are not written. When restarting a problem, RES is defaulted to the value that was used when the restart data block was written. Thus, it must be entered as zero to terminate writing restart data for a restarted problem.

BEG = *nbas* beginning generation number, default = 1. If BEG is greater than 1, then restart data must be available. BEG must be 1 greater than the number of generations retrieved from the restart file.

NGP = *ngp* number of neutron energy groups to be used for tallying in continuous-energy mode. If **NGP** corresponds to a standard SCALE group structure, then the SCALE group structure will be used. If it does not correspond to a standard structure, then an equally spaced in lethargy group structure will be used. If nothing is specified for a continuous-energy problem, the SCALE 252-group structure will be used.

Note: In multigroup mode, default energy boundaries used for tallying are always obtained from the multigroup library used by transport, and **NGP** is defaulted accordingly. In continuous-energy mode, energy group boundaries read from **ENERGY** block override the default ngp value. **NGP** value printed in numeric parameters output edit may not reflect this update. The final **NGP** value is correctly printed in additional information output edit (shown as number of energy groups).

DBR = *lusedbrc* use the Doppler broadening rejection correction method. See Sect. 8.1.7.2.18 for more details. Used only in continuous-energy mode. Default = 2.

0 = no DBRC

1 = DBRC for ^{238}U only

2 = DBRC for all available nuclides (^{232}Th, ^{234}U, ^{235}U, ^{236}U, ^{238}U, ^{237}Np, ^{239}Pu, ^{240}Pu)

DBX = *db_xs_mode* option for performing problem-dependent or on-the-fly Doppler Broadening. See Sect. 8.1.7.2.19 for more details. Default = 2.

0 = no problem-dependent or on-the-fly Doppler Broadening

1 = perform problem-dependent Doppler Broadening for 1D cross sections only.

2 = perform problem-dependent Doppler Broadening for both 1D and 2D (thermal scattering data) cross sections.

CET = *ce_tsunami_mode* mode for CE TSUNAMI (See TSUNAMI-3D manual).

0 = no sensitivity calculations

1 = CLUTCH sensitivity calculation

2 = IFP sensitivity calculation

4 = GEAR-MC calculation (with CLUTCH only)

5 = GEAR-MC calculation (with CLUTCH+IFP)

7 = undersampling metric calculation

CFP = *number_of_latent_generations* number of latent generations used for IFP sensitivity or $F^*(r)$ calculations (See TSUNAMI-3D manual). If CET=1 and CFP= -1 then $F^*(r)$ is assumed to equal one everywhere. If CET=4 and CFP= -1 then $F^*(r)$ is assumed to equal zero everywhere.

NQD = *nquad* quadrature order for angular flux tallies, default = 0, which means do not collect. Angular fluxes are typically only needed for TSUNAMI-3D calculations(See TSUNAMI-3D manual).

PNM = *isctr* highest order of flux moment tallies, default = 0. Flux moments are typically only tallied for TSUNAMI-3D calculations (See TSUNAMI-3D manual).

X1D = *numx1d* number of extra 1D cross sections, default = 0.

NB8 = *nb8* number of blocks allocated for the first direct-access unit, default = 1000.

NL8 = *nl8* length of blocks allocated for the first direct-access unit, default = 512.

NBK = *nbank* number of positions in the neutron bank, default = *npb* + 25.

XNB = *nxnbk* number of extra entries in the neutron bank, default = 0.

NFB = *nfbnk* number of positions in the fission bank, default = *npb.*

XFB = *nxfbk* number of extra entries in the fission bank, default = 0.

Alphanumeric parameter data

CEP = *lcep* key for choosing the calculation mode in stand alone KENO calculations. The parameter is set to the appropriate value by the calling sequence if not stand alone KENO. For stand alone KENO, enter NO for multigroup mode, or enter the continuous energy directory filename for the continuous energy mode. The directory file is the file containing pointers to files significant for the continuous energy run.

FNI = *mode_in* extra field in the input restart file name [restart_*mode_in*.keno_input] and [restart_*mode_in*.keno_calculated]. The default is an empty field.

FNO = *mode_out* extra field in the output restart filename [restart_*mode_out*.keno_input] and [restart_*mode_out*.keno_calculated]. The default is an empty field.

Logical parameter data . . . enter *YES* or *NO*

APP = *lappend* key for appending the restart data, default = NO.

ADJ = *nadj* key for running adjoint calculation, default = NO. Adjoint cross sections must be available to run an adjoint problem. If LIB= is specified, the cross sections will be adjointed by the code. If XSC= is specified, the cross sections must already be in adjoint order.

PTB = *ptb* key for using probability tables in the continuous energy mode, default = YES

PNU = *lpromptnu* key for using prompt-only ν in the continuous energy mode, default = NO – use total.

FRE = *lfree_analytic* no longer supported (**obsolete parameter**).

UUM = *lUnionizedMix* use unionized mixture cross section, default = NO. Only used in continuous-energy mode. See Sect. 8.1.7.2.3 for more details.

M2U = *luseMap2Union* store cross sections for each nuclide on a unionized energy grid, default=NO. Only used in the continuous energy mode. See Sect. 8.1.7.2.3 for further details.

CFX = *nflx* collect fluxes, default = NO.

FLX = *nflx* key for collecting and printing fluxes, default = NO.

FDN = *nfden* key for collecting and printing fission densities, default = YES.

FAR = *lfa* key for generating region-dependent fissions and absorptions for each energy group, default = NO.

GAS = *lgas* key for printing region-dependent fissions and absorptions by energy group, applicable only if FAR = YES. Default = FAR. GAS = YES prints region-dependent data by energy group. GAS = NO suppresses region-dependent data by energy group.

NUB = *nubar* calculate the average number of neutrons per fission and the average energy group at which fission occurred, default = YES.

MFP = *mean-free-path* compute and print the mean-free-path of a neutron by region, default = NO.

SMU = *lmult* calculate the average self-multiplication of a unit, default = NO.

MKP = *larpos* calculate and print matrix k-effective by unit location, default = NO. Unit location may also be referred to as array position or position index.

CKP = *lckp* calculate and print cofactor k-effective by unit location, default = NO. Unit location may also be referred to as array position or position index.

FMP = *pmapos* print fission production matrix by array position, default = NO.

MKU = *lunit* calculate and print matrix k-effective by unit type, default = NO.

CKU = *lcku* calculate and print cofactor k-effective by unit type, default = NO.

FMU = *pmunit* print fission production matrix by unit type, default = NO.

MKH = *lmhole* calculate and print matrix k-effective by hole number, default = NO.

CKH = *lckh* calculate and print cofactor k-effective by hole number, default = NO.

FMH = *pmhole* print fission production matrix by hole number, default = NO.

MKA = *lmarry* calculate and print matrix k-effective by array number, default = NO.

CKA = *lcka* calculate and print cofactor k-effective by array number, default = NO.

FMA = *pmarry* print fission production matrix by array number, default = NO.

HHL = *lhhgh* collect matrix information by hole number at the highest hole nesting level, default = NO.

HAL = *langh* collect matrix information by array number at the highest array nesting level, default = NO.

AMX = *amx* key for printing all mixture cross section data. This is the same as activating *XAP, XS1, XS2, PKI*, and *P1D*. If any of these are entered in addition to *AMX*, then that portion of *AMX* will be overridden, default = NO.

XAP = *prtap* key for printing discrete scattering angles and probabilities for the mixture cross sections, default = NO.

XS1 = *prtp0* key for printing mixture 1D cross sections, default = NO.

XS2 = *prt1* key for printing mixture 2D cross sections, default = NO.

XSL = *prtl* key for printing mixture 2D P_L cross sections, default = NO. The Legendre expansion order L is automatically read from the cross section library.

PKI = *prtchi* print input fission spectrum, default = NO.

P1D = *prtex* print extra 1D cross sections, default = NO.

PWT = *lpwt* print weight average array, default = NO.

PGM = *lgeom* print unprocessed geometry as it is read, default = NO.

PAX = *lcorsp* print the arrays defining the correspondence between the cross section energy group structure and the albedo energy group structure, default = NO.

PMF = *prtmore* print angular fluxes or flux moments if calculated, default = NO.

PMS = *print_mesh_flux* print mesh fluxes if computed, default = NO.

PMM = *print_mesh_moments* print the angular moments of the mesh flux, if computed, default = NO.

PMV = *print_mesh_volumes* print the volume of each mesh interval, if computed. Default = NO.

TFM = *ltfm* perform coordinate transform for flux moments and angular flux calculations, default = NO.

FST = *lprint_FStar* create a .3dmap file that contains the F^*(r) mesh used by a CE-TSUNAMI CLUTCH sensitivity calculation.

SCX = *lxsecSave* save CE cross sections to restart file, default=NO.

HTM = *html_output* produce HTML formatted output for interactive browsing, sorting, and plotting of results, default = YES.

BUG = *ldbug* print debug information, default = NO. Enter *YES* for code debug purposes only.

TRK = *ltrk* print tracking information, default = NO. Enter *YES* for code debug purposes only.

RUN = *lrun* key for determining if the problem is to be executed when data checking is complete, default = YES.

PLT = *lplot* key for drawing specified plots of the problem geometry, default = YES.

Note: The parameters RUN and PLT can also be entered in the PLOT data. See Sect. 8.1.3.11. It is recommended that these parameters be entered only in the parameter data to ensure that the data printed in the Logical Parameters table is actually performed. If RUN and/or PLT are entered in both the parameter data and plot data, the results vary depending on whether the problem is run (1) stand alone, (2) as a restarted problem, (3) as CSAS with parm=check, or (4) as CSAS without parm=check. These conditions are detailed below.

KENO standalone and CSAS with PARM=CHECK The values of RUN and/or PLT entered in KENO parameter data are printed in the Logical Parameters table of the problem output. However, values for RUN and/or PLT entered in the **KENO plot data** will override the values entered in the parameter data.

Restarted KENO The values of RUN and/or PLT printed in the Logical Parameters table of the problem output are the final values from the parent problem unless those values are overridden by values entered in the **KENO parameter data** of the restarted problem. If the problem is restarted at generation 1, **KENO plot data** can be entered, and the values for RUN and/or PLT will override the values printed in the Logical Parameters table.

CSAS Without PARM=CHECK The values of RUN and/or PLT entered in the **KENO parameter data** override values entered in the KENO plot data. The values printed in the Logical Parameters table control whether the problem is to be executed and whether a plot is performed.

Parameters that are either Logical or Integer . . . enter YES or NO, or an integer number

SCD= *l_grid_entropy or mp_grid_entropy* score Shannon entropy on a grid, and then perform fission source convergence diagnostics(ScnvgDiag), default=YES, default grid ID = 10001. See Sect. 8.1.7.7 for further details.

CDS = *lgrid_prod_dens or mp_grid_prod_dens* accumulate neutron fissions on a mesh grid either constructed with **MSH** or defined by **KENO grid data** input to use as fission source in subsequent MAVRIC/Monaco shielding calculation or for visualization, default = NO.

FIS = *lgrid_fis_rate or mp_grid_fis_rate* compute fission rates on a mesh grid either constructed with **MSH** or defined by **KENO grid data** input, default = NO.

GFX = *lgrid_flux or mp_grid_flux* compute grid fluxes on a mesh grid either constructed with **MSH** or defined by **KENO grid data** input, default = NO.

MFX = *lgrid_mat_avg_flux or mp_grid_mat_avg_flux* compute mesh fluxes (averaged over the volume of mixtures/materials in each mesh voxel) on a mesh grid either constructed with **MSH** or defined by **KENO grid data** input, default = NO.

CGD = *lgrid_fstar or mp_grid_fstart* compute the F^*(r) mesh tally for continuous-energy CLUTCH sensitivity calculations. This mesh is defined with either **MSH** parameter or **KENO grid data**, default = NO.

The KENO codes in SCALE versions prior to 6.2 allowed for only one mesh definition in the user input, with either the **MSH** parameter or the **KENO grid data** input, and calculation of a single mesh-based quantity, such as MFX (mesh fluxes) or CDS (fission source accumulation on a mesh), per KENO simulation. Either of these mesh-based quantities can be enabled only by entering **MSH=yes** or **CDS=yes** in the parameter input.

Starting with SCALE 6.2, the option to define multiple spatial meshes during a single simulation was implemented in the KENO codes to add flexibility to mesh-based quantity calculations. This enabled computing desired quantities on different spatial meshes in a single KENO simulation: a finer mesh can be used for grid fluxes to increase the resolution while overlaying these data on the geometry for a problem with a full-core reactor model, whereas a coarser mesh can be used for Shannon entropy tally for source convergence diagnostics for the same problem. The new implementation requires that each mesh definition in the **KENO grid data** input have a unique NUMBER (grid ID), which is used for mesh assignment. Users can assign any spatial grid to mesh-based quantities by setting the mesh parameters to this grid NUMBER (e.g., **GFX=id1 MFX=id2**, etc.)

To support both definition formats (logical entries or integer entries), the parameter processor was redesigned for the parameters **SCD**, **CDS**, **GFX**, **MFX** and **CGD** to allow either integer or logical entries. Integer entries are required if multiple mesh-based quantities are requested with different meshes. In this case, each integer entry must point to a grid ID specified in any **KENO grid data**. See Sect. 8.1.4.11 for several examples for the use of these parameter definitions.

KENO codes in SCALE 6.3 introduce a new mesh-based quantity, fission rates on a mesh grid, which is controlled by parameter **FIS** in the parameter block. Like the above parameters, **FIS** also allows both logical and integer entries.

These entries for all above parameters are detailed below.

> **SCD**= yes enable source convergence diagnostics using the fission source accumulation on the default mesh, which is $5 \times 5 \times 5$ Cartesian mesh overlaying the whole problem geometry, generated automatically. See Sect. 8.1.7.7.
>
> **SCD**=*id* enable source convergence diagnostics using the fission source accumulation on the mesh defined with **KENO grid data** with grid ID, *id*.
>
> **CGD**=*id* enable a mesh grid defined by the **KENO grid data** with grid ID, *id* for CLUTCH F^* (r) calculations.
>
> **MFX**=yes compute mesh fluxes on intervals defined by the **MSH** parameter or by the first specified **grid data** block.

MFX=*id* compute mesh fluxes on a mesh grid defined by the **KENO grid data** with grid ID, *id*.

CDS=yes accumulate fission sources on intervals defined by **MSH** or by the first specified **grid data** block.

CDS=*id* accumulate fission source on a mesh grid defined by the specified **KENO grid data** with grid ID, *id*.

FIS=yes compute fission rates on intervals defined by **MSH** or by the first specified **grid data** block.

FIS=*id* compute fission rates on a mesh grid defined by the **KENO grid data** with grid ID, *id*.

GFX=yes compute grid fluxes (fluxes averaged over a voxel volume) on intervals defined by **MSH** or by the first specified **grid data** block.

GFX=*id* compute grid fluxes on a mesh grid defined by the **KENO grid data** with grid ID, *id*.

All of the above quantities may be requested in a single input using either the same or different grids. See Sect. 8.1.4.11 for more details.

I/O Unit Numbers **XSC** = *xsecs* I/O unit number for a Monte Carlo format mixed cross section library. When LIB $\neq 0$, default = 14. To read a mixed cross section library from a Monte Carlo format library file or CSASI, XSC must be specified.

ALB = *albdo* I/O unit number for albedo data, default = 79.

WTS = *wts* I/O unit number for weights, default = 80.

LIB = *lib* I/O unit number for *AMPX* working format cross section library, default = 0.

SKT = *skrt* I/O unit number for scratch space, default = 16.

RST = *rstrt* I/O unit number for reading restart data, default = 0. Enter a logical unit number to restart if *BEG* > 1.

WRS = *wstrt* I/O unit number for writing restart data, default = 0. A non-zero value must be entered if *RES* > 0.

GRP = *grpbs* I/O unit number for an energy group boundary library, default = 77.

Example: Default values for the parameters **NPG** and **FLX** are overridden by the user-defined values. Code continues calculations with 203 particles per generation, and tallies region-averaged fluxes in starting after **NSK** generations skipped.

```
READ PARAM
   NPG=203 FLX=yes
END PARAM
```

Example: **NSK** has been defined more than once. The last **NSK** value 50 is used. A cubic Cartesian mesh grid with 3 cm side length is constructed using the extents of the bounding box enclosing the global unit (or outermost geometry). Then, fluxes (averaged over voxel volumes) are tallied on this mesh grid after 50 generations skipped.

```
READ PARA
   NSK=13
   MSH=3.0 GFX=yes  NSK=50
END PARA
```

8.1.3.4 Geometry data

The *GEOMETRY_ DATA* consists of a series of **UNIT** descriptions, one of which may be the **GLOBAL UNIT**. The **UNIT** is the basic geometry piece in KENO and often corresponds to a well-defined physical entity (e.g., a fuel pin). A **UNIT**, therefore, may consist of multiple *material regions*. Each **UNIT** has its own, local coordinate system. The **UNIT**s are assembled to construct the problem's global geometry for KENO. The *GEOMETRY_ DATA* must be entered unless the problem is being restarted. See Sect. 8.1.4.6 for detailed examples.

UNITS

Geometric arrangements in KENO are achieved in a manner similar to using a child's building blocks. Each building block is called a **UNIT**. An **ARRAY** or lattice is constructed by stacking these **UNIT**s. Once an **ARRAY** or lattice has been constructed, it can be placed in a **UNIT** by using an **ARRAY** specification.

Each **UNIT** in an **ARRAY** or lattice has its own coordinate system. In KENO V.a, all coordinate systems in all **UNIT**s must have the same orientation. This restriction is removed in KENO-VI. All geometry data used in a problem are correlated to the absolute coordinate system by specifying a **GLOBAL UNIT**. **UNIT**s are constructed of combinations from several allowed **shape**s or geometric regions. These regions can be placed anywhere within a **UNIT**. In KENO V.a the regions are oriented along the coordinate system of the **UNIT** and do not intersect other regions. This means, for example, that a **CYLINDER** must have its axis parallel to one of the coordinate axes, while a rectangular parallelepiped must have its faces perpendicular to a coordinate axis. The most stringent KENO V.a geometry restriction is that none of the options allow geometry regions to intersect. In KENO V.a, each region in a unit must entirely contain each preceding region. The orientation, intersection, and containment restrictions are eliminated in KENO-VI. Fig. 8.1.1 shows some situations that are not allowed in KENO V.a, but are allowed in KENO-VI.

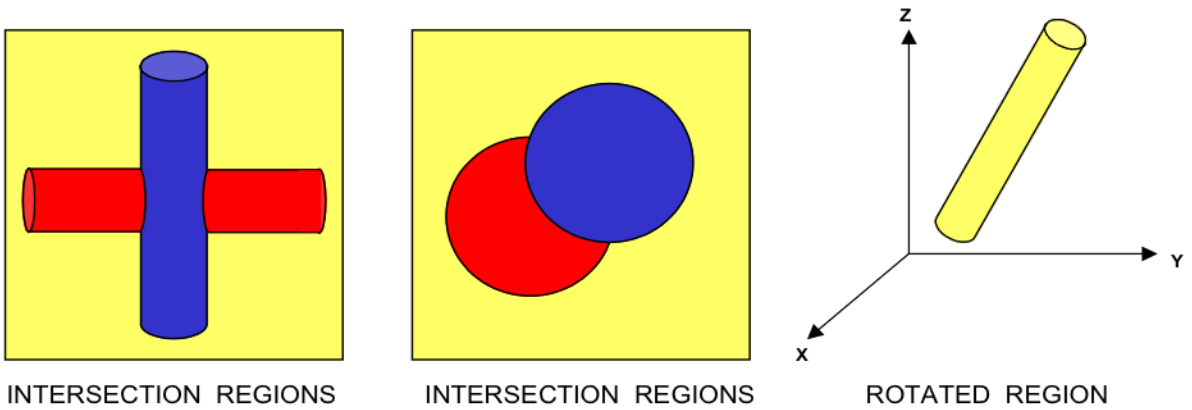


Fig. 8.1.1: Examples of geometry allowed in KENO-VI but not allowed in KENO V.a.

For KENO V.a, unless special options are invoked, each geometric region in a **UNIT** must completely enclose each interior region. Regions may touch at points of tangency and may share faces. See Fig. 8.1.2 for examples of allowable situations.

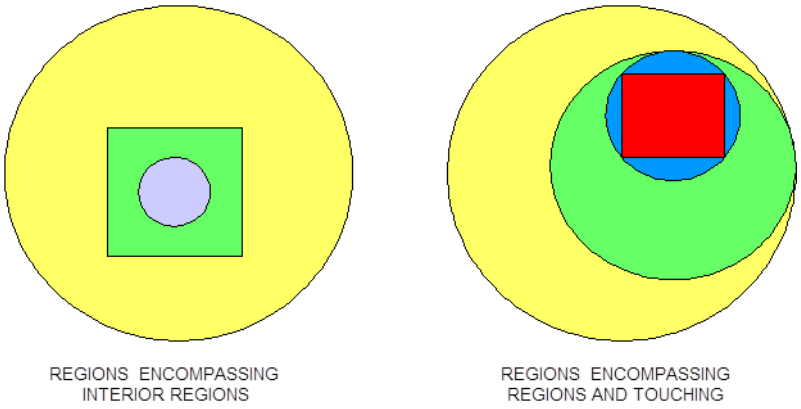


Fig. 8.1.2: Examples of correct KENO V.a units.

Special options are provided to circumvent the complete enclosure restriction in KENO V.a or to enhance the basic geometry package in KENO-VI. These options include **ARRAY** and **HOLE** descriptions. The **HOLE** option is the simplest of these and allows placing a **UNIT** anywhere within a region of another **UNIT.** In KENO V.a, **HOLE**s are not allowed to intersect the region into which they are placed; this restriction does not apply in KENO-VI (see Fig. 8.1.3). In both geometry packages, a **HOLE** cannot intersect the **UNIT** boundary. It is recommended that the outer boundary of a **UNIT** used as a **HOLE** should not be tangent to or share a boundary with another **HOLE** or a region of the **UNIT** containing the **HOLE** because the code may find that the regions are intersecting due to precision and round-off. Since a particle must check every region to determine its location within a **UNIT**, using **HOLE**s to contain complex sections of a problem may decrease the CPU time needed for the problem in KENO-VI. Inclusion of **HOLE**s increases run-time in KENO V.a, but in many cases cannot be avoided. An arbitrary number of **HOLE**s can be placed in a region in combination with a series of surrounding regions. The only restrictions on **HOLE**s are (1) when they are placed in a **UNIT,** they must be entirely contained within the **UNIT**, and (2) they cannot intersect other **HOLE**s or nested **ARRAY**s. **HOLE**s in KENO V.a cannot intersect an **ARRAY**; in KENO-VI, the **HOLE** cannot intersect the **ARRAY** boundary.

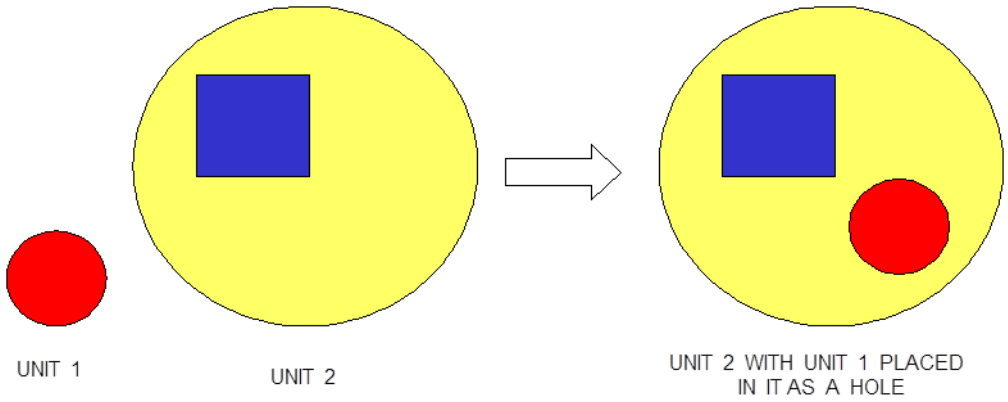


Fig. 8.1.3: Example demonstrating HOLE capability in KENO.

Lattices or arrays are created by stacking **UNIT**s. In KENO V.a, only rectangular parallelepipeds can be organized in an **ARRAY**. **HEXPRISM**s and **DODECAHEDRON**s are allowed in KENO-VI to construct triangular pitched or closed-packed dodecahedral **ARRAY**s, respectively. The adjacent faces of adjacent

UNITs stacked in this manner must match exactly. See Sect. 8.1.4.6.4 for additional clarification and Fig. 8.1.4 and Fig. 8.1.5 for typical examples.

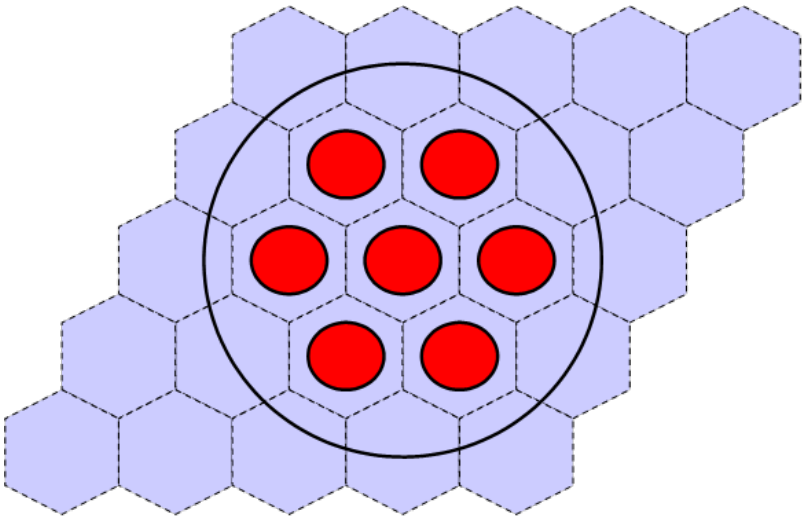


Fig. 8.1.4: Example of triangular pitched ARRAY construction.

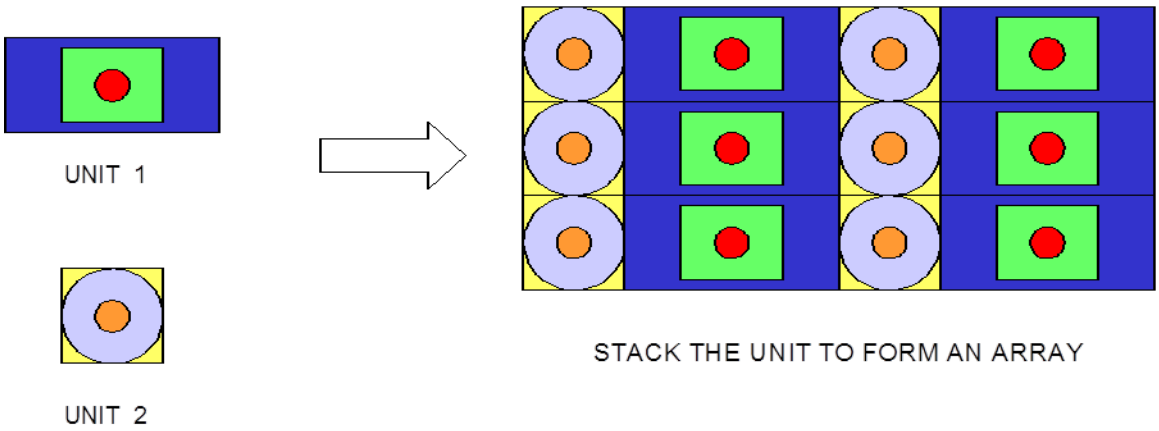


Fig. 8.1.5: Example of ARRAY construction.

The **ARRAY** option is provided to allow for placing an **ARRAY** or lattice within a **UNIT**. In KENO-VI, an **ARRAY** is placed in a **UNIT** by inserting it directly into a geometry/material region as a content record. In KENO V.a, the **ARRAY** is placed directly in the unit like a **CUBOID**: it must be the first region in the **UNIT**, or the **ARRAY** elements must intersect with the smaller region. Subsequent regions in the **UNIT** containing the **ARRAY** must contain it entirely. In KENO-VI, the reverse is true: the region boundary containing the **ARRAY** must coincide with or be contained within the **ARRAY** boundary. Therefore, in KENO-VI the region boundary becomes the **ARRAY** boundary, with the problem ignoring any part of the **ARRAY** outside the boundary. A particle enters or leaves the **ARRAY** when the region boundary is crossed. In KENO V.a, only one **ARRAY** can be placed directly in a UNIT. However, multiple **ARRAY**s can be placed within a **UNIT** by using **HOLE**s. When an **ARRAY** is placed in a **UNIT** via a **HOLE**, the **UNIT** that contains the **ARRAY** (rather than the **ARRAY** itself) is placed in the **UNIT**. **ARRAY**s of dissimilar **ARRAY**s can be created by stacking **UNIT**s that contain **ARRAY**s. In KENO-VI, it is possible to place multiple **ARRAY**s in

a **UNIT** by placing them in separate regions. Also in KENO-VI, using **HOLE**s to insert **ARRAY**s allows the **ARRAY**s to be rotated when placed. See Fig. 8.1.6 for an example of an **ARRAY** composed of **UNIT**s containing **HOLE**s and **ARRAY**s.

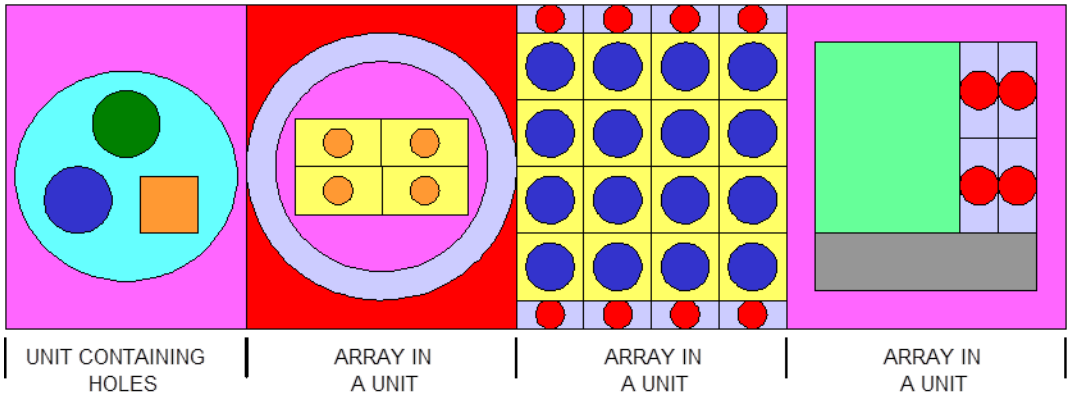


Fig. 8.1.6: Example of an ARRAY composed of UNITs containing ARRAYs and HOLEs.

The method of entering *GEOMETRY_DATA* in the geometry data block follows:

`READ GEOM` *GEOMETRY_ DATA* `END GEOM`

UNIT initialization

The description of a `UNIT` starts out with the `UNIT` *INITIALIZATION* and is terminated by encountering another `UNIT` *INITIALIZATION* or `END GEOM`.

The `UNIT` *INITIALIZATION* has the following format:

`[GLOBAL] UNIT` *u*

u is the identification number (positive integer) assigned to the particular `UNIT`. It may be used later to reference a `UNIT` previously constructed that the user wishes to place in a `HOLE`, or it may be used in an `ARRAY` (see below for more details).

`GLOBAL` is an attribute that specifies that the respective `UNIT` is the most comprehensive `UNIT` in the KENO problem to be solved, the `UNIT` that includes all the other `UNIT`s and defines the overall geometric boundaries of the problem. In general, a `GLOBAL UNIT` must be entered for each problem.

Note: In KENO V.a, the `GLOBAL` specification is optional. If it is used, it can precede either a `UNIT` command or an `ARRAY` *PLACEMENT_DESCRIPTION*. If it is not entered and the problem does not contain `ARRAY` data, `UNIT` 1 is the default `GLOBAL UNIT`. If there is no `GLOBAL UNIT` specified and `UNIT` 1 is absent from the geometry description, an error message is printed. If the geometry description contains an `ARRAY`, KENO V.a defaults the global array to the array referenced by the last `ARRAY` *PLACEMENT_DESCRIPTION* that is not immediately preceded by a unit description. Otherwise, it is the largest array number specified in the array data (Sect. 8.1.3.5).

Examples of initiating a `UNIT`:

1. Initiate input data for `UNIT` No. 6.

UNIT 6

2. Initiate input data for the **GLOBAL UNIT** which is **UNIT** No. 4.

   GLOBAL UNIT 4

For each UNIT, the UNIT's *DESCRIPTION* follows the UNIT's *INITIALIZATION*. The *DESCRIPTION* is realized by combining the commands listed below. The basic principles for constructing a UNIT are different between KENO V.a and KENO-VI. A brief discussion of these principles, together with a few examples, is presented at the end of this section following the description of the basic input used to build the geometry of a UNIT. The keywords that may be used to define a UNIT in KENO are as follows:

shape

COM=

HOLE

ARRAY

REPLICATE (KENO V.a only)

REFLECTOR (KENO V.a only)

MEDIA (KENO-VI only)

BOUNDARY (KENO-VI only)

Shape

Shape is a generic keyword used to describe a basic geometric shape that may be used in building the geometry of a particular **UNIT**. The general format varies between KENO V.a and KENO-VI. In KENO V.a, the **shape** defines a region containing a material, so the user is required to provide both a material and a *bias ID*. In KENO-VI the **shape** is used strictly as a surface, which is later used to define the mono-material regions (using the **MEDIA** card). The user is therefore required to enter a label for this surface so that the **shape** can be referenced later.

KENO V.a:

shape $m\ b\ d_1 \ldots\ d_{\mathrm{N}}\ [a_1 \ldots * [a_{\mathrm{M}}\ ] \ldots]$

KENO-VI:

shape $l\ d_1 \ldots\ d_{\mathrm{N}}\ [a_1 \ldots * [a_{\mathrm{M}}\ ] \ldots]$

shape is a generic keyword that describes a basic predefined KENO shape (e.g., **CUBOID**, **CYLINDER**) that is used to build the geometry of the **UNIT**. The predefined shapes differ between KENO V.a and KENO-VI. See Sect. 8.1.8.1 for a description of the KENO V.a basic shapes and Sect. 8.1.8.2 for the KENO-VI shapes.

m is the mixture number of the material (positive integer) that fills the particular shape in KENO V.a **UNIT** description. A material number of zero indicates a void region (i.e., no material is present in the volume defined by the **shape**).

b is the bias identification number (*bias ID*, a positive integer) assigned to the particular region defined by the **shape** in the KENO V.a **UNIT** description.

l is the label (positive integer) assigned to the particular **shape** in the KENO-VI **UNIT** description. This label is used later to define a certain mono-material region within the **UNIT**.

$d_1 \ldots d_N$ represent the N dimensions (floating point numbers) that define the particular **shape** (e.g., radius of a sphere or cylinder). See Sect. 8.1.8.1 and Sect. 8.1.8.2 for the particular value of N and how each **shape** is described.

$a_1 \ldots a_M$ are M optional *ATTRIBUTES* for the **shape**. The attributes provide additional flexibility in the **shape** description. The attributes that may be used with either KENO V.a or KENO-VI are described below (see shape ATTRIBUTES).

shape ATTRIBUTES

The *ATTRIBUTES* that can be used to enhance the **shape** description are **CHORD**, **ORIG**[**IN**], **CENTER**, and **ROTATE** (KENO-VI only).

The CHORD attribute

This attribute has different formats in KENO V.a and KENO-VI. The user will notice that it is more restrictive in KENO V.a. Only the **HEMISPHERE and HEMICYLINDER shape**s can be **CHORD**ed in KENO V.a, but all 3-D shapes may be **CHORD**ed in KENO-VI.

KENO V.a: **CHORD** ρ

KENO-VI: **CHORD** [**+X**=x+] [-**X**=x-] [**+Y**=y+] [-**Y**=y-] [**+Z**=z+] [-**Z**=z-]

p **is the distance** ρ **from the cut surface to the center of the sphere** or the axis of a hemicylinder. See Fig. 8.1.7 and Fig. 8.1.8. Negative values of ρ indicate that less than half of the **shape** is retained, while positive values indicate that more than half of the **shape** will be retained.

+X=, -X=, +Y=, -Y=, +Z=, -Z= are subordinate keywords that define the axis parallel to the chord. The "+" and "-" signs are used to define the side of the chord which is included in the volume. A "+" in the keyword indicates that the more positive side of the chord is included in the volume. A "-" in the keyword indicates that the more negative side of the chord is included in the volume.

x+, x-, y+, y-, z+, z- are the coordinates of the plane perpendicular to the chord. For each chord added to a body, the keyword **CHORD** must be used, followed by one of the subordinate keywords and its dimension.

In KENO V.a, the CHORD attribute is applicable for only hemispherical and hemicylindrical shapes, *not* for SPHERE, XCYLINDER, YCYLINDER, CYLINDER, ZCYLINDER, CUBE, or CUBOID.

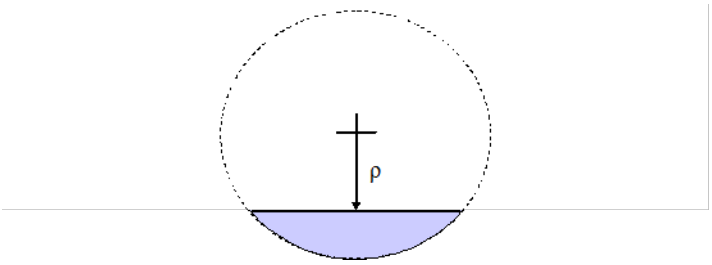


Fig. 8.1.7: Partial hemisphere or hemicylinder; less than half exists (less than half is defined by $\rho < 0$).

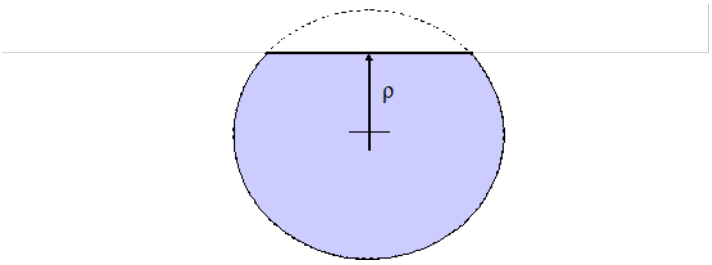


Fig. 8.1.8: Partial hemisphere of hemicylinder; more than half exists (more than half is defined by $\rho > 0$).

Fig. 8.1.9 provides two examples of the use of the CHORD option in KENO-VI.

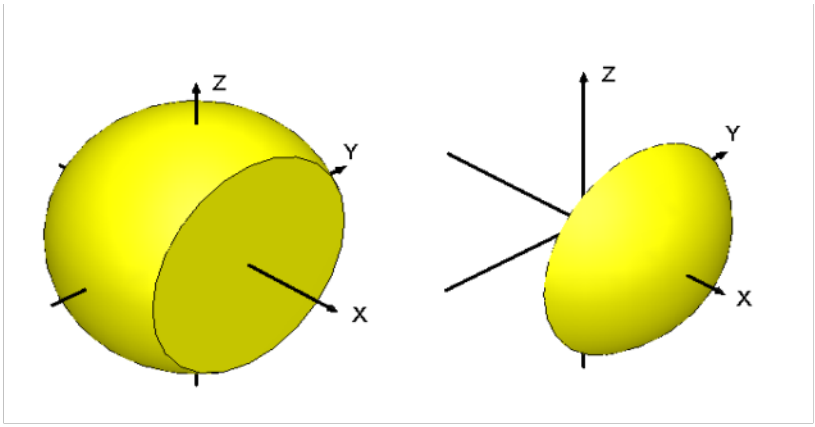


Fig. 8.1.9: Examples of the CHORD option in KENO-VI.

The ORIG[IN] attribute

The format is slightly different between KENO V.a and KENO-VI. Since the entries in KENO-VI are key worded, the user has more flexibility in choosing the order of these entries or in using default values. Only non-zero values must be entered in KENO-VI, but all applicable values, whether zero or non-zero, must be entered in KENO V.a.

KENO V.a: **ORIG**[**IN**] *a b* [*c*]

KENO-VI: **ORIGIN** [**X**=x_0] [**Y**=y_0] [**Z**=z_0]

a is the X coordinate of the origin of a sphere or hemisphere; the X coordinate of the centerline of a Z or Y cylinder or hemicylinder; the Y coordinate of the centerline of an X cylinder or hemicylinder.

b is the Y coordinate of the origin of a sphere or hemisphere; the Y coordinate of the centerline of a Z cylinder or hemicylinder; the Z coordinate of the centerline of an X or Y cylinder or hemicylinder.

c is the Z coordinate of the origin of a sphere or hemisphere; it must be omitted for all cylinders or hemicylinders.

X=, Y=, Z= are the subordinate keywords used to define the new position of the origin of the **shape**. If the a subordinate keyword appears more than once after the **ORIGIN** keyword, the values are summed. If the new value is zero, the particular coordinate does not need to be specified.

x_0, y_0, z_0 are the values for the new coordinates where the origin of the **shape** is to be translated.

The CENTER attribute

This attribute establishes the reference center for the flux moment calculations, which can be useful in TSUNAMI calculations. The syntax for this attribute is:

CENTER *center_type* [*u*] [*x y z*]

center_type is the reference center value, as described in Table 8.1.18. The default value is **global**.

u is the **UNIT** number to be used as a reference center for this region when the *center_type* is **unit**.

x, y, z are the offset from the point specified by the *center_type*. The default is 0.0 for all three entries.

Table 8.1.18: Reference center values

center_type	Reference point
unit	Reference is defined as the origin of **UNIT** *unit_number* plus the offset defined by *x*, *y*, and *z*.
global	Reference is defined as system origin-i.e., (0,0,0) point of the **GLOBAL UNIT**-plus the offset defined by *x*, *y*, and *z*.
local	Reference is defined as the origin of the current **UNIT** plus the offset defined by *x*, *y*, and *z*.
fuelcenter	Reference is defined as the center of all fissile material in the system plus the offset defined by *x*, *y*, and *z*.
wholeunit	When entered for the first region in a unit, the reference for all regions in the unit are defined as the origin of the current unit plus the offset defined by *x*, *y*, and *z*.

The ROTATE attribute

This attribute can only be used in the KENO-VI input. It allows for the rotation of the **shape** or **HOLE** to which it is applied. If **ORIGIN** and **ROTATE** data follow the same **shape** or **HOLE** record, the **shape** is always rotated prior to translation, regardless of the order in which the data appear. Fig. 8.1.10 provides an example of the use of the **ROTATE** option. Its syntax is:

ROTATE [**A1**=a_1] [**A2**=a_2] [**A3**=a_3]

A1=, A2=, A3= are subordinate keywords to specify the angles of rotation of the particular shape with respect to the origin of the coordinate system. The Euler X-convention is used for rotation.

a_1, a_2, a_3 are the values of the Euler rotation angles in degrees. The default is 0 degrees. If a subordinate keyword appears more than once following the ROTATE keyword, the values are summed.

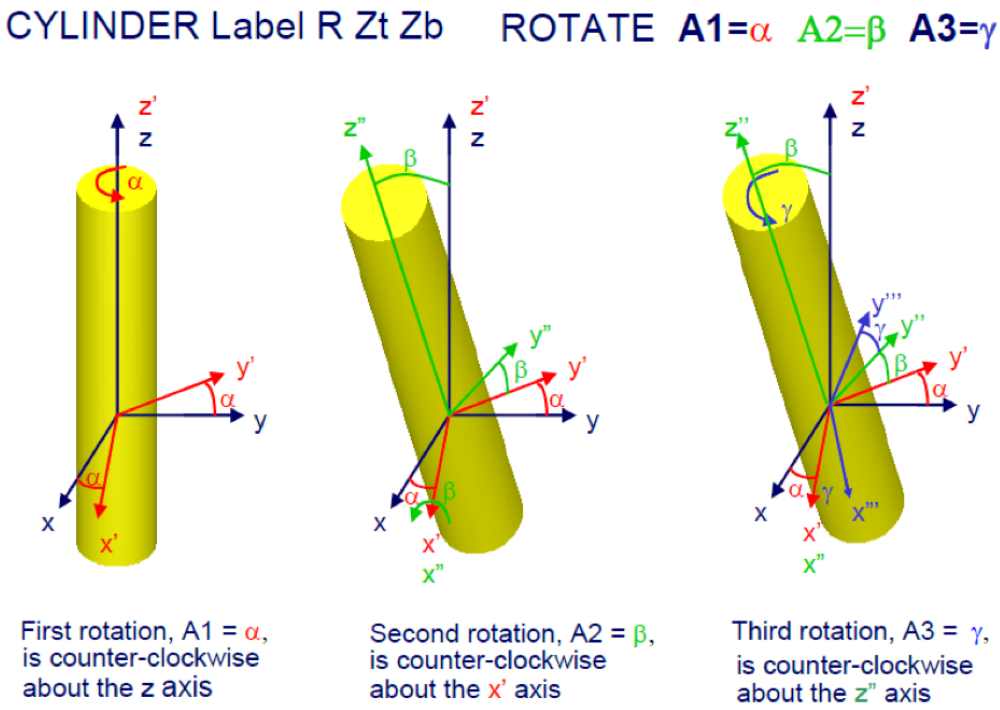


Fig. 8.1.10: Explanation of the ROTATE option.

Examples of **shape**s:

1. Specify a hemisphere labeled 10, containing material 2 with a radius of 5.0 cm which contains only material where Z > 2.0 within the sphere centered at the origin, and its origin translated to X=1.0, Y=1.5, and Z=3.0. KENO V.a (no label, but material and bias ID are the first two numerical entries):

   HEMISPHERE 2 1 5.0 **CHORD** -2.0 **ORIGIN** 1.0 1.5 3.0

   or

   HEMISPHE+Z 2 1 5.0 **CHORD** -2.0 **ORIGIN** 1.0 1.5 3.0

   KENO-VI (no material; this is to be specified with **MEDIA**):

   SPHERE 10 5.0 **CHORD +Z**=2.0 **ORIGIN X**=1.0 **Y**=1.5 **Z**=3.0

2. Specify a hemicylinder labeled 10, containing material 1, having a radius of 5.0 cm and a length extending from Z=2.0 cm to Z=7.0 cm. The hemicylinder has been truncated perpendicular to the X axis at X= -3 such that material 1 does not exist between X= -3 and X= -5. Position the origin of the truncated hemicylinder at X=10 cm and Y=15 cm with respect to the origin of the unit, and rotate it (in KENO-VI input) so it is in the YZ plane at X=10 and at a 45° angle with the Y plane.

   KENO V.a (no rotation possible, no label):

   ZHEMICYL+X 1 1 5.0 7.0 2.0 **CHORD** 3.0 **ORIGIN** 10.0 15.0

   KENO-VI (no material; this is to be specified with **MEDIA** card):

   CYLINDER 10 5.0 7.0 2.0 **CHORD +X**= -3.0 **ORIGIN X**=10.0 **Y**=15.0 **ROTATE A2**= -45

COM=

The keyword **COM**= signals that a comment is to be read. The optional comment can be placed anywhere within a unit definition. Its syntax is:

COM = *delim comment delim*

delim is the delimiter, which may be any one of " , ` , * , ^ , or !

comment is the comment string, up to 132 characters long.

Example of comment within a **UNIT**:

COM="This is a fuel pin"

HOLE

This entry is used to position a **UNIT** within a surrounding **UNIT** relative to the origin of the surrounding **UNIT**. **HOLE**s may share surfaces with but may not intersect other **HOLE**s, the **BOUNDARY** of the **UNIT** which contains the **HOLE**, or an **ARRAY** boundary. In KENO-VI, the **BOUNDARY** record of a **UNIT** placed in a **HOLE** may contain more than one geometry label, but all labels must be positive, indicating inside the respective geometry bodies. The syntax for **HOLE** is:

KENO V.a: **HOLE** *u x y z*

KENO-VI: **HOLE** *u* [a_1 ... [a_M]...]

u is the unit previously defined that is to be placed within the **HOLE**.

x y z is the position of the **HOLE** in the KENO V.a host **UNIT**.

a1 ... aM are optional KENO-VI *ATTRIBUTES* for the **HOLE**. The *ATTRIBUTES* can be **ORIGIN** or **ROTATE** and follow the same syntax previously defined for KENO-VI **shape** *ATTRIBUTES*. These *ATTRIBUTES* allow for the translation and/or rotation of the **HOLE** within the host region.

Examples of **HOLE** use:

Place **UNIT** 2 in the surrounding **UNIT** such that the **ORIGIN** of **UNIT** 2 is at **X**=3, **Y**=3.5, **Z**=4 relative to the origin of the surrounding **UNIT**.

KENO V.a: **HOLE** 2 3 3.5 4

KENO-VI: **HOLE** 2 **ORIGIN X**=3.0 **Y**=3.5 **Z**=4.0

ARRAY

When used within a **UNIT** description, this entry provides an **ARRAY** placement description. In KENO V.a, it always starts a new **UNIT** and generates a rectangular parallelepiped that fits the outer boundaries of the specified **ARRAY**. The specified **ARRAY** is positioned in the **UNIT** according to the most negative point in the **ARRAY** with respect to the coordinate system of the surrounding **UNIT**. Thus, the location of the minimum x, minimum y, and minimum z point in the array is specified in the coordinate system of the **UNIT** into which the **ARRAY** is being placed.

In KENO-VI, the **ARRAY** keyword is used to position an **ARRAY** within a region in a surrounding **UNIT** relative to the origin of the surrounding **UNIT**. When the subordinate keyword **PLACE** is entered, it is followed by six numbers that precisely locate the **ARRAY** within the surrounding **UNIT** as shown in the example below. The first three numbers consist of the element in the **ARRAY** of the **UNIT** selected to position the **ARRAY**. The next three numbers consist of the position of the origin of the selected **UNIT** in

the surrounding **UNIT**. Higher level **ARRAY** boundaries may intersect lower level **ARRAY** boundaries as long as they do not intersect **HOLE**s in the **UNIT**s contained in the **ARRAY** or in **UNIT**s contained in lower level **ARRAY**s.

The syntax for the **ARRAY** card is as follows:

KENO V.a: **ARRAY** *array_id x y z*

KENO-VI: **ARRAY** *array_id* $l_1 \ldots l_N$ [**PLACE** N_x N_y N_z *x y z*]

array_id is the label that identifies the array to be placed.

l1 … lN is the *REGION DEFINITION VECTOR*. These are previously defined **shape** labels, and together they define the region in which the array *array_id* is to be placed. This is used only in KENO-VI.

Nx Ny Nz are three integers that define the element in the **ARRAY** of the **UNIT** selected to position the **ARRAY**. This is used only in KENO-VI.

x y z specify the position of the **ARRAY** in the **UNIT**.

In KENO V.a, the x, y, and z values are the point where the most negative x, y, and z point of the **ARRAY** is to be located in the **UNIT**'s coordinates.

In KENO-VI, the x, y, and z values are the point where the origin of the **UNIT** specified by *Nx*, *Ny*, and *Nz* is to be located in the **shape** specified by the *REGION DEFINITION VECTOR*.

Example of **ARRAY** use:

In KENO V.a, position the most negative point of **ARRAY** 6 at X = 2.0, Y = 3.0, Z = 4.0 relative to the origin of the containing **UNIT**.

ARRAY 6 2.0 3.0 4.0

In KENO-VI, position instead the origin of **UNIT** (1,2,3) of **ARRAY** 6 at X = 2.0, Y = 3.0, Z = 4.0 and specify the **ARRAY** boundary to be the region that is inside the geometry **shape**s labeled 10 and 20 and outside the geometry **shape** labeled 30 used to describe the surrounding **UNIT**.

ARRAY 6 10 20 -30 **PLACE** 1 2 3 2.0 3.0 4.0

REPLICATE and REFLECTOR

These keywords specific to KENO V.a are used to generate additional geometry regions having the **shape** of the previous region. The geometry keyword **REFLECTOR** is a synonym for **REPLICATE**. The desired weighting functions can be applied to those regions by specifying biasing data as described in Sect. 8.1.3.7. The total thickness generated for each surface is the thickness per region for that surface times the number of regions to be generated, *nreg*.

The replicate specification is frequently used to generate weighting regions external to an `ARRAY` placement description. Thus an `ARRAY` placement description followed by a `REPLICATE` description would generate regions of a cuboidal shape. A cylindrical reflector could be generated by following the `ARRAY` placement description with a `CYLINDER` and then a `REPLICATE`. A `HOLE` cannot immediately follow a `REPLICATE`.

Extra regions using default weights can be generated by specifying the first importance region, *imp*, to be one that was not defined in the *BIASING INFORMATION* provided in a `READ BIAS` block. This capability can be used to generate extra regions for collecting information such as fluxes, leakage, etc.

Multiple replicate descriptions can be used in any problem. This capability can be used to model different reflector materials of different thicknesses on different faces.

The number of appropriate region dimensions needed for specifying REPLICATE is determined by the preceding region. For example, if the previous region were a SPHERE, one entry (i.e., t_1) would be required. If the previous region were a CYLINDER, the first entry, t_1, would be the thickness/region in the radial direction, the second entry, t_2, would the thickness/region in the positive length direction, the third entry, t_3, would be the thickness/region in the negative length direction, etc. The REPLICATE specification requirements for a CUBE are the same as for a CUBOID.

Syntax:

REPLICATE|REFLECTOR *m b* $t_1 \ldots t_N$ *nreg*

m is the number of the material (non-negative integer) that fills the particular REPLICATE/REFLECTOR region in the **UNIT** description. A material of zero indicates a void region (i.e., no material is present in the volume defined by the **shape**).

b is the bias identification number (positive integer) assigned to the particular region defined by the shape in the KENO V.a UNIT description. If the specified bias ID is defined in a READ BIAS block, the bias ID number will be incremented automatically, increasing one for each additional region up to *nreg*.

$t_1 \ldots t_N$ represent the thickness (floating point number) per region for each of the *N* surfaces that define the particular shape. If the specified bias ID is one that is defined in the READ BIAS block, the region thicknesses should be consistent with the thicknesses used to generate the bias data being used. See Sect. 9.1.2.7.

nreg is the number of regions (integer) to be generated.

Example:

Create five regions of material 4, each being 3 cm thick, outside a cuboid region (a cuboid has six dimensions). The inner-most of the five generated regions has a *bias id* of 2. The following four regions have *bias id* of 3, 4, 5, and 6.

MEDIA

This card is used in the KENO-VI input file to define the location of a mixture relative to the geometric shapes in the UNIT. Fig. 8.1.11 shows the input for a set of three intersecting SPHEREs in a CUBOID. The total volume data for a region in the problem may be entered as the last entry on the MEDIA card by using the VOL= keyword immediately followed by the volume in cm^3. The volume entered is the volume of the region in the unit multiplied by the number of times the unit occurs in the problem minus any volume excluded from the problem by ARRAY boundaries and HOLEs. The volumes for any or all regions may be entered. If the volume is entered here, this value will be used even if volumes are also entered as a file or calculated (See Sect. 8.1.3.13). Volumes not entered will be determined by the input specified in the VOLUME DATA block. If no volume is supplied, the KENO-VI default volume of -1 will be used. This only affects volume-averaged quantities, i.e., not k_{eff}.

Syntax:

MEDIA *m b* $l_1 \ldots l_N$ [VOL = *v*]

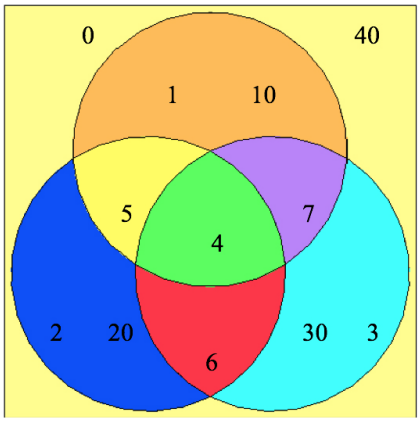
m is the material (positive integer or zero for vacuum) that fills the region defined by MEDIA.

b is the *bias id* for the material sector being defined.

$l_1 \ldots l_N$ is the region definition vector (*N* integers). These are *N* previously defined **shape** labels that together define the material sector.

VOL = is an optional sub-keyword used to input the material sector volume.

v is the volume in cm^3 of the material sector defined by the MEDIA card.




```
GLOBAL UNIT 1
SPHERE 10 10.0 ORIGIN Z=4.5
SPHERE 20 10.0 ORIGIN Y= -4.5 Z= -4.5
SPHERE 30 10.0 ORIGIN Y=4.5 Z= -4.5
CUBOID 40 6P15.0
MEDIA 1 1 10 -20 -30   VOL=2210.8
MEDIA 2 1 20 -10 -30   VOL=2210.8
MEDIA 3 1 30 -10 -20   VOL=2210.8
MEDIA 4 1 10 20 30     VOL=672.39
MEDIA 5 1 10 20 -30    VOL=652.8
MEDIA 6 1 20 30 -10    VOL=652.8
MEDIA 7 1 10 30 -20    VOL=652.8
MEDIA 0 1 40 -10 -20 -30   VOL=17736.81
BOUNDARY 40 ]
```

Fig. 8.1.11: Example of the MEDIA record.

BOUNDARY

This card is used in KENO-VI to define the outer boundary of the UNIT. In KENO V.a, the outer boundary of the UNIT is implicitly defined by the last shape in the UNIT. Each UNIT must have one and only one BOUNDARY card.

Syntax:

BOUNDARY *l1 ... lN*

***l*$_1$ ... *l*$_N$** is the UNIT *BOUNDARY DEFINITION VECTOR* (*N* integers). These are *N* previously defined **shape** labels that together define the outer boundary of the **UNIT**. All entries must be positive for the UNIT being defined to be used subsequently as a HOLE.

8.1.3.5 ARRAY Data

The array definition data block is used to define the size of an **ARRAY** and to position **UNIT**s (defined in the geometry data) in a 3-D lattice that represents the **ARRAY** being described. As many arrays as are necessary can be described in a problem, subject to computer storage limitations. In KENO V.a, only one **ARRAY** may be placed directly in a **UNIT**, but as many **ARRAY**s as are needed may be placed in the **UNIT** by using **HOLE**s. In KENO-VI, any number of arrays can be placed in any **UNIT** either directly or indirectly using **HOLE**s. There is no default global array. If a global array is desired it must be explicitly defined.

The **ARRAY** definition data is entered as:

READ ARRAY ARRAY DATA END ARRAY

The *ARRAY_ DATA* consists of *ARRAY_PARAMETERS* and *UNIT_ORIENTATION_DESCRIPTION*.

ARRAY Parameters

The ARRAY parameters that can be used in the definition of an ARRAY are:

ARA=

GBL=

NUX=, **NUY**=, **NUZ**=

PRT=

COM=

TYP= (KENO-VI only)

ARA=

The **ARA**= parameter defines a reference number for an **ARRAY**. It has no default in KENO-VI. In KENO V.a, if is missing, the default is 1.

Syntax:

ARA=*a*

a is the reference number for the **ARRAY**. It has no default in KENO-VI. In KENO V.a, if is missing, the default is 1.

GBL=

This is used to input the number of the global array.

Syntax:

GBL=*g*

g is the reference number for the global **ARRAY**. In KENO V.a it must not be entered more than once. The default is the largest value for *a*, the reference number for the **ARRAY**. In KENO-VI it is no default value and if entered more than once, the last value is used.

PRT=

This entry is used to enable printing the **ARRAY** of **UNIT** numbers.

Syntax:

PRT=*print*

print is a logical constant which defaults to YES, indicating that the **ARRAY** of **UNIT** numbers is printed. If the value is NO, then a summary table is printed instead containing the number of times each unit is used in each array.

NUX=, NUY=, NUZ=

These entries are used to input the number of units in the X, Y, and Z directions, respectively.

Syntax:

NUX=n_x **NUY**=n_y **NUZ**=n_z

$\boldsymbol{n_x}$ $\boldsymbol{n_y}$ $\boldsymbol{n_z}$ are the number of units in the X, Y, and Z directions, respectively. There is no default in KENO-VI. In KENO V.a, each of them defaults to 1.

TYPE=

This entry is used to specify the type of **ARRAY** and is specific to KENO-VI, where more than one type of arrays can be used. It cannot be used in KENO V.a.

Syntax:

TYP=*atyp*

atyp type of array (**cuboidal** or **square**, **hexagonal** or **triangular**, **rhexagonal**, **shexagonal**, **dodecahedral**), default = **cuboidal**

COM=

This keyword is used to enter a comment.

Syntax:

COM=*delim comment delim*

delim is a delimiter. Acceptable delimiters are ", `, * , ^ , or !.

comment is the comment string. Maximum *comment* length is 132 characters.

ARRAY orientation data

There are two methods to enter the **UNIT** numbers constituting an **ARRAY**: **LOOP** and **FILL**.

LOOP Construct

The `LOOP` construct resembles a FORTRAN DO-loop construct. The arrangement of `UNIT`s may be considered as consisting of a 3-D matrix of `UNIT` numbers, with the `UNIT` position increasing in the positive X, Y, and Z directions, respectively.

Syntax:

`LOOP` *u* ix_1 ix_2 incx iy_1 iy_2 incy iz_1 iz_2 *incz* `END LOOP`

u is the `UNIT` identification number (a positive integer).

ix_1 is the starting position in the X direction; ix_1 must be at least 1 and less than or equal to n_x of Sect. 8.1.3.5.5.

ix_2 is the ending position in the X direction; ix_2 must be at least 1 and less than or equal to *nx*.

incx is the number of `UNIT`s by which increments are made in the positive X direction; *incx* must be greater than zero and less than or equal to n_x.

iy_1 is the starting position in the Y direction; iy_1 must be at least 1 and less than or equal to n_y.

iy_2 is the ending position in the Y direction; iy_2 must be at least 1 and less than or equal to n_y.

incy is the number of **UNIT**s by which increments are made in the positive Y direction; *incy* must be greater than zero and less than or equal to n_y.

iz_1 is the starting position in the Z direction; iz_1 must be at least 1 and less than or equal to n_z.

iz_2 is the ending position in the Z direction;, iz_2 must be at least 1 and less than or equal to n_z.

incz is the number of **UNIT**s by which increments are made in the positive Z direction; *incz* must be greater than zero and less than or equal to n_z.

The syntax for ending the **LOOP** construct is:

`END LOOP`

The sequence *u* through *incz* is repeated until the entire **ARRAY** is described. If any portion of an **ARRAY** is defined in a conflicting manner, the last entry to define that portion will determine the **ARRAY**'s configuration. To use this feature, fill the entire **ARRAY** with the most relevant **UNIT** number and superimpose the other **UNIT** numbers in their proper places. An example showing the use of the **LOOP** option is given below. This $5 \times 4 \times 3$ **ARRAY** of **UNIT**s is a matrix of **UNIT**s that has 5 **UNIT**s stacked in the X direction, 4 **UNIT**s in the Y direction, and 3 **UNIT**s in the Z direction. X increases from left to right, and Y increases from bottom to top. Each Z layer is shown separately.

Given:

```
1 2 1 2 1        2 1 2 1 2       1 1 1 1 1
1 1 1 1 1        2 2 2 2 2       1 3 3 3 1
1 1 1 1 1        2 2 2 2 2       1 3 3 3 1
1 2 1 2 1        2 1 2 1 2       1 1 1 1 1
Z Layer 1        Z Layer 2       Z Layer 3
```

The data for this array could be entered using the following entries.

(1) 1 1 5 1 1 4 1 1 3 1 This fills the entire array with 1s.

(2) 2 2 5 2 1 4 3 1 1 1 This loads the four 2s in the first Z layer.

(3) 2 1 5 1 2 3 1 2 2 1 This loads the second and third rows of 2s in the second Z layer.

(4) 2 1 5 2 1 4 3 2 2 1 This loads the desired 2s in the first and fourth rows of the second Z layer.

(5) 3 2 4 1 2 3 1 3 3 1 This loads the 3s in the third Z layer and completes the array data input.

The second layer could have been defined by substituting the following data for entries (3) and (4):

(3) 2 1 5 1 1 4 1 2 2 1 This completely fills the second layer with 2s.

(4) 1 2 4 2 1 4 3 2 2 1 This loads the four 1s in the second layer.

When using the **LOOP** option, there is no single correct method of entering the data. If a **UNIT** is improperly positioned in the **ARRAY** or if some positions in the **ARRAY** are left undefined, it is often easier to add data to correctly define the **ARRAY** than to try to correct the existing data.

FILL Construct

The **FILL** construct enters data by stringing in **UNIT** numbers starting at X=1, Y=1, Z=1, and varying X, then Y, and then Z to fill the **ARRAY**. n_x x n_y x n_z entries are required. FIDO-like input options specified in Table 8.1.19 are also available for filling the **ARRAY**.

Syntax:

`FILL` $u_1 \ldots u_N$ {`END FILL`}|`T`

$u_1 \ldots u_N$ are the N=n_x x n_y x n_z `UNIT` numbers that make up the `ARRAY`

The syntax for ending the `FILL` construct is

`END FILL`

An alternative to end the `UNIT` data in `FILL` is by entering the letter `T`.

Table 8.1.19: FIDO-like input for mixed box orientation fill option.

Count field	Option field	Operand field	Function
		j	stores *j* at the current position in the array
i	R	*j*	stores *j* in the next *i* positions in the array
i	*	*j*	stores *j* in the next *i* positions in the array
i	$	*j*	stores *j* in the next *i* positions in the array
	F	*j*	fills the remainder of the array with unit number *j*, starting with the current position in the array
	A	*j*	sets the current position in the array to *j*
i	S		increments the current position in the array by *i* (This allows for skipping *i* positions; *i* may be positive or negative.)
i	Q	*j*	repeats the previous *j* entries *i* times (default value of *i* is 1)
i	N	*j*	repeats previous *j* entries *i* times, inverting the sequence each time. (default value of *i* is 1)
i	B	*j*	backs *i* entries. From that position, repeats the previous *j* entries in reverse order (default value of *i* is 1)
i	I	*j k*	provides the end points j and k, with i entries linearly interpolated between them (i.e., a total of i+2 points). At least one blank must separate j and k. When used for an integer array, the I option should only be used to generate integer steps-i.e., (k-j)/(i+1) should be a whole number
	T		terminates the data reading for the array

Note: When entering data using the options in this table, the *count field* and *option field* must be adjacent with no imbedded blanks. The operand field may be separated from the option field by one or more blanks.

Example: Consider a $3 \times 3 \times 1$ `ARRAY` filled with 8 `UNIT` 1s and a `UNIT` 2, as shown below.

```
1     1     1
1     2     1
1     1     1
```

The input data to describe this `ARRAY` could be entered as follows:

Option (1) **`1 1 1 1 2 1 1 1 1 T`** This fills the array one position at a time, starting at the lower left corner. The `T` terminates the data.

or

Option (2) **`F1 A5 2 END FILL`** The F1 fills the entire array with 1s, the A5 locates the fifth position in the array, and the 2 loads a 2 in that position. The `END FILL` terminates the data.

8.1.3.6 Albedo data

Albedo boundary conditions are entered using a *FACE CODE* to define where albedo conditions are to be used, and using an *ALBEDO NAME* to indicate which albedo condition is to be used on that face. The default value for each face is vacuum or void. The default values are overridden only on faces for which other albedo names are specified. Albedo boundary conditions are applied only to the outermost region of a problem (global boundaries). Different albedo options are allowed for different global boundaries, and this may show some differences in both KENO V.a and KENO-VI.

In previous SCALE versions, KENO V.a would have allowed the use of albedo boundary conditions to the boundary faces if and only if the outermost geometry region was a cuboid. For non-cuboidal outermost geometries, only the void or vacuum albedo option could have been applied. This limitation has been relaxed in SCALE 6.3; KENO V.a now allows some albedo options for non-cuboidal boundary shapes.

Unlike KENO V.a, KENO-VI allows a combination of geometry shapes to be used to define the boundaries of the global unit. Combinations of more than one geometry shape may result in re-entrant surfaces in the model. KENO-VI users need to be aware that when a neutron reaches a surface with a vacuum albedo, that neutron exits the model and the history ends. If a model contains features that are reentrant, that is a neutron could exit the model and reenter the model on the other side of an unmodeled region, all neutrons passing through the problem boundary are lost when they reach the unmodeled region. Neutrons are not "transported" across unmodeled areas between reentrant surfaces. It is not possible to create a KENO V.a model with reentrant problem outer boundary surfaces.

The syntax for entering the albedo boundary conditions is as follows. Note that `BODY` is no longer considered as a *FACE CODE* in the KENO-VI albedo boundary conditions capability, unlike the previous versions of SCALE.

KENO V.a:

`READ BOUNDS` **fc**$_1$=a_1 [**fc**$_2$=a_2]... [**fc**$_N$=a_N] `END BOUNDS`

KENO-VI:

`READ BOUNDS` [**BODY**=*body_label*] **fc**$_1$=a_1 [**fc**$_2$=a_2]... [**fc**$_N$=a_N] `END BOUNDS`

fc$_1$ ... **fc**$_N$ are **N** *FACE CODE*s, each refers to a single or combination of multiple faces of the global boundaries (faces of the outermost region of a problem). *FACE CODE*s for different outer shapes are defined in Table 8.1.20.

a_1 ... a_N are the *ALBEDO NAME*s as defined in Table 8.1.23.

BODY= refers to the body or shape label in the global unit input. This optional parameter is available only for KENO-VI. The use of **BODY** with any face code combination is an optional feature for the problems in which the boundary definition vector of the global unit consists of a single body label (e.g. **BOUNDARY 10**). However, any face code used to specify an albedo condition on any face of a boundary shape must follow the **BODY=** parameter when the boundary definition vector of the global unit has more than one boundary labels (e.g., **BOUNDARY 10 20 30**). In such a case, all albedo boundary condition specifications following the **BODY=** definition are applied only on the faces of the body given with that **BODY=** definition.

body_label is the integer label of one of the shapes or bodies listed in the boundary definition vector of the global unit.

Albedo boundary conditions may be entered on each face of the global boundaries multiple times. The boundary condition that applies to the boundary face is the last one entered. If no boundary data are entered or if no albedo boundary condition is applied to a boundary face, then this boundary face is assumed to have a void or vacuum boundary condition. Similarly, albedo boundary conditions may be entered on each face of a boundary body listed in the boundary definition vector of the global unit multiple times. In this KENO-VI specific case, the boundary condition that applies to the boundary face of that body is the last one entered.

Example: Apply the reflective albedo boundary condition to all faces of a cuboidal outer boundary except the positive *X* face.

In this sample problem, KENO with the **ALL** face code first applies the reflective albedo boundary condition to all faces of the outermost geometry. Then, the **+XB=** face code overrides the albedo type as *VACUUM* for the positive *X* face of the cuboidal outer boundary.

KENO V.a:

```
READ BOUNDS
   ALL=mirror +XB=vacuum
END BOUNDS
```

KENO-VI:

```
read geometry
   ...
   global unit 1
      CUBOID 10 ...
      ...
      BOUNDARY 10
end geometry

READ BOUNDS
   ALL=mirror +XB=vacuum
END BOUNDS

' following albedo specification is identical with the above one
' READ BOUNDS
'    BODY=10 ALL=mirror +XB=vacuum
' END BOUNDS
```

Example: Apply a material-specific albedo *H2O* to all faces of the global boundaries except the bottom face of the cuboid, which is one of the boundary shapes. Another material-specific albedo *CONC24* is applied to the bottom face of this cuboid.

This sample problem demonstrates the albedo boundary condition specification for a KENO-VI model in which outer boundary of the global unit is defined by a combination of a cuboid and a sphere (CUBOID 10 and SPHERE 20). KENO-VI, with this input, first applies a material-specific albedo,

H2O, to all faces of BODY 10. Then, it overrides the boundary condition of the bottom face of the cuboid with another material-specific albedo, *CONC24*. Finally, material-specific albedo *H2O* is also applied to the spherical face of BODY 20.

KENO-VI:

```
read geometry
   ...
   global unit 1
      CUBOID 10 ...
      SPHERE 20 ...
      ...
      BOUNDARY 10 20
end geometry

READ BOUNDS
   BODY=10 ALL=H2O -ZB=CONC24
   BODY=20 ALL=H2O
END BOUNDS

' following albedo specification is identical with the above one
' READ BOUNDS
'    BODY=10 ALL=H2O
'    BODY=20 ALL=H2O
'    BODY=10 -ZB=CONC24
' END BOUNDS
```

All available **FACE CODE**s are described in Table 8.1.20. Note that the listed **FACE CODE**s are supported by both KENO V.a and KENO-VI.

Table 8.1.20: Face codes and surface numbers for entering boundary (albedo) conditions.

Face codes	Faces defined by face codes
ALL=	All faces of a single or multiple boundary shape(s)
SURFACE(k)=	Albedo surface enumeration indicates any k^{th} face of the boundary shape (Table 8.1.21 and Table 8.1.22 list shape-specific albedo surface numbers)
+XB=	Positive X face of a cuboidal boundary shape
&XB=	Positive X face of a cuboidal boundary shape
-XB=	Negative X face of a cuboidal boundary shape
+YB=	Positive Y face of a cuboidal boundary shape
&YB=	Positive Y face of a cuboidal boundary shape
-YB=	Negative Y face of a cuboidal boundary shape
+ZB=	Positive Z face of a cuboidal boundary shape
&ZB=	Positive Z face of a cuboidal boundary shape
-ZB=	Negative Z face of a cuboidal boundary shape
XFC=	Both positive and negative X faces of a cuboidal boundary shape
YFC=	Both positive and negative Y faces of a cuboidal boundary shape
ZFC=	Both positive and negative Z faces of a cuboidal boundary shape
+FC=	Positive X, Y, and Z faces faces of a cuboidal boundary shape
&FC=	Positive X, Y, and Z faces of a cuboidal boundary shape
-FC=	Negative X, Y, and Z faces of a cuboidal boundary shape

continues on next page

Table 8.1.20 – continued from previous page

XYF=	Positive and negative X and Y faces of a cuboidal boundary shape
XZF=	Positive and negative X and Z faces of a cuboidal boundary shape
YZF=	Positive and negative Y and Z faces of a cuboidal boundary shape
+XY=	Positive X and Y faces of a cuboidal boundary shape
+YX=	Positive X and Y faces of a cuboidal boundary shape
&XY=	Positive X and Y faces of a cuboidal boundary shape
&YZ=	Positive X and Y faces of a cuboidal boundary shape
+XZ=	Positive X and Z faces of a cuboidal boundary shape
+ZX=	Positive X and Z faces of a cuboidal boundary shape
&XZ=	Positive X and Z faces of a cuboidal boundary shape
&ZX=	Positive X and Z faces of a cuboidal boundary shape
+YZ=	Positive Y and Z faces of a cuboidal boundary shape
+ZY=	Positive Y and Z faces of a cuboidal boundary shape
&YZ=	Positive Y and Z faces of a cuboidal boundary shape
&ZY=	Positive Y and Z faces of a cuboidal boundary shape
-XY=	Negative X and Y faces of a cuboidal boundary shape
-XZ=	Negative X and Z faces of a cuboidal boundary shape
-YZ=	Negative Y and Z faces of a cuboidal boundary shape
YXF=	Positive and negative X and Y faces of a cuboidal boundary shape
ZXF=	Positive and negative X and Z faces of a cuboidal boundary shape
ZYF=	Positive and negative Y and Z faces of a cuboidal boundary shape
-YX=	Negative X and Y faces of a cuboidal boundary shape
-ZX=	Negative X and Z faces of a cuboidal boundary shape
-ZY=	Negative Y and Z faces of a cuboidal boundary shape

Warning: In SCALE 6.3:

- Face codes, which are specific to the cuboidal boundary shape, cannot be used with non-cuboidal boundary shapes.
- **SURFACE(..)** face code **cannot be used** together with any cuboidal face codes listed in Table 8.1.20 if the outermost geometry is cube or cuboid.
- The **ALL** face code refers to all faces of the global boundaries.

Unlike the albedo boundary conditions capability in KENO codes of previous SCALE versions, *FACE CODE* definitions and their use are slightly different in SCALE 6.3.

SURFACE(..)= face code is a valid option for both KENO V.a and KENO-VI. **SURFACE(..)=** face code with a corresponding albedo surface number can be used to apply an albedo boundary condition to the specified surface of **any boundary shape** (cuboidal or non-cuboidal). Albedo surface enumerations for all geometric shapes supported by KENO V.a and KENO-VI are listed in Table 8.1.21 and Table 8.1.22, respectively. Note that entering an illegal albedo surface number with **SURFACE(..)** face code in bounds data input block fails the execution (i.e., SURFACE(2)=... for a sphere, SURFACE(9) for a hexprism, etc.)

The **ALL=** face code will apply the listed boundary condition to all surfaces of a single boundary shape, or multiple boundary shapes (KENO-VI only). It can be used with any boundary shape available in both KENO

V.a and KENO-VI. If it follows a **BODY=** definition, then it refers to all the faces of the body given with that **BODY=** definition (KENO-VI only).

All the face codes, which are listed in Table 8.1.20, except **ALL=** and **SURFACE(..)**, are specific to the cuboidal boundary shape, and they may be applied only to the cuboid or cube boundaries.

Table 8.1.21: Albedo surface numbers related to KENO V.a geometry shapes

GEOMETRY SHAPE	ALBEDO SURFACE ENUMERATION					
	1	2	3	4	5	6
CUBE	+X	-X				
CUBOID	+X	-X	+Y	-Y	+Z	-Z
CYLINDER	Radial	+Z	-Z			
HEMISPHERE	Radial	Cut surface				
HEMICYLINDER	Radial	Top	Bottom	Cut surface		
SPHERE	Radial					
XCYLINDER	Radial	+X	-X			
YCYLINDER	Radial	+Y	-Y			
ZCYLINDER	Radial	+Z	-Z			

Table 8.1.22: Albedo surface numbers related to KENO-VI geometry bodies.

GEOMETRY BODY	ALBEDO SURFACE ENUMERATION											
	1	2	3	4	5	6	7	8	9	10	11	12
CONE	Radial	+Z	-Z									
CUBOID	+X	-X	+Y	-Y	+Z	-Z						
CYLINDER	Radial	+Z	-Z									
DODECAHEDRON	+X	-X	+Y	-Y	+X +Y +Z	-X -Y -Z	-X +Y +Z	+X -Y -Z	-X -Y +Z	+X +Y -Z	+X -Y +Z	-X +Y -Z
ECYLINDER	Radial	+Z	-Z									
ELLIPSOID	Radial											
HEXPRISM	+X	-X	+X +Y	-X -Y	-X +Y	+X -Y	+Z	-Z				
HOPPER	+X	-X	+Y	-Y	+Z	-Z						
PENTAGON	-Y	+X -Y	+X +Y	-X +Y	-X -Y	+Z	-Z					
PLANE	Surface											
QUADRATIC	Surface											
RHEXPRISM	+Y	-Y	-X +Y	+X -Y	+X +Y	-X -Y	+Z	-Z				

continues on next page

Table 8.1.22 – continued from previous page

RING	Inner Radius	Outer Radius	+Z	-Z								
SPHERE	Radial											
WEDGE	-Y	-X +Y	+X +Y	+Z	-Z							
XCYLINDER	Radial	+X	-X									
XPPLANE	+X	-X										
YCYLINDER	Radial	+Y	-Y									
YPPLANE	+Y	-Y										
ZCYLINDER	Radial	+Z	-Z									
ZPPLANE	+Z	-Z										
Surfaces refer to the pre-rotation surface of the body that occurs in the indicated quadrant. Refer to Fig. 8.1.1 through Fig. 8.1.28 for illustrations of each geometry body.												

All albedo names available in both KENO codes are given in Table 8.1.23. Table 8.1.23 lists also some material-specific albedo sets. Care must be exercised when using material-specific albedo types. These data sets were generated using a real problem, and they implicitly reflect the neutron energy spectrum, materials, and geometry from that model. Where neutron energy spectra, materials, and geometry vary from that model, the material-specific albedos may give significantly incorrect results. This may be checked by comparing results from a sample of calculations performed with both explicitly modeled reflectors and material-specific albedos. In general, use of material-specific albedos is not recommended.

Warning: The user should thoroughly understand material-specific albedos (e.g., DP0H2O, CON24, etc.) before attempting to use these reflectors. Misapplication of these problem-specific albedo data can cause the code to produce incorrect results without obvious symptoms.

Table 8.1.23: Albedo names available on the KENO albedo library for use with the face codes *

DP0H2O, DPOH2O, DP0, DPO	12 in. (30.48 cm) double P_0 water differential albedo with 4 incident angles
H2O, WATER	12 in. (30.48 cm) water differential albedo with 4 incident angles
PARAFFIN, PARA, WAX	12 in. (30.48 cm) paraffin differential albedo with 4 incident angles
CARBON, GRAPHITE, C	78.74 in. (200.00 cm) carbon differential albedo with 4 incident angles
ETHYLENE, POLY, CH2	12 in. (30.48 cm) polyethylene differential albedo with 4 incident angles
CONC-4, CON4, CONC4	4 in. (10.16 cm) concrete differential albedo with 4 incident angles
CONC-8, CON8, CONC8	8 in. (20.32 cm) concrete differential albedo with 4 incident angles
CONC-12, CON12, CONC12	12 in. (30.48 cm) concrete differential albedo with 4 incident angles
CONC-16, CON16, CONC16	16 in. (40.64 cm) concrete differential albedo with 4 incident angles
CONC-24, CON24, CONC24	24 in. (60.96 cm) concrete differential albedo with 4 incident angles
VACUUM, VOID	Vacuum condition
SPECULAR, MIRROR, REFLECT	Mirror image reflection
PERIODIC	Periodic boundary condition
WHITE	White boundary condition
* Material-specific albedos (differential albedos) may not be used in continuous energy mode	

Note: Differential albedos (material specific albedos), e.g., H2O and CONC, may not be used in continuous energy calculations. In multigroup mode, none of the differential albedos are allowed for the adjoint transport mode.

Different albedo options are allowed for different global boundaries. If the global boundary is a single cube or

cuboids, then any available albedo option is allowed on any face of this cuboidal global boundary. However, if a periodic boundary condition is to be used, then it must be specified on opposing faces simultaneously.

> **Warning:** There is no consistency check for the surfaces on which **PERIODIC** boundary condition has been specified. It is the user's responsibility to specify the **PERIODIC** boundary condition on opposing faces simultaneously.

Albedo options for non-cuboidal global boundaries may show some differences in KENO V.a and KENO-VI:

KENO V.a:

- Albedo boundary conditions other than **VACUUM** are not allowed if the outermost geometry is either a hemisphere or a hemicylinder.
- Material-specific albedos cannot be applied to the curvilinear faces of the outermost geometry.
- **MIRROR** and **PERIODIC** boundary conditions are not allowed on the curvilinear faces of the outermost geometry.
- Whereas **WHITE** or **VACUUM** may be applied to the radial face of a cylinder, any albedo options can be used for the top and bottom faces.

Example: Use a 24-in. concrete albedo boundary condition on the -Z face of the problem and use mirror image reflection on the +X and -X faces to represent an infinite linear array on a 2 ft thick concrete pad.

```
READ BOUNDS
   XFC=mirror -ZB=CONC24
END BOUNDS
```

Example: A cylinder with arbitrarily chosen length is specularly reflected on the top and bottom faces to create an infinitely long cylinder.

```
READ BOUNDS
   SURFACE(2)=mirror SURFACE(3)=mirror
END BOUNDS
```

KENO-VI:

- Any CHORDed surfaces that are global unit boundaries will use the default (VOID) boundary condition, and this cannot be changed. Because KENO-VI does not provide any input method to assign a non-vacuum albedo boundary condition to the CHORDed surface(s), this restriction may need to be considered when building the geometry of the global unit.
- If the boundary definition vector of the global unit contains only a single body label, then all albedo boundary conditions are allowed on any face of the global boundary.
- If the boundary definition vector of the global unit contains multiple body labels and all these body labels are positive numbers (e.g. **BOUNDARY 10 20 30**), then all albedo boundary conditions may be applied to any of the faces of each body.
- If the boundary definition vector of the global unit contains multiple body labels and any of these body labels are negative numbers (e.g. **BOUNDARY 10 20 -30**), then only the **VACUUM** albedo is allowed on the faces of all bodies.

Caution: No non-vacuum albedo boundary conditions can be applied to a CHORDed surface that is global unit boundary.

Example: A cylinder with an arbitrarily chosen length is specularly reflected on the top and bottom faces to create an infinitely long cylinder.

```
read geometry
   ...
   global unit 1
      CYLINDER 10 ...
      ...
      BOUNDARY 10
end geometry

READ BOUNDS
   ALL=mirror SURFACE(1)=void
END BOUNDS

' identical boundary condition can be applied with another definition
' READ BOUNDS
'    BODY=10 SURFACE(2)=reflective SURFACE(3)=mirror
' END BOUNDS
```

Example: Use a 24 in. concrete albedo boundary condition on the -Z face of the problem and use mirror image reflection on the +X and -X faces to represent an infinite linear array on a 2 ft thick concrete pad.

```
read geometry
   ...
   global unit 1
      CUBOID 101 ...
      ...
      BOUNDARY 101
end geometry

READ BOUNDS
   SURFACE(1)=mirror SURFACE(2)=mirror SURFACE(6)=CON24
END BOUNDS

' identical boundary condition can be applied with another definition
' READ BOUNDS
'    BODY=101 XFC=reflective -ZB=CONC24
' END BOUNDS
```

Example: Use a 24-in. concrete albedo boundary condition on the -Z face of a problem with a hexagonal boundary and use mirror image reflection on all side faces of the hexprism to represent an infinite planar array on a 2-ft-thick concrete pad.

```
read geometry
   ...
   global unit 1
      HEXPRISM 201 ...
      ...
      BOUNDARY 201
end geometry

READ BOUNDS
   SURFACE(1)=mirror SURFACE(2)=mirror
   SURFACE(3)=mirror SURFACE(4)=mirror
   SURFACE(5)=mirror SURFACE(6)=mirror
   SURFACE(7)=vacuum SURFACE(8)=conc24
```

(continues on next page)

(continued from previous page)

```
END BOUNDS

' an alternative albedo specification for the same boundary conditions
' READ BOUNDS
'    ALL=mirror SURFACE(7)=vacuum SURFACE(8)=con24
' END BOUNDS
'

' an alternative albedo specification for the same boundary conditions
' READ BOUNDS
'    BODY=201 ALL=mirror SURFACE(7)=vacuum SURFACE(8)=con24
' END BOUNDS
'

' an alternative albedo specification for the same boundary conditions
' READ BOUNDS
'    ALL=mirror
'    BODY=201 SURFACE(7)=vacuum SURFACE(8)=con24
' END BOUNDS
'
```

Example: The outer boundary of the global unit consists of a cuboid (body label 10) and a sphere (body label 20). The sphere is large enough to cut the corners of the cuboid leaving most of the cuboid intact. Use a 24 in. concrete albedo boundary condition on the -Z face of the cuboid to represent a 2 ft. thick concrete pad. Use the DP0H2O on the other surfaces to represent an infinite water reflector.

```
read geometry
   ...
   global unit 1
      CUBOID 10 ...
      SPHERE 20 ...
      ...
      BOUNDARY 10 20
end geometry

READ BOUNDS
   BODY=10 ALL=DP0H2O -ZB=CONC24
   BODY=20 SURFACE(1)=DPOH2O
END BOUNDS

' an alternative albedo specification for the same boundary conditions
' READ BOUNDS
'    ALL=DPOH2O
'    BODY=10 -ZB=CONC24
' END BOUNDS
```

Note that the **ALL** face code is used in two different ways in the above example: (1) Used with body 10 to apply **DPOH2O** albedo *to all faces of this body* (2) Used without body, which is legitimate for KENO-VI, to apply **DPOH2O** albedo *to all faces of both body 10 and 20* (global unit boundaries defined by CUBOID 10 and SPHERE 20).

Example: KENO geometry specification allows modeling the same physical problem with different geometry definitions. For some cases, boundary condition specification becomes more critical to represent the same problem with different geometry configurations. Users must be extra cautious when building their model and examine the limitations listed in this section carefully to prevent unexpected results.

This example was designed to demonstrate how ignoring such limitations for albedo boundary condition setup affects results in analysis. In this example, a model of a simplified version of a GODIVA sphere was attempted with three different geometry definitions as shown below.

KENO-VI, full GODIVA sphere:

```
...

READ GEOMETRY
   global unit 1
          sphere 30 8.75
      media 1 1 30
      boundary 30
END GEOMETRY

READ BOUNDS
   ALL=vacuum
END BOUNDS
```

KENO-VI, half GODIVA sphere defined by combination of a sphere and a cuboid:

```
...

READ GEOMETRY
   global unit 1
          sphere 30 8.75
      cuboid 40 8.75 0.0 4p8.75
      media 1 1 30 40
      boundary 30 40
END GEOMETRY

READ BOUNDS
   ALL=mirror
       BODY=30 surface(1)=void
END BOUNDS
```

KENO-VI, half GODIVA sphere defined by a sphere and a CHORD modifier:

```
...

READ GEOMETRY
   global unit 1
          sphere 30 8.75 chord +x=0.0
          media 1 1 30
          boundary 30
END GEOMETRY

READ BOUNDS
   ALL=mirror surface(1)=void
END BOUNDS
```

The first geometry definition uses only a sphere to define the Godiva sphere. The second sample input uses the combination of a sphere and a cuboid to define a hemisphere and an appropriate boundary condition on each outer boundary to simulate the Godiva sphere. Similarly, in the third sample input, a modified CHORD was used with a sphere to define a hemisphere and an appropriate boundary condition on each outer boundary to simulate the Godiva sphere.

But even if the sample input with the CHORDed surface given above seems correct at first glance, it cannot model the same physical problem as the other two because the albedo boundary condition specification with ALL=MIRROR is not applied to the CHORDed surface. Therefore, that CHORDed sphere models a half GODIVA sphere rather than a full one. Results from these three models, presented in Table 8.1.24, indicate that using code capabilities improperly might result in significant bias in the results.

Table 8.1.24: k-eff results of GODIVA models used in this example

Model	k-eff
Full sphere	1.02680 +/- 0.00100
Half sphere defined by a sphere and a cuboid	1.02962 +/- 0.00075
Half sphere defined by a sphere and a CHORD	0.78745 +/- 0.00078

Unlike KENO codes in previous versions, in SCALE 6.3, KENO codes print several warning messages to remind the user about such limitations and their potential effects if ignored.

Caution: There is no input method to assign a non-vacuum boundary condition to a CHORDed boundary surface (external boundary). Therefore, KENO-VI always applies VACUUM boundary condition to the CHORDed boundary surfaces, and this cannot be changed.

8.1.3.7 Biasing or weighting data

The biasing data block is used (in only multigroup mode) to define the weight that is given to a neutron surviving Russian roulette. The average weight of a neutron that survives Russian roulette, *wtavg*, is defaulted to *dwtav* (**WTA**= in the parameter data [see Sect. 8.1.3.3]) for all *BIAS IDs* and can be overridden by entering biasing information.

The *biasing_information* is used to relate a *BIAS ID* to the desired energy-dependent values of *wtavg*. This concept is similar to the way the *MIXTURE ID, mat*, is related to the macroscopic cross section data.

The weighting functions used in KENO are energy-dependent values of *wtavg* that are applicable over a given thickness interval of a material. For example, the weighting function for water is composed of sets of energy-dependent values of *wtavg* for 11 intervals, each interval being 3 cm thick. The first set of *wtavg's* is for the 0–3 cm interval of water, the second set of *wtavg's* is for the 3–6 cm interval of water, etc. The eleventh set of *wtavg's* is for the 30–33 cm interval of water.

To input biasing information, a *BIAS ID* must be assigned to correspond to a set of *wtavg*. Biasing data can specify a *MATERIAL ID* from the existing KENO V.a weighting library or from the *AUXILIARY DATA* input. The materials available from the KENO weighting library are listed in Table 8.1.25.

The *biasing_information* is entered in one of the following two forms. The first set is said to input the *CORRELATION DATA*, while the second form is said to input the *AUXILIARY DATA*.

`READ BIAS ID**\ =\ *m ib ie* ``END BIAS`

or

`READ BIAS WT[S]=`*wttitl id s* t_1 i_1 g_1 $w_{1,1}$ ... $w_{1,i1xg1}$ ... t_s i_s g_s $w_{s,1}$ ... $w_{s,isxgs}$ `END BIAS`

`ID=` specifies that *CORRELATION DATA* will be entered next.

`WT=` or `WTS=` specifies that *AUXILIARY DATA* will be entered next.

m is the identification (material ID) for the material whose weighting function is to be used. A material ID can be chosen from the existing KENO weighting library (Table 8.1.25) or from the *auxiliary data* input using the second form of the **BIAS** block as described later. If a material ID appears in both the KENO weighting library and the *auxiliary data*, the *weights* from the *auxiliary data* will be used.

ib is the bias ID of the weighting function for the first interval of material *m*. The geometry record having the bias ID equal to *ib* will use the group-dependent *weights* from the first interval of material *m*.

ie is the bias ID of the group-dependent *weights* from the (*ie* - *ib* + 1)*th* interval of material *m*.

wttitl is an arbitrary title name (12 characters maximum), such as CONCRETE, WATER, SPECIALH2O, etc., to identify the material for which the user is entering data. Embedded blanks are not allowed.

id is an identification number (material ID). The value is arbitrary. However, if the data are to be utilized in the problem, this ID must also be used at least once in the first form of the **BIAS** block.

s is the number of sets of group structures for which *weights* will be read for this ID.

$t_1 \ldots t_s$ are *s* thicknesses of each increment for which *weights* will be read for this ID.

$i_1 \ldots i_s$ are *s* numbers of increments for which *weights* will be read for this ID.

$g_1 \ldots g_s$ are *s* numbers of energy groups for which *weights* will be read.

$w_{1,i1xg1} \ldots w_{s,isxgs}$ are *s* sets of *weights*, each set containing a number of *weights* equal to the product of number of increments times the number of groups for that set. The group index varies the fastest.

Table 8.1.25: IDs, group structure and incremental thickness for weighting data available on the KENO weighting library.

Material	Material ID	Group structure for which weights are available	Increment [a] thickness (cm)	Total number of increments available
Concrete	301	27	5	20
		28	5	20
		56	5	20
		200	5	20
		238	5	20
		252	5	20
Paraffin	400	27	3	10
		28	3	10
		56	3	10
		200	3	10
		238	3	10
		252	3	10
Water	500	27	3	10
		28	3	10
		56	3	10
		200	3	10
		238	3	10
		252	3	10
Graphite	6100	27	20	10
		28	20	10
		56	20	10
		200	20	10
		238	20	10
		252	20	10

continues on next page

Table 8.1.25 – continued from previous page

[a] Group-dependent weight averages are supplied for each increment of the specified incremental thickness (i.e., for any given material) the first ngp (number of energy groups) weights apply to the first increment of the thickness specified here, the next ngp weights apply to the next increment of that thickness, etc CAUTION–If bias IDs defined in the weighting information data are used in the geometry, the region thickness should be consistent with the incremental thickness of the weighting data in order to avoiid overbiasing or underbiasing.

Warning: The user should thoroughly understand weighted tracking before attempting to generate and use auxiliary data for biasing. Incorrect weighting can cause the code to produce incorrect results without obvious symptoms.

Caution: 1. Each set of *AUXILIARY* or *CORRELATION* data must be completely described in conjunction with its keyword. Complete sets of these data can be interspersed in an arbitrary order but data within each set must be entered in the specified order.

2. *AUXILIARY DATA*: If the same *m* is specified in more than one set of data, the last set having the group structure used in the problem is the set that will be utilized. When *AUXILIARY DATA* are entered, *CORRELATION DATA* must also be entered in order to use the *AUXILIARY DATA*.

3. *CORRELATION DATA*: If biasing data define the same bias ID (from the geometry data) more than once, the value that is entered last supersedes previous entries. *Be well aware that multiple definitions for the same bias ID can cause erroneous answers due to overbiasing.*

4. Bias data may not be used in continuous energy mode.

Examples

1. Use the first form of the **BIAS** block to utilize the water biasing factors in bias IDs 2 through 11. From Table 8.1.25, water has material ID m=500 and has bias parameters for 10 intervals that are each 3 cm thick.

```
READ BIAS ID=500 2 11 END BIAS
```

2. Use the second form of the `BIAS` block to specify biasing factors for `SPECIALWATER` to be used in bias IDs 6 and 7. The `SPECIALWATER` biasing factors have a value of 0.69 for BIAS ID 6 and 0.86 for bias ID 7 in each energy group. Sixteen-group cross sections are being used. Each weighting region is 3.048 cm thick. The material ID is arbitrarily chosen to be 510. Note that the first form of the `BIAS` block must be entered to allow the second form of the `BIAS` block to be used for BIAS IDs 6 and 7.

```
READ BIAS WT = SPECIALWATER 510 1 3.048 2 16 16*0.69 16*0.86 ID=510 6 7 END BIAS
```

3. An example of multiple definitions for the same bias ID follows:

```
READ BIAS ID=400 2 7 ID=500 5 7 END BIAS
```

The data for paraffin (`ID`=400) will be used for bias IDs 2, 3, and 4, and the data for water (`ID`=500) will be used for bias IDs 5, 6, and 7. The paraffin data for bias IDs 5, 6, and 7 have been overwritten by water data.

Multiple definitions for the same bias ID are not necessarily incorrect, but the user should be cautious about using multiple definitions and should ensure that the desired biasing or weighting functions are used in the desired geometry regions.

4. An example of how the *bias ID* relates to the energy-dependent values of *weights* is given below.

Assume that a paraffin reflector is to be used, and it is desirable to use the weighting function from the KENO weighting library to minimize the running time for the problem. Also assume that these weighting functions are to be used in the volumes defined in the geometry records having the *bias ID* (defined on a **shape** or `MEDIA` card for KENO V.a and KENO-VI, respectively) equal to 6, 7, 8, and 9. *Correlation data* are then entered and *auxiliary data* will not be entered.

The biasing data would be:

```
READ BIAS ID=400 6 9 END BIAS
```

The results of these data are

(1) the group-dependent *weights* for the 0–3 cm interval of paraffin will be used in the volume defined by the geometry region having *bias ID*= 6.

(2) the group-dependent *weights* for the 3–6 cm interval of paraffin will be used in the volume defined by the geometry region having *bias ID*= 7.

(3) the group-dependent *weights* for the 6–9 cm interval of paraffin will be used in the volume defined by the geometry region having *bias ID*= 8.

(4) the group-dependent *weights* for the 9–12 cm interval of paraffin will be used in the volume defined by the geometry region having *bias ID*= 9.

8.1.3.8 Start data

Special start options are available for controlling the initial neutron distribution. The default starting distribution for **a global array** is flat over the overall array dimensions, in fissile material only. The default starting distribution for a single unit is flat over the system, in fissile material only. See Table 8.1.26 for the starting distributions available in KENO. The syntax for the **START** block is:

`READ START` *p1 ...pN* `END START`

$p_1 \ldots p_N$ are N initializations for the parameters listed below.

The starting information that can be entered is given below. Enter only the data necessary to describe the desired starting distribution.

NST = ***ntypst*** start type, default = 0 Table 8.1.26 lists the available options under the heading, "Start type."

TFX = ***tfx*** the X coordinate of the point at which neutrons are to be started. Default = 0.0. Use for start types 3, 4, and 6.

TFY = ***tfy*** the Y coordinate of the point at which neutrons are to be started. Default = 0.0. Use for start types 3, 4, and 6.

TFZ = ***tfz*** the Z coordinate of the point at which neutrons are to be started. Default = 0.0. Use for start types 3, 4, and 6.

NXS = ***nbxs*** the x index of the unit's position in the global array. Default = 0. Use for start types 2, 3, and 6.

NYS = ***nbys*** the y index of the unit's position in the global array. Default = 0. Use for start types 2, 3, and 6.

NZS = ***nbzs*** the z index of the unit's position in the global array. Default = 0. Use for start types 2, 3, and 6.

KFS = ***kfis*** the mixture whose fission spectrum is to be used for starting neutrons that are not in a fissionable medium. Defaulted to the fissionable mixture having the smallest mixture number. Available for start types 3, 4, and 6.

LNU = ***lfin*** the final neutron to be started at a point. Default = 0. Each *lfin* should be greater than zero, and each successive *lfin* should be greater than the previous one. Use only for start type 6.

NBX = ***nboxst*** the unit in which neutrons will be started. Default = 0. Use for start types 4 and 5.

FCT = ***fract*** the fraction of neutrons that will be started as a spike or the relative fraction of each segment when using to specify a segmented distribution in z. Default = 0. Use for start type 2 and type 8.

XSM = ***xsm*** the -X dimension of the cuboid in which the neutrons will be started. Default = 0.0. For an array problem, XSM is defaulted to the minimum X coordinate of the global array. If the outermost geometry is a cube or cuboid, then XSM is defaulted to the minimum X coordinate of this cuboid. Use for start types 0, 1, 2, 7, and 8.

XSP = ***xsp*** the +X dimension of the cuboid in which the neutrons will be started. Default = 0.0. For an array problem, XSP is defaulted to the maximum X coordinate of the global array. If the outermost geometry is a cube or cuboid, then XSP is defaulted to the maximum X coordinate of this cuboid. Use for start types 0, 1, 2, 7, and 8.

YSM = ***ysm*** the -Y dimension of the cuboid in which the neutrons will be started. Default = 0.0. For an array problem, YSM is defaulted to the minimum Y coordinate of the global array. If the outermost geometry is a cube or cuboid, then YSM is defaulted to the minimum Y coordinate of this cuboid. Use for start types 0, 1, 2, 7, and 8.

YSP = ***ysp*** the +Y dimension of the cuboid in which the neutrons will be started. Default = 0.0. For an array problem, YSP is defaulted to the maximum Y coordinate of the global array. If the outermost geometry is a cube or cuboid, then YSP is defaulted to the maximum Y coordinate of this cuboid. Use for start types 0, 1, 2, 7, and 8.

ZSM = ***zsm*** the -Z dimension of the cuboid in which the neutrons will be started. Default = 0.0 for only start types 0, 1, 2, and 7. For an array problem, ZSM is defaulted to the minimum Z coordinate of the global array. If the outermost geometry is a cube or cuboid, then ZSM is defaulted to the minimum Z coordinate of this cuboid. Use for start types 0, 1, 2, 7, and 8.

ZSP = ***zsp*** the +Z dimension of the cuboid in which the neutrons will be started. Default = 0.0 for only start types 0, 1, 2, and 7. For an array problem, ZSP is defaulted to the maximum Z coordinate of the global array. If the outermost geometry is a cube or cuboid, then ZSP is defaulted to the minimum Z coordinate of this cuboid. Use for start types 0, 1, 2, 7, and 8.

RFL = ***rflkey*** No longer supported (obsolete parameter).

PS6 = ***lprt6*** the key for printing start type 6 input data. If the key is YES, then start type 6 data are printed. If it is NO, then start type 6 data are not printed. Enter YES or NO. Default = NO. Available for start type 6.

PSP = ***lpstp*** the key for printing the neutron starting points using the tracking format. If the key is YES, then print the neutron starting points. If it is NO, then do not print the starting points. Enter YES or NO. Default = NO. Available for all start types.

RDU = ***rdu*** the file from which ASCII start data are to be read for start type 6.

WS6 = ***ws6*** the file to which ASCII start data are written. Available for all start types.

MSS = ***filename.msl*** the file from which ASCII start data are to be read. *filename* may include a valid pathname. Available for start type 9.

Note: All start types can write the initial neutron starting points at the last generation to an ASCII start data file specified by WS6. However, only start type 6 can read starting data from an ASCII start data file specified by RDU. The ASCII start data file format is described in Sect. 8.1.3.8.1.

Table 8.1.26: Starting distributions available in KENO.

Start type	Required data	Optional data	Starting distribution
0	None	NST XSM XSP YSM YSP ZSM ZSP PSP WS6 RFL [a]	Uniform throughout fissile material within the volume defined by (1) the outer region of a single unit, (2) the boundary of the global array, or (3) a cuboid specified by XSM, XSP, YSM, YSP, ZSM, and ZSP.
1	NST	XSM XSP YSM YSP ZSM ZSP PSP WS6 RFL [a]	The starting points are chosen according to a cosine distribution throughout the volume of a cuboid defined by XSM, XSP, YSM, YSP, ZSM, and ZSP. Points that are not in fissile material are discarded.
2	NST NXS NYS NZS FCT	XSM XSP YSM YSP ZSM ZSP PSP WS6 RFL [a]	An arbitrary fraction (FCT) of neutrons are started uniformly in the unit located at position NXS, NYS, NZS in the global array. The remainder of the neutrons is started in fissile material, from points chosen from a cosine distribution throughout the volume of a cuboid defined by XSM, XSP, YSM, YSP, ZSM, ZSP.
3	NST TFX TFY TFZ NXS NYS NZS	KFS PSP WS6	All neutrons are started at position TFX, TFY, TFZ within the unit located at position NXS, NYS, NZS in the global array.

continues on next page

Table 8.1.26 – continued from previous page

4	NST TFX TFY TFZ NBX	KFS PSP WS6	All neutrons are started at position TFX, TFY, TFZ within units NBX in the global array.
5	NST NBX	PSP WS6	Neutrons are started uniformly in fissile material in units NBX in the global array.
6	NST TFX TFY TFZ LNU [b]	NXS NYS NZS KFS PS6 PSP RDU [c] WS6	The starting distribution is arbitrarily input. LNU is the final neutron to be started at a point TFX, TFY, TFZ relative to the global coordinate system or at a point TFX, TFY, TFZ, relative to the unit located at the global array position NXS, NYS, NZS.
7		XSM XSP YSM YSP ZSM ZSP PSP WS6	The starting points are chosen according to a flat distribution in the X- and Y-dimensions and a $(1.0 - \cos(z))^2$ distribution in the Z-dimension throughout the volume of a cuboid defined by XSM, XSP, YSM, YSP, ZSM, and ZSP. Points that are not in fissile material are discarded.
8	NST ZSM ZSP FCT	XSM XSP YSM YSP PSP WS6	Neutrons are started with flat distribution in X and Y, and a segmented distribution in Z, with the X-Y limits defined by XSM, XSP, YSM, YSP and the relative fraction in ZSP-ZSM defined by FCT. FCT must be the last **entry** for each segment.
9	NST MSS		Mesh source from Sourcerer [d]. The starting distribution is read from a previously created mesh source file declared with MSS=*filename.msl*, where *filename* may include a valid pathname.

continues on next page

Table 8.1.26 – continued from previous page

[a] RFL parameter is no longer supported (obsolete parameter)
[b] When entering data for start 6, LNU must be the last entry for each set of data, and the LNU in each successive set of data must be larger than the previous value of LNU. A set of data consists of required and optional data.
[c] Starting points can be read from either a single or multiple ASCII start data file(s) specified with RDU. Each set of data with RDU must be followed by LNU that controls the number of starting points read from each file. Starting points read from an ASCII start data file can be also combined with any set of starting data specified with TFX/TFY/TFZ and/or NXS/NYS/NZS.
[d] Sourcerer sequence has temporarily been removed from SCALE 6.3 because its design relied on legacy SCALE components. A new development has been in progress to redesign Sourcerer capability as a new START data type for both KENO and Shift transport codes, which will be available in next SCALE release.

Warning: Start data input block does not allow multiple start types defined together. Both KENO V.a and KENO-VI only process the start data entered in the first start data input block and discard the others if multiple start data input block defined in user input.

Example-1:

Both KENO V.a and KENO-VI read only start data from the first start data input block and ignore the second one. All starting points are uniformly sampled throughout the fissile materials iside the user-defined box (+x=3.0 -x=-3.0 +y= 1.0 -y=-1.0 +z= 2.0 -z=-2.0).

```
...

  READ START
     NST=0 XSP=3.0 XSM=-3.0 YSP=1.0 YSM=-1.0 ZSP=2.0 ZSM=-2.0
  END START

  READ START
     NST=6 TFX=2.0 TFY= 3.0 TFZ=0.0  LNU=25
  END START

end data
end
```

Optional parameters, **XSM**, **XSP**, **YSM**, **YSP**, **ZSM**, and **ZSP** allowed in start types 0, 1, 2, 7, and 8 are defaulted to 0.0. The default values are internally updated for the problems either with a global array or with a global unit with cuboid booundary shape as described above. User-defined values always override these defaults, and the last user entry always overrides the previous one. See Sect. 8.1.4.8 for details and more examples.

Note: User input specification with **XSM** > **XSP** or **YSM** > **YSP** or **ZSM** > **ZSP** is **not** considered as an error, and calculation is continued by swapping the values. Start type edit in output does not reflect the swapping operations.

Note: KENO-VI and KENO V.a has different interpretation for the source sampling when the user-defined cuboid is not inside the outermost geometry. In such a case, some starting points could be outside the global

geometry, even though they are sampled inside the user-defined cuboid, and these points are considered as an error by KENO V.a; therefore, code terminates the execution after a couple errors have occurred. In contrast, rather than terminating execution, KENO-VI discards the points sampled outside the global geometry, and continues source sampling until all starting points have been sampled in the fissile regions inside the global unit.

Note: Unlike KENO V.a, the outer boundary can be any shape (or combination of shapes) in KENO-VI. For such a case, KENO-VI samples starting positions in the volume of **the first body** entered in the boundary definition vector of the global unit if and only if none of translation and transformation operations are performed on this body.

The start type 6 capability allows neutrons to be started at the arbitrary starting points defined by a set of data. The last entry for each start 6 data set must be LNU, and the LNU value of each successive set of data must be larger than the last. Start 6 data set with any of TFX, TFY, and TFZ, followed by LNU defines points relative to the global coordinate system, and with any of NXS, NYS, NZS, TFX, TFY, and TFZ followed by LNU defines points relative to the unit located at global array position (NXS, NYS, NZS).

In the start type 6 capabilitiy, TFX, TFY, and TFZ values are defaulted to 0.0, and NXS, NYS, and NZS values are defaulted to 0. This capability does not allow multiple entries of NXS, NYS, NZS, TFX, TFY, and TFZ in the same start 6 data set. NXS, NYS, NZS, TFX, TFY, and TFZ values entered in each start type 6 data set override their default values. A start type 6 data set with one or more missing TFX, TFY, and TFZ entries does not halt the execution. Instead, the last updated values of TFX, TFY, and TFZ are used for the missing one.

Start type 6 is capable of reading starting points from an ASCII start data file, which could be created by writing starting points from a previous calculation, defined by RDU in a start type 6 data set. See Sect. 8.1.3.8.1 for the details about a typical ASCII start data file currently supported, and Sect. 8.1.4.8 for more examples.

The following example was designed to illustrate the starting source distributions provided by start data types. Note that only start types 0, 1, 2, 5, 7, and 8 was used for this specific example.

Example-2:

In this example, a $2 \times 2 \times 1$ rectangular array filled with four different units is used to demonstrate the different starting point distributions that can be generated by the starting types available in the start data capability. Spatial distribution of the starting points sampled with only start types 0, 1, 2, 5, 7, and 8 are overlayed on the problem geometry.

```
=kenovi
2x2 pin cell model for starting distribution demonstration
read parameters
   cep=ce_v7.1_endf gen=10  npg=5000 nsk=0 htm=no
end parameter

read mixt
   mix=1
      8016 3.91376E-02 92235 8.73674E-04 92238 1.87428E-02
   mix=2
      8016 3.91376E-02 92235 8.73674E-04 92238 1.87428E-02
   mix=31
     40090 2.17623E-02 40091 4.74582E-03 40092 7.25409E-03
     40094 7.35137E-03 40096 1.18434E-03
```

(continues on next page)

(continued from previous page)

```
   mix=32
     40090 2.17623E-02 40091 4.74582E-03 40092 7.25409E-03
     40094 7.35137E-03 40096 1.18434E-03
   mix=41
     1001 4.77898E-02   1002 5.49646E-06  5010 4.75767E-06
     5011 1.91502E-05   8016 2.38396E-02  8017 9.08111E-06
   mix=42
     1001 4.77898E-02   1002 5.49646E-06  5010 4.75767E-06
     5011 1.91502E-05   8016 2.38396E-02  8017 9.08111E-06
end mixt

read geom
   unit   1
      com='UO2 Fuel Rod'
      cylinder 10   0.3860 6.0 0.0
      cylinder 20   0.4582 6.0 0.0
      cuboid   30 4p0.6375 6.0 0.0
      media  1 1 10
      media 31 1 20 -10
      media 41 1 30 -20
      boundary 30
   unit   2
      com='UO2 Fuel Rod'
      cylinder 10   0.3860 6.0 0.0
      cylinder 20   0.4582 6.0 0.0
      cuboid   30 4p0.6375 6.0 0.0
      media  2 1 10
      media 32 1 20 -10
      media 42 1 30 -20
      boundary 30
   unit   3
      com='UO2 Fuel Rod'
      cylinder 10   0.3860 6.0 0.0
      cylinder 20   0.4582 6.0 0.0
      cuboid   30 4p0.6375 6.0 0.0
      media  1 1 10
      media 31 1 20 -10
      media 41 1 30 -20
      boundary 30
   unit   4
      com='UO2 Fuel Rod'
      cylinder 10   0.3860 6.0 0.0
      cylinder 20   0.4582 6.0 0.0
      cuboid   30 4p0.6375 6.0 0.0
      media  2 1 10
      media 32 1 20 -10
      media 42 1 30 -20
      boundary 30
   global unit 5
      cuboid    10 4p1.275 6.0 0.0
      array  1  10 place 2 2 1  0.6375 0.6375  0.0
      boundary 10
end geom
read array
   ara=1 typ=square nux=2 nuy=2 nuz=1 gbl=1
      fill  1 3
            2 4
      end fill
end array
read bounds
   all=mirror
end bounds
end data
end
```

- **start type 0** (default starting type), samples neutrons initial starting points uniformly throughout four

fissile cylinders inside the global unit. Fig. 8.1.12 shows spatial distributions of the initial neutrons.

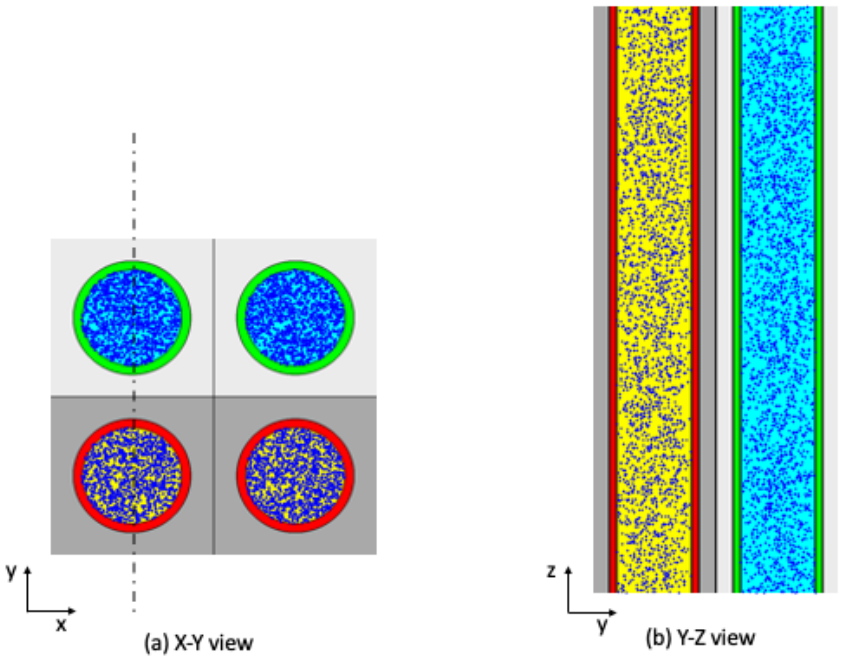


Fig. 8.1.12: Starting point distribution obtained with start type 0 overlayed on the geometry

---

Note: Overlaying starting distribution on Fulcrum-visualized geometry is not currently implemented in Fulcrum. Initial starting points obtained with **PSP** option for all these start type configurations were used with an external utility to overlay them on the geometry visualized by Fulcrum.

- **start type 1**, samples neutrons initial starting points with a cosine distribution (in each dimension) within a user-defined box given in start data input. All points except the ones in four fissile cylinders inside the global unit are discarded. Start type 1 produces initial neutron distributions concentrated in the middle of the problem as depicted in Fig. 8.1.13.

```
...

READ START
   NST=1 XSP=1.275 XSM=-1.275 YSP=1.275 YSM=-1.275 ZSP=6.0 ZSM=0.0
   PSP=YES WS6=keno1.src
END START
```

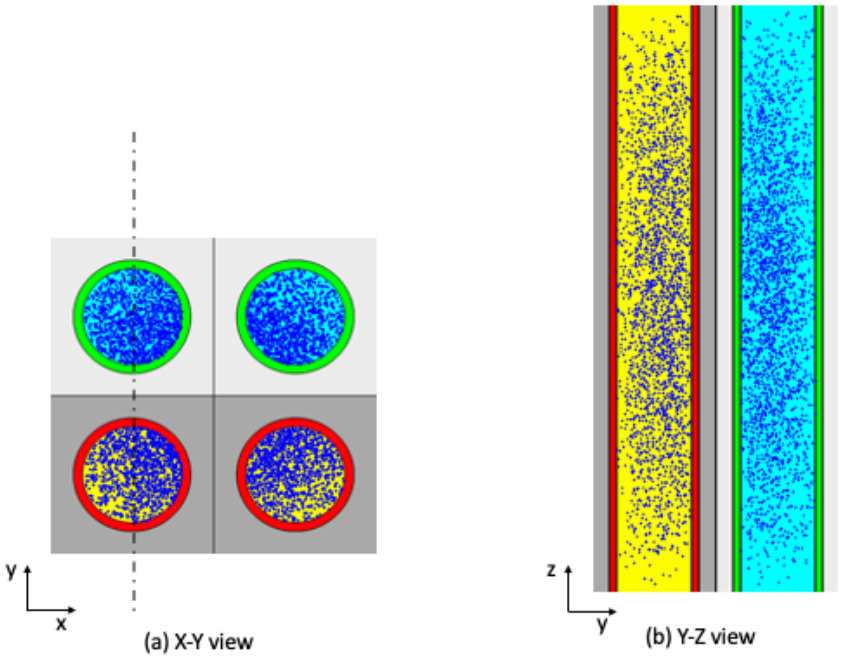


Fig. 8.1.13: Starting point distribution obtained with start type 1 overlayed on the geometry

- **start type 2** is used with the array problems. With the following start data definition, an arbitrary fraction (FCT=0.5 in this case) of neutrons is started uniformly in the unit (unit=1) located at position NXS=1 NYS=1 and NZS=1 in the global array. The remainder of the neutrons are started in fissile material, from points chosen from a cosine distribution throughout the volume of a cuboid defined by +x=1.275 -x=-1.275 +y=1.275 -y=-1.275 +z=6.0 -z=0.0. All points except the ones in four fissile cylinders inside the user-defined cuboid are discarded. Start type 2 produces initial neutron distributions concentrated both in the middle of the problem and in the unit=1 located at the specified global array element. Fig. 8.1.14 shows both vertical and horizontal spatial distributions of the initial neutrons sampled with this start type.

```
...

READ START
   NST=2 XSP=1.275 XSM=-1.275 YSP=1.275 YSM=-1.275 ZSP=6.0 ZSM=0.0
   FCT=0.5 NXS=1 NYS=1 NZS=1
   PSP=YES WS6=keno2.src
END START
```

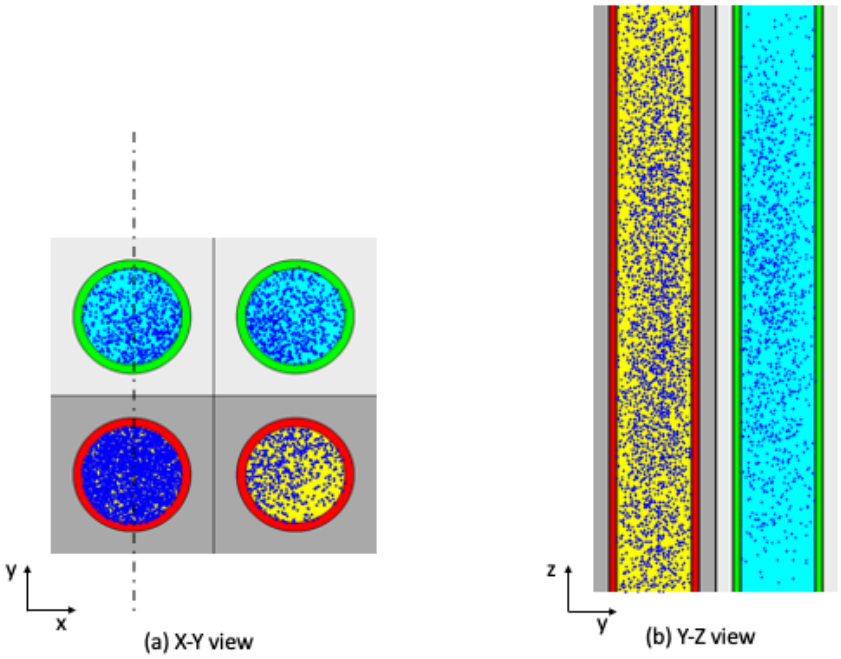


Fig. 8.1.14: Starting point distribution obtained with start type 2 overlayed on the geometry

- **start type 5** uniformly samples initial starting points inside the unit=2 in the global array. Fig. 8.1.15 shows the both vertical and horizontal spatial distributions of the initial neutrons sampled with this start type.

```
...

READ START
   NST=5
   NBX=2
   PSP=YES WS6=keno5.src
END START
```

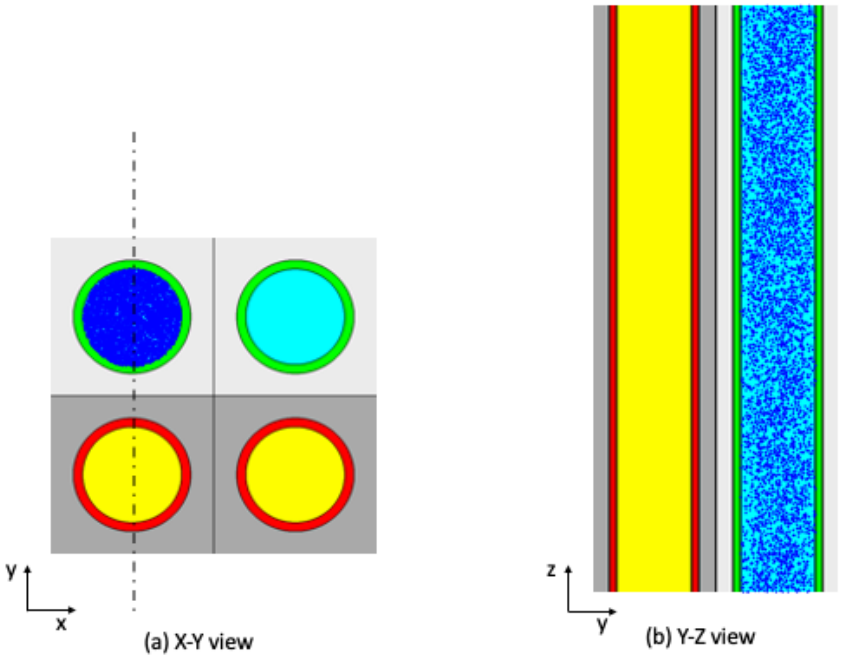


Fig. 8.1.15: Starting point distribution obtained with start type 5 overlayed on the geometry

- **start type 7** samples neutrons initial starting points with a uniform distribution in X-Y and a $(1 - cos)^2$ distribution in Z inside the cuboid defined by +x=1.275 -x=-1.275 +y=1.275 -y=-1.275 +z=6.0 -z=0.0. Start type 7 produces an initial neutron distributions concentrated to the top and bottom parts of the problem as shown in Fig. 8.1.16.

```
...

READ START
   NST=7 XSP=1.275 XSM=-1.275 YSP=1.275 YSM=-1.275  ZSP=6.0 ZSM=0.0
   PSP=YES WS6=keno7.src
END START
```

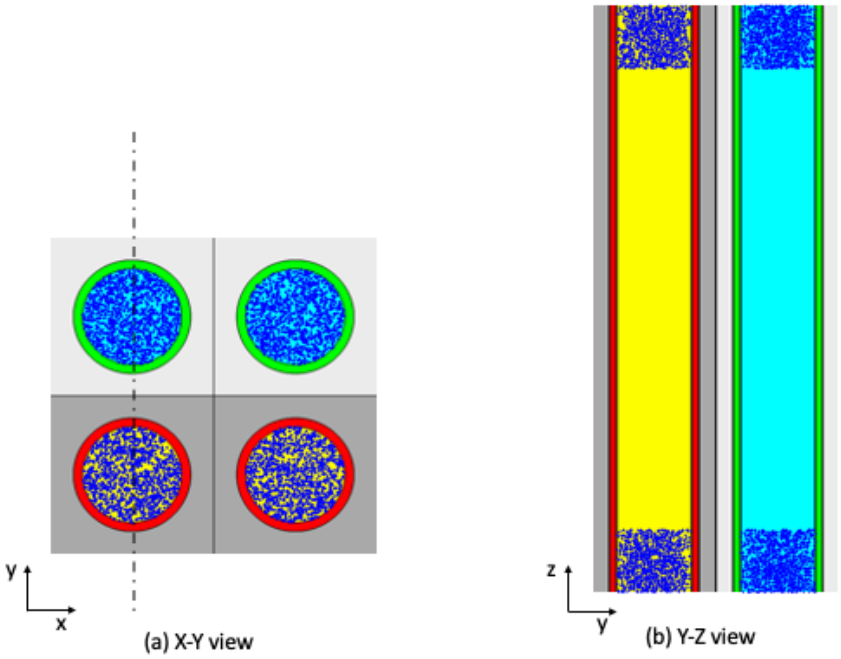


Fig. 8.1.16: Starting point distribution obtained with start type 7 overlayed on the geometry

- **start type 8** samples the neutrons initial starting points with a uniform distribution in X-Y and a segmented distribution in Z inside the cuboid defined by +x=1.275 -x=-1.275 +y=1.275 -y=-1.275 +z=6.0 -z=0.0. Start type 8 with the specified input places the majority of the neutrons in the bottom of the problem and gradually decreases the neutron population from bottom to top using the user-provided segmented distribution as shown in Fig. 8.1.17.

```
...

READ START
   NST=8
   PSP=YES WS6=keno8.src
   XSP=1.275 XSM=-1.275 YSP=1.275 YSM=-1.275
   ZSM=0.0 ZSP=1.0 FCT=0.30
   ZSM=1.0 ZSP=2.0 FCT=0.25
   ZSM=2.0 ZSP=3.0 FCT=0.20
   ZSM=3.0 ZSP=4.0 FCT=0.13
   ZSM=4.0 ZSP=5.0 FCT=0.10
   ZSM=5.0 ZSP=6.0 FCT=0.02
END START
```

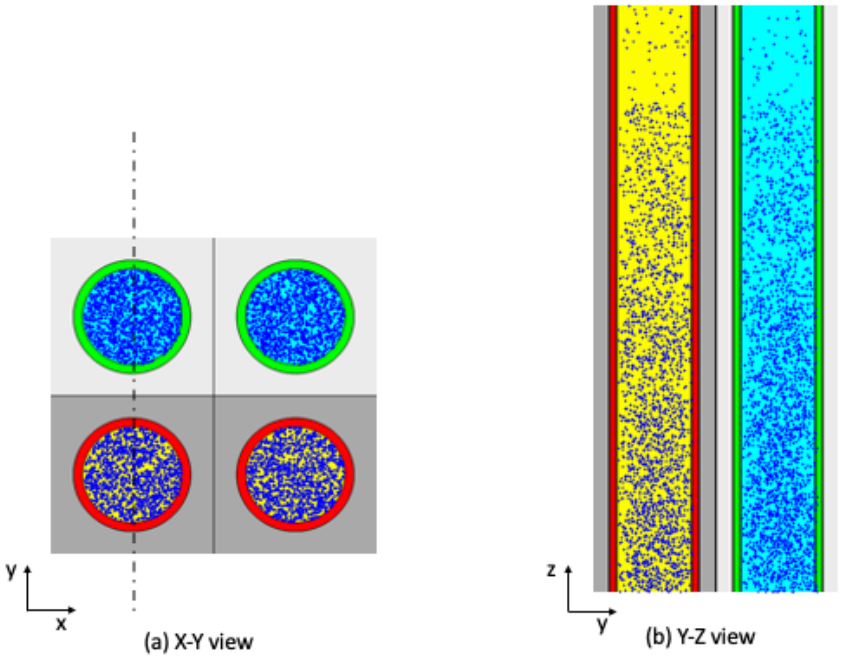


Fig. 8.1.17: Starting point distribution obtained with start type 8 overlayed on the geometry

ASCII Start Data File Format

A sample ASCII start data file is shown below. In a typical ASCII start data file, each line defines a data set. The first three columns are `TFX`, `TFY`, and `TFZ` values, the next four columns are `NXS`, `NYS`, `NZS`, and `KFS`, and the last column is always `LNU`. The file must be ended with a termination line with `LNU` =0. The file format is given with `(3es20.9,5i12)`.

-1.000000000e-01	-2.000000000e+00	-3.300000000e+00	0	0	0	1	1
-3.000000000e-01	-2.400000000e+00	-3.300000000e+00	0	0	0	1	15
-3.000000000e-01	-2.500000000e+00	-3.300000000e+00	0	0	0	1	36
-4.000000000e-01	-2.600000000e+00	-3.300000000e+00	0	0	0	1	37
-4.000000000e-01	-2.700000000e+00	-3.300000000e+00	0	0	0	1	38
-5.000000000e-01	-2.800000000e+00	-3.300000000e+00	0	0	0	1	39
-5.000000000e-01	-2.900000000e+00	-3.300000000e+00	0	0	0	1	40
-5.000000000e-01	-2.900000000e+00	-3.300000000e+00	0	0	0	1	0

8.1.3.9 Extra 1-D XSECS IDs data

Extra 1-D cross section IDs are not required. They are allowed as input in order to simplify future modifications to calculate reaction rates, etc., as well as for compatibility with other SCALE codes. The syntax for the extra 1-D cross section data block is:

`READ X1DS NEUTRON` *i1* ... *ix1d* `END X1DS`

`NEUTRON` is a keyword to indicate that the following ID identifies a neutron interaction.

***i*₁ ... *i*_{x1d}** `X1D` 1-D identification numbers or keyword identifiers for the 1-D cross section to be used. These cross sections must be available on the mixture cross section library. `X1D` entries are expected to be read (see integer PARAMETER data).

8.1.3.10 Mixing table data

A cross section mixing table must be entered if KENO is being run stand alone and a Monte Carlo cross section format library is not being used in the multigroup mode, or KENO is being run stand alone in the continuous energy mode. If the parameter `LIB=` (Sect. 8.1.3.3) is entered, then mixing table data must be entered. A cross section mixing table is entered using the following syntax:

`READ MIXT` p_1 ... p_N `END MIXT`

***p*₁ ... *p*_N** are *N* parameters that might or might not be keyworded.

The possible parameters that can be used in a `MIXT` block are described below.

`SCT` **= *nsct*** is used to input the number of scattering angles and only applies in multigroup mode. *nsct* is the number of discrete scattering angles, default = 1. The number of scattering angles specifies the number of discrete scattering angles to be used for the cross sections. If SCT is not set (i.e., SCT= -1), then the number of scattering angles is determined from the cross section library specified. The number of scattering angles defaults to (ncoef+1)/2, where ncoef is the largest Legendre polynomial order used in the problem. It needs to be entered only once for a problem. If more than one value is entered, the last one is used for the problem. For assistance in determining the number of discrete scattering angles for the cross sections, see Sect. 8.1.4.4.3.

`EPS` **= *pbxs*** is used to enter the cross section message cutoff value, and it only applies in multigroup mode. *pbxs* is the value of the P_0 cross section for each transfer, above which generated warning messages will be printed, default = 3×10^{-5}. The primary purpose of entering this cutoff value is to suppress printing these messages when they are generated during cross section processing. For assistance in determining a value for EPS, see Sect. 8.1.4.4.4.

`MIX` **= *mix*** is used to input the identification number of the mixture being described. *mix* defines the mixture being described.

`NCM` **= *ncmx*** is used to input the nuclide mixture IDs to be used for this mixture. *ncmx* defines the nuclide mixture ID. When `MIX`=*mix* is read, *ncmx* is defaulted to *mix* also. Then, as long as all the nuclides that need to be mixed into *mix* already have *mix* specified as their nuclide mixture (frequently the case when using SCALE), the user does not need to specify `NCM`. The most usual case where `NCM` must be specified is when the mixtures were specified as a different mixture number when they were created in SCALE as compared to the mixture number used for them in KENO. Cell homogenized mixtures also need `NCM` specified.

`TMP|TEM` **= *temperature*** is used to input the desired temperature of the CE cross section data.

`nucl` is the nuclide ID number from the AMPX working format cross section library.

XS=*fname* is used to input the optional continuous energy cross section filename to override the default cross sections. *fname* is the name of the file.

dens is the number density (atoms/b-cm) associated with nuclide ID number *nucl*.

The sequence "*nucl* NCM=*ncmx* [XS=*fname*] *dens*" may be repeated until the mixture defined by MIX=*mix* has been completely described.

The sequence "MIX = *mix* NCM = *ncmx* TMP|TEM = *temperature nucl* NCM=*ncmx* [XS=*fname*] *dens*" may be repeated until all the mixtures have been described.

Note: If a given nuclide ID is entered more than once in the same mixture, then the number densities for that nuclide are summed.

If a mixture number is used as a nuclide ID, then it is treated as a nuclide and the number density associated with it is used as a density modifier. (If the density is entered as 1, then the mixture is mixed in at full density. If it is entered as 0.5, the mixture is mixed in at one half of its full density.) A Monte Carlo formatted cross section library is generated on the unit defined by the parameter XSC=. If this data set is saved, subsequent cases can utilize these mixtures without remixing.

The entry XS=*fname* is optional. If a nuclide is entered more than once in a mixture and this entry is specified, then they must be the same (i.e., cannot use more than one continuous energy cross section sets for a nuclide in a given mixture). Different mixtures may have the same nuclide with different continuous energy cross section sets.

8.1.3.11 Plot data

Plots of slices specified through the geometry can be generated and displayed (1) as character plots using alphanumeric characters to represent mixture numbers, unit numbers or bias ID numbers or (2) as color plots which generate a PNG file using colors to represent mixture numbers, unit numbers or bias ID numbers. Color plots require an independent program to display the PNG file to a PC or workstation monitor or to convert the file to be displayed using a plotting device. The keyword SCR= is used to control the plot display method. SCR=YES, the default value, uses the color plot display method. SCR=NO uses the character plot display method. The value of SCR determines the plot display method for all the plots specified in a problem. If SCR= is entered more than once, the last entry determines the plot display method. In other words, all plots generated by a problem will be either character plots or color plots.

The plot data can include the data for any or all types of plots. A plot by mixture number is the default. The kind of plot is defined by the parameter PIC=. Character plots are printed after the volumes are printed and before the final preparations for tracking are completed. Plot data are not required for a problem, but theyb can be used to verify the problem description. The actual plotting of the picture can be suppressed by entering PLT= NO in the parameter data or plot data. This allows plot data to be kept in the problem input for reference purposes without actually plotting the picture(s). Entering a value for PLT in the plot data will override any value entered in the parameter data. However, if a problem is restarted, the value of PLT from the parameter data is used. The upper left and lower right coordinates of the plot must be specified relative to the origin of the problem. See Sect. 8.1.4.10 for a discussion of plot origins and plot data.

Enter the plot data using the following syntax:

READ PLOT p_1 ... p_N END PLOT

***p*$_1$ ... *p*$_N$** are *N* parameters entered using keywords followed by the appropriate data. The plot title and the plot character string must be contained within delimiters. Enter as many picture parameters as necessary to describe the plot. Multiple sets of plot data can be entered. The parameter input for each plot is terminated by a labeled or unlabeled END. The labeled END cannot use the word PLOT as the first four characters of the label. For example, END PLT1 is a valid label, but END PLOT1 is not. If an unlabeled END is used, it cannot start in column 1.

The possible parameters that can be used in a PLOT block are described below.

TTL=*delim ptitl delim* Enter a one-character delimiter *delim* to signal the beginning of the title (132 characters maximum). The title is terminated when *delim* is encountered the second time. Acceptable delimiters include ” , ` , * , ^ , or !. Default = title of the KENO case.

PIC= *wrd* The plot type, *wrd*, is followed by one or more blanks and must be one of the keywords listed below. The plot type is initialized to MAT; the default is the value from the previous plot.

MAT MIX[T[URE]] MEDI[A]	These keywords will cause the plot to represent the mixture numbers used in the specified geometry slice.
UNT UNIT[TYPE]	These keywords will cause the plot to represent the units used in the specified geometry slice. In the legend of the color plot, the material number actually refers to the units.
IMP BIAS[ID] WTS WEIG[HTS] WGT[S]	These keywords will cause the plot to represent the bias ID numbers used in the specified geometry slice. In the legend of the color plot, the material number actually refers to the bias ID numbers.

TYP = Enter the type desired. XY for an X-Y plot XZ for an X-Z plot YZ for a Y-Z plot Direction cosines do not need to be entered if TYP is entered.

Plot coordinates Enter values for the upper left and lower right coordinates of the plot as described below. **Data must be entered for all nonzero coordinates unless all six values from the previous plot are to be used.**

Upper left coordinates Enter the X, Y, and Z coordinates of the upper left-hand corner of the plot.

XUL=*xul* is used to enter the X coordinate of the upper left-hand corner of the plot. Default = value from previous plot; initialized to zero if any other coordinates are entered.

YUL=*yul* is used to enter the Y coordinate of the upper left-hand corner of the plot. Default = value from previous plot; initialized to zero if any other coordinates are entered.

ZUL=*zul* is used to enter the Z coordinate of the upper left-hand corner of the plot. Default = value from previous plot; initialized to zero if any other coordinates are entered.

Lower right coordinates Enter the X, Y, and Z coordinates of the lower right-hand corner of the plot.

XLR=*xlr* is used to enter the X coordinate of the lower right-hand corner of the plot. Default = value from previous plot; initialized to zero if any other coordinates are entered.

YLR=*ylr* is used to enter the Y coordinate of the lower right-hand corner of the plot. Default = value from previous plot; initialized to zero if any other coordinates are entered.

ZLR=*zlr* is used to enter the Z coordinate of the lower right-hand corner of the plot. Default = value from previous plot; initialized to zero if any other coordinates are entered.

Direction cosines across the plot Enter direction numbers proportional to the direction cosines for the AX axis of the plot. The AX axis is from left to right across the plot. If any one of the AX direction cosines is entered, the other two are set to zero. The direction cosines are normalized by the code.

UAX=*uax* is used to enter the X component of the direction cosines for the AX axis of the plot. Default = value from previous plot; initialized to zero if any other direction cosines are entered.

VAX=*vax* is used to enter the Y component of the direction cosines for the AX axis of the plot. Default = value from previous plot; initialized to zero if any other direction cosines are entered.

WAX=*wax* is used to enter the Z component of the direction cosines for the AX axis of the plot. Default = value from previous plot; initialized to zero if any other direction cosines are entered.

Direction cosines down the plot Enter direction numbers proportional to the direction cosines for the DN axis of the plot. The DN axis is from top to bottom down the plot. If any one of the DN direction cosines is entered, the other two are set to zero. The direction cosines are normalized by the code.

UDN=*udn* is used to enter the X component of the direction cosines for the DN axis of the plot. Default = value from previous plot; initialized to zero if any other direction cosines are entered.

VDN=*vdn* is used to enter the Y component of the direction cosines for the DN axis of the plot. Default = value from previous plot; initialized to zero if any other direction cosines are entered.

WDN=*wdn* is used to enter the Z component of the direction cosines for the DN axis of the plot. Default = value from previous plot; initialized to zero if any other direction cosines are entered.

Scaling parameters Enter one or more scaling parameters to define the size of the plot.

Note: If any of the scaling parameters are entered for a plot, the value of those that were not entered is recalculated. If none of the scaling parameters are specified for a plot, the values from the previous plot are used.

DLX=*dlx* is used to input the horizontal spacing between points on the plot. Default = value from previous plot; initialized to zero if NAX or NDN is entered.

DLD=*dld* is used to input the vertical spacing between points on the plot. Default = value from previous plot; initialized to zero if NAX or NDN is entered.

Note: If either DLX or DLD is entered, the code will calculate the value of the other. If both are entered, the plot may be distorted.

NAX=*nax* is used to input the number of intervals to be printed across the plot. Default = value from previous plot; initialized to zero if DLX or DLD is entered.

NDN=*ndn* is used to input the number of intervals to be printed down the plot. Default = value from previous plot; initialized to zero if DLX or DLD is entered.

Global scaling parameter

LPI=*lpi* is used to input a scaling factor used to control the horizontal to vertical proportionality of a plot or plots. SCALE 4.3 and later versions allow *lpi* to be input as a floating point number. For an undistorted character plot, *lpi* should be specified as the number of characters down the page that occupy the same distance as ten characters across the page. For an undistorted color plot, *lpi* should be entered as ten times the ratio of the vertical pixel dimension to the horizontal pixel dimension. The default value of *lpi* is 8.0 for a character plot and 10.0 for a color plot. *lpi*=10 will usually display an undistorted color plot.

The value entered for *lpi* applies to all plot data following it until a new value of *lpi* is specified.

Note: Plot data must include the specification of the upper left corner of the plot and the direction cosines across and down the plot.

Additional data required to generate a plot are one of the following combinations:

1. the lower right corner of the plot, the global scaling parameter, LPI, and one of the scaling parameters (DLX, DLD, NAX, NDN).

2. the lower right corner of the plot, one of the scaling parameters related to the horizontal specifications of the plot (DLX or NAX), and one of the scaling parameters related to the vertical specification of the plot (DLD or NDN). LPI, even if specified will not be used.

3. NAX and NDN and any two of LPI, DLX, and DLD. If LPI, DLX, and DLD are all specified, LPI is not used.

The data required to generate a plot may be supplied from (1) defaulted values, (2) data from the previous plot, or (3) data that are specifically entered for the current plot.

Miscellaneous parameters Enter miscellaneous parameters

RUN=*run* is used to determine if the problem is executed or is terminated after data checking. A value of YES for *run* means the problem will be executed if all the data were acceptable. A value of NO specifies the problem will be terminated after data checking is completed. The default value of RUN is YES.

PLT=*plt* is used to specify if a plot is to be made. A value of YES for *plt* specifies that a plot is to be made. If plot data are entered, PLT is defaulted to YES.

Note: The parameters RUN and PLT can also be entered in the PARAMETER data. See Sect. 8.1.3.3. It is recommended that these parameters be entered only in the parameter data block in order to ensure that the data printed in the "Logical Parameters" table are what is actually performed.

SCR= *src* This is used to determine the plot display method. The plot display method is specified by entering either YES or NO for *src*. The default value is YES. SCR=YES uses the color plot display method. SCR=NO uses the character plot display method. If SCR is entered more than once in a problem, the last value entered is the one that is used.

NCH= *delim char delim* Enter only if plots are to be made utilizing the character plot display method (SCR=NO). Enter a delimiter (i.e., " , ` , * , ^ , or !) to signal the beginning of character string *char*. The character string is terminated when the *delim* character is encountered the second time. Do

not use the initial delimiter in the *char* string, as it will be read as terminating the string. *char* is a character string with each entry representing a plottable quantity (i.e., media {mixture} number, unit number, or bias ID). These are the characters that will be used in the plot. The first entry represents media, unit, or bias ID zero; the second entry represents the smallest media, unit, or bias ID used in the problem; the third entry represents the next larger media, unit, or bias ID used in the problem; etc. For example, assume PIC=MAT is specified, and 15 mixtures are defined in the mixing table, and the geometry data use only mixtures 3 and 7. By default, a blank will be printed for mixture zero, a 1 will be printed for mixture 3, and a 2 will be printed for mixture 7. If you wish to print a zero for a void (mixture 0), a 3 for mixture 3, and a 7 for mixture 7, enter NCH=`037'.

The default values of CHAR are the following:

Quantity	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SYMBOL		1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Quantity	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SYMBOL	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
Quantity	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	
SYMBOL	W	X	Y	Z	#	,	$	-	+	)		&	>	:	;	
Quantity	47	48	49	50	51	52	53	54	55	56	57	58				
SYMBOL	·	-	"%"	"*"	""“	"="	"!"	"("	"@"	"<"	"/"	0				

CLR= ***n*₁ *r*(*n*₁) *g*(*n*₁) *b*(*n*₁) … *n*N *r*(*n*N) *g*(*n*N) *b*(*n*N)** **END COLOR** this entry is used to define the colors to be used by the color plot. It may be entered only if plots are to be made utilizing the color plot display method (SCR=YES). After entering the keyword CLR=, 4 numbers are entered *N* times. The first number, n_1, represents a media (mixture) number, unit number, or bias ID. The next three numbers, whose values can range from 0 through 255, define the *red, green, and blue* components of the color that will represent this n_1 in the plot. The sequence of 4 numbers is repeated until the colors associated with all of the media (mixture) numbers, unit numbers, or bias IDs used in the problem have been defined. The smallest number that can be entered for n_i is -1, representing undefined regions in the plot. An *ni* of 0 represents void regions; n_i of 1 represents the smallest media, unit, or bias ID used in the problem; n_i of 2 represents the next larger media, unit, or bias ID used in the problem, etc. The color plot definition data are terminated by entering the keywords END COLOR. A total of 256 default colors are provided in Table 8.1.28 Two of those colors represent undefined regions, n_i=-1*, as black and void regions, and n_i=0 as gray. The remaining 254 colors represent the default values for mixtures, bias IDs, or unit numbers used in the problem. If **num** is entered as **-1**, the next three numbers define the color that will be used to represent undefined regions of the plot. The default color

for undefined regions is black, represented as 0 0 0. If n_i is entered as 0, the next three numbers define the color that will represent void regions in the plot. The default color for void is gray, represented as 200 200 200. For example, assume a color plot is to be made for a problem that uses void regions and mixture numbers 1, 3, and 5. By default, the undefined regions (Index -1) will be black; void regions (Index 0) will be gray; the first mixture, mixture 1 (Index 1), will be medium blue; the next larger mixture, mixture 3 (Index 2), will be turquoise2; and the last mixture, mixture 5 (Index 3), will be green2. If these values are acceptable, data do not need to be entered for `CLR=`. If the user decides to define void to be white (255 255 255), mixture 1 to be red (255 0 0), mixture 3 to be bright blue (0 0 255), and mixture 5 to be green (0 255 0), then the following data could be entered:

`CLR=0 255 255 255 1 255 0 0 2 0 0 255 3 0 255 0 END COLOR`

In this example, the first number (0) defines the void, and the next three numbers are the *red, green, and blue* components that combine as the color white. The fifth number (1) represents the smallest mixture number (mixture 1), and the next three numbers are the *red, green, and blue* components of red. The ninth number (2) represents the next larger mixture number (mixture 3), and the next three numbers are the *red, green, and blue* components of bright blue. The thirteenth number (3) represents the next larger mixture number (mixture 5), and the next three numbers are the red, green, and blue components of green. The `END COLOR` terminates the color definition data. Because color data were not entered for *ni* of -1, undefined regions will be represented by the color black, the default specification from Table 8.1.28. The *red, green, and blue* components of some bright colors are listed below.

Display Color	**red**	**green**	**blue**
black	0	0	0
white	255	255	255
“default void gray”	200	200	200
red	255	0	0
green	0	255	0
brightest blue	0	0	255
yellow	255	255	0
brightest cyan	0	255	255
magenta	255	0	255

The 256 default colors are listed in Table 8.1.28.

Table 8.1.28: Default color specifications for the color plot display method

Mixture number	Red	Green	Blue	Color
-1	0	0	0	
0	200	200	200	
1	0	0	205	
2	0	229	238	
3	0	238	0	
4	205	205	0	
5	238	0	0	
6	145	44	238	
7	150	150	150	
8	240	200	220	
9	0	191	255	
10	224	255	255	
11	0	255	127	
12	255	255	224	
13	255	0	0	
14	255	0	255	
15	67	110	238	
16	174	238	238	
17	180	238	180	
18	238	220	130	
19	238	99	99	
20	238	122	233	
21	25	25	112	
22	0	0	128	
23	100	149	237	

Mixture number	Red	Green	Blue	Color
24	72	61	139	
25	106	90	205	
26	123	104	238	
27	30	144	255	
28	135	206	235	
29	135	206	250	
30	70	130	180	
31	176	196	222	
32	176	224	230	
33	0	206	209	
34	72	209	204	
35	95	158	160	
36	102	205	170	
37	127	255	212	
38	0	100	0	
39	85	107	47	
40	143	188	143	
41	60	179	113	
42	32	178	170	
43	152	251	152	
44	0	255	0	
45	127	255	0	
46	0	250	154	
47	173	255	47	
48	50	205	50	

Mixture number	Red	Green	Blue	Color
49	154	205	50	
50	34	139	34	
51	107	142	35	
52	189	183	107	
53	240	230	140	
54	238	232	170	
55	250	250	210	
56	255	255	224	
57	255	255	0	
58	255	215	0	
59	238	221	130	
60	184	134	11	
61	188	143	143	
62	205	92	92	
63	139	69	19	
64	160	82	45	
65	205	133	63	
66	222	184	135	
67	245	245	220	
68	245	222	179	
69	244	164	96	
70	210	105	30	
71	178	34	34	
72	165	42	42	
73	233	150	122	

Mixture number	Red	Green	Blue	Color
74	250	128	114	
75	255	160	122	
76	255	165	0	
77	255	140	0	
78	255	127	80	
79	240	128	128	
80	255	99	71	
81	255	69	0	
82	255	0	0	
83	255	105	180	
84	255	20	147	
85	255	192	203	
86	255	182	193	
87	219	112	147	
88	176	48	96	
89	199	21	133	
90	208	32	144	
91	238	130	238	
92	221	160	221	
93	218	112	214	
94	153	50	204	
95	148	0	211	
96	186	85	211	
97	138	43	226	
98	160	32	240	

Mixture number	Red	Green	Blue	Color
99	147	112	219	
100	216	191	216	
101	238	233	233	
102	238	229	222	
103	238	223	204	
104	238	213	183	
105	238	203	173	
106	238	207	161	
107	238	233	191	
108	238	232	205	
109	238	238	224	
110	224	238	224	
111	238	224	229	
112	238	213	210	
113	224	238	238	
114	122	103	238	
115	67	110	238	
116	0	0	238	
117	28	134	238	
118	92	172	238	
119	0	178	238	
120	126	192	238	
121	164	211	238	
122	185	211	238	
123	188	210	238	

Mixture number	Red	Green	Blue	Color
124	178	223	238	
125	209	238	238	
126	174	238	238	
127	142	229	238	
128	0	238	238	
129	141	238	238	
130	118	238	198	
131	180	238	180	
132	78	238	148	
133	144	238	144	
134	0	238	118	
135	0	238	0	
136	118	238	0	
137	179	238	58	
138	188	238	104	
139	238	230	133	
140	238	220	130	
141	238	238	209	
142	238	238	0	
143	238	201	0	
144	238	180	34	
145	238	173	14	
146	238	180	180	
147	238	99	99	
148	238	121	66	

Mixture number	Red	Green	Blue	Color
149	238	197	145	
150	238	216	174	
151	238	154	73	
152	238	118	33	
153	238	44	44	
154	238	59	59	
155	238	130	98	
156	238	149	114	
157	238	154	0	
158	238	118	0	
159	238	106	80	
160	238	92	66	
161	238	64	0	
162	238	0	0	
163	238	18	137	
164	238	106	167	
165	238	169	184	
166	238	162	173	
167	238	121	159	
168	238	48	167	
169	238	58	140	
170	238	0	238	
171	238	122	233	
172	238	174	238	
173	209	95	238	

Mixture number	Red	Green	Blue	Color
174	178	58	238	
175	145	44	238	
176	159	121	238	
177	238	210	238	
178	255	250	250	
179	139	137	137	
180	255	245	238	
181	255	228	196	
182	255	218	185	
183	255	250	205	
184	255	248	220	
185	255	255	240	
186	240	255	240	
187	255	240	245	
188	255	228	225	
189	240	255	255	
190	131	111	255	
191	72	118	255	
192	30	144	255	
193	99	184	255	
194	0	191	255	
195	135	206	255	
196	176	226	255	
197	198	226	255	
198	202	225	255	

Mixture number	Red	Green	Blue	Color
199	191	239	255	
200	224	255	255	
201	187	255	255	
202	152	245	255	
203	0	245	255	
204	151	255	255	
205	127	255	212	
206	193	255	193	
207	84	255	159	
208	154	255	154	
209	0	255	127	
210	127	255	0	
211	192	255	62	
212	202	255	112	
213	255	246	143	
214	255	236	139	
215	255	255	224	
216	255	215	0	
217	255	193	37	
218	255	185	15	
219	255	193	193	
220	255	106	106	
221	255	130	71	
222	255	211	155	
223	255	231	186	

Mixture number	Red	Green	Blue	Color
224	255	165	79	
225	255	127	36	
226	255	48	48	
227	255	64	64	
228	255	140	105	
229	255	160	122	
230	255	165	0	
231	255	127	0	
232	255	114	86	
233	255	99	71	
234	255	69	0	
235	255	20	147	
236	255	110	180	
237	255	181	197	
238	255	174	185	
239	255	130	171	
240	255	52	179	
241	255	62	150	
242	255	131	250	
243	255	187	255	
244	224	102	255	
245	191	62	255	
246	155	48	255	
247	171	130	255	
248	255	225	255	

Mixture number	Red	Green	Blue	Color
249	139	0	139	
250	139	0	0	
251	0	139	0	
252	0	0	139	
253	0	139	139	
254	139	139	0	

8.1.3.12 Energy group boundary data

Upper energy group boundary data in eV are entered to determine the groups into which the tallies will be collected in the continuous energy mode. For G groups G+1 entries are entered. The last entry is the lower energy boundary of the last group. The values must be in descending order. The parameter NGP is set equal to the number of entries-1. The syntax is:

READ ENERGY u_1 ... uG u_{G+1} END ENERGY

$u_1 \ldots u_G$ are the upper energy limits of energy groups 1 ... G, respectively.

u_{G+1} is the lower energy limit of energy group G.

Example:

```
READ ENERGY

2e7 1e5 1 1e-5

END ENERGY
```

Defines a 3-group structure with group 1 (2e+7 eV to 1e+5 eV), group 2 (1e+5 eV to 1 eV), and group 3 (1 eV to 1e-5 eV) and sets NGP=3.

Energy group boundary data are optional. Default values for the energy group boundaries in the calculations are determined as in the following order:

i. Use energy group boundaries from ENERGY block if specified in the input. The number of entries in the ENERGY block is NGP+1.

ii. If only NGP is specified (in PARAMETER block) in the input and NGP is equal to the number of energy groups in one of the SCALE neutron cross section libraries, the energy group structure from that library will be used.

iii. If only NGP is specified (in PARAMETER block) in the input and NGP is not equal to the number of energy groups in one of the SCALE neutron cross section libraries, NGP equal lethargy bins will be used.

iv. Use SCALE 252-group structure as default, NGP=252.

8.1.3.13 Volume data

If volumes are needed (for calculating fission densities, fluxes, etc.), then the data necessary to determine them are entered. The syntax for this block is:

READ VOLUME p_1 ... p_N END VOLUME

$p_1 \ldots p_N$ are N parameters entered using keywords followed by the appropriate data.

The possible parameters that can be used in a VOLUME block are described below.

READVOL=*vol* used to input the file name (up to 256 characters) of the file from which user-specified volumes are read. This is an optional parameter and only works for KENO-VI. The data are read in sections for each UNIT contained in the problem. First the keyword "UNIT" is read, followed by the UNIT number. For that UNIT the data for each region containing material in the order shown in the input is read as follows: the keyword "MEDIA" is read, followed by the mixture number, followed by the keyword "VOL=", followed by the total volume for that region. Regions containing ARRAYs and HOLEs are skipped. An example of the data contained in a volume file is given later in this section.

TYPE=*vcalc* used to determine the type of volume calculation. *vcalc* can have the values:

NONE: (only works in KENO-VI, where it is also the default) No volume calculation, volumes are set to -1.0 (only in KENO-VI).

TRACE: A trapezoidal integration will be performed (only in KENO-VI).

RANDOM: A Monte Carlo integration will be performed.

NRAYS=*ntotal* the number of intervals used in the trapezoidal integration (default 100,000). Used only with TYPE=TRACE (KENO-VI).

BATCHES=*nloop* the number of batches to be used in the Monte Carlo integration (default 500). Used only with TYPE=RANDOM.

POINTS=*nplp* the number of points per batch used in the Monte Carlo integration (default 1000). Used only with TYPE=RANDOM.

XP= *xp* the pls X face of the encompassing cuboid.

XM=*xm* the minus X face of the encompassing cuboid.

YP=*yp* the plus Y face of the encompassing cuboid.

YM=*ym* the minus Y face of the encompassing cuboid.

ZP=*zp* the plus Z face of the encompassing cuboid.

ZM*zm* the minus Z face of the encompassing cuboid.

SAMPLE_DEN=*sampleden* the density of sampling points per cm^3 per batch. Used only with TYPE=RANDOM.

IFACE =*fname* the face of the enclosing cuboid where the trapezoidal integration will be performed. Enter either `XFACE`, `YFACE`, or `ZFACE`. KENO-VI will integrate over the face with the smallest area by default. This allows specifying a different face. Used only for `TYPE=TRACE` (KENO-VI).

The volume parameters include specifying the type of calculation to determine the volumes and additional parameters needed for the selected type. In KENO-VI the default type is NONE (i.e., no volume calculation will be performed), and the volumes for regions not containing `HOLE`s or `ARRAY`s will be set to -1.0. No other data are needed for this type.

In KENO-VI the volume data may be entered for any or all regions within the geometry data by placing the keyword `VOL=` followed by the total volume of that region in the problem at the end of the `MEDIA` card. See Sect. 8.1.3.4 (Geometry Data) for more details.

For KENO-VI, in the same problem, volumes may be entered using a combination of three methods: (1) in the geometry data using `VOL=`, (2) read from a volume file, and (3) calculated. The calculated volumes (method 3) are obtained for both the regions, and the meshes are defined by a grid (such as in TSUNAMI runs). As for KENO V.a, the mesh volumes must always be calculated (i.e., there is no method to input the mesh volumes). If volumes are entered or calculated using more than one method, the following hierarchy is used to determine which volume is used for the regions.

1. Volumes entered as part of a `MEDIA` card using `VOL=` are always used.
2. Volumes read from the volume file are used if that volume for the region was not specified using VOL= following a `MEDIA` card.

3. Calculated volumes are used if they are not specified using VOL= and if there is no volume file or data for that region on the volume file.
4. Volumes that have not been set or calculated will be set to -1.0. This may result in negative fluxes and fission densities for these regions.
5. Volumes are only calculated for regions containing material. Regions containing ARRAYs or HOLEs have no volume. Those volumes are associated with the UNIT contained in the ARRAY or HOLE.

In KENO V.a, the region volumes are always calculated by the code without the user's intervention. This is possible because KENO V.a has no region intersections, so calculation of the volumes is always possible using analytical methods. The use of the (RANDOM) calculated volumes using the VOLUME block is then only justified when the user needs to calculate the volumes defined by a grid, such as for TSUNAMI calculations.

When volumes are calculated using either RANDOM or TRACE, then a file containing volumes and named _volxxxx (where xxxx is an 18-digit number with the leftmost unused digits padded with zeros) is created in the temporary directory. The program searches the temporary directory for a file name beginning with _vol. If it is not found, the volume file that is created is named _vol000000000000000000. If a file exists, then a new file will be created where the file number is the largest number associated with a previous volume file incremented by 1. The file is automatically copied to the user directory with the input file base name prepended to it, such as inputfile.volxxxx.volumes.

Below is an example of the VOLUME data block associated with a case in which volumes are being calculated using ray tracing. The number of rays used is set to one million, and if the outer unit volume is not a cuboid, then a cuboid will be placed around the global region prior to calculating volumes.

```
read volume
type=trace nrays=1000000
XP=10  XM=-15 YP=15  YM=-15  ZP=15  ZM=-15
end volume
```

Below is an example of the VOLUME data block associated with a case where volumes are being calculated using random sampling. The number of particles per batch is set to 100,000, and the number of batches used is set to 500. After being calculated, the volume data will be written to a file in the temporary directory as discussed above.

```
read volume
type=random points=100000 batches=500
XP=10  XM=-10  YP=15  YM=-15  ZP=25  ZM=15
end volume
```

Below is an example of the VOLUME data block associated with a case where volumes are both read in from the file VOLUME_DATA and calculated using random sampling. The number of particles per generation is set to 1,000,000, and the number of generations used is set to 500. The file VOLUME_DATA must be formatted as shown below. The calculated volume data are written in the temporary working directory to a file as discussed above. Calculating volume data for some volume regions and providing input volume data for others may be useful if only part of the volume data is known and the remaining data need to be calculated.

```
read volume
type=random points=1000000 batches=500
readvol=volume_DAta
end volume
```

Example volume file VOLUME_DATA:

```
UNIT 1
   MEDIA 1 VOL=110.0
   MEDIA 2 VOL=2435.8
   MEDIA 2 VOL=3242.9
UNIT 2
   MEDIA 2 VOL=342.8
   MEDIA 0 VOL=4235.0
```

Below is an example of a sample problem in which volumes are being calculated using random sampling. The number of particles per generation is set to 100,000, and the number of generations used is set to 500. After being calculated, the VOLUME data will be written as described above.

```
=CSAS6
SAMPLE PROBLEM WITH VOLUMES CALCULATED AND PRINTED TO FILE
v7.1-252n
READ COMP
URANIUM 1 DEN=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
READ GEOMETRY
UNIT 3
COM='SINGLE UNIT CENTERED'
SPHERE 10 4.000
CUBOID 20 6P6.0
MEDIA 1 1 10
MEDIA 0 1 20 -10
BOUNDARY 20
UNIT 1
COM='SINGLE UNIT CENTERED'
SPHERE 10 5.000
CUBOID 20 6P6.0
MEDIA 1 1 10
MEDIA 0 1 20 -10
BOUNDARY 20
GLOBAL UNIT 2
COM='Global UNIT'
CUBOID 10 6P18.0
ARRAY 1 10 PLACE 2 2 2 3R0.0
BOUNDARY 10
END GEOMETRY
READ ARRAY ARA=1 NUX=3 NUY=3 NUZ=3 TYP=CUBOIDAL FILL
1 1 1 1 1 1 1 1 3 3 3 3 3 3 3 3 1 1 1 1 1 1 1 1 END ARRAY
READ VOLUME
TYPE=RANDOM POINTS=100000 BATCHES=500
END VOLUME
END DATA
END
```

Example volume file _volxxxx:

```
UNIT 1
   MEDIA 1 VOL=9423.45
   MEDIA 0 VOL=21678.6
UNIT 3
   MEDIA 1 VOL=2410.81
   MEDIA 0 VOL=13143.1
```

Below is an example of a problem with the volumes entered in the geometry data block using VOL= followed by the volume for all MEDIA type content records. Note that the keyword VOL= should never follow a HOLE or ARRAY content record.

```
=csas26
Sample problem with volumes input in geometry data
```

(continues on next page)

(continued from previous page)

```
v7.1-252n
read comp
uranium  1 den=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 end
end comp
read geometry
unit 3
com='unit 3'
sphere   10  4.000
cuboid   20  6p6.0
media  1 1 10       vol=2412.7
media  0 1 20 -10   vol=13139.3
boundary 20
unit 1
com='UNIT 1'
sphere   10  5.000
cuboid   20  6p6.0
media  1 1 10       vol=9424.8
media  0 1 20 -10   vol=21679.2
boundary 20
global unit 2
com='GLOBAL Unit 2'
cuboid 10  6p18.0
array 1    10 place 2 2 2 3r0.0
boundary  10
end geometry
read array  ara=1 nux=3 nuy=3 nuz=3  typ=cuboidal fill
1 1 1 1 1 1 1 1 3 3 3 3 3 3 3 3 1 1 1 1 1 1 1 1  end array
end data
end
```

8.1.3.14 Grid geometry data

This data block is used to input the data needed to define a Cartesian grid for tallying purposes.

READ GRID N $p_1 \ldots p_L$ END GRID

N mesh grid identifier, always entered.

$p_1 \ldots p_L$ are L parameters chosen from the list below. The parameters are entered using keywords followed by the appropriate data, except for the grid identifier, which is always entered first as an integer.

N[UM]XCELLS=*numx* number of cells in the x direction, default = 1.

N[UM]YCELLS=*numy* number of cells in the y direction, default = 1.

N[UM]ZCELLS=*numz* number of cells in the z direction, default = 1.

XMIN=*xmin* minimum cell boundary in the x direction, default = 0.

XMAX=*xmax* maximum cell boundary in the x direction, default = 1.

YMIN=*ymin* minimum cell boundary in the y direction, default = 0.

YMAX=*ymax* maximum cell boundary on the y direction, default = 1.

ZMIN=*zmin* minimum cell boundary in the z direction, default = 0.

ZMAX=*zmax* maximum cell boundary in the z direction, default = 1.

XPLANES*xplanes* the cell boundaries in the x direction followed by end, default = 0, 1 end.

YPLANES*yplanes* the cell boundaries in the y direction followed by end, default = 0, 1 end.

ZPLANES=*zplanes* the cell boundaries in the z direction followed by end, default = 0, 1 end.

XLINEAR=*numcellsx* *x_{min} x_{max} generate y-z planes from x_{min} to x_{max} creating *numcellsx* intervals.

YLINEAR=*numcellsy* *y_{min} y_{max} generate x-z planes from y_{min} to y_{max}, creating *numcellsy* intervals.

ZLINEAR=*numcellsz* *z_{min} z_{max} generate x-y planes from z_{min} to z_{max}, creating *numcellsz* intervals.

TITLE=*title* optional title for this mesh grid. Only used in KENO if an error in the grid causes a debug print.

If *numx, xmin, xmax* are entered, then the code will calculate *numx* equally spaced cells in the x direction between *xmin* and *xmax*.

If *xplanes* is entered, then the code will count the number of unique *xplanes*, and order them from minimum to maximum, deleting any duplicates.

If the user inputs both sets of data, then the code will use the *xplanes* data.

If *xplanes* and *xlinear* are both entered, then the code will retain all unique planes from *xplanes* and all *xlinear* entries provided. The above also applies to Y and Z.

NOTE: The user **MUST** set the minimum and maximum values in each direction so that the actual geometry is totally covered by the mesh for mesh flux tally that is used in TSUNAMI sensitivity calculations.

KENO checks for and eliminates duplicate or nearly duplicate planes.

The user may specify multiple mesh grids; each must be defined in separate READ GRID blocks. In this case, each grid should have different *N* (grid ID number). See Sect. 8.1.4.11 for details and samples.

8.1.3.15 Reaction data

The reaction data block is used to specify the type of tally (e.g., reaction rates, flux, and few group reaction cross sections) and the reaction/nuclide pairs in any mixture used in the problem for reaction tally calculations. This block is operational only with the continuous energy mode, and it provides the specifications for reaction rate, neutron flux, and reaction cross section tallies. See Sect. 8.1.7.6 for more details. For multigroup KENO calculations, use KMART5 or KMART6, which are described in the KMART section of the SCALE manual.

A reaction data block consists of *REACTION FILTERS*, *TALLY TYPE*, *ENERGY GROUP BOUNDARIES*, and *OUTPUT EDITS*. These data types can be entered in any order. A combination of parameters for describing the *REACTION FILTERS* and *TALLY TYPE* must be entered for any reaction or cross section tally calculation. *ENERGY GROUP BOUNDARIES* and *OUTPUT EDITS* data are optional. Tally calculations can be performed for multiple reactions specified by the *REACTION FILTERS*. Only one energy grid, either specified with the data in *ENERGY GROUP BOUNDARIES* or from the READ ENERGY block or from the code defaults, is used for all reaction tally calculations. To provide data for the continuous energy depletion calculations, another energy grid can be specified and used for tallying only the mixture flux.

Enter *REACTION DATA* in the form:

READ REACTION *REACTION FILTERS [TALLY TYPE] [ENERGY GROUP BOUNDARIES][OUTPUT EDITS]* END REACTION

REACTION FILTERS define a reaction map that is used in reaction tally calculations. The *REACTION FILTERS* must be entered in the following order; mixture data (MIX or MIXLIST) followed by nuclide data (NUC or NUCLIST) followed by reaction IDs (MT or MTLIST). Each filter is defined using a combination of the following keywords:

MIX=*mixnum* Mixture number, no default value. Specified mixture number must exist in the mixing table and be used in the problem for a valid filter generation. A wildcard * can be used to define a filter applicable for all mixtures in the problem.

MIXLIST ***mixnum*$_1$ *mixnum*$_2$ ... *mixnum*$_N$** **END** A list of mixture numbers followed by *end,* no default values. Specified mixture numbers must exist in the mixing table and be used in the problem for a valid reaction tally calculation. Within each filter, use either MIX or MIXLIST, but not both.

NUC=*nucid* Nuclide identifier, no default value. Specified nuclide must be a constituent of the mixtures used in this filter definition (specified with MIX or MIXLIST). A wildcard "*" can be used to define a filter applicable for all nuclides in each mixture in this filter definition. Nuclide identifiers are listed for all isotopes in the Standard Composition Library section of the SCALE manual.

NUCLIST ***nucid*$_1$ *nucid*$_2$ ... *nucid*$_N$** **END** A list of nuclide identifiers followed by *end,* no default values. Specified nuclides must be the constituents of the mixtures used in this filter definition (specified by MIX or MIXLIST). Within each filter, use either NUC or NUCLIST, but not both.

MT=*mt* Reaction MT number, no default value. Specified reaction MT number should be available for the nuclides defined in this filter definition (specified by NUC or NUCLIST). Otherwise, the code skips the filter definition with this given reaction MT. A wildcard "*" can be used to define a filter with all reaction MTs. Valid SCALE library MT values are listed in the MT Reaction Types on SCALE Cross-Section Libraries.

MTLIST ***mt*$_1$ *mt*$_2$ ... *mt*$_N$** **END** A list of reaction MT numbers followed by *end,* no default values. Specified MT numbers should be available for the nuclides defined in this filter definition (specified by MIX or MIXLIST). Otherwise, KENO skips that reaction specified in the filter for the reaction tally calculations. Within each filter, use either MT or MTLIST, but not both.

A reaction filter consists of either single or multiple mixture, nuclide and reaction definitions. A valid reaction filter starts with mixture specification, followed by nuclide specification, and ends with reaction specification. Mixture(s) must be specified with either MIX or MIXLIST keywords. Nuclide(s) in these mixtures must be entered with either NUC or NUCLIST, and reactions for each nuclide must be specified with either MT or MTLIST.

Mixture, nuclide, and reaction number are required for mixture average fluxes, even though the nuclide and reaction numbers are not used for the neutron flux tallies.

Multiple reaction filter definitions are allowed. KENO processes all the definitions and creates a reaction map based on them. The following examples demonstrate the reaction filter specifications for different problems. In these examples, reaction filters are specified based on the following composition data used in the problem:

compositions in the example problem

mixture	nuclides
10	92235, 92238, 8016
20	92238, 94239, 8016
30	92235, 92238, 8016
40	1001, 8016
100	1001, 8016, 5010, 5011

Example 8.1.1: Defines a reaction filter used to tally only fission reaction (MT=18) of ^{235}U in mixture 10.

```
READ REACTION
   MIX=10 NUC=92235  MT=18
   ...
END REACTION
```

Example 8.1.2: Defines a reaction filter used to tally all available reactions of ^{235}U in mixture 10

```
READ REACTION
   MIX=10 NUC=92235  MT=*
   ...
END REACTION
```

Example 8.1.3: Defines a reaction filter used to tally the elastic scattering (mt=2), fission (mt=18), and capture (mt=102) reactions of ^{235}U in mixture 10

```
READ REACTION
   MIX=10 NUC=92235  MTLIST 2 18 102 END
   ...
END REACTION
```

Example 8.1.4: Defines a reaction filter used to tally the elastic scattering (`MT=2`), fission (`MT=18`), and capture (`MT=102`) reactions of ^{235}U in mixture 10. Reaction filter definition in this example is identical to the filter definition given in Example 8.1.3.

```
READ REACTION
   MIX=10 NUC=92235  MT=2
   MIX=10 NUC=92235  MT=18
   MIX=10 NUC=92235  MT=102
   ...
END REACTION
```

Example 8.1.5: Defines a reaction filter used to tally the fission reaction (`MT=18`) of ^{235}U and ^{238}U in mixture 10. Code skips the reaction tally request for ^{16}O since the requested reaction is not available for this nuclide in the data library.

```
READ REACTION
   MIX=10 NUC=*  MT=18
   ...
END REACTION
```

Example 8.1.6: Defines a reaction filter used to tally the fission reaction (`MT=18`) of ^{235}U and ^{238}U in mixture 10. Reaction filter definition in this example is identical to the filter definition given in Example 8.1.5.

```
READ REACTION
   MIX=10 NUC=92235  MT=18
   MIX=10 NUC=92238  MT=18
   ...
END REACTION
```

Example 8.1.7: A reaction filter used to tally the capture reaction (`MT=102`) of ^{16}O in all mixtures.

```
READ REACTION
   MIX=* NUC=8016  MT=102
   ...
END REACTION
```

Example 8.1.8: Areaction filter used to tally the capture reaction (MT=102) of ^{16}O in mixtures 10, 20, 30, 40, and 100 respectively. Reaction filter definition in this example is identical to the filter definition given in Example 8.1.7.

```
READ REACTION
     MIX=10  NUC=8016  MT=102
     MIX=20  NUC=8016  MT=102
     MIX=30  NUC=8016  MT=102
     MIX=40  NUC=8016  MT=102
     MIX=100 NUC=8016  MT=102
      ...
END REACTION
```

Example 8.1.9: A reaction filter used to tally the capture reaction (MT=102) of ^{16}O in mixtures 10, 20, 30, 40, and 100 respectively. Reaction filter definition in this example is identical to the filter definition given in Example 8.1.7 and Example 8.1.8.

```
READ REACTION
   MIXLIST 10 20 30 40 100 END NUC=8016  MT=102
   ...
END REACTION
```

Example 8.1.10: Complex reaction tally for U-238, Pu-239, H-1, and O-16 across several mixtures.

```
READ REACTION
     MIXLIST 10 20 30 END NUC=92238  MT=102
     MIX=20  NUC=94239 MT=18
     MIX=40  NUC=1001  MT=*
     MIX=*   NUC=8016  MT=2
     MIX=*   NUC=*     MT=27
      ...
END REACTION
```

Example 8.1.10 defines a complex reaction filter used to tally:

a) Capture reaction (**MT**=102) of ^{238}U in mixtures 10, 20 and 30 respectively,

b) Fission reaction (**MT**=18) of ^{239}Pu in mixture 20,

c) All reactions of ^{1}H in mixture 40,

d) Elastic scattering reaction of ^{16}O in all mixtures,

e) Total absorption reaction of all nuclides in all mixtures.

Parameters of TALLY TYPE are logical parameters used to select quantities (reaction cross section, reaction rate, and mixture flux) that are tallied for the given problem. The user specifies any combination of these TALLY TYPEs once for all filters:

XSTALLY=*lCEXSTally* Enter YES or NO. A value of YES specifies that reaction cross sections be tallied for the reactions listed in *REACTION FILTERS*. The default value of XSTALLY is NO. Computed reaction cross sections are saved in a file named *BASENAME_keno_micro_xs.0* in RTNDIR, which is a SCALE environment variable for the directory from where the calculation was started. BASENAME is a SCALE environment variable that is the base name of the input file. (BASENAME is equal to *"mytest"* if the SCALE input name is *"mytest.inp."*)

RRTALLY=*lCERRTally* Enter YES or NO. A value of YES specifies that reaction rates be tallied for the reactions listed in *REACTION FILTERS*. The default value of RRTALLY is NO. Computed reaction rates are saved in a file named *BASENAME_keno_micro_rr.0* in RTNDIR.

Note: KENO combines and saves reaction rate and reaction cross section tallies to the same file, named *BASENAME_keno_micro_xs_rr.0* in RTNDIR, if both `XSTALLY` and `RRTALLY` are set to YES.

MIXFLX=*lCEMixFlux* Enter YES or NO. A value of YES specifies that mixture fluxes are to be tallied for the mixtures listed in *REACTION FILTERS*. The default value of `MIXFLX` is NO. Computed mixture fluxes are saved in a file named *BASENAME_keno_mixture_flux.0* in RTNDIR.

Mixture, nuclide, and reaction number are required for mixture average fluxes, even though the nuclide and reaction numbers are not used for the neutron flux tallies.

Example 8.1.11: Defines a reaction filter, which uses fission reaction (`MT=18`) of ^{235}U in mixture 10, for tallying reaction cross sections.

```
READ REACTION
   MIX=10 NUC=92235  MT=18
   XSTALLY=YES
   ...
END REACTION
```

Note: Computed data are saved in a file named *BASENAME_keno_micro_xs.0*

Example 8.1.12: Defines a reaction filter, which uses fission reaction (`MT=18`) of ^{235}U in mixture 10, for tallying reaction rates as well as the reaction cross sections.

```
READ REACTION
   MIX=10 NUC=92235  MT=18
   XSTALLY=YES RRTALLY=YES
   ...
END REACTION
```

Note: Computed data are saved in files named *BASENAME_keno_micro_xs_rr.0*

Example 8.1.13: Defines a reaction filter, which uses fission reaction (`MT=18`) of ^{235}U in mixture 10, for tallying reaction cross sections. In addition, mixture flux is tallied for mixture 10 given in this reaction filter.

```
READ REACTION
   MIX=10 NUC=92235  MT=18
   XSTALLY=YES MIXFLX=YES
   ...
END REACTION
```

Note: Computed data are saved in files named *BASENAME_keno_micro_xs.0*, and *BASENAME_keno_mixture_flux.0*, respectively.

ENERGY GROUP BOUNDARIES data define energy group structure other than the defaults for tallying both reaction cross sections/reaction rates and mixture fluxes.

ENER_XS *e1 e2 e3 . . .* END Upper energy boundary for each group. The last entry is the lower energy boundary of the last group. For N groups, there are N+1 entries. Entries must be in descending order. This may be specified once in the REACTION block and, if used, is applied to all cross section and reaction rate tallies.

ENER_FLX *e1 e2 e3 . . .* END Upper energy boundary for each group, default is NGP-group data. The last entry is the lower energy boundary of the last group. For N groups, there are N +1 entries. Entries must be in descending order. This may be specified once in the REACTION block and, if used, is applied to all mixture flux tallies.

Note: Default values for the energy group boundaries in reaction tally calculations are determined as in the order described in Sect. 8.1.3.12.

Example 8.1.14: No READ ENERGY block, no NGP in READ PARAMETER block, no energy group

```
READ REACTION
   MIX=10 NUCLIST 92235 92238 END
   MTLIST 16 17 18 END
   MIXFLX=YES XSTALLY=YES
   ...
END REACTION
```

Default SCALE 252-group energy structure is used for tallying both mixture flux and reaction cross sections.

Example 8.1.15: Energy group bounds specified in READ ENERGY block. 2-group energy structure given in READ ENERGY block is used for tallying both mixture flux and reaction cross sections.

```
READ ENERGY
   20.E6   0.6 1.E-4
END ENERGY
...
READ REACTION
   MIX=10 NUCLIST 92235 92238 END
   MTLIST 16 17 18 END
   MIXFLX=YES XSTALLY=YES
   ...
END REACTION
```

Example 8.1.16: NGP is set in READ PARAMETER block; 4-group energy group structure (4-equal lethargy bins) is used for tallying both mixture flux and reaction cross sections.

```
READ PARAMETER
   ...
   NGP=4
END PARAMETER
...
READ REACTION
   MIX=10 NUCLIST 92235 92238 END
   MTLIST 16 17 18 END
   MIXFLX=YES XSTALLY=YES
   ...
END REACTION
```

Example 8.1.17: 3-group energy group structure given in `reaction data` block is used for tallying reaction cross sections, and default SCALE 252-group energy structure is used for tallying mixture flux.

```
READ REACTION
     MIX=10 NUCLIST 92235 92238 END
     MTLIST 16 17 18 END
     MIXFLX=YES XSTALLY=YES
     ENER_XS 20.E6 1.E3 1.0 1.E-4 END
      ...
END REACTION
```

Example 8.1.18: 2-group energy group structure given in `reaction data` block is used for tallying reaction cross sections, and 3-group energy group structure given in `reaction data` block is used for tallying mixture flux.

```
READ REACTION
     MIX=10 NUCLIST 92235 92238 END
     MTLIST 16 17 18 END
     MIXFLX=YES XSTALLY=YES
     ENER_FLX 20.E6 1.E3 1.0 1.E-4 END
     ENER_XS 20.E6 1.0 1.E-4 END
      ...
END REACTION
```

Parameters of *OUTPUT EDITS* are logical parameters used to print reaction tallies and mixture fluxes in separate files. These parameters are optional parameters.

`PRNTXS`=*lCEprintXS* Enter YES or NO. A value of YES specifies that reaction cross sections tallies for each mixture be written in separate files in RTNDIR (*BASENAME_keno_micro_xs_mix{mixnum}.0, mixnum* is the mixture numbers specified in the reaction filters). The default value of `PRNTXS` is NO.

`PRNTRR`=*lCEprintRR* Enter YES or NO. A value of YES specifies that reaction rate tallies for each mixture be written in separate files in RTNDIR (*BASENAME_keno_micro_rr_mix{mixnum}.0, mixnum* is the mixture numbers specified in the reaction filters). The default value of `PRNTRR` is NO.

`PRNTFLX`=*lCEprintMixFlux* Enter YES or NO. A value of YES specifies that mixture flux tallies for each mixture be written in separate files in RTNDIR (*BASENAME_keno_mixture_flux_mix{mixnum}.0, mixnum* is the mixture numbers specified in the reaction filters). The default value of `PRNTFLX` is NO.

8.1.4 NOTES FOR KENO USERS

This section provides assorted tips designed to assist the KENO user with problem mockups. Some information concerning methods used by KENO is also included.

8.1.4.1 Data entry

The KENO data is entered in blocks that begin and end with keywords as described in Sect. 8.1.3.1. Only one set of parameter data can be entered for a problem. However, for other data blocks, it is possible to enter more than one block of the same kind of data. When this is done, only the last block of that kind of data is retained for use by the problem, except for the GRID block for which all blocks are retained.

Within data blocks, a number, x, can be repeated n times by specifying nRx, n*x, or n$x.

Numbers in engineering notation may be specified with or without an “E” between the base and the exponent. For example; 0.0011 may be specified as 1.1e-3 or as 1.1-3.

Multiple and scattered entries in the mixing table

In the following examples, assume 1001 is the nuclide ID for hydrogen, 8016 is the nuclide ID for oxygen, 92235 is the nuclide ID for ^{235}U, and 92238 is the nuclide ID for ^{238}U. If a given nuclide ID is used more than once in the same mixture, the result is the summing of all the number densities associated with that nuclide. For example:

```
MIX =1 92235 4.3e-2 92238 2.6e-3 1001 3.7e-2 92235 1.1e-3 8016 1.8e-2
```

would be the same as entering:

```
MIX =1 92235 4.41e-2 92238 2.6e-3 1001 3.7e-2 8016 1.8e-2
```

A belated entry for a mixture can be made as follows:

```
MIX =1 1001 6.6e-2 MIX=2 92235 4.3e-2 92238 2.6e-3 MIX=1 8016 3.3e-2
```

This is the same as entering:

```
MIX =1 1001 6.6e-2 8016 3.3e-2 MIX=2 92235 4.3e-2 92238 2.6e-3
```

Multiple entries in geometry data

Individual geometry regions cannot be replaced by adding an additional description. However, entire unit descriptions can be replaced by adding a new description having the same unit number. The last description entered for a unit is used in the calculation. For example, the following geometry descriptions are equivalent in KENO V.a and KENO-VI, respectively:

In KENO V.a:

```
READ GEOM  UNIT 1 SPHERE 1 1 5.0 CUBE 0 1 10.0 -10.0
UNIT 2 CYLINDER 1 1 2.0 5.0 -5.0  CUBE  0 1 10.0 -10.0
UNIT 1  CUBOID  1 1 1.0 -1.5 2.5 -2.0 5.0 -6.0  CUBE  0 1 10.0 -10.0
END GEOM
```

is the same as entering:

```
READ GEOM  UNIT 1  CUBOID 1 1 1.0 -1.5 2.5 -2.0 5.0 -6.0
CUBE  0 1 10.0 -10.0
UNIT 2  CYLINDER  1 1 2.0 5.0 -5.0  CUBE  0 1 10.0 -10.0 END GEOM
```

or

```
READ GEOM  UNIT 2 CYLINDER  1 1 2.0 5.0 -5.0  CUBE  0 1 10.0 -10.0
UNIT 1  CUBOID  1 1 1.0 -1.5 2.5 -2.0 5.0 -6.0  CUBE  0 1 10.0 -10.0
END GEOM
```

In KENO-VI:

```
READ GEOM
UNIT 1 SPHERE 10 5.0
CUBOID 20 10.0 -10.0 10.0 -10.0 10.0 -10.0
MEDIA 1 1 10
MEDIA 0 1 20 -10
BOUNDARY  20
UNIT 2
CYLINDER  10  2.0 5.0 -5.0
CUBOID  20  10.0 -10.0 10.0 -10.0 10.0 -10.0
MEDIA 1 1 10
MEDIA 0 1 20 -10
```

(continues on next page)

(continued from previous page)

```
BOUNDARY  20
UNIT 1
CUBOID  10   1.0  -1.5  2.5  -2.0  5.0  -6.0
CUBOID  20  10.0 -10.0 10.0 -10.0 10.0 -10.0
MEDIA 1 1 10
MEDIA 0 1 -10 20
BOUNDARY  20
END GEOM
```

is the same as entering

```
READ GEOM
UNIT 1
CUBOID  10   1.0  -1.5  2.5  -2.0  5.0  -6.0
CUBOID  20  10.0 -10.0 10.0 -10.0 10.0 -10.0
MEDIA 1 1 10
BOUNDARY  20
MEDIA 0 1 -10  20
UNIT 2
CYLINDER  10   2.0   5.0 -5.0
CUBOID    20  10.0 -10.0 10.0 -10 10.0 -10.0
MEDIA 1 1 10
MEDIA 0 1 -10 20
BOUNDARY  20
END GEOM
```

or

```
READ GEOM
UNIT 2
CYLINDER  30  2.0  5.0 -5.0
CUBOID    40  6P10.0
MEDIA 1 1 30
MEDIA 0 1 -30  40
BOUNDARY  40
UNIT 1
CUBOID  20  1.0 -1.5 2.5 -2.0 5.0 -6.0
CUBOID  10  6P10.0
MEDIA 1 1 20
MEDIA 0 1 10 -20
BOUNDARY  10
END GEOM
```

The order of entry for UNIT descriptions is not important because the UNIT number is assigned as the value following the word UNIT. They do not need to be entered sequentially, and they do not need to be numbered sequentially. It is perfectly acceptable to enter UNITs 2, 3, and 5, omitting Units 1 and 4 as long as UNITs 1 and 4 are not referenced in the problem. It is also acceptable to scramble the order of entry as in entering UNITs 3, 2, and 5.

8.1.4.2 Default logical unit numbers for KENO

The logical unit numbers for data used by KENO are listed in Table 8.1.29.

Table 8.1.29: KENO logical unit numbers

Function	Parameter name	Unit number	Variable name

continues on next page

Table 8.1.29 – continued from previous page

Problem input data (ASCII)		5	INPT
Problem input data (binary)		95	BIN
Program output (ASCII)		6	OUTPT
Albedo data	ALB=	79	ALBDO
Scratch unit	SKT=	16	SKRT
Read restart data	RST=	0[a]	RSTRT
		34[b]	RSTRT
Write restart data	WRS=	0[a]	WSTRT
		35[c]	WSTRT
Direct access storage for input data		8	DIRECT(1)
Direct access storage for super-grouped data		9	DIRECT(2)
Direct access storage for cross section mixing		10	DIRECT(3)
Mixed cross section data set	XSC=	14[d]	ICEXS
Group-dependent weights	WTS=	80	WTS
AMPX working format cross sections	LIB=	0[a]	AMPXS
Group boundary Library (KENO-VI)	GRP=	77	GRPBS
[a] Defaulted to zero [b] Defaulted to 34 if BEG= a number greater than 1 and RSTRT=0. c Defaulted to 35 if ``RES=` a number greater than zero and WSTRT=0. [d] Defaulted to 0; if LIB= a number greater than zero, ICEXS is defaulted to 14.			

8.1.4.3 Parameter input

When the parameter data block is entered for a problem, the same keyword may be entered several times. The last value that is entered is used in the problem. Data may be entered as follows:

```
READ PARAM  FLX=YES NPG=1000 TME=0.5 TME=1.0
NPG=50 TME=10.0 FLX=NO
NPG=500
END PARA
```

This will result in the problem having FLX=NO, TME=10.0, and NPG=500. It may be more convenient for the user to insert a new value than to change the existing data.

Certain parameter default values should not be overridden unless the user has a very good reason to do so. These parameters are as follows:

1. X1D= which defines the number of extra 1-D cross sections. The use of extra 1-D cross sections-other than the use of the fission cross section for calculating the average number of neutrons per fission-requires programming changes to the code;

2. NFB= which defines the number of neutrons that can be entered in the fission bank (the fission bank is where the information related to a fission is stored);

3. XFB= which defines the number of extra positions in the fission bank;

4. NBK= which defines the number of neutrons that can be entered in the neutron bank (the neutron bank contains information about each history);

5. XNB= which defines the number of extra positions in the neutron bank;

6. WTH= which defines the factor that determines when splitting occurs;

7. WTA= which defines the default average weight given to a neutron that survives Russian roulette;

8. WTL= which defines the factor that determines when Russian roulette is played; and

9. LNG= which sets the maximum words of storage available to the program.

It is recommended that BUG=, the flag for printing debug information, never be set to YES. The user would have to look at the FORTRAN coding to determine what information is printed. BUG=YES prints massive amounts of sparsely labeled information. The user should only rarely consider using TRK=YES. This generates thousands of lines of well-labeled output that provides information about each history at key locations during the tracking procedure. All other parameters can be changed at will to provide features the user wishes to activate.

8.1.4.4 Cross sections

In multi-energy group mode, KENO always uses cross sections from a mixed cross section data file. The format of this file is the Monte Carlo processed cross section file. A mixed cross section file can be created by previous KENO run, or by using an AMPX working format library and entering mixing table data in KENO.

Use a mixed cross section Monte Carlo format library

A mixed cross section Monte Carlo format library (premixed cross section data file) from a previous KENO case may be used. This file is specified using the parameter XSC=. If a mixing table data block is entered, the premixed cross section data file will be rewritten. Therefore, a mixing table should not be entered if a premixed cross section data file is used. The user should verify that the mixtures created by a previous KENO case are consistent with those used in the geometry data of the problem.

Use an AMPX working format library

When an AMPX working format library is used, it must be specified using the parameter LIB=, and mixing table data must always be entered. IDs used in the mixing table must match the IDs on the AMPX working format library. The user must provide a file with the correct cross sections with a name that matches the pattern ftNNf001, with NN being the number of the logical unit.

Number of scattering angles

The number of scattering angles is defaulted to 1 (defaulted to 2 when KENO-VI is run as part of the CSAS6 sequence). This stand alone default is not adequate for many applications. The user should specify the scattering angle to be consistent with the cross sections being used. The number of scattering angles is entered in the cross section mixing table by using the keyword SCT=. See Sect. 8.1.3.10.

The order of the last Legendre coefficient to be preserved in the scattering distribution is equal to ($2 \times$ SCT - 1). SCT=1 could be used with a P_1 cross section set such as the 16-group Hansen-Roach cross section library, and SCT=2, for a P_3 cross section set such as the SCALE 27-group cross section library. ENDF/B-VII cross section libraries such as the 44-group or 252-group libraries contain many nuclides having P_5 cross section sets. Isotropic scattering is achieved by entering SCT=0.

Cross section message cutoff

The cross section message cutoff value, *pbxs*, is defaulted to 3×10^{-5}. Warning messages generated when errors are encountered in the P_L expansion of the group-to-group transfers will be suppressed if the P_0 cross section for that particular energy transfer is less than *pbxs*. The value of *pbxs* is specified in the cross section mixing table by using the keyword EPS=. See Sect. 8.1.3.10.

The default value of *pbxs* is sufficient to assure that warning messages will not be printed for most of the SCALE P_1 and P_3 cross section libraries. However, the v7.0-n library may print a few errors if P_5 cross sections are specified.

The error messages below were printed for a problem using the 238-group cross section library and *pbxs* = 3.0×10^{-5}. If the default value of *pbxs* allows too many warning messages to be printed, a value can be determined which does not print the error messages from the printed messages by choosing a number larger than the P_0 component on the first line, as shown below.

THE LEGENDRE EXPANSION OF THE CROSS SECTION (P_0-P_N) IS P_0 P_1 P_2 ... P_n

THE MOMENTS CORRESPONDING TO THIS DISTRIBUTION ARE M_1 M_2 ... M_n

THE MOMENTS CORRESPONDING TO THE GENERATED DISTRIBUTION ARE M_1 M_2 ... M_n

THE LEGENDRE EXPANSION CORRESPONDING TO THESE MOMENTS IS P_0 P_1 P_2 ... P_n

___ MOMENTS WERE ACCEPTED

For the following messages, EPS=6.9e-5 would cause all three messages to be suppressed. A value less than 5.615159e-5 and greater than 4.767635e-5 would suppress the second message, and a value less than 6.855362e-5 and greater than 5.615159e-5 would suppress the first two messages.

```
KMSG060       THE ANGULAR SCATTERING DISTRIBUTION FOR MIXTURE 2 HAS BAD MOMENTS FOR THE TRANSFER FROM GROUP 28␣
↪TO GROUP 72
                  1 MOMENTS WERE ACCEPTED
                THE LEGENDRE EXPANSION OF THE CROSS SECTION (P0-PN) IS
           5.615159E-05   1.155527E-06  -2.804013E-05  -1.732067E-06
THE MOMENTS CORRESPONDING TO THIS DISTRIBUTION ARE
           2.057870E-02   4.234578E-04   8.710817E-06
THE MOMENTS CORRESPONDING TO THE GENERATED DISTRIBUTION ARE
           2.057870E-02   4.235078E-04   8.710817E-06
THE LEGENDRE EXPANSION CORRESPONDING TO THESE MOMENTS IS
           5.615159E-05   1.155527E-06  -2.804011E-05  -1.732066E-06
                THE WEIGHTS/ANGLES FOR THIS DISTRIBUTION ARE
           9.999995E-01   5.268617E-07
           2.057881E-02  -1.973451E-01
THE MOMENTS CORRESPONDING TO THIS DISTRIBUTION ARE
           2.057870E-02   4.235078E-04   8.710817E-06


KMSG060       THE ANGULAR SCATTERING DISTRIBUTION FOR MIXTURE 2 HAS BAD MOMENTS FOR THE TRANSFER FROM GROUP 31␣
↪TO GROUP 75
                  1 MOMENTS WERE ACCEPTED
                THE LEGENDRE EXPANSION OF THE CROSS SECTION (P0-PN) IS
           4.767635E-05   7.834378E-07  -2.381887E-05  -1.174626E-06
                THE MOMENTS CORRESPONDING TO THIS DISTRIBUTION ARE
           1.643242E-02   2.700205E-04   4.451724E-06
                THE MOMENTS CORRESPONDING TO THE GENERATED DISTRIBUTION ARE
           1.643242E-02   2.700282E-04   4.437279E-06
                THE LEGENDRE EXPANSION CORRESPONDING TO THESE MOMENTS IS
           4.767635E-05   7.834378E-07  -2.381885E-05  -1.174627E-06
                THE WEIGHTS/ANGLES FOR THIS DISTRIBUTION ARE
           9.999858E-01   1.420136E-05
```

(continues on next page)

(continued from previous page)

```
           1.643265E-02  -2.334324E-07
               THE MOMENTS CORRESPONDING TO THIS DISTRIBUTION ARE
           1.643242E-02   2.700282E-04   4.437279E-06


KMSG060       THE ANGULAR SCATTERING DISTRIBUTION FOR MIXTURE 2 HAS BAD MOMENTS FOR THE TRANSFER FROM GROUP
32 TO GROUP 74                                                                  (1)
                 1 MOMENTS WERE ACCEPTED                                       (2)
               THE LEGENDRE EXPANSION OF THE CROSS SECTION (P0-PN) IS            (3)
            6.855362E-05   1.341944E-06  -3.423741E-05  -2.011613E-06            (4)
               THE MOMENTS CORRESPONDING TO THIS DISTRIBUTION ARE                (5)
            1.957510E-02   3.831484E-04   7.601939E-06                          (6)
               THE MOMENTS CORRESPONDING TO THE GENERATED DISTRIBUTION ARE (7)
            1.957510E-02   3.832207E-04   7.502292E-06                          (8)
               THE LEGENDRE EXPANSION CORRESPONDING TO THESE MOMENTS IS      (9)
            6.855362E-05   1.341944E-06  -3.423740E-05  -2.011629E-06           (10)
               THE WEIGHTS/ANGLES FOR THIS DISTRIBUTION ARE                     (11)
            9.999056E-01   9.437981E-05                                       (12)
            1.957695E-02  -1.848551E-06                                        (13)
               THE MOMENTS CORRESPONDING TO THIS DISTRIBUTION ARE               (14)
                1.957510E-02   3.832207E-04   7.502292E-06                     (15)
```

The user does not need to attempt to suppress all these messages. They are printed to inform the user of the fact that the moments of the angular distribution are not moments of a valid probability distribution. The original P_n coefficients and their moments are listed in lines 3–6 of the message. Lines 7–10 list the new corrected moments and their corresponding P_n coefficients.

The weights and angles printed in lines 11–13 were generated from the corrected moments. The last two lines of the message list the moments generated from those weights and angles. They should match line 8, which lists the moments corresponding to the generated distribution.

For most criticality problems, the first moment contributions are much more significant than the contributions of the higher order moments. Thus, the higher order moments may not affect the results significantly. The user may compare the original moments and corrected moments to make a judgment as to the significance of the change in the moments.

8.1.4.5 Mixing table

Mixtures can be used in defining other mixtures. When defining mixture numbers, care should be taken to avoid using a mixture number that is identical to a nuclide ID number if the mixture is to be used in defining another mixture. If a mixture number is defined more than once, it results in a summing effect.

The nuclide mixing loop is done before the mixture mixing loop, which performs mixing in the order of data entry. Thus, the order of mixing mixtures into other mixtures is important because a mixture must be defined before it can be used in another mixture. Some examples of correct and incorrect mixing are shown below, using 1001 as the nuclide ID for hydrogen, 8016 as the nuclide ID for oxygen, 92235 as the nuclide ID for ^{235}U, and 92238 as the nuclide ID for ^{238}U.

EXAMPLES OF CORRECT USAGE

```
READ MIXT
  MIX=1 1001 6.6e-2  8016 3.3e-2
  MIX=2 1 0.5
END MIXT
```

This results in mixture 1 being full-density water and mixture 2 being half-density water.

```
READ MIXT
  MIX=1  2  0.5
  MIX=3  1  0.5
  MIX=2  1001  6.6e-2  8016  3.3e-2
END MIXT
```

This results in mixture 1 being half-density water, mixture 2 being full-density water, and mixture 3 being quarter-density water. Because the nuclide mixing loop is done first, mixture 2 is created first and is available to create mixture 1, which is then available to create mixture 3.

```
READ MIXT
   MIX=1   1001   6.6e-2   8016   3.3e-2
   MIX=2   92235   7.5e-4   92238   2.3e-2  8016   4.6e-2  1 .01
END MIXT
```

This results in mixture 1 being full-density water and mixture 2 being uranium oxide containing 0.01 density water.

```
READ MIXT
   MIX=1    1001  6.6e-2    8016  3.3e-2
   MIX=2  92235  4.4e-2  92238  2.6e-3
   MIX=1  1  0.5
END MIXT
```

This results in mixture 1 being water at 1.5 density (1001 9.9e-2 and 8016 4.95e-2) and mixture 2 is highly enriched uranium metal.

EXAMPLES OF INCORRECT USAGE

```
READ MIXT
  MIX=3  1  0.75
  MIX=1  2  0.5
  MIX=2  1001  6.6e-2  8016  3.3e-2
END MIXT
```

Here the intent is for mixture 2 to be full-density water, mixture 1 to be half-density water, and mixture 3 to be 3/8 (0.75×0.5) density water. Instead, the result for mixture 3 is a void, mixture 1 is half-density water, and mixture 2 is full-density water. This is because the nuclide mixing loop is done first, thus defining mixture 2. The mixture mixing loop is done next. Mixture 3 is defined to be mixture 1 multiplied by 0.75, but since mixture 1 has not been defined yet, 0.75 of zero is zero. Mixture 1 is then defined to be mixture 2 multiplied by 0.5. If the definition of mixture 1 preceded the definition of mixture 3, as in item (2) under examples of correct usage, it would work correctly.

```
READ MIXT
    MIX=1         1001   6.6e-2     8016     3.3e-2
    MIX=1001 92235   4.4e-2   92238   2.6e-3
    MIX=2   1001   0.5
END MIXT
```

This results in mixture 1 being full-density water, mixture 1001 being uranium metal, and mixture 2 being hydrogen with a number density of 0.5 because 1001 is the nuclide ID number for hydrogen. When a mixture number is identical to a nuclide ID and is used in mixing, that number is assumed to be a nuclide ID rather than a mixture number. The intent was for mixture 1 to be full-density water, mixture 1001 to be uranium metal, and mixture 2 to be half-density uranium metal.

8.1.4.6 Geometry Considerations

In general, KENO geometry descriptions consist of (1) geometry data (Sect. 8.1.3.4) defining the geometrical shapes present in the problem, and (2) array data (Sect. 8.1.3.5) defining the placement of the units that were defined in the geometry data. The geometry data block is prefaced by READ GEOM, and the array data block is prefaced by READ ARRAY.

When a 3-D geometrical configuration is described as KENO geometry data, it may be necessary to describe portions of the configuration individually. These individual partial descriptions of the configuration are called UNITs. KENO geometry modeling is subject to the following restrictions:

1. **Units are composed of regions. These regions are created using** geometric bodies and surfaces (shapes) that are previously defined.
    1. In KENO-VI the geometric bodies and surfaces may intersect. Regions are defined relative to the geometric bodies and surfaces in a UNIT. HOLEs provide a means of creating complex geometries in a UNIT and then inserting the UNIT into existing UNITs. For complex geometries the use of HOLEs may decrease the CPU time required for the problem.
    2. In KENO V.a, each geometry region in a UNIT must completely enclose all geometry regions which precede it. Boundaries of the surfaces of the regions may be shared or tangent, but they must not intersect. The use of HOLEs in KENO V.a provides an exception to this complete enclosure rule. The use of HOLEs in KENO V.a will increase the CPU time required for the problem.
2. **All geometrical surfaces must be describable in KENO V.a as spheres,** hemispheres, cylinders, hemicylinders, cubes, cuboids, or as a set of quadratic equations in KENO-VI.
3. **When specifying an ARRAY, each UNIT used in a cuboidal** ARRAY must have a CUBE or CUBOID as its outer region (this is the only option in KENO V.a); a hexprism as the outer boundary if it is a hexagonal, triangular, or shexagonal ARRAY; a rhexprism as the outer boundary if it is a rhexagonal ARRAY; and a dodecahedron as the outer boundary if it is a dodecahedral ARRAY. In addition, the outer boundary of a UNIT cannot be rotated or translated in KENO V.a.
4. **When several UNITs are used to describe an ARRAY, the** adjacent faces of the UNITs in contact with each other must be the same size and shape.
5. **UNITs are placed directly into regions using HOLEs. As** many HOLEs as will fit without intersecting other HOLEs, nested ARRAYs or HOLEs, or the UNIT BOUNDARY can be placed in a UNIT without intersecting each other. In KENO V.a, HOLEs cannot intersect any of the regions within the UNIT in which they are placed. HOLEs are described in more detail in Sect. 8.1.4.6.1, and nested HOLEs are described in Sect. 8.1.4.6.2.
6. **Multiple ARRAYs may be required to describe a complicated** system. In KENO V.a, only one ARRAY may be placed directly into a UNIT. However, multiple ARRAYs may be placed in a UNIT by placing the ARRAYs in other UNITs and placing those UNITs in the original UNIT using HOLEs. Multiple ARRAYs may be placed directly into a UNIT in KENO-VI. These UNITs may then be used to create other ARRAYs, or they may be placed in other UNITs using HOLEs. UNITs placed in ARRAYs or HOLEs that are contained in other HOLEs or ARRAYs are said to be nested. The nesting level of a UNIT is the number of ARRAYs and HOLEs between the ARRAY or HOLE in the GLOBAL UNIT or ARRAY and the UNIT. Multiple ARRAYs are described in more detail in Sect. 8.1.4.6.3.

The KENO V.a geometry package allows any applicable shape to be enclosed by any other applicable shape, subject only to the complete enclosure restriction. The KENO-VI geometry package allows any shape describable using quadratic equations to be enclosed or intersected by any other allowable shape. The implication of this type of description is that the entire volume between two adjacent geometrical surfaces contains only one mixture, HOLE, or ARRAY. A void is specified by a mixture ID of zero. If HOLEs are present in the volume between two surfaces, the volume of that region is reduced by the HOLE's volume(s).

In KENO V.a geometry, if the problem requires several UNITs to describe its geometrical characteristics, each UNIT that is used in an ARRAY must have a rectangular parallelepiped as its outer surface. This restriction is relaxed in KENO-VI, where the outer surface may be a rectangular parallelepiped, a hexagonal prism, a 90° rotated hexagonal prism, or a dodecahedron, but all units placed in an array must use the same shape as their outer BOUNDARY. In order to describe the composite overall geometrical characteristics of the problem, these UNITs may be arranged in a rectangular ARRAY for KENO V.a geometry, or in either a rectangular, hexagonal, shexagaonal, rhexagonal, or dodecahedral ARRAY for KENO-VI geometry. This is done by specifying the number of units in the X, Y, and Z directions. If more than one UNIT is involved, data must be entered to define the number assigned to the ARRAY and the placement of the individual UNITs in the ARRAY. The ARRAY number, the number of UNITs in the X, Y, and Z directions, and the placement data are called *array data* (Sect. 8.1.3.5).

In the KENO V.a geometry description, the surrounding regions of any shape may be placed around an ARRAY, and they may consist of any number of regions in any order subject to the complete enclosure restriction. ARRAYs are positioned relative to the UNIT ORIGIN by specifying the location of the most negative point in the array, i.e. the most negative X, most negative Y, and most negative Z corner of the ARRAY. In KENO-VI geometry, an ARRAY may be placed in a region of any shape if the region boundary is contained within the ARRAY or shares the ARRAY boundary and does not cut across nested ARRAYs or HOLEs. In this case, only the section of the ARRAY contained within the region is recognized by the problem. ARRAYs are positioned relative to the UNIT ORIGIN by placing the ORIGIN of a specified UNIT in the ARRAY at a specified location in the UNIT.

To create a geometry mockup from a physical configuration, the user should keep the restrictions mentioned earlier in mind. There may be several ways of correctly describing the same physical configuration. Careful analysis of the system can result in a simpler mockup and shorter computer running time. A mockup with fewer geometry regions may run faster than the same mockup with extraneous regions. The number of UNITs used can affect the running time, because a transformation of coordinates must be made every time a history moves from one UNIT into another. Thus, if the size of a UNIT is small relative to the mean free path, a larger percentage of time is spent processing the transformation of coordinates. Because all boundaries in a UNIT must be checked for crossings, it may be more efficient to break up complex UNITs into several smaller, simpler UNITs. The trade-off involves the time required to process more boundary crossings vs the time required to transform coordinate systems when UNIT boundaries are crossed.

Geometry dimensions: The use of FIDO syntax may help simplify the description of the geometry. For example, a $20 \times 20 \times 2.5$ cm rectangular parallelepiped would have been described as CUBOID 1 1 10.0 -10.0 10.0 -10.0 1.25 -1.25 in KENO V.a and CUBOID 1 10.0 10.0 10.0 10.0 1.25 1.25 in KENO-VI. By using the P option (see Table 8.1.19), the same rectangular parallelepiped could be described as CUBOID 1 1 4P10.0 2P1.25 in KENO V.a and or CUBOID 1 4P10.0 2P1.25 in KENO-VI, where the last 6 entries describe the geometry. The P option simply repeats the dimension following the P for the number of times stated before the P, and it reverses the sign every time. Therefore, 6P8.0 is equivalent to 8.0 -8.0 8.0 -8.0 8.0 -8.0.

Geometry comments: One comment can be entered for each UNIT in the *geometry region data*. Similarly, one comment can be entered for each ARRAY in the *array definition data*. A comment can be entered using the keyword COM=. This is followed by a comment with a maximum length of 132 characters. The comment

must be preceded and terminated by a delimiter character. Acceptable delimiters include ” , ` , * , ^ , or !. One comment is allowed for each UNIT in the *geometry region data*. If multiple comments are entered for a UNIT, the last comment is used. The comment can be entered anywhere after the UNIT number where a keyword is expected (Sect. 8.1.3.4). See the following example.

KENO V.a:

```
READ GEOM
UNIT 1
  COM=*SPHERICAL METAL UNIT*
  SPHERE 1 1 5.0
  CUBE 0 1 2P5.0
UNIT 2
  CYLINDER 1 1 5.0 2P5.0
  CUBE 0 1 2P5.0
  COM=!CYLINDRICAL METAL UNIT!
UNIT 3
  HEMISPHE+X 1 1 5.0
  COM='HEMISPHERICAL METAL UNIT'
  CUBE 0 1 2P5.0
UNIT 4
  COM=^ARRAY OF SPHERICAL UNITS^
  ARRAY 1 3*0.0
UNIT 5
  COM="ARRAY OF CYLINDRICAL UNITS"
  ARRAY 2 3*0.0
UNIT 6
  COM='ARRAY OF HEMISPHERICAL UNITS'
  ARRAY 3 3*0.0
END GEOM
```

KENO-VI:

```
READ GEOM
UNIT 1
  COM=*SPHERICAL METAL UNIT*
  SPHERE 1  5.0
  CUBOID 2  6P5.0
  MEDIA 1 1 1
  MEDIA 0 1 -1 2
  BOUNDARY 2
UNIT 2
  CYLINDER 1 5.0 2P5.0
  CUBOID 2 6P5.0
  MEDIA 1 1 1
  COM=!CYLINDRICAL METAL UNIT!
  MEDIA 0 1 -1 2
  BOUNDARY 2
UNIT 3
  SPHERE 1  5.0 CHORD +X=0.0
  MEDIA 1 1 1
  COM='HEMISPHERICAL METAL UNIT'
  CUBOID 2  6P5.0
  MEDIA 0 1 -1 2
  BOUNDARY 2
UNIT 4
  COM='ARRAY OF SPHERICAL UNITS'
  CUBOID 1 6P15
  ARRAY 1 1 PLACE 2 2 2 3*0.0
  BOUNDARY 1
UNIT 5
  CUBOID 1 6P15.0
  COM='ARRAY OF CYLINDRICAL UNITS'
  ARRAY 2 1 PLACE 2 2 2 3*0.0
```

(continues on next page)

(continued from previous page)

```
  BOUNDARY 1
UNIT 6
  COM='ARRAY OF HEMISPHERICAL UNITS'
  CUBOID 1 6P15.0
  ARRAY 3 1 PLACE 2 2 2 3*0.0
  BOUNDARY 1
GLOBAL UNIT 7
  COM='ARRAY OF ARRAYS'
  CUBOID 1 4P15.0 2P45.0
  ARRAY 4 1 PLACE 1 1 2 3*0.0
  BOUNDARY 1
END GEOM
```

One comment is allowed for each array in the *array definition data*. The rules governing these comments are the same as those listed above. However, the comment for an ARRAY must precede the UNIT arrangement description, and it can precede the ARRAY number (Sect. 8.1.3.5). Examples of correct ARRAY comments are given below.

KENO V.a:

```
READ ARRAY
  COM='ARRAY OF SPHERICAL METAL UNITS'
  ARA=1 NUX=2 NUY=2 NUZ=2 FILL F1 END FILL
  ARA=2 COM='ARRAY OF CYLINDRICAL METAL UNITS'
  NUX=2 NUY=2 NUZ=2 FILL F2 END FILL
  ARA=3 NUX=2 NUY=2 NUZ=2
  COM='ARRAY OF HEMISPHERICAL METAL UNITS'
  FILL F3 END FILL
  ARA=4 COM='COMPOSITE ARRAY OF ARRAYS. Z=1 IS SPHERES, Z=2 IS CYLINDERS,   Z=3 IS HEMISPHERES'
  NUX=1 NUY=1 NUZ=3 FILL 4 5 6 END FILL
END ARRAY
```

KENO-VI:

```
READ ARRAY
  COM='ARRAY OF SPHERICAL METAL UNITS'
  ARA=1 NUX=3 NUY=3 NUZ=3 FILL F1 END FILL
  ARA=2 COM='ARRAY OF CYLINDRICAL METAL UNITS'
  NUX=3 NUY=3 NUZ=3 FILL F2 END FILL
  ARA=3 NUX=3 NUY=3 NUZ=3
  COM='ARRAY OF HEMISPHERICAL METAL UNITS'
  FILL F3 END FILL
  ARA=4 COM='COMPOSITE ARRAY OF ARRAYS. Z=1 IS SPHERES, Z=2 IS CYLINDERS, Z=3 IS  HEMISPHERES'
  NUX=1 NUY=1 NUZ=3 FILL 4 5 6 END FILL
END ARRAY
```

Some of the basics of KENO geometry are illustrated in the following examples:

EXAMPLE 1. Assume a stack of six cylindrical disks each measuring 5 cm in radius and 2 cm thick. The bottom disk is composed of material 1, and the next disk is composed of material 2, etc., alternating throughout the stack. A square plate of material 3 that is 20 cm on a side and 2.5 cm thick is centered on top of the stack. This configuration is shown in Fig. 8.1.18.

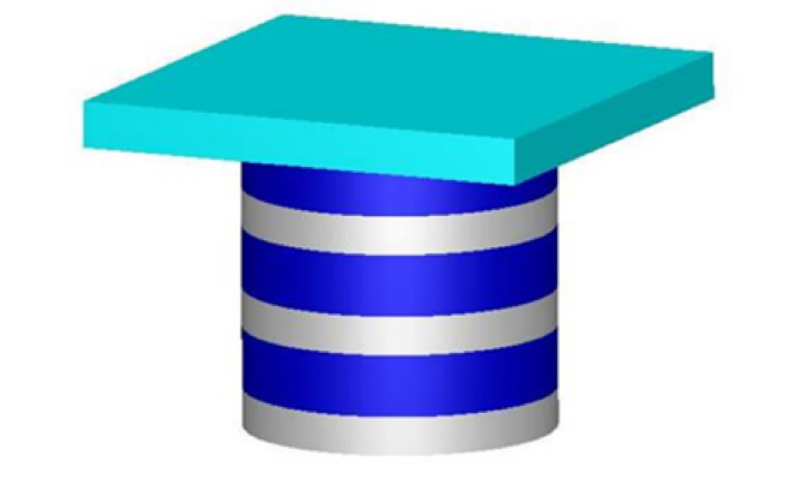


Fig. 8.1.18: Stack of disks with a square top.

This problem can be described as a single UNIT problem, describing the cylindrical portion first. In this instance, the origin has been chosen at the center bottom of the bottom disk. The bottom disk is defined by the first cylinder description, and the next disk is defined by the difference between the first and second cylinder descriptions. Since both disks have a radius of 5.0 and a -Z length of 0.0, the first cylinder containing material 1 exists from Z = 0.0 to Z = 2.0, and the second cylinder containing material 2 exists from Z = 2.0 to Z = 4.0. When all the disks have been described, a void cuboid having the same X and Y dimensions as the square plate and the same Z dimensions as the stack of disks is defined. The square plate of material 3 is then defined on top of the stack. Omission of the first cuboid description would result in the stack of disks being encased in a solid cuboid of material 3 instead of having a flat plate on top of the stack. The geometry input is shown below.

Data description 1, Example 1.

KENO V.a:

```
READ GEOM
CYLINDER 1  1     5.0 2.0    0.0
CYLINDER 2  1     5.0 4.0    0.0
CYLINDER 1  1     5.0 6.0    0.0
CYLINDER 2  1     5.0 8.0    0.0
CYLINDER 1  1     5.0 10.0   0.0
CYLINDER 2  1     5.0 12.0   0.0
CUBOID 0 1 10.0 -10.0 10.0 -10.0 12.0 0.0
CUBOID 3 1 10.0 -10.0 10.0 -10.0 14.5 0.0
```

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
CYLINDER   1   5.0   2.0   0.0
CYLINDER   2   5.0   4.0   2.0
CYLINDER   3   5.0   6.0   4.0
CYLINDER   4   5.0   8.0   6.0
CYLINDER   5   5.0  10.0   8.0
CYLINDER   6   5.0  12.0  10.0
CUBOID     7  10.0 -10.0  10.0 -10.0  12.0  0.0
CUBOID     8  10.0 -10.0  10.0 -10.0  14.5  0.0
```

(continues on next page)

(continued from previous page)

```
MEDIA 1 1 1
MEDIA 2 1 2
MEDIA 1 1 3
MEDIA 2 1 4
MEDIA 1 1 5
MEDIA 2 1 6
MEDIA 0 1 -1 -2 -3 -4 -5 -6 7
MEDIA 3 1 -7 8
BOUNDARY  8
END GEOM
```

An alternative description of the same example is given below. The origin has been chosen at the center of the disk of material 1 nearest the center of the stack. This disk of material 1 is defined by the first cylinder description, and the disks of material 2 on either side of it are defined by the second cylinder description. The top and bottom disks of material 1 are defined by the third cylinder, and the top disk of material 2 is defined by the last cylinder. The square plate is defined by the two cuboids. This description is more efficient than the previous one because there are fewer surfaces to check for crossings.

Data description 2, Example 1.

KENO V.a:

```
READ GEOM
CYLINDER 1 1  5.0   1.0 -1.0
CYLINDER 2 1  5.0   3.0 -3.0
CYLINDER 1 1  5.0   5.0 -5.0
CYLINDER 2 1  5.0   7.0 -5.0
CUBOID   0 1 10.0 -10.0 10.0 -10.0 7.0 -5.0
CUBOID   3 1 10.0 -10.0 10.0 -10.0 9.5 -5.0
```

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
CYLINDER 10  5.0   1.0 -1.0
CYLINDER 20  5.0   3.0 -3.0
CYLINDER 30  5.0   5.0 -5.0
CYLINDER 40  5.0   7.0 -5.0
CUBOID   50 10.0 -10.0 10.0 -10.0 7.0 -5.0
CUBOID   60 10.0 -10.0 10.0 -10.0 9.5 -5.0
MEDIA 1 1 10
MEDIA 2 1 -10 20
MEDIA 1 1 -20 30
MEDIA 2 1 -30 40
MEDIA 0 1 -40 50
MEDIA 3 1 -50 60
BOUNDARY  60
END GEOM
```

Example 1 can also be described as an ARRAY. Define three different UNIT types. UNIT 1 will define a disk of material 1, UNIT 2 will define a disk of material 2, and UNIT 3 will define the square plate of material 3. The origin of each UNIT is defined at the center bottom of the disk or plate being described. As mentioned earlier, only UNITs with a CUBE or CUBOID as their outer boundary can be placed in a cuboidal ARRAY. The geometry input for this arrangement is shown below.

Data description 3, Example 1.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1 1  5.0   2.0  0.0
CUBOID    0 1 10.0 -10.0 10.0 -10.0 2.0 0.0
UNIT 2
CYLINDER  2 1  5.0   2.0  0.0
CUBOID    0 1 10.0 -10.0 10.0 -10.0 2.0 0.0
UNIT 3
CUBOID    3 1 10.0 -10.0 10.0 -10.0 2.5 0.0
END GEOM
READ ARRAY  NUX=1 NUY=1 NUZ=7 FILL 1 2 1 2 1 2 3 END ARRAY
```

Note: The ARRAY is assumed to be the GLOBAL ARRAY because only a single ARRAY is defined.

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER 1  5.0   2.0  0.0
CUBOID  2 10.0 -10.0 10.0 -10.0 2.0 0.0
MEDIA 1 1 1
MEDIA 0 1 -1 2
BOUNDARY 2
UNIT 2
CYLINDER  1  5.0   2.0  0.0
CUBOID    2 10.0 -10.0 10.0 -10.0 2.0 0.0
MEDIA 2 1 1
MEDIA 0 1 2 -1
BOUNDARY 2
UNIT 3
CUBOID    1 10.0 -10.0 10.0 -10.0 2.5 0.0
MEDIA 3 1 1
BOUNDARY 1
GLOBAL UNIT 4
CUBOID 1  10 -10 10 -10 14.5 0.0
ARRAY 1 1  PLACE 1 1 1 3*0.0
BOUNDARY 1
END GEOM
READ ARRAY ARA=1 NUX=1 NUY=1 NUZ=7 FILL 1 2 1 2 1 2 3 END ARRAY
```

If the user prefers for the origin of each unit to be at its center, the geometry region data can be entered as shown below. The array data would be identical to that of data description 3, Example 1.

Data description 4, Example 1.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1 1  5.0   1.0 -1.0
CUBOID    0 1 10.0 -10.0 10.0 -10.0 1.0 -1.0
UNIT 2
CYLINDER  2 1  5.0   1.0 -1.0
CUBOID    0 1 10.0 -10.0 10.0 -10.0 1.0 -1.0
UNIT 3
CUBOID  3 1 10.0 -10.0 10.0 -10.0 1.25 -1.25
END GEOM
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER 1  5.0   1.0 -1.0
CUBOID  2   10.0 -10.0 10.0 -10.0 1.0 -1.0
MEDIA 1 1 1
MEDIA 0 1 -1 2
BOUNDARY  2
UNIT 2
CYLINDER 1  5.0   1.0 -1.0
CUBOID  2  10.0 -10.0 10.0 -10.0 1.0 -1.0
MEDIA 2 1 1
MEDIA 0 1 -1 2
BOUNDARY  2
UNIT 3
CUBOID 1 10.0 -10.0 10.0 -10.0 1.25 -1.25
MEDIA 3 1 1
BOUNDARY  1
GLOBAL UNIT 4
CUBOID 1 10 -10 10 -10 14.5 0.0
ARRAY 1 1 PLACE 1 1 1 0.0 0.0 1.0
BOUNDARY 1
END GEOM
```

Be aware that each UNIT in a geometry description can have its origin defined independent of the other UNITs. It would be correct to use UNITs 1 and 3 from data descriptions 3, and UNIT 2 from data description 4. The *array data* would remain the same as data description 3, Example 1. The user should define the origin of each unit to be as convenient as possible for the chosen description. Care should be taken when assigning coordinates to the UNIT used to PLACE the ARRAY in its surrounding region.

Another method of describing Example 1 as a bare array is to define UNIT 1 to be a disk of material 1 topped by a disk of material 2. The origin has been chosen at the center bottom of the disk of material 1. UNIT 2 is the square plate of material 3 with the origin at the center of the UNIT. The ARRAY consists of three UNIT 1s, topped by a UNIT 2, as shown below.

Data description 5, Example 1.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1 1  5.0   2.0  0.0
CYLINDER  2 1  5.0   4.0  0.0
CUBOID    0 1 10.0 -10.0 10.0 -10.0 4.0 0.0
UNIT 2
CUBOID    3 1 10.0 -10.0 10.0 -10.0 1.25 -1.25
END GEOM
READ ARRAY  NUX=1 NUY=1 NUZ=4 FILL 3R1 2 END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER  1  5.0   2.0  0.0
CYLINDER  2  5.0   4.0  2.0
CUBOID  3   10.0 -10.0 10.0 -10.0 4.0 0.0
MEDIA 1 1 1
MEDIA 2 1 2
MEDIA 0 1 -1 -2 3
BOUNDARY  3
UNIT 2
CUBOID  1  10.0 -10.0 10.0 -10.0 1.25 -1.25
```

(continues on next page)

(continued from previous page)

```
MEDIA 3 1 1
BOUNDARY  1
GLOBAL UNIT 3
CUBOID  1 10 -10 10 -10 14.5 0.0
ARRAY 1 1 PLACE 1 1 1 3*0.0
BOUNDARY  1
END GEOM
READ ARRAY  ARA=1 NUX=1 NUY=1 NUZ=4 FILL 3R1 2 END ARRAY
```

Example 1 can be described as a reflected ARRAY by treating the square plate as a reflector in the positive Z direction. One means of describing this situation is to define UNITs 1 and 2 as in data description 3, Example 1. The origin of the GLOBAL UNIT is defined to be at the center of the ARRAY. The corresponding input geometry is shown below.

Data description 6, Example 1.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1 1   5.0   2.0  0.0
CUBOID    0 1  10.0  -10.0  10.0  -10.0  2.0  0.0
UNIT 2
CYLINDER  2 1   5.0   2.0  0.0
CUBOID      0 1  10.0  -10.0  10.0  -10.0  2.0  0.0
GLOBAL UNIT 3
ARRAY        1 -10.0  -10.0  -6.0
CUBOID     3 1  10.0  -10.0  10.0  -10.0  8.5  -6.0
END GEOM
READ ARRAY  NUX=1 NUY=1 NUZ=6 FILL 1 2 1 2 1 2 END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER  1  5.0   2.0  0.0
CUBOID  2   10.0 -10.0 10.0 -10.0 2.0  0.0
MEDIA 1 1 1
MEDIA 0 1 -1 2
BOUNDARY  2
UNIT 2
CYLINDER  1  5.0   2.0  0.0
CUBOID  2  10.0 -10.0 10.0 -10.0 2.0  0.0
MEDIA 2 1 1
MEDIA 0 1 -1 2
BOUNDARY  2
GLOBAL UNIT 3
CUBOID 1  10.0 -10.0 10.0 -10.0 6.0 -6.0
CUBOID 2  10.0 -10.0 10.0 -10.0 8.5 -6.0
ARRAY 1 1 PLACE 1 1 1 0.0 0.0 -6.0
MEDIA 3 1 -1 2
BOUNDARY  2
END GEOM
READ ARRAY ARA=1 NUX=1 NUY=1 NUZ=6 FILL 1 2 1 2 1 2 END ARRAY
```

The user could have chosen the origin of the ARRAY boundary to be at the center bottom of the ARRAY, in which case the geometry description for the GLOBAL UNIT would be:

KENO V.a:

```
ARRAY       1 -10.0  -10.0  0.0
CUBOID   3 1 10.0  -10.0  10.0  -10.0  14.5  0.0
```

or

```
ARRAY    1  -10.0  -10.0 0.0
REPLICATE  3  4*0.0  2.5  0 1
```

The reflector region at the top of the array can be added by using a CUBOID or by using a REPLICATE description in KENO V.a. Recall that there is no REPLICATE function in KENO-VI.

A simpler method of describing Example 1 as a reflected array is to define only one unit as in data description 5, Example 1. The square plate is treated as a reflector, as in data description 6, Example 1. The input for this arrangement is given below.

Data description 7, Example 1.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1 1   5.0   2.0  0.0
CYLINDER  2 1   5.0   4.0  0.0
CUBOID      0 1  10.0  -10.0  10.0  -10.0  4.0  0.0
GLOBAL UNIT 2
ARRAY        1  -10.0  -10.0  0.0
CUBOID     3  1  10.0  -10.0  10.0  -10.0  14.5  0.0
END GEOM
READ ARRAY NUX=1 NUY=1 NUZ=3 END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER  1   5.0   2.0  0.0
CYLINDER  2   5.0   4.0  2.0
CUBOID   3   10.0 -10.0 10.0 -10.0 4.0 0.0
MEDIA 1 1 1
MEDIA 2 1 2
MEDIA 0 1 -1 -2 3
BOUNDARY  3
GLOBAL UNIT 2
CUBOID  1  10.0 -10.0 10.0 -10.0  12.0 0.0
CUBOID  2  10.0 -10.0 10.0 -10.0 14.5 0.0
ARRAY 1 1 PLACE 1 1 1 3*0.0
MEDIA 3 1 -1 2
BOUNDARY  2
END GEOM
READ ARRAY ARA=1 NUX=1 NUY=1 NUZ=3 FILL F1  END FILL  END ARRAY
```

EXAMPLE 2. Assume that the stack of six disks in Example 1 is placed at the center bottom of a cylindrical container composed of material 6 whose inside diameter is 16.0 cm. The bottom and sides of the container are 0.25 cm thick, the top is open, and the total height of the container is 18.25 cm. Assume the square plate of Example 1 is centered on top of the container.

The geometry input can be described utilizing most of the data description methods associated with Example 1. One method of describing Example 2 as a single UNIT is given below.

Data description 1, Example 2.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1 1  5.0    1.0  -1.0
CYLINDER  2 1  5.0    3.0  -3.0
CYLINDER  1 1  5.0    5.0  -5.0
CYLINDER  2 1  5.0    7.0  -5.0
CYLINDER  0 1  8.0   13.0  -5.0
CYLINDER  6 1  8.25  13.0  -5.25
CUBOID    0 1 10.0  -10.0  10.0 -10.0 13.0 -5.25
CUBOID    3 1 10.0  -10.0  10.0 -10.0 15.5 -5.25
END GEOM
```

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
CYLINDER  1  5.0    1.0  -1.0
CYLINDER  2  5.0    3.0  -3.0
CYLINDER  3  5.0    5.0  -5.0
CYLINDER  4  5.0    7.0  -5.0
CYLINDER  5  8.0   13.0  -5.0
CYLINDER  6  8.25  13.0  -5.25
CUBOID    7  10.0 -10.0  10.0 -10.0 13.0 -5.25
CUBOID    8  10.0 -10.0  10.0 -10.0 15.5 -5.25
MEDIA 1 1 1
MEDIA 2 1 -1 2
MEDIA 1 1 -2 3
MEDIA 2 1 -3 4
MEDIA 0 1 -4 5
MEDIA 6 1 -5 6
MEDIA 0 1 -6 7
MEDIA 3 1 -7 8
BOUNDARY  8
END GEOM
```

In the above description, the origin is defined to be at the center of the disk of material 1 nearest the center of the stack. This disk is defined by the first cylinder description. The disks of material 2 above and below it are defined by the second cylinder description. The disks of material 1 above and below them are defined by the third cylinder description. The top disk of material 2 is defined by the fourth cylinder description. The void interior of the container is defined by the fifth cylinder description. The container is defined by the last cylinder description. The first cuboid description is used to define a void whose X and Y dimensions are the same as the square plate and whose Z dimensions are the same as the container. The last cuboid description defines the square plate. Omission of the first cuboid description would result in the container being encased in a solid cuboid of material 3. Thus, both cuboids are necessary to properly define the square plate.

Example 2 can be described as a reflected **ARRAY**. One of the descriptions uses only one **UNIT** and is similar to data description 7, Example 1. This description is shown below.

Data description 2, Example 2.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1  1  5.0    2.0   0.0
CYLINDER  2  1  5.0    4.0   0.0
CUBOID    0 1  5.0   -5.0   5.0   -5.0  4.0  0.0
GLOBAL UNIT 2
ARRAY          1  -5.0   -5.0   0.0
CYLINDER  0  1  8.0   18.0   0.0
```

(continues on next page)

(continued from previous page)

```
CYLINDER  6  1  8.25  18.0  -0.25
CUBOID      0  1  10.0  -10.0  10.0  -10.0  18.0  -0.25
CUBOID      3  1  10.0  -10.0  10.0  -10.0  20.5  -0.25
END GEOM
READ ARRAY  NUX=1 NUY=1 NUZ=3 END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER  1  5.0    2.0   0.0
CYLINDER  2  5.0    4.0   2.0
CUBOID    3  5.0   -5.0   5.0   -5.0  4.0  0.0
MEDIA 1 1 1
MEDIA 2 1 2
MEDIA 0 1 -1 -2 3
BOUNDARY  3
GLOBAL UNIT 2
CUBOID   1 5.0 -5.0 5.0 -5.0 12.0 0.0
ARRAY 1 1 PLACE 1 1 1 3*0.0
CYLINDER  2  8.0   18.0   0.0
MEDIA 0 1 -1 2
CYLINDER  3  8.25  18.0  -0.25
MEDIA 6 1 -2 3
CUBOID    4 10.0  -10.0  10.0  -10.0 20.5 18.0
CUBOID    5 10.0  -10.0  10.0  -10.0 20.5 -0.25
MEDIA 3 1 4
MEDIA 0 1 -3 -4 5
BOUNDARY  5
END GEOM
READ ARRAY ARA NUX=1 NUY=1 NUZ=3 FILL F1 END FILL END ARRAY
```

In the above data description, the first two cylinder descriptions define a disk of material 1 with a disk of material 2 directly on top of it. A tight fitting void cuboid is placed around them so they can be stacked three high to achieve the stack of disks shown in Example 1, Fig. 8.1.18. This array comprises the array portion of the geometry region description. The origin of the array boundary, a tight fitting cube or cuboid that encompasses the array, is defined by the **ARRAY** description. Everything after the **ARRAY** record is considered part of the reflector. The first cylinder after the **ARRAY** record defines the void interior of the cylindrical container. The next cylinder defines the walls of the container. The second-to-last cuboid defines a void volume outside the container from its bottom to its top and having the same X and Y dimensions as the square plate. The last cuboid defines the square plate of material 3 that is sitting on top of the container.

Another method to describe Example 2 is as an array composed of units that contain both the stack and container. This description requires a minimum of four units to describe the problem. This configuration is given below in data description 3, Example 2.

Data description 3, Example 2.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  6 1  8.25   0.25   0.0
CUBOID    0 1 10.0  -10.0   10.0 -10.0 0.25  0.0
UNIT 2
CYLINDER  1 1  5.0    2.0    0.0
CYLINDER  2 1  5.0    4.0    0.0
CYLINDER  0 1  8.0    4.0    0.0
CYLINDER  6 1  8.25   4.0    0.0
```

(continues on next page)

(continued from previous page)

```
CUBOID    0 1 10.0  -10.0   10.0 -10.0 4.0   0.0
UNIT 3
CYLINDER  0 1  8.0    3.0   -3.0
CYLINDER  6 1  8.25   3.0   -3.0
CUBOID    0 1 10.0  -10.0   10.0 -10.0 3.0  -3.0
CUBOID    3 1 10.0  -10.0   10.0 -10.0 5.5  -3.0
END GEOM
READ ARRAY  NUX=1 NUY=1 NUZ=5 FILL 1 3R2 3 END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER  1  8.25   0.25   0.0
CUBOID  2   10.0  -10.0   10.0 -10.0 0.25  0.0
MEDIA 6 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 2
CYLINDER  10  5.0    2.0    0.0
CYLINDER  20  5.0    4.0    0.0
CYLINDER  30  8.0    4.0    0.0
CYLINDER  40  8.25   4.0    0.0
CUBOID  50   10.0  -10.0   10.0 -10.0 4.0   0.0
MEDIA 1 1 10
MEDIA 2 1 20 -10
MEDIA 0 1 30 -20
MEDIA 6 1 40 -30
MEDIA 0 1 50 -40
BOUNDARY  50
UNIT 3
CYLINDER  1  8.0    3.0   -3.0
CYLINDER  2  8.25   3.0   -3.0
CUBOID    3 10.0  -10.0   10.0 -10.0 5.5   3.0
CUBOID    4 10.0  -10.0   10.0 -10.0 5.5  -3.0
MEDIA 0 1 1
MEDIA 6 1 2 -1
MEDIA 3 1 3
MEDIA 0 1 4 -3 -2
BOUNDARY  4
GLOBAL UNIT 4
CUBOID  1  10.0 -10.0 10.0 -10.0 20.75 0.0
ARRAY 1 1 PLACE 1 1 1 3*0.0
BOUNDARY  1
END GEOM
READ ARRAY  NUX=1 NUY=1 NUZ=5 FILL 1 3R2 3 END ARRAY
```

In the above description, UNIT 1 is the bottom of the cylindrical container. The void CUBOID is only as tall as the bottom of the container, and its X and Y dimensions are the same as the square plate on top of the container. If all the UNITs in the ARRAY use these same dimensions in the X and Y directions, the requirement that adjacent faces of units in contact with each other must be the same size and shape is satisfied. This ARRAY is stacked in the Z direction, so all UNITs must have the same overall dimensions in the X direction and in the Y direction. UNIT 2 will be used in the ARRAY three times to create the stack of disks. It contains a disk of material 1 topped by a disk of material 2. The portion of the container that contains the disks and the cuboid that defines the outer boundaries of the unit are included in UNIT 2. UNIT 3 describes the empty top portion of the container and the square plate on top of it. The Z dimensions of UNIT 3 were determined by subtracting three times the total Z dimension of UNIT 2 from the inside height of the container [18.0 - (3 × 4.0) = 6.0]. This can also be determined from the overall height of the container by subtracting off the bottom thickness of the container and three times the height of UNIT 2 [18.25 - 0.25 - (3 × 4.0) = 6.0]. The origin of UNIT 3 is located at the center of this distance. For the KENO-VI input, a GLOBAL UNIT is also provided,

UNIT 4, containing the ARRAY built with UNITs 1, 2, 3.

EXAMPLE 3. Refer to Example 1, Fig. 8.1.18, and imagine a HOLE 1.5 cm in diameter is drilled along the centerline of the stack through the disks and the square plate. In KENO V.a this HOLE would eliminate the possibility of describing the system as a single unit because the HOLE in the center of the alternating materials of the stack cannot be described in a manner that allows each successive geometry region to encompass the regions interior to it. Therefore, it must be described as an ARRAY. The square plate on the top of the disks is defined as a UNIT in the ARRAY. In the geometry description given below, the square plate is defined in UNIT 3. KENO-VI can easily describe this configuration as a single UNIT.

Data description 1, Example 3.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  0 1  0.75   2.0  0.0
CYLINDER  1 1  5.0    2.0  0.0
CUBOID    0 1 10.0  -10.0 10.0 -10.0 2.0 0.0
UNIT 2
CYLINDER  0 1  0.75   2.0  0.0
CYLINDER  2 1  5.0    2.0  0.0
CUBOID    0 1 10.0  -10.0 10.0 -10.0 2.0 0.0
UNIT 3
CYLINDER  0 1  0.75   2.5  0.0
CUBOID    3 1 10.0  -10.0 10.0 -10.0 2.5 0.0
END GEOM
READ ARRAY  NUX=1 NUY=1 NUZ=7 FILL 1 2 2Q2 3 END FILL END ARRAY
```

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
CYLINDER  1  0.75     7.0  -5.0
CYLINDER  2  5.0      1.0  -1.0
CYLINDER  3 5.0      3.0  -3.0
CYLINDER  4 5.0      5.0  -5.0
CYLINDER  5 5.0      7.0  -5.0
CYLINDER  6 8.0    13.0  -5.0
CYLINDER  7 8.25  13.0  -5.25
CUBOID  8  10.0  -10.0  10.0 -10.0 15.5  13.0
CUBOID   9 10.0  -10.0  10.0 -10.0 15.5 -5.25
MEDIA 0 1  1
MEDIA 1 1 -1  2
MEDIA 2 1 -1 -2  3
MEDIA 1 1 -1 -3  4
MEDIA 2 1 -1 -4  5
MEDIA 0 1 -5  6
MEDIA 6 1 -6  7
MEDIA 3 1  8
MEDIA 0 1 -7 -8  9
BOUNDARY   9
END GEOM
```

In data description 1, Example 3 above, KENO V.a input, UNIT 1 describes a disk of material 1 with a HOLE through its centerline. The first CYLINDER defines the HOLE, the second defines the rest of the disk, and the CUBOID defines the size of the UNIT to be consistent with the square plate so they can be stacked together in an ARRAY. UNIT 2 describes a disk of material 2 in similar fashion. UNIT 3 describes the square plate of material 3 with a HOLE through its center. The CYLINDER defines the HOLE and the CUBOID defines the square plate. These three UNITs are stacked in the Z direction to achieve the composite system. This is represented by FILL 1 2 2Q2 3. The 2Q2 repeats the two entries preceding the 2Q2 twice. Alternatively, this can be

achieved by entering FILL 1 2 1 2 1 2 3 END FILL. The same ARRAY can also be achieved using the LOOP option. An example of the data for this option is:

LOOP 1 6R1 1 5 2 2 6R1 2 6 2 3 6R1 7 7 1 END LOOP.

UNIT 1 is placed at the X = 1, Y = 1, and Z = 1,3,5 positions of the ARRAY by entering 1 6R1 1 5 2. UNIT 2 is positioned at the X = 1, Y = 1 and Z = 2,4,6 positions in the ARRAY by entering 2 6R1 2 6 2. UNIT 3 is placed at the X = 1, Y = 1, Z = 7 position of the ARRAY by entering 3 6R1 7 7 1. See Sect. 8.1.3.5 for additional information regarding ARRAY specifications.

For the KENO-VI input, UNIT 1 contains the entire problem description. The first CYLINDER describes the 1.5 cm diameter hole through the stack. The next four CYLINDERs define the stack. The sixth and seventh CYLINDERs describe the void and container. The two CUBOIDs describe the top plate and surrounding global region of void. The MEDIA cards are used to place the materials in the appropriate regions.

EXAMPLE 4. Assume two large cylinders that are 2.5 cm in radius and 5 cm long are connected by a smaller cylinder that is 0.5 cm in radius and 10 cm long, as shown in Fig. 8.1.19. All three cylinders are composed of material 1. By starting the geometry description in the small cylinder, this system can be described as a single unit.

Fig. 8.1.19: Two large cylinders joined axially by a small cylinder.

Data description 1, Example 4.

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
XCYLINDER  1  0.5  5.0  -5.0
XCYLINDER  2  2.5  5.0  -5.0
XCYLINDER  3  2.5 10.0 -10.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
MEDIA 1 1 3 -2
BOUNDARY  3
END GEOM
```

KENO V.a:

```
READ GEOM
CYLINDER  1 1 0.5  5.0  -5.0
CYLINDER  0 1 2.5  5.0  -5.0
CYLINDER  1 1 2.5 10.0 -10.0
END GEOM
```

The origin is at the center of the small cylinder, which is described by the first cylinder description. The second cylinder description defines a void cylinder surrounding the small cylinder. Its radius is the same as the large cylinders, and its height (length) coincides with that of the small cylinder. The last cylinder description defines the large cylinders on either end of the small cylinder. In KENO V.a, because this problem does not specify otherwise, the length of the CYLINDERs is assumed to coincide with the Z axis. In KENO-VI,

because the problem was created using XCYLINDERs, the long axes of the CYLINDERs coincide with the X axis.

EXAMPLE 5. Assume two large cylinders with a center-to-center spacing of 15 cm, each having a radius of 2.5 cm and length of 5 cm, are connected radially by a small cylinder having a radius of 1.5 cm, as shown in Fig. 8.1.20.

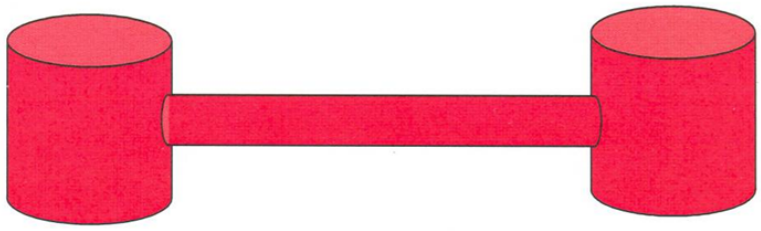


Fig. 8.1.20: Two large cylinders radially connected by a small cylinder.

This system cannot be described rigorously in KENO V.a geometry because the intersection of the cylinders cannot be described. However, it can be approximated two ways, as shown in Fig. 8.1.21. The top approximation is described in data description 1, Example 5. The bottom approximation is described in data description 2, Example 5, and data description 3, Example 5. These may be poor approximations for criticality safety calculations.

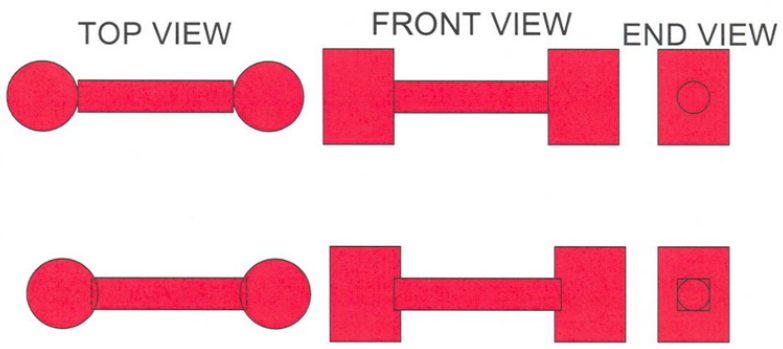


Fig. 8.1.21: KENO V.a approximations of cylindrical intersections.

Data description 1, Example 5.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1 1 2.5 2.5 -2.5
CUBE      0 1 2.5 -2.5
UNIT 2
XCYLINDER 1 1 1.5 5.0 -5.0
CUBOID    0 1 5.0 -5.0 2.5 -2.5 2.5 -2.5
END GEOM
READ ARRAY NUX=3 NUY=1 NUZ=1 FILL 1 2 1 END ARRAY
```

UNIT 1 defines a large CYLINDER, and UNIT 2 describes the small CYLINDER. In both UNITs the origin is at the center of the CYLINDER. The large CYLINDERs have their centerlines along the Z axis and the small CYLINDER has its length along the X axis.

Data description 2, Example 5.

KENO V.a:

```
READ GEOMETRY
UNIT 1
CYLINDER 1 1 2.5 1.0 0.0
CUBOID 0 1 4P2.5 1.0 0.0
UNIT 2
ZHEMICYL-X 1 1 2.5 2P1.5 CHORD 2.0
CUBOID 0 1 2.0 3P-2.5 2P1.5
UNIT 3
ZHEMICYL+X 1 1 2.5 2P1.5 CHORD 2.0
CUBOID 0 1 2.5 -2.0 2P2.5 2P1.5
UNIT 4
XCYLINDER 1 1 1.5 2P5.5
CUBOID 0 1 2P5.5 2P2.5 2P1.5
UNIT 5
CUBOID 0 1 2P5.0 2P2.5 1.0 0.0
UNIT 6
ARRAY 1 3*0.0
UNIT 7
ARRAY 2 3*0.0
END GEOMETRY
READ ARRAY ARA=1 NUX=3 NUY=1 NUZ=1 FILL 1 5 1 END FILL
ARA=2 NUX=3 NUY=1 NUZ=1 FILL 2 4 3 END FILL
ARA=3 NUX=1 NUY=1 NUZ=3 FILL 6 7 6 END FILL
END ARRAY
```

The above geometry description uses ARRAYs of ARRAYs (see Sect. 8.1.4.6.3) to describe the bottom approximation of Fig. 8.1.21. UNIT 1 defines a large CYLINDER that has a radius of 2.5 cm and a height of 10 cm inside a close-fitting CUBOID. This is used in both large CYLINDERs as the portion of the large CYLINDER that exists above and below the region where the small CYLINDER joins it. UNIT 5 is the spacing between the tops of the two large CYLINDERs and the spacing between the bottoms of the two large CYLINDERs. ARRAY 1 thus defines the bottom of the system: two short CYLINDERs (UNIT 1s) separated by 10 cm (UNIT 5 is the separation). UNIT 6 contains ARRAY 1.

UNIT 2 is the left hemicylinder that adjoins the horizontal CYLINDER, and UNIT 3 is the right hemicylinder that adjoins the horizontal CYLINDER. UNIT 4 defines the horizontal CYLINDER. ARRAY 2 contains UNITs 2, 4, and 3 from left to right. This defines the central portion of the system where the horizontal CYLINDER adjoins the two hemicylinders. These hemicylinders are larger than half CYLINDERs. UNIT 7 contains ARRAY 2. The entire system is achieved by stacking a UNIT 6 above and below the UNIT 7 as defined in ARRAY 3, the GLOBAL ARRAY.

Data description 3, Example 5.

KENO V.a:

```
READ GEOMETRY
UNIT 1
CYLINDER 1 1 2.5 1.0 0.0
UNIT 2
CYLINDER 1 1 2.5 1.0 0.0
CUBOID 0 1 17.5 -2.5 2P2.5 1.0 0.0
HOLE 1 15.0 0.0 0.0
UNIT 3
ZHEMICYL-X 1 1 2.5 2P1.5 CHORD 2.0
UNIT 4
ZHEMICYL+X 1 1 2.5 2P1.5 CHORD 2.0
UNIT 5
XCYLINDER  1 1 1.5 2P5.5
```

(continues on next page)

(continued from previous page)

```
CUBOID 0 1  2P10.0 2P2.5 2P1.5
HOLE 3 -7.5 2*0.0
HOLE 4 7.5 2*0.0
END GEOMETRY
READ ARRAY
ARA=1 NUX=1 NUY=1 NUZ=3 FILL 2 5 2 END FILL
END ARRAY
```

The above geometry description uses HOLEs (see Sect. 8.1.4.6.1) to describe the bottom approximation of Fig. 8.1.21. UNIT 1 defines a large CYLINDER with a radius of 2.5 cm and a height of 1.0 cm. UNIT 2 defines the same CYLINDER within a CUBOID that extends from X = -2.5 to X = 17.5, from Y = -2.5 to Y = 2.5, and Z = 0.0 to Z = 1.0. The origin of the CYLINDER is at (0.0,0.0,0.0). Thus UNIT 2 describes the top and bottom of the CYLINDER on the left. UNIT 1 is positioned within this CUBOID as a HOLE with its origin at (15.0,0.0,0.0) to describe the top and bottom of the CYLINDER on the right. UNIT 3 is the left hemicylinder that adjoins the horizontal CYLINDER, and UNIT 4 is the right hemicylinder that adjoins the horizontal CYLINDER. UNIT 5 defines the horizontal CYLINDER with its origin at the center within a CUBOID that extends from X = -10.0 to X = +10.0, Y = -2.5 to Y = 2.5, and Z = -1.5 to Z = 1.5. UNIT 3 is positioned to the left of the horizontal CYLINDER, and UNIT 4 is positioned to the right of the horizontal CYLINDER by using HOLEs. The entire system is achieved by stacking a UNIT 2 above and below UNIT 5 as shown in the ARRAY data.

This same geometry description can be used with UNIT 2 redefined, having its origin defined so that it extends from X = -10 to X = 10, Y = -2.5 to Y = 2.5, and Z = 0.0 to Z = 1. In this instance, the geometry data would be identical except for UNIT 2. This alternative description of UNIT 2 is

KENO V.a:

```
UNIT 2
CYLINDER 1 1 2.5 1.0 0.0 ORIGIN -7.5 0.0
CUBOID 0 1 2P10.0 2P2.5 1.0 0.0
HOLE 1 7.5 0.0 0.0
```

This system can be easily described in KENO-VI geometry because intersections are allowed. The small CYLINDER is rotated in data description 1, Example 5.

Data description 1, Example 5.

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
CYLINDER   1  2.5  2.5 -2.5
CYLINDER   2  2.5  2.5 -2.5 ORIGIN Y=15.0
YCYLINDER  3  1.5 15.0  0.0
CUBOID     4  5.0 -5.0 17.5 -2.5 2.5 -2.5
MEDIA 1 1 1
MEDIA 1 1 2
MEDIA 1 1 3 -1 -2
MEDIA 0 1 4 -3 -2 -1
BOUNDARY  4
END GEOM
```

The first and second CYLINDERs define the two large CYLINDERs, and the third CYLINDER describes the small connecting CYLINDER. The two large CYLINDERs are oriented along the Z axis. The second large cylinder is translated so its origin is at position (0.0, 15.0, 0.0). The small CYLINDER is oriented along the Y axis. Region 1 consists of the material in the first large CYLINDER. Region 2 consists of the material in the second large CYLINDER. Region 3 consists of the material in the small CYLINDER but not in either of the large CYLINDERs. Region 4 is the BOUNDARY region.

EXAMPLE 6. Assume 2 small cylinders 1.0 cm in radius and 10 cm long are connected by a large cylinder 2.5 cm in radius and 5 cm long as shown in Fig. 8.1.22.

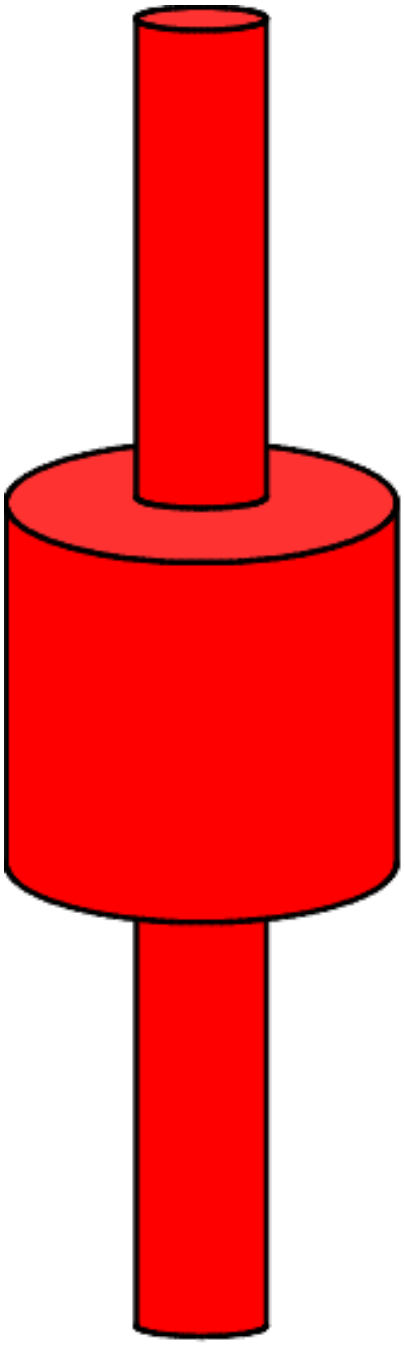


Fig. 8.1.22: Two small cylinders joined axially by a large cylinder.

This problem is very similar to Example 4. It can be described as a single `UNIT` in KENO-VI, but not in KENO V.a where it must be described as an array. In KENO V.a, `UNIT` 1 defines the large cylinder, and `UNIT` 2 defines the small cylinder. The origin of each `UNIT` is at its center. The composite system consists of two `UNIT` 2s and one `UNIT` 1 as shown below. In KENO-VI, `CYLINDER` 1 defines the long thin cylinder, and

CYLINDER 2 defines the short thick cylinder. The origin of each cylinder is at its center. In both KENO V.a and KENO-VI, the centerline of the cylinders lies along the Z axis.

Data description 1, Example 6.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1 1 2.5 2.5 -2.5
CUBE      0 1 2.5 -2.5
UNIT 2
CYLINDER  1 1 1.0 5.0 -5.0
CUBOID    0 1 2.5 -2.5 2.5 -2.5 5.0 -5.0
END GEOM
READ ARRAY NUX=1 NUY=1 NUZ=3 FILL 2 1 2 END ARRAY
```

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
CYLINDER  1  1.0 12.5  -12.5
CYLINDER  2  2.5  2.5   -2.5
CUBOID    3  4P2.5 12.5 -12.5
MEDIA 1 1 1
MEDIA 1 1 2 -1
MEDIA 0 1 3 -2 -1
BOUNDARY  3
END GEOM
```

EXAMPLE 7. Assume an $11 \times 5 \times 3$ square-pitched array of spheres of material 1, radius 3.75 cm, with a center-to-center spacing of 10 cm in the X, Y, and Z directions. The data for this system are given below.

Data description 1, Example 7.

KENO V.a:

```
READ GEOM
SPHERE 1 1 3.75
CUBE   0 1 5.0 -5.0
END GEOM
READ ARRAY NUX=11  NUY=5 NUZ=3 END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
SPHERE  1  3.75
CUBOID  2  6P5.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
GLOBAL UNIT 2
CUBOID 10 55.0 -55.0 25.0 -25.0 15.0 -15.0
ARRAY 1 10 PLACE 6 3 2 3*0.0
BOUNDARY  10
END GEOM
READ ARRAY NUX=11  NUY=5 NUZ=3 FILL F1 END FILL END ARRAY
```

EXAMPLE 8. Assume an $11 \times 5 \times 3$ rectangular-pitched array of spheres of material 1, whose radius is 3.75 cm and whose center-to-center spacing is 10 cm in the X direction, 15 cm in the Y direction, and 20 cm in the Z direction. The input for this geometry is given below.

Data description 1, Example 8.

KENO V.a:

```
READ GEOM
SPHERE  1 1 3.75
CUBOID  0 1 5.0  -5.0 7.5 -7.5 10.0 -10.0
END GEOM
READ ARRAY  NUX=11 NUY=5 NUZ=3 END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
SPHERE  1 3.75
CUBOID  2  5.0  -5.0 7.5 -7.5 10.0 -10.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
GLOBAL UNIT 2
CUBOID  1  55.0 -55.0 37.5 -37.5 30.0 -30.0
ARRAY 1 1 PLACE 6 3 2 3*0.0
BOUNDARY  1
END GEOM
READ ARRAY  NUX=11 NUY=5 NUZ=3 FILL F1 END FILL END ARRAY
```

EXAMPLE 9. Assume an $11 \times 5 \times 3$ square-pitched array of spheres of material 1 with a 3.75 cm radius and 10 cm center-to-center spacing in the X, Y, and Z directions. This array is reflected by 30 cm of material 2 (water) on all faces, and weighted tracking (biasing) is to be used in the water reflector. The array spacing defines the perpendicular distance from the outer layer of spheres to the reflector to be 5 cm in the X, Y, and Z directions. The geometry input for this system is given below.

Data description 1, Example 9.

KENO V.a:

```
READ GEOM
UNIT 1
SPHERE     1  1   3.75
CUBE         0  1   5.0   -5.0
GLOBAL UNIT 2
ARRAY               1  -55.0  -25.0  -15.0
REFLECTOR    2  2  6*3.0  10
END GEOM
READ ARRAY  NUX=11 NUY=5 NUZ=3 END ARRAY
READ BIAS ID=500 2 11 END BIAS
```

KENO-VI:

```
READ GEOM
UNIT 1
SPHERE  1  3.75
CUBOID  2 6P5.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
GLOBAL UNIT 2
CUBOID  1  55.0  -55.0  25.0  -25.0  15.0  -15.0
CUBOID  2  58.0  -58.0  28.0  -28.0  18.0  -18.0
CUBOID  3  61.0  -61.0  31.0  -31.0  21.0  -21.0
CUBOID  4  64.0  -64.0  34.0  -34.0  24.0  -24.0
CUBOID  5  67.0  -67.0  37.0  -37.0  27.0  -27.0
```

(continues on next page)

(continued from previous page)

```
CUBOID  6  70.0  -70.0  40.0  -40.0  30.0  -30.0
CUBOID  7  73.0  -73.0  43.0  -43.0  33.0  -33.0
CUBOID  8  76.0  -76.0  46.0  -46.0  36.0  -36.0
CUBOID  9  79.0  -79.0  49.0  -49.0  39.0  -39.0
CUBOID 10  82.0  -82.0  52.0  -52.0  42.0  -42.0
CUBOID 11  85.0  -85.0  55.0  -55.0  45.0  -45.0
ARRAY 1 1 PLACE 6 3 2 3*0.0
MEDIA 2 2 2 -1
MEDIA 2 3 3 -2
MEDIA 2 4 4 -3
MEDIA 2 5 5 -4
MEDIA 2 6 6 -5
MEDIA 2 7 7 -6
MEDIA 2 8 8 -7
MEDIA 2 9 9 -8
MEDIA 2 10 10 -9
MEDIA 2 11 11 -10
BOUNDARY  11
END GEOM
READ ARRAY  NUX=11 NUY=5 NUZ=3 FILL F1 END FILL  END ARRAY
READ BIAS ID=500 2 11 END BIAS
```

In the KENO V.a input, the ARRAY boundary defines the origin of the REFLECTOR to be at the center of the ARRAY. The 6*3.0 in the REFLECTOR description repeats the 3.0 six times. The REFLECTOR record is used to generate ten REFLECTOR regions, each of which is 3.0 cm thick, on all six faces of the ARRAY.

In the KENO-VI input, the basic UNIT used to construct the ARRAY is defined in UNIT 1. The ARRAY is positioned in UNIT 2 (the GLOBAL UNIT) using the ARRAY card and the PLACE option. The ARRAY is then surrounded by ten REFLECTOR regions, each 3.0 cm thick, on all sides.

The first bias ID is 2, so the last bias ID will be 11 if 10 regions are created. The biasing data block is necessary to apply the desired weighting or biasing function to the reflector. The biasing material ID is obtained from Table 8.1.25. IDs, group structure and incremental thickness for weighting data available on the KENO weighting library. If the biasing data block is omitted from the problem description, the 10 reflector regions will not have a biasing function applied to them, and the default value of the average weight will be used. This may cause the problem to execute more slowly, and therefore require the use of more computer time.

EXAMPLE 10. Assume the reflector in Example 9 is present only on both X faces, both Y faces, and the negative Z face. The reflector is only 15.24 cm thick on these faces. The top of the array (positive Z face) is unreflected.

Data description 1, Example 10.

KENO V.a:

```
READ GEOM
UNIT 1
SPHERE    1  1   3.75
CUBE        0  1   5.0   -5.0
GLOBAL UNIT 2
ARRAY             1  -55.0   -25.0  -15.0
REFLECTOR  2  2  4*3.0    0.0   3.0  5
REFLECTOR  2 7  4*0.24   0.0  0.24  1
READ ARRAY NUX=11 NUY=5 NUZ=3 END ARRAY
READ BIAS ID=500 2 7 END BIAS
```

KENO-VI:

```
READ GEOM
UNIT 1
SPHERE  1 3.75
CUBOID  2 6P5.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
GLOBAL UNIT 2
CUBOID  1  55.0  -55.0  25.0  -25.0  15.0  -15.0
CUBOID  2  58.0  -58.0  28.0  -28.0  15.0  -18.0
CUBOID  3  61.0  -61.0  31.0  -31.0  15.0  -21.0
CUBOID  4  64.0  -64.0  34.0  -34.0  15.0  -24.0
CUBOID  5  67.0  -67.0  37.0  -37.0  15.0  -27.0
CUBOID  6  70.24 -70.24 40.24 -40.24 15.0  -30.24
ARRAY 1 1 PLACE 6 3 2 3*0.0
MEDIA 2 2 2 -1
MEDIA 2 3 3 -2
MEDIA 2 4 4 -3
MEDIA 2 5 5 -4
MEDIA 2 6 6 -5
BOUNDARY  6
END GEOM
READ ARRAY  NUX=11 NUY=5 NUZ=3 FILL F1 END FILL  END ARRAY
READ BIAS ID=500 2 6 END BIAS
```

In the KENO V.a input, the first REFLECTOR description generates five regions around the ARRAY, each region being 3.0 cm thick in the +X, -X, +Y, -Y, and -Z directions, and of zero thickness in the +Z direction. This defines a total thickness of 15 cm of reflector material on the appropriate faces. The second REFLECTOR description generates the last 0.24 cm of material 2 on those faces. Thus, the total reflector thickness is 15.24 cm on each face of the array, except the top which has no reflector. Five reflector regions were generated by the first REFLECTOR description, and one was generated by the second REFLECTOR description; so, six biasing regions must be defined in the biasing data. Thus, the beginning bias ID is 2, and the ending bias ID is 7.

In the KENO-VI input, the first CUBOID in Unit 2 represents the boundary for the ARRAY. The next four CUBOIDs represent the first four regions around the ARRAY, each region being 3.0 cm thick in the +X, X, +Y, -Y, and -Z directions, and of zero thickness in the +Z direction. A total thickness of 12 cm of reflector material is on the appropriate faces. The last CUBOID represents the last 3.24 cm of material 2 on those faces. Thus, the total reflector thickness is 15.24 cm on each face of the array, except the top which has no reflector. The beginning bias ID is 2, and the ending bias ID is 6. The last region could either be larger or smaller than the recommended thickness to complete the reflector.

The biasing material ID and thickness per region are obtained from Table 8.1.25. The thickness per region should be very nearly the thickness per region from the table to avoid over biasing in the reflector. Partial increments at the outer region of a reflector are exempt from this recommendation. If a biasing function is not to be applied to a region generated by the REFLECTOR record, the thickness per region can be any desired thickness and the biasing data block is omitted.

EXAMPLE 11. Assume the array of Example 7 has the central unit of the array replaced by a cylinder of material 4, 5 cm in radius and 10 cm tall. Assume a 20 cm thick spherical reflector of material 3 (concrete) is positioned so its inner radius is 65 cm from the center of the array. The minimum inner radius of a spherical reflector for this array is 62.25 cm ($\sqrt{55^2 + 25^2 + 15^2}$). If the inner radius is smaller than this, the problem cannot be described using KENO V.a geometry.

Data description 1, Example 11.

KENO V.a:

```
READ GEOM
UNIT 1
SPHERE  1  1  3.75
CUBE  0  1  5.0  -5.0
UNIT 2
CYLINDER  4  1  5.0  5.0  -5.0
CUBE  0  1  5.0  -5.0
GLOBAL UNIT 2
ARRAY     1 -55.0  -25.0  -15.0
SPHERE  0  1  65.0
REPLICATE  3  2  5.0  4
END GEOM
READ ARRAY
   NUX=11 NUY=5 NUZ=3
   LOOP 1  1  11  1  1 5 1  1 3 1  2  6 6 1  3 3 1  2 2 1  END LOOP
END ARRAY
READ BIAS ID=301 2 5 END BIAS
```

KENO-VI:

```
READ GEOM
UNIT 1
SPHERE  1 3.75
CUBOID   2 6P5.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 2
CYLINDER 1 5.0 5.0 -5.0
CUBOID  2  6P5.0
MEDIA 4 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
GLOBAL UNIT 3
CUBOID 1  55.0 -55.0 25.0 -25.0 15.0 -15.0
SPHERE 2 65.0
SPHERE 3 70.0
SPHERE 4 75.0
SPHERE 5 80.0
SPHERE 6 85.0
ARRAY 1 1 PLACE 6 3 2 3*0.0
MEDIA 0 1 2 -1
MEDIA 3 2 3 -2
MEDIA 3 3 4 -3
MEDIA 3 4 5 -4
MEDIA 3 5 6 -5
BOUNDARY  6
END GEOM
READ ARRAY
NUX=11 NUY=5 NUZ=3
LOOP 1  1  11  1  1 5 1  1 3 1 2  6 6 1  3 3 1  2 2 1 END LOOP
END ARRAY
READ BIAS ID=301 2 5 END BIAS
```

UNIT 1 describes the SPHERE and spacing used in the ARRAY. UNIT 2 defines the CYLINDER located at the center of the ARRAY. In KENO V.a, the ARRAY record defines the origin of the reflector to be at the center of the ARRAY, while in KENO-VI it defines the origin of the ARRAY to be at the center of the GLOBAL UNIT. The first SPHERE in the GLOBAL UNIT defines the inner radius of the reflector. The next four SPHERE and four MEDIA records of the KENO-VI input and the REPLICATE record of the KENO V.a input will generate four spherical regions of material 3, each 5.0 cm thick. The data for the BIAS block is generated in a similar manner to previous examples, except that concrete (ID=301) is used. The recommended reflector thickness is 5 cm; this thickness is incorporated explicitly in the KENO-VI model and with 4 repetitions of the 5 cm thick reflector via REPLICATE in the KENO V.a model. The first 10 entries following the word LOOP fills the 11 ×

5 × 3 ARRAY with UNITs 1. The next 10 entries position UNIT 2 at the center of the ARRAY (X = 6, Y = 3, and Z = 2), replacing the UNIT 1 that had been placed there by the first 10 entries.

EXAMPLE 12. Assume a data profile such as fission densities is desired in a cylinder at 0.5 cm intervals in the radial direction and 1.5 cm intervals axially. The cylinder is composed of material 1 and has a radius of 5 cm and a height of 15 cm. The REPLICATE or REFLECTOR description can be used to generate these regions in KENO V.a. A biasing data block is not entered because default biasing is desired throughout the cylinder.

Data description 1, Example 12.

KENO V.a:

```
READ GEOM
CYLINDER  1 1 0.5 1.5 0
REFLECTOR 1 1 0.5 1.5 0 9
END GEOM
```

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
CYLINDER 1 0.5 1.5 0
CYLINDER 2 1.0 3.0 0
CYLINDER 3 1.5 4.5 0
CYLINDER 4 2.0 6.0 0
CYLINDER 5 2.5 7.5 0
CYLINDER 6 3.0 9.0 0
CYLINDER 7 3.5 10.5 0
CYLINDER 8 4.0 12.0 0
CYLINDER 9 4.5 13.5 0
CYLINDER 10 5.0 15.0 0
MEDIA 1 1 1
MEDIA 1 1 2 -1
MEDIA 1 1 3 -2
MEDIA 1 1 4 -3
MEDIA 1 1 5 -4
MEDIA 1 1 6 -5
MEDIA 1 1 7 -6
MEDIA 1 1 8 -7
MEDIA 1 1 9 -8
MEDIA 1 1 10 -9
BOUNDARY  10
END GEOM
```

EXAMPLE 13. (KENO-VI due to pipe junctions) Assume a cross composed of two Plexiglas cylinders (material 3) having an inner diameter of 13.335 cm and an outer diameter of 16.19 cm. The bottom and side legs of the cross are closed by a 3.17 cm thick piece of Plexiglas. From the center of the intersection, the bottom and side legs are 91.44 cm long and the top leg is 121.92 cm long. The cross is filled with a UO_2F_2 solution (material 1) to a height of 28.93 cm above the center of the cylinder intersection. The cross is then surrounded by a water reflector (material 2) that extends from the center of the intersection: 111.74 cm in the ±X directions, 20.64 cm in the ±Y directions, 29.03 cm in the +Z direction, and -118.428 cm in the -Z direction. A schematic of the assembly is shown in Fig. 8.1.23.

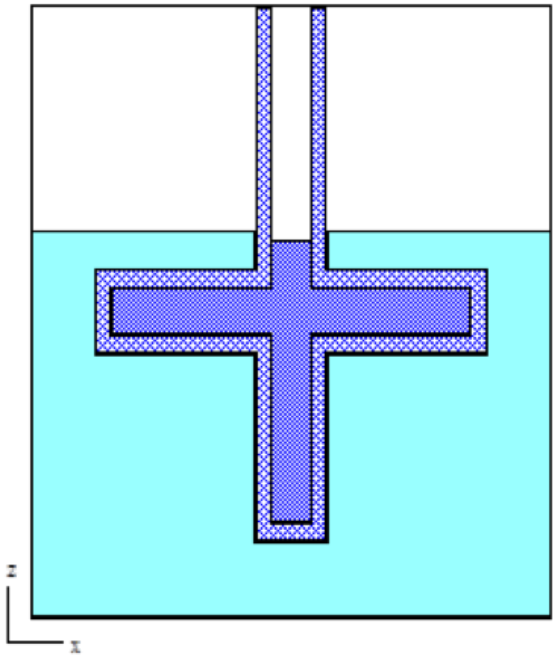


Fig. 8.1.23: Plexiglas UO_2F_2-filled cross.

Data description of Example 13 (KENO-VI only).

```
READ GEOMETRY
GLOBAL UNIT 1
CYLINDER    10 13.335  28.93 -88.27
CYLINDER    20 13.335 121.92 -88.27
CYLINDER    30 16.19  121.92 -91.44
YCYLINDER   40 13.335  88.27 -88.27
YCYLINDER   50 16.19 91.44 -91.44
CUBOID      60 2P111.74 2P20.64  29.03 -118.428
CUBOID      70 2P111.74 2P20.64 121.92 -118.428
MEDIA 1 1 10
MEDIA 0 1 20 -10
MEDIA 3 1 30 -20 -40
MEDIA 1 1 40 -10
MEDIA 3 1 50 -40 -30
MEDIA 2 1 60 -30 -50
MEDIA 0 1 70 -30 -60
BOUNDARY  70
END GEOMETRY
```

EXAMPLE 14. (KENO-VI only because of rotation) Assume a Y-shaped aluminum cylinder (material 2) with a 13.95 cm inner radius and a 0.16 cm wall thickness is filled with a UO_2F_2 solution (material 1). From the center where the Y intersects the cylinder, the bottom leg is 76.7 cm long, the top leg is 135.4 cm long, and the Y leg is 126.04 cm long, canted at a 29.26-degree angle. The bottom of the bottom leg and the top of the Y leg are sealed with 1.3 cm caps. The Y cylinder is filled to a height of 52.8 cm above the center where the Y leg intersects the vertical cylinder. The cylinder is surrounded by a water reflector (material 3) that extends out 37.0 cm in the ±pm`Y direction, and 135.4 and -99.6 in the ±.

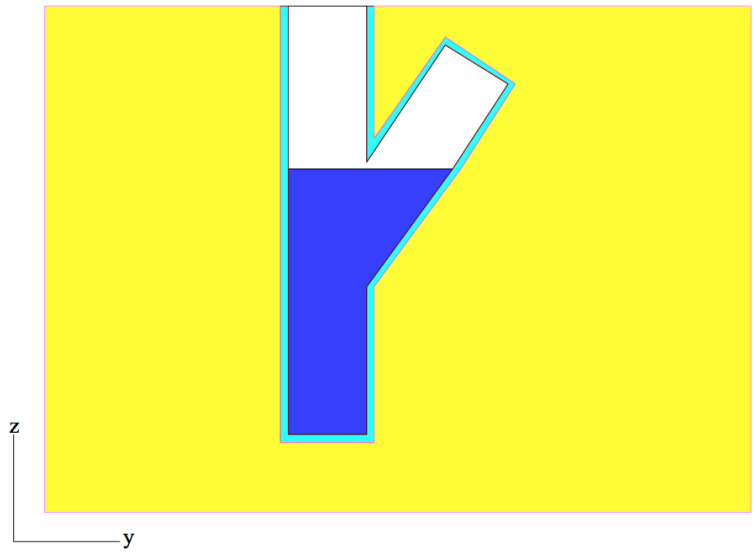


Fig. 8.1.24: Y-shaped UO_2F_2-filled aluminum cylinder.

Data description of Example 14 (KENO-VI only).

```
READ GEOMETRY
GLOBAL UNIT 1
COM='30 DEG Y CYLINDER'
CYLINDER 10 13.95 135.4 -75.4
CYLINDER 20 14.11 135.4 -76.7
CYLINDER 30 13.95 124.74  0.0  ROTATE A2=-29.26
CYLINDER 40 14.11 126.04  0.0 ROTATE A2=-29.26
CUBOID   50 2P37.0 100.0 -37.0 52.8 -75.4
CUBOID   60 2P37.0 100.0 -37.0 135.4 -99.6
MEDIA 1 1 10 50
MEDIA 2 1 20 -10 -30
MEDIA 1 1 30  50 -10
MEDIA 2 1 40 -30 -20
MEDIA 0 1 10 -50
MEDIA 0 1 30 -50 -10
MEDIA 3 1 60 -20 -40
BOUNDARY  60
END GEOMETRY
```

Use of holes in the geometry

Sect. 8.1.4.6 tells how each KENO V.a geometry region in a UNIT must completely enclose all previously described regions in that UNIT and how KENO-VI geometry allows regions in a UNIT to intersect, thus eliminating the need for HOLEs. HOLEs can be used to circumvent the complete enclosure restriction in KENO V.a to some degree. In KENO-VI, they can be useful in simplifying the input of a problem and decreasing the total CPU time needed for a problem. A HOLE is a means of placing an entire UNIT within a geometry region. A separate HOLE description is required for every location in a geometry region where a UNIT is to be placed. The information contained in a HOLE description is (1) the keyword HOLE, (2) the UNIT number of the UNIT to be placed, and (3) any modification data needed to correctly position and rotate (in KENO-VI) the specified UNIT within the containing UNIT. In KENO V.a, a HOLE is placed inside the geometry region that precedes it. This excludes HOLEs ... (i.e., if a CUBE geometry region is followed by four HOLE descriptions, all four HOLEs are located within the CUBE). In KENO V.a, HOLEs are subject to the restriction that they cannot intersect any other geometry region. HOLEs can be nested to any depth (see Sect. 8.1.4.6.2). It is

not advisable to use HOLEs tangent to other HOLEs or geometry, because round-off error may cause them to overlap. It is not uncommon for a problem that runs on one type of computer to fail on another type using the same data. Therefore, it is recommended that tangency and boundaries shared with HOLEs be avoided. This may be accomplished by separating the otherwise collocated or tangent surfaces by a very small (i.e., 10^{-6} cm) distance.

In KENO V.a, tracking in regions that contain holes is less efficient than tracking in regions that do not contain holes. Therefore, holes should be used only when the system cannot be easily described by conventional methods. One example of the use of holes is shown in Fig. 8.1.25, representing nine close-packed rods in an annulus.

In KENO-VI, tracking in regions that contain HOLEs can be more efficient than tracking in regions that do not contain HOLEs because every region boundary in a UNIT must be checked for a crossing whenever a crossing is possible. Putting small but complex geometries in a hole will lessen the number of boundaries that need to be checked for possible crossings. However, the indiscriminate use of holes is not advised since the particle must change coordinate systems every time a hole is entered or exited. Therefore, holes should be used carefully and only when the system can be simplified significantly by their use.

EXAMPLE 15. One example of a unit that requires holes in KENO V.a is better described not using holes in KENO-VI as shown in Fig. 8.1.25, representing nine close-packed rods in an annulus. The large rods are 1.4 cm in radius and composed of mixture 3. The small rods are 0.6 cm in radius and composed of mixture 1. The inside radius of the annulus is 3.6 cm, and the outside radius is 3.8 cm. The annulus is made of mixture 2. The rods and annulus are both 30 cm long. The annulus is centered in a cuboid having an 8 cm^2 cross section and a length of 32 cm. The black and gray areas in Fig. 8.1.25 are void.

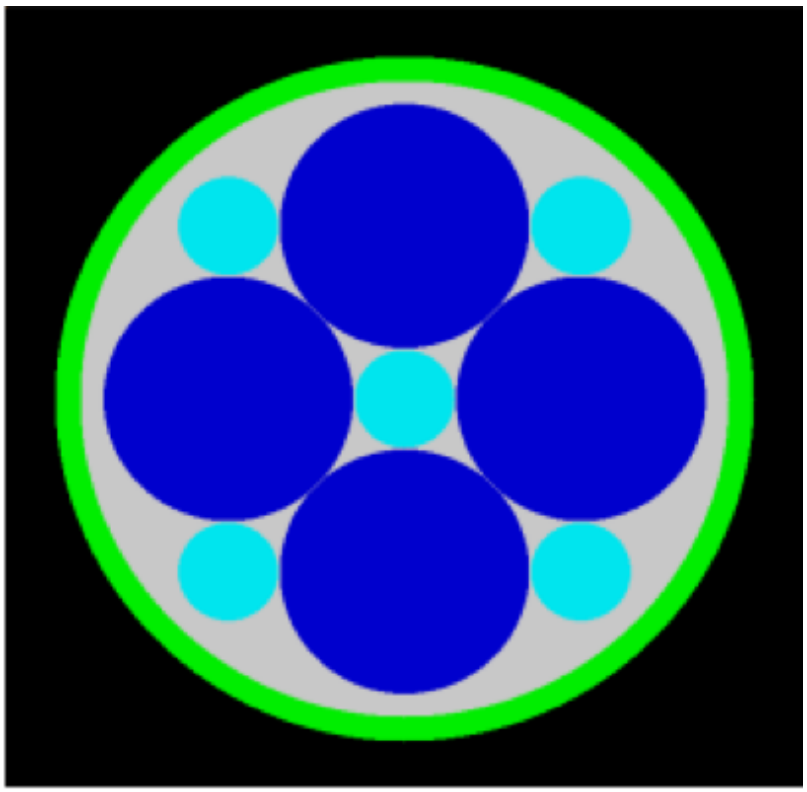


Fig. 8.1.25: Close-packed cylinders in an annulus.

Data description of Example 15.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER 1 1 0.6 2P15.0
UNIT 2
CYLINDER 3 1 1.4 2P15.0
GLOBAL UNIT 3
CYLINDER 1 1 0.6 2P15.0
CYLINDER 0 1 3.6 2P15.0
HOLE 2 0.0 -2.0 0.0
HOLE 1 2.0 -2.0 0.0
HOLE 2 2.0 0.0 0.0
HOLE 1 2.0 2.0 0.0
HOLE 2 0.0 2.0 0.0
HOLE 1 -2.0 2.0 0.0
HOLE 2 -2.0 0.0 0.0
HOLE 1 -2.0 -2.0 0.0
CYLINDER  2 1 3.8 2P15.0
CUBOID    0 1 4P4.0 2P16.0
END GEOM
```

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
CYLINDER 1 0.6 2P15.0
CYLINDER 2 0.6 2P15.0 ORIGIN X=2.0  Y=-2.0
CYLINDER 3 0.6 2P15.0 ORIGIN X=2.0  Y=2.0
CYLINDER 4 0.6 2P15.0 ORIGIN X=-2.0 Y=2.0
CYLINDER 5 0.6 2P15.0 ORIGIN X=-2.0 Y=-2.0
CYLINDER 6 1.4 2P15.0 ORIGIN X=2.0
CYLINDER 7 1.4 2P15.0 ORIGIN Y=2.0
CYLINDER 8 1.4 2P15.0 ORIGIN X=-2.0
CYLINDER 9 1.4 2P15.0 ORIGIN Y=-2.0
CYLINDER 10 3.6 2P15.0
CYLINDER 11 3.8 2P15.0
CUBOID  12  4P4.0 2P16.0
MEDIA 1 1 1
MEDIA 1 1 2
MEDIA 1 1 3
MEDIA 1 1 4
MEDIA 1 1 5
MEDIA 3 1 6
MEDIA 3 1 7
MEDIA 3 1 8
MEDIA 3 1 9
MEDIA 0 1 10 -1 -2 -3 -4 -5 -6 -7 -8 -9
MEDIA 2 1 11 -10
MEDIA 0 1 12 -11
BOUNDARY  12
END GEOM
```

The first HOLE description in the KENO V.a input represents the bottom large rod. It takes UNIT 2 and places its ORIGIN at (0.0,-2.0,0.0) relative to the ORIGIN of UNIT 3. The second HOLE description represents the small rod to the right of the large rod just discussed. It places the origin of UNIT 1 at (2.0,-2.0,0.0) in UNIT 3. The third HOLE description represents the large rod to the right. It places the origin of UNIT 2 at (2.0,0.0,0.0) in UNIT 3. This procedure is repeated in a counterclockwise direction until all eight rods have been placed within the region that defines the inner surface of the annulus. The CYLINDER that defines the outer surface of the annulus is described after all the HOLEs for the previous region have been placed. Then the outer CUBOID is described. This example illustrates that a UNIT that is to be placed using a HOLE description need not have

a CUBE or CUBOID as its last region. Note that including the central rod directly in UNIT 3 reduces the CPU time required for transport compared to the case of all 9 rods being inserted as HOLEs. It is also important that the 9 HOLEs are inserted *after* the void cylinder into which they are inserted. Entering the HOLEs in any other position in the input would generated region intersection errors. The order of the HOLE records in any given region is not important, as they can be interchanged with each other randomly. However, they must always appear immediately after the region in which they are placed.

The KENO-VI input does not need to use HOLEs. The first CYLINDER description in this case represents the middle small rod. The next four CYLINDER records describe the four remaining small rods surrounding the middle rod. The ORIGIN attribute is used to shift the origin of each CYLINDER to the appropriate location. The following four CYLINDER records represent the four large rods. Again, the ORIGIN attribute is used to shift the ORIGIN of each CYLINDER to the appropriate location. Only the nonzero dimensions need to be entered in the ORIGIN data. The tenth CYLINDER record is the void in the annulus that contains the rods. The last CYLINDER record defines the outer surface of the annulus. Finally, the CUBOID record describes the surrounding UNIT boundary.

In KENO-VI, holes may not extend across any array outer boundary, may not intersect with other holes, and may not cross the host UNIT outer boundary. Thus a hole may be placed so that it crosses several regions within an array. The hole description replaces the unit description within the hole domain. Since the holes are placed using the host UNIT coordinate system, the location of the hole record in the unit definition is not relevant.

An array of the arrangement shown in Fig. 8.1.25 can be easily described by altering the array description data. For example, a $5 \times 3 \times 2$ array of these shapes with a center-to-center spacing of 8 cm in X and Y and 32 cm in Z can be achieved by using the following array data:

```
READ ARRAY  NUX=5 NUY=3 NUZ=2  FILL F3  END FILL  END ARRAY
```

or

```
READ ARRAY  NUX=5 NUY=3 NUZ=2  FILL 30*3  END FILL  END ARRAY
```

or

```
READ ARRAY  NUX=5 NUY=3 NUZ=2  LOOP  3 1 5 1 1 3 1 1 2 1  END LOOP END ARRAY
```

Nesting holes

This section illustrates how holes are nested. Holes can be nested to any level. Consider the configuration illustrated in Fig. 8.1.25 and replace the large rods with a complicated geometric arrangement. The resulting Fig. is shown in Fig. 8.1.26. Fig. 8.1.27 shows the complicated geometric arrangement that replaced the large rods of Fig. 8.1.25. Fig. 8.1.28 shows a component of the arrangement shown in Fig. 8.1.26.

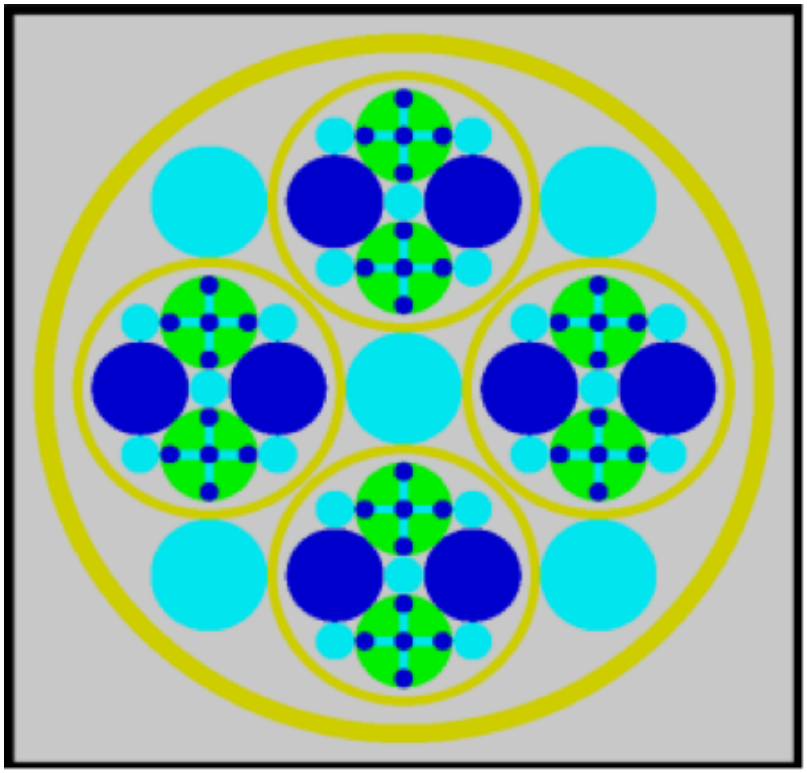


Fig. 8.1.26: Configuration using nested holes.

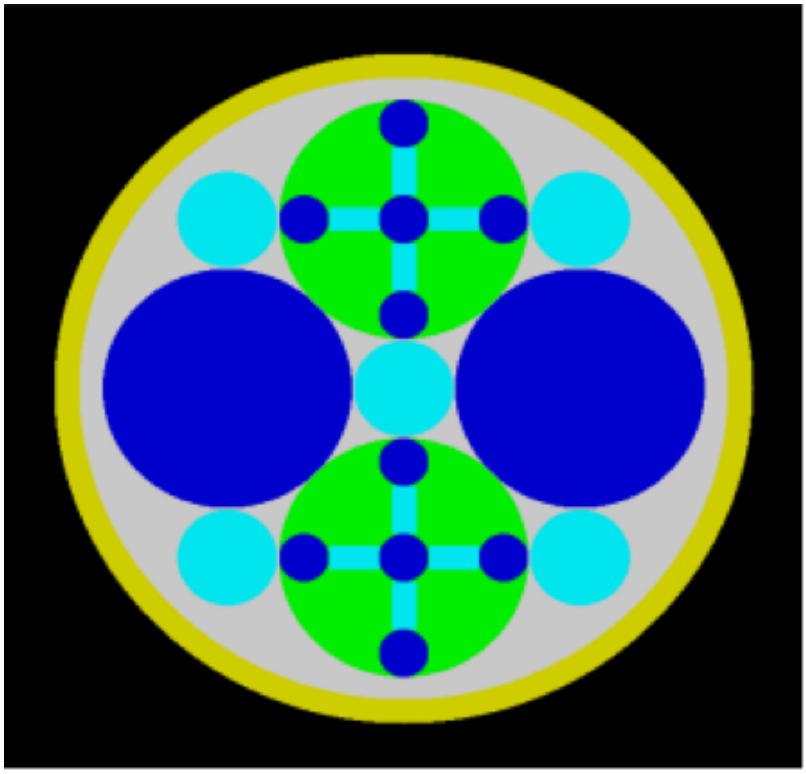


Fig. 8.1.27: Complicated geometric arrangement by Unit 7.

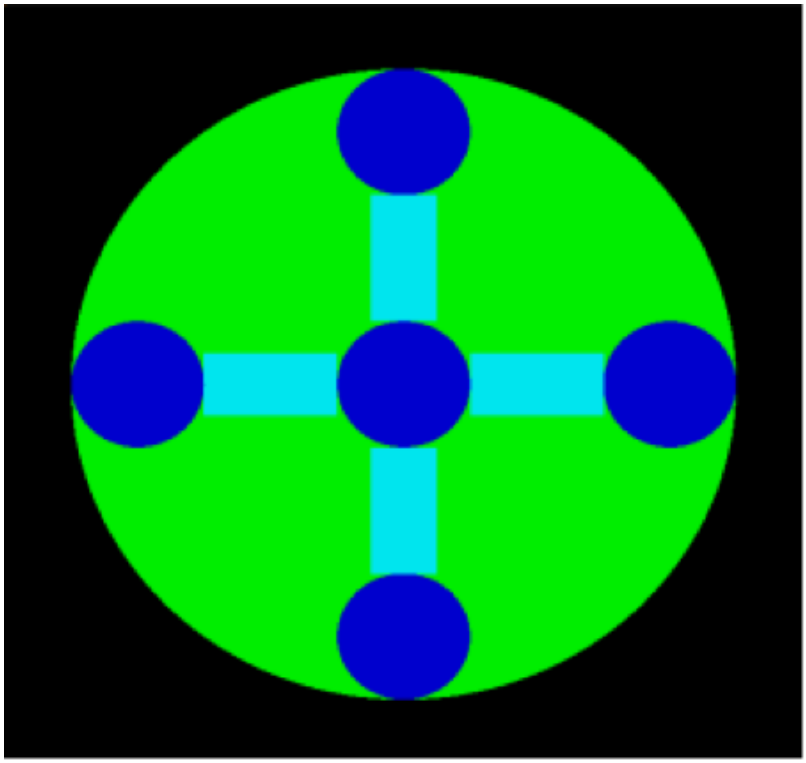


Fig. 8.1.28: Geometric component represented by Unit 4.

EXAMPLE 16. There is no predetermined preferred method to create a geometry mockup for a given physical system. The user should determine the most convenient order. To describe the configuration shown in Fig. 8.1.26 using nested HOLEs, it is likely most convenient to start the geometry mockup at the deepest nesting level, as shown in Fig. 8.1.27. The small CYLINDERs are composed of mixture 1, and they are each 0.1 cm in radius and 30 cm long. There are five small CYLINDERs used in Fig. 8.1.28. Their centers are located at (0,0,0) for the central one, at (0,-0.4,0) for the bottom one, at (0.4,0,0) for the right one, at (0,0.4,0) for the top one, and at (-0.4,0,0) for the left one. The rectangular parallelepipeds (CUBOIDs) are composed of mixture 2. Each one is 30 cm long and 0.1 cm by 0.2 cm in cross section. The large CYLINDER containing the configuration is composed of mixture 3, is 30 cm long and has a radius of 0.5 cm.

A possible geometry mockup for this system is described as follows in KENO V.a:

(1) define a small cylinder to be UNIT 1,

(2) define a small CUBOID with its length in the X direction to be UNIT 2,

(3) define a small CUBOID with its length in the Y direction to be UNIT 3,

(4) define UNIT 4 to be the large cylinder and place the CYLINDERs and CUBOIDs in it using HOLEs.

```
UNIT 1
CYLINDER 1 1 0.1 2P15.0
UNIT 2
CUBOID 2 1 2P0.1 2P0.05 2P15.0
UNIT 3
CUBOID 2 1 2P0.05 2P0.1 2P15.0
UNIT 4
CYLINDER  1 1 0.1 2P15.0
CYLINDER  3 1 0.5 2P15.0
HOLE 1 0.0 -0.4 0.0
HOLE 1 0.4 0.0 0.0
HOLE 1 0.0 0.4 0.0
HOLE 1 -0.4 0.0 0.0
HOLE 2 -0.2 0.0 0.0
HOLE 2 0.2 0.0 0.0
HOLE 3 0.0 -0.2 0.0
HOLE 3 0.0 0.2 0.0
```

The first cylinder description in UNIT 4 places the central rod,

the second cylinder description in UNIT 4 places the outer cylinder,

the *first* HOLE places the bottom CYLINDER,

the *second* HOLE places the CYLINDER at the right,

the *third* HOLE places the top CYLINDER,

the *fourth* HOLE places the CYLINDER at the left,

the *fifth* HOLE places the left CUBOID whose length is in X,

the *sixth* HOLE places the right CUBOID whose length is in X,

the *seventh* HOLE places the bottom CUBOID whose length is in Y, and

the *eighth* HOLE places the top CUBOID whose length is in Y.

A possible geometry mockup for this system is described as follows in KENO-VI:

1. define UNIT 1 to contain the five small cylinders and four blocks,

2. define UNIT 2 to contain the next two larger-sized cylinders and UNIT 1 as HOLEs, and
3. define GLOBAL UNIT 3 to contain the large cylinders and UNIT 2 as HOLEs.

```
UNIT 1
CYLINDER  1 0.1 2P15.0
CYLINDER  2 0.1 2P15.0 ORIGIN Y=-0.4
CYLINDER  3 0.1 2P15.0 ORIGIN X=0.4
CYLINDER  4 0.1 2P15.0 ORIGIN Y=0.4
CYLINDER  5 0.1 2P15.0 ORIGIN X=-0.4
CUBOID    6 -0.3 -0.1  2P0.05 2P15.0
CUBOID    7 0.3 0.1  2P0.05 2P15.0
CUBOID    8 2P0.05 -0.3 -0.1  2P15.0
CUBOID    9 2P0.05 0.3 0.1  2P15.0
CYLINDER 10 0.5 2P15.0
MEDIA 1 1 1
MEDIA 1 1 2
MEDIA 1 1 3
MEDIA 1 1 4
MEDIA 1 1 5
MEDIA 2 1 6
MEDIA 2 1 7
MEDIA 2 1 8
MEDIA 2 1 9
MEDIA 3 1 10 -1 -2 -3 -4 -5 -6 -7 -8 -9
BOUNDARY  10
```

geometry record *1* places the central rod,

geometry record *2* places the bottom CYLINDER,

geometry record *3* places the CYLINDER at the right,

geometry record *4* places the top CYLINDER,

geometry record *5* places the CYLINDER at the left,

geometry record *6* places the left CUBOID whose length is in X,

geometry record *7* places the right CUBOID whose length is in X,

geometry record *8* places the bottom CUBOID whose length is in Y,

geometry record *9* places the top CUBOID whose length is in Y, and

geometry record *10* is the surrounding CYLINDER that defines the unit boundary.

In Fig. 8.1.27, the large plain cylinders are composed of mixture 1 and are 0.5 cm in radius and 30 cm long. The cylindrical component of UNIT 4 for KENO V.a or UNIT 1 for KENO-VI is the same size: an outer radius of 0.5 cm and a length of 30 cm. The small cylinders located in the interstices between the large cylinders are composed of mixture 2, are 0.2 cm in radius, and are 30 cm long. The annulus is composed of mixture 4, has a 1.3 cm inside radius and a 1.4 cm outer radius. The volume between the inner cylinders is void. The large cylinders each have a radius of 0.5 cm and are tangent. Therefore, their origins are offset from the origin of the UNIT by 0.707107. This is from $X^2 + Y^2 = 1.0$, where X and Y are equal.

For KENO V.a, define UNIT 5 to be the large plain cylinder, UNIT 6 to be the small cylinder, and UNIT 7 as the annulus that contains the cylinders. Its origin is at its center. The geometry mockup for this portion of the problem follows:

KENO V.a:

```
UNIT 5
CYLINDER  1 1 0.5 2P15.0
UNIT 6
CYLINDER  2 1 0.2 2P15.0
UNIT 7
CYLINDER  2 1 0.2 2P15.0
CYLINDER  0 1 1.3 2P15.0
HOLE      5 0.707107 0.0 0.0
HOLE      6 0.707107 0.707107 0.0
HOLE      4 0.0 0.707107 0.0
HOLE      6 -0.707107 0.707107 0.0
HOLE      5 -0.707107 0.0 0.0
HOLE      6 -0.707107 -0.707107 0.0
HOLE      4 0.0 -0.707107 0.0
HOLE      6 0.707107 -0.707107 0.0
CYLINDER  4 1 1.4 2P15.0
```

The *first* HOLE places the larger CYLINDER of mixture 1 at the right with its origin at (0.707107,0.0,0.0),

the *second* HOLE places the small CYLINDER in the upper right quadrant,

the *third* HOLE places the top CYLINDER that contains the geometric component defined in UNIT 4,

the *fourth* HOLE places the small CYLINDER in the upper left quadrant,

the *fifth* HOLE places the larger CYLINDER of mixture 1 at the left,

the *sixth* HOLE places the small CYLINDER in the lower lower left quadrant,

the *seventh* HOLE places the bottom CYLINDER that contains the geometric component defined in UNIT 4, and

the *eighth* HOLE places the small CYLINDER in the lower right quadrant.

The last CYLINDER defines the outer surface of the annulus.

For KENO-VI, UNIT 2 is the annulus that contains the cylinders.

KENO-VI:

```
UNIT 2
CYLINDER  1 0.2 2P15.0
CYLINDER  2 0.2 2P15.0 ORIGIN X=0.707107  Y=0.707107
CYLINDER  3 0.2 2P15.0 ORIGIN X=-0.707107 Y=0.707107
CYLINDER  4 0.2 2P15.0 ORIGIN X=-0.707107 Y=-0.707107
CYLINDER  5 0.2 2P15.0 ORIGIN X=0.707107  Y=-0.707107
CYLINDER  6 0.5 2P15.0 ORIGIN X=0.707107
CYLINDER  7 0.5 2P15.0 ORIGIN X=-0.707107
CYLINDER 10 1.3 2P15.0
CYLINDER 11 1.4 2P15.0
MEDIA 2 1 1
MEDIA 2 1 2
MEDIA 2 1 3
MEDIA 2 1 4
MEDIA 2 1 5
MEDIA 1 1 6
MEDIA 1 1 7
HOLE 1 ORIGIN Y=0.707107
HOLE 1 ORIGIN Y=-0.707107
MEDIA 0 1 10 -1 -2 -3 -4 -5 -6 -7
MEDIA 4 1 11 -10
BOUNDARY  11
```

In Unit 2 of the above KENO-VI geometry,

CYLINDER *1* places a small CYLINDER of mixture 2 at the origin,

CYLINDER *2* places the small CYLINDER of mixture 2 in the upper right quadrant,

CYLINDER *3* places the small CYLINDER of mixture 2 in the upper left quadrant,

CYLINDER *4* places the small CYLINDER of mixture 2 in the lower left quadrant,

CYLINDER *5* places the small CYLINDER of mixture 2 in the lower right quadrant,

CYLINDER *6* places the larger CYLINDER of mixture 1 at the right with its origin at (0.707107,0.0,0.0),

CYLINDER *7* places the larger CYLINDER of mixture 1 at the left with its origin at (0.0,0.707107.0.0),

CYLINDER *10* defines the inner surface of the annulus,

CYLINDER *11* defines the outer surface of the annulus and the UNIT boundary,

the first HOLE places the top CYLINDER that contains the geometric component defined in UNIT 1, and

the second HOLE places the bottom CYLINDER that contains the geometric component defined in UNIT 1.

To complete the geometry mockup, consider Fig. 8.1.26.

For KENO V.a geometry, define UNIT 8 to be a cylinder of mixture 2 having a radius of 0.6 cm and a length of 30 cm. Define UNIT 9 to be the central rod and the large annulus of 3.6 cm inner radius, 3.8 cm outer radius, and 30 cm length centered in a CUBOID having an 8 cm^2 cross section and being 32 cm long.

KENO V.a:

```
UNIT 8
CYLINDER  2 1 0.6 2P15.0
UNIT 9
CYLINDER  2 1 0.6 2P15.0
CYLINDER  0 1 3.6 2P15
HOLE   7 2.0 0.0 0.0
HOLE   8 2*2.0 0.0
HOLE   7 0.0 2.0 0.0
HOLE   8 -2.0 2.0 0.0
HOLE   7 -2.0 2*0.0
HOLE   8 2*-2.0 0.0
HOLE   7 0.0 -2.0 0.0
HOLE   8 2P2.0 0.0
CYLINDER  4 1 3.8 2P15.0
CUBOID    0 1 4P4.0 2P16.0
```

In UNIT 9 of the KENO V.a description, the first CYLINDER defines the rod of mixture 2, centered in the annulus. The second CYLINDER defines the void volume between the central rod and the annulus.

The *first* HOLE places the composite annulus of UNIT 7 to the right of the central rod,

the *second* HOLE places a rod defined by UNIT 8 in the upper right quadrant of the annulus,

the *third* HOLE places the composite annulus of UNIT 7 above the central rod,

the *fourth* HOLE places a rod defined by UNIT 8 in the upper left quadrant of the annulus,

the *fifth* HOLE places the composite annulus of UNIT 7 to the left of the central rod,

the *sixth* HOLE places a rod defined by UNIT 8 in the lower left quadrant,

the *seventh* HOLE places the composite annulus of UNIT 7 below the central rod, and

the *eighth* HOLE places a rod defined by UNIT 8 in the lower right quadrant.

The last CYLINDER defines the outer surface of the annulus. The outer CUBOID is the last region.

For KENO-VI geometry, define UNIT 3 to be the central rod and four outer rods of 0.6 cm radius and 30.0 cm length, and the large annulus of 3.6 cm inner radius, 3.8 cm outer radius, and 30 cm length centered in a cuboid having an 8 cm^2 cross section and a length of 32 cm.

KENO-VI:

```
GLOBAL UNIT 3
CYLINDER  1 0.6 2P15.0
CYLINDER  2 0.6 2P15.0 ORIGIN X=2.0 Y=2.0
CYLINDER  3 0.6 2P15.0 ORIGIN X=-2.0 Y=2.0
CYLINDER  4 0.6 2P15.0 ORIGIN X=-2.0 Y=-2.0
CYLINDER  5 0.6 2P15.0 ORIGIN X=2.0 Y=-2.0
CYLINDER 10 3.6 2P15.0
CYLINDER 11 3.8 2P15.0
CUBOID   12 4P4.0 2P16.0
MEDIA 2 1 1
MEDIA 2 1 2
MEDIA 2 1 3
MEDIA 2 1 4
MEDIA 2 1 5
HOLE  2 ORIGEN X=2
HOLE  2 ORIGEN Y=2
HOLE  2 ORIGEN X=-2
HOLE  2 ORIGEN Y=-2
MEDIA 0 1 10 -1 -2 -3 -4 -5
MEDIA 4 1 11 -10
MEDIA 0 1 12 -11
BOUNDARY  12
```

In UNIT 3 of the above KENO-VI description,

CYLINDER *1* defines the rod of mixture 2, centered in the annulus,

CYLINDER *2* places a rod of mixture 2 in the upper right quadrant of the annulus,

CYLINDER *3* places a rod of mixture 2 in the upper left quadrant of the annulus,

CYLINDER *4* places a rod of mixture 2 in the lower left quadrant,

CYLINDER *5* places a rod of mixture 2 in the lower right quadrant,

CYLINDER *10* defines the void volume between the central rod and the annulus,

CYLINDER *11* defines the outer surface of the annulus,

CUBOID *12* defines the unit boundary,

the first HOLE places UNIT 2 to the right of the central rod,

the second HOLE places UNIT 2 above the central rod,

the third HOLE places UNIT 2 to the left of the central rod, and

the fourth HOLE places UNIT 2 below the central rod.

This problem illustrates three levels of HOLE nesting. The total input data for the problem is given below. The geometry description accurately recreates the geometry arrangement of Fig. 8.1.26. The 2D color plot output is shown in Fig. 8.1.29.

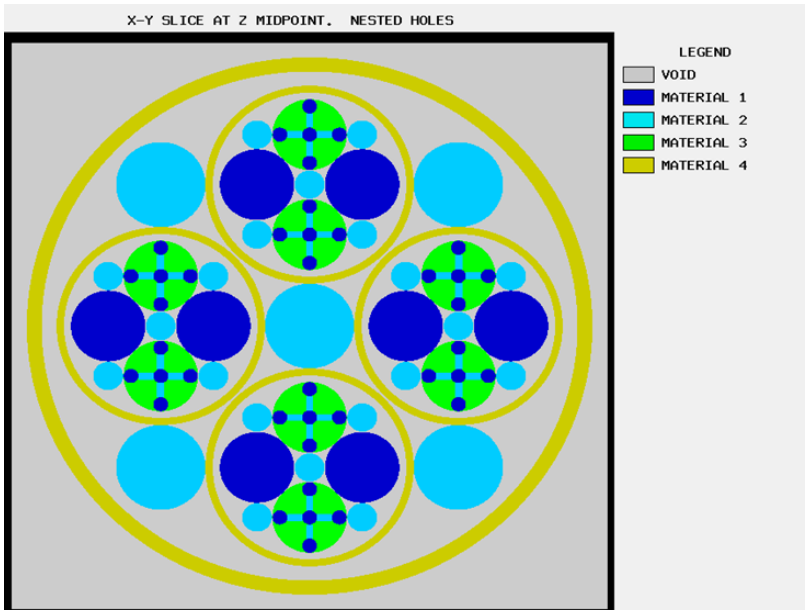


Fig. 8.1.29: Color plot of nested holes example problem.

KENO V.a:

NESTED HOLES SAMPLE

```
READ GEOM
UNIT 1
CYLINDER 1 1 0.1 2P15.0
UNIT 2
CUBOID 2 1 2P0.1 2P0.05 2P15.0
UNIT 3
CUBOID 2 1 2P0.05 2P0.1 2P15.0
UNIT 4
CYLINDER 1 1 0.1 2P15.0
CYLINDER 3 1 0.5 2P15.0
HOLE 1  0.0 -0.4 0.0
HOLE 1  0.4  0.0 0.0
HOLE 1  0.0  0.4 0.0
HOLE 1 -0.4  0.0 0.0
HOLE 2 -0.2  0.0 0.0
HOLE 2  0.2  0.0 0.0
HOLE 3  0.0 -0.2 0.0
HOLE 3  0.0  0.2 0.0
UNIT 5
CYLINDER 1 1 0.5 2P15.0
UNIT 6
CYLINDER 2 1 0.2 2P15.0
UNIT 7
CYLINDER 2 1 0.2 2P15.0
CYLINDER 0 1 1.3 2P15.0
HOLE 5 0.707107 2*0.0
HOLE 6 0.707107 0.707107 0.0
HOLE 4 0.0 0.707107 0.0
HOLE 6 -0.707107 0.707107 0.0
HOLE 5 -0.707107 0.0 0.0
HOLE 6 -0.707107 -0.707107 0.0
HOLE 4 0.0 -0.707107 0.0
HOLE 6 0.707107 -0.707107 0.0
CYLINDER 4 1 1.4 2P15.0
```

(continues on next page)

(continued from previous page)

```
UNIT 8
CYLINDER 2 1 0.6 2P15.0
GLOBAL UNIT 9
CYLINDER 2 1 0.6 2P15.0
CYLINDER 0 1 3.6 2P15.0
HOLE 7 2.0 0.0 0.0
HOLE 8 2*2.0 0.0
HOLE 7 0.0 2.0 0.0
HOLE 8 -2.0 2.0 0.0
HOLE 7 -2.0 2*0.0
HOLE 8 2*-2.0 0.0
HOLE 7 0.0 -2.0 0.0
HOLE 8 2P2.0 0.0
CYLINDER 4 1 3.8 2P15.0
CUBOID 0 1 4P4.0 2P16.0
END GEOM
READ PLOT
  TTL='X-Y SLICE AT Z MIDPOINT. NESTED HOLES'
  XUL=-0.1  YUL=8.1 ZUL=16.0
  XLR=8.1 YLR=-0.1 ZLR=16
  UAX=1.0 VDN=-1.0  NAX=260 NCH=' *-.X' SCR=NO
END PLOT
END DATA
END
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER  1 0.1 2P15.0
CYLINDER  2 0.1 2P15.0 ORIGIN Y=-0.4
CYLINDER  3 0.1 2P15.0 ORIGIN X=0.4
CYLINDER  4 0.1 2P15.0 ORIGIN Y=0.4
CYLINDER  5 0.1 2P15.0 ORIGIN X=-0.4
CUBOID    6 -0.3 -0.1  2P0.05 2P15.0
CUBOID    7 0.3 0.1  2P0.05 2P15.0
CUBOID    8 2P0.05 -0.3 -0.1  2P15.0
CUBOID    9 2P0.05 0.3 0.1  2P15.0

CYLINDER 10 0.5 2P15.0
MEDIA 1 1 1 -6 -7 -8 -9
MEDIA 1 1 2 -8
MEDIA 1 1 3 -7
MEDIA 1 1 4 -9
MEDIA 1 1 5 -6
MEDIA 2 1 6 -1 -5
MEDIA 2 1 7 -1 -3
MEDIA 2 1 8 -1 -2
MEDIA 2 1 9 -1 -4
MEDIA 3 1 -1 -2 -3 -4 -5 -6 -7 -8 -9 10
BOUNDARY  10
UNIT 2
CYLINDER  1 0.2 2P15.0
CYLINDER  2 0.2 2P15.0 ORIGIN X=0.707107  Y=0.707107
CYLINDER  3 0.2 2P15.0 ORIGIN X=-0.707107 Y=0.707107
CYLINDER  4 0.2 2P15.0 ORIGIN X=-0.707107 Y=-0.707107
CYLINDER  5 0.2 2P15.0 ORIGIN X=0.707107  Y=-0.707107
CYLINDER  6 0.5 2P15.0 ORIGIN X=0.707107
CYLINDER  7 0.5 2P15.0 ORIGIN X=-0.707107
CYLINDER 10 1.3 2P15.0
CYLINDER 11 1.4 2P15.0
MEDIA 2 1 1
MEDIA 2 1 2
MEDIA 2 1 3
MEDIA 2 1 4
```

(continues on next page)

(continued from previous page)

```
MEDIA 2 1 5
MEDIA 1 1 6
MEDIA 1 1 7
HOLE  1 ORIGIN Y=0.707107
HOLE  1 ORIGIN Y=-0.707107
MEDIA 0 1 10 -1 -2 -3 -4 -5 -6 -7
MEDIA 4 1 11 -10
BOUNDARY  11
GLOBAL UNIT 3
CYLINDER  1 0.6 2P15.0
CYLINDER  2 0.6 2P15.0 ORIGIN X=2.0  Y=2.0
CYLINDER  3 0.6 2P15.0 ORIGIN X=-2.0 Y=2.0
CYLINDER  4 0.6 2P15.0 ORIGIN X=-2.0 Y=-2.0
CYLINDER  5 0.6 2P15.0 ORIGIN X=2.0  Y=-2.0
CYLINDER 10 3.6 2P15.0
CYLINDER 11 3.8 2P15.0
CUBOID   12 4P4.0 2P16.0
MEDIA 2 1 1
MEDIA 2 1 2
MEDIA 2 1 3
MEDIA 2 1 4
MEDIA 2 1 5
HOLE 2 ORIGIN X=2
HOLE 2 ORIGIN Y=2
HOLE 2 ORIGIN X=-2
HOLE 2 ORIGIN Y=-2
MEDIA 0 1 10 -1 -2 -3 -4 -5
MEDIA 4 1 11 -10
MEDIA 0 1 12 -11
BOUNDARY  12
END GEOM
READ PLOT
  TTL='X-Y SLICE AT Z MIDPOINT. NESTED HOLES'
  XUL=-4.1  YUL=4.1 ZUL=0.0
  XLR=4.1 YLR=-4.1 ZLR=0
  UAX=1.0 VDN=-1.0  NAX=800
END PLOT
END DATA
END
```

Multiple arrays

EXAMPLE 17. Sect. 8.1.4.6 demonstrates how UNITs are composed of geometry regions and how these UNITs can be stacked in an ARRAY. This same procedure can be extended to create multiple ARRAYs. Furthermore, ARRAYs can be used as building blocks within other ARRAYs.

Consider Sample Problem 19 from Sect. 8.1.8.3. This problem is a critical experiment consisting of a composite array [Tho64, Tho73] of four highly enriched uranium metal cylinders and four cylindrical Plexiglas containers filled with uranyl nitrate solution. A photograph of the experiment is given in Fig. 8.1.235. The coordinate system is defined to be Z up the page, Y across the page, and X out of the page.

The Plexiglas containers have an inside radius of 9.525 cm and an outside radius of 10.16 cm. The inside height is 17.78 cm, and the outside height is 19.05 cm. Four of these containers are stacked with a center-to-center spacing of 21.75 cm in the Y direction and 20.48 cm in the Z direction (vertical). This arrangement of four Plexiglas containers can be described as follows: mixture 2 is the uranyl nitrate and mixture 3 is Plexiglas, so the Plexiglas container with its appropriate spacing CUBOID can be described as UNIT 1. This considers the ARRAY to be bare and suspended with no supports.

KENO V.a:

```
UNIT 1
CYLINDER  2 1 9.525  2P8.89
CYLINDER  3 1 10.16  2P9.525
CUBOID    0 1 4P10.875  2P10.24
```

KENO-VI:

```
UNIT 1
CYLINDER 1  9.525  2P8.89
CYLINDER 2  10.16  2P9.525
CUBOID  3   4P10.875  2P10.24
MEDIA 2 1 1
MEDIA 3 1 2 -1
MEDIA 0 1 3 -2
BOUNDARY  3
```

The ARRAY of four Plexiglas containers can be described as ARRAY 1 in the array data as follows:

```
ARA=1 NUX=1 NUY=2 NUZ=2 FILL F1 END FILL
```

The four metal cylinders, comprised of mixture 1, each have a radius of 5.748 cm and are 10.765 cm tall. They have a center-to-center spacing of 13.18 cm in the Y direction and 12.45 cm in the Z direction (vertical). Thus, one of the metal cylinders with its appropriate spacing CUBOID can be described as UNIT 2. This ARRAY is also considered to be bare and unsupported.

KENO V.a:

```
UNIT 2
CYLINDER 1 1  5.748 2P5.3825
CUBOID   0 1  4P6.59 2P6.225
```

KENO-VI:

```
UNIT 2
CYLINDER 1 5.748 2P5.3825
CUBOID   2   4P6.59 2P6.225
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
```

The array of four metal cylinders can be described as ARRAY 2 in the array data.

```
ARA=2 NUX=1 NUY=2 NUZ=2 FILL F2 END FILL
```

Now two ARRAYs have been described. The overall dimensions of the ARRAY of Plexiglas containers are 21.75 cm in X, 43.5 cm in Y, and 40.96 cm in Z. The overall dimensions of the ARRAY of metal cylinders are 13.18 cm in X, 26.36 cm in Y, and 24.9 cm in Z.

In order to describe the composite ARRAY, these two ARRAYs must be positioned within UNITs and stacked together into one ARRAY. In order for them to be stacked into one ARRAY, the adjacent faces must match. This is accomplished by defining a UNIT 3 which contains ARRAY 1, the ARRAY of Plexiglas solution containers. The overall dimensions of this UNIT are 21.75 cm in X, 43.5 cm in Y, and 40.96 cm in Z. These dimensions are calculated by the code and need not be specified. UNIT 3 is defined as follows:

KENO V.a:

```
UNIT 3
ARRAY 1  3*0.0
```

KENO-VI:

```
UNIT 3
CUBOID  1 2P10.875 2P21.75 2P20.48
ARRAY 1 1 PLACE 1 1 1 0.0 -10.875 -10.24
BOUNDARY  1
```

The ARRAY of metal cylinders will be defined to be UNIT 4. However, this ARRAY is 17.14 cm smaller in the Y and 16.06 cm smaller in the Z dimensions than the ARRAY of Plexiglas UNITs. Therefore, a void region must be placed around the ARRAY in those directions so UNIT 4 and UNIT 3 will be the same size in Y and Z.

KENO V.a:

```
UNIT 4
ARRAY  2  3*0.0
REPLICATE  0 1 2*0.0  2*8.57  2*8.03  1
```

KENO-VI:

Now that UNIT 3 and UNIT 4 have been defined, they must be placed in the global or universe ARRAY to define the physical arrangement of the eight pieces. This procedure is implemented via a GLOBAL ARRAY in KENO V.a, while KENO-VI uses a GLOBAL UNIT 3 as follows:

KENO V.a:

```
GBL=3  ARA=3  NUX=2  NUY=1  NUZ=1  FILL  4 3  END FILL
```

KENO-VI:

```
GLOBAL UNIT 5
CUBOID 1 34.93 0.0 43.5 0.0 40.96 0.0
ARRAY 3 1 PLACE 1 1 1 6.59 21.75 20.48
BOUNDARY  1
```

The description of ARRAY 3 in KENO-VI is identical to that shown for KENO V.a.

This completes the geometry description for the problem. The complete geometry input description for the problem is given below.

KENO V.a:

```
=KENOVA
SAMPLE PROBLEM 19 4 AQUEOUS 4 METAL ARRAY OF ARRAYS
READ PARAM  LIB=4  RUN=NO END PARAM
READ MIXT SCT=1
     MIX=1
  1092238     3.2275e-3
  1092235     4.4802e-2
     MIX=2
  20011023    5.81e-2
  2007014     1.9753e-3
  2008016     3.6927e-2
  20092235    9.8471e-4
  20092238    7.7697e-5
     MIX=3
  11006012    3.5552e-2
  11011023    5.6884e-2
  11008016    1.4221e-2
END MIXT
READ GEOM
```

(continues on next page)

(continued from previous page)

```
UNIT 1
  CYLINDER  2 1 9.525 8.89 -8.89
  CYLINDER  3 1 10.16 2P9.525
  CUBOID  0 1 4P10.875 2P10.24
UNIT 2
  CYLINDER  1 1 5.748 2P5.3825
  CUBOID  0 1 4P6.59 2P6.225
UNIT 3
  ARRAY 1 3*0.0
UNIT 4
  ARRAY 2 3*0.0
  REPLICATE 0 1 2*0.0 2*8.57 2*8.03 1
END GEOM
READ ARRAY
  ARA=1 NUX=1 NUY=2 NUZ=2                FILL F1 END FILL
  ARA=2 NUX=1 NUY=2 NUZ=2                FILL F2 END FILL
  GBL=3 ARA=3 NUX=2 NUY=1 NUZ=1   FILL 4 3 END FILL
END ARRAY
  READ PLOT TTL='X-Y SLICE AT Z=10.24'
  XUL=-1.0    YUL=44.5 ZUL=10.24
  XLR=35.93 YLR=-1.0  ZLR=10.24
  UAX=1.0 VDN=-1.0 NAX=640 PIC=MIX END
  TTL='X-Z SLICE AT Y=10.875'
  XUL=-1.0 YUL=10.875 ZUL=41.96 XLR=35.93 YLR=10.875 ZLR=-1.0
  UAX=1.0 WDN=-1.0 PIC=MIX END  END PLOT
END DATA
END
```

KENO-VI:

```
=KENO-VI
SAMPLE PROBLEM 19 4 AQUEOUS 4 METAL ARRAY OF ARRAYS
READ PARAM
  LIB=4 FLX=YES FDN=YES NUB=YES SMU=YES MKP=YES MKU=YES FMP=YES FMU=YES
END PARAM
READ MIXT
  SCT=2
  MIX=1
       1092234 4.82717E-04
       1092235 4.47971E-02
       1092236 9.57233E-05
       1092238 2.65767E-03
  MIX=2
       2001001 5.77931E-02
       2007014 2.13092E-03
       2008016 3.74114E-02
       2092234 1.06784E-05
       2092235 9.84602E-04
       2092236 5.29386E-06
       2092238 6.19414E-05
  MIX=3
       11001001 5.67873E-02
       11006000 3.54921E-02
       11008016 1.41968E-02
END MIXT
READ GEOM
UNIT 1
  CYLINDER 1  9.525  2P8.89
  CYLINDER 2  10.16  2P9.525
  CUBOID   3  4P10.875  2P10.24
  MEDIA 2 1 1
  MEDIA 3 1 2 -1
  MEDIA 0 1 3 -2
  BOUNDARY  3
UNIT 2
```

(continues on next page)

(continued from previous page)

```
  CYLINDER  1 5.748 2P5.3825
  CUBOID    2 4P6.59 2P6.225
  MEDIA 1 1 1
  MEDIA 0 1 2 -1
  BOUNDARY  2
UNIT 3
  CUBOID 1 2P10.875 2P21.75 2P20.48
  ARRAY 1 1 PLACE 1 1 1 0.0 -10.875 -10.24
  BOUNDARY  1
UNIT 4
  CUBOID 1 2P6.59 2P13.18 2P12.45
  CUBOID 2 2P6.59 2P21.75 2P20.48
  ARRAY  2  1 PLACE 1 1 1 0.0 -6.59 -6.225
  MEDIA 0 1 2 -1
  BOUNDARY
GLOBAL UNIT 5
  CUBOID 1 34.93 0.0 43.5 0.0 40.96 0.0
  ARRAY 3 1 PLACE 1 1 1 6.59 21.75 20.48
  BOUNDARY  1
END GEOM
READ ARRAY
  ARA=1 NUX=1 NUY=2 NUZ=2                FILL F1 END FILL
  ARA=2 NUX=1 NUY=2 NUZ=2                FILL F2 END FILL
  GBL=3 ARA=3 NUX=2 NUY=1 NUZ=1    FILL 4 3 END FILL
END ARRAY
READ PLOT TTL='X-Y SLICE AT Z=10.24'
  XUL=-1.0 YUL=44.5 ZUL=10.24
  XLR=35.93 YLR=-1.0 ZLR=10.24
  UAX=1.0 VDN=-1.0 NAX=130 NCH=' *.-' PIC=MIX END
  TTL='X-Z SLICE AT Y=10.875'
  XUL=-1.0 YUL=10.875 ZUL=41.96
  XLR=35.93 YLR=10.875 ZLR=-1.0
  UAX=1.0 WDN=-1.0 PIC=MIX END
END PLOT
END DATA
END
```

A plot of an X-Y slice taken through the bottom layer of the array is shown in Fig. 8.1.30. A plot of an X-Z slice taken through the +Y half of the array is shown in Fig. 8.1.31. These plots were used to verify the geometry mockup.

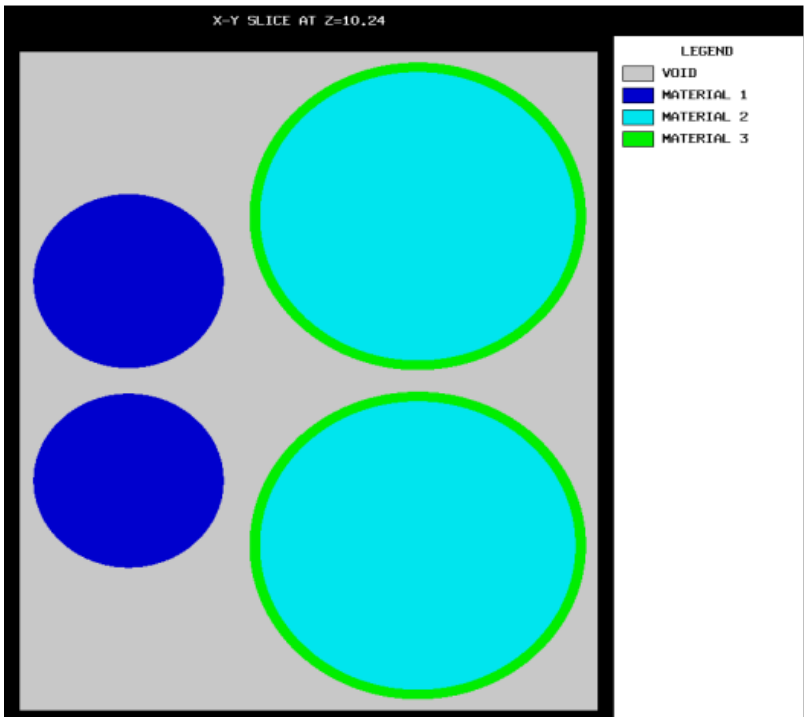


Fig. 8.1.30: X-Y plot of mixed array.

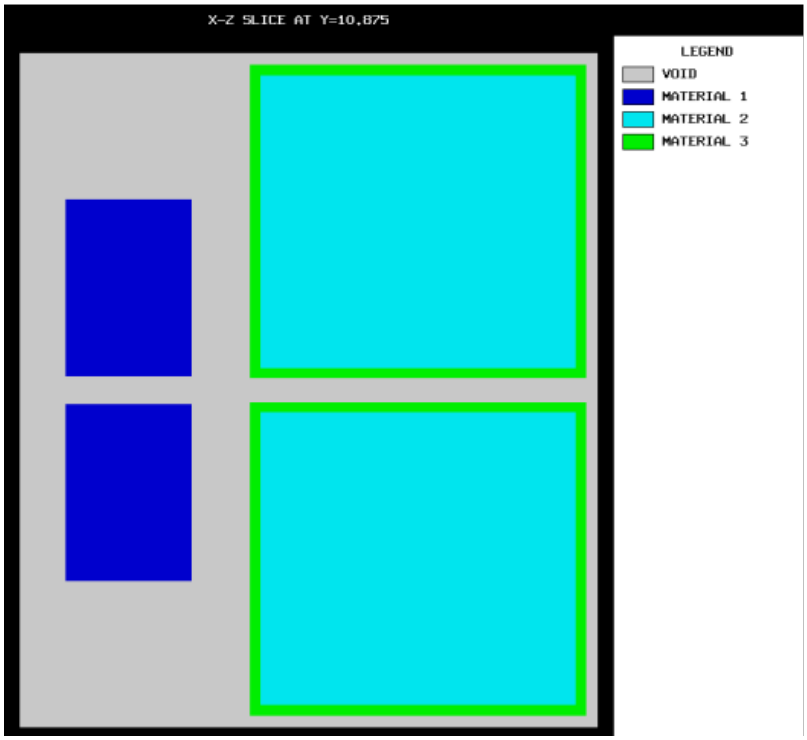


Fig. 8.1.31: X-Z plot of mixed array.

STORAGE ARRAY

EXAMPLE 18. Consider a storage array of highly enriched uranium buttons, each 1 in. tall and 4 in. in diameter. These buttons are stored on stainless steel shelves with a center-to-center spacing of 60.96 cm (2 ft) between them in the Y direction, and only one button on each shelf in the X direction. The shelves are 0.635 cm ($^1/_4$in.) thick (Z dimension), 45.72 cm (18 in.) wide (X dimension), 609.6 cm (20 ft) long (Y dimension), and are 45.72 cm (18 in.) from the top of a shelf to the bottom of the shelf above it. Each rack of storage shelves is four shelves high, with the first shelf being 15.24 cm (6 in.) above the floor. The storage room is 586.56 cm (19.5 ft) in the X direction by 1293.44 cm (43 ft) in the Y direction with 365.76 cm (12 ft.) ceilings in the Z direction. The walls, ceiling, and floor are composed of concrete, 30.48 cm (1 ft) thick. All the aisles between the storage racks are 91.44 cm (3 ft) wide. The racks are arranged with their length in the Y direction and an aisle between them. The arrays of racks are arranged with two in the Y direction and five in the X direction. Mixture 1 is the uranium metal, mixture 2 is the stainless steel, and mixture 3 is the concrete.

The metal button and its center-to-center spacing are described first. The void vertical spacing has arbitrarily been chosen to extend from the bottom of the button to the next shelf above the button. The shelf of stainless steel is described under the button.

KENO V.a:

```
UNIT 1
COM='METAL BUTTONS'
CYLINDER 1 1 5.08 2.54 0.0
CUBOID  0 1 2P22.86 2P30.48 45.72 0.0
CUBOID  2 1 2P22.86 2P30.48 45.72 -0.635
```

KENO-VI:

ARRAY 1 creates an ARRAY of these buttons that fills one shelf. UNIT 2 then contains one of the shelves shown in Fig. 8.1.32.

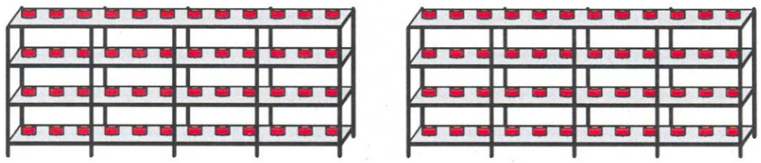


Fig. 8.1.32: Two racks of uranium buttons.

KENO V.a:

```
ARA=1 COM='SINGLE SHELF CONTAINING 10 METAL BUTTONS'
      NUX=1 NUY=10 NUZ=1 FILL F1 END FILL

UNIT 2
COM='SINGLE SHELF (1 X 10 X 1 ARRAY OF METAL BUTTONS ON A SHELF)'
ARRAY 1 3*0.0
```

KENO-VI:

```
ARA=1  NUX=1  NUY=10  NUZ=1  FILL  F1  END FILL

UNIT 2
CUBOID 1 45.72 0.0 609.60 0.0 46.355 0.0
ARRAY  1  1  PLACE  1  1  1  22.86  30.48  0.635
BOUNDARY  1
```

Note: The origin of UNIT 2 is on the bottom of the bottom shelf; it has been moved from the bottom of the button. The X and Y position of the origin is at the front, left-hand corner of the bottom of this lowest shelf.

Stack four UNIT 2s vertically to obtain one of the racks shown in Fig. 8.1.32. One rack is defined by array 2.

```
ARA=2 COM='SINGLE RACK OF 4 SHELVES'

NUX=1 NUY=1 NUZ=4 FILL F2 END FILL
```

Generate a UNIT 3 that contains a rack of shelves and a UNIT 4 that is the aisle between the ends of the two racks in the Y direction.

KENO V.a:

```
UNIT 3
COM='SINGLE RACK (4 SHELVES TALL)'
ARRAY 2 3*0.0

UNIT 4
COM='CENTRAL AISLE UNIT SAME HEIGHT AS 4 SHELVES'
CUBOID 0 1 2P22.86 2P45.72 185.42 0.0
```

KENO-VI:

```
UNIT 3
CUBOID 1 45.72 0.0 609.60 0.0 185.42 0.0
ARRAY  2 1 PLACE 1 1 1 3*0.0
BOUNDARY  1
UNIT 4
CUBOID  1  2P22.86  2P45.72  185.42  0.0
MEDIA 0 1 1
BOUNDARY  1
```

Stack UNITs 3 and 4 together in the Y direction to create UNIT 5 which contains both racks in the Y direction and the aisle between them. This configuration is shown in Fig. 8.1.32.

KENO V.a:

```
ARA=3 COM='TWO RACKS END TO END WITH CENTRAL AISLE'
      NUX=1 NUY=3 NUZ=1 FILL 3 4 3 END FILL

UNIT 5
COM='SET OF TWO RACKS END TO END SEPARATED BY THE CENTRAL AISLE'
ARRAY 3 3*0.0
```

KENO-VI:

```
ARA=3  NUX=1  NUY=3  NUZ=1  FILL  3 4 3  END FILL

UNIT 5
CUBOID 1 45.72 0.0 1310.64 0.0 185.42 0.0
ARRAY 3 1 PLACE 1 1 1 3*0.0
BOUNDARY  1
```

Create a UNIT 6, which is an aisle 91.44 cm (3 ft) wide in the X direction and 1310.64 cm (43 ft) in the Y direction (full length of the room).

KENO V.a:

```
UNIT 6
COM='AISLE BETWEEN ADJACENT SETS OF TWO RACKS & CENTRAL AISLE (UNITS 5)'
CUBOID 0 1 91.44 0.0 1310.64 0.0 185.42 0.0
```

KENO-VI:

```
UNIT 6
CUBOID 1 91.44 0.0  1310.64 0.0  185.42 0.0
MEDIA 0 1 1
BOUNDARY  1
```

Stack UNITs 5 and 6 in the X direction to achieve the array of racks in the room. Then put the 6 in. spacing below the bottom of the racks, the spacing between the top of the top rack and the ceiling, and add the concrete floor, walls, and ceiling around the array. ARRAY 4 describes the array of racks in the room. ARRAY record (first CUBOID description in KENO-VI) encompasses this ARRAY, and the first REFLECTOR (second CUBOID in KENO-VI) descriptions are used to add the spacing between the top rack and the ceiling. The last two REFLECTOR (CUBOIDs 3 through 9 in KENO-VI) descriptions add the ceiling, walls, and floor in 5.0 cm increments to bias the concrete. A perspective of the room is shown in Fig. 8.1.33.

KENO V.a:

```
ARA=4 COM='ENTIRE STORAGE ARRAY'
      NUX=9 NUY=1 NUZ=1 FILL 5 6 3Q2 5 END FILL
GLOBAL
UNIT 7
COM='STORAGE ARRAY IN THE ROOM WITH WALLS, FLOOR AND CEILING'
ARRAY  4  3*0.0
REFLECTOR   0  1  4*0.0  165.1  15.24  1
REFLECTOR  3  2  6*5.0  6
REFLECTOR  3  8  6*0.48  1
```

KENO-VI:

```
GBL=4  ARA=4  NUX=9  NUY=1  NUZ=1  FILL 5 6  3Q2  5  END FILL

GLOBAL UNIT  7
CUBOID 1 594.36   0.0  1310.64   0.0  185.42   0.0
CUBOID 2 594.36   0.0  1310.64   0.0  350.52 -15.24
CUBOID 3 599.36  -5.00 1315.64  -5.00 355.52 -20.24
CUBOID 4 604.36 -10.00 1320.64 -10.00 360.52 -25.24
CUBOID 5 609.36 -15.00 1325.64 -15.00 365.52 -30.24
CUBOID 6 614.36 -20.00 1330.64 -20.00 370.52 -35.24
CUBOID 7 619.36 -25.00 1335.64 -25.00 375.52 -40.24
CUBOID 8 624.36 -30.00 1340.64 -30.00 380.52 -45.24
CUBOID 9 624.84 -30.48 1341.12 -30.48 381.00 -45.72
ARRAY  4 1 PLACE 1 1 1 3*0.0
MEDIA 0 1 2 -1
MEDIA 3 2 3 -2
MEDIA 3 3 4 -3
MEDIA 3 4 5 -4
MEDIA 3 5 6 -5
MEDIA 3 6 7 -6
MEDIA 3 7 8 -7
MEDIA 3 8 9 -8
BOUNDARY  9
```

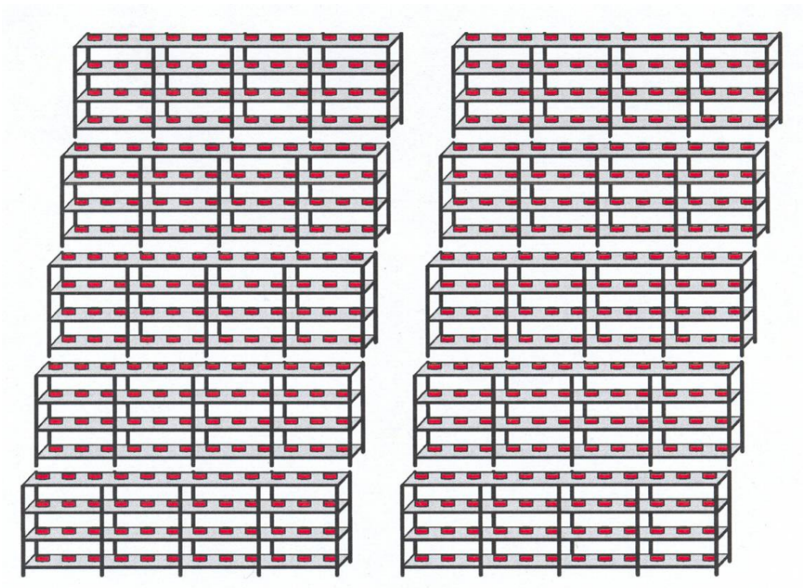


Fig. 8.1.33: Entire storage array in the room.

The complete input for this room is given below: The plots for this problem must be quite large in order to see all the detail because the array is sparse and the shelves are thin. Therefore, the plots for this system are not included as Fig.s. The user can generate the plots if it is desirable to see them. The nuclide IDs used in this problem are for the 16-group Hansen-Roach working format library, which is no longer distributed with SCALE.

KENO V.a:

```
=KENO5A
STORAGE ARRAY
READ PARAMETERS  FDN=YES LIB=41
END PARAMETERS
READ MIXT  SCT=1  MIX=1 92500 4.48006e-2  92800 2.6578e-3  92400 4.827e-4
92600 9.57e-5  MIX=2 200 1.0 MIX=3 301 1 END MIXT
READ GEOMETRY
UNIT 1
COM='METAL BUTTONS'
CYLINDER 1 1 5.08 2.54 0.0
CUBOID  0 1 2P22.86 2P30.48 45.72 0.0
CUBOID  2 1 2P22.86 2P30.48 45.72 -0.635
UNIT 2
COM='SINGLE SHELF (1 X 10 X 1 ARRAY OF METAL BUTTONS ON A SHELF)'
ARRAY 1 3*0.0
UNIT 3
COM='SINGLE RACK (4 SHELVES TALL)'
ARRAY 2 3*0.0
UNIT 4
COM='CENTRAL AISLE UNIT SAME HEIGHT AS 4 SHELVES'
CUBOID 0 1 2P22.86 2P45.72 185.42 0.0
UNIT 5
COM='SET OF TWO RACKS END TO END SEPARATED BY THE CENTRAL AISLE'
ARRAY 3 3*0.0
UNIT 6
COM='AISLE BETWEEN ADJACENT SETS OF TWO RACKS & CENTRAL AISLE (UNITS 5)'
CUBOID 0 1 91.44 0.0 1310.64 0.0 185.42 0.0
GLOBAL
UNIT 7
```

(continues on next page)

(continued from previous page)

```
COM='STORAGE ARRAY IN THE ROOM WITH WALLS, FLOOR AND CEILING'
ARRAY 4 3*0.0
REFLECTOR 0 1 4*0.0 165.1 15.24 1
REFLECTOR 3 2 6*5.0    6
REFLECTOR 3 8 6*0.48  1
END GEOMETRY
READ ARRAY
ARA=1 COM='SINGLE SHELF CONTAINING 10 METAL BUTTONS'
      NUX=1 NUY=10 NUZ=1 FILL F1 END FILL
ARA=2 COM='SINGLE RACK OF 4 SHELVES'
      NUX=1 NUY=1 NUZ=4 FILL F2 END FILL
ARA=3 COM='TWO RACKS END TO END WITH CENTRAL AISLE'
      NUX=1 NUY=3 NUZ=1 FILL 3 4 3 END FILL
ARA=4 COM='ENTIRE STORAGE ARRAY'
      NUX=9 NUY=1 NUZ=1 FILL 5 6 3Q2 5 END FILL
END ARRAY
READ BIAS ID=301 2 8 END BIAS
READ START NST=5  NBX=5 END START
READ PLOT TTL='X-Z SLICE AT Y=30.48 WITH Z ACROSS AND X DOWN'
XUL=594.8 YUL=30.48 ZUL=-1.0 XLR=-0.5 YLR=30.48 ZLR=186.0
WAX=1.0 UDN=-1.0 NAX=640  END
TTL='Y-Z SLICE OF LEFT RACKS, X=22.86 WITH Z ACROSS AND Y DOWN'
XUL=22.86 YUL=1311.0 ZUL=-0.5 XLR=22.86 YLR=-3.0 ZLR=186.0
WAX=1.0 VDN=-1.0 NAX=640 END
TTL='X-Y SLICE OF ROOM THROUGH SHELF Z=0.3175 WITH X ACROSS AND Y DOWN'
XUL=-1.0 YUL=1312.0 ZUL=0.3175 XLR=596.0 YLR=-2.5 ZLR=0.3175
UAX=1.0 VDN=-1.0 NAX=320 END
END PLOT
END DATA
END
```

KENO-VI:

```
=KENOVI
STORAGE ARRAY
READ PARAMETERS  FDN=YES LIB=41  END PARAMETERS
READ MIXT  SCT=1
MIX=1 92500 4.48006e-2  92800 2.6578e-3  92400 4.827e-4  92600 9.57e-5
MIX=2 200 1.0 MIX=3 301 1
END MIXT
READ GEOMETRY
UNIT 1
CYLINDER 1  5.08  2.54  0.0
CUBOID   2   2P22.86   2P30.48  45.72  0.0
CUBOID   3   2P22.86   2P30.48  45.72  -.635
MEDIA 1 1 1
MEDIA 0 1 2 -1
MEDIA 2 1 3 -2
BOUNDARY  3
UNIT 2
CUBOID 1 45.72 0.0 609.60 0.0 46.355 0.0
ARRAY  1  1  PLACE  1  1  1  22.86  30.48  0.635
BOUNDARY  1
UNIT 3
CUBOID 1 45.72 0.0 609.60 0.0 185.42 0.0
ARRAY 2 1 PLACE 1 1 1 3*0.0
BOUNDARY  1
UNIT 4
CUBOID  1  2P22.86  2P45.72  185.42  0.0
MEDIA 0 1 1
BOUNDARY  1
UNIT 5
CUBOID 1 45.72 0.0 1310.64 0.0 185.42 0.0
ARRAY 3 1 PLACE 1 1 1 3*0.0
BOUNDARY  1
```

(continues on next page)

(continued from previous page)

```
UNIT 6
CUBOID 1 91.44 0.0  1310.64 0.0  185.42 0.0
MEDIA 0 1 1
BOUNDARY  1
GLOBAL UNIT  7
CUBOID 1 594.36    0.0 1310.64    0.0 185.42    0.0
CUBOID 2 594.36    0.0 1310.64    0.0 350.52 -15.24
CUBOID 3 599.36  -5.00 1315.64  -5.00 355.52 -20.24
CUBOID 4 604.36 -10.00 1320.64 -10.00 360.52 -25.24
CUBOID 5 609.36 -15.00 1325.64 -15.00 365.52 -30.24
CUBOID 6 614.36 -20.00 1330.64 -20.00 370.52 -35.24
CUBOID 7 619.36 -25.00 1335.64 -25.00 375.52 -40.24
CUBOID 8 624.36 -30.00 1340.64 -30.00 380.52 -45.24
CUBOID 9 624.84 -30.48 1341.12 -30.48 381.00 -45.72
ARRAY  4 1 PLACE 1 1 1 3*0.0
MEDIA 0 1 2 -1
MEDIA 3 2 3 -2
MEDIA 3 3 4 -3
MEDIA 3 4 5 -4
MEDIA 3 5 6 -5
MEDIA 3 6 7 -6
MEDIA 3 7 8 -7
MEDIA 3 8 9 -8
BOUNDARY  9
END GEOMETRY
READ ARRAY
ARA=1 NUX=1 NUY=10 NUZ=1 FILL F1 END FILL
ARA=2 NUX=1 NUY=1 NUZ=4 FILL F2 END FILL
ARA=3 NUX=1 NUY=3 NUZ=1 FILL 3 4 3 END FILL
GBL=4 ARA=4 NUX=9 NUY=1 NUZ=1 FILL 5 6 3Q2 5 END FILL
END ARRAY
READ BIAS ID=301 2 8 END BIAS
READ START NST=5 NBX=5 END START
READ PLOT PLT=YES TTL='X-Z SLICE AT Y=30.48 WITH Z ACROSS AND X DOWN'
XUL=594.8 YUL=30.48 ZUL=-1.0 XLR=-0.5 YLR=30.48 ZLR=186.0
WAX=1.0 UDN=-1.0 NAX=640  END
TTL='Y-Z SLICE OF LEFT RACKS, X=22.86 WITH Z ACROSS AND Y DOWN'
XUL=22.86 YUL=1311.0 ZUL=-0.5 XLR=22.86 YLR=-3.0 ZLR=186.0
WAX=1.0 VDN=-1.0 NAX=640 END
TTL='X-Y SLICE OF ROOM THROUGH SHELF Z=0.3175 WITH X ACROSS AND Y DOWN'
XUL=-1.0 YUL=1312.0 ZUL=0.3175 XLR=596.0 YLR=-2.5 ZLR=0.3175
UAX=1.0 VDN=-1.0 NAX=320 END
END PLOT
END DATA
END
```

Arrays and holes

Sect. 8.1.4.6.1 and Sect. 8.1.4.6.2 describe the use of HOLEs, and Sect. 8.1.4.6.3 describes multiple ARRAYs and ARRAYs of ARRAYs. HOLEs can be used to place ARRAYs at locations in other UNITs. This section contains examples to illustrate the combined use of ARRAYs and HOLEs.

EXAMPLE 19. A SIMPLE CASK

This example consists of cylindrical mild steel container with an inside radius of 4.15 cm and a radial wall thickness of 0.45 cm. The thickness of the ends of the container is 1.27 cm, and the inside height is 10.1 cm. Highly enriched uranium rods 1 cm in diameter and 10 cm long are banded together into square bundles of four. These bundles are then positioned in the mild steel container as shown in Fig. 8.1.34. The rods sit on the floor of the container and have a 0.1 cm gap between their tops and the top of the container.

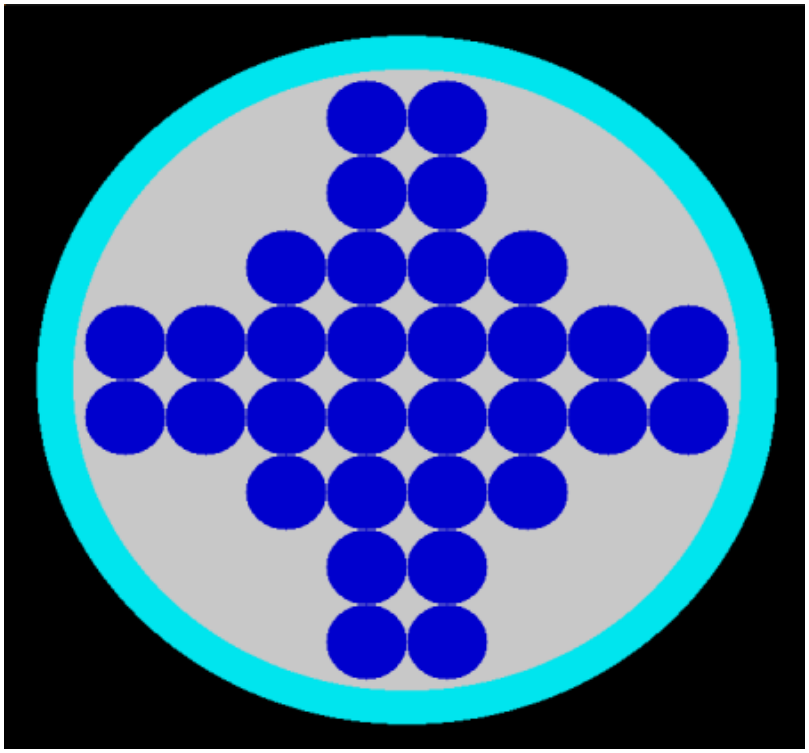


Fig. 8.1.34: Uranium rods in a cylindrical container.

To generate the geometry description for this system, UNIT 1 is defined as one uranium rod and its associated square-pitch close-packed spacing region.

KENO V.a:

```
UNIT 1
CYLINDER  1 1  0.5  2P5.0
CUBOID    0 1  4P0.5  2P5.0
```

KENO-VI:

```
UNIT 1
CYLINDER 1 0.5  2P5.0
CUBOID   2   4P0.5  2P5.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
```

From here, the geometry description diverges between KENO V.a and KENO-VI.

KENO V.a:

ARRAY 1 is defined to be the central square ARRAY consisting of four bundles of rods.

```
ARA=1 NUX=4 NUY=4 NUZ=1 FILL F1 END FILL
```

ARRAY 2 is defined to be a bundle of four rods.

```
ARA=2 NUX=2 NUY=2 NUZ=1 FILL F1 END FILL
```

ARRAY 2 is placed in UNIT 2. This defines the outer boundaries of an imaginary CUBOID that contains the ARRAY. It is convenient to have the origin of the ARRAY at its center, so the most negative point of the array will be (-1, -1,-5).

```
UNIT 2
ARRAY 2  -1.0 -1.0 -5.0
```

An ARRAY record is used to place ARRAY 1 in the GLOBAL UNIT. Then the cylindrical container is described around it and HOLEs are used to place the four outer bundles around the central ARRAY.

```
GLOBAL UNIT 3
ARRAY   1  -2.0  -2.0  -5.0
CYLINDER   0 1  4.15  5.1  -5.0
HOLE   2    0.0 -3.0  0.0
HOLE   2    3.0  0.0  0.0
HOLE   2    0.0  3.0  0.0
HOLE   2   -3.0  0.0  0.0
CYLINDER   2  1  4.6  6.37 -6.27
```

The first HOLE places the bottom bundle of four rods, the second HOLE places the bundle of four rods at the right, the third HOLE places the top bundle of rods and the fourth HOLE places the left bundle of rods.

KENO-VI:

Define UNIT 2 to be a void CUBOID with the same square pitch as the rod square pitch.

> UNIT 2
>
> CUBOID 1 4P0.5 2P5.0
>
> MEDIA 0 1 1
>
> BOUNDARY 1

ARRAY 1 is defined to be the central square 10×10 ARRAY consisting of 32 rods and 68 void positions that can be used to represent the array shown in Fig. 8.1.34.

```
ARA=1 NUX=10 NUY=10 NUZ=1
FILL 14*2 1 1 8*2 1 1 7*2 4*1 4*2 8*1 2 2 8*1 4*2 4*1 7*2 1 1 8*2 1 1 14*2
END FILL
```

ARRAY 1 is placed in UNIT 3. The first CYLINDER card defines the ARRAY BOUNDARY. Everything external to this boundary is not considered part of the problem. The positions in the ARRAY that do not contain rods are filled with cuboids consisting of void. The ARRAY boundary must either coincide with the outer boundary of the ARRAY or be contained within the ARRAY. An exterior void region is placed around the array boundary to coincide with the size of the interior radius of the container. The 10×10 ARRAY with the ARRAY boundary is shown in Fig. 8.1.35.

```
UNIT 3
CYLINDER 1  4.15 5.0 -5.0
CYLINDER 2 4.15 5.1 -5.0
ARRAY 1 1 PLACE 5 5 1   -0.5 -0.5 -0.0
MEDIA 0 1 2 -1
BOUNDARY  2
```

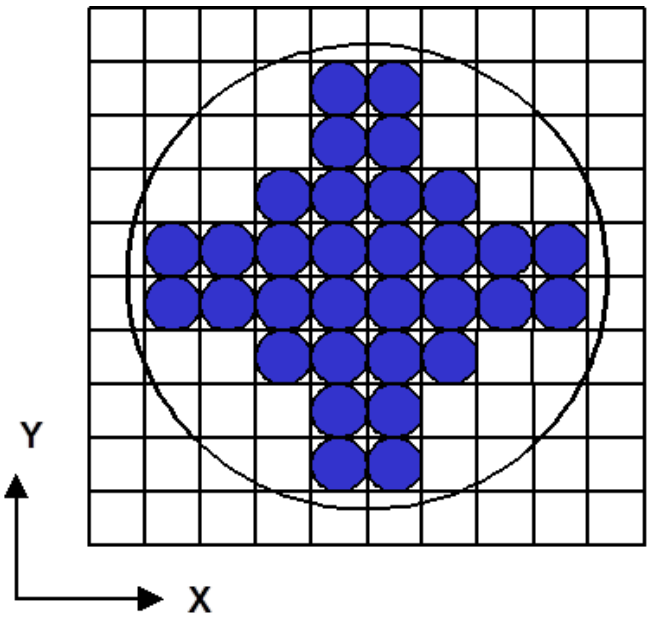


Fig. 8.1.35: The 10 × 10 array of 32 uranium rods and 68 void cuboids with the array boundary.

The UNIT containing the ARRAY is now placed within the global unit using a HOLE content record. The location of the HOLE is determined using ORIGIN data to match the origin of the UNIT in the HOLE with an X, Y, Z position in the surrounding UNIT. In this problem, the origin of the UNIT must be at position (0,0,0). Since only nonzero data must be entered, ORIGIN data are not needed for this problem. The boundary region consists of the steel container.

```
GLOBAL UNIT 4
CYLINDER 2 4.6 6.37 -6.27
HOLE 3 ORIGIN X=0.0 Y=0.0 Z=0.0
MEDIA 2 1 2
BOUNDARY  2
```

The overall problem description is shown below. Two of the color plots used for verification of this mockup are shown in Fig. 8.1.36 and Fig. 8.1.37. The black outside border of the two color plots indicates volume outside the global unit. The plot can be extended just outside the global unit boundary to ensure that the entire problem is included in the plot. This results in a black area surrounding the actual problem.

KENO V.a:

```
=KENOVA
CASK ARRAY
READ PARAMETERS  FDN=YES LIB=41 GEN=10
END PARAMETERS
READ MIXT  SCT=1  MIX=1 92500 4.48006e-2  92800 2.6578e-3  92400 4.827e-4
92600 9.57e-5  MIX=2 100 1.0 END MIXT
READ GEOMETRY
UNIT 1
CYLINDER 1 1 0.5 2P5.0
CUBOID  0 1 4P0.5 2P5.0
UNIT 2
ARRAY 2 -1.0 -1.0 -5.0
GLOBAL UNIT 3
ARRAY  1 -2.0 -2.0 -5.0
```

(continues on next page)

(continued from previous page)

```
CYLINDER 0 1 4.15 5.1 -5.0
HOLE 2 0.0 -3.0 0.0
HOLE 2 3.0 0.0 0.0
HOLE 2 0.0 3.0 0.0
HOLE 2 -3.0 0.0 0.0
CYLINDER 2 1 4.6 6.37 -6.27
END GEOM
READ ARRAY
ARA=1 NUX=4 NUY=4 NUZ=1 FILL F1 END FILL
ARA=2 NUX=2 NUY=2 NUZ=1 FILL F1 END FILL
END ARRAY
READ PLOT TTL='X-Z SLICE AT Y=0.25 WITH X ACROSS AND Z DOWN'
XUL=-5.0 YUL=0.25 ZUL=6.5 XLR=5.0 YLR=0.25 ZLR=-6.5
UAX=1.0 WDN=-1.0 NAX=640  END
TTL='X-Y SLICE AT Z=0.0 WITH X ACROSS AND Y DOWN'
XUL=-5.0 YUL=5.0 ZUL=0.0 XLR=5.0 YLR=-5.0 ZLR=0.0
UAX=1.0 VDN=-1.0 NAX=640 END
END PLOT
END DATA
END
```

KENO-VI:

```
KENO VI
CASK ARRAY
READ PARAMETERS  TME=1.0 FDN=YES LIB=41 GEN=10  END PARAMETERS
READ MIXT  SCT=1
MIX=1 92500 4.48006e-2  92800 2.6578e-3  92400 4.827e-4  92600 9.57e-5
MIX=2 100 1.0 END MIXT
READ GEOMETRY
UNIT 1
CYLINDER 1  0.5  2P5.0
CUBOID   2   4P0.5  2P5.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 2
CUBOID 1 4P0.5 2P5.0
MEDIA 0 1 1
BOUNDARY  1
UNIT 3
CYLINDER 1  4.15 5.0 -5.0
CYLINDER 2 4.15 5.1 -5.0
ARRAY 1 1 PLACE 5 5 1   -0.5 -0.5 -0.0
MEDIA 0 1 2 -1
BOUNDARY  2
GLOBAL UNIT 4
CYLINDER 2 4.6 6.37 -6.27
HOLE 3 ORIGIN X=0.0 Y=0.0 Z=0.0
MEDIA 2 1 2
BOUNDARY  2
END GEOM
READ ARRAY
ARA=1  NUX=10  NUY=10  NUZ=1  FILL  14*2 1 1 8*2 1 1 7*2 4*1 4*2 8*1 2 2 8*1 4*2 4*1 7*2 1 1
8*2 1 1 14*2  END FILL
END ARRAY
READ PLOT TTL='X-Z SLICE AT Y=0.25 WITH X ACROSS AND Z DOWN'
XUL=-5.0 YUL=0.25 ZUL=6.5 XLR=5.0 YLR=0.25 ZLR=-6.5
UAX=1.0 WDN=-1.0 NAX=640  END
TTL='X-Y SLICE AT Z=0.0 WITH X ACROSS AND Y DOWN'
XUL=-5.0 YUL=5.0 ZUL=0.0 XLR=5.0 YLR=-5.0 ZLR=0.0
UAX=1.0 VDN=-1.0 NAX=640 END
END PLOT
END DATA
END
```

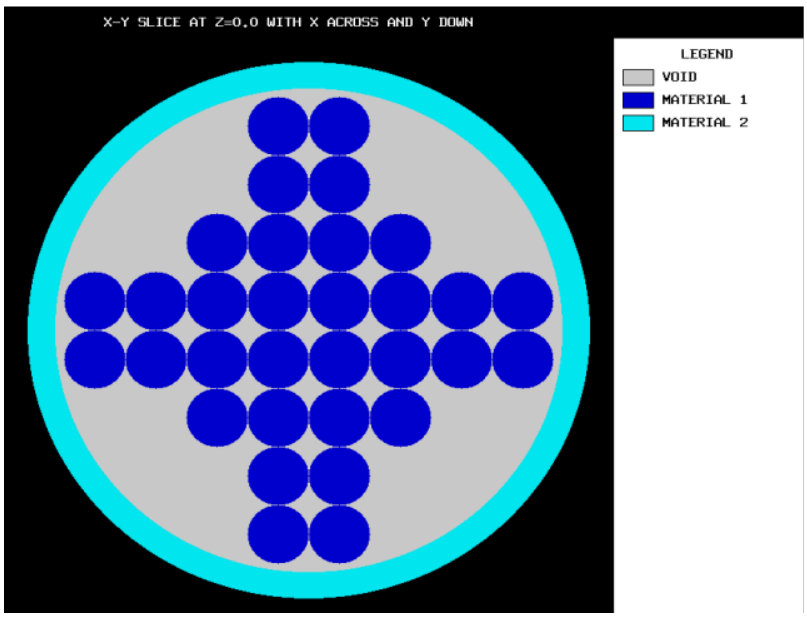


Fig. 8.1.36: X-Y slice of uranium rods in a cylindrical container.

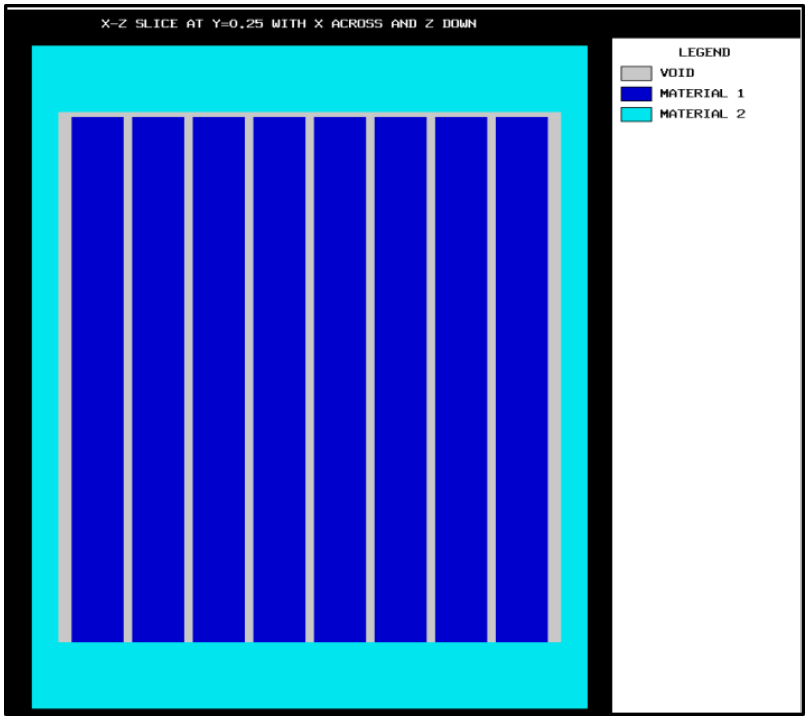


Fig. 8.1.37: X-Z slice of uranium rods in a cylindrical container.

EXAMPLE 20. A TYPICAL PRESSURIZED WATER REACTOR (PWR) SHIPPING CASK

A typical PWR shipping cask is illustrated in Fig. 8.1.38. The interior and exterior shell of the cask is carbon steel (mixture 7), and a depleted uranium gamma shield (mixture 6) is present in the annulus between the steel layers. The shipping cask contains seven PWR fuel assemblies. Each assembly is a 17×17 `ARRAY` of

fuel rods with water holes. Each assembly is contained in a stainless steel (mixture 5) box. Each fuel rod is composed of 4% enriched UO_2 (mixture 1) clad with Zircaloy (mixture 2). Rods of B_4C clad (mixture 4) with stainless steel are positioned between the fuel assemblies. The entire cask is filled with water (mixture 3).

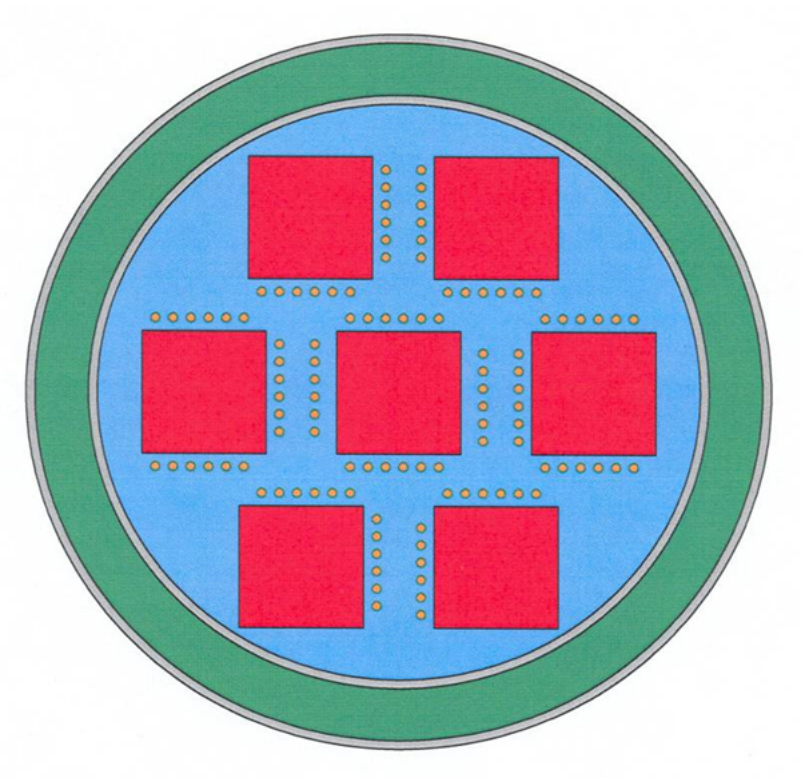


Fig. 8.1.38: Typical PWR shipping cask.

To describe the geometry of the cask, some simple units are defined as shown in Fig. 8.1.39. `UNIT` 1 represents a fuel rod and its associated square pitch spacing region. `UNIT` 2 represents a water hole in a fuel assembly. `UNIT`s 3, 4, and 6 represent the B_4C rods with their various spacings, and `UNIT` 5 is a water hole that is used in association with some of the B_4C rods.

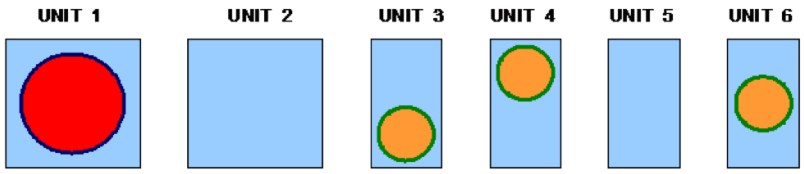


Fig. 8.1.39: Simple units.

KENO V.a:

```
UNIT 1
CYLINDER  1 1 .41148 365.76 0.0
CYLINDER  2 1 .48133 365.76 0.0
CUBOID    3 1 .63754 -.63754 .63754 -.63754 365.76 0.0
```

(continues on next page)

(continued from previous page)

```
UNIT 2
CUBOID    3 1 .63754 -.63754 .63754 -.63754 365.76 0.0


UNIT 3
CYLINDER  4 1 .584 365.76 0.0
CYLINDER  5 1 .635 365.76 0.0
CUBOID    3 1 .9912 -.9912 2.2352 -1.27 365.76 0.0

UNIT 4
CYLINDER  4 1 .584 365.76 0.0
CYLINDER  5 1 .635 365.76 0.0
CUBOID    3 1 .9912 -.9912 1.2702 -2.235 365.76 0.0

UNIT 5
CUBOID    3 1 .9912 -.9912 1.7526 -1.7526 365.76 0.0

UNIT 6
CYLINDER  4 1 .584 365.76 0.0
CYLINDER  5 1 .635 365.76 0.0
CUBOID    3 1 1.1875215 -1.1875215 1.883706 -1.883706 365.76 0.0
```

KENO-VI:

```
UNIT 1
CYLINDER 1  .41148 365.76 0.0
CYLINDER 2  .48133 365.76 0.0
CUBOID   3  .63754 -.63754 .63754 -.63754 365.76 0.0
MEDIA 1 1 1
MEDIA 2 1 2 -1
MEDIA 3 1 3 -2
BOUNDARY  3

UNIT 2
CUBOID   1  .63754 -.63754 .63754 -.63754 365.76 0.0
MEDIA 3 1 1
BOUNDARY  1

UNIT 3
CYLINDER  1  .584 365.76 0.0
CYLINDER 2   .635 365.76 0.0
CUBOID    3   .9912 -.9912 2.2352 -1.27 365.76 0.0
MEDIA 4 1 1
MEDIA 5 1 2 -1
MEDIA 3 1 3 -2
BOUNDARY  3

UNIT 4
CYLINDER 1  .584 365.76 0.0
CYLINDER 2  .635 365.76 0.0
CUBOID   3   .9912 -.9912 1.2702 -2.235 365.76 0.0
MEDIA 4 1 1
MEDIA 5 1 2 -1
MEDIA 3 1 3 -2
BOUNDARY  3

UNIT 5
CUBOID 1   .9912 -.9912 1.7526 -1.7526 365.76 0.0
MEDIA 3 1 1
BOUNDARY  1

UNIT 6
CYLINDER 1  .584 365.76 0.0
CYLINDER 2  .635 365.76 0.0
CUBOID  3    1.1875215 -1.1875215 1.883706 -1.883706 365.76 0.0
```

(continues on next page)

(continued from previous page)

```
MEDIA 4 1 1
MEDIA 5 1 2 -1
MEDIA 3 1 3 -2
BOUNDARY  3
```

UNITs 1 and 2 are stacked together into ARRAY 1 to form the ARRAY of fuel pins and water holes in a fuel assembly as shown in Fig. 8.1.40. This ARRAY is then encompassed with a layer of water and a layer of stainless steel to complete a fuel assembly in a storage cell (UNIT 7) as shown in Fig. 8.1.41.

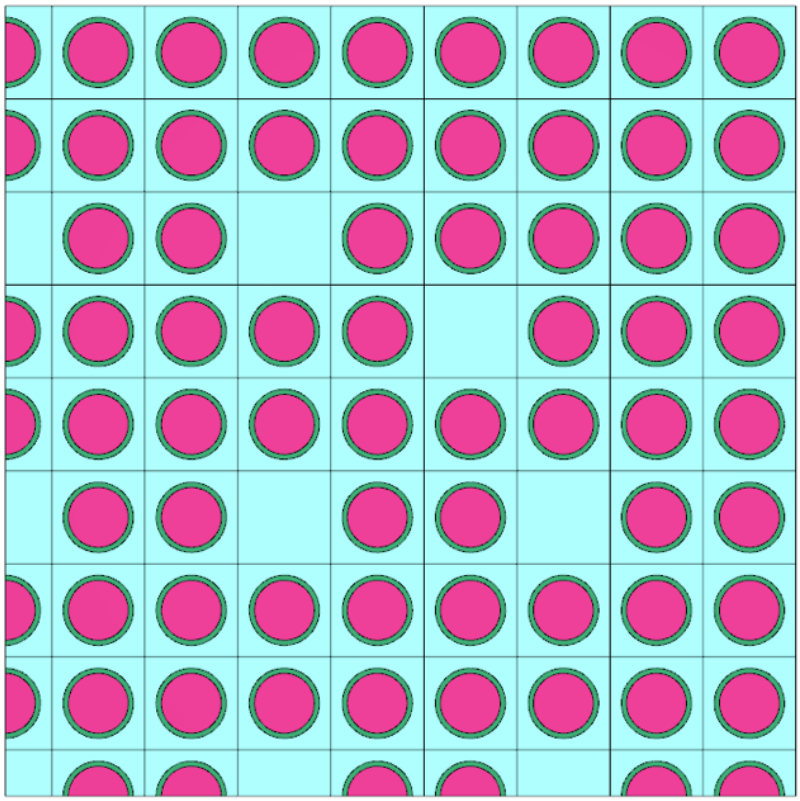


Fig. 8.1.40: Quarter section of fuel pin array.

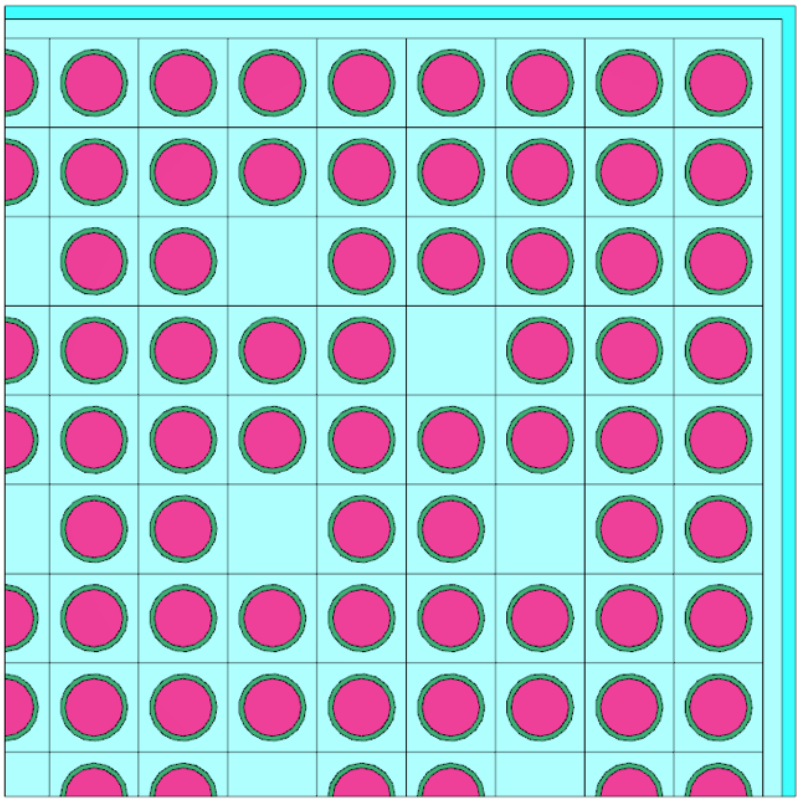


Fig. 8.1.41: Quarter section of fuel assembly.

KENO V.a:

```
ARA=1 NUX=17 NUY=17 NUZ=1 FILL
39R1 2 2Q3 8R1 2 9R1 2 22R1 2 4Q3 38R1 2 4Q3
Q51 22R1 2 Q10 Q9 2Q3 39R1                END FILL

UNIT 7
ARRAY 1 -10.83818 -10.83818 0.0
CUBOID    3 1 11.112495 -11.112495 11.112495 -11.112495 365.76 0.0
CUBOID    8 1 11.302238 -11.302238 11.302238 -11.302238 365.76 0.0
```

KENO-VI:

```
ARA=1 NUX=17 NUY=17 NUZ=1 FILL
39R1 2 2Q3 8R1 2 9R1 2 22R1 2 4Q3 38R1 2 4Q3
Q51 22R1 2 Q10 Q9 2Q3 39R1                END FILL
UNIT 7
CUBOID  1 10.83818 -10.83818 10.83818 -10.83818 365.76 0.0
CUBOID  2 11.112495 -11.112495 11.112495 -11.112495 365.76 0.0
CUBOID  3 11.302238 -11.302238 11.302238 -11.302238 365.76 0.0
ARRAY 1 1  PLACE 9 9 1  3*0.0
MEDIA 3 1 2 -1
MEDIA 5 1 3 -2
BOUNDARY  3
```

An array of `UNIT` 6s is created to represent the array of B_4C rods that is positioned between the fuel assemblies. In KENO V.a, the array of B_4C rods shown in Fig. 8.1.42 is contained in `UNIT` 8 for further use. KENO-VI geometry description does not need the placement of `ARRAY` 2 in a separate `UNIT`.

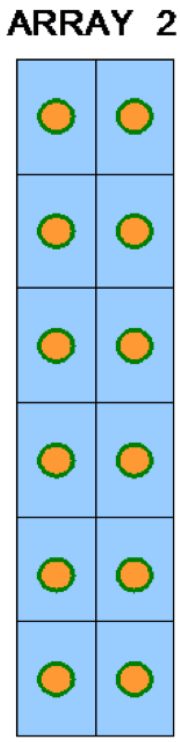


Fig. 8.1.42: 2 × 6 array of B_4C rods.

KENO V.a:

```
ARA=2 NUX=2 NUY=6 NUZ=1    FILL F6        END FILL

UNIT 8
ARRAY 2 0 0 0
```

KENO-VI:

```
ARA=2 NUX=2 NUY=6 NUZ=1    FILL F6        END FILL
```

The next step is to create the central array of three fuel assemblies with B_4C rods between them. In KENO V.a, this is done by stacking fuel assemblies in storage cells (`UNIT` 7) and B_4C rod arrays (`UNIT` 8) into an array (`ARRAY` 3) and placing it in a `UNIT` (`UNIT` 9). In KENO-VI description, `UNIT` 7 (fuel assembly in storage cell) and the array of B_4C rods (`ARRAY` 2) are directly placed into a `UNIT` (`UNIT` 8). The resultant geometry is shown in Fig. 8.1.43.

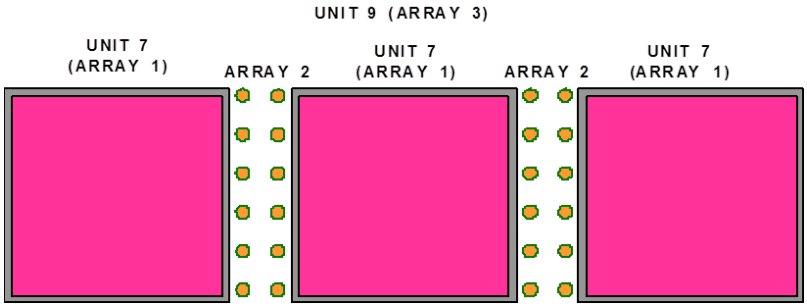


Fig. 8.1.43: Central array.

KENO V.a:

```
ARA=3 NUX=5 NUY=1 NUZ=1    FILL 7 8 7 8 7 END FILL

UNIT 9    ARRAY 3 0 0 0
```

KENO-VI:

```
UNIT 8
CUBOID 4  -11.302238 -16.052324 11.302238 -11.302238 365.76 0.0
CUBOID 5    16.052324 11.302238 11.302236 -11.302236 365.76 0.0
CUBOID 6    38.052324 -38.052324 11.302236 -11.302236 365.76 0.0
HOLE 7
HOLE 7 ORIGIN X= -27.354562
HOLE 7 ORIGIN X= 27.354562
ARRAY 2 4 PLACE 1 1 1 -14.8648025  -9.418530 0.0
ARRAY 2 5 PLACE 1 1 1  12.4897595  -9.418530 0.0
MEDIA 3 1 6 -5 -4
BOUNDARY  6
```

UNITs 3, 4, and 5 are used to define the arrays of B_4C rods that fit above and below the central array, as shown in Fig. 8.1.44.

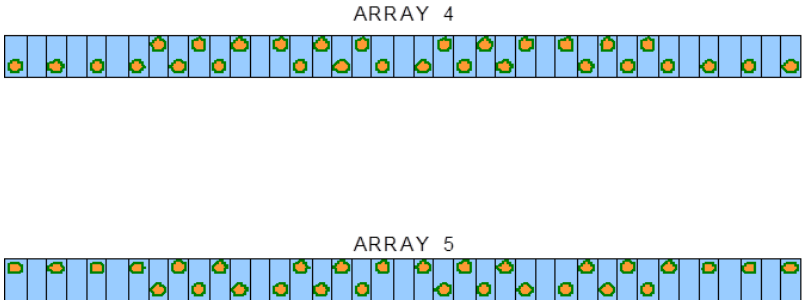


Fig. 8.1.44: Long B_4C rod arrays.

```
ARA=4 NUX=39 NUY=1 NUZ=1   FILL 3 5 2Q2 3 4 2Q2 5 4 3 2Q2 5 3 4 2Q2 5 4 3 2Q2 5 2Q2 3
END FILL

ARA=5 NUX=39 NUY=1 NUZ=1   FILL 4 5 2Q2 4 3 2Q2 5 3 4 2Q2 5 4 3 2Q2 5 3 4 2Q2 5 2Q2 4
END FILL
```

In KENO V.a these ARRAYs are placed in UNITs (UNITs 10 and 11) for further use.

KENO V.a:

```
UNIT 10   ARRAY 4 0 0 0

UNIT 11   ARRAY 5 0 0 0
```

UNITs 9, 10, and 11 in the KENO V.a description, or UNIT 8 and ARRAYs 3 and 4 in the KENO-VI description, are stacked to form the central array with B_4C rods as shown in Fig. 8.1.45.

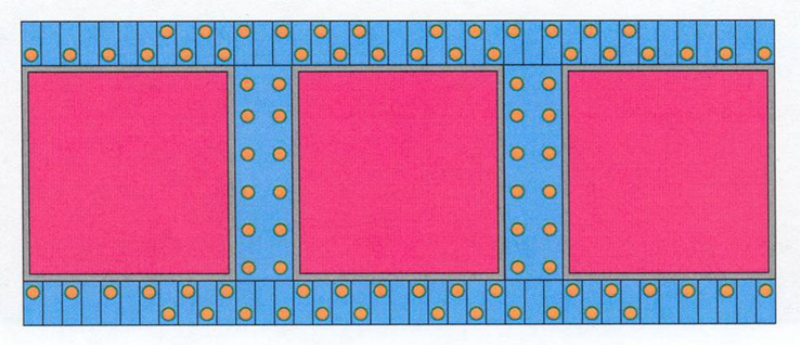


Fig. 8.1.45: Central array with long B_4C arrays.

KENO V.a:

```
ARA=6 NUX=1 NUY=3 NUZ=1    FILL 11 9  10  END FILL
```

KENO-VI:

```
CUBOID 2  38.052324 -38.052324 14.807436  11.302236  365.76 0.0
CUBOID 3  38.052324 -38.052324 11.302236 -14.807436  365.76 0.0
HOLE 8
ARRAY 3 2 PLACE 20 1 1 0.0 13.054836  0.0
ARRAY 4 3 PLACE 20 1 1 0.0 13.054836  0.0
```

This completes the three central fuel assemblies and all the B_4C rods associated with them. Next, UNITs 7 and 8 in KENO V.a geometry, UNIT 7 and ARRAY 2 in the KENO-VI geometry, are stacked together to form the array of two fuel assemblies separated by B_4C rods as shown in Fig. 8.1.46. This is designated as ARRAY 7 and UNIT 12 in KENO V.a, and UNIT 9 in KENO-VI. The origin of UNIT 12 for KENO V.a is specified at the center of the array in the X and Y directions and the bottom of the fuel assemblies (Z=27.94 cm). The origin of UNIT 9 in the KENO-VI description is specified at the center of the B_4C array in the X and Y directions and the bottom of the array in the Z direction.

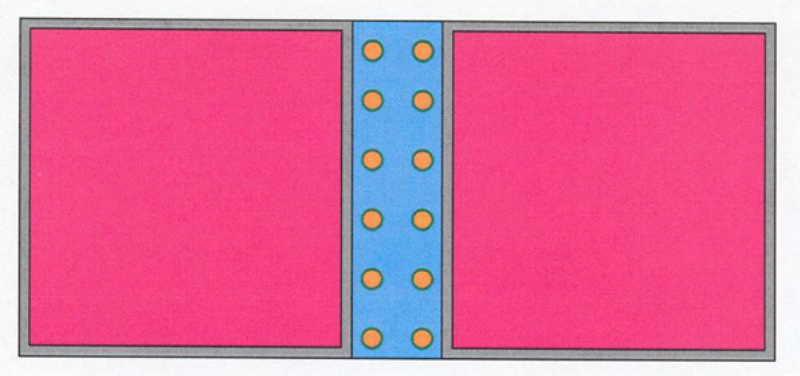


Fig. 8.1.46: Two fuel assemblies and B_4C rods.

KENO V.a:

```
ARA=7 NUX=3 NUY=1 NUZ=1    FILL 7 8 7     END FILL

UNIT 12    ARRAY 7 -24.979519 -11.302238 27.94
```

KENO-VI:

```
UNIT 9
CUBOID 1 2.375043 -2.375043  11.302236 -11.302236 365.76 0.0
CUBOID 4  24.979519  -23.7919975 11.302238 -11.302238 365.76 0.0
ARRAY 2 1 PLACE 1 1 1 -1.1875215 -9.418530 0.0
HOLE 7 ORIGIN X=-13.67728
HOLE 7 ORIGIN X=13.67728
MEDIA 0 1 4 -1
BOUNDARY  4
```

In KENO V.a, UNIT 13 is simply a cylindrical lid that fits on top of the shipping cask. It is described relative to the origin of the shipping cask and is made of depleted uranium.

KENO V.a:

```
UNIT 13
CYLINDER  6 1 47.625 457.2 449.58
```

All necessary subassemblies that make up the shipping cask have been built. The shipping cask is completed by specifying the origin of the central section (ARRAY 6 in KENO V.a) (see Fig. 8.1.45) to be at the center of the array in X and Y and the bottom of the array in the Z direction. A cylinder of water defining the interior of the shipping cask is described around the array. In the KENO V.a geometry, a HOLE is used to place UNIT 12 (Fig. 8.1.46) below the array, and a second HOLE is used to place another UNIT 12 above the array. In the KENO-VI geometry, UNIT 9 is placed as a HOLE above and below the central array. Then a cylinder of steel is placed around the water, which is in turn encased by depleted uranium. The depleted uranium is then contained in the outer steel cylinder of the shipping cask. In KENO V.a description, a third HOLE is used to place the depleted uranium lid (UNIT 13) on the shipping cask. This completes the shipping cask description of Fig. 8.1.38

The geometry data for this shipping cask are shown below. The plot data have been included for verification of the geometry description. However, the plot generated by this data is quite large and is therefore not included in this document.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  1 1 .41148 365.76 0.0
CYLINDER  2 1 .48133 365.76 0.0
CUBOID    3 1 .63754 -.63754 .63754 -.63754 365.76 0.0
UNIT 2
CUBOID    3 1 .63754 -.63754 .63754 -.63754 365.76 0.0
UNIT 3
CYLINDER  4 1 .584 365.76 0.0
CYLINDER  5 1 .635 365.76 0.0
CUBOID    3 1 .9912 -.9912 2.2352 -1.27 365.76 0.0
UNIT 4
CYLINDER  4 1 .584 365.76 0.0
CYLINDER  5 1 .635 365.76 0.0
CUBOID    3 1 .9912 -.9912 1.2702 -2.235 365.76 0.0
UNIT 5
CUBOID    3 1 .9912 -.9912 1.7526 -1.7526 365.76 0.0
UNIT 6
CYLINDER  4 1 .584 365.76 0.0
CYLINDER  5 1 .635 365.76 0.0
CUBOID    3 1 1.1875215 -1.1875215 1.883706 -1.883706 365.76 0.0
UNIT 7    ARRAY 1 -10.83818 -10.83818 0.0
CUBOID    3 1 11.112495 -11.112495 11.112495 -11.112495 365.76 0.0
CUBOID    8 1 11.302238 -11.302238 11.302238 -11.302238 365.76 0.0
UNIT 8    ARRAY 2 0 0 0
UNIT 9    ARRAY 3 0 0 0
UNIT 10   ARRAY 4 0 0 0
UNIT 11   ARRAY 5 0 0 0
UNIT 12   ARRAY 7 -24.979519 -11.302238 27.94
UNIT 13
CYLINDER  6 1 47.625 457.2 449.58
ARRAY     6  -38.6568 -14.807438 27.94
CYLINDER  3 1 47.625 447.04 16.51
HOLE     12    0.0 -26.1097 0.0
HOLE     12    0.0  26.1097 0.0
CYLINDER  7 1 48.895 447.04 13.335
CYLINDER  6 1 59.06 447.04 3.81
CYLINDER  7 1 63.01 462.28 0.0
HOLE     13    0.0 0.0 0.0
END GEOM
READ ARRAY
ARA=1 NUX=17 NUY=17 NUZ=1 FILL
39R1 2 2Q3 8R1 2 9R1 2 22R1 2 4Q3 38R1 2 4Q3
Q51 22R1 2 Q10 Q9 2Q3 39R1                END FILL
ARA=2 NUX=2 NUY=6 NUZ=1    FILL F6        END FILL
ARA=3 NUX=5 NUY=1 NUZ=1    FILL 7 8 7 8 7 END FILL
ARA=4 NUX=39 NUY=1 NUZ=1   FILL 3 5 2Q2 3 4 2Q2 5 4 3 2Q2 5 3 4 2Q2
5 4 3 2Q2 5 2Q2 3                         END FILL
ARA=5 NUX=39 NUY=1 NUZ=1   FILL 4 5 2Q2 4 3 2Q2 5 3 4 2Q2 5 4 3 2Q2
5 3 4 2Q2 5 2Q2 4                         END FILL
ARA=6 NUX=1 NUY=3 NUZ=1    FILL 11 9  10  END FILL
ARA=7 NUX=3 NUY=1 NUZ=1    FILL 7 8 7     END FILL
END ARRAY
READ PLOT
TTL=? SHIPPING CASK IF-300 X-Y SLICE  ?
XUL=-63 YUL=63 ZUL=180 XLR=63 YLR=-63 ZLR=180
UAX=1 VDN=-1 NAX=350
PLT=NO
END PLOT
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER  1 .41148 365.76 0.0
CYLINDER  2 .48133 365.76 0.0
```

(continues on next page)

(continued from previous page)

```
CUBOID  3    .63754 -.63754 .63754 -.63754 365.76 0.0
MEDIA 1 1 1
MEDIA 2 1 2 -1
MEDIA 3 1 3 -2
BOUNDARY  3
UNIT 2
CUBOID  1  .63754 -.63754 .63754 -.63754 365.76 0.0
MEDIA 3 1 1
BOUNDARY  1
UNIT 3
CYLINDER 1   .584 365.76 0.0
CYLINDER 2   .635 365.76 0.0
CUBOID   3    .9912 -.9912 2.2352 -1.27 365.76 0.0
MEDIA 4 1 1
MEDIA 5 1 2 -1
MEDIA 3 1 3 -2
BOUNDARY  3
UNIT 4
CYLINDER 1  .584 365.76 0.0
CYLINDER 2  .635 365.76 0.0
CUBOID  3    .9912 -.9912 1.2702 -2.235 365.76 0.0
MEDIA 4 1 1
MEDIA 5 1 2 -1
MEDIA 3 1 3 -2
BOUNDARY  3
UNIT 5
CUBOID 1   .9912 -.9912 1.7526 -1.7526 365.76 0.0
MEDIA 3 1 1
BOUNDARY  1
UNIT 6
CYLINDER 1  .584 365.76 0.0
CYLINDER 2  .635 365.76 0.0
CUBOID  3    1.1875215 -1.1875215 1.883706 -1.883706 365.76 0.0
MEDIA 4 1 1
MEDIA 5 1 2 -1
MEDIA 3 1 3 -2
BOUNDARY  3
UNIT 7
CUBOID 1  10.83818 -10.83818 10.83818 -10.83818 365.76 0.0
CUBOID 2  11.112495 -11.112495 11.112495 -11.112495 365.76 0.0
CUBOID 3  11.302238 -11.302238 11.302238 -11.302238 365.76 0.0
ARRAY 1 1  PLACE 9 9 1  3*0.0
MEDIA 3 1 2 -1
MEDIA 8 1 3 -2
BOUNDARY  3
UNIT 8
CUBOID 4  -11.302238 -16.052324 11.302238 -11.302238 365.76 0.0
CUBOID 5    16.052324 11.302238 11.302236 -11.302236 365.76 0.0
CUBOID 6    38.052324 -38.052324 11.302236 -11.302236 365.76 0.0
HOLE 7
HOLE 7 ORIGIN X= -27.354562
HOLE 7 ORIGIN X= 27.354562
ARRAY 2 4 PLACE 1 1 1 -14.8648025  -9.418530 0.0
ARRAY 2 5 PLACE 1 1 1  12.4897595  -9.418530 0.0
MEDIA 0 1 6 -5 -4
BOUNDARY  6
UNIT 10
CUBOID 1 1.1875215 -1.1875215  11.302236 -11.302236 365.76 0.0
CUBOID 4  23.7919975 -23.7919975 11.302238 -11.302238 365.76 0.0
ARRAY 2 1 PLACE 1 1 1 3*0.0
HOLE 7 ORIGIN X=-12.4897595
HOLE 7 ORIGIN X=12.4897595
MEDIA 0 1 4 -1
BOUNDARY  4
GLOBAL UNIT 11
```

(continues on next page)

(continued from previous page)

```
CUBOID 2  38.052324 -38.052324 13.284638  11.302236  365.76 0.0
CUBOID 3  38.052324 -38.052324 11.302236 -13.284638 365.76 0.0
CYLINDER 6 47.625 419.10  -11.43
CYLINDER 7   48.895 419.10 -14.605
CYLINDER 8  59.06 419.10 -24.13
CYLINDER 9  47.625 429.26 421.64
CYLINDER 10  63.01 434.34 -27.94
HOLE 8
ARRAY 3 2 PLACE 20 1 1 0.0 12.293438  0.0
ARRAY 4 3 PLACE 20 1 1 0.0 12.293438  0.0
HOLE 9 ORIGIN Y=24.586876
HOLE 9 ORIGIN Y=-24.586876
MEDIA 3 1 6 -3 -2
MEDIA 7 1 7 -6 -3 -2
MEDIA 6 1 8 -7
MEDIA 7 1 9
MEDIA 6 1 10 -9 -8
BOUNDARY  10
END GEOM
READ ARRAY
ARA=1 NUX=17 NUY=17 NUZ=1 FILL
39R1 2 2Q3 8R1 2 9R1 2 22R1 2 4Q3 38R1 2 4Q3
Q51 22R1 2 Q10 Q9 2Q3 39R1                END FILL
ARA=2 NUX=2 NUY=6 NUZ=1    FILL F6        END FILL
ARA=3 NUX=5 NUY=1 NUZ=1    FILL 7 8 7 8 7 END FILL
ARA=4 NUX=39 NUY=1 NUZ=1   FILL 3 5 2Q2 3 4 2Q2 5 4 3 2Q2 5 3 4 2Q2
5 4 3 2Q2 5 2Q2 3                         END FILL
ARA=5 NUX=39 NUY=1 NUZ=1   FILL 4 5 2Q2 4 3 2Q2 5 3 4 2Q2 5 4 3 2Q2
5 3 4 2Q2 5 2Q2 4                         END FILL
ARA=6 NUX=1 NUY=3 NUZ=1    FILL 11 9  10  END FILL
ARA=7 NUX=3 NUY=1 NUZ=1    FILL 7 8 7     END FILL
END ARRAY
READ PLOT
TTL=? SHIPPING CASK IF-300 X-Y SLICE  ?
XUL=-63 YUL=63 ZUL=180 XLR=63 YLR=-63 ZLR=180
UAX=1 VDN=-1 NAX=350
PLT=NO
END PLOT
```

Triangular pitched arrays in KENO-VI

EXAMPLE 21.

Triangular pitched ARRAYs can be described in KENO-VI by defining the UNITs that make up the ARRAY as HEXPRISM and in the array data block setting TYP=TRIANGULAR, HEXAGONAL, SHEXAGONAL, or RHEXAGONAL. This includes close-packed triangular pitched arrays. Since the ARRAYs are constructed by stacking hexprisms, care must be taken to ensure that the ARRAY boundary is completely enclosed within the stacked UNIT. Below is an example of a triangular pitched ARRAY.

The first and second UNITs are the HEXPRISMs that make up the ARRAY. UNIT 1 is the fuel cell stacked in a triangular pitched or hexagonal lattice. UNIT 2 is a dummy UNIT used to fill in the ARRAY so the array boundary is contained within the stacked UNITs. Since the ARRAY is not moderated, UNIT 2 contains void. Fig. 8.1.47 shows an X-Y cross section schematic of UNITs 1 and 2.

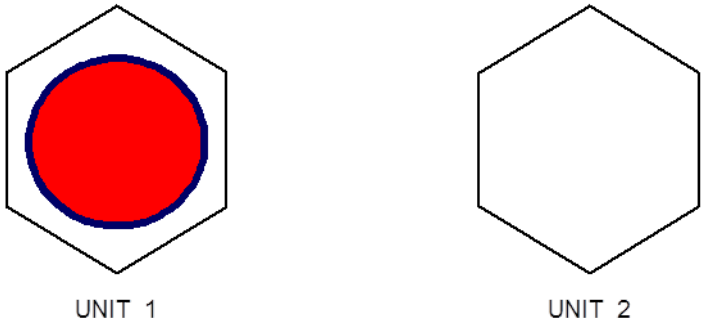


Fig. 8.1.47: Fuel cell and empty cell set up as hexprisms.

```
UNIT  1
COM='SINGLE CELL FUEL CAN IN HEXPRISM'
CYLINDER 10  10.16  18.288  0.0
CYLINDER 20  10.312 18.288 -0.152
HEXPRISM 30  10.503 18.288 -0.152
MEDIA  1  1  10
MEDIA  2  1  20  -10
MEDIA  0  1  30  -20
BOUNDARY  30
UNIT  2
COM='EMPTY CELL'
HEXPRISM  10  10.503  18.288  -0.152
MEDIA  0  1  10
BOUNDARY  10
```

UNIT 3 is the GLOBAL UNIT that contains the ARRAY and ARRAY BOUNDARY. The ARRAY is an unmoderated triangular pitched assembly of 7 cells. Unrotated triangular or hexagonal pitched arrays can be set up in two array configurations. The first configuration, selected using the TYP= followed by keyword HEXAGONAL or TRIANGULAR, stacks the UNITs so that the faces perpendicular to the X axis meet. Each consecutive row in the Y direction begins 1/2 of the face-to-face dimension farther over in the positive X direction than in the previous row. The second configuration, selected using the TYP= followed by keyword SHEXAGONAL, also stacks the UNITs so that the faces perpendicular to the X axis meet. However, for this type of ARRAY, the odd numbered rows in the Y direction (1, 3, 5, etc.) begin at the left edge of the ARRAY, and the even numbered rows in the Y direction (2, 4, 6, etc.) begin 1/2 of the face-to-face dimension to the right of the left edge of the ARRAY. Fig. 8.1.48 and Fig. 8.1.49 show X-Y cross section schematics of the assembly for the two different unrotated hexagonal ARRAY types.

```
GLOBAL UNIT 3
COM='7 CYLINDERS IN A CIRCLE WITH CYLINDRICAL BOUNDARY'
CYLINDER 10  32.000  18.288  -0.152
ARRAY  1  10 PLACE 3 3 1 3*0.0
BOUNDARY 10
READ ARRAY GBL=1 ARA=1 TYP=HEXAGONAL NUX=5 NUY=5 NUZ=1
FILL 7*2 2*1 2*2 3*1 2*2 2*1 7*2 END FILL  END ARRAY
```

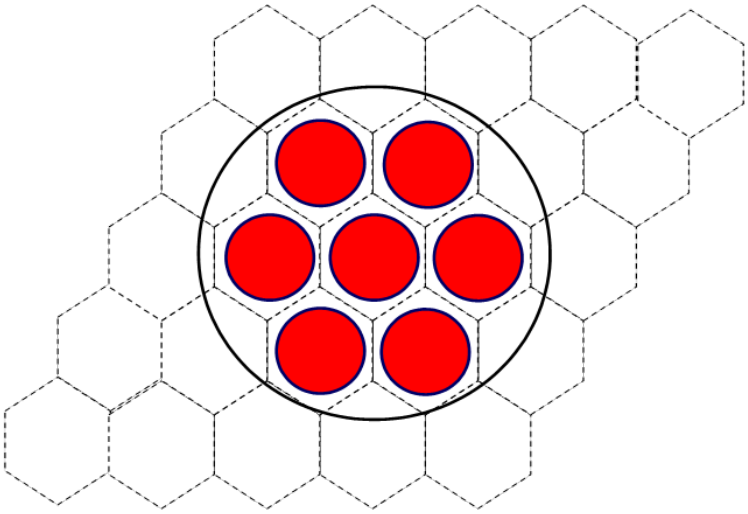


Fig. 8.1.48: Seven cylinders stacked in a HEXAGONAL array with cylindrical array boundary.

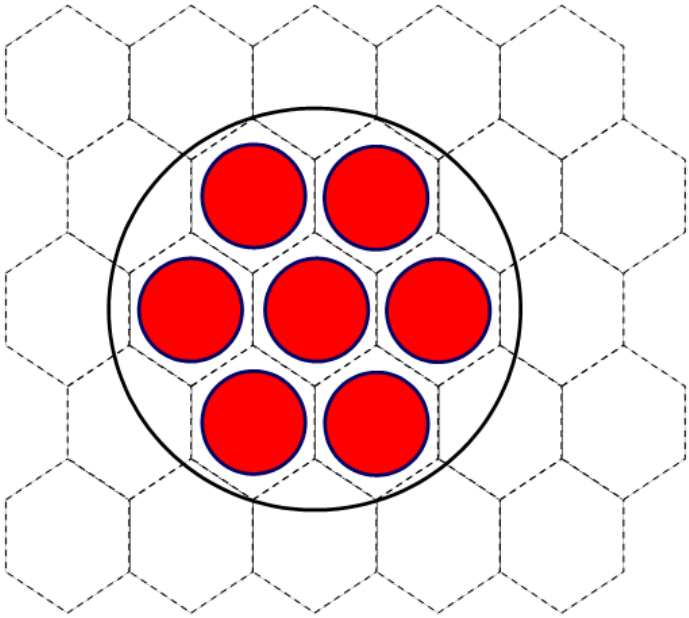


Fig. 8.1.49: Seven cylinders stacked in a SHEXAGONAL array with a cylindrical array boundary.

The overall problem description is shown below. The cross section library would be generated in a separate CSAS-MG step. An X-Y cross section color plot used for verification of this mockup is shown in Fig. 8.1.50.

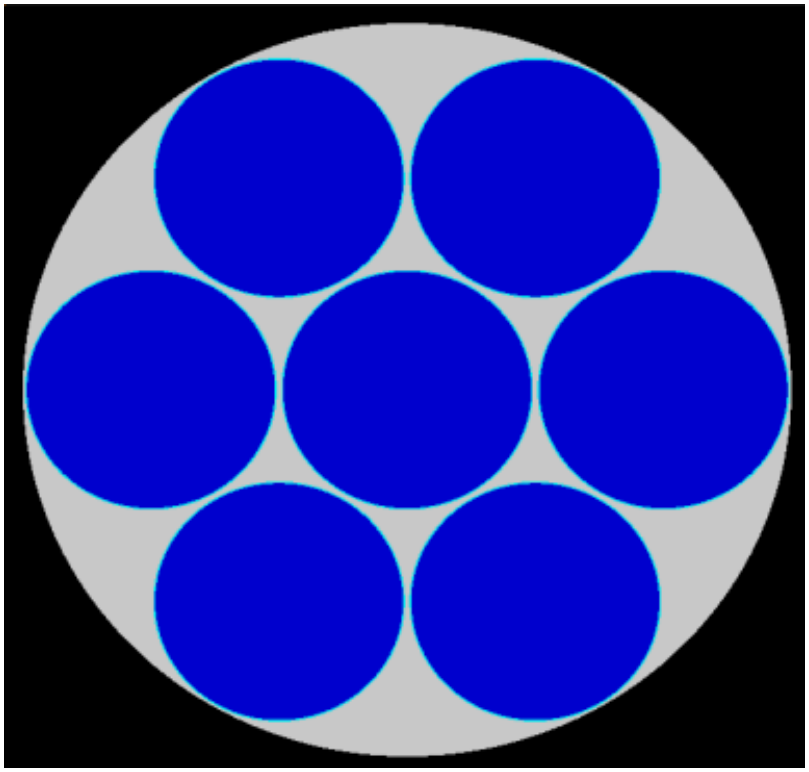


Fig. 8.1.50: X-Y slice of 7 cylinders in a triangular pitch array.

Data description of Example 21.

```
=KENOVI
TRIANGULAR PITCHED ARRAY 7 PINS IN A CIRCLE
READ PARAMETERS LNG=20000 LIB=4 END PARAMETERS
READ MIXT SCT=2
MIX=1 NCM=8 92235 1.37751E-03  92238 9.92354E-05  8016 3.32049E-02
       9019 2.95349E-03  1001 6.05028E-02
MIX=2 NCM=14 13027 6.02374E-02
END MIXT
READ GEOMETRY
UNIT 1
COM='SINGLE CELL FUEL CAN IN HEXPRISM'
CYLINDER 10  10.16  18.288  0.0
CYLINDER 20  10.312 18.288 -0.152
HEXPRISM 30  10.503 18.288 -0.152
MEDIA  1 1 10
MEDIA  2 1 20 -10
MEDIA 0 1 30 -20
BOUNDARY  30
UNIT 2
COM='EMPTY CELL'
HEXPRISM 10  10.503 18.288 -0.152
MEDIA  0 1 10
BOUNDARY  10
GLOBAL UNIT 3
CYLINDER 10  32.000 18.288 -0.152
COM='7 CYLINDERS IN A CIRCLE WITH CYLINDRICAL BOUNDARY'
ARRAY  1 10 PLACE 3 3 1 3*0.0
BOUNDARY  10
END GEOMETRY
READ ARRAY  GBL=1 TYP=HEXAGONAL NUX=5 NUY=5 NUZ=1
```

(continues on next page)

(continued from previous page)

```
FILL 7*2 2*1 2*2 3*1 2*2 2*1 7*2 END FILL  END ARRAY
READ PLOT
TTL='TRIANGULAR PITCHED ARRAY, 7 PINS IN A CIRCLE'
XUL=-33.0  YUL=33.0  ZUL=0.0
XLR=33.0  YLR=-33.0  ZLR=0.0
UAX=1  VDN=-1  NAX=640    END
END PLOT
END DATA
END
```

EXAMPLE 21a.

Another hexagonal ARRAY type involves stacking rotated hexprisms, which are hexprisms rotated 30°/ 90° so that the flat faces perpendicular to the Y axis now meet. Rotated hexagonal arrays are specified by setting TYP=RHEXAGONAL in the array data block. Because the ARRAYs are constructed by stacking hexprisms, care must be taken to ensure the array boundary is completely enclosed within the stacked UNIT. Below is an example of a rotated hexagonal pitched ARRAY.

The first and second UNITs are the rotated hexprisms that make up the ARRAY. They are specified using the geometry keyword RHEXPRISM. UNIT 1 is the fuel cell that is stacked in a rotated hexagonal lattice. UNIT 2 is a dummy UNIT used to fill in the ARRAY so that the ARRAY BOUNDARY is contained within the stacked UNITs. Since the ARRAY is not moderated, UNIT 2 contains void. Fig. 8.1.51 shows an X-Y cross section schematic of UNITs 1 and 2.

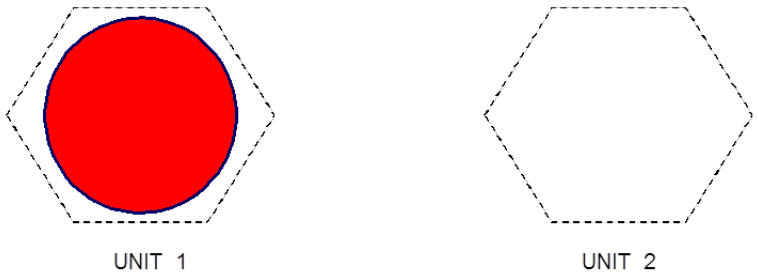


Fig. 8.1.51: Fuel cell and empty cell set up as rotated hexprism (RHEXPRISM).

```
UNIT  1
COM='SINGLE CELL FUEL CAN IN HEXPRISM'
CYLINDER 10  10.16  18.288  0.0
CYLINDER 20  10.312 18.288 -0.152
RHEXPRISM 30  10.503 18.288 -0.152
MEDIA  1  1  10
MEDIA  2  1  20  -10
MEDIA  0  1  30  -20
BOUNDARY  30
UNIT  2
COM='EMPTY CELL'
RHEXPRISM  10  10.503  18.288  -0.152
MEDIA  0  1  10
BOUNDARY  10
```

UNIT 3 is the GLOBAL UNIT that contains the ARRAY and ARRAY BOUNDARY. The ARRAY is an unmoderated rotated hexagonal pitched assembly of 7 cells. The rotated hexagonal array type is specified in the array data block using TYP= with the keyword RHEXAGONAL. This ARRAY type stacks the UNITs so the faces perpendicular to the Y axis meet. In the odd numbered columns (i.e., 1, 3, 5, etc.), the UNITs are stacked so

the columns begin at the lower edge of the array and in the even numbered columns (i.e., 2, 4, 6, etc.), the UNITs are stacked so the columns begin 1/2 of the face-to-face dimension above the lower edge of the ARRAY. Fig. 8.1.52 shows the X-Y cross section schematic of the assembly for the rotated hexagonal ARRAY type.

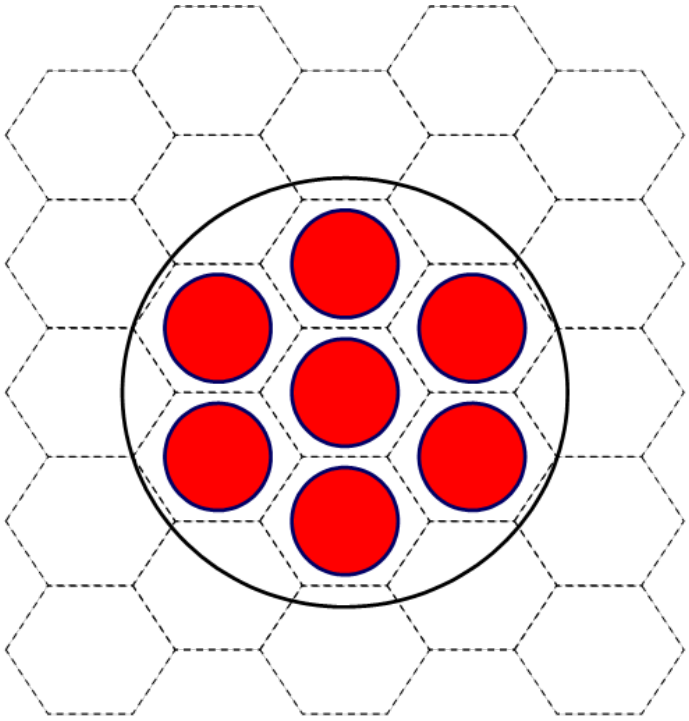


Fig. 8.1.52: Seven cylinders stacked in a RHEXAGONAL array with a cylindrical array boundary.

```
GLOBAL UNIT 3
COM='7 CYLINDERS IN A CIRCLE WITH CYLINDRICAL BOUNDARY'
CYLINDER 10  32.000  18.288  -0.152
ARRAY  1  10 PLACE 3 3 1 3*0.0
BOUNDARY 10
READ ARRAY GBL=1 ARA=1 TYP=RHEXAGONAL NUX=5 NUY=5 NUZ=1
FILL 6*2 3*1 2*2 3*1 3*2 1*1 7*2 END FILL  END ARRAY
```

The overall problem description is shown below. The cross section library would be generated in a separate CSAS-MG step. An X-Y cross section color plot used for verification of this mockup is shown in Fig. 8.1.53.

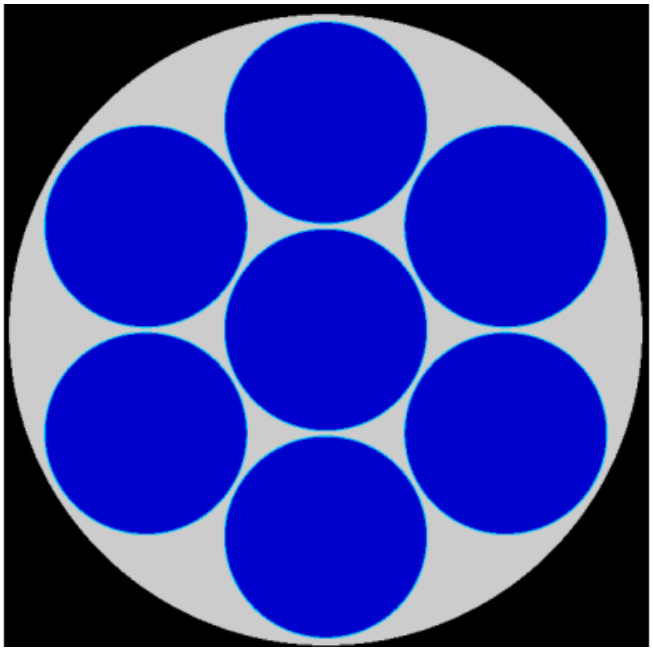


Fig. 8.1.53: X-Y slice of 7 cylinders in a rotated hexagonal pitched array.

Data description of Example 21a.

```
=KENOVI
TRIANGULAR PITCHED ARRAY 7 PINS IN A CIRCLE
READ PARAMETERS LNG=20000 LIB=4 END PARAMETERS
READ MIXT SCT=2
MIX=1 NCM=8 92235 1.37751E-03  92238 9.92354E-05  8016 3.32049E-02
       9019 2.95349E-03  1001 6.05028E-02
MIX=2 NCM=14 13027 6.02374E-02
END MIXT
READ GEOMETRY
UNIT 1
COM='SINGLE CELL FUEL CAN IN HEXPRISM'
CYLINDER 10  10.16  18.288  0.0
CYLINDER 20  10.312 18.288 -0.152
RHEXPRISM 30  10.503 18.288 -0.152
MEDIA  1 1 10
MEDIA  2 1 20 -10
MEDIA 0 1 30 -20
BOUNDARY  30
UNIT 2
COM='EMPTY CELL'
RHEXPRISM 10  10.503 18.288 -0.152
MEDIA  0 1 10
BOUNDARY  10
GLOBAL UNIT 3
CYLINDER 10  32.000 18.288 -0.152
COM='7 CYLINDERS IN A CIRCLE WITH CYLINDRICAL BOUNDARY'
ARRAY  1 10 PLACE 3 3 1 3*0.0
BOUNDARY  10
END GEOMETRY
READ ARRAY  GBL=1 TYP=RHEXAGONAL NUX=5 NUY=5 NUZ=1
FILL 6*2 3*1 2*2 3*1 3*2 1*1 7*2 END FILL  END ARRAY
READ PLOT
TTL='ROTATED HEXAGONAL ARRAY, 7 PINS IN A CIRCLE'
XUL=-33.0  YUL=33.0  ZUL=0.0
```

(continues on next page)

(continued from previous page)

```
XLR=33.0  YLR=-33.0  ZLR=0.0
UAX=1  VDN==-1  NAX=640    END
END PLOT
END DATA
END
```

Triangular pitched Arrays in KENO V.a

Triangular pitched arrays can be described in KENO V.a geometry by properly defining the basic unit from which the array can be built. This includes close-packed triangular pitched arrays. Two geometry configurations are described below.

EXAMPLE 1. Bare Triangular pitched Array.

Fig. 8.1.54 illustrates a small close-packed triangular pitched **ARRAY**. Each rod in the **ARRAY** has a radius of 2.0 cm, and the pitch of the **ARRAY** is 4 cm. To create this **ARRAY**, describe five units as defined in Fig. 8.1.55.

Assume the rods described in the **ARRAY** are each 2.0 cm in radius and 100 cm tall. The rods are composed of mixture 1. The geometry descriptions for the first four **UNIT**s are given below.

```
UNIT 1
ZHEMICYL-Y 1 1 2.0 50.0 -50.0

UNIT 2
ZHEMICYL+Y 1 1 2.0 50.0 -50.0

UNIT 3
ZHEMICYL-X 1 1 2.0 50.0 -50.0

UNIT 4
ZHEMICYL+X 1 1 2.0 50.0 -50.0
```

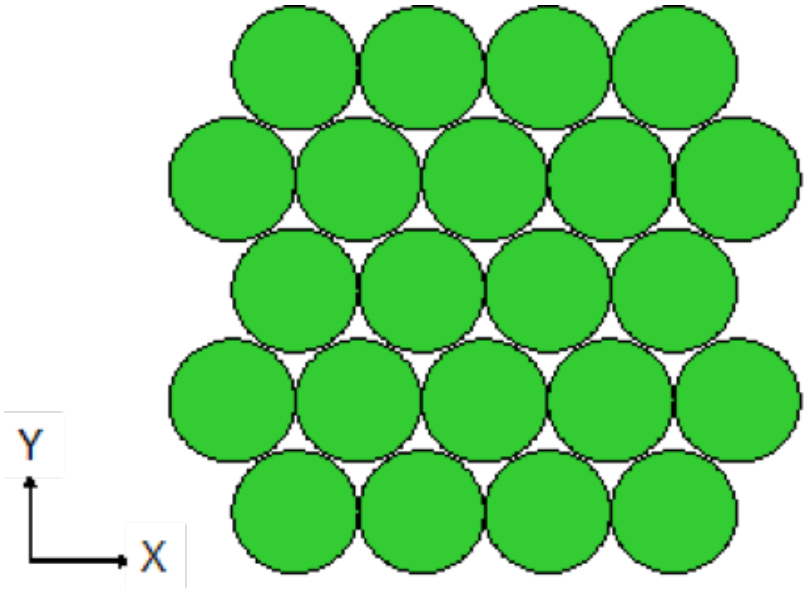


Fig. 8.1.54: Bare triangular pitched ARRAY.

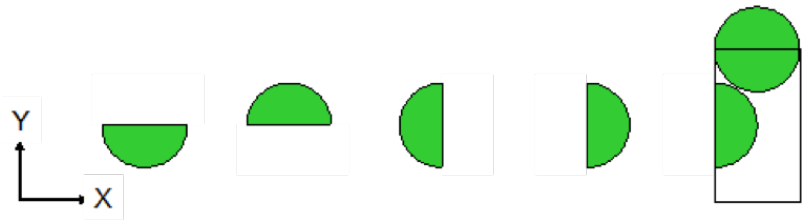


Fig. 8.1.55: Units used to describe a bare triangular pitched ARRAY.

To describe UNIT 5, the origin of the UNIT is defined to be at its center. One of the hemicylinders is built into the box, and the other three are added as holes. In this example, the +X hemicylinder is built into the box, and the other hemicylinders are inserted as holes. (Because the +X hemicylinder is built into UNIT 5, UNIT 4 is not used in the problem.) The half dimension of the box in the X dimension is equal to the radius, 2.0 cm. The half dimension of the box in the Y direction is $\frac{\sqrt{3}}{2}$ times the pitch (0.866025 * 4.0) or 3.46411 cm. UNIT 5 is described below.

```
UNIT 5
ZHEMICYL+X  1 1 2.0 50.0 -50.0 ORIGIN -2.0 0.0
CUBOID   0 1 2P2.0 2P3.46411 2P50.0
HOLE 1 0.0   3.46411    0.0
HOLE 2 0.0 -3.46411    0.0
HOLE 3 2.0             0.0
```

In the description of UNIT 5, the ZHEMICYL+X places the hemicylinder at the left of the box. HOLE 1 places the top hemicylinder, HOLE 2 places the bottom hemicylinder, and HOLE 3 places the hemicylinder at the right of the box.

Next, a UNIT 6 is defined that can be used to complete the lower rod of UNIT 5. A UNIT 7 is defined that can be used to complete the upper rod of UNIT 5. UNIT 8 is defined to complete the left rod of UNIT 5, and UNIT 9 is defined to complete the right rod of UNIT 5. UNIT 10 is defined to complete the corners of the overall ARRAY. The input data for these UNITs are given below and are illustrated in Fig. 8.1.56.

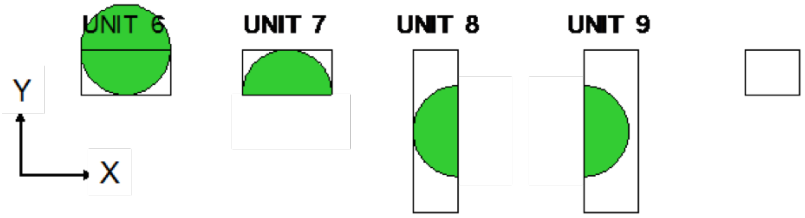


Fig. 8.1.56: UNITs to complete the triangular pitched ARRAY.

```
UNIT 6
ZHEMICYL-Y       1 1 2.0 50.0 -50.0
CUBOID           0 1 2P2.0 0.0 -2.0 50.0 -50.0

UNIT 7
ZHEMICYL+Y   1 1 2.0 50.0 -50.0
CUBOID           0 1 2P2.0 2.0 0.0 2P50.0

UNIT 8
ZHEMICYL-X       1 1 2.0 50.0 -50.0
CUBOID           0 1 0.0 -2.0 2P3.46411 2P50.0
```

(continues on next page)

(continued from previous page)

```
UNIT 9
ZHEMICYL+X   1 1 2.0 50.0 -50.0
CUBOID           0 1 2.0 0.0 2P3.46411 2P50.0

UNIT 10
CUBOID           0 1 2.0 0.0 2.0 0.0 2P50.0
```

Fig. 8.1.57 shows the arrangement of the UNITs to complete the ARRAY. The data to describe the ARRAY are shown below.

```
ARA=1 NUX=6 NUY=4 NUZ=1
FILL 10 4R6 10 8 4R5 9 1Q6 10 4R7 10 END FILL
```

The bottom row of the ARRAY is described by the data entries 10 4R6 10. The second row of the ARRAY is described by the data entries 8 4R5 9. The third row is filled by repeating the previous six entries (1Q6). It could also have been described by entering 8 4R5 9. The top row of the ARRAY is described by the data entries 10 4R7 10.

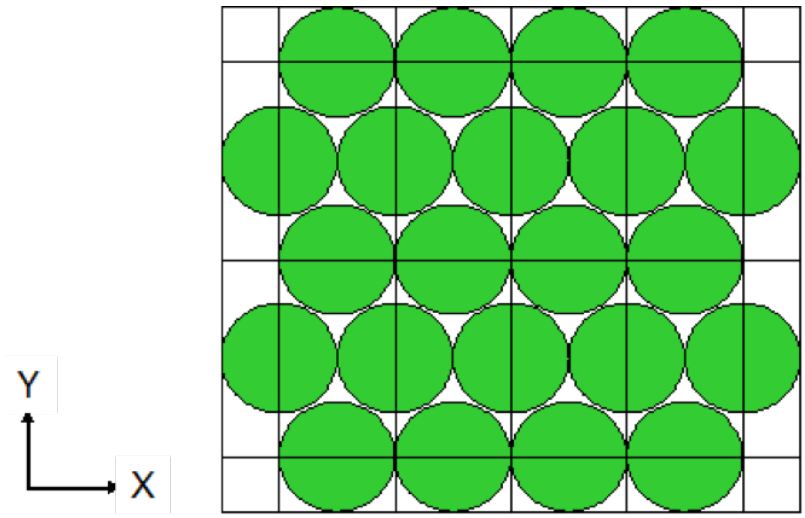


Fig. 8.1.57: Completed ARRAY.

EXAMPLE 2a. Triangular Pitched ARRAY in a Cylinder

Fig. 8.1.58 illustrates a close-packed triangular pitched **ARRAY** in a cylinder. This array may be described by defining five basic units that are the same as those of Example 1 shown in Fig. 8.1.55.

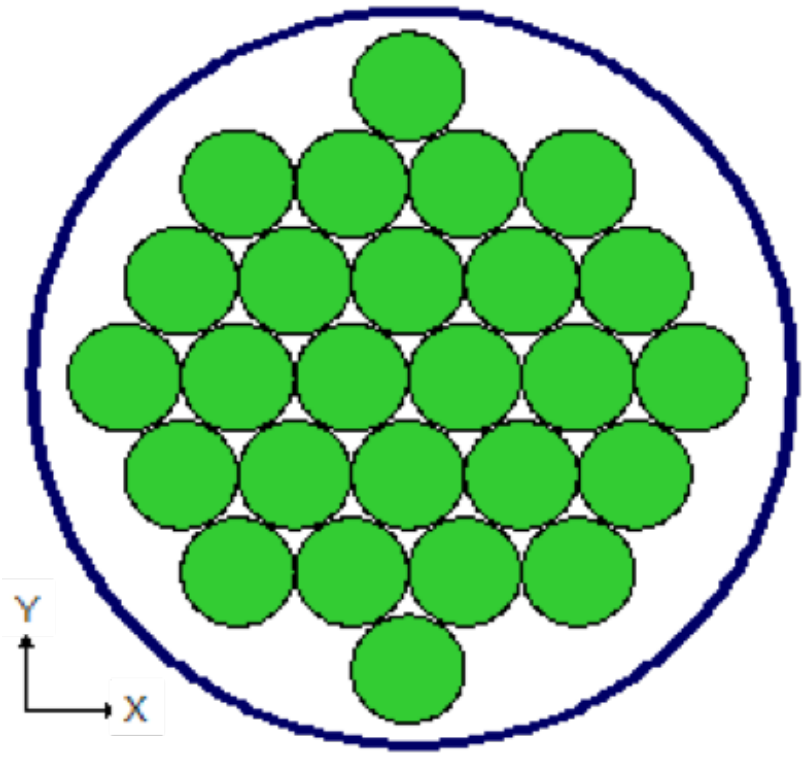


Fig. 8.1.58: Triangular pitched ARRAY within a cylinder.

```
UNIT 1
ZHEMICYL-Y 1 1 2.0 50.0 -50.0

UNIT 2
ZHEMICYL+Y 1 1 2.0 50.0 -50.0

UNIT 3
ZHEMICYL-X 1 1 2.0 50.0 -50.0

UNIT 4
ZHEMICYL+X 1 1 2.0 50.0 -50.0
```

To describe UNIT 5, the origin of the UNIT to be at its center is defined. One of the hemicylinders is built into the box, and the other three are added as HOLEs. In this example, the +X hemicylinder is built into the box, and the other hemicylinders are inserted as HOLEs. The half dimension of the box in the X dimension is equal to the radius, 2.0 cm. The half dimension of the box in the Y direction is $\frac{\sqrt{3}}{2}$ times the pitch (0.866025 * 4.0) or 3.46411 cm. UNIT 5 is described below.

```
UNIT 5
ZHEMICYL+X 1 1 2.0 50.0 -50.0 ORIGIN -2.0 0.0
CUBOID     0 1 2P2.0 2P3.46411 2P50.0
HOLE 1  0.0  3.46411 0.0
HOLE 2  0.0 -3.46411 0.0
HOLE 3  2.0  0.0     0.0
```

In the description of UNIT 5, the ZHEMICYL+X places the hemicylinder at the left of the box. HOLE 1 places the top hemicylinder, HOLE 2 places the bottom hemicylinder, and HOLE 3 places the hemicylinder at the right of the box.

To describe the base ARRAY of the problem, UNITs 5 is stacked in a $4 \times 2 \times 1$ array as shown in Fig. 8.1.59. The input data for the ARRAY are the following:

```
ARA=1 NUX=4 NUY=2 NUZ=1  FILL F5 END FILL
```

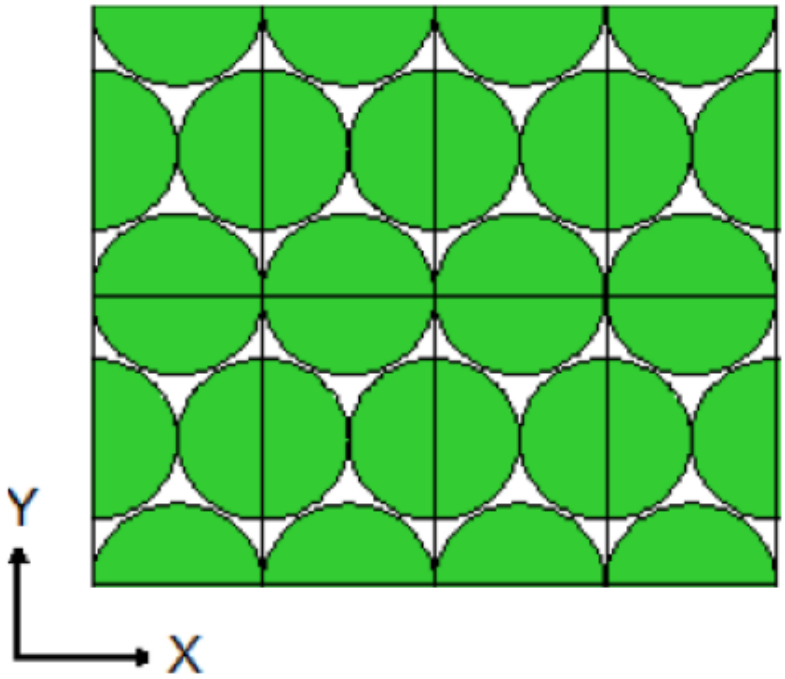


Fig. 8.1.59: $4 \times 2 \times 1$ array to be placed within a cylinder.

Next, the ARRAY is placed within the cylinder. This is done by placing the ARRAY in a UNIT, defined here to be UNIT 6. The origin of the cylinder has been defined to be at the center of the ARRAY. The resulting geometry is shown in Fig. 8.1.60.

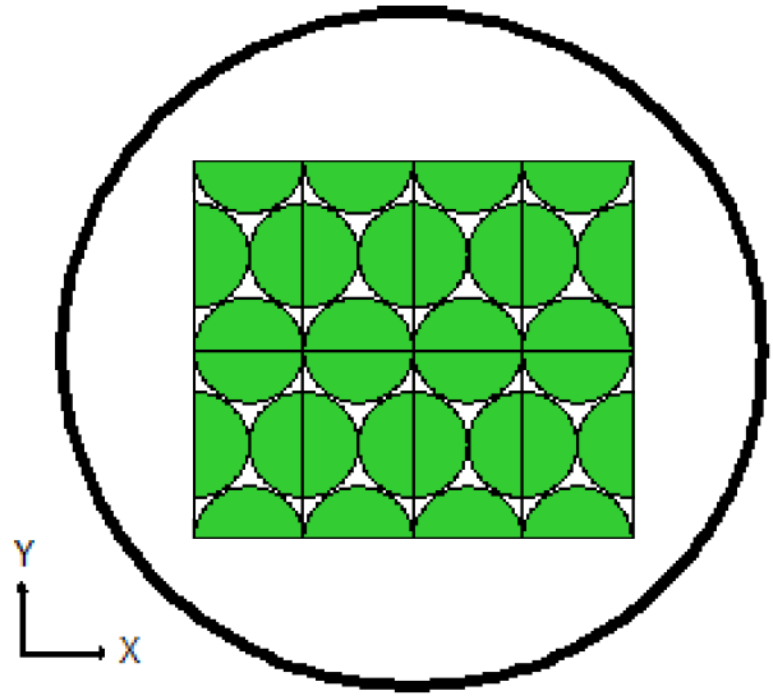


Fig. 8.1.60: $4 \times 2 \times 1$ ARRAY within a cylinder.

```
UNIT 6
ARRAY 1 -8.0 -6.92822 -50.0
CYLINDER 0 1 12.4  2P50.0
CYLINDER 2 1 12.65 2P50.0
```

Next, the hemicylinders necessary to complete all of the half cylinders remaining in Fig. 8.1.60 are added. This is done by placing four UNITs 1 at the appropriate positions along the bottom of the ARRAY, four UNITs 2 at the top of the ARRAY, two UNITs 3 at the left of the ARRAY, and two UNITs 4 at the right of the ARRAY. The input data are shown below, and the resulting configuration is shown in Fig. 8.1.61. In UNIT 6, the first HOLE 1 places a UNIT 1 under the lower left UNIT of the ARRAY. The second HOLE 1 places a UNIT 1 under the next ARRAY UNIT to the right of the first one. This procedure is repeated for the next two lower ARRAY UNITs, thus completing the lower row of cylinders. Similarly, the first HOLE 2 places a UNIT 2 above the upper left UNIT of the ARRAY. The second HOLE 2 places a UNIT 1 to the right of the first one, etc., until the four cylinders at the top of the ARRAY are completed. The first HOLE 3 places a UNIT 3 at the lower left side of the ARRAY to complete that rod. The second HOLE 3 completes the rod above it. The first HOLE 4 completes the lower rod on the right side of the ARRAY. The second HOLE 4 completes the rod above it. The geometry data listed below result in the configuration shown in Fig. 8.1.61.

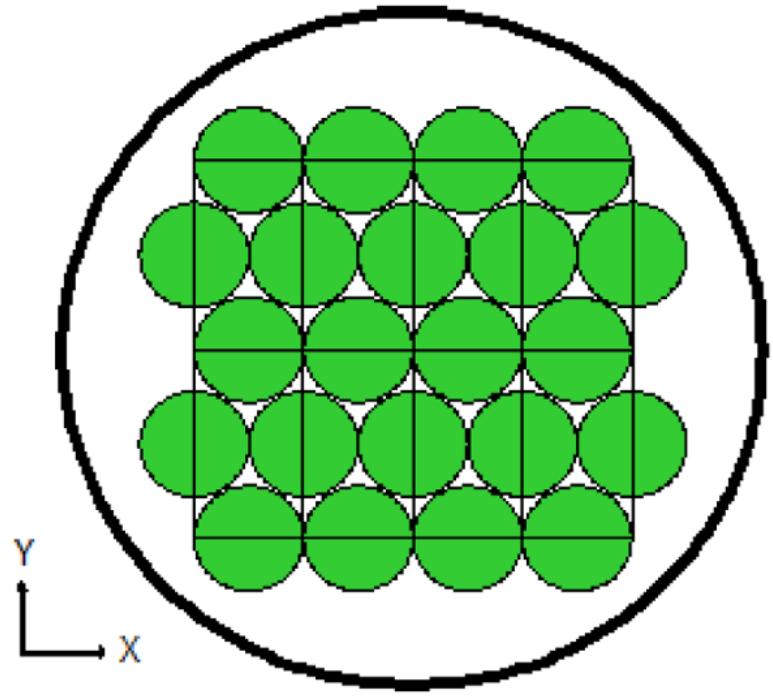


Fig. 8.1.61: Partially completed triangular pitched ARRAY in a cylinder.

```
UNIT 6
ARRAY 1 -8.0 -6.92822 -50.0
CYLINDER 0 1 12.4  2P50.0
HOLE 1  -6.0 -6.92822 0.0
HOLE 1  -2.0 -6.92822 0.0
HOLE 1   2.0 -6.92822 0.0
HOLE 1   6.0 -6.92822 0.0
HOLE 2  -6.0  6.92822 0.0
HOLE 2  -2.0  6.92822 0.0
HOLE 2   2.0  6.92822 0.0
HOLE 2   6.0  6.92822 0.0
HOLE 3  -8.0 -3.46411 0.0
HOLE 3  -8.0  3.46411 0.0
HOLE 4   8.0 -3.46411 0.0
HOLE 4   8.0  3.46411 0.0
CYLINDER 2 1 12.65 2P50.0
```

To complete the desired configuration, a cylinder is defined, UNIT 7, and it is placed at the four appropriate positions as shown below. The first HOLE 7 places the cylinder of UNIT 7 at the left of the ARRAY, the second HOLE 7 places the cylinder at the top of the ARRAY, the third HOLE 7 places the cylinder at the right of the ARRAY, and the fourth HOLE 7 places the cylinder at the bottom of the ARRAY. The completed configuration is shown in Fig. 8.1.62. It is not necessary for UNIT 7 to precede UNIT 6. It is allowable to place UNIT 7 after UNIT 6 in the input data. Because the final configuration is defined in UNIT 6, it must be designated as the GLOBAL UNIT. The total geometry input for this example is listed below.

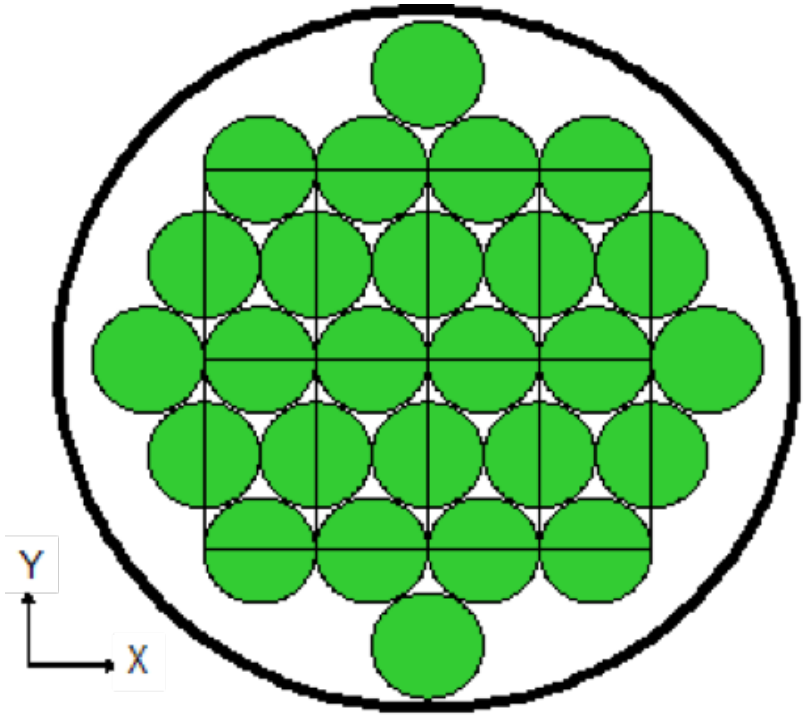


Fig. 8.1.62: Completed triangular pitched array in a cylinder.

```
READ GEOM
UNIT 1
ZHEMICYL-Y 1 1 2.0 50.0 -50.0

UNIT 2
ZHEMICYL+Y 1 1 2.0 50.0 -50.0

UNIT 3
ZHEMICYL-X 1 1 2.0 50.0 -50.0

UNIT 4
ZHEMICYL+X 1 1 2.0 50.0 -50.0

UNIT 5
ZHEMICYL+X 1 1 2.0 50.0 -50.0 ORIGIN -2.0 0.0
CUBOID     0 1 2P2.0 2P3.46411 2P50.0
HOLE 1  0.0  3.46411 0.0
HOLE 2  0.0 -3.46411 0.0
HOLE 3  2.0  0.0     0.0

UNIT 7
CYLINDER 1 1 2.0 2P50.0

GLOBAL UNIT 6
ARRAY 1 -8.0 -6.92822 -50.0
CYLINDER 0 1  12.4  2P50.0
HOLE 1  -6.0  -6.92822 0.0
HOLE 1  -2.0  -6.92822 0.0
HOLE 1   2.0  -6.92822 0.0
HOLE 1   6.0  -6.92822 0.0
HOLE 2  -6.0   6.92822 0.0
HOLE 2  -2.0   6.92822 0.0
HOLE 2   2.0   6.92822 0.0
HOLE 2   6.0   6.92822 0.0
```

(continues on next page)

(continued from previous page)

```
HOLE 3  -8.0  -3.46411 0.0
HOLE 3  -8.0   3.46411 0.0
HOLE 4   8.0  -3.46411 0.0
HOLE 4   8.0   3.46411 0.0
HOLE 7 -10.0   0.0     0.0
HOLE 7   0.0  10.39233 0.0
HOLE 7  10.0   0.0     0.0
HOLE 7   0.0 -10.39233 0.0
CYLINDER 2 1  12.65 2P50.0
END GEOM

READ ARRAY
ARA=1 NUX=4 NUY=2 NUZ=1  FILL F5 END FILL
END ARRAY
```

EXAMPLE 2b. Alternative Mockup of Triangular Pitched ARRAY in a Cylinder.

Consider the triangular pitched ARRAY shown in Fig. 8.1.58. Another method of describing this configuration is given below. Four basic UNITs are defined, as listed below. These are the same UNITs shown in Fig. 8.1.55.

```
UNIT 1
ZHEMICYL-Y 1 1 2.0 50.0 -50.0

UNIT 2
ZHEMICYL+Y 1 1 2.0 50.0 -50.0

UNIT 3
ZHEMICYL-X 1 1 2.0 50.0 -50.0

UNIT 4
ZHEMICYL+X 1 1 2.0 50.0 -50.0
```

UNIT 5 is the same as previously defined in Example 2a, and pictured in Fig. 8.1.55.

```
UNIT 5
ZHEMICYL+X 1 1 2.0 50.0 -50.0 ORIGIN -2.0 0.0
CUBOID     0 1 2P2.0 2P3.46411 2P50.0
HOLE 1  0.0  3.46411 0.0
HOLE 2  0.0 -3.46411 0.0
HOLE 3  2.0  0.0     0.0
```

To describe the basic array for the problem, UNITs 5 is stacked in a $4 \times 2 \times 1$ as shown in Fig. 8.1.59. The input data for the ARRAY are the following:

```
ARA=1 NUX=4 NUY=2 NUZ=1  FILL F5 END FILL
```

Next, the ARRAY (ARRAY 1) is placed in UNIT 6, and UNITs 7 and 8 are defined to be placed to the left and right of it (see Fig. 8.1.63). UNIT 7 will complete the two rods at the left boundary of the ARRAY and will contain half of the far left rod in the completed configuration. In the description of UNIT 7, the ZHEMICYL+X is half of the far right rod and is located with its cut face at the left boundary of a box that is as tall as the entire ARRAY of Fig. 8.1.59. The first HOLE 3 in UNIT 7 completes the lower left rod of that ARRAY, and the second HOLE 3 completes the upper left rod. UNIT 8 is constructed in similar fashion to complete the two rods at the right of the ARRAY shown in Fig. 8.1.59. UNIT 8 is the mirror image of UNIT 7. UNITs 6, 7, and 8 are stacked in an ARRAY (ARRAY 2) to achieve the configuration shown in Fig. 8.1.63. The data to accomplish this are listed below.

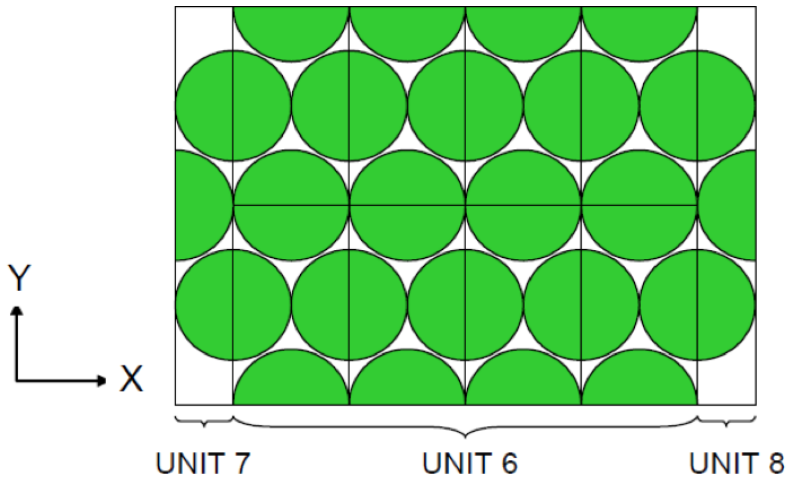


Fig. 8.1.63: Array to be placed in a cylinder.

```
UNIT 6
ARRAY 1 -8.0 -6.92822 -50.0

GLOBAL
UNIT 7
ZHEMICYL+X 1 1 2.0 50.0 -50.0
CUBOID     0 1 2.0 0.0 2P6.92822 2P50.0
HOLE 3 2.0 -3.46411 0.0
HOLE 3 2.0  3.46411 0.0

UNIT 8
ZHEMICYL-X 1 1 2.0 50.0 -50.0
CUBOID     0 1 0.0 -2.0 2P6.92822 2P50.0
HOLE 4 -2.0 -3.46411 0.0
HOLE 4 -2.0  3.46411 0.0

ARA=2 NUX=3 NUY=1 NUZ=1 FILL 7 6 8 END FILL
```

Next, `ARRAY 2` is placed in the cylinder as shown in Fig. 8.1.64. The data are listed below.

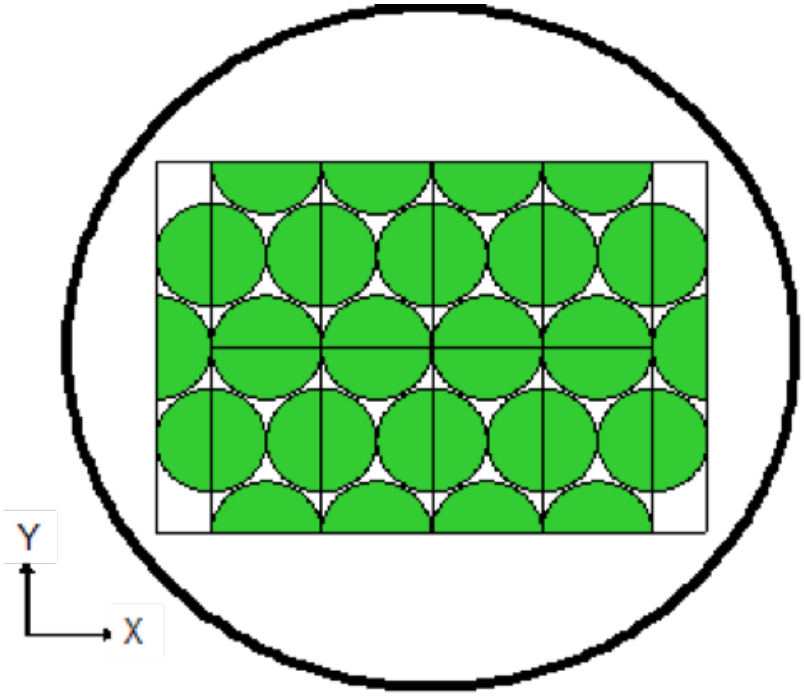


Fig. 8.1.64: Partially completed array in a cylinder.

```
UNIT 9
ARRAY 2 -10.0 -6.92822 -50.0
CYLINDER 0 1 12.4  2P50.0
CYLINDER 2 1 12.65 2P50.0
```

Now UNITs 10 and 11 are described and placed above and below the ARRAY. These UNITs are shown in Fig. 8.1.65. UNIT 10 is described to complete the two central rods at the top of the ARRAY of Fig. 8.1.64. Fig. 8.1.65 and Fig. 8.1.66 illustrate these UNITs. The ZHEMICYL-Y is placed at the top of the UNIT to describe half of the rod at the very top of the ARRAY. The first HOLE 2 completes the left center rod at the top of the ARRAY pictured in Fig. 8.1.64. The second HOLE 2 completes the right center rod at the top of the ARRAY. UNIT 11 is described in similar fashion. It is the mirror image of UNIT 10 and is placed below the ARRAY of Fig. 8.1.64. The resulting configuration is shown in Fig. 8.1.65.

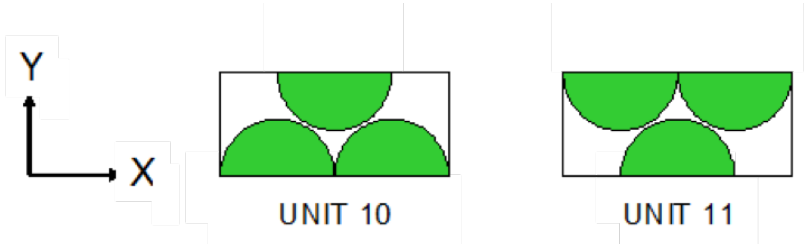


Fig. 8.1.65: Description of UNITs 10 and 11 for Example 2b.

```
UNIT 10
ZHEMICYL-Y 1 1 2.0 2P50.0
CUBOID  0 1 2P4.0 0.0 -3.46411 2P50.0
```

(continues on next page)

(continued from previous page)

```
HOLE 2 -2.0 -3.46411 0.0 2P50.0
HOLE 2  2.0 -3.46411 0.0

UNIT 11
ZHEMICYL+Y 1 1 2.0 2P50.0
CUBOID  0 1 2P4.0 3.46411 0.0 2P50.0
HOLE 1 -2.0 3.46411 0.0
HOLE 1  2.0 3.46411 0.0
```

Now UNITs 10 and 11 are placed above and below the ARRAY of Fig. 8.1.64 to obtain the configuration shown in Fig. 8.1.66.

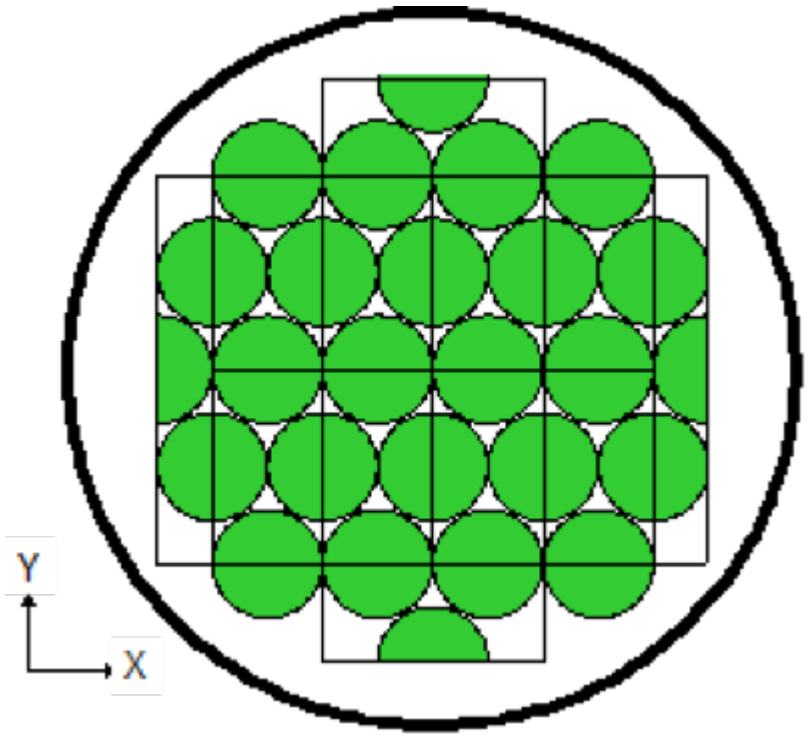


Fig. 8.1.66: Partially completed triangular pitched ARRAY in a cylinder.

```
UNIT 9
ARRAY 2 -10.0 -6.92822 -50.0
CYLINDER 0 1 12.4  2P50.0
HOLE 10 0.0 10.39233 0.0
HOLE 11 0.0 -10.39233 0.0
CYLINDER 2 1 12.65 2P50.0
```

To complete the array, the remaining half cylinders must be entered as HOLEs, as shown below. The first HOLE 1 completes the half rod at the lower left of Fig. 8.1.66. The second HOLE 1 completes the half rod at the lower center, and the third HOLE 1 completes the half rod at the lower right of Fig. 8.1.66. Similarly, the first HOLE 2 completes the half rod at the upper left of Fig. 8.1.66. The second HOLE 2 completes the half rod at the upper center, and the third HOLE 2 completes the half rod at the upper right. HOLE 3 completes the half rod at the left of Fig. 8.1.66, and HOLE 4 completes the half rod at the right. The final geometry configuration is shown in Fig. 8.1.67. UNIT 9 must be specified as the GLOBAL UNIT because it defines the overall configuration.

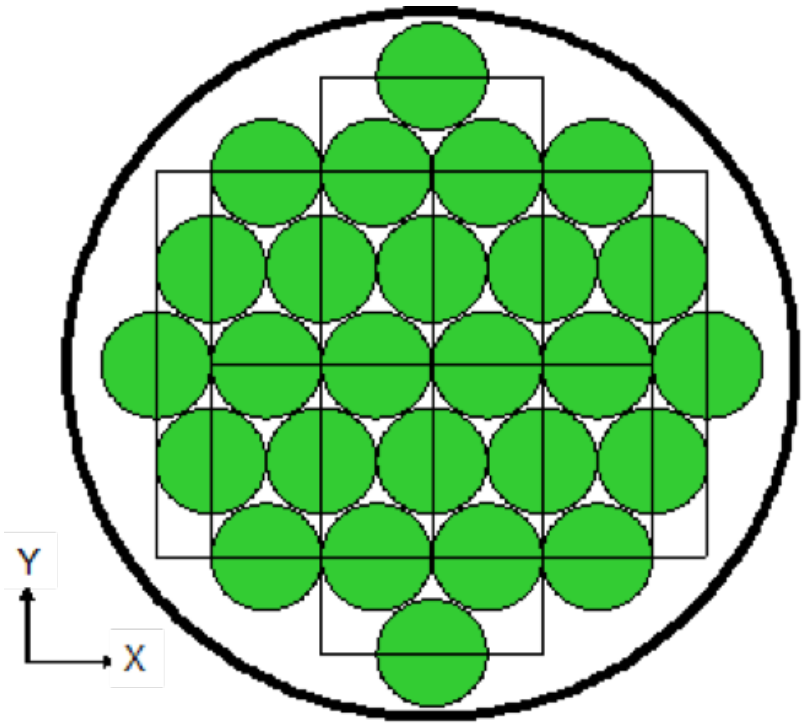


Fig. 8.1.67: Final configuration of triangular pitched ARRAY in a cylinder.

```
GLOBAL  UNIT 9
ARRAY 2 -10.0 -6.92822 -50.0
CYLINDER 0 1 12.4  2P50.0
HOLE 10  0.0 10.39233 0.0
HOLE 11  0.0 -10.39233 0.0
HOLE 1  -6.0  -6.92822 0.0
HOLE 1   0.0 -10.39233 0.0
HOLE 1   6.0  -6.92822 0.0
HOLE 2  -6.0   6.92822 0.0
HOLE 2   0.0  10.39233 0.0
HOLE 2   6.0   6.92822 0.0
HOLE 3 -10.0   0.0     0.0
HOLE 4  10.0   0.0     0.0
CYLINDER 2 1 12.65 2P50.0
```

The geometry data for Example 2b are given below.

```
READ GEOM

UNIT 1
ZHEMICYL-Y 1 1 2.0 50.0 -50.0

UNIT 2
ZHEMICYL+Y 1 1 2.0 50.0 -50.0

UNIT 3
ZHEMICYL-X 1 1 2.0 50.0 -50.0

UNIT 4
ZHEMICYL+X 1 1 2.0 50.0 -50.0

UNIT 5
ZHEMICYL+X 1 1 2.0 50.0 -50.0 ORIGIN -2.0 0.0
```

(continues on next page)

(continued from previous page)

```
CUBOID     0 1 2P2.0 2P3.46411 2P50.0
HOLE 1  0.0  3.46411 0.0
HOLE 2  0.0 -3.46411 0.0
HOLE 3  2.0  0.0     0.0

UNIT 6
ARRAY 1 -8.0 -6.92822 -50.0

UNIT 7
ZHEMICYL+X 1 1 2.0 50.0 -50.0
CUBOID     0 1 2.0 0.0 2P6.92822 2P50.0
HOLE 3 2.0 -3.46411 0.0
HOLE 3 2.0  3.46411 0.0

UNIT 8
ZHEMICYL-X 1 1 2.0 50.0 -50.0
CUBOID     0 1 0.0 -2.0 2P6.92822 2P50.0
HOLE 3 0.0 -3.46411 0.0
HOLE 3 0.0  3.46411 0.0

UNIT 10
YZHEMICYL-Y 1 1 2.0 2P50.0
CUBOID  0 1 2P4.0 0.0 -3.46411 2P50.0
HOLE 2 -2.0 -3.46411 0.0
HOLE 2  2.0 -3.46411 0.0

UNIT 11
YZHEMICYL+Y 1 1 2.0 2P50.0
CUBOID  0 1 2P4.0 3.46411 0.0 2P50.0
HOLE 1 -2.0 3.46411 0.0
HOLE 1  2.0 3.46411 0.0

GLOBAL
UNIT 9
ARRAY 2 -10.0 -6.92822 -50.0
CYLINDER 0 1 12.4  2P50.0
HOLE 10  0.0  10.39233        0.0
HOLE 11  0.0  -10.39233       0.0
HOLE 1  -6.0    -6.92822      0.0
HOLE 1   0.0  -10.39233       0.0
HOLE 1   6.0    -6.92822      0.0
HOLE 2  -6.0     6.92822      0.0
HOLE 2   0.0   10.39233       0.0
HOLE 2   6.0     6.92822      0.0
HOLE 3 -10.0     0.0          0.0
HOLE 4  10.0     0.0          0.0
CYLINDER 2 1 12.65 2P50.0
END GEOM

READ ARRAY
ARA=1 NUX=4 NUY=2 NUZ=1  FILL F5 END FILL
ARA=2 NUX=3 NUY=1 NUZ=1 FILL 7 6 8 END FILL
END ARRAY
```

Dodecahedral pitched arrays

EXAMPLE 22.

Dodecahedral pitched **ARRAY**s can be described in KENO-VI by defining the **UNIT**s that make up the **ARRAY** as dodecahedra and in the **ARRAY** data block setting **TYP**=DODECAHEDRAL. Since the **ARRAY**s are constructed by stacking dodecahedra, care must be taken to ensure the **ARRAY** boundary is completely enclosed within the stacked unit. Below is an example of a dodecahedral **ARRAY** that represents a close packed **ARRAY** of spheres.

The first and second **UNIT**s are the dodecahedra that make up the **ARRAY**. **UNIT** 1 is the fuel sphere stacked in a dodecahedral lattice. **UNIT** 2 is a dummy **UNIT** used to fill in the **ARRAY** so the **ARRAY** boundary is contained within the stacked **UNIT**s. Since the **ARRAY** is not moderated, **UNIT** 2 contains void. Fig. 8.1.68 shows an isometric, cross section view of **UNIT**s 1 and 2.

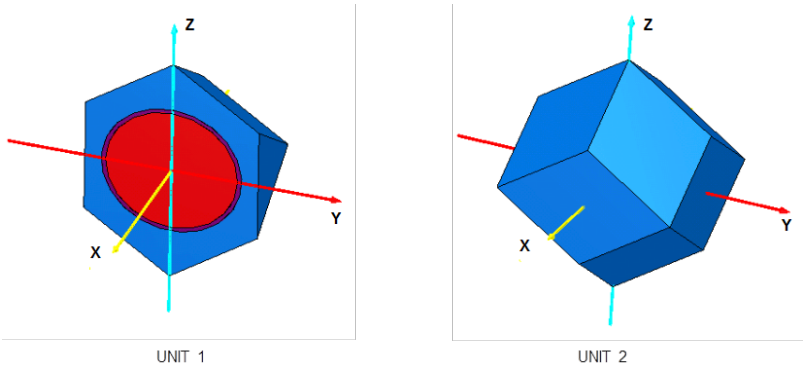


Fig. 8.1.68: Fuel cell and empty cell set up as dodecahedra.

```
UNIT  1
COM='SINGLE CELL FUEL CAN IN DODECAHDRON'
SPHERE       10   8.0
SPHERE       20   8.5
DODECAHEDRON 30  10.5
MEDIA  1  1  10
MEDIA  2  1  20  -10
MEDIA  0  1  30  -20
BOUNDARY     30
UNIT  2
COM='EMPTY CELL'
DODECAHEDRON 10  10.5
MEDIA  0  1  10
BOUNDARY     10
```

UNIT 3 is the **GLOBAL UNIT** that contains the **ARRAY** and **ARRAY BOUNDARY**. The **ARRAY** is an unmoderated triangular pitched assembly of 17 fuel spheres. Dodecahedral **ARRAY**s are specified by using **TYP**= followed by keyword dodecahedral in the array data block. The X and Y coordinates are stacked together as a square pitched **ARRAY** with the pitch equal to twice the dodecahedron radius. In the Z dimension, the odd Z planes (Z = 1, 3, 5, etc.) have the X and Y **UNIT**s begin at the most negative edge of the **ARRAY**, while the even Z planes (Z = 2, 4, 6, etc.) have the X and Y **UNIT**s begin one dodecahedron inscribed sphere radius in the positive direction from the most negative edge of the **ARRAY**. Also, the Z distance between the centers of the **UNIT**s in successive Z planes is the square root of 2.0 times the dodecahedron radius (or the dodecahedron diameter divided by the square root of 2.0). Fig. 8.1.69 and Fig. 8.1.70 show X-Y cross sectional color plots of the assembly at an odd and even Z plane.

```
GLOBAL UNIT 3
COM='17 CLOSE PACKED FUEL SPHERES IN A CYLINDER'
CYLINDER  10  41.0  44.5   0.0
CYLINDER  20  42.0  44.5  -1.0
ARRAY  1  10 PLACE 3 3 1 3*0.0
MEDIA  3  20  -10
BOUNDARY 20

READ ARRAY GBL=1 ARA=1 TYP=DODECAHEDRAL NUX=5 NUY=5 NUZ=5
FILL 25*2
     6*2  2*1  3*2  2*1  12*2
     6*2  3*1  2*2  3*1  2*2  3*1  6*2
     6*2  2*1  3*2  2*1  12*2
     25*2 END FILL  END ARRAY
```

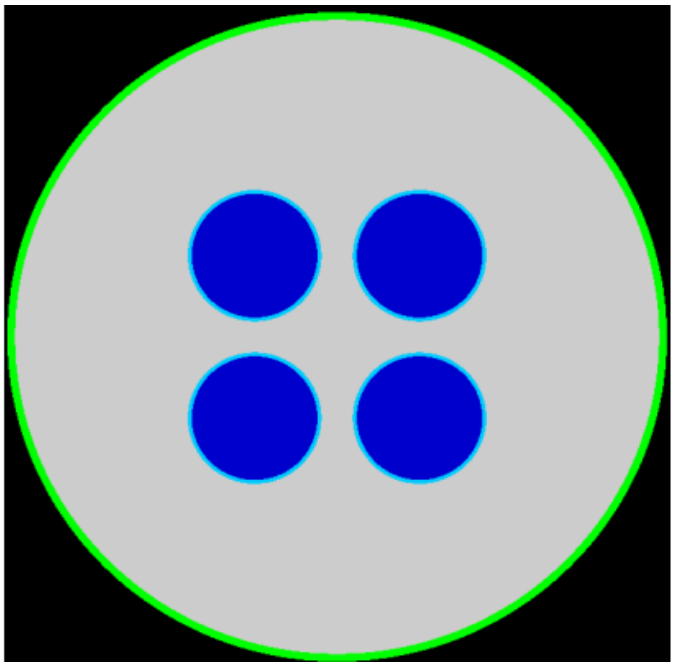


Fig. 8.1.69: X-Y slice of dodecahedral array at even level Z = 2.

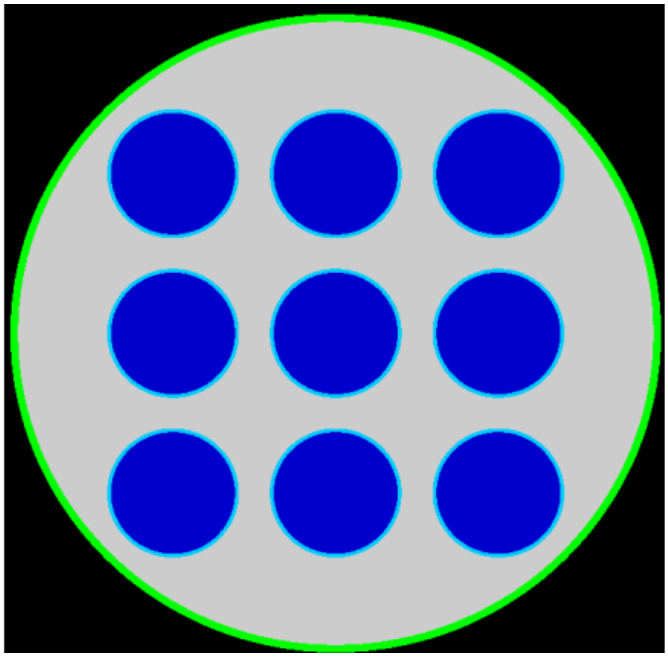


Fig. 8.1.70: X-Y slice of dodecahedral array at odd level Z = 3.

The overall problem description is shown below. The cross section library would be generated in a separate CSAS-MG step.

Data description of Example 22.

```
=KENOVI
CLOSE PACKED DODECAHEDRAL ARRAY 17 FUEL SPHERES IN A CYLINDER
READ PARAMETERS LNG=20000 LIB=4 END PARAMETERS
READ MIXT SCT=2
MIX=1 NCM=8 92235 1.37751E-03  92238 9.92354E-05  8016 3.32049E-02
       9019 2.95349E-03  1001 6.05028E-02
MIX=2 NCM=14 13027 6.02374E-02
MIX=3 NCM=14 13027 6.02374E-02
END MIXT
READ GEOMETRY
UNIT  1
COM='SINGLE CELL FUEL CAN IN DODECAHDRON'
SPHERE       10   8.0
SPHERE       20   8.5
DODECAHEDRON 30  10.5
MEDIA  1  1  10
MEDIA  2  1  20  -10
MEDIA  0  1  30  -20
BOUNDARY     30
UNIT  2
COM='EMPTY CELL'
DODECAHEDRON 10  10.5
MEDIA  0  1  10
BOUNDARY     10
GLOBAL UNIT 3
COM='9 CLOSE PACKED FUEL SPHERES IN A CYLINDER'
CYLINDER  10  41.0  44.5   0.0
CYLINDER  20  42.0  44.5  -1.0
ARRAY  1  10 PLACE 3 3 1 3*0.0
MEDIA  3  1 20  -10
BOUNDARY 20
```

(continues on next page)

(continued from previous page)

```
END GEOMETRY
READ ARRAY GBL=1 ARA=1 TYP=DODECAHEDRAL NUX=5 NUY=5 NUZ=5
FILL 25*2
     6*2  2*1  3*2  2*1  12*2
     6*2  3*1  2*2  3*1  2*2  3*1  6*2
     6*2  2*1  3*2  2*1  12*2
     25*2 END FILL  END ARRAY
READ PLOT
TTL='DODECAHEDRAL ARRAY, 4 SPHERES - Z LEVEL = 2'
XUL=-43.0  YUL=43.0  ZUL=14.85  XLR=43.0  YLR=-43.0  ZLR=14.85
UAX=1  VDN=-1  NAX=640    END  PLT0
TTL='DODECAHEDRAL ARRAY, 9 SPHERES - Z LEVEL = 3'
XUL=-43.0  YUL=43.0  ZUL=29.70  XLR=43.0  YLR=-43.0  ZLR=29.70
UAX=1  VDN=-1  NAX=640    END  PLT1
END PLOT
END DATA
END
```

8.1.4.7 Alternative sample problem mockups

The geometry data for KENO can often be described correctly in several ways. Some alternative geometry descriptions are given here for sample problems C.12 and C.13. (See Sect. 8.1.8.3 of the KENO manual.)

Sample Problem C.12, First Alternative

This mockup maintains the same overall unit dimensions that were used in sample problem C.12. In sample problem C.12, the origin of **UNIT** 1, the solution cylinder, is at the center of the unit; the origin of **UNIT**s 2, 3, 4, and 5, the metal cylinders, are at the center of the cylinders. In this mockup, the unit numbers remain the same and the origin of each unit is at the center of the unit. In each unit the cylinder is offset by specifying the position of its centerline relative to the origin of the **UNIT**.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER  2 1 9.525 8.89 -8.89
CYLINDER  3 1 10.16 9.525 -9.525
CUBOID    0 1 10.875 -10.875 10.875 -10.875 10.24 -10.24
UNIT 2
CYLINDER  1 1 5.748 9.3975 -1.3975 ORIG 4.285 4.285
CUBOID    0 1 10.875 -10.875 10.875 -10.875 10.24 -10.24
UNIT 3
CYLINDER  1 1 5.748 9.3975 -1.3675 ORIG 4.285 -4.285
CUBOID    0 1 10.875 -10.875 10.875 -10.875 10.24 -10.24
UNIT 4
CYLINDER  1 1 5.748 1.3675 -9.3975 ORIG 4.285 4.285
CUBOID    0 1 10.875 -10.875 10.875 -10.875 10.24 -10.24
UNIT 5
CYLINDER  1 1 5.748 1.3675 -9.3975 ORIG 4.285 -4.285
CUBOID    0 1 10.875 -10.875 10.875 -10.875 10.24 -10.24
END GEOM
READ ARRAY  NUX=2 NUY=2 NUZ=2 FILL 2 1 3 1 4 1 5 1 END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER 1  9.525 8.89 -8.89
CYLINDER 2  10.16 9.525 -9.525
CUBOID  3   10.875 -10.875 10.875 -10.875 10.24 -10.24
```

(continues on next page)

(continued from previous page)

```
MEDIA 2 1 1
MEDIA 3 1 2 -1
MEDIA 0 1 3 -2
BOUNDARY  3
UNIT 2
CYLINDER 1  5.748 9.3975 -1.3675 ORIG  X=4.285 Y=4.285
CUBOID  2   10.875 -10.875 10.875 -10.875 10.24 -10.24
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 3
CYLINDER 1  5.748 9.3975 -1.3675 ORIG X=4.285 Y=-4.285
CUBOID  2   10.875 -10.875 10.875 -10.875 10.24 -10.24
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 4
CYLINDER 1  5.748 1.3675 -9.3975 ORIG X=4.285 Y=4.285
CUBOID  2   10.875 -10.875 10.875 -10.875 10.24 -10.24
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 5
CYLINDER 1  5.748 1.3675 -9.3975 ORIG X=4.285 Y=-4.285
CUBOID  2   10.875 -10.875 10.875 -10.875 10.24 -10.24
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
GLOBAL UNIT 6
CUBOID 1 21.75 -21.75 21.75 -21.75 20.48 -20.48
ARRAY 1 PLACE 1 1 1 -10.875 -10.875 -10.24
BOUNDARY  1
END GEOM
READ ARRAY ARA=1  NUX=2 NUY=2 NUZ=2 FILL 2 1 3 1 4 1 5 1 END ARRAY
```

Sample Problem C.12, Second Alternative

In this mockup, the outer boundaries of the system are made as close fitting as possible on all six faces. The origin of each **UNIT** is located at the center of the cylinder. **UNIT**s 1, 3, 5, and 7 contain the metal cylinders. **UNIT**s 2, 4, 6, and 8 contain the solution cylinders.

KENO V.a:

```
READ GEOM
UNIT 1
CYLINDER 1 1 5.748 5.3825 -5.3825
CUBOID   0 1 6.59 -5.748 6.59 -14.445 6.225 -13.54
UNIT 2
CYLINDER  2 1 9.525 8.89 -8.89
CYLINDER  3 1 10.16 9.525 -9.525
CUBOID    0 1 10.16 -10.875 10.875 -10.16 10.24 -9.525
UNIT 3
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID    0 1 6.59 -5.748 14.444 -6.59 6.225 -13.54
UNIT 4
CYLINDER   2 1 9.525 8.89 -8.89
CYLINDER   3 1 10.16 9.525 -9.525
CUBOID     0 1 10.16 -10.875 10.16 -10.875 10.24 -9.525
UNIT 5
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID    0 1 6.59 -5.748 6.59 -14.445 13.54 -6.225
UNIT 6
CYLINDER   2 1 9.525 8.89 -8.89
```

(continues on next page)

(continued from previous page)

```
CYLINDER   3 1 10.16 9.525 -9.525
CUBOID     0 1 10.16 -10.875 10.875 -10.16 9.525 -10.24
UNIT 7
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID    0 1 6.59 -5.748 14.445 -6.59 13.54 -6.225
UNIT 8
CYLINDER   2 1 9.525 8.89 -8.89
CYLINDER   3 1 10.16 9.525 -9.525
CUBOID     0 1 10.16 -10.875 10.16 -10.875 9.525 -10.24
READ ARRAY  NUX=2 NUY=2 NUZ=2 FILL 6I1 8 END FILL END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
CYLINDER 1 5.748 5.3825 -5.3825
CUBOID  2  6.59 -5.748 6.59 -14.445 6.225 -13.54
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 2
CYLINDER 1  9.525 8.89 -8.89
CYLINDER 2  10.16 9.525 -9.525
CUBOID  3   10.16 -10.875 10.875 -10.16 10.24 -9.525
MEDIA 2 1 1
MEDIA 3 1 2 -1
MEDIA 0 1 3 -2
BOUNDARY  3
UNIT 3
CYLINDER  1 5.748 5.3825 -5.3825
CUBOID  2   6.59 -5.748 14.444 -6.59 6.225 -13.54
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 4
CYLINDER 1  9.525 8.89 -8.89
CYLINDER 2  10.16 9.525 -9.525
CUBOID   3   10.16 -10.875 10.16 -10.875 10.24 -9.525
MEDIA 2 1 1
MEDIA 3 1 2 -1
MEDIA 0 1 3 -2
BOUNDARY  3
UNIT 5
CYLINDER 1 5.748 5.3825 -5.3825
CUBOID   2  6.59 -5.748 6.59 -14.445 13.54 -6.225
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 6
CYLINDER 1  9.525 8.89 -8.89
CYLINDER 2  10.16 9.525 -9.525
CUBOID   3   10.16 -10.875 10.875 -10.16 9.525 -10.24
MEDIA 2 1 1
MEDIA 3 1 2 -1
MEDIA 0 1 3 -2
BOUNDARY  3
UNIT 7
CYLINDER 1  5.748 5.3825 -5.3825
CUBOID  2   6.59 -5.748 14.445 -6.59 13.54 -6.225
MEDIA 1 1 1MEDIA 0 1 2 -1
BOUNDARY  2
UNIT 8
CYLINDER 1  9.525 8.89 -8.89
CYLINDER 2  10.16 9.525 -9.525
CUBOID  3    10.16 -10.875 10.16 -10.875 9.525 -10.24
```

(continues on next page)

(continued from previous page)

```
MEDIA 2 1 1
MEDIA 3 1 2 -1
MEDIA 0 1 3 -2
BOUNDARY  3
GLOBAL UNIT 9
CUBOID 1 20.035 -12.748 20.67 -20.67 19.765 -19.765
ARRAY 1 1 PLACE 1 1 1 -7.00 -6.225 -6.225
BOUNDARY  1
END GEOM
READ ARRAY ARA=1  NUX=2 NUY=2 NUZ=2 FILL 6I1 8 END FILL END ARRAY
```

Sample Problem C.13, Alternative

This mockup maintains the same overall **UNIT** dimensions that were used in sample problem C.13, (KENO Manual, Sect. 8.1.8.3). In sample problem C.13, the origin of **UNIT**s 1, 2, and 3 is located at the center of the base of the uranium metal cuboid. In this mockup, the origin of **UNIT**s 1 and 2 is located at the center of the cylinder. In **UNIT** 3, the origin is at the center of the **UNIT**.

KENO V.a:

```
READ GEOM
UNIT 1
CUBOID  1 1 0.2566 -12.4434 6.35 -6.35 3.81 -3.81
CYLINDER  0 1 13.97 3.81 -3.81
CYLINDER  1 1 19.05 3.81 -3.81
CUBOID  0 1 19.05 -19.05 19.05 -19.05 3.81 -3.81
UNIT 2
CUBOID  1 1 12.4434 -0.2566 6.35 -6.35 4.28 -4.28
CYLINDER  0 1 13.97 4.28 -4.28
CYLINDER  1 1 19.05 4.28 -4.28
CUBOID  0 1 19.05 -19.05 19.05 -19.05 4.28 -4.28
UNIT 3
CUBOID  1 1 12.4434 -0.2566 6.35 -6.35 1.308 -1.308
CUBOID  0 1 19.05 -19.05 19.05 -19.05 1.308 -1.308
END GEOM
READ ARRAY  NUX=1 NUY=1 NUZ=3  FILL 1 2 3 END ARRAY
```

KENO-VI:

```
READ GEOM
UNIT 1
CUBOID  1 0.2566 -12.4434 6.35 -6.35 3.81 -3.81
CYLINDER 2  13.97 3.81 -3.81
CYLINDER 3  19.05 3.81 -3.81
CUBOID 4  19.05 -19.05 19.05 -19.05 3.81 -3.81
MEDIA 1 1 1
MEDIA 0 1 2 -1
MEDIA 1 1 3 -2 -1
MEDIA 0 1 4 -3 -2 -1
BOUNDARY  4
UNIT 2
CUBOID  1 12.4434 -0.2566 6.35 -6.35 4.28 -4.28
CYLINDER 2 13.97 4.28 -4.28
CYLINDER 3 19.05 4.28 -4.28
CUBOID 4  19.05 -19.05 19.05 -19.05 4.28 -4.28
MEDIA 1 1 1
MEDIA 0 1 2 -1
MEDIA 1 1 3 -2 -1
MEDIA 0 1 4 -3 -2 -1
BOUNDARY  4
UNIT 3
CUBOID 1 12.4434 -0.2566 6.35 -6.35 1.308 -1.308
```

(continues on next page)

(continued from previous page)

```
CUBOID 2 19.05 -19.05 19.05 -19.05 1.308 -1.308
MEDIA 1 1 1
MEDIA 0 1 2 -1
BOUNDARY  2
GLOBAL UNIT 4
CUBOID 1 19.05 -19.05 19.05 -19.05 6.896  -11.90
ARRAY 1 1 PLACE 1 2 1 0.0 0.0 0.0
BOUNDARY  1
END GEOM
READ ARRAY ARA=1  NUX=1 NUY=1 NUZ=3  FILL 1 2 3 END ARRAY
```

8.1.4.8 Initial starting distributions with Start data

Sect. 8.1.3.8 discusses the input directions for entering different starting distributions in the start data input block. The following start type 0 and start type 6 examples demonstrate how KENO codes process the entered start data parameters.

Example-1:

An 8 cm radius fissile sphere without a start data block is modeled with both KENO-VI and KENO V.a geometries. Start type 0, which is the default starting option, is used, and neutrons are started uniformly in the entire volume defined by the outermost geometry (fissile sphere).

KENO-VI:

```
...
 read geometry
    global unit 1
       sphere 1  8
       media 1 1 1
       boundary 1
 end geometry
end data
end
```

KENO V.a:

```
' input without global unit
...
 read geometry
    unit 1
       sphere 1 1 8
 end geometry
end data
end


' input with global unit
...
 read geometry
    global unit 1
       sphere 1 1 8
 end geometry
end data
end
```

Example-2:

This input models a simple $2 \times 2 \times 2$ array of fissile cylinders with the default starting option, which is start type 0. In this array problem, **XSM**, **YSM**, and **ZSM** are defaulted to the minimum X, Y, and Z coordinates of the global array, and **XSP**, **YSP**, and **ZSP** are defaulted to the maximum coordinates of the global array. A

bounding box is constructed with these values, and neutrons are started at the points uniformly sampled inside the fissile cylinders within this cuboid (For KENO V.a model, the cuboid is defined by +x=27.48, -x=0.0, +y=27.48, -y=0.0, +z= 26.02, -z= 0.0, and for KENO-VI model, it is defined by +x=13.74, -x=-13.74, +y=13.74, -y=-13.74, +z= 13.01, -z=-13.01).

```
KENO-VI:

...
 read geometry
    unit 1
       cylinder 1 5.748 2p5.3825
       cuboid   2 4p6.87 2p6.505
       media 1 1 1
       media 0 1 2 -1
       boundary 2
    global unit 2
       cuboid  10 4p13.74 2p13.01
       array 1 +10 place 1 1 1 2r-6.87 -6.505
       boundary 10
 end geometry
 read array
   gbl=1 ara=1 nux=2 nuy=2 nuz=2 fill f1  end fill
 end array
end data
end

KENO V.a:

' input with a global array
...
 read geometry
     unit 1
        cylinder 1 1 5.748 5.3825 -5.3825
        cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
 end geometry
 read array
    gbl=1 ara=1 nux=2 nuy=2 nuz=2 fill f1  end fill
 end array
end data
end
```

Example-3:

Neutrons are started uniformly throughout the fissile materials in a user-defined box that is inside the outermost geometry (+x=3.0 -x=-3.0 +y= 1.0 -y=-1.0 +z= 2.0 -z=-2.0).

```
KENO-VI:

...
 read geometry
    global unit 1
       sphere 1  8
       media 1 1 1
       boundary 1
 end geometry

 READ START
    NST=0
    XSP=3.0 XSM=-3.0 YSP=1.0 YSM=-1.0 ZSP=2.0 ZSM=-2.0
 END START

end data
end
```

Example-4:

This test case is another example of the start type 0 specification with a user-defined cuboid, but whole cuboid is not inside the outermost geometry. KENO V.a immediately terminates the execution with some error messages since some starting points are sampled outside the outermost geometry.

Unlike KENO V.a, KENO-VI continues the sampling process by discarding these points, which are outside the global geometry, until all starting points have been successfully sampled throughout the fissile regions inside the global unit.

```
KENO-VI:
...
 read geometry
    global unit 1
       sphere 1  8
       media 1 1 1
       boundary 1
 end geometry

 READ START
    NST=0
    XSP=10.0 XSM=-10.0 YSP=10.0 YSM=-10.0 ZSP=10.0 ZSM=-10.0
 END START

end data
end
```

Note: Note that the KENO-VI sampling process for such a case will be terminated if

- the first starting point has not been sampled in *tbtch* minutes. *tbtch* is the time per generation in minutes, defaulted to 10 minutes. User can control this with parameter **TBA=** in parameter input; see Sect. 8.1.3.3 for details, or
- all points have not been successfully sampled inside the global units in total *tbtch* $\times$ *9.5* minutes.

Example-5:

This test input demonstrates how KENO-VI performs source sampling with the default starting type (start type 0) if the boundary definition vector of the global unit has the multiple body labels. For such cases, KENO-VI starts neutrons at the points that are uniformly sampled in the first body entered in the boundary definition vector of the global unit if and only if none of translation and transformation operations are performed on this body. For this KENO-VI specific example, all starting points are sampled throughout the fissile materials inside the *cylinder 3*.

```
KENO-VI:

...
 read geometry
    global unit 1
       sphere 1 8.741
       sphere 2 10.0
       cylinder 3 8.0 2p8.0
       media 1 1 1
       media 2 1 2 3 -1
       BOUNDARY  3 2
 end geometry

end data
end
```

Example-6:

Start type 0 specification with one or more missing optional parameters; their default values (0.0) are used to construct a cuboid, and neutrons are started uniformly throughout the fissile materials in this box (+x=3.0 -x=0.0 +y= 1.0 -y=0.0 +z=0.0 -z=-2.0).

```
KENO-VI:

...
 read geometry
    global unit 1
       sphere 1  8
       media 1 1 1
       boundary 1
 end geometry

 READ START
    NST=0
    XSP=3.0 YSP=1.0 ZSM=-2.0
 END START

end data
end
```

Note: In start data input, user-defined cuboid shape is considered as a deformed cuboid shape if the minimum and maximum bounds in any dimension are equal entries (**XSP = XSM** or/and **YSP = YSM** or/and **ZSP = ZSM**). Deformed shape specification is a legitimate input specification across SCALE sequences (i.e., a box is deformed into a plane if minimum and maximum bounds in one dimension are the same, e.g., +x = -x). In such a case, user-defined entries for **XSM**, **XSP**, **YSM**, **YSP**, **ZSM**, and **ZSP** are ignored, and source sampling is performed throughout the volume defined by the outermost geometry card.

The following two examples with start type 0 demonstrate code behavior for the deformed shapes for start types 0, 1, 2, and 7.

Example-7:

Box defined with missing **ZSM**, and **ZSP** is considered as a deformed cuboid since these parameters are identical after defaulting their values to 0.0. Instead of this deformed cuboid box, neutrons are started uniformly throughout the fissile materials in the global unit.

```
KENO-VI:

...
 read geometry
    global unit 1
       sphere 1  8
       media 1 1 1
       boundary 1
 end geometry

 READ START
    NST=0
    XSP=3.0 XSM=-3.0 YSP=1.0 YSM=-1.0
 END START

end data
end
```

Example-8:

A box defined with identical **XSM** and **XSP** entries is considered as a deformed cuboid since these parameters are identical after defaulting their values to 0.0. Instead of this deformed cuboid box, neutrons are started uniformly throughout the fissile materials in the global unit.

```
KENO-VI:

...
 read geometry
    global unit 1
       sphere 1  8
       media 1 1 1
       boundary 1
 end geometry

 READ START
    NST=0
    XSP=3.0 XSM=3.0 YSP=1.1 YSM=-2.1 ZSP=4.0 ZSM=-4.1
 END START

end data
end
```

In the start type 6 capability, the selection process for the initial fission source points depends on the values of the last LNU value and number per generation (parameter NPG= in the parameter input Sect. 8.1.3.3). The following rules are applied when selecting the starting points:

- Start NPG initial fission neutrons at first-NPG starting points defined by start type 6 data if **NPG** < **LNU**. Remaining starting points beyond NPG will be discarded.
- Start NPG initial fission neutrons at LNU starting points defined by start type 6 data if **NPG** = **LNU**.
- Start LNU initial fission neutrons at the starting points defined by start type 6 data, then randomly select the remaining fission source points (NPG-LNU) from these starting points if **NPG** > **LNU**.

Warning: It is the user's responsibility to enter all arbitrary starting points consistent with the geometry specified. Both KENO V.a and KENO-VI terminate the execution with several error messages if any user-specified starting point entered is outside the outermost geometry.

Example-9:

In this example, 120 starting points have been defined by start type 6 data, and NPG=100 starting points are selected from the defined start type 6 data, 25 neutrons are started at the point (2.0, 3.0, 0.0), 20 neutrons are started at the point (3.0, -2.0, -6.0), and the remaining 55 neutrons are started at (3.0, 5.2, 1.0). All selected points are relative to the global coordinate system. Missing **TFX** entry in the last start type 6 data set is set to the last updated **TFX** value which is 3.0.

```
KENO-VI:

...

 READ PARAMETER
    NPG=100
 END PARAMETER

 read geometry
    global unit 1
       sphere 1  8
```

(continues on next page)

(continued from previous page)

```
      media 1 1 1
      boundary 1
 end geometry

 READ START
    NST=6
    TFX=2.0 TFY= 3.0 TFZ=0.0  LNU=25
    TFX=3.0 TFY=-2.0 TFZ=-6.0 LNU=45
            TFY=5.2  TFZ=1.0  LNU=120
 END START

end data
end
```

Example-10:

In this example, only 45 starting points have been defined by start type 6 data. The first 25 neutrons are started at the point (2.0, 0.0, 0.0), and 20 neutrons are started at the point (2.0, 3.0, -6.0). The remaining 55 neutrons are started at the points randomly selected from these starting points. All selected points are relative to the global coordinate system. Missing **TFY** and **TFZ** entries are defaulted to 0.0 in the first start type 6 data set. The missing **TFX** entry in the last start type 6 data set is set to the last updated **TFX** value, which is 2.0 (from the previous start type 6 data set).

```
KENO-VI:

...

 READ PARAMETER
    NPG=100
 END PARAMETER

 read geometry
    global unit 1
       sphere 1  8
       media 1 1 1
       boundary 1
 end geometry

 READ START
    NST=6
    TFX=2.0          LNU=25
    TFY=3.0 TFZ=-6.0 LNU=45
 END START

end data
end
```

Example-11:

In this example, 35 neutrons are started at the point (0.0, 0.0, 0.0) relative to the global array element (2,1,1), and the remaining 65 neutrons are started at the point (2.0, 1.0, 3.0) relative to the global array element (1,2,2). The missing **TFX**, **TFY**, and **TFZ** entries in each data set are defaulted to 0.0.

```
KENO V.a:

...

 READ PARAMETER
    NPG=100
 END PARAMETER
```

(continues on next page)

(continued from previous page)

```
 read geometry
    unit 1
       cylinder 1 1 5.748 5.3825 -5.3825
       cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
 end geometry

 read array
    gbl=1 ara=1 nux=2 nuy=2 nuz=2 fill f1  end fill
 end array

 READ START
    NST=6
    NXS=2 NYS=1 NZS=1 LNU=35
        TFX=2.0 TFY=1.0 TFZ=3.0 NXS=1 NYS=2 NZS=2 LNU=100
 END START

end data
end
```

Start type 6 is capable of reading starting points from an ASCII start data file, which could be created by writing starting points from a previous calculation, defined by RDU in a start type 6 data set. See Sect. 8.1.3.8.1 for the details about a typical ASCII start data file currently supported.

Reading starting data from multiple ASCII start data files is also supported. Similarly, single or multiple start type 6 date sets with RDU specification can be used together with other start type 6 data sets discussed above. The following examples demonstrate this capability.

Example-12:

In this example, all starting points stored in the ASCII start data file *sample.src* are read, and **NPG** neutrons are started at these points. If the number of starting points (last LNU read from the file) is less than **NPG**, **LNU** number of neutrons are started at the points read, and the remaining neutrons (NPG-LNU) are started at the points randomly sampled from the starting points already read. The starting points beyond **NPG** are discarded if **NPG** is less than the number of starting points read.

```
...
READ START
   NST=6
   RDU=sample.src
END START
```

Example-13:

The first 32 starting points stored in the ASCII start data file *sample.src* are read, and **NPG** neutrons are started at these points. If **NPG** > **LNU** = 32, then 32 neutrons are started at these starting points, and the remaining neutrons are started at the points randomly sampled from these 32 starting points. The starting points beyond **NPG** is discarded if **NPG** < 32.

```
...
READ START
   NST=6
   RDU=sample.src LNU=32
END START
```

Note: Reading starting data from multiple ASCII start data files is also supported. Similarly, single or multiple start type 6 date sets with the RDU specification can be used together with other start type 6 data

sets discussed above. The following examples demonstrates this capability.

Example-14:

A total of 30 starting points are read both from the specified ASCII start data files and from the start type 6 data set defined with **TFX**, **TFY**, and **TFZ**. Neutrons are started at these points. If there is a shortage of starting points (NPG >> last LNU), then the required number of starting points is randomly sampled from these 30 starting points.

```
...
READ START
   NST=6
   RDU=sample1.src LNU=10
   RDU=sample2.src LNU=25
       TFX=1.0 TFY=3.0 TFZ=4.0 LNU=30
END START
```

8.1.4.9 Biasing or weighting data for multigroup mode

Sect. 8.1.4.6.3 discusses the basis of weighting or biasing. The use of biasing data in reflected problems has been illustrated in Examples 9, 10, and 11 of Sect. 8.1.4.6. Sect. 8.1.3.7 discusses the input directions for entering biasing data.

Every **shape** card in KENO V.a, or **MEDIA** card in KENO-VI requires a bias ID to associate that geometry region with a biasing or weighting function. A biasing or weighting function is a set of energy-dependent values of the average weight that are applicable in a given region. The default function for all bias IDs is constant through all energy groups and is defined to be the default value of weight average which can be specified in the parameter data. A bias ID can be associated with a biasing function (other than default) by specifying it in the biasing input data. This function can be chosen from the weighting library, or it can be entered from records. Table 8.1.25 lists the materials and energy group structures for biasing functions available from the weighting library.

> **Caution:** In general, the use of biasing should be restricted to external reflectors unless the user has generated correct biasing functions for other applications. Improper use of biasing functions can result in erroneous answers without giving any indication that they are invalid. Caution should be exercised in the generation and use of biasing functions.

Biasing functions are most applicable to thick external reflectors. Their use can significantly reduce the amount of computer time required to obtain answers in KENO. If the user wishes to use a biasing function for a concrete reflector, for example, the following steps must be included in preparing the input data:

1. The geometry region data must define the shape and dimensions of the reflector using the mixture ID for concrete and a sequence of bias IDs that associate the geometry region with the appropriate interval of the concrete weighting function. CAUTION: THE THICKNESS AND SEQUENTIAL LOCATION OF EACH REGION USING BIASING FUNCTIONS MUST MATCH OR VERY NEARLY MATCH THE INCREMENT THICKNESS AND ORDER OF THE WEIGHTING DATA. NO CHECK IS MADE ON THE REQUIREMENT. IT IS THE USER'S RESPONSIBILITY TO ENSURE CONSISTENCY.

2. Biasing data must be entered. This must include the material ID for the reflector material (from Table 8.1.25 or as specified on records) and a beginning and ending bias ID. The beginning bias ID is used to

select the first set of energy-dependent average weights, and the subsequent sets of energy-dependent average weights are assigned consecutive IDs until the ending bias ID is reached.

Small deviations in reflector region thickness are allowed, such as using three generated regions with a thickness per region of 5.08 cm to generate a 15.24 cm thick reflector of concrete, or using five generated regions with a thickness per region of 3.048 cm to generate a 15.24 cm thick reflector of water. See Table 8.1.25 for a list of the increment thicknesses for the materials in the weighting library. It is acceptable for the thickness of the last reflector region to be significantly different than the increment thickness. For example, a reflector record specifying five generated regions with a thickness per region of 3.0 cm could be followed by a reflector record specifying one region with a thickness per region of 0.24 cm. Assuming that material 2 is water and a 15.24 cm thick cuboidal reflector of water is desired, the required reflector description and biasing data could be entered as follows:

KENO V.a:

```
REFLECTOR 2 2 6*3.0  5
REFLECTOR 2 7 6*0.24 1
READ BIAS ID=500 2 7 END BIAS
```

KENO-VI:

```
GLOBAL UNIT 1
CUBOID 1 6P10.0
CUBOID 2 6P13.0
CUBOID 3 6P16.0
CUBOID 4 6P19.0
CUBOID 5 6P22.0
CUBOID 6 6P25.0
CUBOID 7 6P25.24
MEDIA 1 1 1
MEDIA 2 2 2 -1
MEDIA 2 3 3 -2
MEDIA 2 4 4 -3
MEDIA 2 5 5 -4
MEDIA 2 6 6 -5
MEDIA 2 7 7 -6
BOUNDARY  7
READ BIAS ID=500 2 7 END BIAS
```

The same 15.24 cm thick reflector can be described by including the extra 0.24 cm in the last region as shown below:

KENO V.a:

```
REFLECTOR 2 2 6*3.0  4
REFLECTOR 2 6 6*3.24 1
READ BIAS  ID=500 2 6 END BIAS
```

KENO-VI:

```
GLOBAL UNIT 1
CUBOID 1 6P10.0
CUBOID 2 6P13.0
CUBOID 3 6P16.0
CUBOID 4 6P19.0
CUBOID 5 6P22.0
CUBOID 6 6P25.24
MEDIA 1 1 1
MEDIA 2 2 2 -1
```

(continues on next page)

(continued from previous page)

```
MEDIA 2 3 3 -2
MEDIA 2 4 4 -3
MEDIA 2 5 5 -4
MEDIA 2 6 6 -5
BOUNDARY  6
READ BIAS ID=500 2 6 END BIAS
```

Here the weighting functions associated with bias IDs 2, 3, 4, and 5 each have a thickness of 3.0 cm, corresponding exactly to the increment thickness for water in Table 8.1.25 Bias ID 6 is used for the last generated region which is 3.24 cm thick.

The following examples illustrate the use of biasing data. Suppose the user wishes to use the weighting function for water from Table 8.1.25 for bias IDs 2 through 6. The biasing input data would then be:

```
READ BIAS  ID=500 2 6 END BIAS
```

The energy-dependent values of weight average for the first 3 cm interval of water will be used for weighting the geometry regions that specify a bias ID of 2. The energy-dependent values of weight average for the second 3 cm interval of water will be used for geometry regions that specify a bias ID of 3, etc. Thus, the energy-dependent values of weight average for the fifth 3 cm interval of water will be used for geometry regions that specify a bias ID of 6. Geometry regions that use bias IDs other than 2, 3, 4, 5, and 6 will use the default value of weight average that is constant for all energies as a biasing function.

Several sets of biasing data can be entered at once. Assume the user wishes to use the weighting function for concrete from Table 8.1.25 for bias IDs 2 through 4 and the weighting function for water for bias IDs 5 through 7. The appropriate input data block is the following:

```
READ BIAS  ID=301 2 4 ID=500 5 7 END BIAS
```

The energy-dependent values of weight average for the first 5 cm interval of concrete will be used for the geometry regions that specify a bias ID of 2, the energy-dependent values of weight average for the second 5 cm interval of concrete will be used for the geometry regions that specify a bias ID of 3, and the energy-dependent values of weight average for the third 5 cm interval of concrete will be used for the geometry regions that specify a bias ID of 4. The energy-dependent values of weight average for the first 3 cm interval of water will be used for geometry regions that specify a bias ID of 5, the values for the second 3 cm interval of water will be used for geometry regions that specify a bias ID of 6, and the values for the third 3 cm interval of water will be used for geometry regions that specify a bias ID of 7. The default value of weight average will be used for all bias IDs outside the range of 2–7.

If the biasing data block defines the same bias ID more than once, the value that is entered last supersedes previous entries. Assuming that the following data block is entered,

```
READ BIAS  ID=400 2 7 ID=500 5 7 END BIAS
```

the data for paraffin (ID=400) will be used for bias IDs 2, 3, and 4, and the data for water (ID=500) will be used for bias IDs 5, 6, and 7.

EXAMPLE 1. USE OF BIASING DATA

It is assumed that a 5 cm radius sphere of material 2 is reflected by a 20 cm thickness of material 1 (concrete). The concrete reflector is spherical and close fitting upon the sphere of material 2. The mixing table must specify material 1 and material 2. Material 1 must be defined as concrete. The geometry and biasing data should be entered as follows:

KENO V.a:

```
READ GEOM
SPHERE 2 1 5.0
REPLICATE 1 2 5.0 4
END GEOM
READ BIAS ID=301 2 5 END BIAS
```

KENO-VI:

```
READ GEOM
GLOBAL UNIT 1
SPHERE 1 5.0
SPHERE 2 10.0
SPHERE 3 15.0
SPHERE 4 20.0
SPHERE 5 25.0
MEDIA 2 1 1
MEDIA 1 2 2 -1
MEDIA 1 3 3 -2
MEDIA 1 4 4 -3
MEDIA 1 5 5 -4
BOUNDARY  5
END GEOM
READ BIAS ID=301 2 5 END BIAS
```

The bias ID for the first generated region is 2, the second is 3, the third is 4, and the fourth is 5. The biasing data block specifies that the biasing function for material ID 301 (concrete) will be used from the weighting library. The bias ID to which the energy-dependent weighting function for the first 5.0 cm interval of concrete is applied is 2; the energy-dependent weighting function for the fourth 5 cm interval of concrete is applied to the fourth generated geometry region. This generated region has a bias ID of 5.

In KENO V.a, Example 1 can also be described without using a reflector record as shown below. The records that are generated by the reflector record in the previous set of data are identical to the last four spheres in this mockup.

EXAMPLE 1. Use of biasing without a reflector record.

KENO V.a:

```
READ GEOM
SPHERE 2 1 5.0
SPHERE 1 2 10.0
SPHERE 1 3 15.0
SPHERE 1 4 20.0
SPHERE 1 5 25.0
END GEOM
READ BIAS  ID=301 2 5 END BIAS
```

8.1.4.10 Color plots

Plots are generated only if a plot data block has been entered for the problem and **PLT**=NO has not been entered in the parameter data or the plot data. See Sect. 8.1.3.11 for a description of plot data. When a plot is to be made, the user MUST correctly specify the upper left-hand corner of the plot with respect to the origin of the plot. The origin of a plot is defined as the origin of the **GLOBAL UNIT**.

Plots can represent mixture numbers, unit numbers, or bias ID numbers. A title can be entered for each plot. If plot titles are omitted, the title of the KENO case will be printed for each plot title until a plot title is entered. If a plot title is entered and a subsequent plot title is omitted, the last plot title prior to the omitted one will be used for the omitted one.

The upper left and lower right coordinates define the area (i.e., the slice and its location) for which the plot is to be made. The direction cosines across the plot and the direction cosines down the plot define the direction of the vector across the plot and the vector down the plot with respect to the geometry coordinate system. One of the simplest ways of generating a plot is to specify the desired coordinates of the upper left and lower right corners of the plot. Then one must determine which plot axis is to be across the plot and which is to be down. The sign of the direction cosine should be consistent with the direction of that component when moving from the upper left to lower right corner. For example, to draw a plot of an X-Z slice at Y = 5.0 with X across the plot and Z down the plot for a system whose X coordinates ranges from 0.0 to 10.0 and whose Z coordinates range from 0.0 to 20.0, the upper left coordinate could be **XUL**=0.0 **YUL**=5.0 **ZUL**=20.0 and the lower right coordinates could be **XLR**=10.0 **YLR**=5.0 **ZLR**=0.0. Since X is to be plotted across the plot with X = 0.0 at the left and X = 10.0 at the right, only the X component of the direction cosines across the plot need be entered. It should be positive because going from 0.0 to 10.0 is moving in the positive direction. Thus, **UAX**=1.0 would be entered for the direction cosines across the plot. **VAX** and **WAX** could be omitted. Z is to be plotted down the plot with Z = 20.0 at the top and Z = 0.0 at the bottom. Therefore, only the Z component of the direction cosines down the plot needs to be defined. It should be negative because moving from 20.0 to 0.0 is moving in the negative direction. Thus, **WDN** = -1.0 would be entered for the direction cosines down the plot. **UDN** and **VDN** could be omitted. The sign of the direction cosines should be consistent with the coordinates of the upper left and lower right corners in order to get a plot.

It is not necessary that the plot be made for a slice orthogonal to one of the axes. Plots can be made of slices cut at any desired angle, but the user should exercise caution and be well aware of the distortion of shapes that can be introduced. (Nonorthogonal slices through cylinders plot as ellipses.)

The user can specify the horizontal and vertical spacing between points on the plot. It is usually advisable to enter one or the other. Entering both can cause distortion of the plot. **DLX**= is used to specify the horizontal spacing between points and **DLD**= is used to specify the vertical spacing between points. When only one of them is specified, the code calculates the correct value of the other so the plot will not be distorted. **DLX** or **DLD** can be specified by the user to be small enough to show the desirable detail in the plot. The plot is generated by starting at the upper left corner of the plot and generating a point every **DLX** across the plot; then moving down **DLD** and repeating the generation of the points across the plot.

NAX specifies the number of intervals (pixels) that will be printed across the plot. **NDN** specifies the number of intervals (pixels) that will be printed down the plot. If both **NAX** and **NDN** are entered, the plot may be distorted. If one of them is entered, the value of the other will be calculated so the plot will not be distorted.

When a plot is being made, the first pixel represents the coordinates of the upper left corner. The value of **DELV** is added to the coordinate that is to be printed across the plot, and the next pixel is printed. **DELV** is added to that value to determine the location of the next pixel, that is, a point is determined every **DELV** across the plot and a pixel is printed for each point. When a line has been completed, a new line is begun **DELU** from the first line. This procedure is repeated until the plot is complete.

EXAMPLE 1. SINGLE UNIT WITH CENTERED ORIGIN

Consider two concentric cylinders in a cuboid. The inner cylinder is 5.2 cm in diameter. The outer cylinder has an inside diameter of 7.2 cm and an outside diameter of 7.6 cm. Both cylinders are 30 cm high. They are contained in a tight-fitting box with a wall thickness of 0.5 cm and top and bottom thickness of 1.0 cm. The inner cylinder is composed of mixture 1, the outer cylinder is made of mixture 4, and the box is made of mixture 2. The problem can be described with its origin at the center of the inner cylinder. The problem description for this arrangement is shown below:

KENO V.a:

```
=KENOVA
SINGLE UNIT CONCENTRIC CYLINDERS IN CUBOID WITH ORIGIN AT CENTER
READ PARAM  RUN=NO LIB=41 END PARAM
READ MIXT  SCT=1
MIX=1 92500  4.7048e-2
MIX=2 200 1.0
MIX=3 502 0.1
MIX=4 200 1.0
END MIXT
READ GEOM
UNIT 1
CYLINDER 1 1 2.6 2P15.0
CYLINDER 0 1 3.6 2P15.0
CYLINDER 4 1 3.8 2P15.0
CUBOID 0 1 4P3.8 2P15.0
CUBOID 2 1 4P4.3 2P16.0
END GEOM

READ PLOT
TTL='X-Y SLICE AT Z MIDPOINT. SINGLE UNIT CONCENTRIC CYLS'
XUL=-4.6 YUL=4.6   ZUL=0.0  XLR=4.6  YLR=-4.6   ZLR=0.0
UAX=1.0   VDN=-1.0   NAX=640   END
PIC=UNIT   END
END PLOT
END DATA
END
```

KENO-VI:

```
=KENOVI
SINGLE UNIT CONCENTRIC CYLINDERS IN CUBOID WITH ORIGIN AT CENTER
READ PARAM  RUN=NO LIB=41 TME=0.5 END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 502 0.1
MIX=4 200 1.0
END MIXT
READ GEOM
GLOBAL UNIT 1
CYLINDER 1 2.6 2P15.0
CYLINDER 2 3.6 2P15.0
CYLINDER 3 3.8 2P15.0
CUBOID 4 4P3.8 2P15.0
CUBOID 5 4P4.3 2P16.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
MEDIA 4 1 3 -2
MEDIA 0 1 4 -3
MEDIA 2 1 5 -4
BOUNDARY  5
END GEOM


READ PLOT
TTL='X-Y SLICE AT Z MIDPOINT. SINGLE UNIT CONCENTRIC CYLS.'
XUL=-4.6 YUL=4.6  ZUL=0.0  XLR=4.6  YLR=-4.6  ZLR=0.0
UAX=1.0  VDN=-1.0  NAX=640  END
PIC=UNIT   END
END PLOT
END DATA
END
```

The plot data blocks included above are set up to draw a mixture map of an X-Y slice taken at the half height (Z=0.0) and a unit map for the same slice. In the above examples, the geometry dimensions extend from X = -4.3 to X = 4.3, from Y = -4.3, to Y = 4.3, and from Z = -16.0 to Z = 16.0. An X-Y slice is be printed at the half height (Z = 0.0). The desired plot data sets the upper left-hand corner of the plot to be X = -4.6 and Y = 4.6. The lower right-hand corner of the plot is specified as X = 4.6 and Y = -4.6. These data are entered

by specifying the upper left-hand corner as **XUL**= -4.6 **YUL**=4.6 **ZUL**=0.0 and the lower right-hand corner as **XLR**=4.6 **YLR**= -4.6 **ZLR**=0.0. It is desired to print X across the plot and Y down the plot. Therefore, the X direction cosine is specified across the plot, in the direction from X = -4.6 to X = 4.6 as **UAX**=1.0. The Y direction cosine is specified down the plot, from Y = 4.6 to Y = -4.6 as **VDN**= -1.0.

A black border will be printed for points outside the range of the problem geometry description. By setting the plot dimension slightly larger than the geometry dimension, a black border will be printed around the specified plot. This verifies that the outer boundaries of the geometry are contained within the plot dimensions. NAX is the number of pixels across for a color plot. An initial recommended range for NAX is between 600 and 800 pixels. The resulting plots are shown in Fig. 8.1.71 and Fig. 8.1.72. The associated data for both plots are shown in Example 8.1.19 and Example 8.1.20.

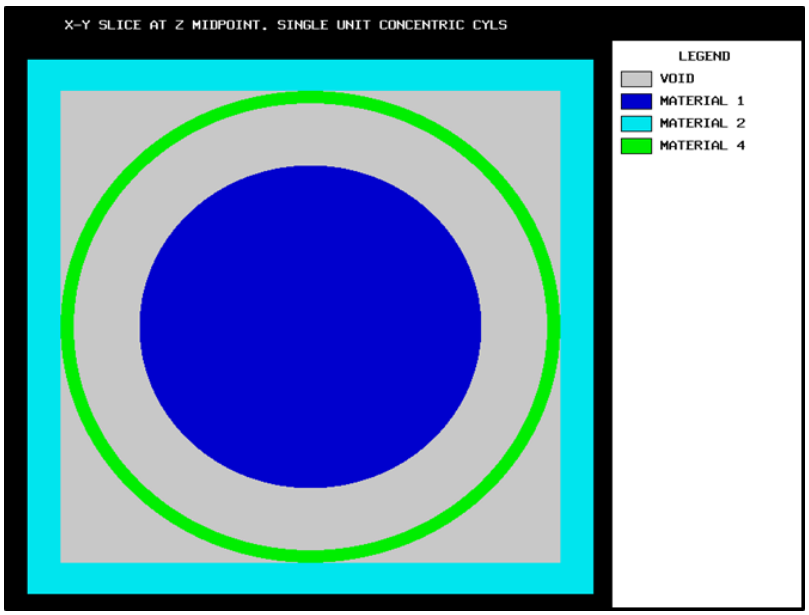


Fig. 8.1.71: Mixture color plot for single unit with centered origin.

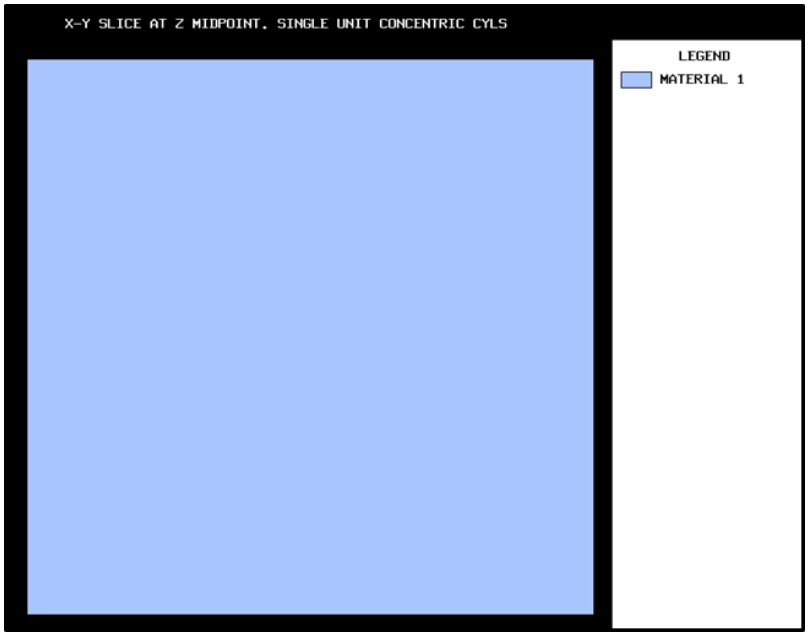


Fig. 8.1.72: Unit map color plot for single unit with centered origin.

Example 8.1.19: Associated data for single unit mixture map.

```
 mixture map

 mixture  0 1 2 4
  symbol    1 2 3
       upper  left          lower right
       coordinates          coordinates
 x     -4.6000e+00           4.6000e+00
 y      4.6000e+00          -4.6000e+00
 z      0.0000e+00           0.0000e+00

           u axis      v axis
           (down)      (across)
 x         0.00000      1.00000
 y        -1.00000      0.00000
 z         0.00000      0.00000

 nu=  640   nv=  640     delu= 1.4375e-02     delv= 1.4375e-02     lpi=  10.000
```

Example 8.1.20: Associated data for single unit map.

```
 x-y slice at z midpoint. single unit concentric cyls

unit map

    unit  1
  symbol  1
       upper  left          lower right
       coordinates          coordinates
 x     -4.6000e+00           4.6000e+00
 y      4.6000e+00          -4.6000e+00
 z      0.0000e+00           0.0000e+00
           u axis      v axis
```

(continues on next page)

(continued from previous page)

```
           (down)      (across)
 x         0.00000      1.00000
 y        -1.00000      0.00000
 z         0.00000      0.00000
 nu=  640   nv=  640     delu= 1.4375e-02     delv= 1.4375e-02     lpi=  10.000
```

EXAMPLE 2. SINGLE UNIT WITH OFFSET ORIGIN.

The physical problem is the same as that described in Example 1: two concentric cylinders in a cuboid. The dimensions are exactly the same, and the difference is in the choice of the origin. In this geometry description, the origin was specified as the most negative point of the unit. Thus, the cylinders must have an origin specified to center them in the cuboid, and the cuboid extends from 0.0 to 8.6 in X and Y and from 0.0 to 32 in Z as shown in the problem description below.

KENO V.a:

```
=KENO5
SINGLE UNIT CONCENTRIC CYLINDERS IN CUBOID WITH ORIGIN AT CORNER
READ PARAM  RUN=NO LIB=41 END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 502 0.1
MIX=4 200 1.0
END MIXT
READ GEOM
UNIT 1
CYLINDER 1 1 2.6 31.0 1.0 ORIGIN 4.3 4.3
CYLINDER 0 1 3.6 31.0 1.0 ORIGIN 4.3 4.3
CYLINDER 4 1 3.8 31.0 1.0 ORIGIN 4.3 4.3
CUBOID 0 1 8.1 0.5 8.1 0.5 31.0 1.0
CUBOID 2 1 8.6 0.0 8.6 0.0 32.0 0.0
END GEOM
READ PLOT
TTL='X-Y SLICE AT Z MIDPOINT. SINGLE UNIT CONCENTRIC CYLS.'
XUL=-0.3 YUL=8.9 ZUL=16.0 XLR=8.9 YLR=-0.3 ZLR=16.0
UAX=1.0 VDN=-1.0 NAX=640 END
PIC=UNIT  END
END PLOT
END DATA
END
```

KENO-VI:

```
=KENOVI
SINGLE UNIT CONCENTRIC CYLINDERS IN CUBOID WITH ORIGIN AT CORNER
READ PARAM  RUN=NO LIB=41 TME=0.5 END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 502 0.1
MIX=4 200 1.0
END MIXT
READ GEOM
GLOBAL UNIT 1
CYLINDER 1 2.6 31.0 1.0 ORIGIN X=4.3 Y=4.3
CYLINDER 2 3.6 31.0 1.0 ORIGIN X=4.3 Y=4.3
CYLINDER 3 3.8 31.0 1.0 ORIGIN X=4.3 Y=4.3
CUBOID  4 8.1 0.5 8.1 0.5 31.0 1.0
CUBOID 5  8.6 0.0 8.6 0.0 32.0 0.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
MEDIA 4 1 3  -2 -1
MEDIA 0 1 4 -3 -2 -1
MEDIA 2 1 5 -4
BOUNDARY  5
END GEOM
READ PLOT
```

(continues on next page)

(continued from previous page)

```
TTL='X-Y SLICE AT Z MIDPOINT. SINGLE UNIT CONCENTRIC CYLS.'
XUL=-0.3  YUL=8.9  ZUL=16.0  XLR=8.9  YLR=-0.3  ZLR=16.0
UAX=1.0  VDN=-1.0  NAX=640 END
PIC=UNIT NCH='01' END
END PLOT
END DATA
END
```

The plot data included above will draw a mixture map of an X-Y slice taken at the half height (Z = 16.0). It will also draw a unit map of the same slice. The plot dimensions extend 0.3 cm beyond the problem dimensions to provide a black border around the plot. The associated plot data specification for the mixture map is shown in Example 8.1.21, the mixture map is shown in Fig. 8.1.73, and the associated plot data for the unit map is shown in Example 8.1.22. The unit map is identical to Fig. 8.1.72 and is not included.

Example 8.1.21: Associated data for mixture map of single unit with offset origin.

```
x-y slice at z midpoint. single unit concentric cyls.

mixture map

 mixture  0 1 2 4
  symbol    1 2 3
       upper  left          lower right
       coordinates          coordinates
 x     -3.0000e-01           8.9000e+00
 y      8.9000e+00          -3.0000e-01
 z      1.6000e+01           1.6000e+01
           u axis      v axis
           (down)      (across)
 x         0.00000      1.00000
 y        -1.00000      0.00000
 z         0.00000      0.00000
 nu=  640   nv=  640     delu= 1.4375e-02     delv= 1.4375e-02     lpi=  10.000
```

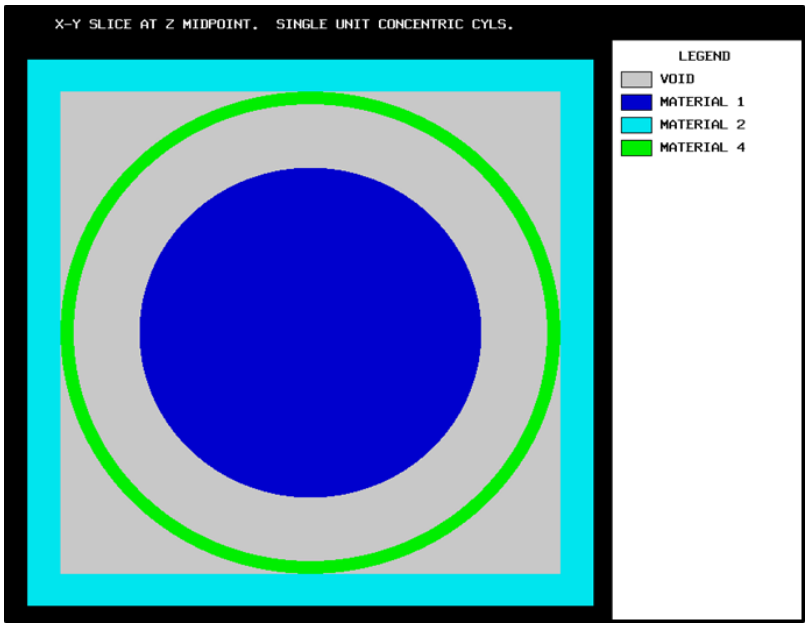


Fig. 8.1.73: Mixture map of single unit with offset origin.

Example 8.1.22: Associated data for unit map of single unit with offset origin.

```
 x-y slice at z midpoint. single unit concentric cyls.

unit map

   unit  1
 symbol  1
      upper  left          lower right
      coordinates          coordinates
x     -3.0000e-01           8.9000e+00
y      8.9000e+00          -3.0000e-01
z      1.6000e+01           1.6000e+01
          u axis      v axis
          (down)      (across)
x         0.00000      1.00000
y        -1.00000      0.00000
z         0.00000      0.00000
nu=  640   nv=  640     delu= 1.4375e-02     delv= 1.4375e-02     lpi=  10.000
```

EXAMPLE 3. A $2 \times 2 \times 2$ UNREFLECTED ARRAY OF CONCENTRIC CYLINDERS IN CUBOIDS

The physical representation of this example is a $2 \times 2 \times 2$ array of the configuration described in Example 1 of this section. The input data description for this array is given below:

KENO V.a:

```
=KENOVA
2x2x2 BARE ARRAY OF CONCENTRIC CYLINDERS IN CUBOID
READ PARAM  RUN=NO LIB=41 END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 502 0.1
MIX=4 200 1.0
END MIXT
READ GEOM
UNIT 1
CYLINDER 1 1 2.6 2P15.0
CYLINDER 0 1 3.6 2P15.0
CYLINDER 4 1 3.8 2P15.0
CUBOID 0 1 4P3.8 2P15.0
CUBOID 2 1 4P4.3 2P16.0
END GEOM
READ ARRAY NUX=2 NUY=2 NUZ=2 END ARRAY
READ PLOT
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER.'
XUL=-0.3 YUL=17.5 ZUL=16.0 XLR=17.5 YLR=-0.3 ZLR=16.0
UAX=1.0 VDN=-1.0 NAX=640  END
TTL='X-Z SLICE THROUGH FRONT ROW, Y=12.9.'
XUL=-1.0 YUL=12.9 ZUL=65.0 XLR=18.2 YLR=12.9 ZLR=-1.0
UAX=1.0 WDN=-1.0 NAX=320 END
END PLOT
END DATA
END
```

KENO-VI:

```
=KENOVI
2x2x2 BARE ARRAY OF CONCENTRIC CYLINDERS IN CUBOID
READ PARAM  RUN=NO LIB=41 TME=8.5 END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 502 0.1
MIX=4 200 1.0
END MIXT
READ GEOM
```

(continues on next page)

(continued from previous page)

```
UNIT 1
CYLINDER 1 2.6 2P15.0
CYLINDER 2 3.6 2P15.0
CYLINDER 3 3.8 2P15.0
CUBOID 4 4P3.8 2P15.0
CUBOID 5 4P4.3 2P16.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
MEDIA 4 1 3 -2 -1
MEDIA 0 1 4 -3 -2 -1
MEDIA  2 1 -5 -4
BOUNDARY  5
GLOBAL UNIT 2
CUBOID 1 17.2 0.0 17.2 0.0 64.0 0.0
ARRAY 1 1 PLACE 1 1 1 4.3 4.3 16.0
BOUNDARY  1
END GEOM
READ ARRAY  ARA=1 NUX=2 NUY=2 NUZ=2 FILL F1 END FILL  END ARRAY

READ PLOT
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER.'
XUL=-0.3 YUL=17.5 ZUL=16.0 XLR=17.5 YLR=-0.3 ZLR=16.0
UAX=1.0 VDN=-1.0 NAX=640 END
TTL='X-Z SLICE THROUGH FRONT ROW, Y=12.9.'
XUL=-1.0 YUL=12.9 ZUL=65.0 XLR=18.2 YLR=12.9 ZLR=-1.0
UAX=1.0 WDN=-1.0 NAX=320 END
END PLOT
END DATA
END
```

As stated at the beginning of Sect. 8.1.4.10, the origin of a plot is defined as the origin of the **GLOBAL UNIT**. Each individual unit in the array is 8.6 cm wide in X and Y and 32 cm high in Z. Since the array has two units stacked in each direction, the array is 17.2 cm wide in X and Y and is 64 cm high. Therefore, the array exists from X = 0.0 to X = 17.2, from Y = 0.0 to Y = 17.2 and from Z = 0.0 to Z = 64.0.

The first color plot is to generate an X-Y slice through the array at the half height (Z = 16.0 cm) of the first layer as shown in Fig. 8.1.74. It is desirable to create an image with a larger extent than the global unit to ensure that the boundaries are as expected. This is achieved by setting the boundaries of the plot larger than the array. In this case, the boundaries were arbitrarily set 0.3 cm larger than the array, resulting in a black border around the array. If the plot were to exclude everything external to the array, the following coordinates could have been entered: **XUL**=0.0 **YUL**=17.2 **ZUL**=16.0 **XLR**=17.2 **YLR**=0.0 **ZLR**=16.0. This would have eliminated the black border. The existing plot was made using **XUL**= -0.3 **YUL**=17.5 **ZUL**=16.0 **XLR**=17.5 **YLR**= -0.3 **ZLR**=16.0.

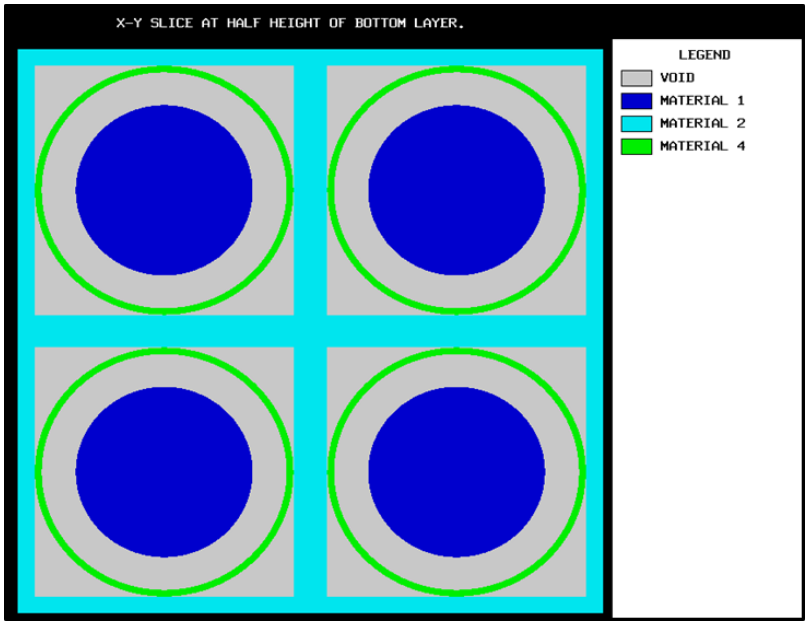


Fig. 8.1.74: X-Y plot of $2 \times 2 \times 2$ bare array.

The second color plot is to generate an X-Z slice through the center of the front row of the array. In order to obtain a black border, the coordinates of X and Z were arbitrarily set 1.0 cm larger than the boundaries of the array. The center of the front row occurs at Y = 12.9. The coordinates of the plot were: **XUL**= -1.0 **ZUL**=65.0 **YUL**=12.9 **XLR**=18.2 **ZLR**= -1.0 **YLR**=12.9. The resulting mixture map is shown in Fig. 8.1.75.

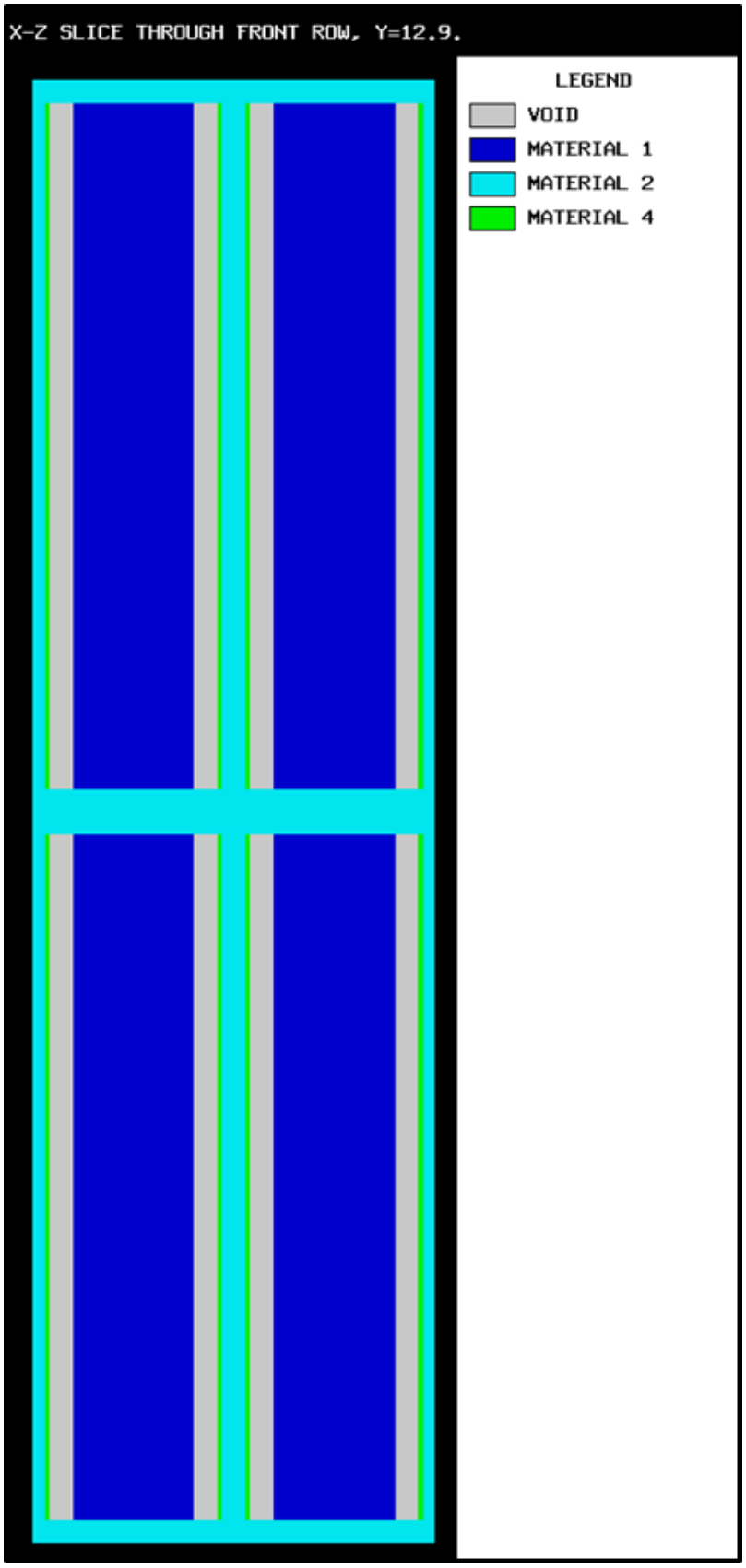


Fig. 8.1.75: X-Z plot of $2 \times 2 \times 2$ bare array.

EXAMPLE 4. A 2 × 2 × 2 REFLECTED ARRAY WITH THE ORIGIN AT THE MOST NEGATIVE POINT OF THE ARRAY

The **ARRAY** is described in Example 3 of this section with a 6 in. concrete reflector on all faces. The input data description for this array is given below.

KENO V.a:

```
=KENOVA
2x2x2 REFLECTED ARRAY OF CONCENTRIC CYLINDERS IN CUBOID
READ PARAM  RUN=NO LIB=41 END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 301 1.0
MIX=4 200 1.0
END MIXT
READ GEOM
UNIT 1
CYLINDER 1 1 2.6 2P15.0
CYLINDER 0 1 3.6 2P15.0
CYLINDER 4 1 3.8 2P15.0
CUBOID 0 1 4P3.8 2P15.0
CUBOID 2 1 4P4.3 2P16.0
GLOBAL
UNIT 2
ARRAY 1  3*0.0
REFLECTOR 3 2 6*5.0 3
REFLECTOR 3 5 6*0.24 1
END GEOM
READ BIAS ID=301 2 5 END BIAS
READ ARRAY ARA=1 NUX=2 NUY=2 NUZ=2 FILL F1 END FILL END ARRAY
READ PLOT
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER.INCLUDES REFL.'
XUL=-16.24 YUL=33.44 ZUL=16.0 XLR=33.44 YLR=-16.24 ZLR=16.0
UAX=1.0 VDN=-1.0 NAX=640  END
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER, INCLUDE 3 CM OF REFL.'
XUL=-3.0 YUL=20.2 ZUL=16.0 XLR=20.2 YLR=-3.0 ZLR=16.0
UAX=1.0 VDN=-1.0 NAX=640 END
TTL='X-Z SLICE THROUGH FRONT ROW, Y=12.9. INCLUDE REFLECTOR'
XUL=-16.24 YUL=12.9 ZUL=80.24 XLR=33.44 YLR=12.9 ZLR=-16.24
UAX=1.0 WDN=-1.0 NAX=640 END
TTL='X-Z SLICE THROUGH FRONT ROW, Y=12.9. INCLUDE 3 CM OF REFLECTOR'
XUL=-3.0 YUL=12.9 ZUL=67.0 XLR=20.2 YLR=12.9 ZLR=-3.0
UAX=1.0 WDN=-1.0 NAX=640 END
END PLOT
END DATA
END
```

KENO-VI:

```
=KENOVI
2x2x2 REFLECTED ARRAY OF CONCENTRIC CYLINDERS IN CUBOID
READ PARAM  RUN=NO LIB=41 TME=0.5 END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 301 1.0
MIX=4 200 1.0
END MIXT
READ GEOM
UNIT 1
CYLINDER 1 2.6 2P15.0
CYLINDER 2 3.6 2P15.0
CYLINDER 3 3.8 2P15.0
CUBOID 4  4P3.8 2P15.0
CUBOID 5  4P4.3 2P16.0
MEDIA 1 1 1
MEDIA 0 1 2 -1
MEDIA 4 1 3 -2
MEDIA 0 1 4 -3
```

(continues on next page)

(continued from previous page)

```
MEDIA 2 1 5 -4
BOUNDARY  5
GLOBAL UNIT 2
CUBOID 1  17.2       0.0    17.2       0.0    64.0        0.0
CUBOID 2  22.20    -5.00  22.20    -5.00  69.00     -5.00
CUBOID 3  27.20  -10.00  27.20  -10.00  74.00   -10.00
CUBOID 4  32.20  -15.00  32.20  -15.00  79.00   -15.00
CUBOID 5  32.44  -15.24  32.44  -15.24  79.24   -15.24
ARRAY 1  1  PLACE 1 1 1 4.3 4.3 16.0
MEDIA 3  2  2  -1
MEDIA 3  3  3  -2
MEDIA 3  4  4  -3
MEDIA 3  5  5  -4
BOUNDARY  5
END GEOM
READ BIAS ID=301 2 5 END BIAS
READ ARRAY  ARA=1  NUX=2  NUY=2  NUZ=2  FILL F1  END FILL   END ARRAY
READ PLOT
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER.INCLUDES REFL.'
XUL=-16.24  YUL=33.44  ZUL=16.0  XLR=33.44  YLR=-16.24  ZLR=16.0
UAX=1.0  VDN=-1.0  NAX=640  END
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER, INCLUDE 3 CM OF REFL.'
XUL=-3.0  YUL=20.2  ZUL=16.0  XLR=20.2  YLR=-3.0  ZLR=16.0
UAX=1.0  VDN=-1.0  NAX=640  END
TTL='X-Z SLICE THROUGH FRONT ROW, Y=12.9. INCLUDE REFLECTOR'
XUL=-16.24  YUL=12.9  ZUL=80.24  XLR=33.44  YLR=12.9  ZLR=-16.24
UAX=1.0  WDN=-1.0  NAX=640  END
TTL='X-Z SLICE THROUGH FRONT ROW, Y=12.9. INCLUDE 3 CM OF REFLECTOR'
XUL=-3.0  YUL=12.9  ZUL=67.0  XLR=20.2  YLR=12.9  ZLR=-3.0
UAX=1.0  WDN=-1.0  NAX=640  END
END PLOT
END DATA
END
```

The **ARRAY** record specifies the array number and the coordinates of the most negative point of the array to be (0.0,0.0,0.0) and the coordinates of the most positive point to be (17.2,17.2,64.0). Thus the reflected array extends from -15.24 cm to +32.44 cm in X and Y and from -15.24 to +79.24 in Z.

The first color plot for this example is to show an X-Y slice through the array and reflector at the half height of the bottom layer. A black border is used to verify that the entire reflector has been shown. This is accomplished by arbitrarily setting the plot boundaries 1 cm beyond the reflector boundaries. The coordinates used for this plot are **XUL** = -16.24 **YUL** = 33.44 **ZUL** = 16.0 **XLR** = 33.44 **YLR** = 16.24 **ZLR** = 16.0. The plot data description is shown in Example 8.1.23, and the plot is shown in Fig. 8.1.76.

Example 8.1.23: Plot data for X-Y slice of Example 4.

```
x-y slice at half height of bottom layer. includes refl.

mixture map

 mixture  0 1 2 3 4
  symbol    1 2 3 4
       upper  left          lower right
       coordinates          coordinates
 x     -1.6240e+01           3.3440e+01
 y      3.3440e+01          -1.6240e+01
 z      1.6000e+01           1.6000e+01
          u axis      v axis
          (down)      (across)
 x         0.00000      1.00000
 y        -1.00000      0.00000
```

(continues on next page)

(continued from previous page)

```
z          0.00000      0.00000
nu=  640   nv=  640     delu= 7.7625e-02     delv= 7.7625e-02     lpi=  10.000
```

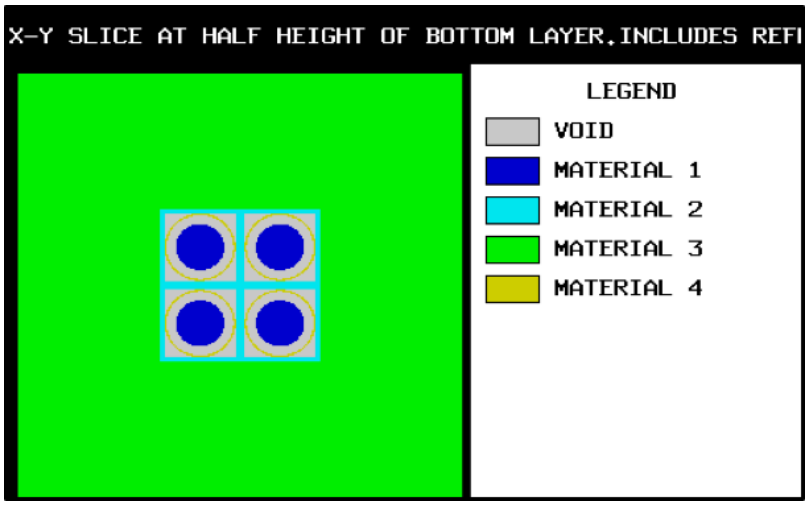


Fig. 8.1.76: X-Y plot of $2 \times 2 \times 2$ reflected array.

The next color plot is the same as the previous plot, except this plot includes only the first 3 cm of the reflector in order to show more detail. The coordinates used for this plot are **XUL**= -3.0 **YUL**=20.2 **ZUL**=16.0 **XLR**=20.2 **YLR**= -3.0 **ZLR**=16.0. This plot data description is given in Example 8.1.24, and the plot is shown in Fig. 8.1.77.

Example 8.1.24: Plot data for enlarged X-Y slice of Example 4.

```
x-y slice at half height of bottom layer, include 3 cm of refl.
mixture map

 mixture  0 1 2 3 4
  symbol    1 2 3 4
       upper  left          lower right
       coordinates          coordinates
 x     -3.0000e+00           2.0200e+01
 y      2.0200e+01          -3.0000e+00
 z      1.6000e+01           1.6000e+01
          u axis      v axis
          (down)      (across)
 x         0.00000      1.00000
 y        -1.00000      0.00000
 z         0.00000      0.00000
 nu=  640   nv=  640     delu= 3.6250e-02     delv= 3.6250e-02     lpi=  10.000
```

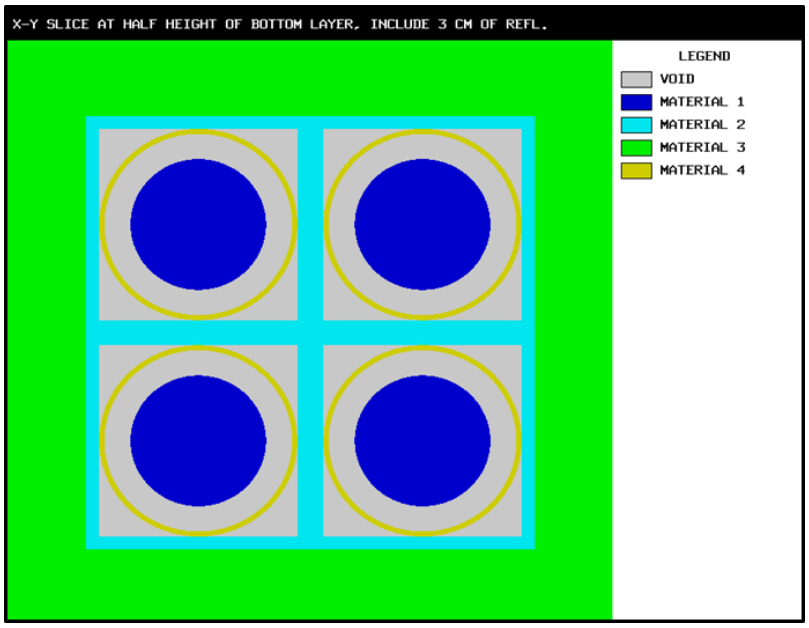


Fig. 8.1.77: Enlarged X-Y plot of $2 \times 2 \times 2$ reflected array.

The third color plot for this example is an X-Z slice through the center of the front row. An extra 1 cm is included in the coordinates to provide a black border around the plot. The coordinates are: **XUL**= -16.24 **YUL**=12.9 **ZUL**=80.24 **XLR**=33.44 **YLR**=12.9 **ZLR**= -16.24. The resultant plot data and plot are shown in Example 8.1.25 and Fig. 8.1.78. X-Z plot for $2 \times 2 \times 2$ reflected array.

Example 8.1.25: Plot data for X-Z slice of Example 4.

```
x-z slice through front row, y=12.9. include reflector

mixture map

 mixture  0 1 2 3 4
  symbol    1 2 3 4
```

(continues on next page)

(continued from previous page)

```
      upper  left          lower right
      coordinates          coordinates
x     -1.6240e+01           3.3440e+01
y      1.2900e+01           1.2900e+01
z      8.0240e+01          -1.6240e+01
          u axis      v axis
          (down)      (across)
x         0.00000      1.00000
y         0.00000      0.00000
z        -1.00000      0.00000
nu= 1242   nv=  640     delu= 7.7625e-02     delv= 7.7625e-02     lpi=  10.000
```

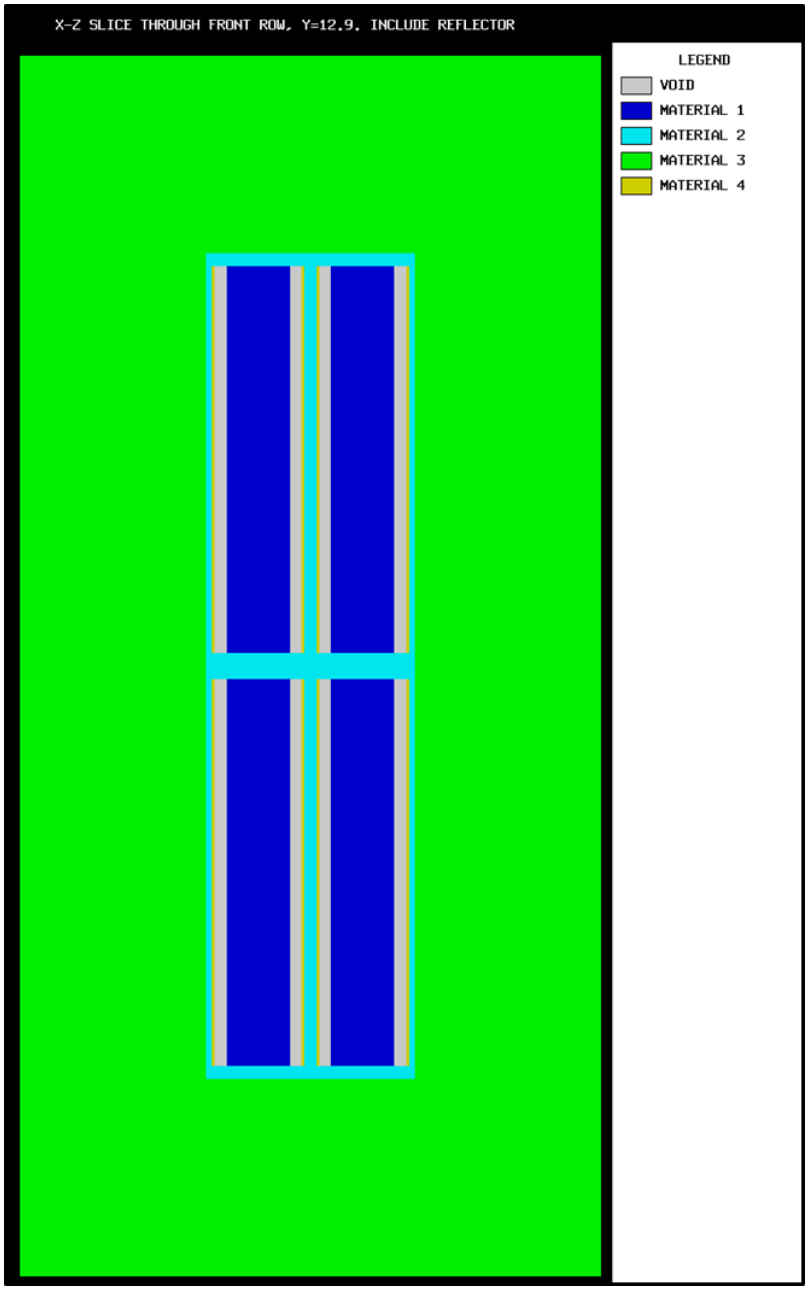


Fig. 8.1.78: X-Z plot for $2 \times 2 \times 2$ reflected array.

The last color plot for this example is the same as the previous one, except only 3 cm of the reflector is included in the plot. The plot data and associated plot are shown in Example 8.1.26 and Fig. 8.1.79.

Example 8.1.26: Plot data for enlarged X-Z slice of Example 4.

```
x-z slice through front row, y=12.9. include 3 cm of reflector

mixture map

 mixture  0 1 2 3 4
  symbol    1 2 3 4
       upper  left          lower right
       coordinates          coordinates
 x     -3.0000e+00           2.0200e+01
 y      1.2900e+01           1.2900e+01
 z      6.7000e+01          -3.0000e+00
           u axis      v axis
           (down)      (across)
 x         0.00000      1.00000
 y         0.00000      0.00000
 z        -1.00000      0.00000
 nu= 1931   nv=  640     delu= 3.6250e-02     delv= 3.6250e-02     lpi=  10.000
```

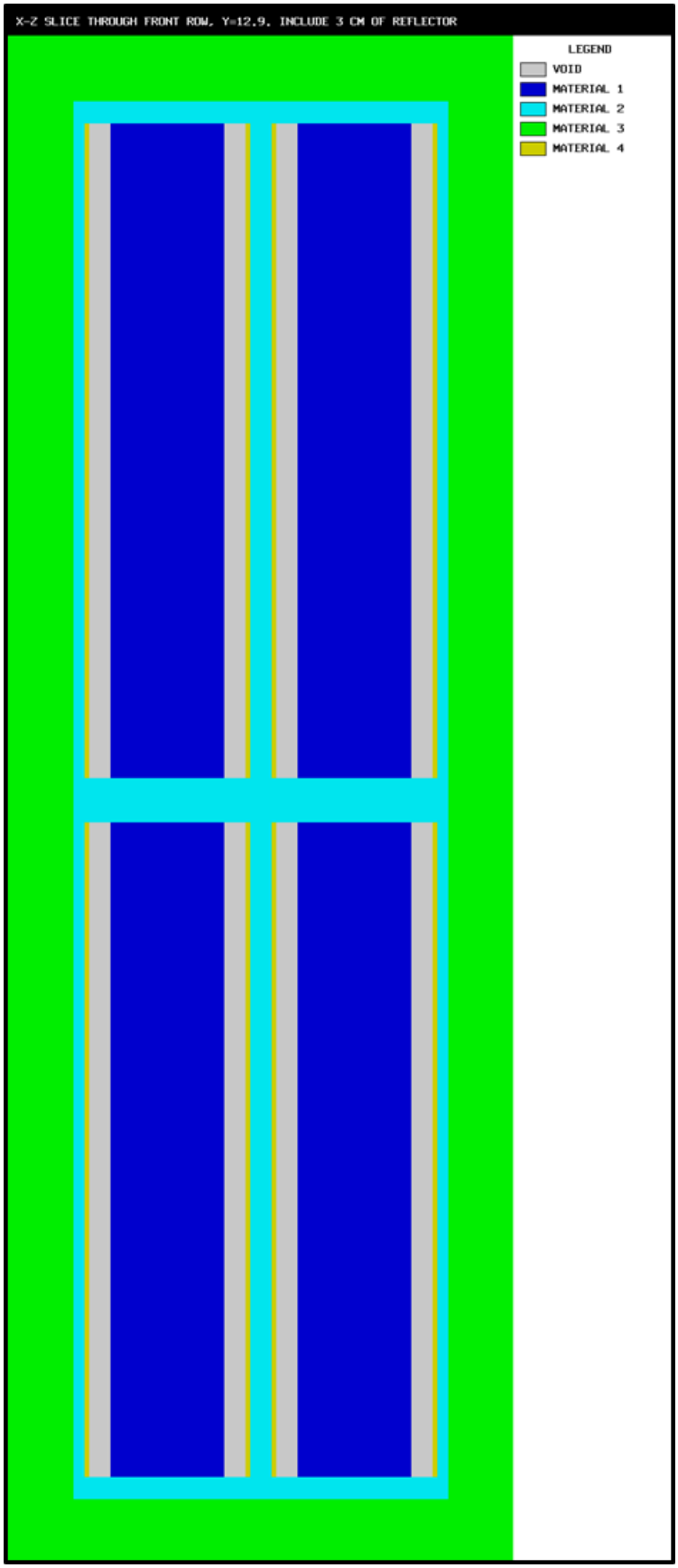


Fig. 8.1.79: Enlarged X-Z plot of $2 \times 2 \times 2$ reflected array.

EXAMPLE 5. A 2 × 2 × 2 REFLECTED ARRAY WITH THE ORIGIN CENTERED IN THE ARRAY

This example is physically identical to Example 4. The difference is in the specification of the origin. The bare array is 17.2 cm wide in X and Y and 64 cm high. The origin (0,0,0) can be placed at the exact center of the array by specifying the most negative point of the array as X = -8.6, Y = -8.6 and Z = -32.0. This is done using the **ARRAY** description in the geometry block. Because the origin is located at a different position, the coordinates of the plots will also be different. The input data description for this example is given below.

KENO V.a:

```
=KENOVA
2x2x2 REFLECTED ARRAY OF CONCENTRIC CYLINDERS IN CUBOID
READ PARAM  RUN=NO LIB=41 TME=0.5 END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 301 1.0
MIX=4 200 1.0
END MIXT
READ GEOM
UNIT 1
CYLINDER 1 1 2.6 2P15.0
CYLINDER 0 1 3.6 2P15.0
CYLINDER 4 1 3.8 2P15.0
CUBOID 0 1 4P3.8 2P15.0
CUBOID 2 1 4P4.3 2P16.0
GLOBAL
UNIT 2
ARRAY 1 1 2*-8.6 -32.0
REFLECTOR 3 2 6*5.0 3
REFLECTOR 3 5 6*0.24 1
END GEOM
READ BIAS ID=301 2 5 END BIAS
READ ARRAY ARA=1 NUX=2 NUY=2 NUZ=2 FILL F1 END FILL  END ARRAY
READ PLOT
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER.INCLUDES REFL.'
XUL=-24.84 YUL=24.84 ZUL=-8.0 XLR=24.84 YLR=-24.84 ZLR=-8.0
UAX=1.0 VDN=-1.0 NAX=640  END
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER, INCLUDE 3 CM OF REFL.'
XUL=-11.6 YUL=11.6 ZUL=-8.0 XLR=11.6 YLR=-11.6 ZLR=-8.0
UAX=1.0 VDN=-1.0 NAX=640  END
TTL='X-Z SLICE THROUGH FRONT ROW. Y=4.3  INCLUDE REFLECTOR'
XUL=-24.84 YUL=4.3 ZUL=48.24 XLR=24.84 YLR=4.3 ZLR=-48.24
UAX=1.0 WDN=-1.0 NAX=640 END
TTL='X-Z SLICE THROUGH FRONT ROW, Y=4.3  INCLUDE 3 CM OF REFLECTOR'
XUL=-11.6 YUL=4.3 ZUL=35.0 XLR=11.6 YLR=4.3 ZLR=-35.0
UAX=1.0 WDN=-1.0 NAX=640 END
END PLOT
END DATA
END
```

KENO-VI:

```
=KENOVI
2x2x2 REFLECTED ARRAY OF CONCENTRIC CYLINDERS IN CUBOID
READ PARAM  RUN=NO LIB=41 TME=0.5 END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 301 1.0
MIX=4 200 1.0
END MIXT
READ GEOM
UNIT 1
CYLINDER 1 2.6 2P15.0
CYLINDER 2 3.6 2P15.0
CYLINDER 3 3.8 2P15.0
CUBOID 4 4P3.8 2P15.0
CUBOID 5 4P4.3 2P16.0
MEDIA 1 1
```

(continues on next page)

(continued from previous page)

```
MEDIA 0 1 2 -1
MEDIA 4 1 3 -2
MEDIA 0 1 4 -3
MEDIA 2 1 5 -4
BOUNDARY  5
GLOBAL UNIT 2
CUBOID 1 8.6 -8.6 8.6 -8.6 32.0 -32.0
CUBOID 2 13.6 -13.6 13.6 -13.6 37.0 -37.0
CUBOID 3 18.6 -18.6 18.6 -18.6 42.0 -42.0
CUBOID 4 23.6 -23.6 23.6 -23.6 47.0 -47.0
CUBOID 5 23.84 -23.84 23.84 -23.84 47.24 -47.24
ARRAY 1 1 PLACE 1 1 1 -4.3 -4.3 -16.0
MEDIA 3 2 2 -1
MEDIA 3 3 3 -2
MEDIA 3 4 4 -3
MEDIA 3 5 5 -4
BOUNDARY  5
END GEOM
READ BIAS ID=301 2 5 END BIAS
READ ARRAY ARA=1 NUX=2 NUY=2 NUZ=2I FILL F1 END FILL   END ARRAY
READ PLOT
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER INCLUDES REFL.'
XUL=-24.84 YUL=24.84 ZUL=-8.0 XLR=24.84 YLR=-24.84 ZLR=-8.0
UAX=1.0  VDN=-1.0  NAX=130  NCH=' *=.X'  END
TTL='X-Y SLICE AT HALF HEIGHT OF BOTTOM LAYER, INCLUDE 3 CM OF REFL.'
XUL=-11.6  YUL=11.6  ZUL=-8.0  XLR=11.6  YLR=-11.6  ZLR=-8.0
UAX=1.0  VDN=-1.0  NAX=640  END
TTL='X-Z SLICE THROUGH FRONT ROW. Y=4.3  INCLUDE REFLECTOR'
XUL=-24.84  YUL=4.3  ZUL=48.24  XLR=24.84  YLR=4.3  ZLR=-48.24
UAX=1.0  WDN=-1.0  NAX=640 END
TTL='X-Z SLICE THROUGH FRONT ROW, Y=4.3  INCLUDE 3 CM OF REFLECTOR'
XUL=-11.6  YUL=4.3  ZUL=35.0  XLR=11.6  YLR=4.3  ZLR=-35.0
UAX=1.0  WDN=-1.0  NAX=640  END
END PLOT
END DATA
END
```

The first color plot for this example covers exactly the same area as the first plot for Example 4. The plot data and the plot are given in Example 8.1.27 and Fig. 8.1.80, respectively.

Example 8.1.27: Plot data for X-Y slice of Example 5.

```
x-y slice at half height of bottom layer.includes refl.
mixture map

 mixture  0 1 2 3 4
  symbol    1 2 3 4
       upper  left          lower right
       coordinates          coordinates
 x     -2.4840e+01           2.4840e+01
 y      2.4840e+01          -2.4840e+01
 z     -8.0000e+00          -8.0000e+00
           u axis      v axis
           (down)      (across)
 x         0.00000      1.00000
 y        -1.00000      0.00000
 z         0.00000      0.00000
 nu=  640   nv=  640     delu= 7.7625e-02     delv= 7.7625e-02     lpi=  10.000
```

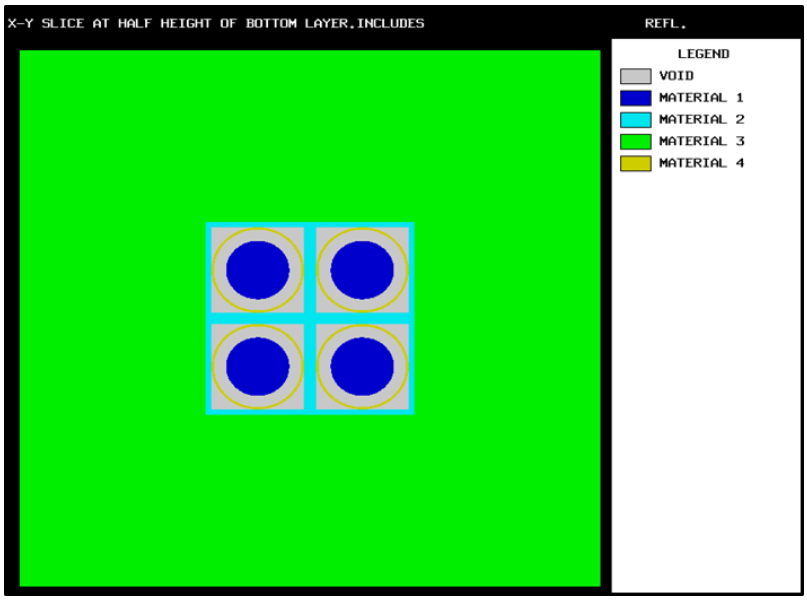


Fig. 8.1.80: X-Y plot of $2 \times 2 \times 2$ reflected array with centered origin.

The Example 5 plot data and associated plots for an enlarged X-Y plot, an X-Z plot and an enlarged X-Z plot are given in Example 8.1.28 through Fig. 8.1.83.

Example 8.1.28: Plot data for an enlarged X-Y slice of Example 5.

```
x-y slice at half height of bottom layer, include 3 cm of refl.

mixture map

 mixture  0 1 2 3 4
  symbol    1 2 3 4
       upper  left          lower right
       coordinates          coordinates
 x     -1.1600e+01           1.1600e+01
 y      1.1600e+01          -1.1600e+01
```

(continues on next page)

(continued from previous page)

```
z     -8.0000e+00          -8.0000e+00
          u axis       v axis
          (down)       (across)
x         0.00000       1.00000
y        -1.00000       0.00000
z         0.00000       0.00000
nu=  640   nv=  640     delu= 3.6250e-02     delv= 3.6250e-02     lpi=  10.000
```

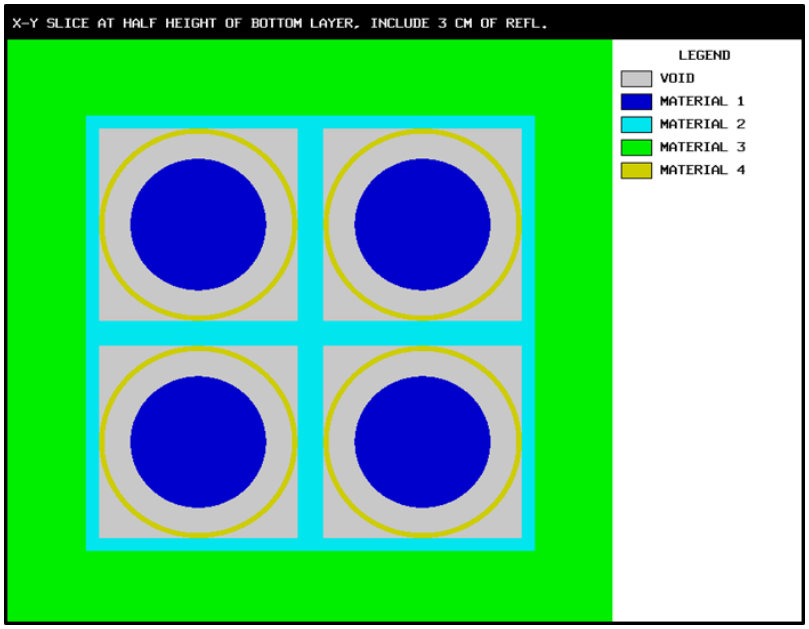


Fig. 8.1.81: Enlarged X-Y plot of $2 \times 2 \times 2$ reflected array with centered origin.

Example 8.1.29: Plot data for X-Z slice of Example 5.

```
x-z slice through front row. y=4.3  include reflector

mixture map

 mixture  0 1 2 3 4
  symbol    1 2 3 4
       upper  left          lower right
       coordinates          coordinates
 x     -2.4840e+01           2.4840e+01
 y      4.3000e+00           4.3000e+00
 z      4.8240e+01          -4.8240e+01
          u axis      v axis
          (down)      (across)
 x        0.00000      1.00000
 y        0.00000      0.00000
 z       -1.00000      0.00000
 nu= 1242   nv=  640     delu= 7.7625e-02     delv= 7.7625e-02     lpi=  10.000
```

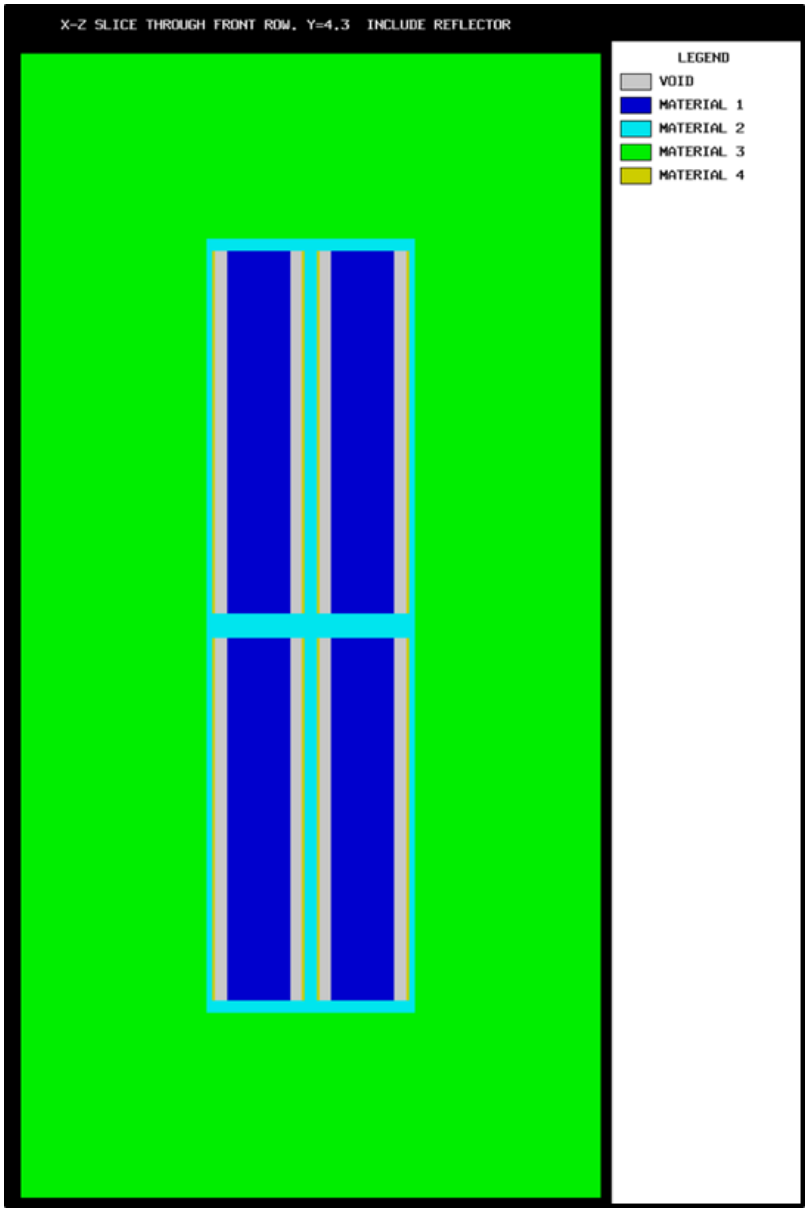


Fig. 8.1.82: X-Z plot of reflected 2 × 2 × 2 array with centered origin.

Example 8.1.30: Plot data for enlarged X-Z slice of Example 5.

```
x-z slice through front row, y=4.3  include 3 cm of reflector

mixture map

 mixture  0 1 2 3 4
  symbol    1 2 3 4
       upper  left          lower right
       coordinates          coordinates
 x     -1.1600e+01           1.1600e+01
 y      4.3000e+00           4.3000e+00
 z      3.5000e+01          -3.5000e+01
```

(continues on next page)

(continued from previous page)

```
           u axis      v axis
           (down)      (across)
x          0.00000      1.00000
y          0.00000      0.00000
z         -1.00000      0.00000
nu= 1931   nv=  640     delu= 3.6250e-02     delv= 3.6250e-02     lpi=  10.000
```

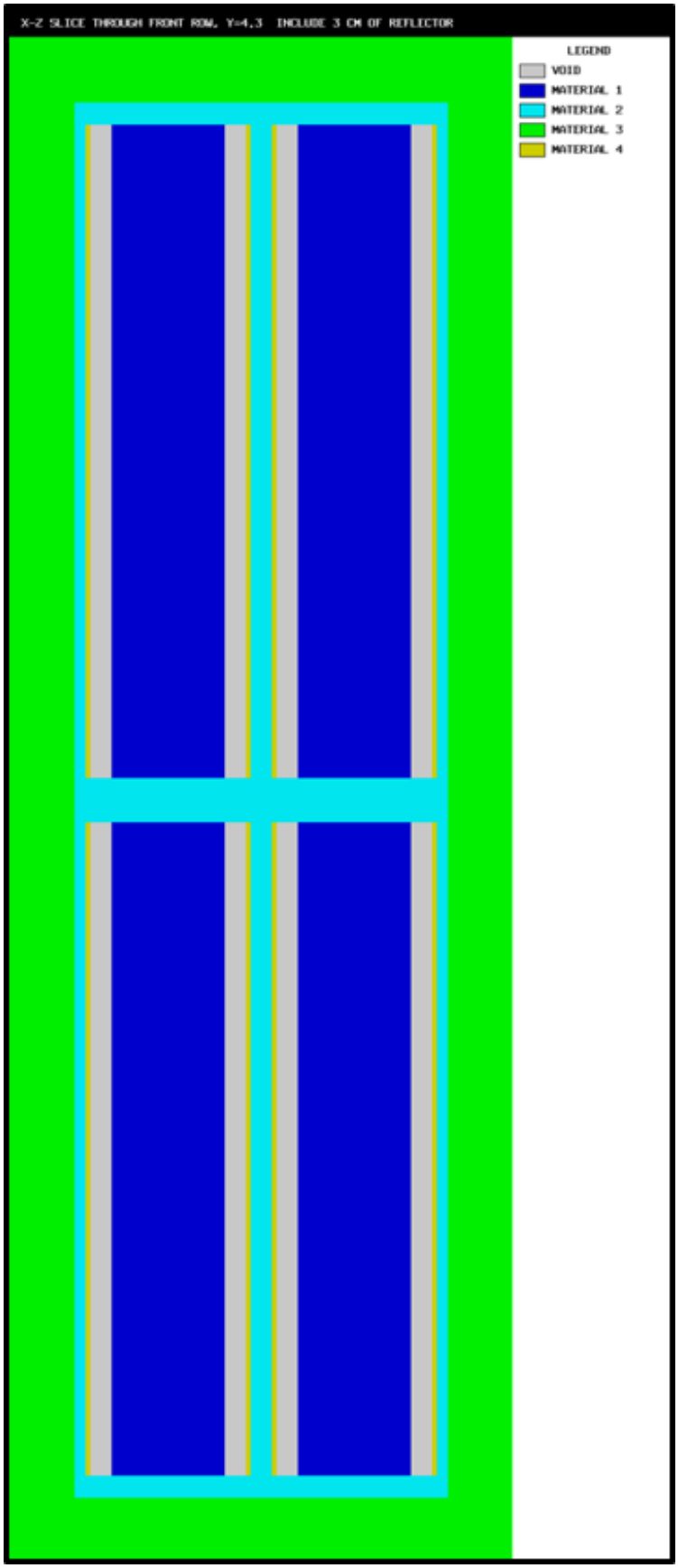


Fig. 8.1.83: Enlarged X-Z plot of reflected $2 \times 2 \times 2$ array with centered origin.

EXAMPLE 6. NESTED HOLES.

The nested hole description is provided in Sect. 8.1.4.6.2. Because this example involves a complicated placement of units, it is helpful useful to the user to generate a mixture plot and/or a unit plot for the problem. The resultant mixture plot for this problem is shown in Fig. 8.1.84, and the data description for Example 6 follows.

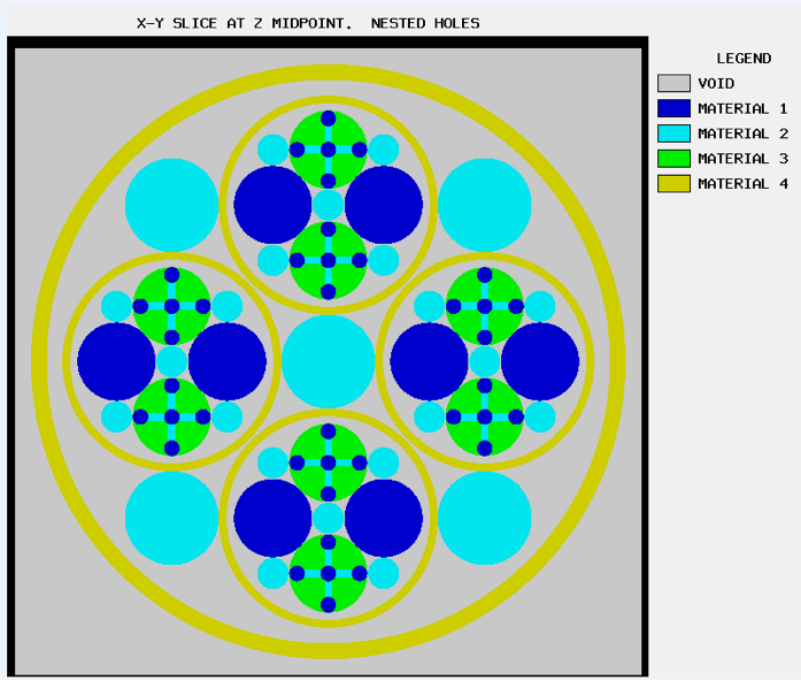


Fig. 8.1.84: Mixture plot of nested holes problem.

Example 8.1.31: Plot data for unit plot of nested holes.

```
x-y slice at z midpoint. nested holes. unit map
0                                        unit map
    unit  1 2 3 4 5 6 7 8 9
  symbol  1 2 3 4 5 6 7 8 9

       overall system coordinates:
       xmin= 0.00000e+00   xmax= 8.00000e+00   ymin= 0.00000e+00   ymax= 8.00000e+00
       zmin= 0.00000e+00   zmax= 3.20000e+01
       upper  left          lower right
       coordinates          coordinates
 x     -1.0000e-01           8.1000e+00
 y      8.1000e+00          -1.0000e-01
 z      1.6000e+01           1.6000e+01
          u axis      v axis
          (down)      (across)
 x         0.00000      1.00000
 y        -1.00000      0.00000
 z         0.00000      0.00000
 nu=  640   nv=  640     delu= 1.2813e-02     delv= 1.2813e-02     lpi=  10.000
```

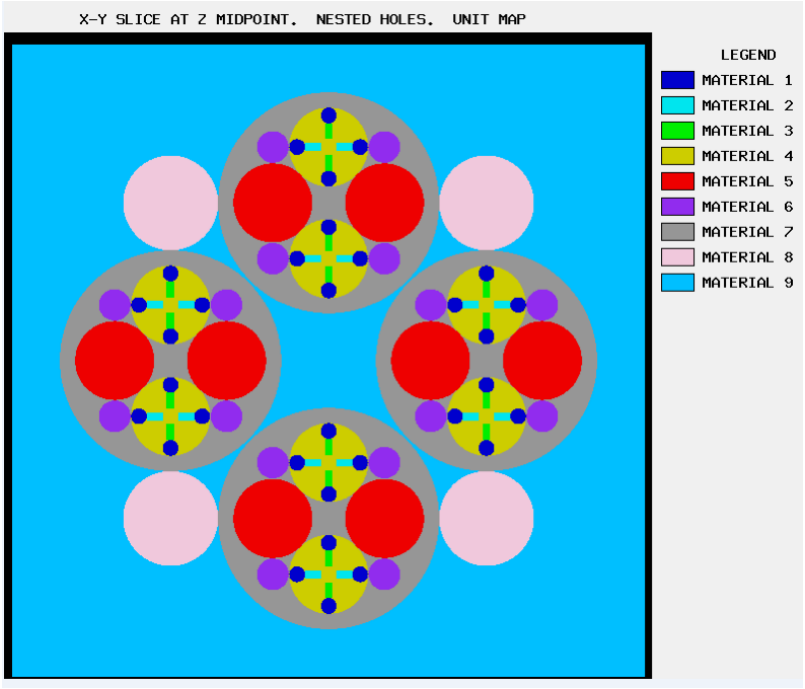


Fig. 8.1.85: Unit plot of nested holes problem.

KENO V.a:

```
=KENOVA
NESTED HOLES SAMPLE
READ PARAM  RUN=NO LIB=41  END PARAM
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 502 0.1
MIX=4 200 1.0
END MIXT
READ GEOM
```

(continues on next page)

(continued from previous page)

```
UNIT 1
CYLINDER 1 1 0.1 2P15.0
UNIT 2
CUBOID 2 1 2P0.1 2P0.05 2P15.0
UNIT 3
CUBOID 2 1 2P0.05 2P0.1 2P15.0
UNIT 4
CYLINDER 1 1 0.1 2P15.0
CYLINDER 3 1 0.5 2P15.0
HOLE 1  0.0 -0.4 0.0
HOLE 1  0.4  0.0 0.0
HOLE 1  0.0  0.4 0.0
HOLE 1 -0.4  0.0 0.0
HOLE 2 -0.2  0.0 0.0
HOLE 2  0.2  0.0 0.0
HOLE 3  0.0 -0.2 0.0
HOLE 3  0.0  0.2 0.0
UNIT 5
CYLINDER 1 1 0.5 2P15.0
UNIT 6
CYLINDER 2 1 0.2 2P15.0
UNIT 7
CYLINDER 2 1 0.2 2P15.0
CYLINDER 0 1 1.3 2P15.0
HOLE 5 0.707107 2*0.0
HOLE 6 0.707107 0.707107 0.0
HOLE 4 0.0 0.707107 0.0
HOLE 6 -0.707107 0.707107 0.0
HOLE 5 -0.707107 0.0 0.0
HOLE 6 -0.707107 -0.707107 0.0
HOLE 4 0.0 -0.707107 0.0
HOLE 6 0.707107 -0.707107 0.0
CYLINDER 4 1 1.4 2P15.0
UNIT 8
CYLINDER 2 1 0.6 2P15.0
UNIT 9
CYLINDER 2 1 0.6 2P15.0
CYLINDER 0 1 3.6 2P15.0
HOLE 7 2.0 0.0 0.0
HOLE 8 2*2.0 0.0
HOLE 7 0.0 2.0 0.0
HOLE 8 -2.0 2.0 0.0
HOLE 7 -2.0 2*0.0
HOLE 8 2*-2.0 0.0
HOLE 7 0.0 -2.0 0.0
HOLE 8 2P2.0 0.0
CYLINDER 4 1 3.8 2P15.0
CUBOID 0 1 4P4.0 2P16.0
END GEOM
READ ARRAY ARA=1 NUX=1 NUY=1 NUZ=1  FILL 9 END ARRAY
READ PLOT
TTL='X-Y SLICE AT Z MIDPOINT. NESTED HOLES'
XUL=-0.1 YUL=8.1 ZUL=16.0 XLR=8.1 YLR=-0.1 ZLR=16
UAX=1.0 VDN=-1.0 NAX=640  END
TTL='X-Y SLICE AT Z MIDPOINT. NESTED HOLES. UNIT MAP'
PIC=UNIT  END
END PLOT
END DATA
END
```

KENO-VI:

```
=KENOVI
NESTED HOLES SAMPLE
READ PARAM  RUN=NO LIB=41 TME=0.5 END PARAM
```

(continues on next page)

(continued from previous page)

```
READ MIXT  SCT=1  MIX=1 92500 4.7048-2 MIX=2 200 1.0 MIX=3 502 0.1
MIX=4 200 1.0
END MIXT
READ GEOM
UNIT 1
CYLINDER 1 0.1 2P15.0
CYLINDER 2 0.1 2P15.0 ORIGIN Y=-0.4
CYLINDER 3 0.1 2P15.0 ORIGIN X=0.4
CYLINDER 4 0.1 2P15.0 ORIGIN Y=0.4
CYLINDER 5 0.1 2P15.0 ORIGIN X=-0.4
CUBOID 6 2P0.1 2P0.05 2P15.0 ORIGIN X=-0.2
CUBOID 7 2P0.1 2P0.05 2P15.0 ORIGIN X=0.2
CUBOID 8 2P0.05 2P0.1 2P15.0 ORIGIN Y=-0.2
CUBOID 9 2P0.05 2P0.1 2P15.0 ORIGIN Y=0.2
CYLINDER 10 0.5 2P15.0
MEDIA 1 1 1 -6 -7 -8 -9
MEDIA 1 1 2 -8
MEDIA 1 1 3 -7
MEDIA 1 1 4 -9
MEDIA 1 1 5 -6
MEDIA 2 1 6 -1 -5
MEDIA 2 1 7 -1 -3
MEDIA 2 1 8 -1 -2
MEDIA 2 1 9 -1 -4
MEDIA 3 1 -1 -2 -3 -4 -5 -6 -7 -8 -9 10
BOUNDARY  10
UNIT 2
CYLINDER 1 0.2 2P15.0
CYLINDER 2 0.2 2P15.0 ORIGIN X=0.707107 Y=0.707107
CYLINDER 3 0.2 2P15.0 ORIGIN X=-0.707107 Y=0.707107
CYLINDER 4 0.2 2P15.0 ORIGIN X=-0.707107 Y=-0.707107
CYLINDER 5 0.2 2P15.0 ORIGIN X=0.707107 Y=-0.707107
CYLINDER 6 0.5 2P15.0 ORIGIN X=0.707107
CYLINDER 7 0.5 2P15.0 ORIGIN X=-0.707107
CYLINDER 8 0.5 2P15.0 ORIGIN Y=0.707107
CYLINDER 9 0.5 2P15.0 ORIGIN Y=-0.707107
CYLINDER 10 1.3 2P15.0
CYLINDER 11 1.4 2P15.0
MEDIA 2 1 1
MEDIA 2 1 2
MEDIA 2 1 3
MEDIA 2 1 4
MEDIA 2 1 5
MEDIA 1 1 6 -8 -9
MEDIA 1 1 7 -8 -9
HOLE 1 8 -3 -6 ORIGIN Y=0.707107
HOLE 1 9 -6 -7 ORIGIN Y=-0.707107
MEDIA 0 1 10 -1 -2 -3 -4 -5 -6 -7 -8 -9
MEDIA 4 1 11 -10
BOUNDARY  11
GLOBAL UNIT 3
CYLINDER 1 0.6 2P15.0
CYLINDER 2 0.6 2P15.0 ORIGIN X=2.0 Y=2.0
CYLINDER 3 0.6 2P15.0 ORIGIN X=-2.0 Y=2.0
CYLINDER 4 0.6 2P15.0 ORIGIN X=-2.0 Y=-2.0
CYLINDER 5 0.6 2P15.0 ORIGIN X=2.0 Y=-2.0
CYLINDER 6 1.4 2P15.0 ORIGIN X=2.0
CYLINDER 7 1.4 2P15.0 ORIGIN Y=2.0
CYLINDER 8 1.4 2P15.0 ORIGIN X=-2.0
CYLINDER 9 1.4 2P15.0 ORIGIN Y=-2.0
CYLINDER 10 3.6 2P15.0
CYLINDER 11 3.8 2P15.0
CUBOID  12  4P4.0 2P16.0
MEDIA 2 1 1 -6 -7 -8 -9
MEDIA 2 1 2 -6 -7
```

(continues on next page)

(continued from previous page)

```
MEDIA 2 1 3 -7 -8
MEDIA 2 1 4 -8 -9
MEDIA 2 1 5 -9 -6
HOLE 2 6 -1 -5 -2 ORIGIN X=2
HOLE 2 7 -1 -2 -3 ORIGIN Y=2
HOLE 2 8 -1 -3 -4 ORIGIN X=-2
HOLE 2 9 -1 -4 -5 ORIGIN Y=-2
MEDIA 0 1 10 -1 -2 -3 -4 -5 -6 -7 -8 -9
MEDIA 4 1 11 -10
MEDIA 0 1 12 -11
BOUNDARY  12
END GEOM
READ PLOT
TTL='X-Y SLICE AT Z MIDPOINT. NESTED HOLES'
XUL=-4.1  YUL=4.1 ZUL=0.0 XLR=4.1 YLR=-4.1 ZLR=0
UAX=1.0 VDN=-1.0  NAX=640 END
TTL='X-Y SLICE AT Z MIDPOINT. NESTED HOLES, UNIT MAP.'
PIC=UNIT END
END PLOT
END DATA
END
```

.he plot data description for the **UNIT** plot of nested holes is shown in Example 8.1.31. The **UNIT** plot is shown in Fig. 8.1.85. The user can reference this map to verify the correct placement of the **UNIT**s. Note that the **UNIT** map plots the **UNIT**s present at the deepest nesting level for the 2-D slice. It does not show any detail within a **UNIT**. The apparent detail within **UNIT** 7 includes **UNIT**s 1–6, which were placed there via the hole option. In the legend of the plot, the material number actually refers to the **UNIT** number.

EXAMPLE 7. LARGE STORAGE ARRAY.

The storage array described Example 18 in Sect. 8.1.4.6.3 and Fig. 8.1.33 is such a sparse array that the mixture map had to be very large in order to show the detail of the shelves and uranium buttons. The mixture maps for this configuration were not presented in Sect. 8.1.4.6.3, but the data description was listed so the user can generate them. It may be useful to generate a **UNIT** map for this kind of problem. The input data for generating unit maps for this storage array is given below.

```
READ PLOT  PIC=UNIT
TTL='X-Z SLICE THROUGH STORAGE ARRAY ROOM AT Y=30.48 WITH Z ACROSS AND X DOWN'
XUL=624.84 YUL=30.48 ZUL=-45.72 XLR=-30.48 YLR=30.48 ZLR=381.0
WAX=1.0 UDN=-1.0 NAX=320  END
TTL='X-Y SLICE THROUGH STORAGE ARRAY ROOM AT Z=0.3175 WITH X ACROSS AND Y DOWN'
XUL=-30.48 YUL=1341.1 ZUL=0.3175 XLR=624.84 YLR=-30.48 ZLR=0.3175
UAX=1.0 VDN=-1.0 NAX=320 END
END PLOT
```

The plot data and **UNIT** map for an X-Z slice through the array at Y=30.48 cm is given in Example 8.1.32 and Fig. 8.1.86. The Z direction, which extends from -45.72 cm to 381.0 cm, is plotted in 320 pixels across the plot. This **UNIT** map was created with Z across the plot and X down the plot.

Example 8.1.32: Plot data for X-Z slice of storage array.

```
x-z slice through storage array room at y=30.48 with z across and x down

unit map

          unit  1 2 3 4 5 6 7
        symbol  1 2 3 4 5 6 7
            upper  left          lower right
```

(continues on next page)

(continued from previous page)

```
          coordinates          coordinates
      x    6.2484e+02          -3.0480e+01
      y    3.0480e+01           3.0480e+01
      z   -4.5720e+01           3.8100e+02
              u axis      v axis
              (down)      (across)
      x      -1.00000     0.00000
      y       0.00000     0.00000
      z       0.00000     1.00000
      nu=  491   nv=  320    delu= 1.3335e+00     delv= 1.3335e+00     lpi=  10.000
```

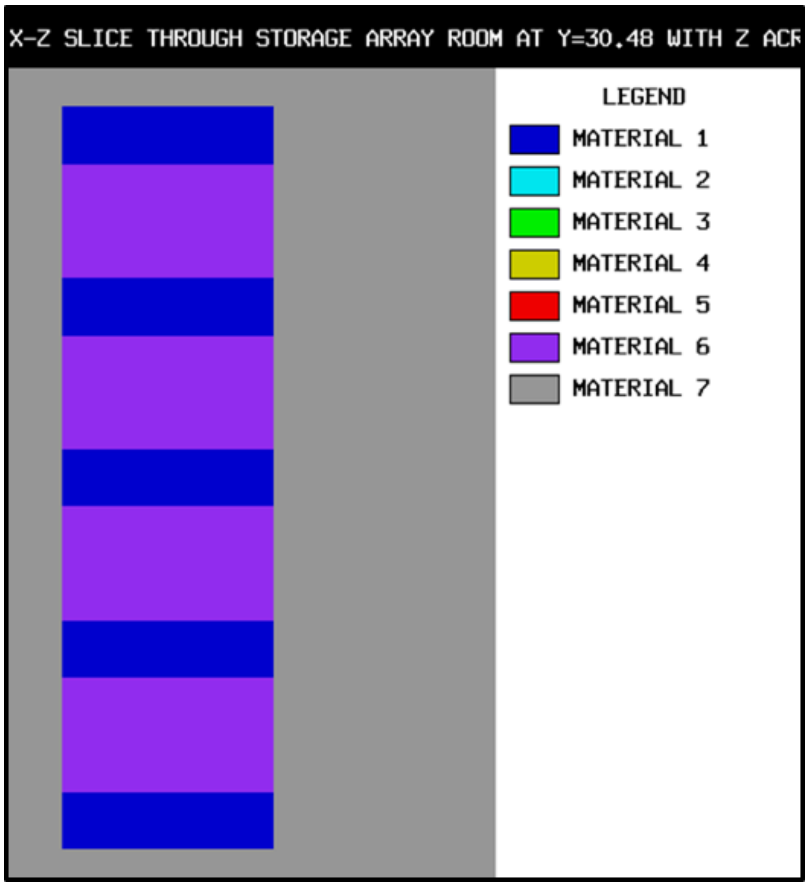


Fig. 8.1.86: X-Z plot of storage array.

The plot data and **UNIT** map for an X-Y slice through the shelf are given in Example 8.1.33 and Fig. 8.1.87. This **UNIT** map was created with X across the plot and Y down the plot. This shows five rows of shelves in the X direction.

Example 8.1.33: Plot data for X-Y slice of storage array.

```
x-y slice through storage array room at z=0.3175 with x across and y down
unit map

    unit  1 2 3 4 5 6 7
  symbol  1 2 3 4 5 6 7
       upper  left          lower right
```

(continues on next page)

(continued from previous page)

```
        coordinates          coordinates
x     -3.0480e+01           6.2484e+02
y      1.3411e+03          -3.0480e+01
z      3.1750e-01           3.1750e-01
          u axis      v axis
          (down)      (across)
x         0.00000      1.00000
y        -1.00000      0.00000
z         0.00000      0.00000
nu=  669   nv=  320     delu= 2.0479e+00     delv= 2.0479e+00     lpi=  10.000
```

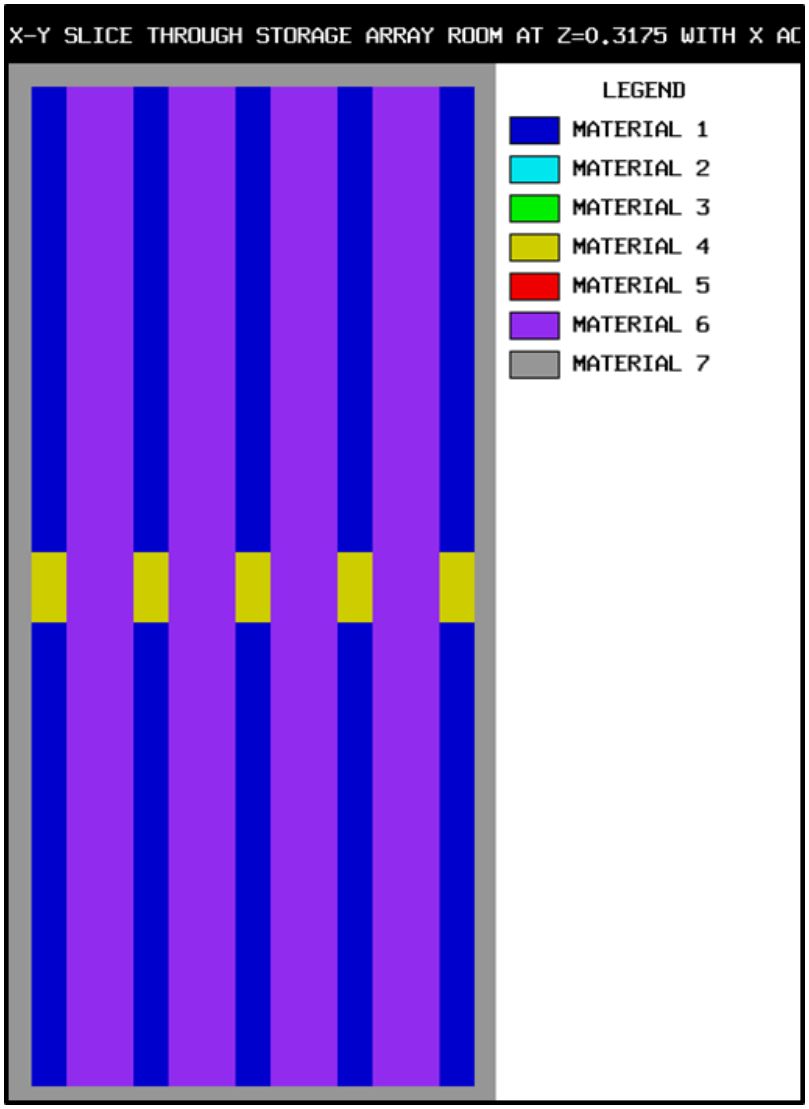


Fig. 8.1.87: X-Y plot of storage array.

8.1.4.11 KENO Multiple Mesh and Mesh-based Quantity Specifications

KENO constructs a grid object and stores several mesh definitions in this object. The following mesh definitions are supported by KENO:

i. Default mesh grid (grid `ID` =10001):

KENO generates a $5 \times 5 \times 5$ Cartesian mesh grid which overlays the entire geometry. This mesh is used only for fission source convergence diagnostics if the user does not specify any spatial mesh for this quantity.

Note: Currently, users cannot enter grid `ID`s greater than 9999.

---

ii. A cubic mesh grid (grid `ID` = 20001):

KENO generates a cubic mesh with the mesh size specified by `MSH` parameter, which is introduced in Table 8.1.1. This mesh definition can be used by all mesh-based quantities except the Shannon Entropy tallies used for fission source convergence diagnostics.

iii. Mesh with `grid geometry data` input (grid `ID` = NUMBER):

The user can specify either a mesh with a single `grid data` input block or multiple meshes by repeating the `grid data` input block with different mesh definitions. Note that the entire block, including `READ GRID` and `END GRID`, must be repeated each time; this behavior is different from all other blocks of KENO input.

After processing all grid data and constructing all grid meshes, KENO prints the specification of each mesh grid in the Grid Definitions output edit; see Sect. 8.1.5.19 for details.

Before starting the transport process, KENO tries to match the IDs specified in the parameter block with the keywords `SCD`, `MFX`, `CDS`, `CGD`, `FIS`, and `GFX` (See Sect. 8.1.3.3) to the grid `ID` in each grid definition. Each mesh-based quantity requested by the user is associated with a grid if the requested grid `ID` exists.

Note: Parameters `CDS`, `CGD`, `FIS`, `GFX`, and `MFX` with *YES* entry always require a mesh defined by either parameter `MSH` or `grid data` input.

Parameters `CDS`, `CGD`, `FIS`, `GFX`, and `MFX` with *YES* entry always use the mesh defined by the first `grid data` input if there are multiple meshes defined by both `MSH` and `grid data` input block(s).

The following examples demonstrate the mesh features in KENO:

Example: Source Convergence Diagnostics with Default Mesh KENO always generates a default mesh for source convergence diagnostics if the user does not specify any mesh for this quantity. The following example (only KENO-VI version is shown) with some updates in `parameter data` and `grid data` input blocks will be used in all examples in this section to summarize several mesh definition scenarios with several mesh-based quantity requests.

In this sample problem, a bounding box is defined to enclose the entire geometry with xmin=-13.74, xmax=13.74, ymin=-13.74, ymax=13.74, zmin=-13.01, and zmax=13.01. KENO uses this bounding box and creates a $5 \times 5 \times 5$ grid for source convergence diagnostics.

KENO-VI:

```
=kenovi
mesh test - scd with default mesh (5x5x5), scd=yes (default)
READ PARAMETER
   CEP=ce_v7.1_endf NPG=10000
END PARAMETER

read mixt
   mix=1
      92234  4.82716E-04 92235  4.47972E-02
             92236  9.57234E-05 92238  2.65768E-03
end mixt

read geometry
   unit 1
      com='single 2c8 unit centered'
      cylinder 10  5.748 5.3825 -5.3825
      cuboid   20  4p6.87 2p6.505
      media  1 1 10      vol=8938.968624
      media  0 1 20 -10  vol=10710.044784
      boundary 20
   global unit 2
      cuboid 10  4p13.74 2p13.01
      com='2x2x2  2c8 array'
      array 1 +10 place 1 1 1 2r-6.87 -6.505
      boundary  10
end geometry

read array
   ara=1 nux=2 nuy=2 nuz=2  fill f1  end fill
end array
end data
end
```

In the output, KENO notifies users about the default mesh usage for source convergence diagnostics with the message *k6-316*. The default grid definition with some details is shown in Example 8.1.34.

Fig. 8.1.88 shows the Fulcrum visualized Shannon Entropy by generations plot for this sample problem.

Example 8.1.34: Grid Definitions output edit

```
***** warning ***** keno message number k6-316 follows:
No mesh provided for Source Convergence Diagnostics (SCD). Continue with default mesh.

...

          ********************************************************************************************************************
    ***                                                                                                                  ***
    ***                                          grid definitions                                                        ***
    ***                                                                                                                  ***
    ***     Only a single grid geometry is either specified or internally setup for this problem                         ***
    ***                                                                                                                  ***
    ********************************************************************************************************************

                              **** Grid geometries utilized in this problem ****

Grid Geometry: 10001
  title: Default 5 x 5 x 5 Cartesian mesh which overlays the entire geometry
  Plane Summary
    x:    5 cells from -1.37400E+01 to  1.37400E+01
    y:    5 cells from -1.37400E+01 to  1.37400E+01
    z:    5 cells from -1.30100E+01 to  1.30100E+01
  Total number of cells: 125

                     x-planes              y-planes              z-planes
  -----  --------------------  --------------------  --------------------
      1  -1.37400137400000E+01  -1.37400137400000E+01  -1.30100130100000E+01
      2  -8.24400824399998E+00  -8.24400824399998E+00  -7.80600780599998E+00
      3  -2.74800274799999E+00  -2.74800274799999E+00  -2.60200260199999E+00
      4   2.74800274799999E+00   2.74800274799999E+00   2.60200260199999E+00
      5   8.24400824399998E+00   8.24400824399998E+00   7.80600780599998E+00
      6   1.37400137400000E+01   1.37400137400000E+01   1.30100130100000E+01
  -----  --------------------  --------------------  --------------------
```

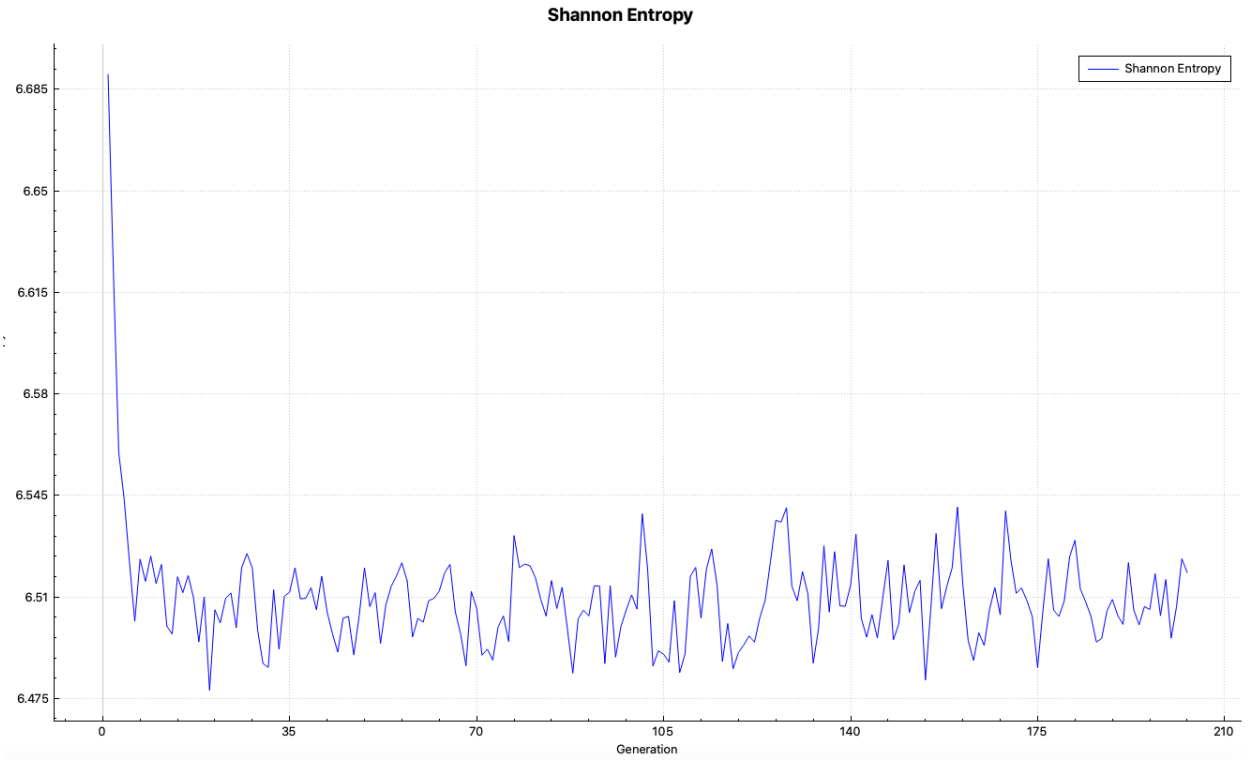


Fig. 8.1.88: Shannon entropy tallied on default mesh.

Example: Source Convergence Diagnostics with User-defined Mesh. The user can request the accumulation of the fission source on a different grid rather than the default grid for source convergence diagnostics. In this example, the sample problem specified in previous example is modified for this purpose. A new $6 \times 6 \times 6$ grid is defined over the entire geometry in a **grid geometry data** input with grid **ID** 12, and source convergence diagnostics are requested by setting **SCD** parameter to *12* in the parameter block that matches the NUMBER specification in the **grid geometry data** input.

```
=kenovi
mesh test - scd with user-defined grid

READ PARAMETERS
   CEP=ce_v7.1_endf NPG=10000 SCD=12
END PARAMETERS

...

READ GRID
   12
   TITLE "test scd with this mesh"
   NUMXCELLS=6 NUMYCELLS=6 NUMZCELLS=6
   XMIN=-13.74 XMAX=13.74
   YMIN=-13.74 YMAX=13.74
   ZMIN=-13.01 ZMAX=13.01
END GRID
...
```

Note: Starting from this example, only a summary of Grid Definiton output edit will be presented for the remaining examples.

A User-defined grid is printed in Grid Definitons output edit, and Example 8.1.35 presents only a summary of this output section. Fig. 8.1.89 shows the variation of Shannon Entropy (tallied on user-defined mesh) over generations for the same problem given in the previous example.

Example 8.1.35: Grid Definitions output edit

```
...

   Grid Geometry: 12
 title: test scd with this mesh
 Plane Summary
   x:    6 cells from -1.37400E+01 to  1.37400E+01
   y:    6 cells from -1.37400E+01 to  1.37400E+01
   z:    6 cells from -1.30100E+01 to  1.30100E+01
 Total number of cells: 216

   ...
```

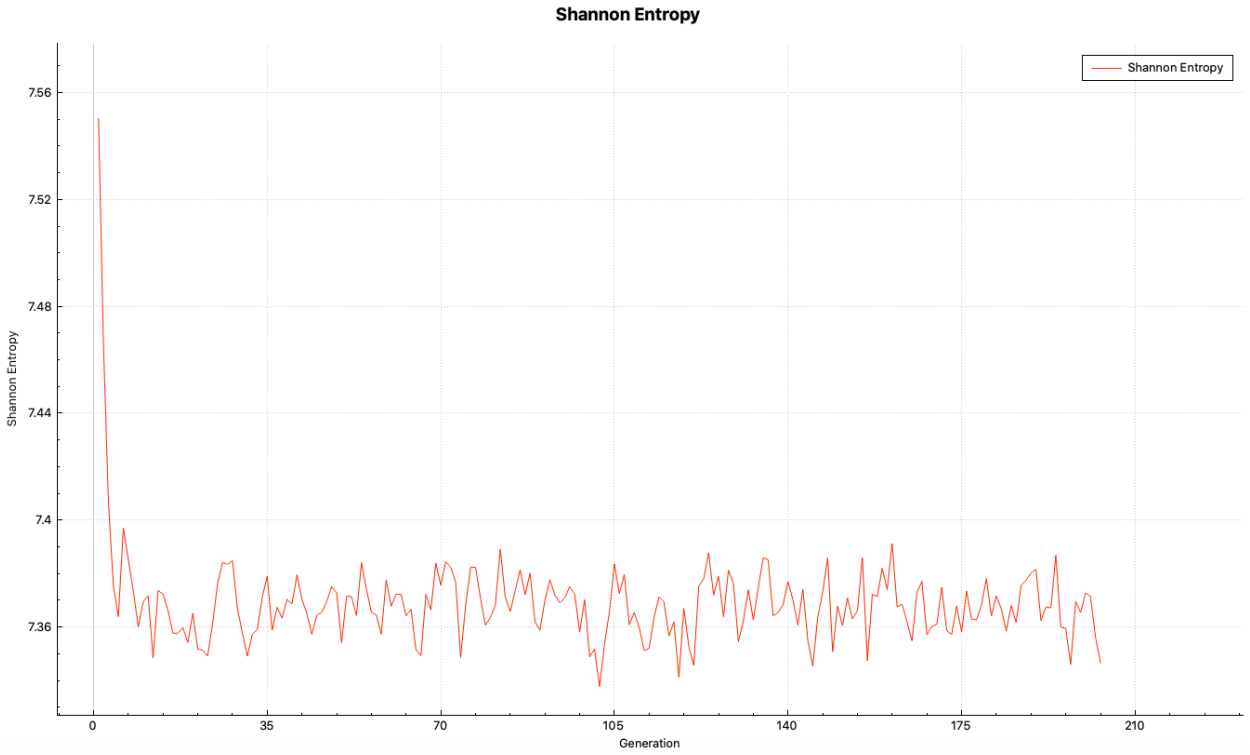


Fig. 8.1.89: Shannon entropy tallied on user-defined mesh.

Example: Source Convergence Diagnostics with Non-Existing Mesh. This example demonstrates the code behavior if there is a mismatch between the NUMBER entry in **grid geometry data** input and the **SCD** parameter in the **parameter data**. The sample input in the first example was modified as (a) a single mesh is defined with grid **ID** = 12, and (b) source convergence diagnostics is intended to use a grid with grid **ID** = 99, which does not exist.

```
=kenovi
mesh test - scd with non-existing grid id

READ PARAMETERS
```

(continues on next page)

(continued from previous page)

```
   CEP=ce_v7.1_endf NPG=10000 SCD=99
END PARAMETERS

...

READ GRID
   12
   TITLE "test scd with this mesh"
   NUMXCELLS=6 NUMYCELLS=6 NUMZCELLS=6
   XMIN=-13.74 XMAX=13.74
   YMIN=-13.74 YMAX=13.74
   ZMIN=-13.01 ZMAX=13.01
END GRID
...
```

In this example, calculation is stopped with an error message since the requested grid definition for source convergence diagnostics does not exist in the input.

```
...

***** error ***** keno message number k6-315 follows:
Mesh < 99 > specified for Shannon entropy tally and source convergence diagnostics(SCD) is not found in  input.

...

***** error ***** keno message number k6-100 follows:
this problem will not be run because errors were encountered in the input data.
...
```

Example: Fission Rate Mesh Tally with a Cubic Mesh Definition with MSH Parameter. In this example, the sample problem described in the first example was modified as (a) a cubic mesh defined with the **MSH** parameter, and (b) fission rate mesh tally is requested on this cubic mesh. In this case, a default mesh with grid **ID**=10001 is constructed and used only for source convergence diagnostics. The cubic mesh with grid **ID**=20001 is also constructed with the length of one side as equals to *2.748 cm* and is used for the fission rate mesh tally.

```
=kenovi
mesh test - fis with uniform mesh defined by MSH

READ PARAMETERS
   CEP=ce_v7.1_endf NPG=10000 FIS=yes MSH=2.748
END PARAMETERS

...
```

The output prints the details of mesh grid entries in Grid Definitions edit, as shown in Example 8.1.36. For this sample problem, the cubic mesh grid is constructed in a box that is larger than the bounding box around the global unit (+x=13.74, -x=-13.74, +y=13.74, -y=-13.74, +z=13.01 and -z=-13.01). The extents of the global boundary in each dimension are not equal; therefore, KENO tries to adjust the box size to place the cubic meshes properly inside this box. A grid mesh perfectly overlapping the problem geometry can be defined only by the **grid data** input rather than using the **MSH** parameter.

Example 8.1.36: Grid Definitions output edit

```
                          **** Grid geometries utilized in this problem ****

Grid Geometry: 20001
```

(continues on next page)

(continued from previous page)

```
      title: Cubic mesh grid automatically generated with mesh_size =   2.74799991
      Plane Summary
        x:   12 cells from -1.64880E+01 to  1.64880E+01
        y:   12 cells from -1.64880E+01 to  1.64880E+01
        z:   12 cells from -1.64880E+01 to  1.64880E+01
      Total number of cells: 1728

                          x-planes               y-planes               z-planes
      -----  --------------------  --------------------  --------------------
          1  -1.64879994392395E+01  -1.64879994392395E+01  -1.64879994392395E+01
          2  -1.37399995326996E+01  -1.37399995326996E+01  -1.37399995326996E+01

...

Grid Geometry: 10001
    title: Default 5 x 5 x 5 Cartesian mesh which overlays the entire geometry
    Plane Summary
      x:    5 cells from -1.37400E+01 to  1.37400E+01
      y:    5 cells from -1.37400E+01 to  1.37400E+01
      z:    5 cells from -1.30100E+01 to  1.30100E+01
    Total number of cells: 125

...
```

Specifications of fission rate mesh tally in Mesh Tallies output edit are listed in Example 8.1.37. Fission rate mesh tally is saved in a 3dmap file named *example4.fission_density.3dmap*, and results from this file overlayed on the geometry are shown in Fig. 8.1.90.

Example 8.1.37: Mesh tallies output edit

```
                                                  **** mesh tallies ****

  1 mesh tallies computed for this problem


   mesh tally (requested with parameter fis)
       response         : fission_density
       grid id          : 20001
       energy id        : Default
       memory allocated : 6.645 MB
       output           : example4.fission_density.3dmap

       energy boundaries:
            group   energy (eV)
           ------  ------------
                1   2.00000E+07
                2   1.73300E+07
                3   1.56800E+07
                .              .
              250   7.50000E-04
              251   5.00000E-04
              252   1.00000E-04
                    1.00000E-05
           ------  ------------

       grid summary:
             x:   12 cells from -1.64880E+01 to  1.64880E+01
             y:   12 cells from -1.64880E+01 to  1.64880E+01
             z:   12 cells from -1.64880E+01 to  1.64880E+01
           Total number of cells: 1728
```

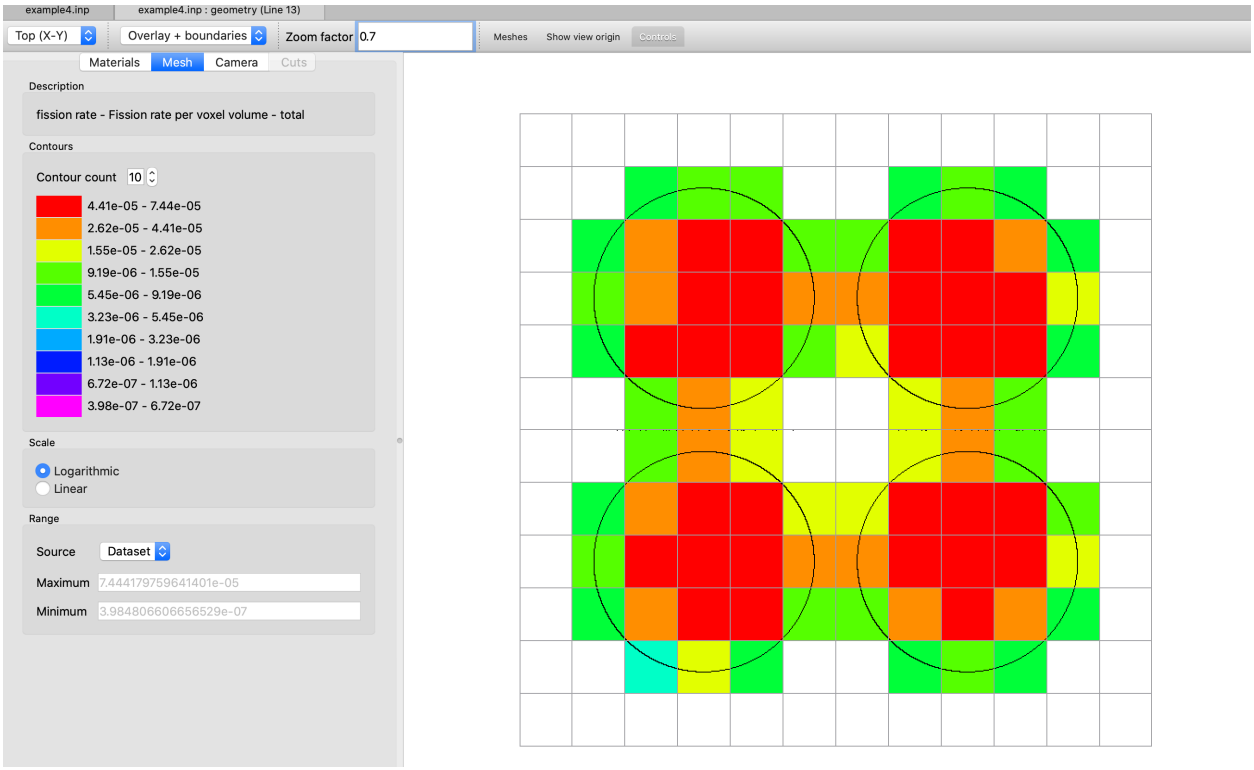


Fig. 8.1.90: Fission rate mesh tally computed on a uniform mesh defined by **MSH=2.748** (X-Y view at z=5.0).

Example: Fission Rate Mesh Tally with a NUMBER. In this example, problem given in the first example was modified to request a fission rate mesh tally calculation over a mesh defined in the **grid geometry data** input. A default mesh with grid **ID**=10001 is constructed and used for the source convergence diagnostics. The user defined mesh (grid **ID**=17) is used for the fission rate mesh tally.

```
=kenovi
mesh test - fis with a mesh defined by grid data block

READ PARAMETERS
   CEP=ce_v7.1_endf NPG=10000 FIS=17
END PARAMETERS
...

READ GRID
   17
   TITLE "test fis with this mesh"
   NUMXCELLS=20 NUMYCELLS=20 NUMZCELLS=6
   XMIN=-13.74 XMAX=13.74
   YMIN=-13.74 YMAX=13.74
   ZMIN=-13.01 ZMAX=13.01
END GRID
...
```

A summary of the mesh grids utilized in this sample problem is printed in Grid Definitions, as shown in Example 8.1.38.

Example 8.1.38: Grid Definitions output edit

```
                              **** Grid geometries utilized in this problem ****

Grid Geometry: 17
    title: test fis with this mesh
    Plane Summary
      x:   20 cells from -1.37400E+01 to  1.37400E+01
      y:   20 cells from -1.37400E+01 to  1.37400E+01
      z:    6 cells from -1.30100E+01 to  1.30100E+01
    Total number of cells: 2400

...

Grid Geometry: 10001
    title: Default 5 x 5 x 5 Cartesian mesh which overlays the entire geometry
    Plane Summary
      x:    5 cells from -1.37400E+01 to  1.37400E+01
      y:    5 cells from -1.37400E+01 to  1.37400E+01
      z:    5 cells from -1.30100E+01 to  1.30100E+01
    Total number of cells: 125
```

Example 8.1.39 is a snapshot of the Mesh Tallies output edit, which summarizes the specification of this mesh quantity. The fission rate mesh tally is saved in a 3dmap file named *example5.fission_density.3dmap*. Tally results from this file overlayed over the geometry are shown in Fig. 8.1.91.

Example 8.1.39: Mesh tallies output edit

```
                                                  **** mesh tallies ****

  1 mesh tallies computed for this problem


   mesh tally (requested with parameter fis)
       response          : fission_density
       grid id           : 17
       energy id         : Default
       memory allocated  : 9.229 MB
       output            : example5.fission_density.3dmap

       energy boundaries:
            group   energy (eV)
           ------  ------------
                1   2.00000E+07
                2   1.73300E+07
                3   1.56800E+07
                .             .
              250   7.50000E-04
              251   5.00000E-04
              252   1.00000E-04
                    1.00000E-05
           ------  ------------

       grid summary:
             x:   20 cells from -1.37400E+01 to  1.37400E+01
             y:   20 cells from -1.37400E+01 to  1.37400E+01
             z:    6 cells from -1.30100E+01 to  1.30100E+01
           Total number of cells: 2400
```

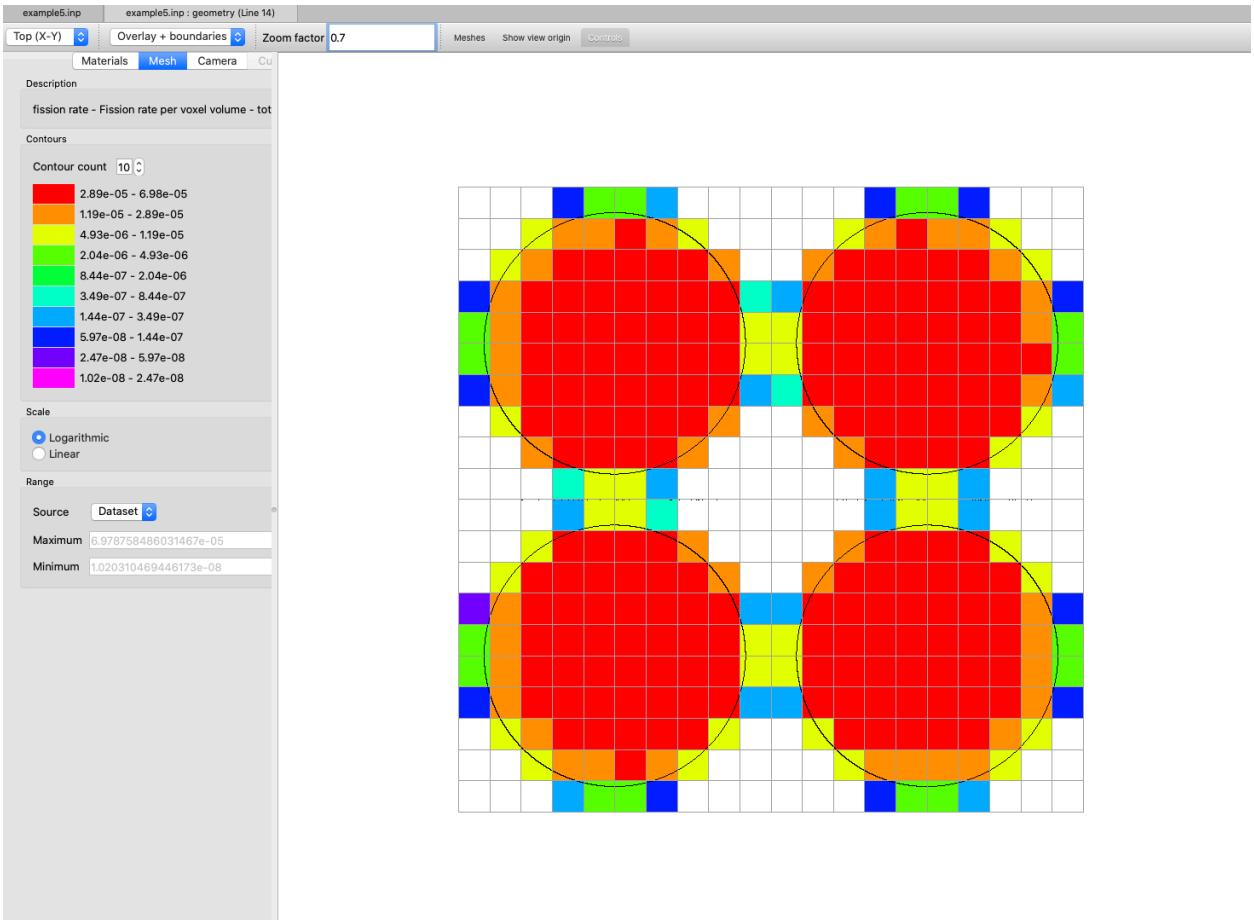


Fig. 8.1.91: Fission rate mesh tally computed on a mesh defined by **grid geometry data** input (X-Y view at z=5.0).

Example: Fission Rate Mesh Tally with Multiple Mesh Definitions. In this example, the sample input from the previous example is used with the old-style definition so that fission rate mesh tally is requested by setting the **FIS** parameter to *YES*.

Additional mesh grids were added to the sample input to demonstrate how KENO assigns a mesh grid to a mesh-based quantity when the value of mesh quantity has been entered as *YES* and multiple mesh grids have already been defined: (a) two mesh grids with grid **IDs** *118* and *113* are defined in **grid geometry data** inputs, and (b) another mesh grid is also defined with the **MSH** parameter.

KENO uses the default mesh grid (grid **ID**=10001) for the source convergence diagnostics, using the mesh with grid **ID**= 118 from the grid data container for the fission rate mesh tally.

```
=kenovi
mesh test - fis=yes with multiple mesh defined by grid data block

READ PARAMETERS
   CEP=ce_v7.1_endf NPG=10000 MSH=2.5 FIS=yes
END PARAMETERS
...

READ GRID
   118
```

(continues on next page)

(continued from previous page)

```
   TITLE "test fis with this mesh partially covering the geometry"
   NUMXCELLS=20 NUMYCELLS=20 NUMZCELLS=18
   XMIN=-1.74 XMAX=13.74
   YMIN=-1.74 YMAX=13.74
   ZMIN=-1.01 ZMAX=13.01
END GRID

READ GRID
   113
   TITLE "mesh will not be used"
   NUMXCELLS=3 NUMYCELLS=3 NUMZCELLS=3
       XMIN=-1.74 XMAX=13.74
       YMIN=-1.74 YMAX=13.74
       ZMIN=-1.01 ZMAX=13.01
END GRID

READ GRID
   17
   TITLE "test fis with this mesh"
   NUMXCELLS=20 NUMYCELLS=20 NUMZCELLS=6
   XMIN=-13.74 XMAX=13.74
   YMIN=-13.74 YMAX=13.74
   ZMIN=-13.01 ZMAX=13.01
END GRID


...
```

A summary of Grid Definitions output edit is given in Example 8.1.40. Note that KENO does not print anything for the cubic mesh (requested by setting **MSH=2.5**, grid **ID**= 20001) because this mesh was not constructed. KENO ignores the mesh definition request done with the **MSH** parameter if any mesh grid is defined by **grid data** input.

Example 8.1.40: Grid Definitions output edit

```
                          **** Grid geometries utilized in this problem ****

 Grid Geometry: 118
    title: test fis with this mesh partially covering the geometry
    Plane Summary
      x:   20 cells from -1.74001E+00 to  1.37400E+01
      y:   20 cells from -1.74001E+00 to  1.37400E+01
      z:   18 cells from -1.01001E+00 to  1.30100E+01
    Total number of cells: 7200

...

 Grid Geometry: 113
     title: mesh will not be used
     Plane Summary
       x:    3 cells from -1.74001E+00 to  1.37400E+01
       y:    3 cells from -1.74001E+00 to  1.37400E+01
       z:    3 cells from -1.01001E+00 to  1.30100E+01
     Total number of cells: 27

...

Grid Geometry: 17
    title: test fis with this mesh
    Plane Summary
      x:   20 cells from -1.37400E+01 to  1.37400E+01
      y:   20 cells from -1.37400E+01 to  1.37400E+01
```

(continues on next page)

(continued from previous page)

```
      z:    6 cells from -1.30100E+01 to  1.30100E+01
    Total number of cells: 2400

...

 Grid Geometry: 10001
     title: Default 5 x 5 x 5 Cartesian mesh which overlays the entire geometry
     Plane Summary
       x:    5 cells from -1.37400E+01 to  1.37400E+01
       y:    5 cells from -1.37400E+01 to  1.37400E+01
       z:    5 cells from -1.30100E+01 to  1.30100E+01
     Total number of cells: 125

...
```

KENO assigns the first grid entry from the grid data container to any mesh quantity whose value is entered as *YES* in the **parameter data** input. Each grid defined in the grid data blocks is stored in the grid data container unit in the order in which it is processed.

In this example, the grid entry *118*, which is processed and stored first, is assigned to the fission rate mesh tally calculation. Mesh Tallies output edit shown in Example 8.1.41 reflects this grid assignment to the fission ate mesh tally.

Example 8.1.41: Mesh tallies output edit

```
                                        **** mesh tallies ****

  1 mesh tallies computed for this problem


   mesh tally (requested with parameter fis)
       response         : fission_density
       grid id          : 118
       energy id        : Default
       memory allocated : 27.686 MB
       output           : example6.fission_density.3dmap

       energy boundaries:
            group   energy (eV)
           ------  ------------
                1   2.00000E+07
                2   1.73300E+07
                3   1.56800E+07
                .              .
              250   7.50000E-04
              251   5.00000E-04
              252   1.00000E-04
                    1.00000E-05
           ------  ------------

       grid summary:
             x:   20 cells from -1.74001E+00 to  1.37400E+01
             y:   20 cells from -1.74001E+00 to  1.37400E+01
             z:   18 cells from -1.01001E+00 to  1.30100E+01
           Total number of cells: 7200

...
```

Fig. 8.1.92 illustrates the fission rate mesh tally overlayed on the geometry. Note that the mesh used in the fission rate mesh tally calculation does not cover the entire geometry, and this is not a requirement for this tally.

Note: The fission rate mesh tally capability as well as fission source and grid flux tallies (averaged over mesh volumes) do not require that the mesh grid used in the calculation cover the entire geometry.

However, the mesh flux capability (calculated over the volume of materials in each mesh voxel) activated by the **MFX** parameter always requires that the mesh grid used in its calculations cover the entire geometry.

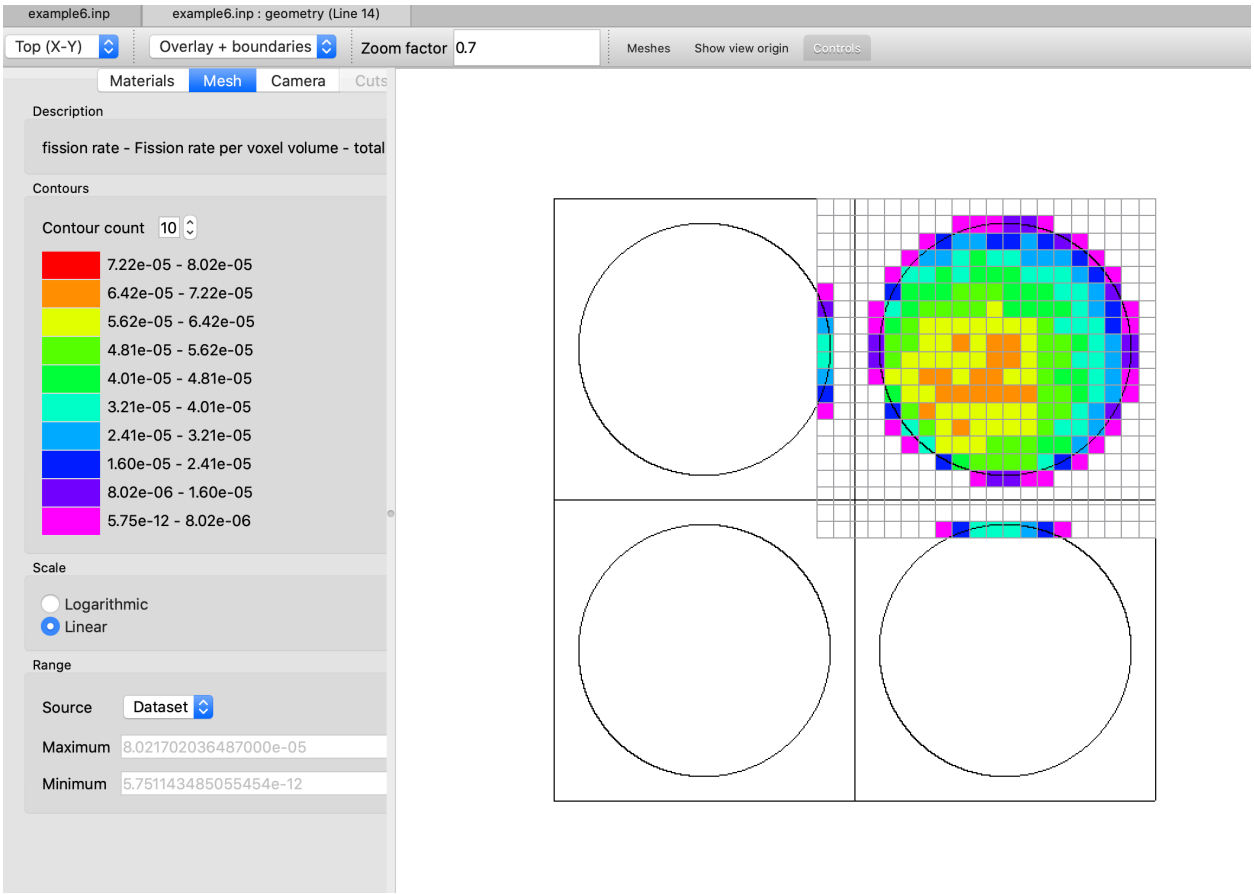


Fig. 8.1.92: Fission rate mesh tally computed over a mesh grid defined by **grid geometry data** input. Defined mesh grid with **ID**= 118 partially covers the geometry (X-Y view at z=6.8).

Example: Mesh Flux Tally with Multiple Mesh Definitions. In this example, the sample problem described in previous example is modified to request a mesh flux calculation (**MFX**) on a mesh grid which partly covers the problem geometry. The mesh volume calculation required by the mesh flux calculation is already requested by adding a volume block to the input.

Note: Calculation of mesh fluxes in all regions enclosed by a mesh voxel requires the volume of each region in this mesh voxel. KENO V.a automatically activates stochastic volume calculation with RANDOM volume estimate for such a case. The user can override the parameters in the **volume data** input block. However, users cannot change the volume calculation method from RANDOM volume estimate to TRACE volume estimate.

In the KENO-VI case, it is the user's responsibility to add a **volume data** input block to activate

the stochastic mesh volume calculation. Like KENO V.a, KENO-VI only allows RANDOM volume estimate for mesh volume calculation. See Sect. 8.1.3.13 and Sect. 8.1.5.17 for more details.

After these updates, KENO with this sample problem uses the default mesh (grid **ID**=10001) for the source convergence diagnostics, and the user-defined mesh with grid **ID**=118 for accumulating the mesh fluxes. Unlike the fission rate mesh tally case discussed in the previous example, execution is terminated for this scenario since the mesh flux capability requested with **MFX** parameter requires a mesh grid covering the entire geometry.

```
=kenovi
mesh test - mfx=yes with a grid mesh covering the part of the geometry

READ PARAMETERS
   CEP=ce_v7.1_endf NPG=10000 MFX=yes
END PARAMETERS

...

READ GRID
   118
   TITLE "test fis with this mesh partially covering the geometry"
   NUMXCELLS=20 NUMYCELLS=20 NUMZCELLS=18
   XMIN=-1.74 XMAX=13.74
   YMIN=-1.74 YMAX=13.74
   ZMIN=-1.01 ZMAX=13.01
END GRID

READ GRID
   113
   TITLE "mesh will not be used"
   NUMXCELLS=3 NUMYCELLS=3 NUMZCELLS=3
       XMIN=-1.74 XMAX=13.74
       YMIN=-1.74 YMAX=13.74
       ZMIN=-1.01 ZMAX=13.01
END GRID

READ GRID
   17
   TITLE "test fis with this mesh"
   NUMXCELLS=20 NUMYCELLS=20 NUMZCELLS=6
   XMIN=-13.74 XMAX=13.74
   YMIN=-13.74 YMAX=13.74
   ZMIN=-13.01 ZMAX=13.01
END GRID

...

READ VOLUME
  TYPE=RANDOM BATCHES=100 POINTS=1000
  XP=+13.01  XM=-13.01
  YP=+13.74  YM=-13.74
  ZP=+13.74  ZM=-13.74
END VOLUME

...
```

The code is stopped with the following error message in the output:

```
...

***** warning ***** keno message number k6-316 follows:
No mesh provided for Source Convergence Diagnostics (SCD). Continue with default mesh.
```

(continues on next page)

(continued from previous page)

```
 1                                        mesh test - fis  with multiple grid data input

                              volumes for those units utilized in this problem

                     Mesh Volume Sampling Parameters
                     -------------------------------

                     The number of points per batch was specified as 1000
                     This gives a sampling density of  5.08931E-02 points per cc per batch.
                     The number of batches is 100

 ***** error ***** keno message number k6-305 follows:
 Mesh fluxes have been specified, but the mesh does not completely cover the geometry.
 The point x=  3.65030E+00 y=  1.90999E+00 z= -1.02504E+01 lies outside the mesh.
 The problem will not be run.  Fix the mesh and resubmit the case.

 ...

  ***** error ***** keno message number k6-100 follows:
  this problem will not be run because errors were encountered in the input data.

 stop code        129
    execution terminated due to errors. completion code 129. cpu time used 3.15388 seconds
 ***Error: Application failed - see messages file for details.

...
```

Example: Multiple Mesh Quantity with Multiple Mesh Definitions. This example demonstrates the multiple mesh definition for multiple mesh-based quantities. Our sample problem from a previous example is modified as (a) the source convergence diagnostics and mesh fluxes are requested with (grid *ID*= 17), (b) grid fluxes are requested with (grid *ID*= 113), and (c) fission rate mesh tally is requested with *YES*.

The first grid entry (grid *ID*=118) in **grid geometry data** input is assigned to the fission rate mesh tally.

```
=kenovi
mesh test - test mfx, gfx, and fis together

READ PARAMETERS
   CEP=ce_v7.1_endf NPG=10000 MFX=17 scd=17 gfx=113 fis=yes pms=yes
END PARAMETERS

...

READ GRID
   118
   TITLE "test fis with this mesh partially covering the geometry"
   NUMXCELLS=20 NUMYCELLS=20 NUMZCELLS=18
   XMIN=-1.74 XMAX=13.74
   YMIN=-1.74 YMAX=13.74
   ZMIN=-1.01 ZMAX=13.01
END GRID

READ GRID
   113
   TITLE "mesh will not be used"
   NUMXCELLS=3 NUMYCELLS=3 NUMZCELLS=3
       XMIN=-1.74 XMAX=13.74
       YMIN=-1.74 YMAX=13.74
       ZMIN=-1.01 ZMAX=13.01
END GRID
```

(continues on next page)

(continued from previous page)

```
READ GRID
   17
   TITLE "test fis with this mesh"
   NUMXCELLS=20 NUMYCELLS=20 NUMZCELLS=6
   XMIN=-13.74 XMAX=13.74
   YMIN=-13.74 YMAX=13.74
   ZMIN=-13.01 ZMAX=13.01
END GRID

...

READ VOLUME
  TYPE=RANDOM BATCHES=100 POINTS=1000
  XP=+13.01  XM=-13.01
  YP=+13.74  YM=-13.74
  ZP=+13.74  ZM=-13.74
END VOLUME

...
```

All three user-defined mesh grids and default mesh grids constructed by KENO are printed in the Grid Definitions output edit as shown in Example 8.1.42. In this output edit, a summary of the volume calculation results performed on the mesh grid *17* is also printed as part of the grid geometry 17 output edit. Further details on the mesh volume calculation can be seen in the volume information edit in the output. All mesh volumes can be printed by setting **PMV=YES** in the **parameter data** input (see Table 8.1.1).

Example 8.1.42: Grid Definitions output edit

```
                          **** Grid geometries utilized in this problem ****

Grid Geometry: 118
    title: test fis with this mesh partially covering the geometry
    Plane Summary
      x:   20 cells from -1.74001E+00 to  1.37400E+01
      y:   20 cells from -1.74001E+00 to  1.37400E+01
      z:   18 cells from -1.01001E+00 to  1.30100E+01
    Total number of cells: 7200
...

Grid Geometry: 113
   title: mesh will not be used
   Plane Summary
     x:    3 cells from -1.74001E+00 to  1.37400E+01
     y:    3 cells from -1.74001E+00 to  1.37400E+01
     z:    3 cells from -1.01001E+00 to  1.30100E+01
   Total number of cells: 27
...

Grid Geometry: 17
   title: test fis with this mesh
   Plane Summary
     x:   20 cells from -1.37400E+01 to  1.37400E+01
     y:   20 cells from -1.37400E+01 to  1.37400E+01
     z:    6 cells from -1.30100E+01 to  1.30100E+01
   Total number of cells: 2400

                       x-planes               y-planes               z-planes
   -----  ---------------------  ---------------------  ---------------------
       1  -1.37400137400000E+01  -1.37400137400000E+01  -1.30100130100000E+01
       2  -1.23660000000000E+01  -1.23660000000000E+01  -8.67333333333333E+00
...
      20   1.23660000000000E+01   1.23660000000000E+01
```

(continues on next page)

(continued from previous page)

```
     21   1.37400137400000E+01   1.37400137400000E+01
  -----  --------------------  --------------------  --------------------
...

  Region volumes in each mesh voxel have been calculated.
    Total number of volume elements   : 7468
    Number of non-zero volume elements: 3734
  Mesh/region pairs indexed together to speed up mesh flux scoring.
```

KENO prints the summary of the specifications of the mesh quantities computed in Mesh tallies output edit as shown in Example 8.1.43. The code saves the fission rate mesh tally and grid fluxes in two separate 3dmap files named *example8.fission_density.3dmap* and *example8.flux.3dmap*, respectively. Mesh fluxes are also saved in an ASCII file since the **PMS** parameter is set to *YES*. Fig. 8.1.93 presents the fission rate mesh tally, grid fluxes, and Shannon entropy computed on the meshes defined in **grid geometry data** input.

Example 8.1.43: Mesh tallies output edit

```
...
                                                  **** mesh tallies ****

 2 mesh tallies computed for this problem


  mesh tally (requested with parameter gfx)
      response          : flux
      grid id           : 113
      energy id         : Default
      memory allocated  : 0.104 MB
      output            : example8.flux.3dmap

      energy boundaries:
           group   energy (eV)
          ------  ------------
               1   2.00000E+07
               2   1.73300E+07
               3   1.56800E+07
               .              .
             250   7.50000E-04
             251   5.00000E-04
             252   1.00000E-04
                   1.00000E-05
          ------  ------------

      grid summary:
            x:    3 cells from -1.74001E+00 to  1.37400E+01
            y:    3 cells from -1.74001E+00 to  1.37400E+01
            z:    3 cells from -1.01001E+00 to  1.30100E+01
          Total number of cells: 27


  mesh tally (requested with parameter fis)
      response          : fission_density
      grid id           : 118
      energy id         : Default
      memory allocated  : 27.686 MB
      output            : example8.fission_density.3dmap

      energy boundaries:
           group   energy (eV)
          ------  ------------
               1   2.00000E+07
               2   1.73300E+07
```

(continues on next page)

(continued from previous page)

```
                3   1.56800E+07
                .             .
              250   7.50000E-04
              251   5.00000E-04
              252   1.00000E-04
                    1.00000E-05
            ------  ------------

       grid summary:
             x:   20 cells from -1.74001E+00 to  1.37400E+01
             y:   20 cells from -1.74001E+00 to  1.37400E+01
             z:   18 cells from -1.01001E+00 to  1.30100E+01
           Total number of cells: 7200

...
```

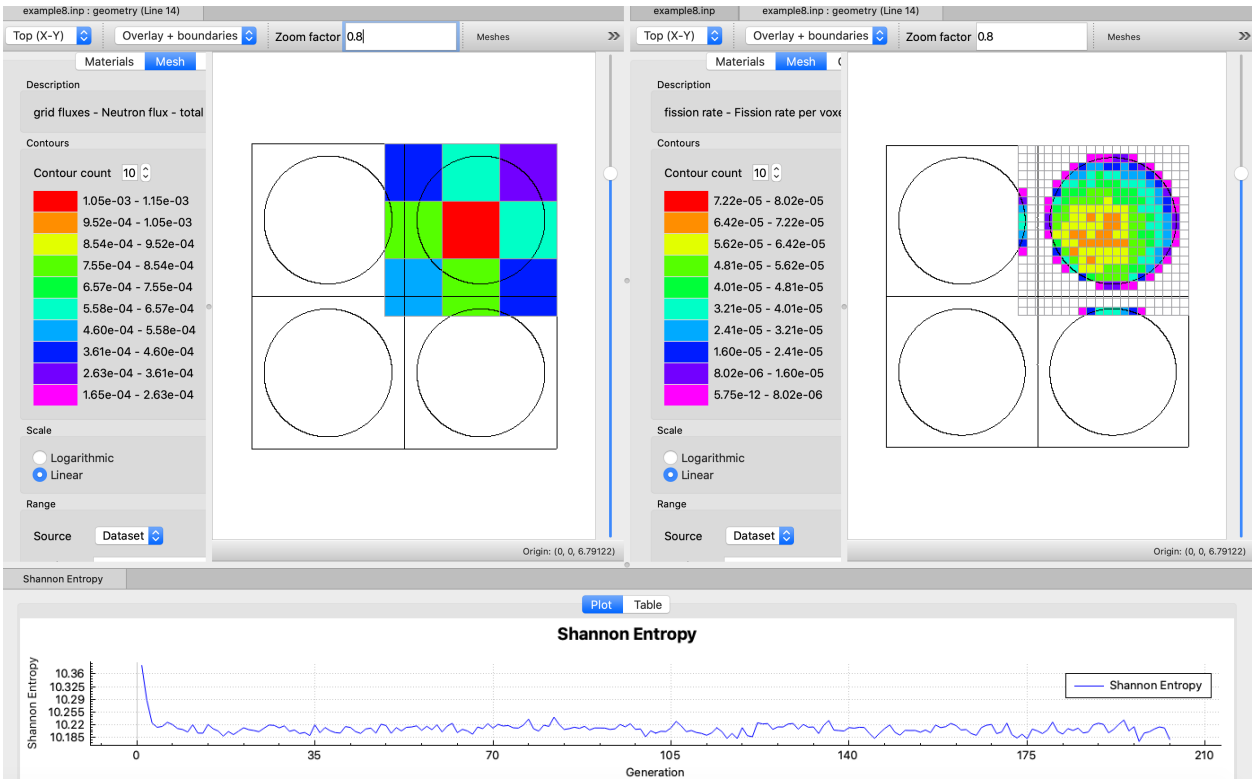


Fig. 8.1.93: Fission rate mesh tally, grid fluxes, and Shannon entropy computed over the different meshes defined by **grid geometry data** input.

8.1.4.12 Random sequence

The random-number package used by KENO always starts with the same seed and thus always reproduces the same sequence of random numbers. Any random number except the one printed as the starting random number in the parameter table can be used to activate a different random sequence. The user can rerun a problem with a different random sequence by simply entering a hexadecimal random number other than the starting random number in the parameter data. For example, by entering RND=A10C1893E6D5 in the parameter data, the problem will be run with a different random sequence.

8.1.4.13 Matrix k-effective

Matrix k-effective calculations provide an alternative method of calculating the k-effective of the system. Cofactor k-effectives and source vectors are additional information that can be provided when the matrix k-effective is calculated. The necessary source and fission weight data are collected during the neutron tracking procedure. This information is converted to a FISSION PRODUCTION MATRIX, which is the number of next generation neutrons produced at J by a neutron born at I. The principal eigenvalue of the fission probability matrix is the matrix k-effective. KENO offers four alternatives when calculating matrix k-effective as discussed below:

(1) If MKP=YES is specified in the parameter data, the fission production matrix is collected by array position or position index in the GLOBAL ARRAY. The position index is used to reference a given location in a 3-D lattice. For a $2 \times 2 \times 2$ array, there are nine unique position indices as shown below. Position zero contains everything outside the GLOBAL ARRAY.

Table 8.1.30: Array index guide for a sample $2 \times 2 \times 2$ array

	POSITION		
	X	Y	Z
0	0	0	0
1	1	1	1
2	2	1	1
3	1	2	1
4	2	2	1
5	1	1	2
6	2	1	2
7	1	2	2
8	2	2	2

The fission production matrix is the number of next generation neutrons produced at index J by a neutron born at index I. This matrix is used to calculate the matrix k-effective, cofactor k-effectives and the source vector by position index. Because the size of the fission probability matrix is the square of the array size (for a $4 \times 4 \times 4$ array there are 4,096 entries), it can use vast amounts of computer memory.

(2) If MKU=YES is specified in the parameter data, the fission production matrix is collected by UNIT. It is the number of next generation neutrons produced in UNIT J by a neutron born in UNIT I. This matrix is used to calculate the matrix k-effective, cofactor k-effectives and source vector by UNIT.

(3) If MKH=YES is specified in the parameter data, the fission production matrix is collected by the HOLE number. Matrix information can be collected at either the highest HOLE nesting level (first level of nesting) or the deepest HOLE nesting level. HHL=YES specifies that the matrix information will be collected at the first nesting level. By default, the matrix information is collected at the deepest nesting level. The fission production matrix is the number of next generation neutrons produced in HOLE J by a neutron born in HOLE I. This matrix is used to calculate the matrix k-effective, cofactor k-effectives and the source vector by HOLE.

(4) If MKA=YES is specified in the parameter data, the fission production matrix is collected by ARRAY number. It can be collected at the highest ARRAY level (first level of nesting) or at the deepest ARRAY level. HAL=YES specifies that the matrix information will be collected at the first nesting level. By default, the matrix information is collected at the deepest nesting level. The fission production matrix is the number of next generation neutrons produced in ARRAY J by a neutron born in ARRAY I. This matrix is used to calculate the matrix k-effective, cofactor k-effectives and the source vector by ARRAY.

The user can simultaneously implement all methods of calculating the matrix k-effective. The results are labeled in the printout. Matrix k-effectives cannot be calculated for a single unit problem. If the user wishes to do so, the geometry description must have a cube or cuboid as its outer region, and the problem description should include `READ ARRAY END ARRAY`. These two actions convert the single unit problem into a $1 \times 1 \times 1$ array.

A cofactor k-effective is the eigenvalue of the fission production matrix reduced by the row and column that references the specified `UNIT` or position index. The difference between the k-effective for the system and the cofactor k-effective for a `UNIT` or position index is an indication of the *in situ* k-effective of that `UNIT` or the contribution that `UNIT` makes to the k-effective of the system. The cofactor k-effective of a `UNIT` devoid of fissile material should approximate the k-effective of the system.

8.1.4.14 Deviations

When a deviation is calculated by KENO, it is the standard deviation of the mean. This assumes a large sample having a normal distribution. KENO calculates the real variance using an iterative approach and lag covariance data between generations as follows [BDL09, Dem99a]

1. The sample variance and covariance estimates are calculated.
2. The apparent variance is set equal to the sample variance and the apparent covariance is set equal to the sample covariance.
3. The real covariance is set equal to the apparent covariance and the real variance is calculated.
4. Using the real variance and apparent covariance calculate the real covariance.
5. The real variance is recalculated.
6. Steps 4 and 5 are repeated until the real variance converges within a preset tolerance.

The covariance estimates are only calculated for the previous 20 generations. A maximum of 50 iterations are allowed for the real variance to converge.

8.1.4.15 Generation time and lifetime

The generation time and lifetime calculations use the average velocity. The validity of these calculations is determined by how accurately the average velocity represents the spectrum over the range of the energy group. The lifetime and generation time calculated by KENO are not kinetics parameters. The lifetime is the average life span of a neutron (in seconds) from the time it is born until it is absorbed or leaks from the system. The generation time is the average time (in seconds) between successive neutron generations.

8.1.4.16 Energy of the Average Lethargy of Fission

The energy of the average lethargy of neutrons causing fission (*EALF*) is a parameter calculated in KENO to characterize the neutron energy spectrum or fastness of a system. The *EALF* is given in units of eV in the KENO output. An *EALF* value that is high (> 100 keV) indicates that most fissions in the system are being caused by fast neutrons, and an *EALF* value that is low (< 1 eV) indicates that most fissions are induced by thermal neutrons.

The *EALF* is calculated by determining the lethargy, u, a measure of how much neutron energy changes from its initial or birth energy:

$$u = ln\left(\frac{E_0}{E}\right), \tag{8.1.1}$$

where E is the energy of the neutron colliding with a nucleus and E_0 is the maximum possible energy of fission neutrons (assumed in KENO to be 10 MeV). The average lethargy of all fission events is calculated by weighting the lethargy for each collision by the probability that the collision will create a fission event, and averaging this quantity. Using a log-scale parameter like lethargy to represent the fastness of systems is more convenient than directly averaging the energy of neutrons causing fission both because of the wide range of neutrons in a problem (potentially more than seven orders of magnitude), and because fast neutrons that are slowing down lose about the same fraction of their energy during each collision.

$$EALF = E_o e^{-u_{\text{avg}}} \tag{8.1.2}$$

Equation Eq. (8.1.2) is then applied again to transform the average lethargy of neutrons causing fission into a neutron energy that corresponds to the average lethargy of neutrons causing fission (this is the *EALF*); this transformation back to units of energy is done because energy has a much more intuitive meaning than units of lethargy.

8.1.5 DESCRIPTION OF OUTPUT

This section contains a brief description and explanation of the KENO output. Portions of the printout will not be printed for every problem. Some printout is optional, as noted in this section. This section provides representative samples of the output format. The actual data contained in this section are not necessarily consistent with results computed by the current version of KENO.

KENO offers an HTML output format including a series of files that can be viewed in a standard web browser. The HTML formatted output offers interactive output that is easy to read and navigate. Many of the tables of data can be sorted in ascending or descending order by clicking on the heading of the column for which sorting is desired. In this section, the standard text output description is followed by a description of the optional HTML formatted output. The HTML formatted output can be deactivated by entering HTM=NO in the parameter data section.

Warning: The HTML-formatted output is no longer actively developed, and the output files created with it may therefore not include recently added capabilities or changes. For example, the fission rate mesh tally and some CE TSUNAMI edits are not included in the HTML output.

8.1.5.1 Program verification information

Program verification information Fig. 8.1.94 is printed after the header page. It lists the name of the program, the date the load module was created, the library that contains the load module, the computer code name from the configuration control table, and the revision number. The job name, date, and time of execution are also printed. This information may be used for quality assurance purposes.

```
*****************************************************************************************************************************
*****************************************************************************************************************************
*****************************************************************************************************************************
*****                                                                                                                       *****
*****                                      program verification information                                                 *****
*****                                                                                                                       *****
*****                                    code system:  SCALE    version:  6.3                                               *****
*****                                                                                                                       *****
*****************************************************************************************************************************
*****************************************************************************************************************************
*****                                                                                                                       *****
*****                                                                                                                       *****
*****              program:  kenovi                                                                                         *****
*****                                                                                                                       *****
*****        creation date:  03_nov_2021                                                                                    *****
*****                                                                                                                       *****
*****              library:  /ornldev/code/Scale/S63/INSTALL/G_63_b1129/bin                                                 *****
*****                                                                                                                       *****
*****                                                                                                                       *****
*****            test code:  kenovi                                                                                         *****
*****                                                                                                                       *****
*****              version:  6.3                                                                                            *****
*****                                                                                                                       *****
*****              jobname:  5kq                                                                                            *****
*****                                                                                                                       *****
*****         machine name:  kbb-mac                                                                                        *****
*****                                                                                                                       *****
*****    date of execution:  04_nov_2021                                                                                    *****
*****                                                                                                                       *****
*****    time of execution:  15:19:58.73                                                                                    *****
*****                                                                                                                       *****
*****                                                                                                                       *****
*****************************************************************************************************************************
*****************************************************************************************************************************
*****************************************************************************************************************************
```

Fig. 8.1.94: Sample program verification table.

The program verification information is the first page shown in the KENO HTML output, after selecting KENO from the SCALE HTML index page. This page can also be displayed by selecting the *Program Verification Information* link under the *General Information* submenu and is shown in Fig. 8.1.94.

8.1.5.2 Tables of parameter data

The first two tables printed by KENO list the numeric parameters and logical parameters used in the problem. The user should always verify that the parameter data block was entered as desired. An example of numeric parameters table is shown in Fig. 8.1.95. An example of the logical parameters table is shown in Fig. 8.1.96.

```
*** sample problem 18   1f27  critical experiment ***

***                ******       numeric parameters       ******                ***

***                tme         maximum problem time (min)                  0.00  ***
***                tba         time per generation (min)                  10.00  ***
***                gen         number of generations                       103   ***
***                npg         number per generation                      1000   ***
***                nsk         number of generations to be skipped           3   ***
***                beg         beginning generation number                   1   ***
***                res         generations between checkpoints               0   ***
***                x1d         number of extra 1-d cross sections            1   ***
***                nbk         neutron bank size                          1025   ***
***                xnb         extra positions in neutron bank               0   ***
***                nfb         fission bank size                          1000   ***
***                xfb         extra positions in fission bank               0   ***
***                sig         cut off standard deviation                0.0000  ***
***                wta         default value of weight average           0.5000  ***
***                wth         weight high for splitting                 3.0000  ***
***                wtl         weight low for russian roulette           0.3333  ***
***                rnd         starting random number           0000F12C09ED2195 ***
***                nb8         number of d.a. blocks on unit  8           1000   ***
***                nl8         length of d.a. blocks on unit  8            512   ***
***                nqd         quadrature order for angular fluxes           0   ***
***                pnm         highest order of flux moments                 0   ***
***                msh         mesh size for mesh flux tally             0.0000  ***
***                adj         mode of calculation                      forward  ***
***                tps         sampling sites per track length               5   ***
***                cgs         number of secondary groups to sampl           0   ***
***                cas         number of secondary angles to sampl           0   ***
***                wtc         Woodcock tracking cut-off value            0.90   ***
***                ttl         temperature tolerance for CE xsecs        -1.00   ***
***                            input data written on restart unit           no   ***
***                dbl         Low Energy Cutoff (in eV) for DBRC         0.40   ***
***                dbh         High Energy Cutoff (in eV) for DBRC      210.00   ***
***                cet         CE-TSUNAMI sensitivity method              NONE   ***
***                cfp         Number of latent gens in CE-TSUNAMI           5   ***
***                dbr         DBRC isotope selection                        0   ***
***                dbx         Doppler Broadening Method                     2   ***
***                thc         Thermal Energy Cutoff                     10.00   ***
***                ngp         no. of energy groups for tallying           252   ***
```

Fig. 8.1.95: Sample table of numeric parameter data.

```
*****************************************************************************************************************
***                                                                                                           ***
***                              sample problem 18   1f27  critical experiment                              ***
***                                                                                                           ***
*****************************************************************************************************************
***                                  ******      logical parameters        ******                             ***
***                                                                                                           ***
***  run  execute problem after checking data  yes            plt  plot picture map(s)                  yes   ***
***                                                                                                           ***
***       compute fluxes (cfx, flx, cfe or mfp yes            fdn  compute fission densities            yes   ***
***                                                                                                           ***
***  smu  compute avg unit self-multiplication  no            nub  compute nu-bar & avg fission group    yes   ***
***                                                                                                           ***
***  mku  compute matrix k-eff by unit number  yes            mkp  compute matrix k-eff by unit location  no   ***
***                                                                                                           ***
***  cku  compute cofactor k-eff by unit numbe yes            ckp  compute cofactor k-eff by unit location  no  ***
***                                                                                                           ***
***  fmu  print fiss prod matrix by unit numbe yes            fmp  print fiss prod matrix by unit location  no  ***
***                                                                                                           ***
***  mkh  compute matrix k-eff by hole number   no            mka  compute matrix k-eff by array number   yes  ***
***                                                                                                           ***
***  ckh  compute cofactor k-eff by hole numbe  no            cka  compute cofactor k-eff by array number  yes  ***
***                                                                                                           ***
***  fmh  print fiss prod matrix by hole numbe yes            fma  print fiss prod matrix by array number  yes  ***
***                                                                                                           ***
***  hhl  collect matrix by highest hole level  no            hal  collect matrix by highest array level   no  ***
***                                                                                                           ***
***  amx  print all mixed cross sections       yes            far  print fis. and abs. by region          yes   ***
***                                                                                                           ***
***  xs1  print 1-d mixture x-sections         yes            gas  print far by group                      yes   ***
***                                                                                                           ***
***  xs2  print 2-d mixture x-sections         yes            pax  print xsec-albedo correlation tables   yes   ***
***                                                                                                           ***
***  xsl  print 2-d mixture Pl arrays          yes            pwt  print weight average array            yes   ***
***                                                                                                           ***
***  xap  print mixture angles & probabilities yes            pgm  print input geometry                  yes   ***
***                                                                                                           ***
***  pki  print fission spectrum               yes            bug  print debug information                no   ***
***                                                                                                           ***
***  pld  print extra 1-d cross sections       yes            trk  print tracking information             no   ***
***                                                                                                           ***
***  tfm  coordinate transform for fluxes       no            pmf  print angular fluxes and flux moments   no   ***
***                                                                                                           ***
***       print fluxes (flx)                   yes            app  append, not overwrite, restart data    no   ***
***                                                                                                           ***
***  mfx  compute mesh fluxes                   no            pms  print mesh fluxes if calculated        no   ***
***                                                                                                           ***
***  mfp  compute region mean free paths        no            pmm  print mesh flux moments if calculated  no   ***
***                                                                                                           ***
***  sen  compute derivative sensitivites       no            pmv  print mesh volumes                     no   ***
***                                                                                                           ***
***  cep  continuous energy calculation         no            ptb  use probability tables                yes   ***
***                                                                                                           ***
***  fre  use analytic free gas kernel         yes            pnu  use prompt neutron spectrum only       no   ***
***                                                                                                           ***
***  udi  use double indexing                   no            cbt  compute contributons                   no   ***
***                                                                                                           ***
***  pct  print contributons                    no            cds  collect CADIS fissions                  no   ***
***                                                                                                           ***
***  cfe  use collision flux estimator          no            wdk  use woodcock tracking                  no   ***
***                                                                                                           ***
***  scx  save CE cross sections in restart     no            uum  use unionized mixture cross sections   no   ***
***                                                                                                           ***
***  htm  produce HTML output                   no            scd  mesh based source convergence diag.   yes   ***
***                                                                                                           ***
***  cgd  use F*(r) Mesh                        no            m2u  use unionized nuclide cross sections   no   ***
***                                                                                                           ***
***  gfx  grid flux, averaged over voxel volum  no            fst  print F*(r) mesh values               no   ***
***                                                                                                           ***
***  fis  fission rate mesh tally               no                                                           ***
***                                                                                                           ***
***                                                                                                           ***
*****************************************************************************************************************
*****************************************************************************************************************
*****************************************************************************************************************
                                    parameter input completed

                        ........ finished reading the parameter data      ........

                       *************** data reading completed ***************
```

Fig. 8.1.96: Sample table of logical parameter data.

For the logical parameter data table, messages concerning the parameter data may be printed at the bottom of the table. If the problem is being restarted, the title of the parent case is printed at the bottom of the table. If

the restart title or messages are not printed, the bottom section of the table is omitted.

8.1.5.3 Unprocessed geometry input data

This printout is optional and is usually used to locate code difficulties, to show all the geometry input data when only part of it is used in the problem, or to show the order in which units were entered. It is considered debug information and is printed only if **PGM**=YES is specified in the parameter input data. Standard KENO use does not require printing these data because the processed geometry that is used in the problem is always printed. See Sect. 8.1.5.56.9 and Sect. 8.1.5.15 for examples of the standard printed KENO geometry data.

When the unprocessed geometry input is printed, the problem title is located at the top of the page, followed by the heading "GEOMETRY DESCRIPTION INPUT." The region-dependent geometry information is then printed. If the problem contains a unit orientation array, the problem title is printed again, followed by the unit orientation. This is followed by a statement affirming the completion of the data input.

8.1.5.4 Table of data sets used in the problem

This table is the third table of data printed by KENO. It should be carefully scrutinized to verify the desired data set name is associated with the proper unit number and volume. An example of this table is shown in Fig. 8.1.97.

```
*******************************************************************************************************************
***                                                                                                                 ***
***                            sample problem 18   1f27  critical experiment                                         ***
***                                                                                                                 ***
*******************************************************************************************************************
*******************************************************************************************************************
***                                                                                                                 ***
***          unit                                              volume                                               ***
***         number              data set name                   name        unit function                           ***
***         ------              -------------                   ----        -------------                           ***
***                                                                                                                 ***
***      xsc  14     ->_0q9_bbm4x_x5_b/T/scale.5kq.65846/ft14f001            mixed cross sections                    ***
***                                                                                                                 ***
***      alb  79      unknown                                                input albedos                           ***
***                                                                                                                 ***
***      wts  80      unknown                                                input weights                           ***
***                                                                                                                 ***
***      skt  16      unknown                                                write scratch data                      ***
***                                                                                                                 ***
***      rst  35     ->_x5_b/T/scale.5kq.65846/restart.keno_input            read restart data                       ***
***                                                                                                                 ***
***      lib   4     ->_0q9_bbm4x_x5_b/T/scale.5kq.65846/ft04f001            input ampx working library              ***
***                                                                                                                 ***
***            8     ->_0q9_bbm4x_x5_b/T/scale.5kq.65846/xfile008            input data direct access                ***
***                                                                                                                 ***
***           10      unknown                                                xsec mixing direct access               ***
***                                                                                                                 ***
*******************************************************************************************************************

                      ........ finished preparing input data    ........
```

Fig. 8.1.97: Sample table of data sets used in the problem.

This table lists unit numbers specified in the parameter data or that are defaulted in the code, along with the information pertinent to them. This information is given in the following order, left to right: (1) the keyword used in the parameter data to define the unit number, (2) the unit number, (3) the data set name, (4) the name of the volume on which the data set resides, and (5) the type of data contained on the data set. This table can be useful for quality assurance purposes. Information for units for which default values have not been overridden is printed even though they may not be used in the problem. Information for every unit specified in the parameter data is also printed. Units 8 and 10 are the direct-access devices, and their unit numbers are fixed within the code. When this table is printed, Unit 10 has not yet been defined. This causes its data set names to be listed as FT10F001 or as "UNKNOWN" on some systems. If KENO is run as part of a CSAS sequence, this table will include two entries for Unit 35: one for binary input data, and one for read restart data.

8.1.5.5 Table of additional information

The fourth table of data printed by KENO contains additional information determined from the input data. An example of this table is shown in Fig. 8.1.98.

This table should be used to verify the problem input. The NUMBER OF ENERGY GROUPS for a multigroup problem is read either from the Monte Carlo formatted library, identified by the keyword **XSC** and the unit function name MIXED CROSS SECTIONS from the Table of Data Sets in Sect. 8.1.5.4, or from the restart unit, identified by the keyword **RST** and the unit function name, READ RESTART DATA. The NO. OF FISSION SPECTRUM SOURCE GROUP is the number of different energy groups for which a fission spectrum is defined. In the present version, this number should always be 1. The NO. OF SCATTERING ANGLES IN XSECS is the number of scattering angles to be used in processing the cross sections. The default value is 3, and it may be overridden by specifying the parameter **SCT=** in the mixing table input. One scattering angle yields P_1 cross sections, two scattering angles yield P_3 cross sections, three scattering angles yield P_5 cross sections, etc.

```
*****************************************************************************************************
***                                                                                               ***
***          sample problem 18   1f27  critical experiment                                        ***
***                                                                                               ***
*****************************************************************************************************
*****************************************************************************************************
***                                                                                               ***
***                               ****** additional information ******                            ***
***                                                                                               ***
***  use a global unit                        yes    use lattice geometry                    yes  ***
***                                                                                               ***
***  no. of scattering angles in xsecs          2    global array number                       5  ***
***                                                                                               ***
***  number of mixtures used                    3    number of units in the global x dir.      1  ***
***                                                                                               ***
***  number of bias id's used                   6    number of units in the global y dir.      1  ***
***                                                                                               ***
***  number of differential albedos used        1    number of units in the global z dir.      1  ***
***                                                                                               ***
***  total input geometry regions              15    number of energy groups                 252  ***
***                                                                                               ***
***  number of geometry regions used           15    no. of fission spectrum source grps.      1  ***
***                                                                                               ***
***  use nested arrays                        yes    use nested holes                         no  ***
***                                                                                               ***
***  number of arrays used                      5    number of holes                           0  ***
***                                                                                               ***
***  maximum array nesting level                2    maximum hole nesting level                0  ***
***                                                                                               ***
***  largest array number                       5    largest geometry unit number              3  ***
***                                                                                               ***
***                                                                                               ***
***  boundary label 60               cuboid                                                       ***
***                                                                                               ***
***    +x boundary condition          vacuum         -x boundary condition              vacuum  ***
***                                                                                               ***
***    +y boundary condition          vacuum         -y boundary condition              vacuum  ***
***                                                                                               ***
***    +z boundary condition          vacuum         -z boundary condition                 h2o  ***
***                                                                                               ***
*****************************************************************************************************
```

Fig. 8.1.98: Sample table of additional information.

The NUMBER OF MIXTURES USED is the number of different mixtures (media) used in the geometry data used by the problem. This may be less than the total number of different mixtures specified in the geometry data if portions of the geometry data are not used in the problem.

The NUMBER OF BIAS IDS USED is the number of different biasing regions used in the problem. This will always be one unless a biasing data block is entered.

The NUMBER OF DIFFERENTIAL ALBEDOS USED is the number of different differential albedo reflectors used in the problem. This will always be zero unless the boundary condition data specify the use of differential albedo reflection on one or more faces of the system as described in Sect. 8.1.5.8. The BOUNDARY CONDITION data printed in this table should also be checked. The number of different differential albedos specified on the faces should be consistent with the NUMBER OF DIFFERENTIAL ALBEDOS USED. Specular, mirror, vacuum, and periodic are not differential albedos. Several different keywords may be used to specify the same differential albedo.

The TOTAL INPUT GEOMETRY REGIONS is the number of geometry regions specified in the problem input. This excludes UNIT label and comments provided using COM=, but it includes the array boundary description. It excludes the automatic reflector description, but it includes the geometry regions generated by it. The NUMBER OF GEOMETRY REGIONS USED is the number of geometry regions used in the problem. It may be less than or equal to the TOTAL INPUT GEOMETRY REGIONS. The LARGEST GEOMETRY UNIT NUMBER is the largest unit number defined in the geometry data, including unused units and implicitly defined units. Implicitly defined unit numbers are created when a core boundary specification is not immediately preceded by a specification. The unit number is assigned by the code by adding one to the largest unit number encountered in the geometry region data. For example, if two such core boundary specifications are contained in a problem whose largest explicitly defined unit number is 7, then a unit number of 8 is assigned to the first one, and a unit number of 9 is assigned to the second one. A value of 9 would be printed for the LARGEST GEOMETRY UNIT NUMBER. The LARGEST ARRAY NUMBER is the largest array number specified in the array data.

USE LATTICE GEOMETRY is determined by the logical flag that indicates whether or not the problem is a single unit problem. This should be YES for any problem that is not a single unit problem and NO for a single unit problem. By definition, a single unit problem does not use array data in any form. Sect. 8.1.5.14 describes array data. The GLOBAL ARRAY NUMBER is the number of the array designated as the global, overall, or universal array. The global array can be thought of as the array that defines the overall system.

The NUMBER OF UNITS IN THE GLOBAL X/Y/Z DIR. defines the size of the global array in terms of the number of units that are located along the edge of the array boundaries in the X/Y/Z directions. For a single unit, all three of these should be zero. For a simple $1 \times 1 \times 1$ array consisting of one unit type, all three of these numbers should be 1.

USE NESTED HOLES is set YES if holes are nested deeper than one level.

NUMBER OF HOLES is the number of HOLES that are entered in the geometry region data.

The MAXIMUM HOLE NESTING LEVEL is the deepest level of hole nesting.

USE NESTED ARRAYS is set YES if arrays are nested deeper than one level.

The NUMBER OF ARRAYS USED is the number of array descriptions actually used in the problem description.

MAXIMUM ARRAY NESTING LEVEL is the deepest level of array nesting.

BOUNDARY CONDITIONs are printed near the bottom of the table. They show the type of boundary condition that is applied to each surface of the system. These should all be VACUUM unless albedo boundary conditions are applied to one or more faces of the system. One should refer to the NUMBER OF DIFFERENTIAL ALBEDOS USED, as discussed previously in the description of this table of information.

The outer boundary of the global unit can consist of multiple shapes in KENO-VI geometry. For such a case, BOUNDARY CONDITIONS are printed for all shapes enclosing the global unit as shown in Fig. 8.1.99.

```
***                                                                                                  ***
***  boundary label 10               sphere                                                          ***
***                                                                                                  ***
***    surface(1 )                    vacuum                                                          ***
***                                                                                                  ***
***  boundary label 20             cylinder                                                          ***
***                                                                                                  ***
***    surface(1 )                    vacuum         surface( 2 )                        vacuum    ***
***                                                                                                  ***
***    surface(3 )                    vacuum                                                          ***
***                                                                                                  ***
***  boundary label 30               cuboid                                                          ***
***                                                                                                  ***
***    +x boundary condition          vacuum         -x boundary condition               vacuum    ***
***                                                                                                  ***
***    +y boundary condition          vacuum         -y boundary condition               vacuum    ***
***                                                                                                  ***
***    +z boundary condition          vacuum         -z boundary condition               vacuum    ***
***                                                                                                  ***
***  boundary label -40            cylinder                                                          ***
***                                                                                                  ***
***    surface(1 )                    vacuum         surface( 2 )                        vacuum    ***
***                                                                                                  ***
***    surface(3 )                    vacuum                                                          ***
***                                                                                                  ***
********************************************************************************************************
```

Fig. 8.1.99: Boundary conditions of a KENO-VI model in which combination of multiple shapes encloses the global unit.

Note: Boundary condition edit in Table of additional information does not list the surfaces added to each boundary body by CHORD operations.

8.1.5.6 Cross section data edits for the continuous energy mode

Unlike with the multigroup mode, KENO with continuous-energy mode loads each nuclide data directly from libraries. When loading data, Doppler broadening temperature correction (controlled by **DBX** parameter, see Sect. 8.1.7.2.19 for further details) is performed by default, on all nuclide cross sections, if the requested temperature is more than 4 K from the library temperature. If the Doppler broadening temperature correction is not enabled, then the nuclides' data are loaded from libraries with the closest temperature. When running KENO standalone, material temperatures can be entered by the **TEMP** parameter in the *MIXT data block*. When running KENO as part of a SCALE sequence, all material temperatures can be entered by each composition data definition in the *COMPOSITION data block*.

Cross sections data edits in KENO output summarize the data setup in the continuous-energy calculations. Examples of this summary are given in Fig. 8.1.100 and Fig. 8.1.101 for a sample continuous-energy calculation with and without Doppler broadening temperature correction.

This edit starts with printing the thermal energy cutoff utilized in the problem that, followed by a diagnostic edit, notifies that inverse velocity calculations have completed. In the next section, nuclide id, temperature at which data are being processed, and name of temperature-independent and -dependent cross section data files loaded are printed. When Doppler broadening temperature correction has been enabled (**DBX=** 2), one of the temperature-dependent data files used in this correction is printed as shown in Fig. 8.1.100. This is one of the caveats of this edit since full details of broadening are not listed here. Some messages from the cross section data loader related to broadening can be shown in the message file. After this section, nuclides

for which DBRC data loaded are listed if DBRC has been enabled in the calculations. Finally, elapsed time required for data processing (loading + broadening) is reported.

```
    Thermal cutoff value for freegas and bound nuclide kinematics treatment :  1.00000E+01

                              ........ finished calculating inverse velocities    ........


Processing nuclide 1 of 3, ID= 92234 at T=  440.00K
Temperature Independent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92234000701_0
  Temperature Dependent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92234000701_293.0


Processing nuclide 2 of 3, ID= 92235 at T=  440.00K
Temperature Independent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92235000701_0
  Temperature Dependent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92235000701_293.0


Processing nuclide 3 of 3, ID= 92238 at T=  440.00K
Temperature Independent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92238000701_0
  Temperature Dependent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92238000701_293.0

 List of DBRC enabled nuclides:
     92234        92235        92238


   0.05783 minutes were required for loading CE data.
```

Fig. 8.1.100: Example of cross section data summary for continuous-energy mode (with DBX=2, DBRC=2)

When Doppler broadening temperature correction is turned off (**DBX=** 0), KENO loads the temperature-dependent data from libraries with the closest temperature and prints their names as shown in Fig. 8.1.101. KENO always notifies the user with a warning message as shown in Fig. 8.1.101 if there is a difference between the temperature at which data would be loaded and the closest temperature for which data has been loaded.

```
    Thermal cutoff value for freegas and bound nuclide kinematics treatment :  1.00000E+01

                              ........ finished calculating inverse velocities    ........


Processing nuclide 1 of 3, ID= 92234 at T=  440.00K

 ***** warning ***** keno message number k6-290 follows:
 could not find temp:  440.0 on cross-section file.  using closest temperature:  565.0
Temperature Independent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92234000701_0
  Temperature Dependent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92234000701_565.0


Processing nuclide 2 of 3, ID= 92235 at T=  440.00K

 ***** warning ***** keno message number k6-290 follows:
 could not find temp:  440.0 on cross-section file.  using closest temperature:  565.0
Temperature Independent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92235000701_0
  Temperature Dependent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92235000701_565.0


Processing nuclide 3 of 3, ID= 92238 at T=  440.00K

 ***** warning ***** keno message number k6-290 follows:
 could not find temp:  440.0 on cross-section file.  using closest temperature:  565.0
Temperature Independent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92238000701_0
  Temperature Dependent CE XS File: /ornldev/code/Scale/S63/INSTALL/G_63_v6.3.x/data/cekenolib_7.1/92238000701_565.0

 List of DBRC enabled nuclides:
     92234        92235        92238


   0.02667 minutes were required for loading CE data.
```

Fig. 8.1.101: Example of cross section data summary for continuous-energy mode (with DBX=0, DBRC=2)

Note: Mixture temperatures printed in the mixing table output edit discussed in the next section are the user-defined temperatures at which cross section data processing has been requested. On the contrary, **nuclide**

temperatures printed in the mixing table output edit show the exact temperatures utilized at which nuclide data are processed for the current calculation.

8.1.5.7 Mixing table data edits

If **LIB=** is entered in the KENO parameter data and a mixing table data block is provided to KENO, mixing table data will be printed. This output edit is not considered optional because it cannot be suppressed if the necessary data are present. There are some differences between the mixing table data edits for both multigroup and continuous-energy modes.

Multigroup mode mixing table data edit

Sample mixing table data for multigroup mode are shown in Fig. 8.1.102. In the HTML output, the mixing table data can be accessed with the *Mixing Table* link in the *Input Data* section.

The data printed in this table include the problem title, the number of scattering angles, and the cross section message threshold. Data are then printed for each mixture. First the mixture number, density, and temperature are printed, followed by a table of the nuclides which make up the mixture. This table contains the following data: nuclide ID number, nuclide mixture ID number, atom density, weight fraction of nuclide in mixture, ZA number, atomic weight, temperature, and nuclide title. Mixture temperature is the same as the nuclides' temperatures for the multigroup calculations. After all mixture data have been printed, a table of nuclides and descriptive titles is printed for all nuclides included in the mixtures. If extra 1-D cross sections were specified in the problem (see **X1D**=, Sect. 8.1.3.3), the extra 1-D cross section IDs will be printed under the heading "1-D CROSS SECTION ARRAY ID NUMBERS." If $\bar{\nu}$ is to be calculated (parameter **NUB**=YES), six MT numbers will be printed. The MT number for the total cross section (Σ_T) is 1; the MT number for the sum of the transfer array normalized by Σ_T is 2002; the MT number for the normalized fission-product cross section ($\nu\Sigma_f/\Sigma_t$) is 1452; the MT number for the normalized absorption cross section ($\sum_{abs}/\Sigma_T$) is 27; the MT number for the normalized fission cross section (Σ_f/Σ_T) is 18; and the MT number for the fission spectrum (χ) is 1018. χ is summed and normalized to 1.0. Other MT numbers in this list have been specified by the user. If the number of blocks on the direct access data set are insufficient to hold the cross section data, a message is printed stating THE NUMBER OF DIRECT ACCESS BLOCKS ON UNIT_____ HAS BEEN INCREASED TO ____. If the problem is to write a restart data set (parameter **RES**=), a message is printed stating that restart information was written, and the restart I/O unit number is specified. This is followed by a statement of the number of I/Os used in preparing the cross sections. The user should examine the mixing table carefully to verify that the proper nuclides are specified for the proper mixtures and that all the data are correct. The mixing table is printed in subroutine PRTMIX.

```
                                cross sections read from the ampx working library on unit    4

1                                       sample problem 18   1f27 demonstration of options problem

                                                mixing table

                                          number of scattering angles =  2
                                       cross section message threshold = 1.0E+00

 mixture =     1          density(g/cc) =  1.5552             temperature(K) =    300.00
    nuclide    nucmix   atom-dens.   wgt. frac.     za      awt          temp             nuclide title
       1001       1    5.77865E-02  6.21826E-02    1001     1.0078       300.00     h_h2o 1 fast: h1 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:50 2018
       1002       1    6.64621E-06  1.42926E-05    1002     2.0141       300.00     d_d2o 11 fast: h2 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:13 2018
       7014       1    2.12317E-03  3.17441E-02    7014    14.0031       300.00     n14 725 endf-7 rel1 rev7 mod0 Sun Apr  8 12:12:08 2018
       7015       1    7.75657E-06  1.24228E-04    7015    15.0001       300.00     n15 728 endf-7 rel1 rev7 mod0 Sun Apr  8 12:12:09 2018
       8016       1    3.73205E-02  6.37360E-01    8016    15.9949       300.00     o16 825 endf-7 rel1 rev7 mod4 Sun Apr  8 12:12:27 2018
       8017       1    1.42163E-05  2.58030E-04    8017    16.9991       300.00     o17 828 endf-7 rel1 rev7 mod0 Sun Apr  8 12:12:27 2018
       8018       1    7.66934E-05  1.47389E-03    8018    17.9992       300.00     Injected O-18 zero cross sections
      92234       1    1.06784E-05  2.66842E-03   92234   234.0410       300.00     u234 9225 endf-7 rel1 rev7 mod2 Sun Apr  8 12:14:12 2018
      92235       1    9.84603E-04  2.47096E-01   92235   235.0439       300.00     u235 9228 endf-7 rel1 rev7 mod7 Sun Apr  8 12:14:13 2018
      92236       1    5.29387E-06  1.33421E-03   92236   236.0456       300.00     u236 9231 endf-7 rel1 rev7 mod1 Sun Apr  8 12:14:14 2018
      92238       1    6.19415E-05  1.57437E-02   92238   238.0508       300.00     u238 9237 endf-7 rel1 rev7 mod5 Sun Apr  8 12:14:18 2018

 mixture =     2          density(g/cc) =  1.1800             temperature(K) =    293.00
    nuclide    nucmix   atom-dens.   wgt. frac.     za      awt          temp             nuclide title
       1001       2    5.67766E-02  8.05238E-02    1001     1.0078       293.00     h_h2o 1 fast: h1 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:50 2018
       1002       2    6.53006E-06  1.85083E-05    1002     2.0141       293.00     d_d2o 11 fast: h2 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:13 2018
       6000       2    3.54895E-02  5.99840E-01    6000    12.0107       293.00     c 600 endf-7 rel1 rev7 mod0 Sun Apr  8 12:10:10 2018
       8016       2    1.41613E-02  3.18752E-01    8016    15.9949       293.00     o16 825 endf-7 rel1 rev7 mod4 Sun Apr  8 12:12:27 2018
       8017       2    5.39440E-06  1.29044E-04    8017    16.9991       293.00     o17 828 endf-7 rel1 rev7 mod0 Sun Apr  8 12:12:27 2018
       8018       2    2.91013E-05  7.37111E-04    8018    17.9992       293.00     Injected O-18 zero cross sections

 mixture =     3          density(g/cc) = 0.90000             temperature(K) =    293.00
    nuclide    nucmix   atom-dens.   wgt. frac.     za      awt          temp             nuclide title
    9001001       3    7.99137E-02  1.48599E-01    1001     1.0078       293.00     h_ch2 37 fast: h1 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:49 2018
       6000       3    3.84201E-02  8.51401E-01    6000    12.0107       293.00     c 600 endf-7 rel1 rev7 mod0 Sun Apr  8 12:10:10 2018

 mixture =     4          density(g/cc) = 0.99820E-09         temperature(K) =    293.00
    nuclide    nucmix   atom-dens.   wgt. frac.     za      awt          temp             nuclide title
       1001       4    6.67279E-11  1.11873E-01    1001     1.0078       293.00     h_h2o 1 fast: h1 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:50 2018
       1002       4    7.67459E-15  2.57139E-05    1002     2.0141       293.00     d_d2o 11 fast: h2 endf-7 rel1 rev7 mod0 Tue Apr 10 15:50:13 2018
       8016       4    3.32867E-11  8.85694E-01    8016    15.9949       293.00     o16 825 endf-7 rel1 rev7 mod4 Sun Apr  8 12:12:27 2018
       8017       4    1.26798E-14  3.58566E-04    8017    16.9991       293.00     o17 828 endf-7 rel1 rev7 mod0 Sun Apr  8 12:12:27 2018
       8018       4    6.84039E-14  2.04816E-03    8018    17.9992       293.00     Injected O-18 zero cross sections


  ***** warning ***** keno message number k5-222 follows:
  441 transfers for mixture 1 were corrected for bad moments.

  ***** warning ***** keno message number k5-222 follows:
  531 transfers for mixture 2 were corrected for bad moments.

  ***** warning ***** keno message number k5-222 follows:
  528 transfers for mixture 3 were corrected for bad moments.

  ***** warning ***** keno message number k5-222 follows:
  526 transfers for mixture 4 were corrected for bad moments.

                                ........ finished mixing cross-sections       ........

                                 1-d cross section array id numbers

                                           neutron
                                 reaction name    reaction id
                                        total              1
                                non-absorption           2002
                                    nu-fission           1452
                                    absorption             27
                                       fission             18
                                           chi           1018

                                ........ finished preparing the cross sections    ........
```

Fig. 8.1.102: Example of mixing table data for multigroup mode.

Note: See Sect. 8.1.4.4.4 for the details for the warning message printed in mixing table edit.

Continuous-energy mode mixing table data edit

In continuous energy mode, each mixture data is printed immediately after the problem title. For a mixture, first the mixture number, density, and temperature are printed, followed by a table of the nuclides which make up this mixture. Format of the nuclide edits in this table is same as the one used for multigroup mode, it contains the following data: nuclide ID number, nuclide mixture ID number, atom density, weight fraction of nuclide in mixture, ZA number, atomic weight, temperature, and nuclide title.

Sample mixing table data for continuous-energy mode are shown in Fig. 8.1.103. The nuclides' titles are relatively short compared to those for the multigroup mode since these data are not provided by AMPX, and KENO creates a short title while loading each nuclide data from the continuous-energy libraries.

```
1                                        test ce xs loading and mixing table edit

                                                    mixing table

 mixture =     1          density(g/cc) =  18.742              temperature(K) =    440.00
    nuclide   atom-dens.   wgt. frac.      za       awt          temp     nuclide title
      92234  4.91995E-04  1.02020E-02    92234   234.0410        440.00     u-234 @ 440K
      92235  4.49996E-02  9.37112E-01    92235   235.0439        440.00     u-235 @ 440K
      92238  2.49800E-03  5.26861E-02    92238   238.0508        440.00     u-238 @ 440K
```

Fig. 8.1.103: Example of mixing table data for continuous energy mode with **DBX=2**.

Unlike the multigroup calculations, mixture temperature may be different from the nuclides' temperatures for the continuous-energy calculations if Doppler broadening temperature correction is not enabled (setting **DBX=** to 0 disables the temperature correction) because the nuclides are loaded from libraries with the closest temperature. Fig. 8.1.104 is a sample mixing table data edits for such a case.

```
1                                        test ce xs loading and mixing table edit

                                                    mixing table

 mixture =     1          density(g/cc) =  18.742              temperature(K) =    440.00
    nuclide   atom-dens.   wgt. frac.      za       awt          temp     nuclide title
      92234  4.91995E-04  1.02020E-02    92234   234.0410        565.00     u-234 @ 565K
      92235  4.49996E-02  9.37112E-01    92235   235.0439        565.00     u-235 @ 565K
      92238  2.49800E-03  5.26861E-02    92238   238.0508        565.00     u-238 @ 565K
```

Fig. 8.1.104: Example of mixing table data for multigroup mode with **DBX=0**.

8.1.5.8 Albedo cross section correspondence

Printing the albedo cross section correspondence tables is optional. The headings for the tables are printed in subroutine CORRE, and then subroutine RATIO prints the data. These tables are printed only if **PAX**=YES is specified in the parameter data as described in Sect. 8.1.3.3. Examples of these tables are shown in Fig. 8.1.105 and Fig. 8.1.106.

The table shown in Fig. 8.1.105 contains, left to right, the cross section energy group, the lower and upper lethargy bounds, the corresponding albedo energy groups, and the cumulative probability associated with each albedo energy group for choosing the albedo energy group corresponding to the cross section energy group. The table shown in Fig. 8.1.106 is the inverse of the table shown in Fig. 8.1.105. It provides the cumulative probabilities for choosing the cross section energy group corresponding to the albedo energy group. The information in these tables is automatically generated by KENO.

xsec energy group	cross section group lethargy boundaries		cumulative probabilities for choosing the corresponding albedo group for each xsec group							
			albedo group	probability	albedo group	probability	albedo group	probability	albedo group	probability
1	-0.693147	-0.549854	1	1.00000						
2	-0.549854	-0.449801	1	1.00000						
3	-0.449801	-0.375006	1	1.00000						
4	-0.375006	-0.324978	1	1.00000						
5	-0.324978	-0.249980	1	1.00000						
6	-0.249980	0.00000	1	1.00000						
7	0.00000	0.200038	1	1.00000						
8	0.200038	0.440989	1	1.00000						
9	0.440989	0.733969	1	1.00000						
10	0.733969	0.843040	1	1.00000						
11	0.843040	1.20397	1	1.00000						
12	1.20397	1.39473	2	1.00000						
11	16.4058	16.4748	14	1.00000						
...										
230	18.1975	18.4207	15	1.00000						
231	18.4207	18.5260	16	1.00000						
232	18.5260	18.6438	16	1.00000						
233	18.6438	18.7774	16	1.00000						
234	18.7774	18.9315	16	1.00000						
235	18.9315	19.1138	16	1.00000						
236	19.1138	19.3370	16	1.00000						
237	19.3370	19.6247	16	1.00000						
238	19.6247	19.7950	16	1.00000						
239	19.7950	20.7233	16	1.00000						
240	20.7233	21.0109	16	1.00000						
241	21.0109	21.4164	16	1.00000						
242	21.4164	21.6396	16	1.00000						
243	21.6396	21.9272	16	1.00000						
244	21.9272	22.1096	16	1.00000						
245	22.1096	22.3327	16	1.00000						
246	22.3327	22.6204	16	1.00000						
247	22.6204	22.8435	16	1.00000						
248	22.8435	23.0259	16	1.00000						
249	23.0259	23.3135	16	1.00000						
250	23.3135	23.7190	16	1.00000						
251	23.7190	25.3284	16	1.00000						
252	25.3284	27.6310	16	1.00000						

Fig. 8.1.105: Cumulative probabilities for correlating the albedo energy group to the cross section energy group.

albedo energy group	albedo group lethargy boundaries		cumulative probabilities for choosing the corresponding xsec group for each albedo group							
			xsec group	probability	xsec group	probability	xsec group	probability	xsec group	probability
1	-0.405465	1.20397	3	0.01893	4	0.05001	5	0.09661	6	0.25193
			7	0.37622	8	0.52593	9	0.70797	10	0.77574
			11	1.00000						
2	1.20397	1.96611	12	0.25029	13	0.31818	14	0.63430	15	0.90947
			16	1.00000						
3	1.96611	2.40795	17	0.07227	18	0.13832	19	0.25650	20	0.34889
			21	0.54582	22	0.73902	23	0.95026	24	1.00000
4	2.40795	3.21888	25	0.03474	26	0.05449	27	0.11479	28	0.22483
			29	0.34747	30	0.36392	31	0.50000	32	0.55678
			33	0.60730	34	0.74472	35	0.80113	36	0.88247
			37	0.93983	38	1.00000				
5	3.21888	4.60517	39	0.13877	40	0.28352	41	0.50000	42	0.71234
			43	0.82024	44	1.00000				
6	4.60517	6.37713	45	0.09172	46	0.11200	47	0.16235	48	0.17761
			49	0.28828	50	0.36904	51	0.39118	52	0.45064
			53	0.67946	54	0.90828	55	1.00000		
7	6.37713	8.11173	56	0.15465	57	0.33548	58	0.43239	59	0.62997
			60	0.84875	61	0.87290	62	1.00000		
8	8.11173	9.80818	63	0.10747	64	0.16958	65	0.18283	66	0.30111
			67	0.38926	68	0.40859	69	0.56521	70	0.67783
			71	0.87234	72	0.88366	73	1.00000		
9	9.80818	11.5129	74	0.34586	75	0.38565	76	0.48645	77	0.53749
			78	0.56618	79	0.57209	80	0.58757	81	0.61430
			82	0.61888	83	0.62814	84	0.63064	85	0.65521
			86	0.68874	87	0.79019	88	0.88335	89	0.89796
			90	0.90540	91	0.91294	92	0.92831	93	0.95486
			94	0.97138	95	0.99300	96	1.00000		
10	11.5129	12.7169	96	0.02530	97	0.08751	98	0.16788	99	0.18534
			100	0.22794	101	0.27285	102	0.32645	103	0.35780
			104	0.38376	105	0.41055	106	0.45244	107	0.52107
			108	0.56581	109	0.60445	110	0.65954	111	0.68189
			112	0.71266	113	0.74055	114	0.76940	115	0.77996
			116	0.80366	117	0.81179	118	0.81977	119	0.82290
			120	0.82581	121	0.84857	122	0.86018	123	0.87197
			124	0.90217	125	0.91457	126	0.95291	127	0.96609
			128	1.00000						
11	12.7169	13.8155	129	0.07920	130	0.16596	131	0.26186	132	0.29272
			133	0.31603	134	0.34659	135	0.36907	136	0.39680
			137	0.44003	138	0.51700	139	0.57218	140	0.66809
			141	0.76821	142	0.84166	143	0.87278	144	1.00000
12	13.8155	15.0195	145	0.07833	146	0.17502	147	0.27864	148	0.29625
			149	0.31121	150	0.32645	151	0.35780	152	0.39038
			153	0.42428	154	0.51179	155	0.57572	156	0.62711
			157	0.74055	158	0.81910	159	0.87196	160	0.94639
			161	0.97276	162	1.00000				
13	15.0195	16.1181	163	0.00915	164	0.04032	165	0.07260	166	0.10607
			167	0.14082	168	0.17695	169	0.21073	170	0.24185
			171	0.27819	172	0.31603	173	0.36907	174	0.39680
			175	0.43513	176	0.48027	177	0.52777	178	0.57789
			179	0.63093	180	0.66179	181	0.69373	182	0.72683
			183	0.76119	184	0.79689	185	0.81528	186	0.83404
			187	0.85321	188	0.87278	189	0.88073	190	0.88875
			191	0.89684	192	0.90501	193	0.91324	194	0.92156
			195	0.92995	196	0.93841	197	0.94696	198	0.95559
			199	0.96430	200	0.97309	201	0.98198	202	0.99094
			203	1.00000						
14	16.1181	17.0344	204	0.02763	205	0.05598	206	0.08508	207	0.11499
			208	0.17737	209	0.24353	210	0.31396	211	0.38926
			212	0.47014	213	0.51294	214	0.55749	215	0.65245
			216	0.75647	217	0.87146	218	1.00000		
15	17.0344	18.4207	219	0.04655	220	0.09632	221	0.14978	222	0.20752
			223	0.27028	224	0.33904	225	0.41504	226	0.50000
			227	0.59632	228	0.70752	229	0.83904	230	1.00000
16	18.4207	23.0259	231	0.02288	232	0.04845	233	0.07745	234	0.11092
			235	0.15052	236	0.19897	237	0.26144	238	0.29844
			239	0.50000	240	0.56247	241	0.65051	242	0.69897
			243	0.76144	244	0.80103	245	0.84949	246	0.91195
			247	0.96041	248	1.00000				

Fig. 8.1.106: Cumulative probabilities for correlating the cross section energy group to the albedo energy group.

8.1.5.9 1-D macroscopic cross sections

The decision to print the 1-D mixture cross sections is optional. They are printed only if **XS1**=YES is specified in the parameter data. When the 1-D cross sections are to be printed, they are printed a group at a time for each mixture. The 1-D mixture cross sections for a mixture are shown in Fig. 8.1.107.

1 sample problem 18 1f27 demonstration of options problem

mixture id = 1 mixture index = 1 mixture number 1

neutron cross sections

group	sigt	sigs	siga	sum	signu	chi	mwa1	mwa2	mwa3
1	1.01785E-01	9.45089E-01	6.47494E-02	1.00984E+00	1.04583E-01	1.13271E-09	1	252	1
2	1.03027E-01	9.31130E-01	7.70553E-02	1.00818E+00	1.01900E-01	3.44665E-09	253	503	2
3	1.10450E-01	9.12964E-01	9.36923E-02	1.00666E+00	8.92615E-02	6.11147E-09	504	753	3
4	1.09496E-01	9.13469E-01	9.30402E-02	1.00651E+00	8.58342E-02	7.66991E-08	754	1002	4
5	1.11283E-01	9.13141E-01	9.36065E-02	1.00675E+00	7.56141E-02	5.32758E-05	1003	1250	5
6	1.12977E-01	9.03980E-01	1.03998E-01	1.00798E+00	6.33236E-02	1.18609E-03	1251	1497	6
7	1.15246E-01	9.34641E-01	7.17118E-02	1.00635E+00	5.89280E-02	3.94561E-03	1498	1743	7
8	1.20836E-01	9.56124E-01	4.76222E-02	1.00375E+00	4.56120E-02	1.42240E-02	1744	1928	8
9	1.46047E-01	9.75248E-01	2.51097E-02	1.00036E+00	2.39848E-02	4.34884E-02	1929	2129	9
...									
236	3.09353E+00	8.38248E-01	1.61752E-01	1.00000E+00	3.26783E-01	3.72138E-11	30998	31026	223
237	3.41776E+00	8.30064E-01	1.69937E-01	1.00000E+00	3.42918E-01	3.72138E-11	31027	31054	224
238	3.75936E+00	8.23357E-01	1.76643E-01	1.00000E+00	3.55708E-01	1.74905E-11	31055	31082	224
239	4.45723E+00	8.08418E-01	1.91582E-01	1.00000E+00	3.83493E-01	5.69371E-11	31083	31111	224
240	5.63046E+00	7.79400E-01	2.20600E-01	1.00000E+00	4.38457E-01	9.30344E-12	31112	31139	225
241	6.20234E+00	7.61119E-01	2.38881E-01	1.00000E+00	4.73979E-01	9.30344E-12	31140	31167	225
242	6.80423E+00	7.41488E-01	2.58512E-01	1.00000E+00	5.12460E-01	3.72138E-12	31168	31195	225
243	7.29650E+00	7.25608E-01	2.74392E-01	1.00000E+00	5.43756E-01	3.72138E-12	31196	31224	224
244	7.81985E+00	7.10134E-01	2.89866E-01	1.00000E+00	5.74322E-01	1.86069E-12	31225	31253	224
245	8.29416E+00	6.97433E-01	3.02567E-01	1.00000E+00	5.99437E-01	1.86069E-12	31254	31282	224
246	8.95499E+00	6.81681E-01	3.18319E-01	1.00000E+00	6.30626E-01	1.86069E-12	31283	31311	224
247	9.74211E+00	6.66032E-01	3.33968E-01	1.00000E+00	6.61639E-01	1.11641E-12	31312	31340	224
248	1.04452E+01	6.54555E-01	3.45445E-01	1.00000E+00	6.84381E-01	7.44275E-13	31341	31369	224
249	1.13284E+01	6.42663E-01	3.57337E-01	1.00000E+00	7.07965E-01	9.30344E-13	31370	31398	224
250	1.28584E+01	6.27151E-01	3.72849E-01	1.00000E+00	7.38752E-01	9.30344E-13	31399	31427	224
251	1.68445E+01	5.98844E-01	4.01156E-01	1.00000E+00	7.95519E-01	1.48855E-12	31428	31456	224
252	3.79891E+01	5.81383E-01	4.18617E-01	1.00000E+00	8.29391E-01	3.34924E-13	31457	31485	224

group	fission reaction
1	2.10576E-02
2	2.14899E-02
3	1.97992E-02
4	1.95559E-02
5	1.77511E-02
6	1.60046E-02
7	1.60416E-02
8	1.33569E-02
9	7.66452E-03
...	
236	1.34109E-01
237	1.40730E-01
238	1.45979E-01
239	1.57382E-01
240	1.79939E-01
241	1.94517E-01
242	2.10309E-01
243	2.23153E-01
244	2.35697E-01
245	2.46003E-01
246	2.58803E-01
247	2.71531E-01
248	2.80864E-01
249	2.90543E-01
250	3.03177E-01
251	3.26474E-01
252	3.40375E-01

Fig. 8.1.107: Example of macroscopic 1-D cross sections.

When the 1-D mixture cross sections are printed, the problem title is printed at the top of the page. The mixture ID, mixture index, and mixture number are then printed. ID is the mixture number from the mixing

table, and mixture index is the index used to reference it and mixture number is its identifier. This step is followed by a heading to identify the different 1-D cross sections. *GROUP* is the energy group, *sigt* is the total cross section for the mixture, *sigs* is the nonabsorption probability, *siga* is the absorption probability, *signu* is the production probability, *chi* is the fission spectrum, *mwa1* is the pointer for the first position of the cross sections for the energy group, *mwa2* is the pointer for the last position of the cross sections for the energy group, and *mwa3* contains the group for the transfer corresponding to the first position. *SUM* is the sum of the absorption probability and the nonabsorption probability. The absorption probability is defined as the absorption cross section divided by the total cross section. The nonabsorption probability is the sum of the group-to-group transfers for this group, divided by the total cross section. The production probability is defined as the fission production cross section divided by the total cross section $\left(\nu\Sigma_f/\Sigma_T\right)$. The nonabsorption probability and the production probability are not true probabilities in that they may be greater than 1. This is because the nonabsorption probability has the (n,2n) transfer array summed into the total transfer array twice, and the (n,3n) is summed three times, etc.

8.1.5.10 Extra 1-d cross sections

Printing the extra 1-D cross sections is optional. They are printed if **P1D**=YES is specified in the parameter data. Extra 1-D cross sections are not used in KENO unless **NUB**=YES is specified in the parameter data or the user has altered the code to access and utilize other 1-D cross sections. If **NUB**=YES is specified, the extra 1-D cross section is the fission cross section, which is used to calculate the average number of neutrons per fission. This is printed only for fissile mixtures as shown in Fig. 8.1.107. The fission cross section heading follows the table of 1D cross sections. The fission cross section heading is XSEC ID 18, and it follows the table of 1-D cross sections.

8.1.5.11 2-D macroscopic cross sections

The decision to print the 2-D mixture cross sections is optional. They are printed only if **XS2**=YES is specified in the parameter data. They are printed after the 1-D cross sections for the mixture. A heading is printed, followed by the transfer data. An example of the 2-D mixture cross sections is given in Fig. 8.1.108.

```
   primary-to-primary scattering transfer array for material     1

     from   grp   1     grp   2     grp   3     grp   4     grp   5     grp   6     grp   7     grp   8     grp   9     grp  10
to grp
  +  0    3.46700E-01 3.06373E-01 1.88229E-01 1.67434E-01 2.13034E-01 2.55041E-01 1.94480E-01 2.23373E-01 3.07452E-01 2.03452E-01
  +  1    1.43236E-01 1.11537E-01 1.51027E-01 1.90533E-01 2.74060E-01 2.10591E-01 2.49047E-01 2.84912E-01 1.38036E-01 3.84723E-01
  +  2    3.49496E-02 3.81987E-02 8.98138E-02 1.71236E-01 5.80425E-02 8.79871E-02 1.03626E-01 4.38749E-02 1.95463E-01 6.92324E-02
  +  3    1.61039E-02 3.32688E-02 1.03464E-01 5.20543E-02 7.21292E-02 8.55131E-02 3.15558E-02 1.16266E-01 5.99519E-02 1.66733E-02
  +  4    1.90561E-02 7.15188E-02 5.69775E-02 8.29922E-02 1.13299E-01 3.35973E-02 8.97854E-02 4.72779E-02 1.44405E-02 6.75688E-02
  +  5    8.18223E-02 7.83076E-02 9.26824E-02 7.73648E-02 2.63466E-02 1.13780E-01 4.75567E-02 1.14741E-02 5.86475E-02 4.73278E-02
  +  6    5.79328E-02 6.45878E-02 6.80265E-02 2.05585E-02 6.76245E-02 4.78141E-02 1.37706E-02 4.68179E-02 4.13502E-02 1.36207E-02
  +  7    5.22961E-02 6.42870E-02 2.24695E-02 6.55701E-02 3.09970E-02 9.75124E-03 6.88355E-02 3.47734E-02 1.19552E-02 6.00987E-03
  +  8    6.30351E-02 2.48377E-02 7.10586E-02 3.15097E-02 7.18278E-03 3.61544E-02 5.30020E-02 1.08720E-02 5.27872E-03 5.33405E-03
  +  9    1.99708E-02 7.05778E-02 3.35641E-02 8.88525E-03 2.89032E-02 2.19785E-02 1.44089E-02 4.94072E-03 4.69097E-03 9.17517E-03

  ...

  +236    6.84166E-14 5.14200E-14 2.72490E-13 9.65527E-14 1.00143E-13 1.21755E-14 1.66756E-15
  +237    3.21558E-14 1.67389E-13 4.45245E-14 9.65527E-14 4.00570E-14 1.21755E-14 8.33779E-16
  +238    1.04677E-13 2.73511E-14 4.45245E-14 3.86210E-14 4.00571E-14 6.08776E-15 8.33778E-16
  +239    1.71041E-14 2.73511E-14 1.78098E-14 3.86211E-14 1.99569E-14 6.08776E-15 8.33779E-16
  +240    1.71041E-14 1.09404E-14 1.78098E-14 1.93105E-14 1.99569E-14 6.08776E-15 5.00267E-16
  +241    6.84165E-15 1.09404E-14 8.90490E-15 1.93105E-14 1.99569E-14 3.65266E-15 3.33511E-16
  +242    6.84165E-15 5.47022E-15 8.90490E-15 1.93105E-14 1.19741E-14 2.38240E-15 4.16889E-16
  +243    3.42083E-15 5.47021E-15 8.90490E-15 1.15185E-14 7.98276E-15 2.97801E-15 4.16889E-16
  +244    3.42083E-15 5.47022E-15 5.34294E-15 7.67897E-15 9.97845E-15 2.97800E-15 6.67023E-16
  +245    3.42083E-15 3.28213E-15 3.48988E-15 9.59872E-15 9.97844E-15 4.87021E-15 1.50080E-16
  +246    1.90142E-15 2.12957E-15 4.45245E-15 9.59872E-15 1.59655E-14 1.07208E-15
  +247    1.26762E-15 2.66196E-15 4.45245E-15 1.54484E-14 3.54107E-15
  +248    1.58452E-15 2.66196E-15 7.12392E-15 3.41510E-15
  +249    1.58452E-15 4.37617E-15 1.52870E-15
  +250    2.73666E-15 8.44783E-16
  +251    4.94500E-16
```

Fig. 8.1.108: Example of 2-D macroscopic cross sections.

8.1.5.12 Probabilities and angles

Printing the probabilities and angles is optional. They are printed if the number of scattering angles is greater than zero and **XAP**=YES is specified in the parameter data. Examples of the probabilities are shown in Fig. 8.1.109. Examples of the angles are shown in Fig. 8.1.110. If the group-to-group transfer for a mixture is isotropic, the first angle for that transfer will be set to -2.0 as a flag to the code.

```
   primary-to-primary probability   1  array for material     1

     from   grp   1     grp   2     grp   3     grp   4     grp   5     grp   6     grp   7     grp   8     grp   9     grp  10
to grp
  +  0    9.63560E-01 9.62828E-01 9.80909E-01 9.85157E-01 9.81520E-01 8.76762E-01 8.76370E-01 8.25660E-01 7.93634E-01 9.35668E-01
  +  1    6.83026E-01 6.49903E-01 9.07656E-01 8.60920E-01 7.39515E-01 5.86246E-01 6.16569E-01 6.27510E-01 6.58375E-01 6.95968E-01
  +  2    6.37900E-01 5.00130E-01 5.74879E-01 6.34650E-01 9.86023E-01 8.98079E-01 9.86625E-01 9.93634E-01 7.79399E-01 9.96454E-01
  +  3    7.87266E-01 6.83375E-01 7.29838E-01 9.91190E-01 9.73194E-01 9.62238E-01 9.93305E-01 9.90003E-01 9.93389E-01 9.96465E-01
  +  4    9.44584E-01 9.39042E-01 9.75018E-01 8.87582E-01 7.75554E-01 8.84810E-01 9.75905E-01 9.88045E-01 9.93572E-01 9.94122E-01
  +  5    8.99122E-01 8.34797E-01 8.02371E-01 7.55004E-01 6.84087E-01 7.96112E-01 8.83006E-01 9.84957E-01 9.90026E-01 9.90748E-01
  +  6    7.40397E-01 7.49610E-01 7.90216E-01 9.36050E-01 8.51962E-01 7.27408E-01 8.39683E-01 9.76932E-01 9.84456E-01 9.88300E-01
  +  7    8.59508E-01 8.50254E-01 7.74468E-01 8.14731E-01 7.27313E-01 7.76257E-01 7.88605E-01 9.52672E-01 9.79624E-01 9.87435E-01
  +  8    7.96923E-01 7.98377E-01 7.86498E-01 7.65578E-01 8.07244E-01 7.64599E-01 7.13670E-01 9.09394E-01 9.77973E-01 9.86929E-01
  +  9    7.41183E-01 7.23635E-01 7.39315E-01 7.55225E-01 7.83032E-01 8.63365E-01 6.16747E-01 8.97387E-01 9.76837E-01 9.86450E-01

  ...

  +199    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00             9.86342E-01 9.72628E-01
  +200    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00             9.96722E-01 9.88589E-01
  +201    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00                         9.99305E-01
  +202    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
        transfers from grp+203 to grp+245 are the same as above
  +246    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  +247    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  +248    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  +249    0.00000E+00 0.00000E+00 0.00000E+00
  +250    0.00000E+00 0.00000E+00
  +251    0.00000E+00
```

Fig. 8.1.109: Example of macroscopic probabilities.

```
   primary-to-primary angle   1  array for material     1

     from   grp   1     grp   2     grp   3     grp   4     grp   5     grp   6     grp   7     grp   8     grp   9     grp  10
to grp
  +  0     9.41414E-01 9.39418E-01 9.52898E-01 9.56065E-01 9.43782E-01 9.20823E-01 9.22795E-01 9.06942E-01 8.84001E-01 9.12106E-01
  +  1     8.65205E-01 8.73290E-01 8.96937E-01 8.92093E-01 8.15144E-01 7.92073E-01 8.09832E-01 8.24878E-01 8.11288E-01 8.17374E-01
  +  2     8.59187E-01-8.58557E-02 1.14599E-01 7.83885E-01 8.25043E-01 8.07109E-01 7.97655E-01 8.04090E-01 8.13177E-01 7.78489E-01
  +  3     8.69005E-01 8.72481E-01 8.40319E-01 8.00722E-01 7.63962E-01 7.17893E-01 7.18987E-01 7.22049E-01 7.15606E-01 7.30790E-01
  +  4     8.46138E-01 8.31169E-01 7.94347E-01 7.48199E-01 6.35662E-01 6.52689E-01 6.55832E-01 6.25347E-01 6.71744E-01 6.82165E-01
  +  5     7.82955E-01 7.36950E-01 6.82401E-01 6.11001E-01 5.62609E-01 5.98973E-01 6.04256E-01 5.87640E-01 6.27148E-01 6.08859E-01
  +  6     6.67726E-01 6.50334E-01 6.17316E-01 5.96705E-01 5.62619E-01 5.02612E-01 5.79504E-01 5.49159E-01 5.60425E-01 5.66818E-01
  +  7     6.54310E-01 6.32518E-01 6.08768E-01 5.62528E-01 5.27400E-01 4.48794E-01 5.74618E-01 5.18556E-01 5.22290E-01 5.52846E-01
  +  8     6.05564E-01 6.03293E-01 5.87176E-01 5.79796E-01 5.10429E-01 4.51373E-01 4.35201E-01 5.14474E-01 5.09614E-01 5.44562E-01
  +  9     5.62779E-01 5.47529E-01 5.17000E-01 5.84487E-01 5.22808E-01 4.35614E-01 4.20075E-01 5.03786E-01 5.02343E-01 5.33959E-01

  ...

  +170   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00 9.97963E-01 9.99845E-01 9.99998E-01 1.00000E+00 9.99999E-01
  +171   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00 9.97819E-01 9.99824E-01 9.99998E-01 9.99999E-01 1.00000E+00
  +172   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00 9.99764E-01 9.99998E-01 9.99999E-01 9.99998E-01
  +173   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00 9.99613E-01 9.99999E-01 9.99999E-01 9.99999E-01
  +174   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00 9.99381E-01 9.99999E-01 9.99999E-01 9.99998E-01
  +175   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00 9.99999E-01 1.00000E+00 1.00000E+00
        transfers from grp+176 to grp+184 are the same as above
  +185   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00             9.99997E-01 9.99999E-01
        transfers from grp+186 to grp+200 are the same as above
  +201   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00                         1.00000E+00
  +202   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00
        transfers from grp+203 to grp+245 are the same as above
  +246   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00
  +247   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00
  +248   -2.00000E+00-2.00000E+00-2.00000E+00-2.00000E+00
  +249   -2.00000E+00-2.00000E+00-2.00000E+00
  +250   -2.00000E+00-2.00000E+00
  +251   -2.00000E+00
```

Fig. 8.1.110: Example of macroscopic angles.

8.1.5.13 Energy boundaries

KENO codes in the previous SCALE versions supported only single energy group boundaries used in all tally calculations. The energy group boundaries were obtained from the multigroup library used by transport process if the calculation mode is multigroup. These energy bounds may not be overridden by users in multigroup mode. In continuous-energy mode, energy group boundaries were defaulted to the library structure read from the SCALE 252-group multigroup library. The user could override the default energy boundaries either by defining the number of energy groups with **NGP** parameter, or by entering the energy points in **energy data** input block.

In SCALE 6.3, KENO codes can store multiple energy group boundaries; each may be used for different tallies. However, this capability is not enabled for standalone KENO calculations. If KENO is run as part of a CSAS or a TRITON sequence, then multiple energy boundaries can be entered in the **definitions data** input block available for these sequences (see Sect. 2.1.4.2.1).

Energy boundaries output edit always prints **DEFAULT** energy boundaries that are used in all tally calculations if it has not been otherwise requested. In multigroup mode, **DEFAULT** energy boundaries are always read from the multigroup library used by transport, and it cannot be overridden by the users. An example of energy boundaries output edit printed by a standalone multigroup KENO calculation is given in Fig. 8.1.111.

```
***** warning ***** keno message number k5-378 follows:
in multi group mode, DEFAULT energy boundaries used for tallying are always obtained from the multi group library used by
transport.


      ********************************************************************************************************************
      ***                                                                                                              ***
      ***                                          energy boundaries                                                   ***
      ***                                                                                                              ***
      ***     Only a single set of energy group boundaries is either specified or internally setup for this problem    ***
      ***                                                                                                              ***
      ********************************************************************************************************************

                               **** Energy group boundaries utilized in this problem ****

   Energy Boundaries DEFAULT
       title: Acquired from 8 group SCALE multi group library used by transport
        group   energy (eV)
       ------  ------------
            1   2.00000E+07
            2   8.20000E+05
            3   2.00000E+04
            4   1.05000E+02
            5   5.00000E+00
            6   6.50000E-01
            7   1.50000E-01
            8   4.00000E-02
                1.00000E-05
       ------  ------------


***** warning ***** keno message number k5-410 follows:
energy boundaries marked as DEFAULT will be used in all tally calculations if otherwise not requested.
```

Fig. 8.1.111: Example of energy boundaries output edit for multi group mode.

In continuous-energy mode, **DEFAULT** energy boundaries are defaulted to the library group structure of the SCALE 252-group library and can be overridden by (1) setting the number of energy groups with **NGP** parameter as shown in Example 8.1.44 and Example 8.1.45, (2) entering all energy points in **energy data** block Example 8.1.46, or (3) entering energy boundaries in the **definitions data** block (valid when running KENO codes as part of a sequence, see Sect. 2.1.4.2.1 for more details).

Example 8.1.44: Energy boundaries are set by NGP=11

```
...

read parameter
   ...
   NGP=11
end parameter

...
```

Example 8.1.45: Energy boundaries are set by NGP=8

```
...

read parameter
   ...
       NGP=8
end parameter

...
```

Example 8.1.46: Energy boundaries are read from energy data block

```
...

read energy
   2.e7 1.e5 1.e1 0.65 1.e-4
end energy

...
```

Three examples of energy boundaries output edits for continuous-energy KENO calculations are given in the Fig. 8.1.112, Fig. 8.1.113, and Fig. 8.1.114. In the first example, the energy boundaries constructed with having 11 equal lethargy intervals by setting **NGP=11** overwrites the **DEFAULT** energy boundaries. Fig. 8.1.113 shows the energy group output edits for a case in which the **DEFAULT** energy boundaries are overridden by the energy boundaries read from the SCALE 8 group test library that was provided by setting **NGP=8**. And the **DEFAULT** energy boundaries are overridden in the last example whose output edit shown in Fig. 8.1.114 by reading the energy boundaries from the **energy data** input block.

```
*****************************************************************************************************************
***                                                                                                           ***
***                                          energy boundaries                                                ***
***                                                                                                           ***
***     Only a single set of energy group boundaries is either specified or internally setup for this problem    ***
***                                                                                                           ***
*****************************************************************************************************************

                            **** Energy group boundaries utilized in this problem ****

   Energy Boundaries DEFAULT
       title: NGP specified as 11; constructed 11 equal lethargy neutron group structure
        group   energy (eV)
       ------  ------------
            1   2.00000E+07
            2   1.52319E+06
            3   1.16006E+05
            4   8.83494E+03
            5   6.72865E+02
            6   5.12451E+01
            7   3.90281E+00
            8   2.97236E-01
            9   2.26374E-02
           10   1.72405E-03
           11   1.31303E-04
                1.00000E-05
       ------  ------------


***** warning ***** keno message number k5-410 follows:
energy boundaries marked as DEFAULT will be used in all tally calculations if otherwise not requested.
```

Fig. 8.1.112: Example of energy boundaries output edit for continuous-energy mode (NGP=11).

```
*****************************************************************************************************************
***                                                                                                           ***
***                                          energy boundaries                                                ***
***                                                                                                           ***
***     Only a single set of energy group boundaries is either specified or internally setup for this problem    ***
***                                                                                                           ***
*****************************************************************************************************************

                            **** Energy group boundaries utilized in this problem ****

   Energy Boundaries DEFAULT
       title: NGP specified as 8; acquired from 8 group SCALE multi group library.
        group   energy (eV)
       ------  ------------
            1   2.00000E+07
            2   8.20000E+05
            3   2.00000E+04
            4   1.05000E+02
            5   5.00000E+00
            6   6.50000E-01
            7   1.50000E-01
            8   4.00000E-02
                1.00000E-05
       ------  ------------


***** warning ***** keno message number k5-410 follows:
energy boundaries marked as DEFAULT will be used in all tally calculations if otherwise not requested.
```

Fig. 8.1.113: Example of energy boundaries output edit for continuous-energy mode (NGP=8).

```
*********************************************************************************************************************
***                                                                                                               ***
***                                          energy boundaries                                                    ***
***                                                                                                               ***
***     Only a single set of energy group boundaries is either specified or internally setup for this problem    ***
***                                                                                                               ***
*********************************************************************************************************************

                            **** Energy group boundaries utilized in this problem ****

  Energy Boundaries DEFAULT
      title: Acquired from the specification in 'read energy'
       group   energy (eV)
       ------  ------------
            1   2.00000E+07
            2   1.00000E+05
            3   1.00000E+01
            4   6.50000E-01
                1.00000E-04
       ------  ------------


***** warning ***** keno message number k5-410 follows:
energy boundaries marked as DEFAULT will be used in all tally calculations if otherwise not requested.
```

Fig. 8.1.114: Example of energy boundaries output edit for continuous-energy mode (energy data).

8.1.5.14 Array summary

The arrays that are used in the problem are summarized in the table shown in Fig. 8.1.115. This table is printed whenever more than one array is used in the problem.

```
*********************************************************************************
**                                                                             **
**      array          units in       units in       units in       nesting    **
**      number          x dir.         y dir.         z dir.         level     **
**                                                                             **
**        1                2              2              2              1     **
**                                                                             **
**        2                2              2              1              1     **
**                                                                             **
**        3                1              2              3              1     **
**                                                                             **
**        4                3              1              3              1     **
**                                                                             **
*********************************************************************************

 ........ finished loading the data    ........
```

Fig. 8.1.115: Example of array summary.

The ARRAY NUMBER is the number by which the array is designated in the input data. The number of units in the X, Y, and Z directions is listed for each array. The NESTING LEVEL indicates the level of nesting for each array. The global, overall, or universe array is flagged by the word GLOBAL. The global array should always appear at the first nesting level. Arrays that have been placed in the global reflector by using holes should also appear at the first nesting level. A nesting level of one is the highest or first nesting levels. The larger the number in the nesting level column, the deeper the nesting level will be.

8.1.5.15 Geometry data edits

KENO V.a geometry edit

The geometry region data used by the problem are always printed and cannot be suppressed. They should be carefully examined by the user to verify the mixture number, bias ID, and geometry specifications used in the problem. If geometry region data are entered but are not referenced in the unit orientation array data, they will not be printed here. An example would be to enter geometry region data describing Units 1, 2, 3, and 4 and to use only Units 1, 3, and 4 in the unit orientation array. Then the geometry region data for Unit 2 will not be printed. An example of the KENO V.a geometry region printout for a problem is given in Fig. 8.1.116.

The problem title and a heading are printed at the top of each page. REGION is the region number within a unit. Each unit has its regions numbered sequentially, beginning with one. MEDIA NUM is the mixture number or mixture ID that occupies the volume defined by the region. BIAS ID is the bias ID that corresponds to the desired set of weight average for biasing the region. The unit number is printed at the top of each unit's geometry region description near the center of the page. The data printed for each geometry region include (1) the region number relative to the unit (numbered sequentially within the unit), (2) the shape of the geometry region, (3) the mixture ID of the material within the volume defined by the region, (4) the bias ID to define the average weight of a neutron in the region, and (5) the dimensions defining the outer boundaries of the geometry region. If additional geometry surrounds an array, a heading is printed stating: UNIT ____ EXTERNAL TO LATTICE ____. The lattice number is the number of the array that is surrounded by the specified geometry. The unit number is the unit that contains the specified geometry. In the case of an external reflector for the global array, the unit number is assigned by the code.

```
                          ........ finished loading the data    ........
1                                  sample problem 18   1f27 demonstration of options problem

                media bias      geometry description for those units utilized in this problem
 region          num   id

                                                -----   unit     1   -----

   1 cylinder         1  1  radius =  9.5200     +z =  8.7804     -z = -8.7804     centerline is at  x =  0.0000      y =  0.0000
   2 cylinder         0  1  radius =  9.5200     +z =  8.9896     -z = -8.7804     centerline is at  x =  0.0000      y =  0.0000
   3 cylinder         2  1  radius =  10.160     +z =  9.6296     -z = -9.4204     centerline is at  x =  0.0000      y =  0.0000
   4 cuboid           4  1      +x =  18.450     -x = -18.450     +y =  18.450     -y = -18.450     +z =  17.895     -z = -17.685

                                     -----   unit     2  external to lattice  1   -----

   1 array number     1         +x =  73.800     -x =  0.0000     +y =  73.800     -y =  0.0000     +z =  71.160     -z =  0.0000

                                     -----   unit     3  external to lattice  2   -----

   1 array number     2         +x =  73.800     -x =  0.0000     +y =  73.800     -y =  0.0000     +z =  35.580     -z =  0.0000

                                     -----   unit     4  external to lattice  3   -----

   1 array number     3         +x =  36.900     -x =  0.0000     +y =  73.800     -y =  0.0000     +z =  106.74     -z =  0.0000

                                     -----   unit     5  external to lattice  4   -----

   1 array number     4         +x =  110.70     -x =  0.0000     +y =  36.900     -y =  0.0000     +z =  106.74     -z =  0.0000

                                     ******************   global   ******************
                                                -----   unit     6   -----

   1 cuboid           4  1      +x =  55.350     -x = -55.350     +y =  55.350     -y = -55.350     +z =  53.370     -z = -53.370
     hole number      1       at x = -55.350      y = -18.450      z = -17.790     is unit number      2

     hole number      2       at x = -55.350      y = -18.450      z = -53.370     is unit number      3

     hole number      3       at x =  18.450      y = -18.450      z = -53.370     is unit number      4

     hole number      4       at x = -55.350      y = -55.350      z = -53.370     is unit number      5

   2 cuboid           3  2      +x =  58.350     -x = -58.350     +y =  58.350     -y = -58.350     +z =  56.370     -z = -56.370
   3 cuboid           3  3      +x =  61.350     -x = -61.350     +y =  61.350     -y = -61.350     +z =  59.370     -z = -59.370
   4 cuboid           3  4      +x =  64.350     -x = -64.350     +y =  64.350     -y = -64.350     +z =  62.370     -z = -62.370
   5 cuboid           3  5      +x =  67.350     -x = -67.350     +y =  67.350     -y = -67.350     +z =  65.370     -z = -65.370
   6 cuboid           3  6      +x =  70.350     -x = -70.350     +y =  70.350     -y = -70.350     +z =  68.370     -z = -68.370
   7 cuboid           3  7      +x =  70.590     -x = -70.590     +y =  70.590     -y = -70.590     +z =  68.610     -z = -68.610
```

Fig. 8.1.116: Example of KENO V.a geometry region data

KENO-VI geometry edit

The geometry region data utilized by the problem are printed and cannot be suppressed. They should be carefully examined by the user to verify the mixture number, bias ID, and geometry specifications used in the problem. If geometry region data are input but are not referenced in the unit orientation array data, they will not be printed here. An example would be to input geometry region data describing Units 1, 2, 3, and 4 and to utilize only Units 1, 3, and 4 in the unit orientation array. Then the geometry region data for Unit 2 will not be printed. An example of the geometry region printout for a problem is given in Fig. 8.1.117 and Fig. 8.1.118.

First, a table of geometry parameters for the problem is printed. This specifies the overall size of the problem, number of components in the problem, and the size of the array needed to store the geometry. Next, the quadratic equations by unit are printed. The problem title and a heading are printed at the top of each page. The unit number followed by the GEOMETRY data for that unit is then printed. Each geometry record type used in the unit, numbered in the order they appear in the unit, is printed out. Following each geometry record type is the set of quadratic equations that describe the input geometry for that geometry record. The CONTENTS data, consisting of four columns, are then listed in the order they appear in the problem. The first column contains the content keyword. The second column contains the media/hole/array number. The third column labeled IMP contains the bias ID number if the content keyword is MEDIA. Otherwise, this column is blank. The fourth column contains the sector definition array, which describes the region location relative to the GEOMETRY records listed above for that unit.

The data consisting of the unit number, GEOMETRY data, and CONTENT data are repeated for each unit used in the problem. The global unit has a heading printed before the unit number to identify it as the global unit.

```
1       ********************************************************************************************************************
        ***                                                                                                              ***
        ***                                                                                                              ***
        ********************************************************************************************************************
        ***                                ******      geometry parameters       ******                                  ***
        ***                                                                                                              ***
        ***                                                                                                              ***
        ***                    niar        number of independent array references              5                        ***
        ***                                                                                                              ***
        ***                    ngblu       global unit number                                  3                        ***
        ***                                                                                                              ***
        ***                    nboxt       number of units in the problem                      3                        ***
        ***                                                                                                              ***
        ***                    nquad       number of quadratics in the problem                42                        ***
        ***                                                                                                              ***
        ***                    ngwrds      number of geometry words read                      15                        ***
        ***                                                                                                              ***
        ***                    maxgwd      maximum geometry words in a unit                    6                        ***
        ***                                                                                                              ***
        ***                    maxsfu      largest number of surfaces in a unit               18                        ***
        ***                                                                                                              ***
        ***                    maxreg      largest number of media in a unit                   6                        ***
        ***                                                                                                              ***
        ***                    regtot      number of spatial volumes defined                  15                        ***
        ***                                                                                                              ***
        ***                    sectot      number of entries in the sector array              92                        ***
        ***                                                                                                              ***
        ***                    nucom       number of comments in the geometry data             0                        ***
        ***                                                                                                              ***
        ***                    numhol      number of holes in the problem                      0                        ***
        ***                                                                                                              ***
        ********************************************************************************************************************
```

Fig. 8.1.117: Example of KENO-VI geometry parameter table.

```
1                                    sample problem 18   1f27  critical experiment

                                geometry description for those units utilized in this problem


                                                  -----   unit 1   -----

      1        cylinder       10                          quadratic surfaces

     X**2         Y**2         Z**2           XY           XZ           YZ           X            Y            Z      Constant

  -1.00000E+00 -1.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +9.06304E+01

  +0.00000E+00 +0.00000E+00 -1.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +7.70954E+01

      2        cylinder       20                          quadratic surfaces

     X**2         Y**2         Z**2           XY           XZ           YZ           X            Y            Z      Constant

  -1.00000E+00 -1.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +9.06304E+01

  +0.00000E+00 +0.00000E+00 -1.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +2.09200E-01 +7.89323E+01

      3        cylinder       30                          quadratic surfaces

     X**2         Y**2         Z**2           XY           XZ           YZ           X            Y            Z      Constant

  -1.00000E+00 -1.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +1.03226E+02

  +0.00000E+00 +0.00000E+00 -1.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +2.09200E-01 +9.07147E+01

      4        cuboid         40                          quadratic surfaces

     X**2         Y**2         Z**2           XY           XZ           YZ           X            Y            Z      Constant

  -1.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +3.40402E+02

  +0.00000E+00 -1.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +3.40402E+02

  +0.00000E+00 +0.00000E+00 -1.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +0.00000E+00 +2.09200E-01 +3.16473E+02

                    sector
               imp      definitions

 media 1    1          10

 media 0    1          -10 20

 media 2    1          -10 -20 30

 media 0    1          40 -20 -30

 boundary                 40
```

Fig. 8.1.118: Example of KENO-VI geometry region data.

8.1.5.16 Unit orientation description

Each unit orientation description defines the location of units in the 3-D lattice that represent the specified array. The array described is identified in the heading UNIT ORIENTATION DESCRIPTION FOR ARRAY ____. The arrays used in the problem are stacked together to represent the physical problem being analyzed. The unit orientation description is not printed if only Unit 1 is described in the problem. The user should carefully examine the unit orientation descriptions to ensure proper placement of the units in each lattice. A sample unit orientation description is shown in Fig. 8.1.119 .

```
1                         sample problem 18   1f27 demonstration of options problem

                    -------  unit orientation description for array 1         -------

 z layer 1, x column 1 to 2 left to right   y row 1 to 2  bottom to top

  1 1
  1 1

 z layer 2, x column 1 to 2 left to right   y row 1 to 2  bottom to top

  1 1
  1 1
                    -------  unit orientation description for array 2         -------

 z layer 1, x column 1 to 2 left to right   y row 1 to 2  bottom to top

  1 1
  1 1
                    -------  unit orientation description for array 3         -------

 z layer 1, x column 1 to 1 left to right   y row 1 to 2  bottom to top

  1
  1

 z layer 2, x column 1 to 1 left to right   y row 1 to 2  bottom to top

  1
  1

 z layer 3, x column 1 to 1 left to right   y row 1 to 2  bottom to top

  1
  1
1                         sample problem 18   1f27 demonstration of options problem

                    -------  unit orientation description for array 4         -------

 z layer 1, x column 1 to 3 left to right   y row 1 to 1  bottom to top

  1 1 1

 z layer 2, x column 1 to 3 left to right   y row 1 to 1  bottom to top

  1 1 1

 z layer 3, x column 1 to 3 left to right   y row 1 to 1  bottom to top

  1 1 1
```

Fig. 8.1.119: Example of unit orientation description.

If a very large array is used by the problem, its unit orientation description may be spread over several pages. When checking the printout, the user should pay careful attention to the headings that indicate the portion of each lattice being printed. The detailed printout of an array may be suppressed using the ARRAY parameter **PRT=no**.

8.1.5.17 Volume information

The volume information printout differs between KENO V.a and KENO-VI. Because it does not allow intersections of bodies, KENO V.a is always able to calculate analytically the volume of any region in the geometry. KENO-VI, on the other hand, never calculates volumes analytically. In fact, KENO-VI never calculates volumes unless it is directed to do so, but it can read volumes off the **MEDIA** cards or off a **VOLUME** file (Sect. 8.1.3.13).

KENO V.a volume edit

Three tables of volumes are always printed and cannot be suppressed. The problem title is printed at the top of the page, followed by the heading "VOLUMES FOR THOSE UNITS UTILIZED IN THIS PROBLEM." An example of the volume printout is given in Fig. 8.1.120.

The first table is arranged by ascending unit number. It includes (1) the unit number, (2) the region number within the unit, (3) the overall geometry region number, (4) the net volume of each individual region, and (5) the cumulative volume through each region in the unit. The cumulative volume of the last region in a unit is the total volume of the unit. The unit number is printed under the heading UNIT. Data listed under the heading REGION refer to the number of the geometry region within the unit. The geometry regions within a unit are numbered sequentially starting with 1. Data entered under the heading GEOMETRY REGION refer to the entry number of the individual geometry region. These are numbered sequentially, starting with 1, through the TOTAL INPUT GEOMETRY REGIONS defined in Fig. 8.1.120. The net volume of each individual region is calculated by subtracting the volume of the interior region from the volume of the region and is listed under the heading VOLUME. The data listed under the heading CUMULATIVE VOLUME are calculated from the dimensions of the region. A simple example demonstrating how volumes are calculated can be given by assuming a unit that is composed of three concentric cubes. Region 1 is a cube 3 cm on a side, region 2 is a cube 4 cm on a side, and region 3 is a cube 5 cm on a side. The cumulative volume of region 1 is 27 cm^3 (3^3); the cumulative volume of region 2 is 64 cm^3 (4^3); the cumulative volume of region 3 is 125 cm^3 (5^3). The volume of region 1 is 27 cm^3 (3^3), the volume of region 2 is 37 cm^3 (64 - 27), and the volume of region 3 is 61 cm^3 (125 - 64).

```
1                                  sample problem 18   1f27 demonstration of options problem
                                  volumes for those units utilized in  this problem

                                    geometry                                        cumulative
unit    region      region            volume                                        volume

  1         1         1         4.99998E+03 cm**3         4.99998E+03 cm**3
            2         2         5.95642E+01 cm**3         5.05954E+03 cm**3
            3         3         1.11824E+03 cm**3         6.17778E+03 cm**3
            4         4         4.22683E+04 cm**3         4.84461E+04 cm**3

surrounding geometry volumes -  geometry region      5 is an array placement boundary region

  2         1         5         3.87569E+05 cm**3         3.87569E+05 cm**3

surrounding geometry volumes -  geometry region      6 is an array placement boundary region

  3         1         6         1.93784E+05 cm**3         1.93784E+05 cm**3

surrounding geometry volumes -  geometry region      7 is an array placement boundary region

  4         1         7         2.90677E+05 cm**3         2.90677E+05 cm**3

surrounding geometry volumes -  geometry region      8 is an array placement boundary region

  5         1         8         4.36015E+05 cm**3         4.36015E+05 cm**3

  6         1         9         7.17736E+00 cm**3         1.30805E+06 cm**3
            2        10         2.27350E+05 cm**3         1.53540E+06 cm**3
            3        11         2.52272E+05 cm**3         1.78767E+06 cm**3
            4        12         2.78490E+05 cm**3         2.06616E+06 cm**3
            5        13         3.06005E+05 cm**3         2.37217E+06 cm**3
            6        14         3.34815E+05 cm**3         2.70698E+06 cm**3
            7        15         2.80686E+04 cm**3         2.73505E+06 cm**3

                unit     uses    region     mixture        total volume

                  1        27        1          1         1.34999E+05 cm**3
                                     2          0         1.60823E+03 cm**3
                                     3          2         3.01924E+04 cm**3
                                     4          4         1.14124E+06 cm**3

                  2         1        1                    3.87569E+05 cm**3

                  3         1        1                    1.93784E+05 cm**3

                  4         1        1                    2.90677E+05 cm**3

                  5         1        1                    4.36015E+05 cm**3

                  6         1        1          4         7.17736E+00 cm**3
                                     2          3         2.27350E+05 cm**3
                                     3          3         2.52272E+05 cm**3
                                     4          3         2.78490E+05 cm**3
                                     5          3         3.06005E+05 cm**3
                                     6          3         3.34815E+05 cm**3
                                     7          3         2.80686E+04 cm**3

                          total mixture volumes
                      mixture        total volume              mass(g)
                          0         1.60823E+03 cm**3
                          1         1.34999E+05 cm**3        2.09955E+05
                          2         3.01924E+04 cm**3        3.56269E+04
                          3         1.42700E+06 cm**3        1.28430E+06
                          4         1.14125E+06 cm**3        1.13920E-03
```

Fig. 8.1.120: Sample volume information.

The second table contains (1) the unit number, (2) the number of times the unit is used in the problem, (3) the region number within the unit, (4) the mixture number present in the region, and (5) the total volume associated with the region in the whole problem. The unit number is printed under the heading UNIT, data printed under the heading USES indicates the number of times the unit is used in the problem. Data printed under the heading MIXTURE indicates the mixture number used in the region. The total volume of each region is printed under the heading TOTAL VOLUME and is determined by multiplying the VOLUME of the region listed in the first table by the number of times the unit containing that region is used in the problem.

The third table is printed following the heading “TOTAL MIXTURE VOLUMES.” In this table, the mixtures

used in the problem are listed with their associated total volume and total mass. The mixture numbers are printed under the heading MIXTURE, the total volume of each mixture is printed under the heading TOTAL VOLUME, and the mass of each mixture is printed under the heading MASS(G). All masses will be printed as zero if the working format cross section library does not contain the data required to calculate the densities of the mixtures used in the problem.

KENO-VI volume edit

Two tables of volumes are printed by subroutine VOLUME and cannot be suppressed. The first table will list the number of times each unit is used in the problem and the total volume of each region throughout the entire problem description. The second table will list the total volume and mass of each mixture used in the problem.

If a volume calculation type of NONE is specified and no volumes are read in for the regions, then the volumes will be set to -1.0, which will result in negative fluxes and fission densities. A sample volume output edit for such a case is given in Fig. 8.1.121. Calculations with zero volumes always provide a notification with a warning message as shown in Fig. 8.1.121.

```
1                                   keno-vi sample problem 3  2c8  15.24 cm paraffin refl


                    volumes for those units utilized in this problem

                volumes not specified in the input were set to -1.0

                                 geometry
                   unit     uses   region     mixture        total region volume (cm**3)

                     1          8       1           1         -1.00000E+00
                                        2           0         -1.00000E+00

                     2          1       1
                                        2           2         -1.00000E+00
                                        3           2         -1.00000E+00
                                        4           2         -1.00000E+00
                                        5           2         -1.00000E+00
                                        6           2         -1.00000E+00


                      mixture       total mixture volume (cm**3)            total mixture mass (gm)

                          1          0.00000E+00                             0.00000E+00
                          2          0.00000E+00                             0.00000E+00
                                    -------------                           -------------
                                     0.00000E+00                             0.00000E+00

 ***** warning ***** keno message number k6-239 follows:
 The volumes for all regions are not set.
 Some region volumes are not set by input, not in the space for which volumes are calculated,
 missed by the volume calculation, or actually 0.0. For these regions, volumes are set to -1.0
 and the fluxes and fission densities are divided by -1.0 instead of the real volume.
```

Fig. 8.1.121: Sample volume information (volumes are neither defined nor calculated).

For the calculated volumes, an error estimate is printed after the volumes. A summary of the parameters used in both **TRACE** and **RANDOM** volume estimates and some diagnostic information is always printed before the first table if any region volume is calculated by one of these methods. Two examples of the KENO-VI volume output edits with these volume calculation methods are given in Fig. 8.1.122 and Fig. 8.1.123, respectively.

```
1                                   keno-vi sample problem 3  2c8  15.24 cm paraffin refl

                        volumes for those units utilized in this problem

                    Volume Sampling Parameters
                    --------------------------

                    The number of points per batch was specified as 100000
                    This gives a sampling density of  2.19467E-01 points per cc per batch.
                    The number of batches is 1000
                    Time spent calculating volumes:    67.89 seconds.

                    volumes not specified in the input were calculated using randomly generated points
                    number of batches:                  1000
                    number of particles per batch:     100000
                    total number of particles:      100000000

                                       geometry
                       unit     uses    region     mixture        total region volume (cm**3)

                         1         8         1           1          8.92811E+03 +/- 5.97045E+00
                                             2           0          1.07100E+04

                         2         1         1
                                             2           2          4.41067E+04
                                             3           2          5.54420E+04 +/- 1.46060E+01
                                             4           2          6.80712E+04
                                             5           2          8.20248E+04 +/- 1.82121E+01
                                             6           2          1.05705E+05 +/- 1.87796E+01


                           mixture       total mixture volume (cm**3)            total mixture mass (gm)

                                0         1.07100E+04
                                1         8.92811E+03 +/- 5.97045E+00             1.67491E+05 +/- 1.12005E+02
                                2         3.55350E+05 +/- 1.46545E+01             3.19807E+05 +/- 1.31888E+01
                                         -------------                           -------------
                                          3.74988E+05                             4.87298E+05

 The maximum standard deviation associated with the calculated volumes is    0.0669 percent.

                                       Time to compute volumes (sec):      67.89
```

Fig. 8.1.122: Sample volume information, undefined volumes are calculated by RANDOM estimate.

```
1                                   keno-vi sample problem 3  2c8  15.24 cm paraffin refl

                        volumes for those units utilized in this problem


                    volumes not specified in the input were calculated using ray tracing
                    ray tracing is performed parallel to the y axis
                    number of rays along the x axis:                      1009
                    number of rays along the z axis:                       990
                    total number of rays used to estimate volumes:      998910
                    distance between rays in the x directions:     7.67493E-02
                    distance between rays in the z directions:     7.67475E-02
                    cross-sectional area represented by a ray:     5.89031E-03

                                    geometry
                       unit     uses    region     mixture        total region volume (cm**3)

                         1         8         1           1          8.91874E+03 +/- 8.92361E+00
                                             2           0          1.07100E+04

                         2         1         1
                                             2           2          4.41067E+04
                                             3           2          5.54461E+04 +/- 5.54763E+01
                                             4           2          6.80712E+04
                                             5           2          8.19374E+04 +/- 8.19821E+01
                                             6           2          1.05363E+05 +/- 1.05421E+02


                          mixture       total mixture volume (cm**3)            total mixture mass (gm)

                              0         1.07100E+04
                              1         8.91874E+03 +/- 8.92361E+00             1.67315E+05 +/- 1.67407E+02
                              2         3.54925E+05 +/- 7.30177E+01             3.19425E+05 +/- 6.57144E+01
                                        ------------                            ------------
                                        3.74554E+05                             4.86740E+05

 The maximum order of error associated with the calculated volumes is    0.1001 percent.

                                    Time to compute volumes (sec):      13.15
```

Fig. 8.1.123: Sample volume information, undefined volumes are calculated by TRACE estimate.

The unit number is printed under the heading **UNIT**. The REGION refers to the number of the geometry region within the unit determined by the order of the CONTENT records in the unit. USES indicates the number of times the unit is used in the problem. MIXTURE is the mixture number used in the region. A blank indicates an array or unit, placed in a hole, in that region. TOTAL VOLUME is the total volume of that region followed by +/- and an error estimate if the volume for the region was calculated.

If mesh volumes are used, there will be a table listing the cumulative volumes generated by summing the mesh volumes, the deviations, and the deltas between the summed volumes and the overall volumes for the regions. If the overall volume has been calculated, then the delta should be zero, but if it was entered, then the delta will show the difference.

In the last table, the mixtures used in the problem are listed along with their total volumes and masses. Along with the total volumes and masses, an error estimate is printed for any mixtures having regions that were calculated using either the **RANDOM** or **TRACE** volume estimates..

8.1.5.18 Mesh volumes output edit

With the mesh flux accumulator, fluxes are tabulated for each region of each unit in the defined mesh grid. KENO computes the volume of each region in each mesh interval so that fluxes per unit volume can be determined. For KENO V.a models where the mesh flux accumulator is activated by setting **MFX**=YES and either (a) the mesh size set to a positive value with **MSH**=, or (b) mesh grid defined by **grid geometry data** input, a mesh volume sampling summary and the cumulative volume of all mesh intervals for a given region are printed as shown in Fig. 8.1.124. Details of mesh grid 20001 on which a mesh volume calculation was performed are printed in the *grid definitions* output edit. An example of grid definitions output edit can be shown in Sect. 8.1.5.19.

Note that KENO V.a allows users to alter the default settings for the mesh volume calculation. However, **TRACE** volume estimate is not available for mesh volume; therefore, the **RANDOM** volume estimate is always enforced when *type=trace* is entered in **volume data**. A corresponding warning message (message number k?=370) is printed in the output to notify the user about this change.

```
1                                  sample problem 3  2c8  15.24 cm paraffin refl
                                  volumes for those units utilized in  this problem

                                    geometry                                  cumulative
                          unit    region      region            volume            volume

                            1         1         1         1.11737E+03 cm**3         1.11737E+03 cm**3
                                      2         2         1.14249E+04 cm**3         1.25423E+04 cm**3

                          surrounding geometry volumes -  geometry region      3 is an array placement boundary region

                            2         1         3         1.00338E+05 cm**3         1.00338E+05 cm**3
                                      2         4         4.41067E+04 cm**3         1.44445E+05 cm**3
                                      3         5         5.54410E+04 cm**3         1.99886E+05 cm**3
                                      4         6         6.80712E+04 cm**3         2.67957E+05 cm**3
                                      5         7         8.19974E+04 cm**3         3.49955E+05 cm**3
                                      6         8         1.05694E+05 cm**3         4.55649E+05 cm**3

                                           unit     uses    region     mixture        total volume

                                             1         8       1          1         8.93897E+03 cm**3
                                                               2          0         9.13995E+04 cm**3

                                             2         1       1                    1.00338E+05 cm**3
                                                               2          2         4.41067E+04 cm**3
                                                               3          2         5.54410E+04 cm**3
                                                               4          2         6.80712E+04 cm**3
                                                               5          2         8.19974E+04 cm**3
                                                               6          2         1.05694E+05 cm**3

                                                    total mixture volumes
                                                 mixture        total volume              mass(g)
                                                     0         9.13995E+04 cm**3
                                                     1         8.93897E+03 cm**3        1.67695E+05
                                                     2         3.55310E+05 cm**3        3.19772E+05

                                    Mesh Volume Sampling Parameters (Grid Number=20001)
                                    ---------------------------------------------------

                                    The default sampling density is  2.21616E-01 points per cc per batch.
                                    The number of points per batch for calculating volumes has been changed from 1000 to 100979.
                                    The number of batches is 500
                                    Time spent calculating mesh volumes:    12.43 seconds.
1                                            sample problem 3  2c8  15.24 cm paraffin refl
                                            cumulative mesh volumes for those units utilized in this problem
                          Per cent delta is the difference between the cumulative mesh volume and the cumulative analytic volume
                          Number of batches used to compute the mesh volumes

                                          Number of non zero mesh volumes 352

                                            geometry      cumulative                per cent           per cent
                          unit    region      region         volume                 deviation           delta

                            1         1         1         8.93276E+03 cm**3         1.00040E-01         6.94774E-02
                                      2         2         9.13968E+04 cm**3         3.14901E-02         3.00076E-03

                            2         1         3         0.00000E+00 cm**3         0.00000E+00         0.00000E+00
                                      2         4         4.40999E+04 cm**3         4.52652E-02         1.55548E-02
                                      3         5         5.54512E+04 cm**3         4.03670E-02         1.83840E-02
                                      4         6         6.80780E+04 cm**3         3.66147E-02         9.95418E-03
                                      5         7         8.20097E+04 cm**3         3.31634E-02         1.49694E-02
                                      6         8         1.05680E+05 cm**3         2.92803E-02         1.27053E-02
```

Fig. 8.1.124: Sample sum of mesh volumes edit (KENO V.a with cubic mesh, MSH=20.0).

Note: Unlike the KENO V.a implementation, mesh flux accumulation with KENO-VI always needs a **volume data** input block to activate the mesh volume calculation; otherwise, calculation is terminated with an error message. KENO-VI does not support mesh volume calculation with **TRACE** volume estimation. Just as KENO V.a does, KENO-VI continues by using the **RANDOM** volume estimation for the mesh volume calculation.

An optional edit of volumes for each mesh interval for each region can be activated by entering **PMV**=YES

in the parameter input. This edit can be very large, especially if a small mesh size is used with a large model. A sample of the volume by mesh for each region is shown in Fig. 8.1.125.

```
                Unit     1 Region     1
Z    Y    X      Mesh Vol  +/- Deviation     X      Mesh Vol  +/- Deviation
3    3    3     1.11646E+03   2.65595E-03    4     1.11974E+03   2.98595E-03
3    4    3     1.11679E+03   2.82875E-03    4     1.12005E+03   2.82394E-03

4    3    3     1.11624E+03   2.72476E-03    4     1.11665E+03   2.80074E-03
4    4    3     1.11275E+03   2.97251E-03    4     1.11407E+03   2.82853E-03


                Unit     1 Region     2
Z    Y    X      Mesh Vol  +/- Deviation     X      Mesh Vol  +/- Deviation     X      Mesh Vol  +/- Deviation     X      Mesh Vol  +/- Deviation
2    2    2     3.32196E+01   1.66560E-02    3     1.91502E+02   6.62690E-03    4     1.91999E+02   6.46629E-03    5     3.30662E+01   1.65420E-02
2    3    2     1.90311E+02   6.91900E-03    3     1.10184E+03   2.62040E-03    4     1.10325E+03   2.77220E-03    5     1.91250E+02   6.94470E-03
2    4    2     1.92161E+02   6.87280E-03    3     1.10043E+03   2.63093E-03    4     1.10386E+03   3.06249E-03    5     1.91557E+02   7.02654E-03
2    5    2     3.30843E+01   1.66603E-02    3     1.89968E+02   7.06502E-03    4     1.92125E+02   6.79234E-03    5     3.34904E+01   1.69270E-02

3    2    2     2.43647E+02   5.90675E-03    3     1.39675E+03   2.59862E-03    4     1.39337E+03   2.53642E-03    5     2.44964E+02   6.07556E-03
3    3    2     1.39466E+03   2.56304E-03    3     6.88671E+03   1.14910E-03    4     6.87797E+03   1.16346E-03    5     1.38802E+03   2.57214E-03
3    4    2     1.39513E+03   2.53674E-03    3     6.87471E+03   1.16592E-03    4     6.88259E+03   1.13055E-03    5     1.39062E+03   2.56434E-03
3    5    2     2.42555E+02   5.99206E-03    3     1.39571E+03   2.51428E-03    4     1.39428E+03   2.58297E-03    5     2.42140E+02   6.05211E-03

4    2    2     2.42221E+02   6.00102E-03    3     1.39350E+03   2.74577E-03    4     1.38670E+03   2.54432E-03    5     2.43177E+02   6.00705E-03
4    3    2     1.38839E+03   2.62767E-03    3     6.87403E+03   1.20502E-03    4     6.88518E+03   1.13485E-03    5     1.39884E+03   2.44403E-03
4    4    2     1.38804E+03   2.63486E-03    3     6.87565E+03   1.11821E-03    4     6.87810E+03   1.11507E-03    5     1.38831E+03   2.43160E-03
4    5    2     2.40398E+02   6.07647E-03    3     1.39092E+03   2.64789E-03    4     1.39800E+03   2.49478E-03    5     2.41138E+02   6.06353E-03

5    2    2     3.33008E+01   1.65273E-02    3     1.88813E+02   7.37733E-03    4     1.90627E+02   6.74738E-03    5     3.43567E+01   1.63116E-02
5    3    2     1.91809E+02   7.14122E-03    3     1.10104E+03   2.67744E-03    4     1.10070E+03   2.93997E-03    5     1.92125E+02   7.26510E-03
5    4    2     1.90627E+02   7.29589E-03    3     1.10304E+03   2.83623E-03    4     1.10410E+03   2.97534E-03    5     1.90780E+02   7.24190E-03
5    5    2     3.23442E+01   1.76331E-02    3     1.90428E+02   6.51665E-03    4     1.89291E+02   6.94074E-03    5     3.38604E+01   1.53700E-02


                Unit     2 Region     2
```

Fig. 8.1.125: Sample of volume by mesh for each region (only a small section of the output).

8.1.5.19 Grid definitions

This table specifies the mesh grids to be used in the problem. Each mesh grid is either defined by the **MSH** parameter and/or **grid geometry data** input, or constructed internally. An example of grid information is given in Fig. 8.1.126.

```
*****************************************************************************************************************
***                                                                                                               ***
***                                          grid definitions                                                     ***
***                                                                                                               ***
***     3 grid geometries have been defined (either read from user input or constructed for internal use)         ***
***                                                                                                               ***
***     Only 2 of them are stored and utilized for the requested tallies in this problem                          ***
***                                                                                                               ***
*****************************************************************************************************************

                              **** Grid geometries utilized in this problem ****

Grid geometries specified with Ids = (12 and 15) have identical grid definitions.
Only below specification is stored and all grid geometries with Ids listed above will point to this:

Grid Geometry: 12
    title: test fission rate mesh tally
    Plane Summary
      x:   10 cells from -6.00001E+01 to  6.00001E+01
      y:   10 cells from -6.00001E+01 to  6.00001E+01
      z:    5 cells from -6.00001E+01 to  6.00001E+01
    Total number of cells: 500

                        x-planes               y-planes               z-planes
    -----  ---------------------  ---------------------  ---------------------
        1  -6.00000599999998E+01  -6.00000599999998E+01  -6.00000599999998E+01
        2  -4.80000000000000E+01  -4.80000000000000E+01  -3.60000000000000E+01
        3  -3.60000000000000E+01  -3.60000000000000E+01  -1.20000000000000E+01
        4  -2.40000000000000E+01  -2.40000000000000E+01   1.20000000000000E+01
        5  -1.20000000000000E+01  -1.20000000000000E+01   3.60000000000000E+01
        6   0.00000000000000E+00   0.00000000000000E+00   6.00000599999998E+01
        7   1.20000000000000E+01   1.20000000000000E+01
        8   2.40000000000000E+01   2.40000000000000E+01
        9   3.60000000000000E+01   3.60000000000000E+01
       10   4.80000000000000E+01   4.80000000000000E+01
       11   6.00000599999998E+01   6.00000599999998E+01
    -----  ---------------------  ---------------------  ---------------------

Grid Geometry: 10001
    title: Default 5 x 5 x 5 Cartesian mesh which overlays the entire geometry
    Plane Summary
      x:    5 cells from -7.05902E+01 to  7.05902E+01
      y:    5 cells from -7.05902E+01 to  7.05902E+01
      z:    5 cells from -6.86102E+01 to  6.86102E+01
    Total number of cells: 125

                        x-planes               y-planes               z-planes
    -----  ---------------------  ---------------------  ---------------------
        1  -7.05901705900998E+01  -7.05901705900998E+01  -6.86101686100998E+01
        2  -4.23541023540599E+01  -4.23541023540599E+01  -4.11661011660599E+01
        3  -1.41180341180200E+01  -1.41180341180200E+01  -1.37220337220200E+01
        4   1.41180341180200E+01   1.41180341180200E+01   1.37220337220200E+01
        5   4.23541023540599E+01   4.23541023540599E+01   4.11661011660599E+01
        6   7.05901705900998E+01   7.05901705900998E+01   6.86101686100998E+01
    -----  ---------------------  ---------------------  ---------------------
```

Fig. 8.1.126: Sample of grid definitions.

Grid definitions output edit lists all user-defined and internally constructed mesh grids, even some that are not used for any mesh tally. Only one of the grids is stored in the grid data container if the multiple grids are defined with identical grid data. This can be easily seen in the diagnostic messages followed by *grid definitions* header as well as the messages printed just before each stored grid entry. The sample grid definitions shown in Fig. 8.1.126 specify that two mesh grids with grid **IDs=** 12 and 15 were defined by two **grid geometry data** inputs, but only one of them was stored since their definitions were identical. A default mesh grid was also shown in Fig. 8.1.126 that was constructed for Shannon entropy tallies.

8.1.5.20 Biasing information

This table specifies the weighting or biasing data to be used in the problem. An example of biasing information is given in Fig. 8.1.127.

```
*****************************************************************************************************************
***                                                                                                           ***
***                                        biasing information                                                ***
***                                                                                                           ***
***     weighting intervals   1 to   5 for paraffin    ,mat id=  400 will be used for bias id's    2 to    6     ***
***                                                                                                           ***
***     a default weight of    0.500 will be used for all other bias id's.                                    ***
***                                                                                                           ***
*****************************************************************************************************************
```

Fig. 8.1.127: Biasing information.

The user is responsible for determining from the input data whether the group-dependent weights (wtavg) for the specified material(s) were obtained from the weighting library or were entered by the user. The group-dependent weights can be printed for verification purposes as shown in Sect. 8.1.5.21.

8.1.5.21 Group-dependent weights

Printing the group-dependent weights is optional. They are printed if **PWT**=YES is entered in the parameter data. An example of the printed group-dependent weights is shown in Fig. 8.1.128.

```
1                                    sample problem 18   1f27 demonstration of options problem

                                                group dependent weights

 energy        bias          bias          bias          bias          bias          bias          bias
 group          id   1        id   2        id   3        id   4        id   5        id   6        id   7
 -----      ----------    ----------    ----------    ----------    ----------    ----------    ----------
   1        5.00000E-01   6.41222E-01   8.92920E-01   1.17715E+00   1.53432E+00   1.98277E+00   2.54552E+00
   2        5.00000E-01   6.44641E-01   9.04533E-01   1.20428E+00   1.58773E+00   2.07821E+00   2.70550E+00
   3        5.00000E-01   6.44296E-01   9.05535E-01   1.21290E+00   1.61257E+00   2.13216E+00   2.80737E+00
   4        5.00000E-01   6.43775E-01   9.06473E-01   1.22029E+00   1.63408E+00   2.17944E+00   2.89754E+00
   5        5.00000E-01   6.42455E-01   9.02606E-01   1.21828E+00   1.63966E+00   2.20208E+00   2.95205E+00
   6        5.00000E-01   6.38363E-01   8.94813E-01   1.21849E+00   1.66458E+00   2.27855E+00   3.12146E+00
   7        5.00000E-01   6.41099E-01   9.10396E-01   1.27094E+00   1.79576E+00   2.55876E+00   3.66361E+00
   8        5.00000E-01   6.24669E-01   8.76952E-01   1.24463E+00   1.81237E+00   2.68214E+00   4.00523E+00
   9        5.00000E-01   5.94198E-01   8.17904E-01   1.19010E+00   1.81300E+00   2.83897E+00   4.51577E+00
  10        5.00000E-01   5.94838E-01   8.31594E-01   1.24905E+00   1.99013E+00   3.29279E+00   5.57746E+00
  11        5.00000E-01   6.01741E-01   8.75513E-01   1.40025E+00   2.41119E+00   4.34442E+00   8.03922E+00

 ...

 236        5.00000E-01   9.99572E-01   4.21510E+00   1.75728E+01   7.32513E+01   3.05345E+02   1.27281E+03
 237        5.00000E-01   1.00882E+00   4.26455E+00   1.77790E+01   7.41111E+01   3.08928E+02   1.28775E+03
 238        5.00000E-01   1.01571E+00   4.30036E+00   1.79283E+01   7.47336E+01   3.11523E+02   1.29857E+03
 239        5.00000E-01   1.02516E+00   4.34761E+00   1.81253E+01   7.55547E+01   3.14946E+02   1.31284E+03
 240        5.00000E-01   1.03835E+00   4.40990E+00   1.83849E+01   7.66369E+01   3.19457E+02   1.33164E+03
 241        5.00000E-01   1.04464E+00   4.43791E+00   1.85017E+01   7.71237E+01   3.21487E+02   1.34010E+03
 242        5.00000E-01   1.05073E+00   4.46429E+00   1.86117E+01   7.75820E+01   3.23397E+02   1.34806E+03
 243        5.00000E-01   1.05506E+00   4.48266E+00   1.86882E+01   7.79011E+01   3.24727E+02   1.35361E+03
 244        5.00000E-01   1.05893E+00   4.49880E+00   1.87555E+01   7.81816E+01   3.25897E+02   1.35848E+03
 245        5.00000E-01   1.06194E+00   4.51118E+00   1.88071E+01   7.83966E+01   3.26793E+02   1.36222E+03
 246        5.00000E-01   1.06533E+00   4.52490E+00   1.88643E+01   7.86350E+01   3.27787E+02   1.36636E+03
 247        5.00000E-01   1.06855E+00   4.53777E+00   1.89179E+01   7.88586E+01   3.28718E+02   1.37025E+03
 248        5.00000E-01   1.07081E+00   4.54666E+00   1.89550E+01   7.90132E+01   3.29363E+02   1.37293E+03
 249        5.00000E-01   1.07306E+00   4.55536E+00   1.89912E+01   7.91643E+01   3.29993E+02   1.37556E+03
 250        5.00000E-01   1.07602E+00   4.56670E+00   1.90385E+01   7.93614E+01   3.30814E+02   1.37898E+03
 251        5.00000E-01   1.08025E+00   4.58234E+00   1.91037E+01   7.96332E+01   3.31947E+02   1.38371E+03
 252        5.00000E-01   1.08669E+00   4.60526E+00   1.91993E+01   8.00315E+01   3.33608E+02   1.39063E+03
```

Fig. 8.1.128: Example of biasing data.

The title is printed at the top of the table. The average weight (wtavg) is printed for each energy group and each BIAS ID. The BIAS ID number printed at the top of the column corresponds to the BIAS ID used in the geometry region description or MEDIA record and printed in the biasing information.

8.1.5.22 Plot representation

Plots representing 2-D slices through the geometrical description of the problem are optional. They are created if plot data are entered unless **PLT**=NO is specified either in the plot data or the parameter data. Plots can be generated and displayed as (1) character plots with alphanumeric characters representing mixture numbers, unit numbers, or bias ID numbers or (2) color plots with colors representing mixture numbers, unit numbers, or bias ID numbers. Color plots generate a PNG file and require an independent program to be displayed.

An example of the output generated using the character plot method is given in Fig. 8.1.129 and Fig. 8.1.130. An example of the output generated using the color plot method is given in Fig. 8.1.131 and Fig. 8.1.132.

```
  1f27 xy plot at z=0.0
                  the following will be a character plot

                                        mixture map

mixture  0 1 2 3 4
 symbol  . * - 3

      upper  left          lower right
      coordinates          coordinates

x     -7.1000e+01           7.1000e+01
y      7.1000e+01          -7.1000e+01
z      0.0000e+00           0.0000e+00

          u axis      v axis
          (down)      (across)

x          .00000      1.00000
y        -1.00000       .00000
z          .00000       .00000

nu=   78   nv=  130     delu= 1.8205e+00     delv= 1.0923e+00     lpi=   6.000
```

Fig. 8.1.129: Summary of character plot symbols, coordinates, and data.

Fig. 8.1.129 summarizes the data used to generate the character plot. Fig. 8.1.130 is an example of a character plot of the 2-D slice specified through the geometrical description of the problem. In Fig. 8.1.129, the plot title is printed at the top of the page, followed by a statement that "THE FOLLOWING WILL BE A CHARACTER PLOT." If a plot title was not entered in the plot data, the plot title is defaulted to the problem title. The title is followed by a heading specifying the type of plot (MIXTURE MAP, BIAS ID MAP, or UNIT MAP). This is followed by a table that correlates the symbols to be used in the character plot with the mixture numbers, bias ID numbers, or unit numbers that were used in the problem. If the problem is a bare array, the overall system coordinates are printed. Then the coordinates of the upper left corner and lower right corner of the plot are printed. This is followed by the direction cosines down and across the plot. The remaining plot parameters (including both input data and calculated values) are then printed. NU is the number of characters printed in the U (down) direction, NV is the number of characters printed in the V (across) direction, DELU is the incremental distance, in cm, represented by each character in the U (down) direction, DELV is the incremental distance, in cm, represented by each character in the V (across) direction, and LPI is the vertical to horizontal scaling factor for plot proportionality.

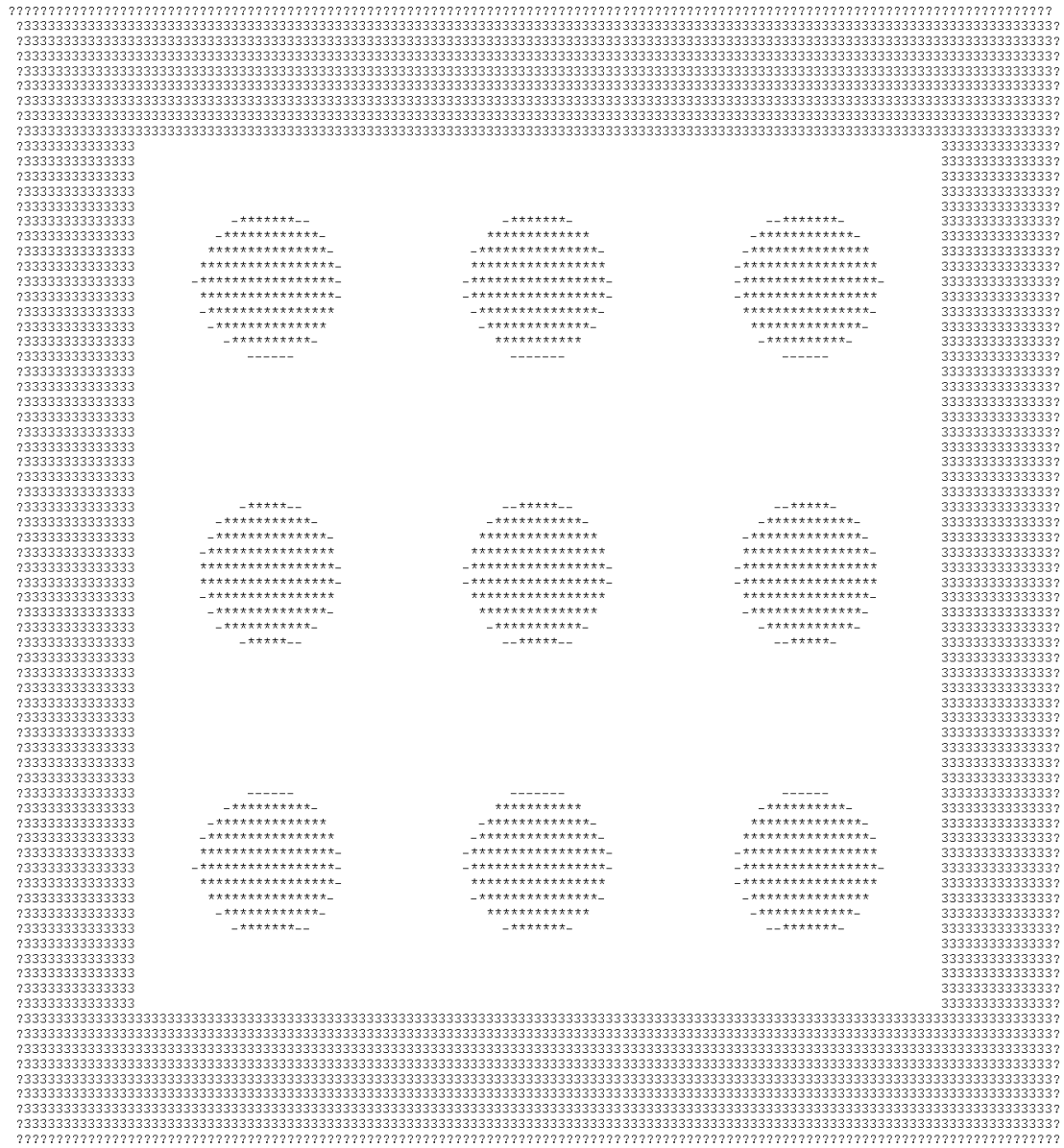


Fig. 8.1.130: Sample character plot representation.

Fig. 8.1.130 shows a character plot of a 2-D slice specified through the geometrical description of the problem. These plots aid the user in verifying that the problem is described correctly. Any number of plots can be made in one problem.

Fig. 8.1.131 summarizes the data used to generate the color plot. Fig. 8.1.132 is an example of a color plot of the 2-D slice specified through the geometrical description of the problem. This plot does not appear in the KENO printout. It is generated from a PNG file that is created when a color plot has been specified in the KENO input data and requires special processing by the user.

In Fig. 8.1.131, the plot title is printed at the top of the page, followed by a statement that "THE FOLLOWING WILL BE A COLOR PLOT." If a plot title was not entered in the plot data, the plot title is defaulted to the

problem title. The title is followed by a heading specifying the type of plot (MIXTURE MAP, BIAS ID MAP, or UNIT MAP). If the problem is a bare array, the overall system coordinates are printed. Then the coordinates of the upper left corner and lower right corner of the plot are printed. This is followed by the direction cosines down and across the plot. The remaining plot parameters (including both input data and calculated values) are then printed. NU is the number of characters printed in the U (down) direction, NV is the number of characters printed in the V (across) direction, DELU is the incremental distance, in cm, represented by each character in the U (down) direction, DELV is the incremental distance, in cm, represented by each character in the V (across) direction, and LPI is the vertical to horizontal scaling factor for plot proportionality.

```
  1f27 xy plot at z=0.0
                  the following will be a color plot

                                       mixture map

mixture  0 1 2 3 4
 symbol    1 2 3 4

      upper  left          lower right
      coordinates          coordinates

x     -7.1000e+01           7.1000e+01
y      7.1000e+01          -7.1000e+01
z      0.0000e+00           0.0000e+00

          u axis      v axis
          (down)      (across)

x          .00000      1.00000
y        -1.00000       .00000
z          .00000       .00000

nu= 640   nv= 640     delu= 2.2187e-01     delv= 2.2187e-01     lpi=  10.000
```

Fig. 8.1.131: Summary of color plot symbols, coordinates, and data.

Fig. 8.1.132 shows an example of a color plot of a 2-D slice specified through the geometrical description of the problem. Any number of plots can be made in one problem. The color plots can be valuable tools to assist the user in verifying that a problem is described correctly.

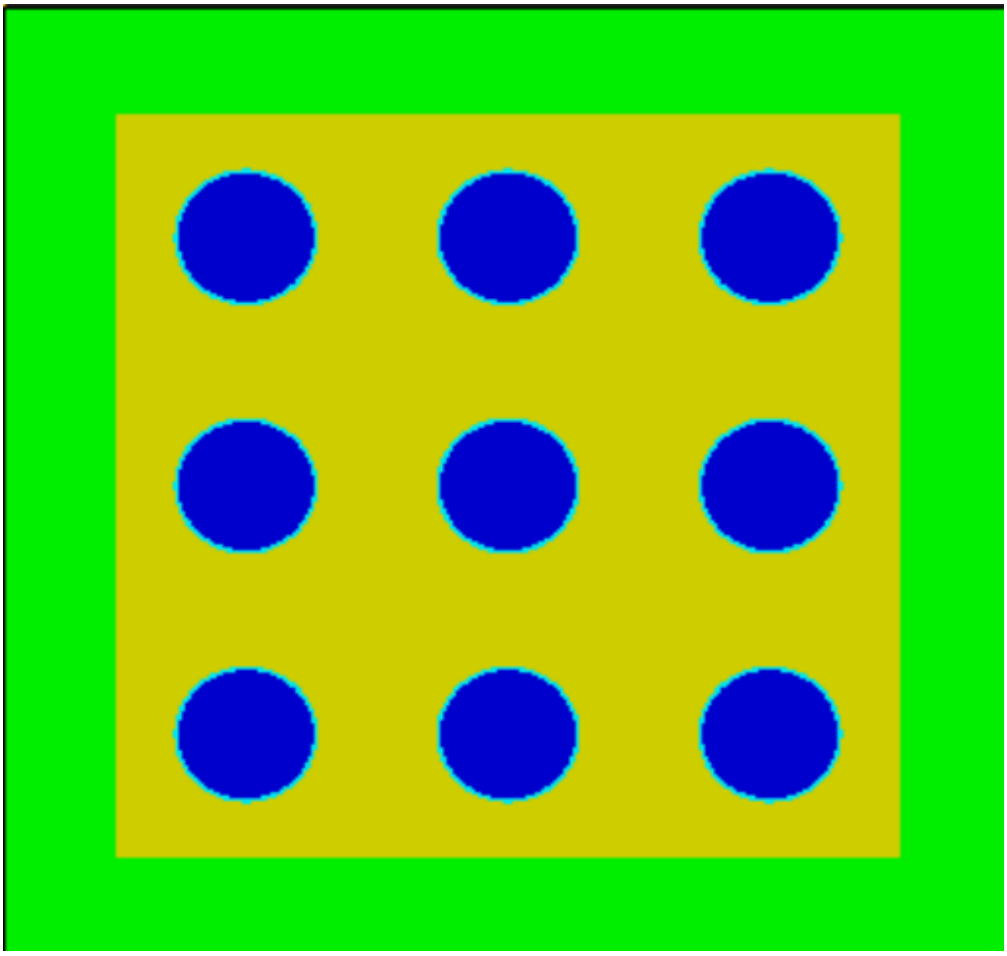


Fig. 8.1.132: Sample color plot representation.

8.1.5.23 Initial source and final pretracking edits

Prior to calculating the *keff* values for each generation, KENO prints the number of I/Os used before tracking and the number of minutes used processing the data. If the problem is not a restart problem, initial source information follows those two lines as shown in both Fig. 8.1.133 (KENO V.a example) and Fig. 8.1.134 (KENO-VI example).

The VOLUME FRACTION OF FISSILE MATERIAL IN THE SYSTEM is the first line of data printed for KENO V.a case. This is followed by the start type, other data used to generate the initial source distribution, and finally, the amount of time used to generate the initial source distribution and the total elapsed time. Fig. 8.1.133 illustrates starting information printed for start type 0. The cuboid for choosing starting points is (1) the bounding box around the global array (either bare or reflected) (2) the overall system if the outermost region is a cuboid, or (2) a cuboid specified by the user via the keywords **XSM**=, **XSP**=, **YSM**=, **YSP**=, **ZSM**=, and **ZSP**=.

Fig. 8.1.133 illustrates typical starting data for start type 0 (a flat start in fissile material). The parameter used in this example was **NST**=0.

```
                              ........ finished in KENO-V.a before tracking       ........

                              ........     0.00017 minutes were used processing data.        ........

volume fraction of fissile material in the system= 1.00000E+00

start type 0 was used.

the neutrons were started with a flat distribution in a cuboid defined by:
                      +x= 3.00000E+00  -x=-3.00000E+00  +y= 1.00000E+00  -y=-1.00000E+00  +z= 2.00000E+00  -z=-2.00000E+00

  0.00017 minutes were required for starting.   total elapsed time is   0.00033 minutes.
```

Fig. 8.1.133: Example of initial source information for start type 0 (KENO V.a).

In KENO-VI case, the first line is a warning message followed by the VOLUME FRACTION OF FISSILE MATERIAL IN THE SYSTEM line. This warning message notifies users about the potential issues that might be encountered when the volume fraction of fissile material in the system is calculated, especially for the cases in which volumes are partially defined or computed. In such cases, the volume fraction of fissile could be a negative values since KENO-VI resets the undefined volumes to -1.0.

This message is followed by the start type, other data used to generate the initial source distribution, and finally, the amount of time used to generate the initial source distribution and the total elapsed time. Fig. 8.1.134 illustrates the starting information printed for start type 0.

```
                              ........ finished in KENO-VI before tracking       ........

                              ........     0.00017 minutes were used processing data.        ........

 ***** warning ***** keno message number k6-380 follows:
 Calculated volume fraction of fissile material in the system reported below may not reflect
 the actual value since some region volumes were neither specified nor calculated. Even negative
 volume fraction for fissile material may be reported since all undefined volumes were set to -1.0.


volume fraction of fissile material in the system= 1.00000E+00

start type 0 was used.

the neutrons were started with a flat distribution in a cuboid defined by:
                      +x= 3.00000E+00  -x=-3.00000E+00  +y= 1.00000E+00  -y=-1.00000E+00  +z= 2.00000E+00  -z=-2.00000E+00

  0.00000 minutes were required for starting.   total elapsed time is   0.00017 minutes.
```

Fig. 8.1.134: Example of initial source information for start type 0 (KENO-VI).

8.1.5.24 Reference center for flux moment/angular flux transform

When **TFM**=YES is entered, the flux moments and angular fluxes are computed in a transformed coordinate system, relative to a "reference center" point. The default position for the reference center is the center of all fueled regions in the model. However, the center can be specified by the user for each region defined in the system model using the "center" modifier on a region input card. Edits of the fuel center and the position of the reference center for each region are shown when **TFM**=YES as shown in Fig. 8.1.135.

```
Fuel Center is x= 6.5307E+01, y= 1.7801E+01, z= 4.8576E+01 Relative to the Global Unit

                                reference center for flux moment/angular flux transform

      unit     region     option      relative to         x-offset       y-offset       z-offset

         1          1         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    2         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    3         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    4         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    5         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01

         2          1         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    2         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    3         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01

         3          1         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    2         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01

         4          1         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    2         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    3         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01

         5          1         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    2         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01

         6          1         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
                    2         -2       fuel center         6.5307E+01     1.7801E+01     4.8576E+01
```

Fig. 8.1.135: Example fuel center and reference center edits.

8.1.5.25 Print starting points

This printout is optional and is used to verify the initial source starting points. This option is activated by specifying the parameter **PSP**=YES in **start data** block, and it is valid with all start types available. Examples of this information are given in Fig. 8.1.136.

The information pertinent to the initial source distribution is printed three lines at a time and appears under the designated headings. For example, the coordinates X, Y, and Z are printed on one line, the direction cosines U, V, and W are printed directly under them, and the global coordinates XGBL, YGBL, and ZGBL on the last line. The data printed for each source neutron include the following:

NEUTRON is the ID number of the neutron.

X, Y, and Z are the coordinates of the starting point relative to the coordinate system of the unit.

WT is the starting weight (WT is always 1.0 for a neutron when it is started).

K is the region number that contains the point X, Y, Z.

NOW is the array number.

NBX, NBY, NBZ are the coordinates of the unit within the array.

LL is the unit number located at NBX, NBY, NBZ.

KR is the mixture number present at the starting point.

KI is the bias ID number or importance region at the starting point.

KCOR is the region in the unit that contains the array where the neutron is located.

U, V, and W are the direction cosines defining the direction the history is traveling.

IG is the energy group.

K1 is the region number of the first region in unit LL.

K2 is the region number of the last region in unit LL.

IGEO is an integer that defines the geometry shape of the region.

KCOL is the region number in which the last collision occurred.

RANDOM NUMBER is the current random number.

XGBL, YGBL, ZGBL are the coordinates of the starting point relative to the coordinate system of the **global unit**.

UGBL, VGBL, WGBL are the direction cosines defining the direction the history is traveling in the global coordinate system.

Note: When starting points are printed, **many of the above named variables have not been initialized**. For example, RANDOM NUMBER does not reflect the random number used to sample the starting point whose details are printed.

For starting, the variables of interest are X, Y, Z, NBX, NBY, NBZ, LL, XGBL, YGBL, and, ZGBL.

```
  starting points will be printed using the debug tracking format because start parameter psp= was entered as yes.
  neutron is the id number of the neutron.
  x, y and z are the coordinates of the starting point.
  wt is the starting weight.
  k is the region number.
  now is the array number.
  nbx, nby and nbz define a location in the now array.
  ll is the unit number at location nbx, nby, nbz.
  kr is the media or mixture number in region k.
  ki is the importance region.
  kcor is the region number surrounding the now array.
  u, v and w are the direction cosines defining the direction.
  ig is the energy group.
  k1 is the region number of the first region in unit ll.
  k2 is the region number of the last region in unit ll.
  igeo defines the geometry of region k.
  kcol is the region number of the last collision.
  random number is the current random number.
  xgbl, ygbl, and zgbl are the global coordinates of starting point.
  ugbl, vgbl, and wgbl are the direction cosines in the global coordinate system.

 1where    neutron            x             y             z             wt          k  now  nbx  nby  nbz   ll   kr   ki kcor
                              u             v             w                        ig   k1   k2 igeo kcol      random number
                              xgbl          ygbl          zgbl           ugbl           vgbl           wgbl

  start        100         2.00000E+00   3.00000E+00   0.00000E+00   1.00000E+00    1    0    0    0    0    1    1    1    0
                           5.12119E-01   5.84406E-01   6.29447E-01                  1    0    0    0    0    EB43716CB242507E
                           2.00000E+00   3.00000E+00   0.00000E+00   0.00000E+00    0.00000E+00    0.00000E+00

  start         99         2.00000E+00   3.00000E+00   0.00000E+00   1.00000E+00    1    0    0    0    0    1    1    1    0
                           6.91446E-01   7.08928E-01   1.39009E-01                  2    0    0    0    0    EB43716CB242507E
                           2.00000E+00   3.00000E+00   0.00000E+00   0.00000E+00    0.00000E+00    0.00000E+00
```

Fig. 8.1.136: Example of initial source points.

8.1.5.26 K-effectives by generation

At the completion of each generation, KENO prints the *keff* for that generation and associated information. An example of this printout is given in Fig. 8.1.137.

```
1sample problem 18   1f27 demonstration of options problem

             generation        average        avg k-eff       generation        matrix        matrix k-eff
  generation k-effective      k-effective      deviation        entropy       k-effective      deviation
        1    1.37343E+00    1.00000E+00    0.00000E+00    0.00000E+00    1.37343E+00    0.00000E+00
        2    1.12507E+00    1.00000E+00    0.00000E+00    2.21813E+00    1.12507E+00    0.00000E+00
        3    1.04953E+00    1.04953E+00    0.00000E+00    4.04101E+00    1.04953E+00    0.00000E+00
        4    1.04539E+00    1.04746E+00    2.06804E-03    4.88720E+00    1.04539E+00    0.00000E+00
        5    1.01200E+00    1.03564E+00    1.18783E-02    5.42853E+00    1.02870E+00    1.66931E-02
        6    9.73438E-01    1.02009E+00    1.76736E-02    5.81274E+00    1.01028E+00    2.07883E-02
        7    9.86846E-01    1.00442E+00    4.93099E-02    5.90986E+00    1.00442E+00    1.58237E-02
        8    9.79738E-01    9.99483E-01    3.08141E-02    5.93798E+00    9.99483E-01    1.32136E-02
        9    1.03932E+00    1.00612E+00    2.15395E-02    5.90015E+00    1.00612E+00    1.26678E-02
       10    1.02002E+00    1.00811E+00    1.86916E-02    5.98197E+00    1.00811E+00    1.08888E-02
       11    9.95417E-01    1.00652E+00    1.48057E-02    5.92588E+00    1.00652E+00    9.56242E-03
       12    9.90089E-01    1.00469E+00    1.29409E-02    5.95920E+00    1.00469E+00    8.62863E-03
       13    9.82895E-01    1.00251E+00    1.19083E-02    5.97106E+00    1.00251E+00    8.01966E-03
       14    9.83588E-01    1.00079E+00    1.11151E-02    5.96738E+00    1.00079E+00    7.45533E-03
       15    9.87757E-01    9.99708E-01    1.02239E-02    5.89743E+00    9.99708E-01    6.89192E-03
       16    1.00830E+00    1.00037E+00    9.19028E-03    5.93295E+00    1.00037E+00    6.37402E-03
       17    1.00698E+00    1.00084E+00    8.47007E-03    5.91416E+00    1.00084E+00    5.92007E-03
       18    1.01892E+00    1.00205E+00    8.04971E-03    5.86822E+00    1.00205E+00    5.64154E-03
       19    1.03531E+00    1.00413E+00    8.29237E-03    5.90224E+00    1.00413E+00    5.67193E-03
       20    1.01017E+00    1.00448E+00    7.85630E-03    5.88509E+00    1.00448E+00    5.33971E-03
       21    1.00173E+00    1.00433E+00    7.33186E-03    5.92249E+00    1.00433E+00    5.03666E-03
       22    1.04402E+00    1.00642E+00    7.25842E-03    5.92151E+00    1.00642E+00    5.20214E-03
       23    1.00731E+00    1.00646E+00    6.85749E-03    5.93526E+00    1.00646E+00    4.93539E-03
       24    1.02493E+00    1.00734E+00    6.57917E-03    5.92403E+00    1.00734E+00    4.77615E-03
       25    1.01784E+00    1.00782E+00    6.34972E-03    5.96020E+00    1.00782E+00    4.57881E-03
       26    1.01512E+00    1.00814E+00    6.08232E-03    5.89705E+00    1.00814E+00    4.38672E-03
       27    9.96152E-01    1.00764E+00    5.78261E-03    5.99640E+00    1.00764E+00    4.22955E-03
       28    1.01943E+00    1.00811E+00    5.50345E-03    5.98163E+00    1.00811E+00    4.08417E-03
       29    1.01000E+00    1.00818E+00    5.27679E-03    5.91891E+00    1.00818E+00    3.92461E-03
       30    1.03185E+00    1.00906E+00    5.44541E-03    5.93582E+00    1.00906E+00    3.87684E-03
       31    9.83940E-01    1.00816E+00    5.08944E-03    5.98745E+00    1.00816E+00    3.84201E-03
       32    1.01896E+00    1.00853E+00    4.91117E-03    5.98576E+00    1.00853E+00    3.72582E-03
       33    1.04234E+00    1.00966E+00    4.72878E-03    5.95528E+00    1.00966E+00    3.77176E-03
       34    1.01354E+00    1.00979E+00    4.60058E-03    5.95442E+00    1.00979E+00    3.65022E-03
       35    1.01942E+00    1.01009E+00    4.46799E-03    5.90714E+00    1.01009E+00    3.54711E-03
       36    1.00950E+00    1.01007E+00    4.31930E-03    5.85958E+00    1.01007E+00    3.43799E-03
       37    9.87975E-01    1.00942E+00    4.23564E-03    5.78317E+00    1.00942E+00    3.39805E-03
       38    1.01466E+00    1.00957E+00    4.08236E-03    5.88258E+00    1.00957E+00    3.30293E-03
       39    9.84770E-01    1.00888E+00    4.16167E-03    5.92782E+00    1.00888E+00    3.28295E-03
       40    9.95841E-01    1.00853E+00    4.10470E-03    5.89502E+00    1.00853E+00    3.21238E-03
       41    1.03694E+00    1.00927E+00    3.85863E-03    5.91203E+00    1.00927E+00    3.21487E-03
       42    1.01748E+00    1.00949E+00    3.80692E-03    5.88547E+00    1.00949E+00    3.13841E-03
 keno message number k5-132 follows:
only 2983 independent fission points were generated for generation 43
       43    9.83709E-01    1.00884E+00    3.72198E-03    5.88354E+00    1.00884E+00    3.12609E-03
       44    1.00402E+00    1.00872E+00    3.64936E-03    5.84766E+00    1.00872E+00    3.05115E-03
       45    1.01016E+00    1.00876E+00    3.55587E-03    5.84029E+00    1.00876E+00    2.97782E-03
       46    1.02745E+00    1.00919E+00    3.50519E-03    5.92593E+00    1.00919E+00    2.94005E-03
       47    1.04248E+00    1.00995E+00    3.61710E-03    5.89003E+00    1.00995E+00    2.97044E-03
       48    1.02258E+00    1.01023E+00    3.61061E-03    5.92747E+00    1.01023E+00    2.91722E-03
       49    9.94772E-01    1.00989E+00    3.51374E-03    5.92852E+00    1.00989E+00    2.87282E-03
       50    9.84846E-01    1.00936E+00    3.53234E-03    5.90019E+00    1.00936E+00    2.86110E-03
       51    9.94389E-01    1.00905E+00    3.52002E-03    5.87279E+00    1.00905E+00    2.81817E-03
       52    1.00847E+00    1.00904E+00    3.44520E-03    5.90029E+00    1.00904E+00    2.76009E-03
       53    1.00437E+00    1.00894E+00    3.37374E-03    5.89811E+00    1.00894E+00    2.70593E-03
       54    9.87656E-01    1.00853E+00    3.34307E-03    5.94600E+00    1.00853E+00    2.68498E-03
       55    9.95486E-01    1.00828E+00    3.31884E-03    5.93614E+00    1.00828E+00    2.64476E-03
       56    1.01496E+00    1.00840E+00    3.24562E-03    5.90015E+00    1.00840E+00    2.59744E-03
       57    9.99005E-01    1.00823E+00    3.17933E-03    5.95567E+00    1.00823E+00    2.55482E-03
       58    9.84745E-01    1.00780E+00    3.17338E-03    5.91686E+00    1.00780E+00    2.54402E-03
       59    9.88469E-01    1.00746E+00    3.18316E-03    5.95380E+00    1.00746E+00    2.52192E-03
       60    9.66947E-01    1.00674E+00    3.28852E-03    5.98986E+00    1.00674E+00    2.57720E-03
       61    1.03647E+00    1.00726E+00    3.15276E-03    5.87920E+00    1.00726E+00    2.58373E-03
       62    1.02060E+00    1.00748E+00    3.14631E-03    5.90762E+00    1.00748E+00    2.54961E-03
       63    1.01251E+00    1.00757E+00    3.09953E-03    5.86758E+00    1.00757E+00    2.50816E-03
       64    1.00833E+00    1.00758E+00    3.04685E-03    5.89236E+00    1.00758E+00    2.46673E-03
       65    1.04489E+00    1.00818E+00    3.06417E-03    5.95750E+00    1.00818E+00    2.50013E-03
       66    1.00053E+00    1.00806E+00    2.98962E-03    5.94444E+00    1.00806E+00    2.46312E-03
       67    9.59864E-01    1.00731E+00    3.07132E-03    5.92222E+00    1.00731E+00    2.53860E-03
       68    1.03001E+00    1.00766E+00    2.95119E-03    5.90115E+00    1.00766E+00    2.52352E-03
       69    1.02042E+00    1.00785E+00    2.93668E-03    5.91423E+00    1.00785E+00    2.49251E-03
       70    9.90063E-01    1.00758E+00    2.88429E-03    5.91154E+00    1.00758E+00    2.46933E-03
       71    9.96137E-01    1.00742E+00    2.86157E-03    5.86176E+00    1.00742E+00    2.43857E-03
       72    9.84541E-01    1.00708E+00    2.85884E-03    5.83867E+00    1.00708E+00    2.42573E-03
       73    1.01623E+00    1.00722E+00    2.80369E-03    5.89851E+00    1.00722E+00    2.39439E-03
       74    9.84662E-01    1.00690E+00    2.76516E-03    5.88324E+00    1.00690E+00    2.38170E-03
       75    9.98290E-01    1.00678E+00    2.74243E-03    5.92248E+00    1.00678E+00    2.35143E-03
       76    1.00471E+00    1.00675E+00    2.70486E-03    5.92432E+00    1.00675E+00    2.31917E-03
       77    1.03739E+00    1.00716E+00    2.69731E-03    5.94053E+00    1.00716E+00    2.32479E-03
       78    1.02658E+00    1.00742E+00    2.71602E-03    5.93406E+00    1.00742E+00    2.30814E-03
       79    1.01679E+00    1.00755E+00    2.69485E-03    5.92118E+00    1.00755E+00    2.28091E-03
       80    1.06020E+00    1.00823E+00    2.82317E-03    5.84635E+00    1.00823E+00    2.35266E-03
       81    1.02221E+00    1.00841E+00    2.85619E-03    5.86683E+00    1.00841E+00    2.32920E-03
       82    1.02794E+00    1.00866E+00    2.85873E-03    5.90182E+00    1.00866E+00    2.31279E-03
       83    1.02843E+00    1.00890E+00    2.91837E-03    5.85544E+00    1.00890E+00    2.29703E-03
       84    9.88707E-01    1.00865E+00    2.85215E-03    5.91539E+00    1.00865E+00    2.28216E-03
       85    9.83353E-01    1.00835E+00    2.83410E-03    5.89386E+00    1.00835E+00    2.27517E-03
       86    9.95756E-01    1.00819E+00    2.80634E-03    5.84798E+00    1.00819E+00    2.25271E-03
       87    1.02967E+00    1.00845E+00    2.74541E-03    5.85808E+00    1.00845E+00    2.24037E-03
       88    1.02775E+00    1.00868E+00    2.70856E-03    5.90049E+00    1.00868E+00    2.22547E-03
       89    1.02323E+00    1.00885E+00    2.69698E-03    5.89188E+00    1.00885E+00    2.20594E-03
       90    1.02619E+00    1.00905E+00    2.68639E-03    5.90053E+00    1.00905E+00    2.18953E-03
       91    1.02288E+00    1.00920E+00    2.70569E-03    5.87532E+00    1.00920E+00    2.17021E-03
       92    9.95527E-01    1.00905E+00    2.63626E-03    5.90908E+00    1.00905E+00    2.15118E-03
       93    1.01566E+00    1.00912E+00    2.60277E-03    5.83510E+00    1.00912E+00    2.12842E-03
       94    9.99760E-01    1.00902E+00    2.57220E-03    5.92091E+00    1.00902E+00    2.10741E-03
       95    1.03077E+00    1.00926E+00    2.54533E-03    5.93108E+00    1.00926E+00    2.09775E-03
       96    9.95534E-01    1.00911E+00    2.50691E-03    5.89062E+00    1.00911E+00    2.08031E-03
       97    1.01657E+00    1.00919E+00    2.47600E-03    5.94039E+00    1.00919E+00    2.05959E-03
       98    1.02239E+00    1.00933E+00    2.45824E-03    5.98176E+00    1.00933E+00    2.04253E-03
       99    1.00535E+00    1.00929E+00    2.42973E-03    5.92882E+00    1.00929E+00    2.02157E-03
      100    1.03329E+00    1.00953E+00    2.41322E-03    5.86505E+00    1.00953E+00    2.01586E-03
      101    1.03435E+00    1.00979E+00    2.46459E-03    5.95519E+00    1.00979E+00    2.01119E-03
      102    1.01763E+00    1.00987E+00    2.44757E-03    5.98694E+00    1.00987E+00    1.99235E-03
      103    9.87762E-01    1.00964E+00    2.39213E-03    5.95038E+00    1.00964E+00    1.98467E-03

 keno message number k5-123          execution terminated due to completion of the specified number of generations.

 the matrix k-effective is the largest eigenvalue of the fission production by unit number matrix.
```

Fig. 8.1.137: Example of k-effectives and source entropy by generation.

The data printed include (1) the generation number, (2) the k-effective calculated for the generation, (3) the average value of k-effective through the current generation (excluding the *nskip-1* generations), (4) the deviation associated with the average k-effective, (5) Shannon entropy for the generation, (6) the matrix k-effective for the generation, and (7) the deviation associated with the matrix k-effective. Column 5 is omitted if the user disabled source convergence diagnostics. The last two columns are filled with zeros if the user did not specify matrix k-effective calculations. The matrix k-effective is the largest eigenvalue of the fission production matrix. Matrix information can be calculated based on (1) position index, (2) unit number, (3) hole number, and (4) array number. The matrix k-effective printed in the sixth column is based on this order. If the matrix k-effective is calculated by position index, it is the one printed in the sixth column. The matrix k-effective by unit number is given second preference, followed by hole number and then array number.

After the last generation, a message is printed to indicate why execution was terminated. If matrix k-effectives were calculated, this is followed by a message stating the method used to determine the matrix k-effective.

The user should examine this portion of the printed results to ensure that the two methods of calculating k-effective are in acceptable agreement and to verify that the average value of k-effective has become relatively stable. If the k-effectives appear to be oscillating or drifting significantly, then the user should consider rerunning the problem with a larger number of histories per generation.

If a problem is restarted, then the generation numbers and k-effectives are printed and the words FROM RESTART UNIT are printed in the elapsed time column. All other columns are blank. When the generation at which the problem is to be restarted is reached, the print reverts to the normal format as shown in Fig. 8.1.137.

8.1.5.27 Problem characterization edit

The problem characterization edit follows the k-effective by generation edit. The title is printed at the top of the page, followed by the lifetime and the generation time and their associated deviations. The lifetime is the average lifespan of a neutron (in seconds) from the time it is born until it is absorbed or leaks from the system. The generation time is the average time (in seconds) between successive neutron generations. If **NUB**=YES is specified in the parameter data, (Sect. 8.1.3.3) the average number of neutrons per fission, NU BAR, and its associated deviation are printed and the AVERAGE FISSION GROUP (the average energy group at which fission occurs) and its associated deviation are printed. Then the ENERGY(EV) OF THE AVERAGE LETHARGY OF NEUTRONS CAUSING FISSION and its associated deviation are printed, followed by the system mean free path. If **SMU**=YES is specified in the parameter data, the average self-multiplication of a unit and its associated deviation is printed. This self-multiplication results from fissions caused by neutrons born in the unit. Fissions caused by neutrons that exit the unit and return are not included. The problem characterization edit is shown in Fig. 8.1.138.

```
1                                          sample problem 18   1f27 demonstration of options problem

 lifetime =  2.01416E-04 + or -  3.89547E-06            generation time =  1.56496E-04 + or -  1.34654E-06
 nu bar   =  2.44053E+00 + or -  1.63032E-05      average fission group =  2.11839E+02 + or -  5.12062E-02
                    energy(ev) of the average lethargy causing fission =  2.08321E-01 + or -  8.05618E-04
                                          system mean free path (cm) =  3.29281E-01 + or -  2.46316E-03
                                                self multiplication =  5.34926E-01 + or -  5.12336E-03
```

Fig. 8.1.138: Example problem characterization edit.

8.1.5.28 Final k-effective edit

The final k-effective edit prints the average k-effective and its associated deviation and the limits of k-effective for the 67, 95, and 99% confidence intervals. The number of histories used in calculating the average k-effective is also printed. This is done skipping various numbers of generations. The user should carefully examine the final k-effective edit to determine if the average k-effective is relatively stable. If a noticeable drift is apparent as the number of initial generations skipped increases, it may indicate a problem in converging the source. If this appears to be the case, the problem should be rerun with a better initial source distribution and should be run for a sufficient number of generations that the average k-effective becomes stable. The final k-effective edit is printed as shown in Fig. 8.1.139.

no. of initial generations skipped	average k-effective	deviation	67 per cent confidence interval	95 per cent confidence interval	99 per cent confidence interval	number of histories	deviation of variance (per cent)
3	1.00964	+ or - 0.00239	1.00725 to 1.01204	1.00486 to 1.01443	1.00247 to 1.01682	300000	8.4093
4	1.00928	+ or - 0.00243	1.00685 to 1.01171	1.00442 to 1.01414	1.00199 to 1.01657	297000	8.1551
5	1.00926	+ or - 0.00246	1.00679 to 1.01172	1.00433 to 1.01418	1.00186 to 1.01665	294000	8.0697
6	1.00962	+ or - 0.00247	1.00715 to 1.01210	1.00468 to 1.01457	1.00221 to 1.01704	291000	8.0282
7	1.00986	+ or - 0.00246	1.00740 to 1.01233	1.00493 to 1.01479	1.00247 to 1.01725	288000	8.2463
8	1.01018	+ or - 0.00249	1.00769 to 1.01267	1.00520 to 1.01516	1.00271 to 1.01765	285000	8.1879
9	1.00987	+ or - 0.00251	1.00736 to 1.01238	1.00485 to 1.01489	1.00233 to 1.01740	282000	8.1717
10	1.00976	+ or - 0.00256	1.00720 to 1.01232	1.00464 to 1.01488	1.00209 to 1.01743	279000	8.0337
11	1.00992	+ or - 0.00255	1.00737 to 1.01246	1.00482 to 1.01501	1.00227 to 1.01756	276000	8.2745
12	1.01013	+ or - 0.00251	1.00762 to 1.01265	1.00510 to 1.01516	1.00259 to 1.01768	273000	8.6905
17	1.01108	+ or - 0.00253	1.00855 to 1.01361	1.00601 to 1.01614	1.00348 to 1.01867	258000	9.5177
22	1.01040	+ or - 0.00260	1.00780 to 1.01300	1.00521 to 1.01559	1.00261 to 1.01819	243000	9.9290
27	1.01028	+ or - 0.00280	1.00748 to 1.01308	1.00468 to 1.01587	1.00189 to 1.01867	228000	9.6209
32	1.01010	+ or - 0.00296	1.00714 to 1.01306	1.00418 to 1.01602	1.00121 to 1.01898	213000	9.7290
37	1.00976	+ or - 0.00319	1.00657 to 1.01296	1.00337 to 1.01615	1.00018 to 1.01935	198000	9.4523
42	1.00975	+ or - 0.00330	1.00644 to 1.01305	1.00314 to 1.01635	0.99984 to 1.01965	183000	10.2488
47	1.00940	+ or - 0.00378	1.00562 to 1.01319	1.00184 to 1.01697	0.99805 to 1.02076	168000	9.0268
52	1.01023	+ or - 0.00391	1.00631 to 1.01414	1.00240 to 1.01806	0.99849 to 1.02197	153000	10.1599
57	1.01131	+ or - 0.00398	1.00733 to 1.01528	1.00335 to 1.01926	0.99938 to 1.02324	138000	12.1314
62	1.01275	+ or - 0.00402	1.00873 to 1.01678	1.00470 to 1.02080	1.00068 to 1.02483	123000	13.7684
67	1.01380	+ or - 0.00442	1.00938 to 1.01822	1.00496 to 1.02264	1.00054 to 1.02706	108000	10.1017
72	1.01534	+ or - 0.00434	1.01100 to 1.01968	1.00666 to 1.02403	1.00231 to 1.02837	93000	13.6645
77	1.01670	+ or - 0.00466	1.01204 to 1.02136	1.00738 to 1.02603	1.00272 to 1.03069	78000	16.2167
82	1.01336	+ or - 0.00449	1.00887 to 1.01785	1.00438 to 1.02234	0.99989 to 1.02683	63000	13.2196
87	1.01592	+ or - 0.00394	1.01198 to 1.01985	1.00805 to 1.02379	1.00411 to 1.02772	48000	24.6692
92	1.01446	+ or - 0.00523	1.00923 to 1.01970	1.00399 to 1.02493	0.99876 to 1.03016	33000	27.2013

1 sample problem 18 1f27 demonstration of options problem

no. of initial generations skipped	average k-effective	deviation	67 per cent confidence interval	95 per cent confidence interval	99 per cent confidence interval	number of histories	deviation of variance (per cent)
97	1.01680	+ or - 0.01174	1.00506 to 1.02853	0.99332 to 1.04027	0.98158 to 1.05201	18000	22.9106

Fig. 8.1.139: Example of the final k-effective edit.

8.1.5.29 Plot of average k-effective by generation run

This plot consists of average k-effectives plotted versus the number of generations run. The limits of one standard deviation are plotted on either side of each average k-effective. These average k-effectives are not necessarily the same as the average k-effectives described in Sect. 8.1.5.26. The code omits the k-effectives of the first *nskip* generations when the average k-effectives for this plot are calculated. Although the k-effective of the *nskip* generation is summed into the average k-effective, it is not plotted because standard deviations cannot be calculated for a single point. Thus, if *nskip* is 3 (i.e., the first three generations are skipped), the first value plotted is the average k-effective corresponding to the fifth generation. The dotted line represents the value of the average k-effective corresponding to the smallest deviation when the average k-effective and its deviation are computed for each generation over the range of *nskip* through the total number of generations.

Fig. 8.1.140 is an example of this type of plot in the text output. The primary use for this plot is to determine if the problem has source convergence difficulties.

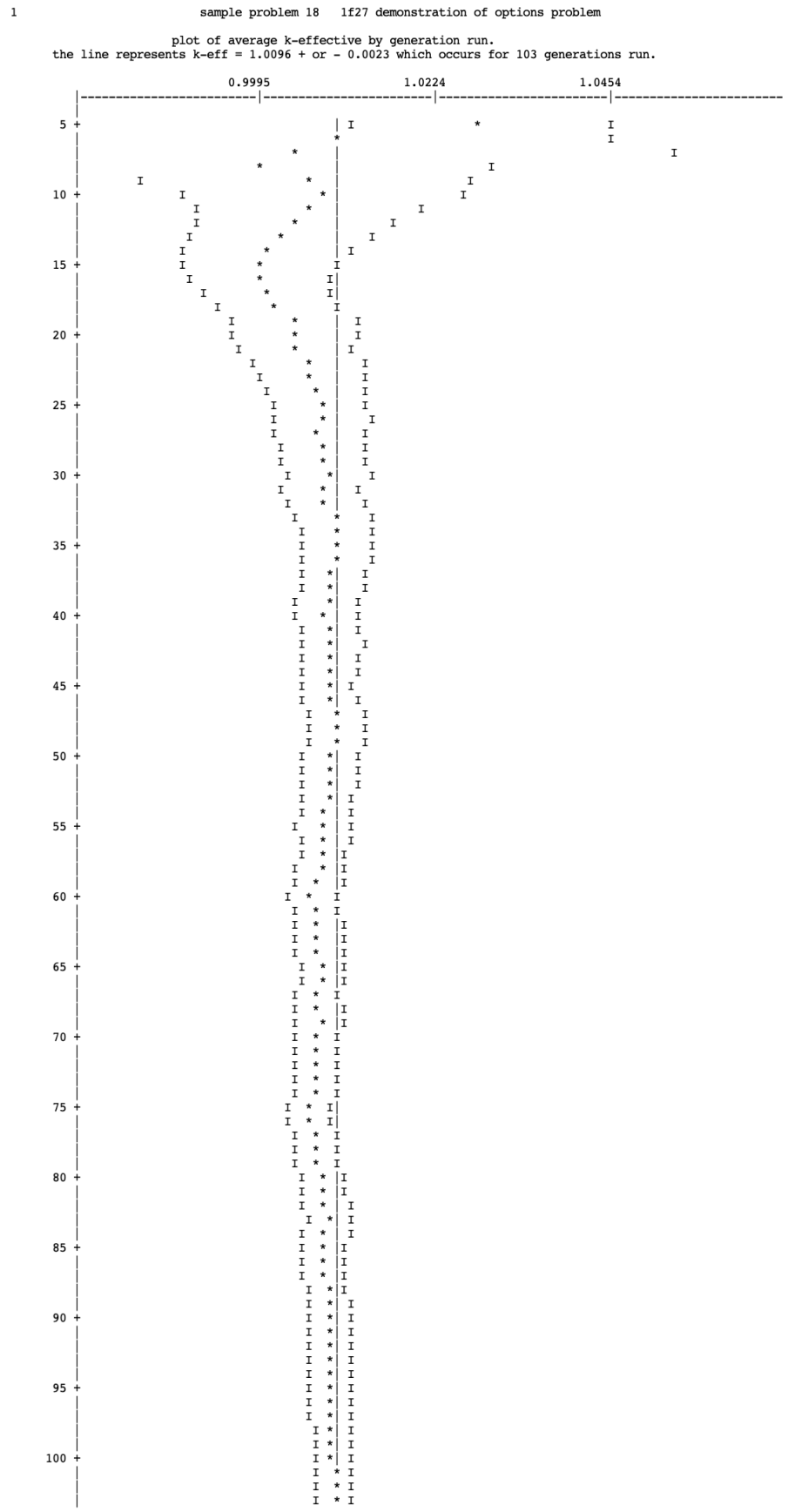


Fig. 8.1.140: Sample plot of average k-effective by generation run.

8.1.5.30 Plot of average k-effective by generations skipped

This plot illustrates the average k-effective versus the number of generations skipped as shown in Fig. 8.1.141. The limits of one standard deviation are plotted on either side of the average k-effective. The dotted line represents the value of the average k-effective corresponding to the smallest deviation when the average k-effective and its deviation are computed for the number of generations skipped over the range of *nskip*+1 through $^2/_3$ the total number of generations calculated. The plot is essentially a plot of the data described in Sect. 8.1.5.28. It is useful for determining if source convergence has been achieved.

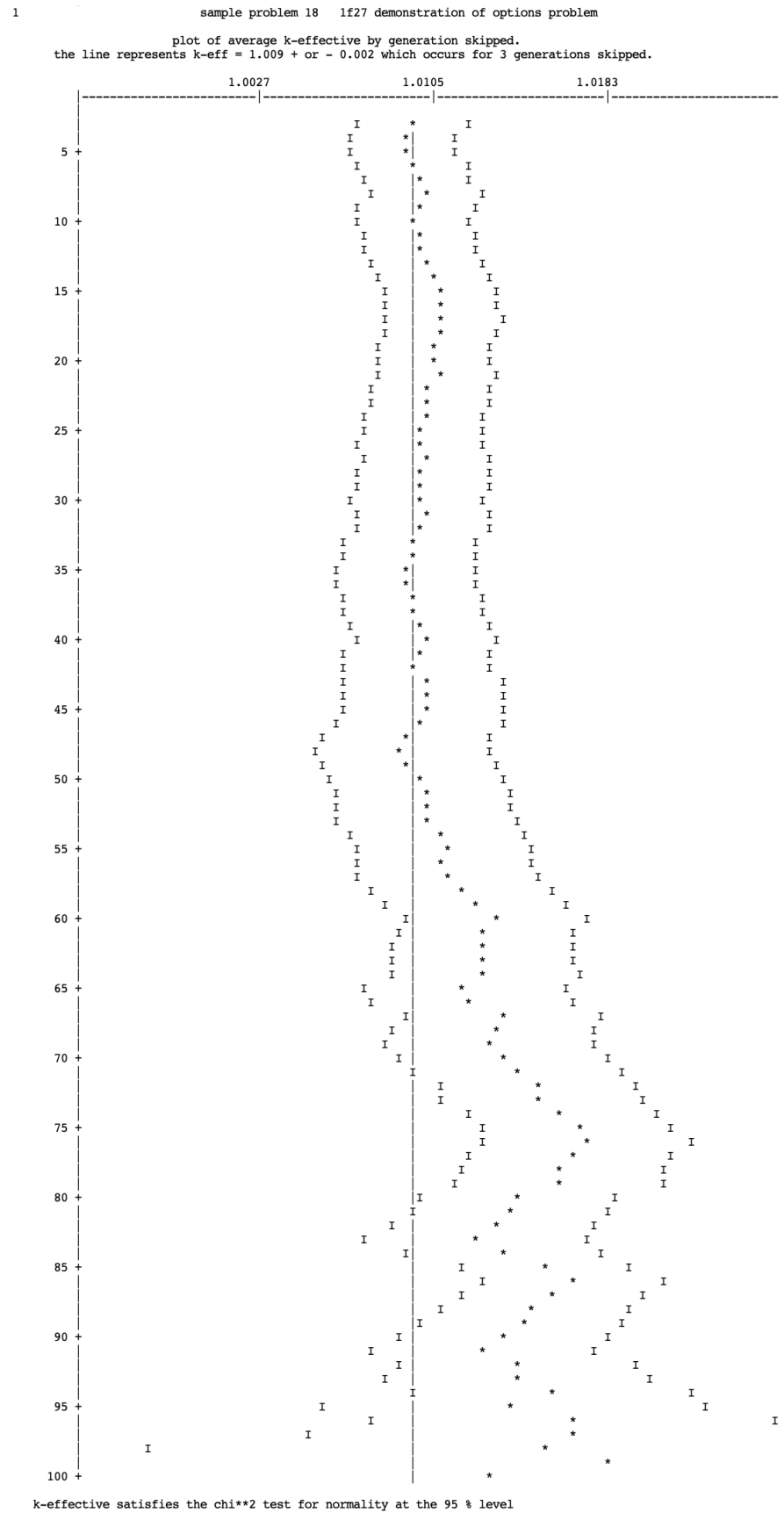


Fig. 8.1.141: Sample plot of average k-effective by generations skipped.

8.1.5.31 Final edit of fissions, absorptions, and leakage

The final edit of fissions, absorptions, and leakage follows the final k-effective edit and includes the fission fraction for each group and the fission production, absorptions, and leakage, each with its associated percent deviation. Examples of the final edits of fissions, absorptions, and leakage are shown in Fig. 8.1.142-Fig. 8.1.144.

If **FAR**=NO is specified, region-dependent fissions and absorptions are not printed. Fig. 8.1.142 demonstrates the printout if **FAR**=NO.

```
1                          sample problem 18   1f27 demonstration of options problem
                                                                                  skipping  3 generations

 group  fission  unit   region    fissions     percent       absorptions    percent         leakage      percent
        fraction                               deviation                    deviation                    deviation

    1    0.0000                 0.00000E+00     0.0000       0.00000E+00     0.0000       0.00000E+00     0.0000
    2    0.0000                 0.00000E+00     0.0000       0.00000E+00     0.0000       0.00000E+00     0.0000
    3    0.0000                 0.00000E+00     0.0000       0.00000E+00     0.0000       0.00000E+00     0.0000
    4    0.0000                 0.00000E+00     0.0000       0.00000E+00     0.0000       0.00000E+00     0.0000
    5    0.0000                 2.52047E-06    30.1511       3.99551E-06    25.3836       0.00000E+00     0.0000
    6    0.0001                 5.13912E-05     7.8068       1.12468E-04     6.4218       6.69783E-05    20.5476
    7    0.0002                 1.55029E-04     3.8580       2.91984E-04     3.1608       1.73723E-04    16.2801
    8    0.0005                 4.58707E-04     1.9129       5.64846E-04     1.7259       5.21995E-04     8.7294
    9    0.0009                 8.72597E-04     1.1043       9.41696E-04     1.0849       1.08036E-03     5.3918
   10    0.0005                 4.91001E-04     1.4073       5.77985E-04     1.3742       5.70486E-04     9.5987
   11    0.0023                 2.31031E-03     0.6455       1.81431E-03     0.6388       1.40680E-03     6.0927

  ...

  237    0.0848                 8.55887E-02     0.3748       9.97049E-02     1.1717       0.00000E+00     0.0000
  238    0.0425                 4.29020E-02     0.4963       5.30371E-02     1.2500       0.00000E+00     0.0000
  239    0.1370                 1.38352E-01     0.3891       1.86181E-01     1.3560       0.00000E+00     0.0000
  240    0.0192                 1.93769E-02     0.7003       2.89164E-02     1.5583       0.00000E+00     0.0000
  241    0.0168                 1.69847E-02     0.7843       2.66220E-02     1.5968       0.00000E+00     0.0000
  242    0.0057                 5.71381E-03     1.2315       9.25230E-03     2.1112       0.00000E+00     0.0000
  243    0.0052                 5.25186E-03     1.2956       8.61367E-03     2.4872       0.00000E+00     0.0000
  244    0.0023                 2.32178E-03     1.9370       4.01985E-03     3.9018       0.00000E+00     0.0000
  245    0.0021                 2.09092E-03     2.2432       3.53229E-03     2.5286       0.00000E+00     0.0000
  246    0.0018                 1.84013E-03     2.1985       3.13050E-03     3.0420       0.00000E+00     0.0000
  247    0.0009                 9.19754E-04     3.0665       1.59373E-03     3.5545       0.00000E+00     0.0000
  248    0.0005                 5.50119E-04     3.9924       1.08161E-03     4.0719       0.00000E+00     0.0000
  249    0.0006                 6.15877E-04     4.3337       1.14337E-03     7.1032       0.00000E+00     0.0000
  250    0.0005                 5.01928E-04     4.4143       8.80942E-04     3.4846       0.00000E+00     0.0000
  251    0.0004                 4.32411E-04     4.8759       9.59854E-04     4.6179       0.00000E+00     0.0000
  252    0.0000                 2.68928E-05    19.6909       3.64220E-05    39.5303       0.00000E+00     0.0000

system total =                  1.00964E+00     0.1966       9.83478E-01     0.8770       2.24251E-02    19.3544


the weight lost in the albedo portion of the problem =   2.0042E-03  + or -  0.0004


elapsed time   0.58667 minutes

random number=   11CE906036E5F7EC
```

Fig. 8.1.142: Sample of the final edit of fissions, absorptions, and leakage with all region-dependent information suppressed (**FAR**=NO).

```
1                          sample problem 18   1f27 demonstration of options problem
                                                                                                  skipping  3 generations

 group  fission  unit   region    fissions     percent       absorptions    percent         leakage      percent
        fraction                               deviation                    deviation                    deviation

    1    0.0000                  0.00000E+00     0.0000       0.00000E+00     0.0000       0.00000E+00     0.0000
                    1     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                          2      0.00000E+00     0.0000       0.00000E+00     0.0000
                          3      0.00000E+00     0.0000       0.00000E+00     0.0000
                          4      0.00000E+00     0.0000       0.00000E+00     0.0000
                    2     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                    3     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                    4     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                    5     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                    6     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                          2      0.00000E+00     0.0000       0.00000E+00     0.0000
                          3      0.00000E+00     0.0000       0.00000E+00     0.0000
                          4      0.00000E+00     0.0000       0.00000E+00     0.0000
                          5      0.00000E+00     0.0000       0.00000E+00     0.0000
                          6      0.00000E+00     0.0000       0.00000E+00     0.0000
                          7      0.00000E+00     0.0000       0.00000E+00     0.0000
       - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

  ...

  252    0.0000                  2.68928E-05    19.6909       3.64220E-05    39.5303       0.00000E+00     0.0000
                    1     1      2.68928E-05    19.6909       1.35736E-05    19.6909
                          2      0.00000E+00     0.0000       0.00000E+00     0.0000
                          3      0.00000E+00     0.0000       6.94580E-07    21.9739
                          4      0.00000E+00     0.0000       0.00000E+00     0.0000
                    2     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                    3     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                    4     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                    5     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                    6     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                          2      0.00000E+00     0.0000       3.15122E-06    21.8842
                          3      0.00000E+00     0.0000       3.15860E-07    74.4009
                          4      0.00000E+00     0.0000       4.67707E-06    70.6111
                          5      0.00000E+00     0.0000       1.40097E-05   100.0000
                          6      0.00000E+00     0.0000       0.00000E+00     0.0000
                          7      0.00000E+00     0.0000       0.00000E+00     0.0000
       - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
1                          sample problem 18   1f27 demonstration of options problem
                                                                                                  skipping  3 generations

                unit   region    fissions     percent       absorptions    percent         leakage      percent
                                               deviation                    deviation                    deviation

 system total =                 1.00964E+00     0.1966       9.83478E-01     0.8770       2.24251E-02    19.3544

                    1     1      1.00964E+00     0.0971       5.36686E-01     0.0932
                          2      0.00000E+00     0.0000       0.00000E+00     0.0000
                          3      0.00000E+00     0.0000       1.01053E-02     0.2210
                          4      0.00000E+00     0.0000       0.00000E+00     0.0000

                    2     1      0.00000E+00     0.0000       0.00000E+00     0.0000

                    3     1      0.00000E+00     0.0000       0.00000E+00     0.0000

                    4     1      0.00000E+00     0.0000       0.00000E+00     0.0000

                    5     1      0.00000E+00     0.0000       0.00000E+00     0.0000

                    6     1      0.00000E+00     0.0000       0.00000E+00     0.0000
                          2      0.00000E+00     0.0000       1.69256E-01     0.2000
                          3      0.00000E+00     0.0000       1.34186E-01     0.4824
                          4      0.00000E+00     0.0000       7.89655E-02     1.2110
                          5      0.00000E+00     0.0000       3.92324E-02     3.3516
                          6      0.00000E+00     0.0000       1.49093E-02     8.5648
                          7      0.00000E+00     0.0000       1.37407E-04    48.4906


 the weight lost in the albedo portion of the problem =   2.0042E-03  + or -  0.0004


 elapsed time   0.59183 minutes

 random number=   11CE906036E5F7EC
```

Fig. 8.1.143: Sample of the final edit of region-dependent fissions, absorptions, and leakage and the region-dependent totals (**FAR**=YES **GAS**=YES).

```
1                          sample problem 18   1f27 demonstration of options problem
                                                                                                  skipping  3 generations

 group  fission  unit   region    fissions     percent       absorptions    percent         leakage      percent
        fraction                                deviation                   deviation                    deviation

    1    0.0000                 0.00000E+00     0.0000       0.00000E+00     0.0000       0.00000E+00     0.0000
    2    0.0000                 0.00000E+00     0.0000       0.00000E+00     0.0000       0.00000E+00     0.0000
    3    0.0000                 0.00000E+00     0.0000       0.00000E+00     0.0000       0.00000E+00     0.0000
    4    0.0000                 0.00000E+00     0.0000       0.00000E+00     0.0000       0.00000E+00     0.0000
    5    0.0000                 2.52047E-06    30.1511       3.99551E-06    25.3836       0.00000E+00     0.0000
    6    0.0001                 5.13912E-05     7.8068       1.12468E-04     6.4218       6.69783E-05    20.5476
    7    0.0002                 1.55029E-04     3.8580       2.91984E-04     3.1608       1.73723E-04    16.2801
    8    0.0005                 4.58707E-04     1.9129       5.64846E-04     1.7259       5.21995E-04     8.7294
    9    0.0009                 8.72597E-04     1.1043       9.41696E-04     1.0849       1.08036E-03     5.3918
   10    0.0005                 4.91001E-04     1.4073       5.77985E-04     1.3742       5.70486E-04     9.5987
   11    0.0023                 2.31031E-03     0.6455       1.81431E-03     0.6388       1.40680E-03     6.0927

  ...

  232    0.0321                 3.24196E-02     0.4985       2.98478E-02     0.9727       0.00000E+00     0.0000
  233    0.0396                 3.99764E-02     0.4153       3.83855E-02     0.8955       5.14063E-04   100.0000
  234    0.0493                 4.97934E-02     0.4751       4.97612E-02     0.9880       0.00000E+00     0.0000
  235    0.0609                 6.15104E-02     0.4136       6.39036E-02     1.0283       0.00000E+00     0.0000
  236    0.0727                 7.34247E-02     0.3781       8.05670E-02     1.0703       3.67134E-04   100.0000
  237    0.0848                 8.55887E-02     0.3748       9.97049E-02     1.1717       0.00000E+00     0.0000
  238    0.0425                 4.29020E-02     0.4963       5.30371E-02     1.2500       0.00000E+00     0.0000
  239    0.1370                 1.38352E-01     0.3891       1.86181E-01     1.3560       0.00000E+00     0.0000
  240    0.0192                 1.93769E-02     0.7003       2.89164E-02     1.5583       0.00000E+00     0.0000
  241    0.0168                 1.69847E-02     0.7843       2.66220E-02     1.5968       0.00000E+00     0.0000
  242    0.0057                 5.71381E-03     1.2315       9.25230E-03     2.1112       0.00000E+00     0.0000
  243    0.0052                 5.25186E-03     1.2956       8.61367E-03     2.4872       0.00000E+00     0.0000
  244    0.0023                 2.32178E-03     1.9370       4.01985E-03     3.9018       0.00000E+00     0.0000
  245    0.0021                 2.09092E-03     2.2432       3.53229E-03     2.5286       0.00000E+00     0.0000
  246    0.0018                 1.84013E-03     2.1985       3.13050E-03     3.0420       0.00000E+00     0.0000
  247    0.0009                 9.19754E-04     3.0665       1.59373E-03     3.5545       0.00000E+00     0.0000
  248    0.0005                 5.50119E-04     3.9924       1.08161E-03     4.0719       0.00000E+00     0.0000
  249    0.0006                 6.15877E-04     4.3337       1.14337E-03     7.1032       0.00000E+00     0.0000
  250    0.0005                 5.01928E-04     4.4143       8.80942E-04     3.4846       0.00000E+00     0.0000
  251    0.0004                 4.32411E-04     4.8759       9.59854E-04     4.6179       0.00000E+00     0.0000
  252    0.0000                 2.68928E-05    19.6909       3.64220E-05    39.5303       0.00000E+00     0.0000
1                          sample problem 18   1f27 demonstration of options problem
                                                                                                  skipping  3 generations

                unit   region    fissions     percent       absorptions    percent         leakage      percent
                                                deviation                   deviation                    deviation

 system total =                 1.00964E+00     0.1966       9.83478E-01     0.8770       2.24251E-02    19.3544

                   1      1     1.00964E+00     0.0971       5.36686E-01     0.0932
                          2     0.00000E+00     0.0000       0.00000E+00     0.0000
                          3     0.00000E+00     0.0000       1.01053E-02     0.2210
                          4     0.00000E+00     0.0000       0.00000E+00     0.0000

                   2      1     0.00000E+00     0.0000       0.00000E+00     0.0000

                   3      1     0.00000E+00     0.0000       0.00000E+00     0.0000

                   4      1     0.00000E+00     0.0000       0.00000E+00     0.0000

                   5      1     0.00000E+00     0.0000       0.00000E+00     0.0000

                   6      1     0.00000E+00     0.0000       0.00000E+00     0.0000
                          2     0.00000E+00     0.0000       1.69256E-01     0.2000
                          3     0.00000E+00     0.0000       1.34186E-01     0.4824
                          4     0.00000E+00     0.0000       7.89655E-02     1.2110
                          5     0.00000E+00     0.0000       3.92324E-02     3.3516
                          6     0.00000E+00     0.0000       1.49093E-02     8.5648
                          7     0.00000E+00     0.0000       1.37407E-04    48.4906


 the weight lost in the albedo portion of the problem =   2.0042E-03  + or -  0.0004


 elapsed time   0.59100 minutes

 random number=   11CE906036E5F7EC
```

Fig. 8.1.144: Sample of the final edit of fissions, absorptions, and leakage and the region-dependent totals with print by region suppressed (**FAR**=YES **GAS**=NO).

If **FAR**=YES is specified in the parameter data, the fissions and absorptions for each geometry region used in the problem are calculated for each energy group. Leakage is not collected by geometry region, but rather it represents the leakage from the system. GROUP is the energy group number; FISSION FRACTION is the fraction of the fissions that occur in that energy group. The percent deviation for the fission fraction is the same as that of the fissions in the same group. The heading **UNIT** refers to the unit, and REGION is the

region number within the specified unit. The geometry regions are numbered sequentially within each unit, starting with 1. The sum of the fissions for every region for a given energy group is the total printed for that energy group. The same is true of absorptions. The fissions, absorptions, and leakages are given in units of per source neutron. The SYSTEM TOTAL is the sum over all the energy groups of the fissions, absorptions, and leakage. The associated percent deviation is printed for each.

The parameter **GAS** is used to control printing of fission productions and absorptions for each geometry region by energy group as shown in Fig. 8.1.143. **GAS**=YES causes this data to be printed, provided **FAR**=YES is also specified. **GAS**=NO turns off this print by group as shown in Fig. 8.1.144.

The sum of the leakage and absorptions printed for the system total should be close to 1. The fissions printed for the system total should be the same as the first k-effective printed in the final k-effective edit described in Sect. 8.1.5.28. If differential albedos are used, the leakage does not include the weight lost in the albedo reflection. A message stating the weight lost in the albedo is printed. This is the weight lost due to absorptions in the albedo reflector and leakage from the albedo reflector. No leakage is associated with faces having specular, mirror image, or periodic reflection. Thus there is no leakage associated with an infinite problem. The total elapsed time and final random number are printed at the end of this edit.

8.1.5.32 Matrix k-effective by position index

The matrix k-effective by unit location (also referred to as array position or position index) is calculated if **MKP**=YES is specified in the parameter data and a global array is present in the model. It is the largest eigenvalue of the fission production matrix collected by position index. Sect. 8.1.4.13 provides a complete discussion of matrix k-effective. The POSITION INDEX is a number referencing a position in a 3-D lattice. POSITION is the X, Y, and Z location within the lattice. **UNIT** is the unit located at the specified location in the lattice. Thus, in Fig. 8.1.148 , **UNIT** 1 is located at the lower left-hand front corner of the array or 3-D lattice representing the problem (X=1, Y=1, Z=1) and the corresponding POSITION INDEX is 1. POSITION INDEX 8 is the top right-hand back corner of the lattice, POSITION X=2, Y=2, Z=2 and the unit located at that position is **UNIT** 8. An example of the matrix k-effective by position is given in Fig. 8.1.145 . The text output is contained within two rows of asterisks to draw attention to it.

```
1sample problem 2  2c8 bare with 8 unit types matrix calculation

 ********************************************************************************************************************************************************

  global array position k-effective=  9.99716E-01  + or -  3.10531E-03
  the global array position k-effective is the largest eigenvalue of the fission production by global array position index matrix.

 ********************************************************************************************************************************************************

 elapsed time   0.05717 minutes
```

Fig. 8.1.145: Example of matrix k-effective by position index.

8.1.5.33 Fission production by position index matrix

To obtain fission production by position information, the user must specify **MKP**=YES and **FMP**=YES in the parameter data. The number of entries in the fission production matrix by position index is the square of the array size. Thus for a $2 \times 2 \times 2$ array there are 64 entries, and for a $4 \times 4 \times 4$ array there are 4,096 entries in the fission production matrix by position index. An example of the fission production matrix by position index for a $2 \times 2 \times 2$ array is shown in Fig. 8.1.146.

fission production by global array position index matrix
production is the number of next generation neutrons produced at the sink position by a neutron born at the source position.

source index	sink index	frac. dev.	sink index	frac. dev.	sink index	frac. dev.	sink index	frac. dev.	sink index	frac. dev.	sink index	frac. dev.
	0		1		2		3		4		5	
0	0.000E+00	(0.0000)	0.000E+00	(0.0000)	0.000E+00	(0.0000)	0.000E+00	(0.0000)	0.000E+00	(0.0000)	0.000E+00	(0.0000)
1	0.000E+00	(0.0000)	7.258E-01	(0.0125)	5.284E-02	(0.0266)	5.547E-02	(0.0271)	2.863E-02	(0.0386)	7.138E-02	(0.0259)
2	0.000E+00	(0.0000)	5.527E-02	(0.0274)	7.254E-01	(0.0118)	2.796E-02	(0.0399)	5.495E-02	(0.0281)	2.579E-02	(0.0348)
3	0.000E+00	(0.0000)	5.337E-02	(0.0259)	2.711E-02	(0.0338)	7.223E-01	(0.0103)	4.992E-02	(0.0283)	2.534E-02	(0.0356)
4	0.000E+00	(0.0000)	2.931E-02	(0.0399)	5.573E-02	(0.0269)	5.499E-02	(0.0304)	7.230E-01	(0.0133)	2.133E-02	(0.0442)
5	0.000E+00	(0.0000)	6.632E-02	(0.0246)	2.253E-02	(0.0415)	2.405E-02	(0.0389)	1.786E-02	(0.0425)	7.327E-01	(0.0106)
6	0.000E+00	(0.0000)	2.629E-02	(0.0413)	6.841E-02	(0.0258)	2.080E-02	(0.0421)	2.456E-02	(0.0411)	5.680E-02	(0.0272)
7	0.000E+00	(0.0000)	2.522E-02	(0.0396)	1.910E-02	(0.0422)	6.626E-02	(0.0257)	2.495E-02	(0.0414)	5.245E-02	(0.0271)
8	0.000E+00	(0.0000)	1.870E-02	(0.0421)	2.475E-02	(0.0403)	2.444E-02	(0.0395)	7.089E-02	(0.0244)	2.885E-02	(0.0362)

source index	sink index	frac. dev.	sink index	frac. dev.	sink index	frac. dev.
	6		7		8	
0	0.000E+00	(0.0000)	0.000E+00	(0.0000)	0.000E+00	(0.0000)
1	2.444E-02	(0.0396)	2.541E-02	(0.0396)	1.935E-02	(0.0468)
2	6.796E-02	(0.0265)	1.921E-02	(0.0428)	2.428E-02	(0.0404)
3	1.836E-02	(0.0463)	6.822E-02	(0.0260)	2.515E-02	(0.0384)
4	2.760E-02	(0.0375)	2.450E-02	(0.0415)	7.039E-02	(0.0254)
5	5.250E-02	(0.0272)	5.516E-02	(0.0302)	2.874E-02	(0.0377)
6	7.132E-01	(0.0121)	3.147E-02	(0.0340)	5.318E-02	(0.0284)
7	2.914E-02	(0.0361)	7.289E-01	(0.0096)	5.374E-02	(0.0269)
8	5.451E-02	(0.0275)	5.274E-02	(0.0297)	7.278E-01	(0.0095)

Fig. 8.1.146: Sample fission production matrix by position index.

For each position index in the array, the number of next-generation neutrons produced at position index J per neutron born at position index I is determined. The fission production matrix by position index is used to determine the matrix k-effective, cofactor k-effective and source vector by position index.

8.1.5.34 Source vector by position index

Source vector by position index information is printed if **MKP**=YES is specified in the parameter data. The source vector by position index is the eigenvector of the fission production matrix by position index and should sum to 1.0. It represents the fission source for the specified locations in the 3-D lattice representing the physical problem being analyzed. Position zero contains all material outside the global array. An example of the source vector by position index is shown in Fig. 8.1.147. The average self-multiplication by array position is the overall average of the self-multiplication of all units used in the problem.

1sample problem 2 2c8 bare with 8 unit types matrix calculation

source vector by global array position index

index	vector
0	0.00000E+00
1	1.25636E-01
2	1.22401E-01
3	1.23812E-01
4	1.22575E-01
5	1.31289E-01
6	1.20477E-01
7	1.28009E-01
8	1.25801E-01

Fig. 8.1.147: Example of source vector by position index.

8.1.5.35 Cofactor k-effective by position index

The cofactor k-effective by position index edit is printed if **MKP**=YES is specified in the parameter data. This means that the fission production matrix is collected by position index. Calculating and printing cofactor k-effectives by position index can be avoided by specifying **CKP**=NO in the parameter data. An example of the cofactor k-effective by position index is shown in Fig. 8.1.148. See KENO Sect. 8.1.8.3 for a description of the problem used for the example.

lsample problem 2 2c8 bare with 8 unit types matrix calculation

position index	position x	y	z	unit	cofactor k-effective	deviation
0	0	0	0	0	9.99715E-01	3.10512E-03
1	1	1	1	1	9.61078E-01	3.24293E-03
2	2	1	1	2	9.62412E-01	3.24747E-03
3	1	2	1	3	9.62859E-01	3.32117E-03
4	2	2	1	4	9.61258E-01	3.22394E-03
5	1	1	2	5	9.60567E-01	3.32004E-03
6	2	1	2	6	9.62089E-01	3.26912E-03
7	1	2	2	7	9.61184E-01	3.37780E-03
8	2	2	2	8	9.61290E-01	3.34370E-03

elapsed time   0.05783 minutes

Fig. 8.1.148: Example of cofactor k-effective by position index.

The cofactor k-effective for a given position index is the largest eigenvalue of the fission production matrix collected by position index, reduced by the row and column associated with that position index. Thus the cofactor k-effective is the value of k-effective for the system calculated without the fission source of the unit located at the specified position index.

8.1.5.36 Matrix k-effective by unit number

The matrix k-effective by unit number (unit k-effective) is the largest eigenvalue of the fission production by unit matrix. It is calculated only if **MKU**=YES is specified in the parameter data. An example of the matrix k-effective by unit is given in Fig. 8.1.149.

lsample problem 2 2c8 bare with 8 unit types matrix calculation

****

unit k-effective= 9.99716E-01 + or - 3.10531E-03
the unit k-effective is the largest eigenvalue of the fission production by unit number matrix.

****

elapsed time   0.05783 minutes

Fig. 8.1.149: Example of matrix k-effective by unit number.

8.1.5.37 Fission production by unit number matrix

The fission production by unit number matrix is computed and printed if **MKU**=YES and **FMU**=YES are specified in the parameter data. Thus, for each unit in the array, the number of next-generation neutrons produced in Unit J per neutron born in Unit I is determined. This is the fission production matrix by unit, and it is used to determine the matrix k-effective by unit, the cofactor k-effective by unit, and the source vector by unit. An example of the fission production matrix by unit is shown in Fig. 8.1.150 for text output.

fission production by unit number matrix
production is the number of next generation neutrons produced in the sink unit by a neutron born in the source unit.

source index	sink index 1	frac. dev.	sink index 2	frac. dev.	sink index 3	frac. dev.	sink index 4	frac. dev.	sink index 5	frac. dev.	sink index 6	frac. dev.
1	7.258E-01	(0.0125)	5.284E-02	(0.0266)	5.547E-02	(0.0271)	2.863E-02	(0.0386)	7.138E-02	(0.0259)	2.444E-02	(0.0396)
2	5.527E-02	(0.0274)	7.254E-01	(0.0118)	2.796E-02	(0.0399)	5.495E-02	(0.0281)	2.579E-02	(0.0348)	6.796E-02	(0.0265)
3	5.337E-02	(0.0259)	2.711E-02	(0.0338)	7.223E-01	(0.0103)	4.992E-02	(0.0283)	2.534E-02	(0.0356)	1.836E-02	(0.0463)
4	2.931E-02	(0.0399)	5.573E-02	(0.0269)	5.499E-02	(0.0304)	7.230E-01	(0.0133)	2.133E-02	(0.0442)	2.760E-02	(0.0375)
5	6.632E-02	(0.0246)	2.253E-02	(0.0415)	2.405E-02	(0.0389)	1.786E-02	(0.0425)	7.327E-01	(0.0106)	5.250E-02	(0.0272)
6	2.629E-02	(0.0413)	6.841E-02	(0.0258)	2.080E-02	(0.0421)	2.456E-02	(0.0411)	5.680E-02	(0.0272)	7.132E-01	(0.0121)
7	2.522E-02	(0.0396)	1.910E-02	(0.0422)	6.626E-02	(0.0257)	2.495E-02	(0.0414)	5.245E-02	(0.0271)	2.914E-02	(0.0361)
8	1.870E-02	(0.0421)	2.475E-02	(0.0403)	2.444E-02	(0.0395)	7.089E-02	(0.0244)	2.885E-02	(0.0362)	5.451E-02	(0.0275)

source index	sink index 7	frac. dev.	sink index 8	frac. dev.
1	2.541E-02	(0.0396)	1.935E-02	(0.0468)
2	1.921E-02	(0.0428)	2.428E-02	(0.0404)
3	6.822E-02	(0.0260)	2.515E-02	(0.0384)
4	2.450E-02	(0.0415)	7.039E-02	(0.0254)
5	5.516E-02	(0.0302)	2.874E-02	(0.0377)
6	3.147E-02	(0.0340)	5.318E-02	(0.0284)
7	7.289E-01	(0.0096)	5.374E-02	(0.0269)
8	5.274E-02	(0.0297)	7.278E-01	(0.0095)

Fig. 8.1.150: An example of the fission probability matrix by unit.

8.1.5.38 Source vector by unit number

The source vector by unit is the eigenvector of the fission production matrix by unit and is printed if **MKU**=YES is specified in the parameter data. It represents the fission source for the units used in the problem. The components of the source vector should sum to 1.0. An example of the source vector by unit is given in Fig. 8.1.151. The average self-multiplication by unit is printed following the source vector. This value of self-multiplication includes those histories born in the unit which cause fissions in the same unit regardless of whether or not it exited and then returned. Therefore, this value will not agree with the value printed for the self-multiplication of the unit as described in Sect. Sect. 8.1.5.34 if the problem uses multiple units, if the system is reflected, or if a differential albedo is used in the problem.

1sample problem 2 2c8 bare with 8 unit types matrix calculation

source vector by unit

unit	vector
1	1.25636E-01
2	1.22401E-01
3	1.23812E-01
4	1.22575E-01
5	1.31289E-01
6	1.20477E-01
7	1.28009E-01
8	1.25801E-01

Fig. 8.1.151: Example of the source vector by unit.

8.1.5.39 Cofactor k-effective by unit number

The cofactor k-effective for a given unit is the k-effective of the system calculated without the fission source of that unit. Cofactor k-effectives are printed if **MKU**=YES is specified in the parameter data. Calculating and printing cofactor k-effectives by unit can be avoided by specifying **CKU**=NO in the parameter data. This step is accomplished by determining the eigenvalue of the fission production matrix by unit after it has been reduced by the row and column associated with that unit. An example of the cofactor k-effective by unit is given in Fig. 8.1.152.

1sample problem 2 2c8 bare with 8 unit types matrix calculation

unit	cofactor k-effective	deviation
1	9.61078E-01	3.24293E-03
2	9.62412E-01	3.24747E-03
3	9.62859E-01	3.32117E-03
4	9.61258E-01	3.22394E-03
5	9.60567E-01	3.32004E-03
6	9.62089E-01	3.26912E-03
7	9.61184E-01	3.37780E-03
8	9.61290E-01	3.34370E-03

elapsed time 0.05783 minutes

Fig. 8.1.152: Example of cofactor k-effective by unit number.

8.1.5.40 Matrix k-effective by hole number

The matrix k-effective by hole number is the largest eigenvalue of the fission production matrix collected by hole number and is calculated if **MKH**=YES was specified in the parameter data. An example of the matrix k-effective by hole number is given in Fig. 8.1.153.

1sample problem 18 1f27 demonstration of options problem

hole k-effective= 1.00946E+00 + or - 3.93611E-03
the hole k-effective is the largest eigenvalue of the fission production by hole number matrix.

elapsed time 0.30217 minutes

Fig. 8.1.153: Example of matrix k-effective by hole number.

8.1.5.41 Fission production by hole number matrix

This is the fission production matrix collected by hole number. It is printed only if **MKH**=YES and **FMH**=YES were specified in the parameter data. An example of this fission production matrix is given in Fig. 8.1.154. This matrix indicates the number of next generation neutrons produced in **HOLE** number J by a neutron born in **HOLE** number I.

fission production by hole number matrix
production is the number of next generation neutrons produced in the sink hole by a neutron born in the source hole.

source index	sink index 0	frac. dev.	sink index 1	frac. dev.	sink index 2	frac. dev.	sink index 3	frac. dev.	sink index 4	frac. dev.
0	0.000E+00	(0.0000)	0.000E+00	(0.0000)	0.000E+00	(0.0000)	0.000E+00	(0.0000)	0.000E+00	(0.0000)
1	0.000E+00	(0.0000)	7.449E-01	(0.0103)	6.649E-02	(0.0201)	9.359E-02	(0.0173)	1.238E-01	(0.0192)
2	0.000E+00	(0.0000)	1.331E-01	(0.0201)	6.656E-01	(0.0117)	9.037E-02	(0.0232)	1.241E-01	(0.0205)
3	0.000E+00	(0.0000)	1.177E-01	(0.0176)	5.786E-02	(0.0223)	7.037E-01	(0.0091)	1.180E-01	(0.0148)
4	0.000E+00	(0.0000)	1.114E-01	(0.0144)	5.534E-02	(0.0189)	8.029E-02	(0.0174)	7.506E-01	(0.0067)

Fig. 8.1.154: Example of fission production matrix by hole.

8.1.5.42 Source vector by hole number

The source vector by hole is the eigenvalue of the fission production matrix by hole number and is printed if **MKH**=YES is specified in the parameter data. The source vector should sum to 1.0. An example of the source vector by hole is shown in Fig. 8.1.155. The average self-multiplication by hole is the overall average of the self-multiplication of all the holes in the problem.

1sample problem 18 1f27 demonstration of options problem

source vector by hole

hole	vector
0	0.00000E+00
1	3.08670E-01
2	1.48641E-01
3	2.22511E-01
4	3.20178E-01

Fig. 8.1.155: Example of source vector by hole number.

8.1.5.43 Cofactor k-effective by hole number

The cofactor k-effective for a given hole is the k-effective of the system calculated without the fission source of that hole and is calculated if **CKH**=YES is entered in the parameter data. These values are computed by determining the eigenvalue of the fission production matrix by hole after it has been reduced by the row and column associated with that hole. An example of the cofactor k-effective by hole number is given in Fig. 8.1.156.

1sample problem 18 1f27 demonstration of options problem

hole	unit	cofactor k-effective	deviation
0	0	1.00948E+00	3.93736E-03
1	2	8.82812E-01	3.86206E-03
2	3	9.48706E-01	4.13587E-03
3	4	9.25450E-01	4.43989E-03
4	5	8.93137E-01	5.12180E-03

elapsed time 0.85983 minutes

Fig. 8.1.156: Example of cofactor k-effective by hole number.

8.1.5.44 Matrix k-effective by array number

The matrix k-effective by array number is the largest eigenvalue of the fission production matrix collected by array number and is calculated if **MKA**=YES is entered in the parameter data. An example is given in Fig. 8.1.157 . The number of next generation neutrons produced in array number J by a neutron born in array number I is given in this fission production matrix.

```
1sample problem 18   1f27 demonstration of options problem

 ************************************************************************************************************************************************

  array k-effective=  1.00946E+00  + or -  3.93611E-03
  the array k-effective is the largest eigenvalue of the fission production by array number matrix.

 ************************************************************************************************************************************************

 elapsed time   0.86000 minutes
```

Fig. 8.1.157: Example of matrix k-effective by array number.

8.1.5.45 Fission production by array number matrix

The fission production matrix collected by array number is shown in Fig. 8.1.158 . It is printed only if **MKA**=YES and **FMA**=YES are specified in the parameter data.

```
  fission production by array number matrix
   production is the number of next generation neutrons produced in the sink array by a neutron born in the source array.

source     sink      frac.     sink      frac.     sink      frac.     sink      frac.     sink      frac.
index      index     dev.      index     dev.      index     dev.      index     dev.      index     dev.
             0                   1                   2                   3                   4
    0   0.000E+00 (0.0000)  0.000E+00 (0.0000)  0.000E+00 (0.0000)  0.000E+00 (0.0000)  0.000E+00 (0.0000)
    1   0.000E+00 (0.0000)  7.449E-01 (0.0103)  6.649E-02 (0.0201)  9.359E-02 (0.0173)  1.238E-01 (0.0192)
    2   0.000E+00 (0.0000)  1.331E-01 (0.0201)  6.656E-01 (0.0117)  9.037E-02 (0.0232)  1.241E-01 (0.0205)
    3   0.000E+00 (0.0000)  1.177E-01 (0.0176)  5.786E-02 (0.0223)  7.037E-01 (0.0091)  1.180E-01 (0.0148)
    4   0.000E+00 (0.0000)  1.114E-01 (0.0144)  5.534E-02 (0.0189)  8.029E-02 (0.0174)  7.506E-01 (0.0067)
```

Fig. 8.1.158: An example of the fission production matrix by array number.

8.1.5.46 Source vector by array number

The source vector by array number is the eigenvector of the fission production matrix by array number and is printed if **MKA**=YES is specified in the parameter data. The source vector should sum to 1.0. An example of the source vector by array number is shown in Fig. 8.1.159. The average self-multiplication by array number is the overall self-multiplication of all the arrays in the problem.

```
1sample problem 18   1f27 demonstration of options problem

   source vector by array

    array     vector
      0    0.00000E+00
      1    3.08670E-01
      2    1.48641E-01
      3    2.22511E-01
      4    3.20178E-01
```

Fig. 8.1.159: Example of source vector by array number.

8.1.5.47 Cofactor k-effective by array number

The cofactor k-effective by array number is the k-effective of the system calculated without the fission source of that array and is calculated if **CKA**=YES is entered in the parameter data. This is achieved by determining the eigenvector of the fission production matrix by array after reducing it by the row and column associated with the specified array. Fig. 8.1.160 is an example of the cofactor k-effective by array number.

```
1sample problem 18   1f27 demonstration of options problem

                array        array       cofactor
                index       number      k-effective     deviation
                  0            0       1.00948E+00    3.93736E-03
                  1            1       8.82812E-01    3.86206E-03
                  2            2       9.48706E-01    4.13587E-03
                  3            3       9.25450E-01    4.43989E-03
                  4            4       8.93137E-01    5.12180E-03

 elapsed time   0.86000 minutes
```

Fig. 8.1.160: Example of cofactor k-effective by array number.

8.1.5.48 Mesh tallies

The mesh tallies edit is optional. KENO prints the specification of each mesh tally defined by **CDS=**, **FIS=**, and **GFX=** parameters. A sample mesh tallies edit is given in Fig. 8.1.161. This edit does not print any details for both the mesh flux tally activated by **MFX=** and the $F^*(r)$ mesh tally controlled by **CGD=** parameters.

At the top of this output edit, the number of tallies computed for the given problem is printed. Then, a summary section including the tally response, spatial and energy grid identifiers, approximate memory required to compute and write this tally data, and tally 3dmap output filename is provided. The grid identifier points to the spatial mesh grid defined either with **MSH=** parameter or **grid geometry data** input block. Similarly, the energy identifier points to the DEFAULT energy boundaries (see Sect. 8.1.5.13). A summary of the energy and spatial grids is presented after this section. Details of both energy boundaries and grid definitions used for these mesh tallies can be seen in Sect. 8.1.5.13 and Sect. 8.1.5.19.

```
                                                  **** mesh tallies ****

3 mesh tallies computed for this problem


 mesh tally (requested with parameter cds)
     response         : fission_source
     grid id          : 17
     energy id        : Default
     memory allocated : 9.229 MB
     output           : example8.fission_source.3dmap

     energy boundaries:
          group   energy (eV)
         ------  ------------
              1   2.00000E+07
              2   1.73300E+07
              3   1.56800E+07
              .              .
            250   7.50000E-04
            251   5.00000E-04
            252   1.00000E-04
                  1.00000E-05
         ------  ------------

     grid summary:
           x:   20 cells from -1.37400E+01 to  1.37400E+01
           y:   20 cells from -1.37400E+01 to  1.37400E+01
           z:    6 cells from -1.30100E+01 to  1.30100E+01
         Total number of cells: 2400


 mesh tally (requested with parameter gfx)
     response         : flux
     grid id          : 113
     energy id        : Default
     memory allocated : 0.104 MB
     output           : example8.flux.3dmap

     energy boundaries:
          group   energy (eV)
         ------  ------------
              1   2.00000E+07
              2   1.73300E+07
              3   1.56800E+07
              .              .
            250   7.50000E-04
            251   5.00000E-04
            252   1.00000E-04
                  1.00000E-05
         ------  ------------

     grid summary:
           x:    3 cells from -1.74001E+00 to  1.37400E+01
           y:    3 cells from -1.74001E+00 to  1.37400E+01
           z:    3 cells from -1.01001E+00 to  1.30100E+01
         Total number of cells: 27


 mesh tally (requested with parameter fis)
     response         : fission_density
     grid id          : 118
     energy id        : Default
     memory allocated : 27.686 MB
     output           : example8.fission_density.3dmap

     energy boundaries:
          group   energy (eV)
         ------  ------------
              1   2.00000E+07
              2   1.73300E+07
              3   1.56800E+07
              .              .
            250   7.50000E-04
            251   5.00000E-04
            252   1.00000E-04
                  1.00000E-05
         ------  ------------

     grid summary:
           x:   20 cells from -1.74001E+00 to  1.37400E+01
           y:   20 cells from -1.74001E+00 to  1.37400E+01
           z:   18 cells from -1.01001E+00 to  1.30100E+01
         Total number of cells: 7200
```

Fig. 8.1.161: Example of mesh tallies output edit.

Note: Tally results stored in 3dmap files can be visualized by Fulcrum mesh tally plotting capability. Similarly, they can be processed by using the utility tools provided by MAVRIC utilities.

8.1.5.49 Reaction tally

Reaction tally calculations are performed if the required data are specified in the reaction data block (Sect. 8.1.3.15). KENO prints a table in the output that summarizes the parameter specifications in reaction tally calculations. An example of the parameter table for the reaction tally calculations is shown in Fig. 8.1.162.

```
1reaction tally calculations

 cxm, computation mode                      : 1 use all available reactions,  tally xs/rr in NGP group
 tally reaction cross sections              : yes
 tally reaction rates                       : no
 tally mixture fluxes                       : yes
 generate data for CEDepletion              : no
 ------------------------------------------------------------------------------------------------------
 save fluxes in flux.bin file               : no
 save cross section in a working library    : no
 write mixture fluxes in an ascii file      : no
 write cross sections  in an ascii file     : no
 write reaction rates in an ascii file      : no
 ------------------------------------------------------------------------------------------------------
 number of mixtures in reaction calculations : 1
 total number of nuclides in reaction calc.  : 3
 number of energy groups, xs/rr tally        : 252
 number of energy groups, mix. flux tally    : 252

 Now writing reaction cross-sections to the file(s): keno_micro_xs*

 Now writing mixture flux to the file(s): keno_mixture_flux*

 elapsed time   0.82800 minutes
```

Fig. 8.1.162: Example of reaction tally summary table.

8.1.5.50 Source convergence diagnostics edit

KENO calculates the Shannon entropy of the fission source distribution at each generation and reports this in the problem output (Sect. 8.1.5.26). At the end of calculation, KENO performs three tests to check the fission source convergence and reports these test results in the output. (See Sect. 8.1.7.7 for the definition of these tests.) An example of the fission source convergence diagnostics is given in Fig. 8.1.163. The last entry in this output edit is the plot of Shannon entropy by generations run, and a sample Shannon entropy plot is shown in Fig. 8.1.164.

```
*****************************************************************************************************************************************************

                                                  ***** Shannon entropy tests *****

*****************************************************************************************************************************************************


-------------------------------------------------------------------------------------|----------------
 Test 1: Is the final fission source converged?                                      | Test 1: passed
-------------------------------------------------------------------------------------|----------------


Test 1 details:

Is the mean square posterior relative entropy <  the center mean square Shannon entropy?

 ( msq(D) < cmsq(H(S)) )

generations skipped     msq          cmsq
                  3  1.28186E-03  1.43212E-03
                 13  1.28191E-03  1.47644E-03
                 23  1.29666E-03  1.47190E-03
                 33  1.26859E-03  1.50310E-03
                 43  1.22198E-03  1.54536E-03
                 53  1.17043E-03  1.60423E-03
                 63  1.13928E-03  1.58489E-03
                 73  1.09979E-03  1.59530E-03
                 83  1.07759E-03  1.59344E-03
                 93  1.07754E-03  1.48353E-03
                103  1.02973E-03  1.54029E-03
                113  9.68436E-04  1.62409E-03
                123  1.01185E-03  1.69994E-03
                133  9.35175E-04  1.51718E-03
                143  9.66254E-04  1.19156E-03
                153  1.00845E-03  1.29367E-03
                163  9.62220E-04  1.40100E-03
                173  7.01901E-04  9.82465E-04
                183  5.82646E-04  9.96442E-04
                193  4.67163E-04  1.53894E-03


-------------------------------------------------------------------------------------|----------------
 Test 2: Are all the active generations within epsilon of the average?              | Test 2: passed
-------------------------------------------------------------------------------------|----------------


Test 2 details:

Does the Shannon entropy of each active generation vary from the average Shannon entropy of the system?

 ( |D(S||T)-[H(T-H(S))]| < 0.1 )

  0 active generations failed the test.

 The maximum value for the test is 9.67646E-02 which occurred for active generation 167.


-------------------------------------------------------------------------------------|----------------
 Test 3: Are there adequate active generations after the source is converged?       | Test 3: passed
-------------------------------------------------------------------------------------|----------------


Test 3 details:

Does the Shannon entropy for the last half of the active generations vary from the Shannon entropy of all active generations?

 ( |H(T) - Hb| < 0.1 )

   |H(T) - Hb| =  3.42598E-02
```

Fig. 8.1.163: Summary of fission source convergence diagnostics.

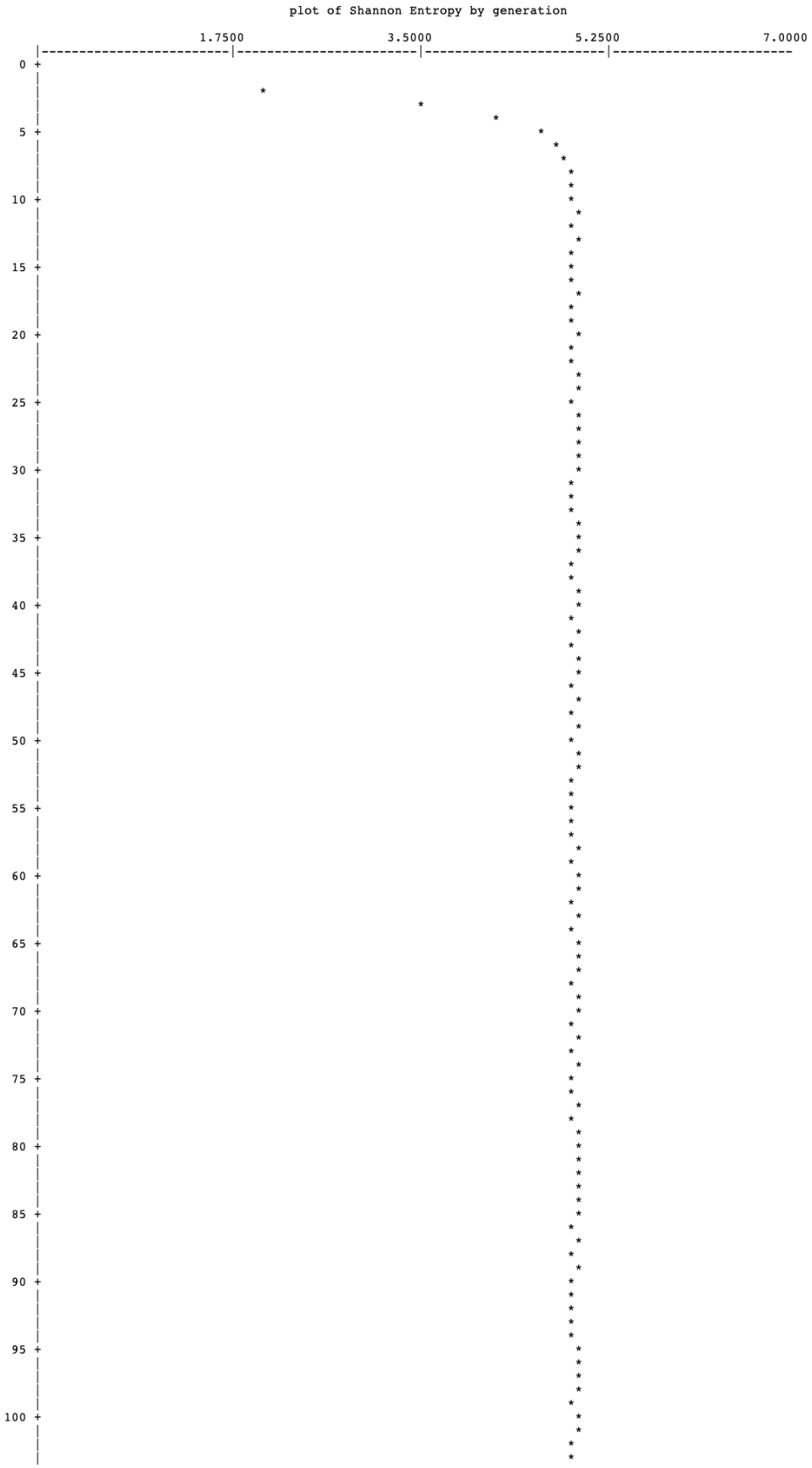


Fig. 8.1.164: A sample Shannon entropy plot.

8.1.5.51 Fission density edit

The fission density edit is optional. Subroutine KEDIT prints the neutron production density and the fission density for each geometry region if **FDN**=YES and **NUB**=YES is specified in the parameter data (these are the default values). If **NUB**=NO is specified, but **FDN**=YES, then only the production density will be given. An example of the fission density edit is shown in Fig. 8.1.165.

1 sample problem 18 1f27 demonstration of options problem

**** fission densities ****

unit	region	production density	percent deviation	total productions	fission density	percent deviation	total fissions
1	1	7.479E-06	0.197	1.010E+00	3.064E-06	0.197	4.137E-01
	2	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
	3	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
	4	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
2	1	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
3	1	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
4	1	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
5	1	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
global unit							
6	1	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
	2	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
	3	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
	4	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
	5	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
	6	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00
	7	0.000E+00	0.000	0.000E+00	0.000E+00	0.000	0.000E+00

Fig. 8.1.165: Example of the fission density edit.

The UNIT is the unit number from the geometry data, the REGION is the region number relative to the unit, the PRODUCTION DENSITY is the neutrons produced per cm^3 per source-neutron, and the FISSION DENSITY is the fissions per cm^3 per source-neutron for that geometry region, the PERCENT DEVIATION is 100 times the fractional standard deviation associated with the production density and/or the fission density, the TOTAL PRODUCTIONS is the total number of neutrons produced per source neutron, and the TOTAL FISSIONS is the total number of fissions per source neutron in the geometry region.

8.1.5.52 Flux edit

Printing the region fluxes is optional; they are only printed if **FLX**=YES is specified in the parameter data. The fluxes are printed for each unit and each geometry region in the unit for every energy group. A sample of a flux edit is given in Fig. 8.1.166.

The title of the problem is printed at the top of the page. The heading FLUXES FOR UNIT____ indicates the geometry unit for which fluxes are being printed. The region numbers relative to the unit are identified by the heading REGION____. The geometry regions within each unit are numbered sequentially, beginning with 1. Then, the volume of each region is printed. Note that if the region volume is printed as -1.0, it indicates that either volume is not defined for this region, or that volume is not calculated, or it is calculated as 0.0. GROUP is the heading for the energy groups. The headings FLUX and PERCENT DEVIATION are printed for each geometry region in the unit. The flux and its associated percent deviation are printed for every energy group and every geometry region. The last entry printed for the flux edit is the TOTL which is the total flux integrated over all energy groups for each region. The flux is in units of neutrons/cm^2 /source neutron.

1 sample problem 18 1f27 demonstration of options problem

fluxes for Unit 1

group	region 1 (vol= 1.350E+05) flux	percent deviation	region 2 (vol= 1.608E+03) flux	percent deviation	region 3 (vol= 3.019E+04) flux	percent deviation	region 4 (vol= 1.141E+06) flux	percent deviation
1	0.000E+00	0.00	0.000E+00	0.00	0.000E+00	0.00	0.000E+00	0.00
2	0.000E+00	0.00	0.000E+00	0.00	0.000E+00	0.00	0.000E+00	0.00
3	0.000E+00	0.00	0.000E+00	0.00	0.000E+00	0.00	0.000E+00	0.00
4	0.000E+00	0.00	0.000E+00	0.00	0.000E+00	0.00	0.000E+00	0.00
5	1.895E-09	33.46	6.001E-10	100.00	1.364E-09	47.55	9.454E-10	46.52
6	5.786E-08	7.14	7.473E-08	46.05	3.522E-08	8.10	2.047E-08	7.89
7	1.660E-07	3.94	1.243E-07	13.16	1.071E-07	5.62	5.830E-08	5.47
8	6.051E-07	2.03	4.439E-07	7.15	3.836E-07	2.61	2.394E-07	2.25
9	1.842E-06	1.20	1.400E-06	3.19	1.150E-06	1.61	7.074E-07	1.54
10	1.060E-06	1.44	7.742E-07	4.75	6.767E-07	1.44	4.121E-07	1.91
11	4.933E-06	0.67	3.607E-06	2.16	3.031E-06	0.87	1.839E-06	0.97
12	3.792E-06	0.74	2.654E-06	2.46	2.350E-06	0.92	1.445E-06	1.09
...								
229	4.563E-07	0.56	6.842E-07	5.29	7.887E-07	0.98	9.418E-07	1.23
230	6.209E-07	0.49	1.089E-06	3.73	1.342E-06	0.80	1.690E-06	1.16
231	3.115E-07	0.64	5.925E-07	3.76	7.990E-07	1.00	9.907E-07	1.17
232	3.561E-07	0.58	8.222E-07	4.62	1.009E-06	0.93	1.282E-06	1.29
233	4.053E-07	0.56	8.923E-07	4.27	1.228E-06	0.75	1.641E-06	0.92
234	4.578E-07	0.60	1.148E-06	3.69	1.548E-06	0.74	2.035E-06	1.02
235	5.127E-07	0.50	1.411E-06	3.68	1.860E-06	0.87	2.471E-06	0.89
236	5.372E-07	0.46	1.618E-06	3.14	2.197E-06	0.62	2.893E-06	0.93
237	5.421E-07	0.47	1.821E-06	5.21	2.426E-06	0.75	3.257E-06	0.98
238	2.382E-07	0.58	8.392E-07	3.97	1.192E-06	0.95	1.586E-06	1.00
239	5.993E-07	0.45	2.338E-06	2.94	3.447E-06	0.61	4.630E-06	0.85
240	5.834E-08	0.88	2.476E-07	8.12	3.929E-07	1.25	5.258E-07	1.94
241	4.291E-08	0.95	1.798E-07	7.55	3.122E-07	1.12	4.236E-07	2.20
242	1.198E-08	1.70	5.308E-08	12.88	9.660E-08	1.91	1.324E-07	3.80
243	9.919E-09	1.93	4.870E-08	13.11	8.108E-08	2.11	1.120E-07	3.69
244	3.844E-09	2.95	1.613E-08	18.58	3.192E-08	3.04	4.129E-08	6.30
245	3.198E-09	3.02	1.268E-08	19.90	2.752E-08	3.14	3.478E-08	7.11
246	2.391E-09	3.16	1.254E-08	21.30	2.183E-08	3.50	3.182E-08	7.94
247	1.073E-09	4.86	2.664E-09	35.53	1.029E-08	4.63	1.442E-08	10.25
248	5.827E-10	5.01	8.699E-10	72.36	5.254E-09	6.44	8.823E-09	18.64
249	5.334E-10	6.07	3.537E-09	36.36	5.541E-09	5.93	7.180E-09	13.93
250	3.974E-10	6.78	1.082E-09	51.97	4.450E-09	6.73	5.144E-09	17.51
251	2.157E-10	6.90	2.275E-09	51.58	2.491E-09	8.35	3.761E-09	24.73
252	5.896E-12	23.80	1.473E-10	100.00	4.030E-11	31.14	0.000E+00	0.00
TOTL	8.974E-05	0.15	7.490E-05	0.58	7.337E-05	0.21	6.209E-05	0.27

Fig. 8.1.166: An example of a flux edit.

8.1.5.53 Frequency distributions

A frequency distribution consists of a bar graph indicating the number of generations having k-effective in a specified interval. The intervals are determined by the code, based on the upper and lower limits of the k-effectives calculated for the generations. In this example, one asterisk is printed for each generation k-effective. Four frequency distributions are printed as shown in Fig. 8.1.167. These four plots are: (1) all active generations, (2) the last 3/4 of active generations, (3) the last half of active generations, and (4) the last quarter of active generations. Note that departures from normal distributions in these plots and radical shifts among them may be indications of source convergence issues in the calculation.

```
1sample problem 18   1f27 demonstration of options problem

                              frequency for generations       4 to     103 each asterisk represents   1.0000 generations
 0.9549 to 0.9622     *
 0.9622 to 0.9695     *
 0.9695 to 0.9768     *
 0.9768 to 0.9841     ******
 0.9841 to 0.9914     **************
 0.9914 to 0.9987     ***********
 0.9987 to 1.0060     ********
 1.0060 to 1.0133     ***********
 1.0133 to 1.0206     ******************
 1.0206 to 1.0279     ***********
 1.0279 to 1.0352     ********
 1.0352 to 1.0425     *******
 1.0425 to 1.0498     ***
 1.0498 to 1.0571
 1.0571 to 1.0644     *

                              frequency for generations      29 to     103 each asterisk represents   1.0000 generations
 0.9549 to 0.9622     *
 0.9622 to 0.9695     *
 0.9695 to 0.9768
 0.9768 to 0.9841     ***
 0.9841 to 0.9914     ***********
 0.9914 to 0.9987     *********
 0.9987 to 1.0060     *******
 1.0060 to 1.0133     ******
 1.0133 to 1.0206     ************
 1.0206 to 1.0279     **********
 1.0279 to 1.0352     ********
 1.0352 to 1.0425     *****
 1.0425 to 1.0498     *
 1.0498 to 1.0571
 1.0571 to 1.0644     *

                              frequency for generations      54 to     103 each asterisk represents   1.0000 generations
 0.9549 to 0.9622     *
 0.9622 to 0.9695     *
 0.9695 to 0.9768
 0.9768 to 0.9841     *
 0.9841 to 0.9914     ********
 0.9914 to 0.9987     ******
 0.9987 to 1.0060     *****
 1.0060 to 1.0133     **
 1.0133 to 1.0206     *******
 1.0206 to 1.0279     ********
 1.0279 to 1.0352     *******
 1.0352 to 1.0425     **
 1.0425 to 1.0498     *
 1.0498 to 1.0571
 1.0571 to 1.0644     *
1sample problem 18   1f27 demonstration of options problem

                              frequency for generations      79 to     103 each asterisk represents   1.0000 generations
 0.9549 to 0.9622
 0.9622 to 0.9695
 0.9695 to 0.9768
 0.9768 to 0.9841     *
 0.9841 to 0.9914     **
 0.9914 to 0.9987     ***
 0.9987 to 1.0060     **
 1.0060 to 1.0133
 1.0133 to 1.0206     ****
 1.0206 to 1.0279     ******
 1.0279 to 1.0352     ******
 1.0352 to 1.0425
 1.0425 to 1.0498
 1.0498 to 1.0571
 1.0571 to 1.0644     *
```

Fig. 8.1.167: An example of a frequency distribution.

8.1.5.54 Summary of parallel performance

KENO summarizes parallel performance in the output if parallel execution is requested by the user. The performance table, shown in Fig. 8.1.168, can be used to evaluate the code parallel performance with the given problem parameters. The quantities like speedup, parallel efficiency, and wall-clock time for each section (serial section, parallel section, and communication time among the processors) can be used to estimate resource requirement for similar problems to accomplish a faster execution.

```
1       ********************************************************************************************************************************
        ********************************************************************************************************************************
        ***                                                                                                                          ***
        ***                                P E R F O R M A N C E    A N A L Y S I S                                                  ***
        ***                                                                                                                          ***
        ********************************************************************************************************************************
        ********************************************************************************************************************************
        ***                                                                                                                          ***
        ***            -- Parallel execution --                                                                                      ***
        ***                                                                                                                          ***
        ***        number of MPI tasks initialized for particle tracking       3                                                     ***
        ***                                                                                                                          ***
        ***        speedup                                                      2.62                                                 ***
        ***                                                                                                                          ***
        ***        parallel efficiency(%)                                      87.25                                                 ***
        ***                                                                                                                          ***
        ***                                                                                                                          ***
        ***            -- Wall-Clock Times (minutes) for  --                                                                         ***
        ***                                                                                                                          ***
        ***        serial code section                                          0.1027                                               ***
        ***                                                                                                                          ***
        ***        parallel code section                                        0.7240                                               ***
        ***                                                                                                                          ***
        ***        communication interface                                      0.0471                                               ***
        ***                                                                                                                          ***
        ***        total simulation time (parallel with    3 MPI Tasks)         0.8690                                               ***
        ***                                                                                                                          ***
        ***        total simulation time (serial  -- estimated)                 2.2748                                               ***
        ***                                                                                                                          ***
        ********************************************************************************************************************************
        ********************************************************************************************************************************
```

Fig. 8.1.168: Example of performance analysis table.

8.1.5.55 Final results table

The final results table contains a summary of the most important physics parameters of the system and the number of error and warning messages generated during execution. The table contains the best-estimate system k-effective with one standard deviation, the energy of the average lethargy of fission, the average system nu-bar, the average mean free path of a neutron throughout the system, the number of warning and error messages generated during code execution, and a final statement on the convergence of the χ^2 test results. A summary of the Shannon entropy source convergence diagnostics is printed here. See the details of tests in Sect. 8.1.5.50 and Sect. 8.1.7.7. An example of this table is shown in Fig. 8.1.169. Also shown is the terminal edit of KENO, detailing the time required to traverse the “perilous path” through KENO.

```
1       ************************************************************************************************************************
        ***                                                                                                                  ***
        ***   sample problem 18   1f27 demonstration of options problem                                                      ***
        ***                                                                                                                  ***
        ************************************************************************************************************************
        ***                                                                                                                  ***
        ***                                  ******      final results table      ******                                  ***
        ***                                                                                                                  ***
        ***        best estimate system k-eff                                    1.0096 + or - 0.0023                        ***
        ***                                                                                                                  ***
        ***        Energy of average lethargy of Fission (eV)                    2.08321E-01 + or -  8.05618E-04             ***
        ***                                                                                                                  ***
        ***        system nu bar                                                 2.44053E+00 + or -  1.63032E-05             ***
        ***                                                                                                                  ***
        ***        system mean free path (cm)                                    3.29281E-01 + or -  2.46316E-03             ***
        ***                                                                                                                  ***
        ***        number of warning messages                                   10                                           ***
        ***                                                                                                                  ***
        ***        number of error messages                                      0                                           ***
        ***                                                                                                                  ***
        ***        k-effective satisfies the chi**2 test for normality at the 95 % level                                    ***
        ***                                                                                                                  ***
        ***                                                                                                                  ***
        ************************************************************************************************************************
        ************************************************************************************************************************
        ***                                                                                                                  ***
        ***        there are      125   source meshes                                                                       ***
        ***                                                                                                                  ***
        ***        source convergence tests failed                                                                           ***
        ***                                                                                                                  ***
        ***                                                                                                                  ***
        ************************************************************************************************************************
        ************************************************************************************************************************


 ************************************************************************************************************************************

          Congratulations!  You have successfully traversed the perilous path through KENO-V.a in     0.89800 minutes

 ************************************************************************************************************************************
```

Fig. 8.1.169: An example of the final results table.

8.1.5.56 HTML output

The HTML-formatted output is no longer actively developed, and the output files created with it may therefore not include recently added capabilities or changes. For example, the fission rate mesh tally and some CE TSUNAMI edits are not included in the HTML output.

HTML output: Program verification information

The program verification information is the first page shown in the KENO HTML output, after selecting KENO from the SCALE HTML index page. This page can also be displayed by selecting the *Program Verification Information* link under the *General Information* submenu and is shown in Fig. 8.1.170.

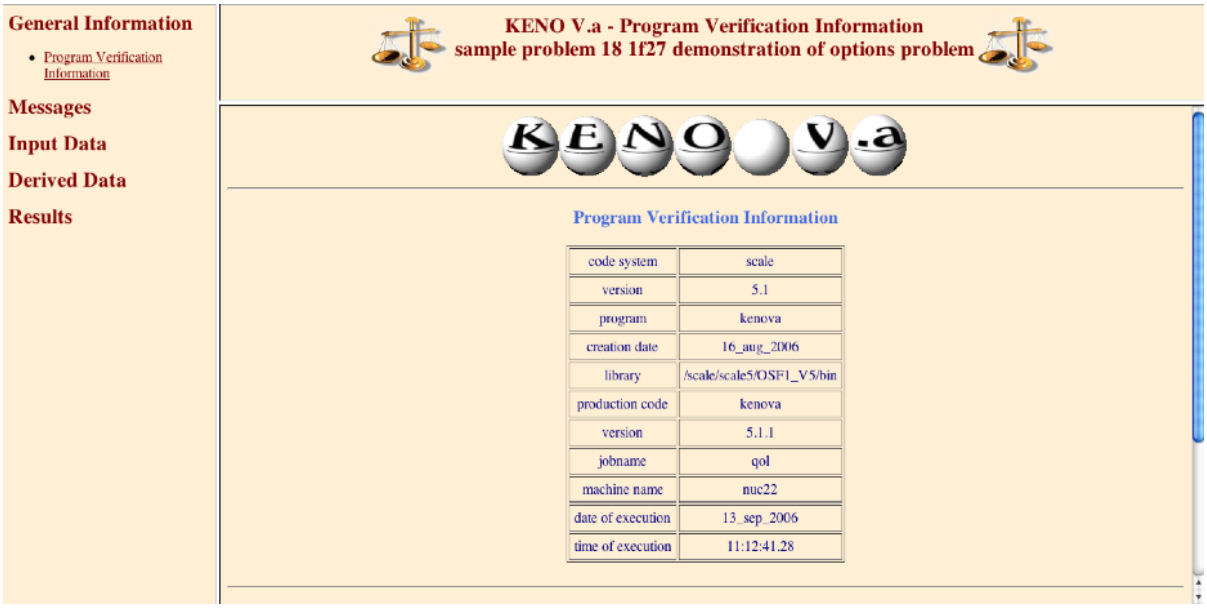


Fig. 8.1.170: Sample program verification information in HTML output.

HTML output: Messages

In the HTML output, all error and warning messages are consolidated into the *Messages* section of the output. *Error Messages* and *Warning Messages* links are available to the user as needed. An example edit with warning messages is shown in Fig. 8.1.171.

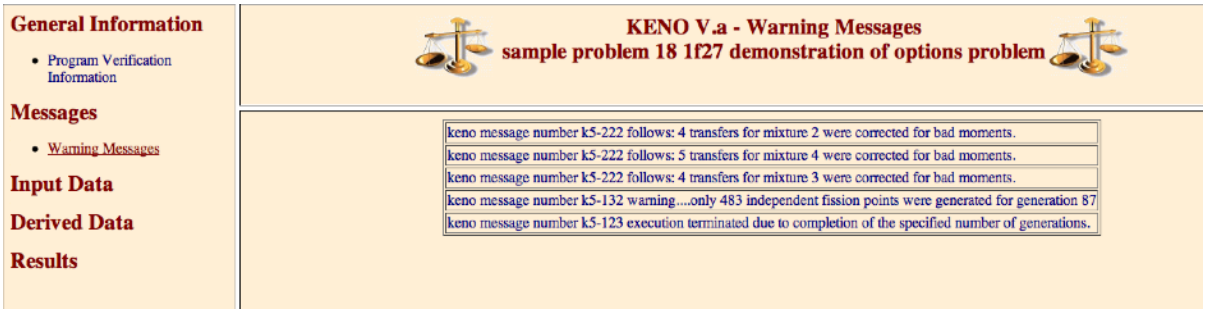


Fig. 8.1.171: Example messages edit in HTML output.

HTML output: Tables of parameter data

The first two tables generated by KENO HTML output list the numeric parameters and logical parameters that are used in the problem. The user should always verify that the parameter data block was input as desired. An example of numeric parameters table is shown in Fig. 8.1.172. An example of the logical parameters table is shown in Fig. 8.1.173. The HTML output is accessed with the *Numeric Parameters* and *Logical Parameters* links in the *Input Data* section.

Keno-V.a - Numeric Parameters
sample problem 18 1f27 demonstration of options problem

Parameter	Description	Value
tme	maximum problem time (min)	0.00
tba	time per generation (min)	10.00
gen	number of generations	103
npg	number per generation	500
nsk	number of generations to be skipped	3
beg	beginning generation number	1
res	generations between checkpoints	0
x1d	number of extra 1-d cross sections	1
nbk	neutron bank size	525
xnb	extra positions in neutron bank	0
nfb	fission bank size	500
xfb	extra positions in fission bank	0
sig	cut off standard deviation	0.0000
wta	default value of weight average	0.5000
wth	weight high for splitting	3.0000
wtl	weight low for russian roulette	0.3333
rnd	starting random number	0000F12C09ED2195
nb8	number of d.a. blocks on unit 8	1000
nl8	length of d.a. blocks on unit 8	512
nqd	quadrature order for angular fluxes	0
pnm	highest order of flux moments	0
msh	mesh size for mesh flux tally	0.0000
adj	mode of calculation	forward
tps	sampling sites per track length	5
cgs	number of secondary groups to sampl	0
cas	number of secondary angles to sampl	0

Fig. 8.1.172: Sample numeric parameters in HTML output.

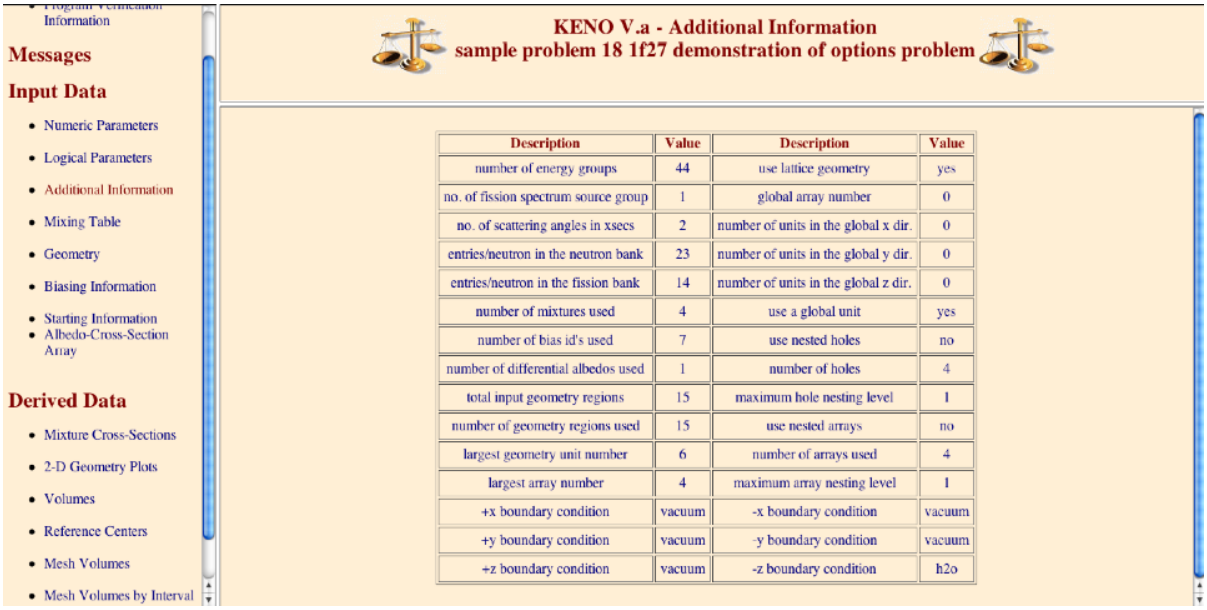
Keno-V.a - Logical Parameters
sample problem 18 1f27 demonstration of options problem

Parameter	Description	Value
run	execute problem after checking data	yes
plt	plot picture map(s)	yes
	compute fluxes (cfx, flx, cfe or mfp)	yes
fdn	compute fission densities	yes
smu	compute avg unit self-multiplication	no
nub	compute nu-bar & avg fission group	yes
mku	compute matrix k-eff by unit number	yes
cku	compute cofactor k-eff by unit number	no
fmu	print fiss prod matrix by unit number	yes
mkp	compute matrix k-eff by unit location	no
ckp	compute cofactor k-eff by unit location	no
fmp	print fiss prod matrix by unit location	no
mkh	compute matrix k-eff by hole number	yes
ckh	compute cofactor k-eff by hole number	no
fmh	print fiss prod matrix by hole number	yes
hhl	collect matrix by highest hole level	no
mka	compute matrix k-eff by array number	yes
cka	compute cofactor k-eff by array number	no
fma	print fiss prod matrix by array number	yes
hal	collect matrix by highest array level	no
amx	print all mixed cross sections	yes
xs1	print 1-d mixture x-sections	yes
xs2	print 2-d mixture x-sections	yes
xsl	print 2-d mixture Pl arrays	yes
xap	print mixture angles & probabilities	yes
pki	print fission spectrum	yes

Fig. 8.1.173: Sample logical parameters in HTML output.

HTML output: Table of additional information

The fourth table of data printed by KENO contains additional information determined from the input data. An example of this table is shown in Fig. 8.1.174. The HTML output is accessed with the *Additional Information* link in the *Input Data* section.

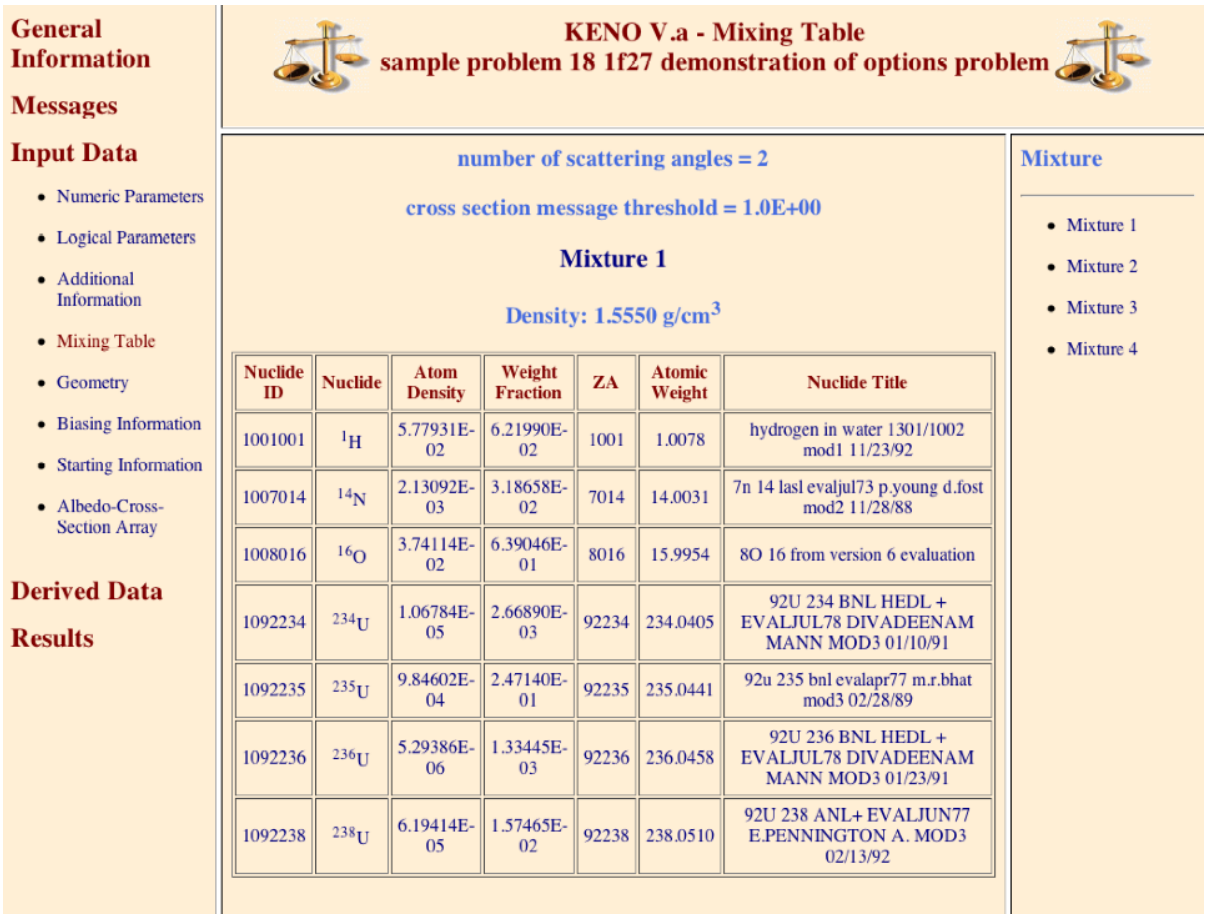


Description	Value	Description	Value
number of energy groups	44	use lattice geometry	yes
no. of fission spectrum source group	1	global array number	0
no. of scattering angles in xsecs	2	number of units in the global x dir.	0
entries/neutron in the neutron bank	23	number of units in the global y dir.	0
entries/neutron in the fission bank	14	number of units in the global z dir.	0
number of mixtures used	4	use a global unit	yes
number of bias id's used	7	use nested holes	no
number of differential albedos used	1	number of holes	4
total input geometry regions	15	maximum hole nesting level	1
number of geometry regions used	15	use nested arrays	no
largest geometry unit number	6	number of arrays used	4
largest array number	4	maximum array nesting level	1
+x boundary condition	vacuum	-x boundary condition	vacuum
+y boundary condition	vacuum	-y boundary condition	vacuum
+z boundary condition	vacuum	-z boundary condition	h2o

Fig. 8.1.174: Sample table of additional information in HTML output.

HTML output: Mixing table data

If **LIB=** is entered in the KENO parameter data and a mixing table data block is provided to KENO, then mixing table data will be printed. Sample mixing table data are shown in Fig. 8.1.175. In the HTML output, the mixing table data can be accessed with the *Mixing Table* link in the *Input Data* section.

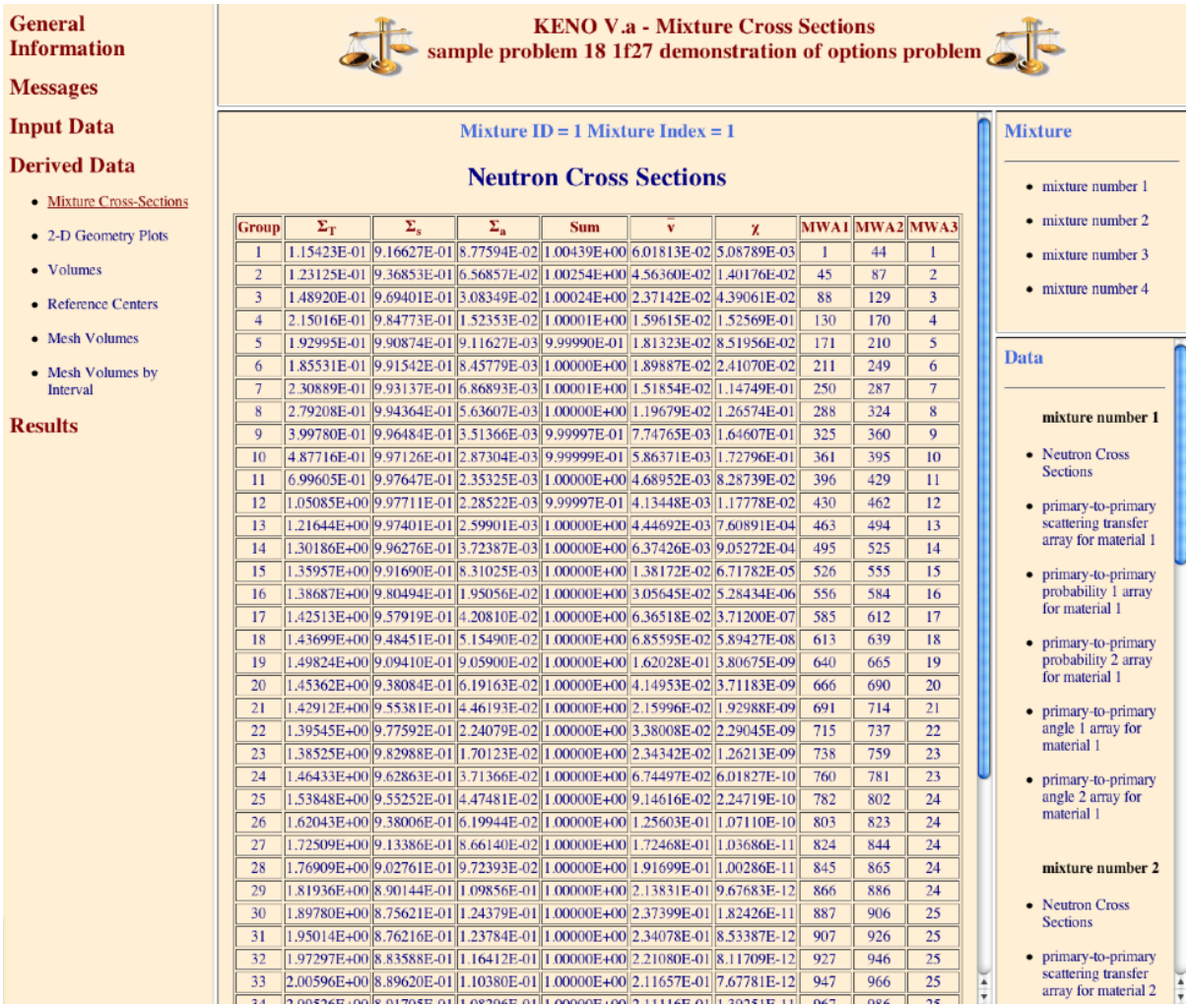


Nuclide ID	Nuclide	Atom Density	Weight Fraction	ZA	Atomic Weight	Nuclide Title
1001001	^{1}H	5.77931E-02	6.21990E-02	1001	1.0078	hydrogen in water 1301/1002 mod1 11/23/92
1007014	^{14}N	2.13092E-03	3.18658E-02	7014	14.0031	7n 14 lasl evaljul73 p.young d.fost mod2 11/28/88
1008016	^{16}O	3.74114E-02	6.39046E-01	8016	15.9954	8O 16 from version 6 evaluation
1092234	^{234}U	1.06784E-05	2.66890E-03	92234	234.0405	92U 234 BNL HEDL + EVALJUL78 DIVADEENAM MANN MOD3 01/10/91
1092235	^{235}U	9.84602E-04	2.47140E-01	92235	235.0441	92u 235 bnl evalapr77 m.r.bhat mod3 02/28/89
1092236	^{236}U	5.29386E-06	1.33445E-03	92236	236.0458	92U 236 BNL HEDL + EVALJUL78 DIVADEENAM MANN MOD3 01/23/91
1092238	^{238}U	6.19414E-05	1.57465E-02	92238	238.0510	92U 238 ANL+ EVALJUN77 E.PENNINGTON A. MOD3 02/13/92

Fig. 8.1.175: Example of mixing table data in HTML output.

HTML output: 1D macroscopic cross sections

The decision to print the 1-D mixture cross sections is optional. They are printed only if **XS1**=YES is specified in the parameter data. The 1-D mixture cross sections for a mixture are shown in Fig. 8.1.176. The HTML output is accessed with the *Mixture Cross Sections* link in the *Derived Data* section. The top menu on the right side of the window is used to set the mixture displayed in the bottom menu on the right side of the window. The bottom menu contains hyperlinks to all available edits for mixture cross sections. The 1-D cross sections are accessed with the *Neutron Cross Sections* link in the bottom menu for the desired mixture.



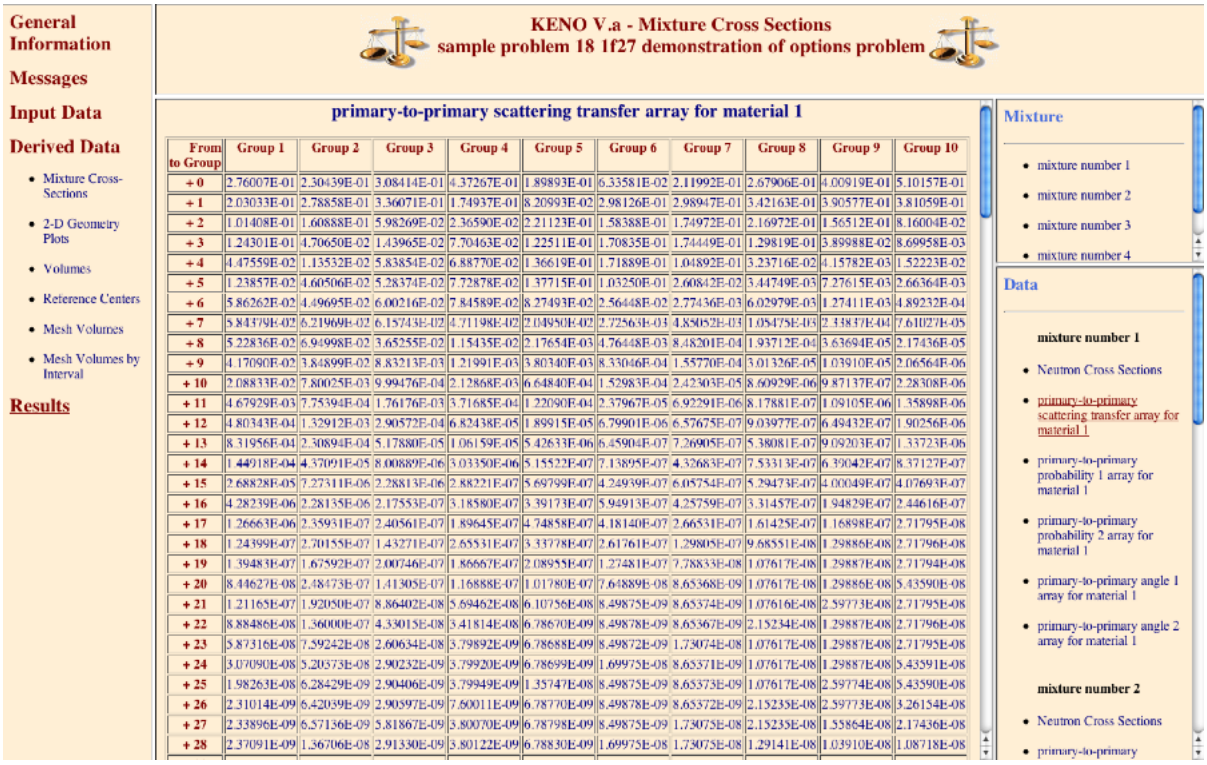
Group	Σ_T	Σ_s	Σ_a	Sum	$\bar{\nu}$	χ	MWA1	MWA2	MWA3
1	1.15423E-01	9.16627E-01	8.77594E-02	1.00439E+00	6.01813E-02	5.08789E-03	1	44	1
2	1.23125E-01	9.36853E-01	6.56857E-02	1.00254E+00	4.56360E-02	1.40176E-02	45	87	2
3	1.48920E-01	9.69401E-01	3.08349E-02	1.00024E+00	2.37142E-02	4.39061E-02	88	129	3
4	2.15016E-01	9.84773E-01	1.52353E-02	1.00001E+00	1.59615E-02	1.52569E-01	130	170	4
5	1.92995E-01	9.90874E-01	9.11627E-03	9.99990E-01	1.81323E-02	8.51956E-02	171	210	5
6	1.85531E-01	9.91542E-01	8.45779E-03	1.00000E+00	1.89887E-02	2.41070E-02	211	249	6
7	2.30889E-01	9.93137E-01	6.86893E-03	1.00001E+00	1.51854E-02	1.14749E-01	250	287	7
8	2.79208E-01	9.94364E-01	5.63607E-03	1.00000E+00	1.19679E-02	1.26574E-01	288	324	8
9	3.99780E-01	9.96484E-01	3.51366E-03	9.99997E-01	7.74765E-03	1.64607E-01	325	360	9
10	4.87716E-01	9.97126E-01	2.87304E-03	9.99999E-01	5.86371E-03	1.72796E-01	361	395	10
11	6.99605E-01	9.97647E-01	2.35325E-03	1.00000E+00	4.68952E-03	8.28739E-02	396	429	11
12	1.05085E+00	9.97711E-01	2.28522E-03	9.99997E-01	4.13448E-03	1.17778E-02	430	462	12
13	1.21644E+00	9.97401E-01	2.59901E-03	1.00000E+00	4.44692E-03	7.60891E-04	463	494	13
14	1.30186E+00	9.96276E-01	3.72387E-03	1.00000E+00	6.37426E-03	9.05272E-04	495	525	14
15	1.35957E+00	9.91690E-01	8.31025E-03	1.00000E+00	1.38172E-02	6.71782E-05	526	555	15
16	1.38687E+00	9.80494E-01	1.95056E-02	1.00000E+00	3.05645E-02	5.28434E-06	556	584	16
17	1.42513E+00	9.57919E-01	4.20810E-02	1.00000E+00	6.36518E-02	3.71200E-07	585	612	17
18	1.43699E+00	9.48451E-01	5.15490E-02	1.00000E+00	6.85595E-02	5.89427E-08	613	639	18
19	1.49824E+00	9.09410E-01	9.05900E-02	1.00000E+00	1.62028E-01	3.80675E-09	640	665	19
20	1.45362E+00	9.38084E-01	6.19163E-02	1.00000E+00	4.14953E-02	3.71183E-09	666	690	20
21	1.42912E+00	9.55381E-01	4.46193E-02	1.00000E+00	2.15996E-02	1.92988E-09	691	714	21
22	1.39545E+00	9.77592E-01	2.24079E-02	1.00000E+00	3.38008E-02	2.29045E-09	715	737	22
23	1.38525E+00	9.82988E-01	1.70123E-02	1.00000E+00	2.34342E-02	1.26213E-09	738	759	23
24	1.46433E+00	9.62863E-01	3.71366E-02	1.00000E+00	6.74497E-02	6.01827E-10	760	781	23
25	1.53848E+00	9.55252E-01	4.47481E-02	1.00000E+00	9.14616E-02	2.24719E-10	782	802	24
26	1.62043E+00	9.38006E-01	6.19944E-02	1.00000E+00	1.25603E-01	1.07110E-10	803	823	24
27	1.72509E+00	9.13386E-01	8.66140E-02	1.00000E+00	1.72468E-01	1.03686E-11	824	844	24
28	1.76909E+00	9.02761E-01	9.72393E-02	1.00000E+00	1.91699E-01	1.00286E-11	845	865	24
29	1.81936E+00	8.90144E-01	1.09856E-01	1.00000E+00	2.13831E-01	9.67683E-12	866	886	24
30	1.89780E+00	8.75621E-01	1.24379E-01	1.00000E+00	2.37399E-01	1.82426E-11	887	906	25
31	1.95014E+00	8.76216E-01	1.23784E-01	1.00000E+00	2.34078E-01	8.53387E-12	907	926	25
32	1.97297E+00	8.83588E-01	1.16412E-01	1.00000E+00	2.21080E-01	8.11709E-12	927	946	25
33	2.00596E+00	8.89620E-01	1.10380E-01	1.00000E+00	2.11657E-01	7.67781E-12	947	966	25

Fig. 8.1.176: Example 1-D macroscopic cross section in HTML output.

When the 1-D mixture cross sections are printed, the problem title is printed at the top of the page. The mixture ID, mixture index and mixture number are then printed. ID is the mixture number from the mixing table and mixture index is the index used to reference it and mixture number is its identifier. This step is followed by a heading to identify the different 1-D cross sections. *GROUP* is the energy group, *sigt* is the total cross section for the mixture, *sigs* is the nonabsorption probability, *siga* is the absorption probability, *signu* is the production probability, *chi* is the fission spectrum, *mwa1* is the pointer for the first position of the cross sections for the energy group, *mwa2* is the pointer for the last position of the cross sections for the energy group, and *mwa3* contains the group for the transfer corresponding to the first position. *SUM* is the sum of the absorption probability and the nonabsorption probability. The nonabsorption probability and the production probability are not true probabilities in that they may be greater than 1. This is because the nonabsorption probability has the (n,2n) transfer array summed into the total transfer array twice, and the (n,3n) is summed three times, etc. The absorption probability is defined as the absorption cross section divided by the total cross section. The nonabsorption probability is the sum of the group-to-group transfers for this group divided by the total cross section. The production probability is defined as the fission production cross section divided by the total cross section $\left(\nu\Sigma_f/\Sigma_T\right)$.

HTML output: 2-D macroscopic cross sections

The decision to print the 2-D mixture cross sections is optional. An example of the 2-D mixture cross sections is given in Fig. 8.1.177. The HTML output is accessed with the *Mixture Cross Sections* link in the *Derived Data* section. The top menu on the right side of the window is used to set the mixture displayed in the bottom menu on the right side of the window. The bottom menu contains hyperlinks to all available edits for mixture cross sections. The 1-D macroscopic cross sections are accessed with the *primary-to-primary scattering transfer array* link in the bottom menu for the desired mixture.

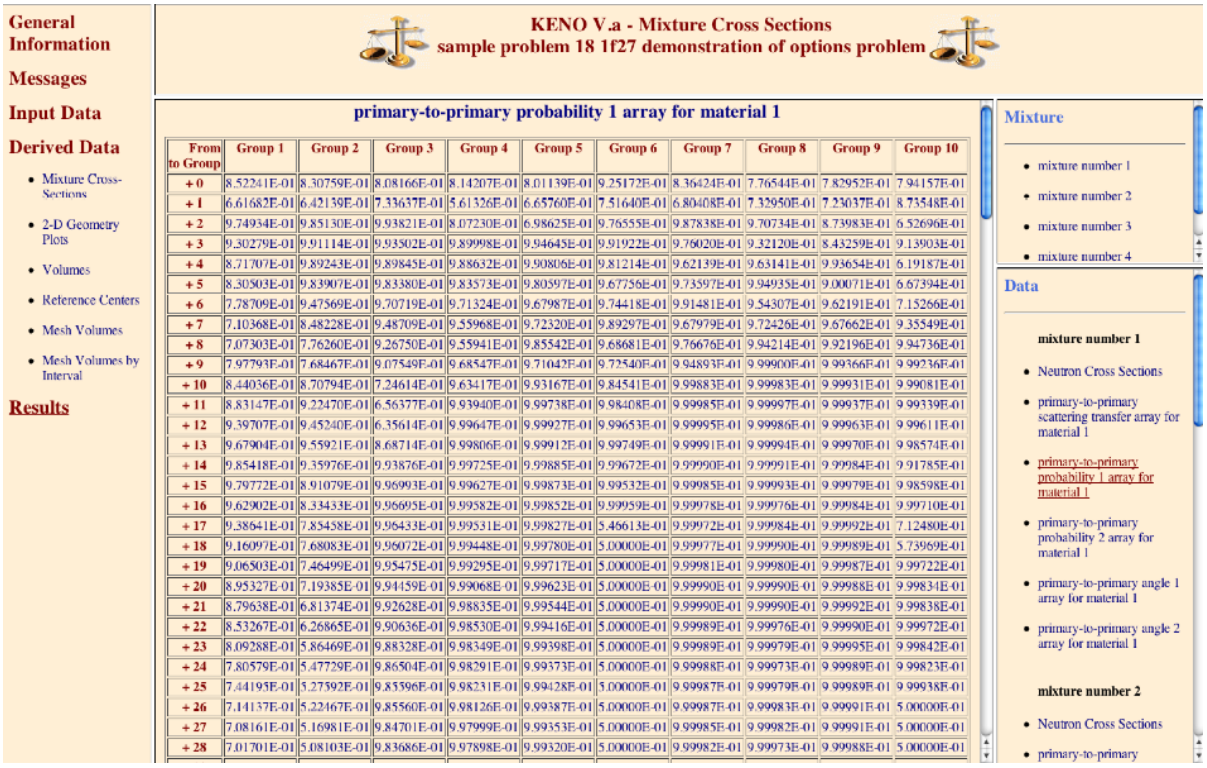


From to Group	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6	Group 7	Group 8	Group 9	Group 10
+ 0	2.76007E-01	2.30439E-01	3.08414E-01	4.37267E-01	1.89893E-01	6.33581E-02	2.11992E-01	2.67906E-01	4.00919E-01	5.10157E-01
+ 1	2.03033E-01	2.78858E-01	3.36071E-01	1.74937E-01	8.20993E-02	2.98126E-01	2.98947E-01	3.42163E-01	3.90577E-01	3.81059E-01
+ 2	1.01408E-01	1.60888E-01	5.98269E-02	2.36590E-02	2.21123E-01	1.58388E-01	1.74972E-01	2.16972E-01	1.56512E-01	8.16004E-02
+ 3	1.24301E-01	4.70650E-02	1.43965E-02	7.70463E-02	1.22511E-01	1.70835E-01	1.74449E-01	1.29819E-01	3.89988E-02	8.69958E-03
+ 4	4.47559E-02	1.13532E-02	5.83854E-02	6.88770E-02	1.36619E-01	1.71889E-01	1.04892E-01	3.23716E-02	4.15782E-03	1.52223E-02
+ 5	1.23857E-02	4.60506E-02	5.28374E-02	7.72878E-02	1.37715E-01	1.03250E-01	2.60842E-02	3.44749E-03	7.27615E-03	2.66364E-03
+ 6	5.86262E-02	4.49695E-02	6.00216E-02	7.84589E-02	8.27493E-02	2.56448E-02	2.77436E-03	6.02979E-03	1.27411E-03	4.89232E-04
+ 7	5.84379E-02	6.21969E-02	6.15743E-02	4.71198E-02	2.04950E-02	2.72563E-03	4.85052E-03	1.05475E-03	2.43837E-04	7.61027E-05
+ 8	5.22836E-02	6.94998E-02	3.65255E-02	1.15435E-02	2.17654E-03	4.76448E-03	8.48201E-04	1.93712E-04	3.63694E-05	2.17436E-05
+ 9	4.17090E-02	3.84899E-02	8.83213E-03	1.21991E-03	3.80340E-03	8.33046E-04	1.55770E-04	3.01326E-05	1.03910E-05	2.06564E-06
+ 10	2.08833E-02	7.80025E-03	9.99476E-04	2.12868E-03	6.64840E-04	1.52983E-04	2.42303E-05	8.60929E-06	9.87137E-07	2.28308E-06
+ 11	4.67929E-03	7.75394E-04	1.76176E-03	3.71685E-04	1.22090E-04	2.37967E-05	6.92291E-06	8.17881E-07	1.09105E-06	1.35898E-06
+ 12	4.80343E-04	1.32912E-03	2.90572E-04	6.82438E-05	1.89915E-05	6.79901E-06	6.57675E-07	9.03977E-07	6.49432E-07	1.90256E-06
+ 13	8.31956E-04	2.30894E-04	5.17880E-05	1.06159E-05	5.42633E-06	6.45904E-07	7.26905E-07	5.38081E-07	9.09203E-07	1.33723E-06
+ 14	1.44918E-04	4.37091E-05	8.00889E-06	3.03350E-06	5.15522E-07	7.13895E-07	4.32683E-07	7.53313E-07	6.39042E-07	8.37127E-07
+ 15	2.68828E-05	7.27311E-06	2.28813E-06	2.88221E-07	5.69799E-07	4.24939E-07	6.05754E-07	5.29473E-07	4.00049E-07	4.07693E-07
+ 16	4.28239E-06	2.28135E-06	2.17553E-07	3.18580E-07	3.39173E-07	5.94913E-07	4.25759E-07	3.31457E-07	1.94829E-07	2.44616E-07
+ 17	1.26663E-06	2.35931E-07	2.40561E-07	1.89645E-07	4.74858E-07	4.18140E-07	2.66531E-07	1.61425E-07	1.16898E-07	2.71795E-08
+ 18	1.24399E-07	2.70155E-07	1.43271E-07	2.65531E-07	3.33778E-07	2.61761E-07	1.29805E-07	9.68551E-08	1.29886E-08	2.71796E-08
+ 19	1.39483E-07	1.67592E-07	2.00746E-07	1.86667E-07	2.08955E-07	1.27481E-07	7.78833E-08	1.07617E-08	1.29887E-08	2.71794E-08
+ 20	8.44627E-08	2.48473E-07	1.41305E-07	1.16888E-07	1.01780E-07	7.64889E-08	8.65368E-09	1.07617E-08	1.29886E-08	5.43590E-08
+ 21	1.21165E-07	1.92050E-07	8.86402E-08	5.69462E-08	6.10756E-08	8.49875E-09	8.65374E-09	1.07616E-08	2.59773E-08	2.71795E-08
+ 22	8.88486E-08	1.36000E-07	4.33015E-08	3.41814E-08	6.78670E-09	8.49878E-09	8.65367E-09	2.15234E-08	1.29887E-08	2.71796E-08
+ 23	5.87316E-08	7.59242E-08	2.60634E-08	3.79892E-09	6.78688E-09	8.49872E-09	1.73074E-08	1.07617E-08	1.29887E-08	2.71795E-08
+ 24	3.07090E-08	5.20373E-08	2.90232E-09	3.79920E-09	6.78699E-09	1.69975E-08	8.65371E-09	1.07617E-08	1.29887E-08	5.43591E-08
+ 25	1.98263E-08	6.28429E-09	2.90406E-09	3.79949E-09	1.35747E-08	8.49875E-09	8.65373E-09	1.07617E-08	2.59774E-08	5.43590E-08
+ 26	2.31014E-09	6.42039E-09	2.90397E-09	7.60011E-09	6.78770E-09	8.49878E-09	8.65372E-09	2.15235E-08	2.59773E-08	3.26154E-08
+ 27	2.33896E-09	6.57136E-09	5.81867E-09	3.80070E-09	6.78798E-09	8.49875E-09	1.73075E-08	2.15235E-08	1.55864E-08	2.17436E-08
+ 28	2.37091E-09	1.36706E-08	2.91330E-09	3.80122E-09	6.78830E-09	1.69975E-08	1.73075E-08	1.29141E-08	1.03910E-08	1.08718E-08

Fig. 8.1.177: Example of 2-D macroscopic cross sections in HTML output.

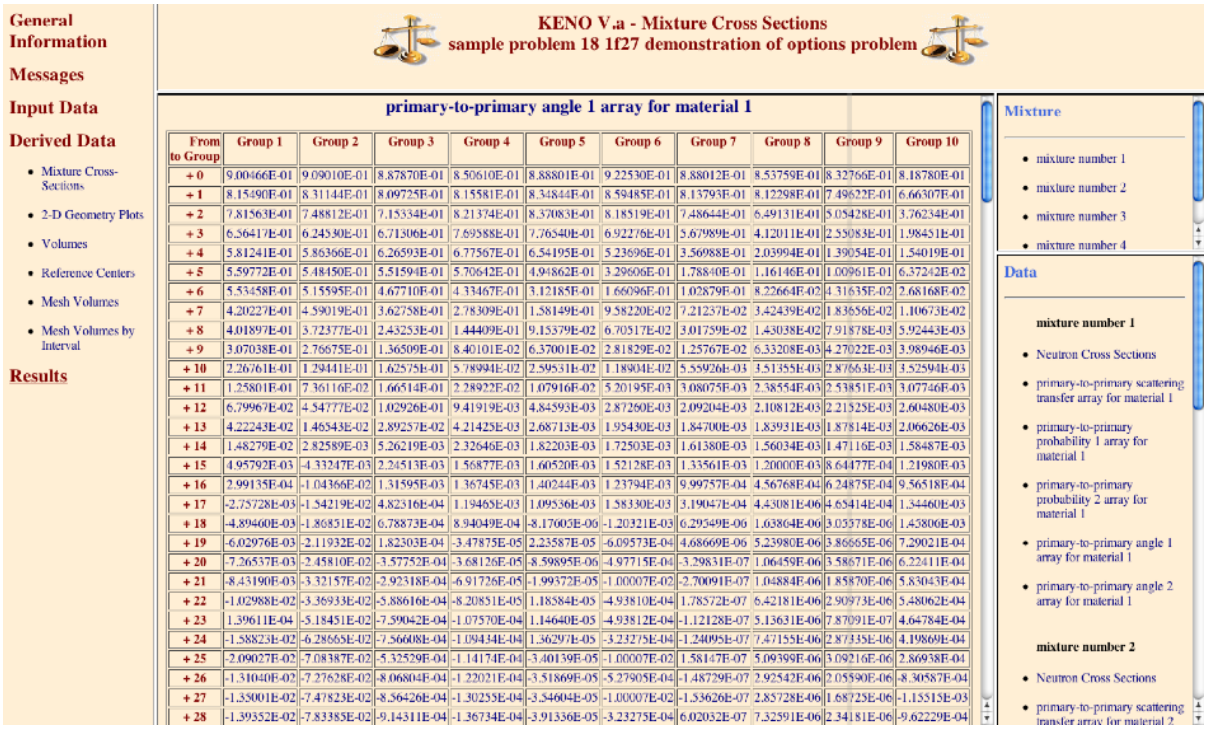
HTML output: Probabilities and angles

Printing the probabilities and angles is optional. The HTML output, Fig. 8.1.178, is accessed with the *Mixture Cross Sections* link in the *Derived Data* section. The top menu on the right side of the window is used to set the mixture displayed in the bottom menu on the right side of the window. The bottom menu contains hyperlinks to all available edits for mixture cross sections. The probabilities and angles edits, shown in Fig. 8.1.179, are accessed with the *primary-to-primary probability* link in the bottom menu for the desired mixture.



From to Group	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6	Group 7	Group 8	Group 9	Group 10
+ 0	8.52241E-01	8.30759E-01	8.08166E-01	8.14207E-01	8.01139E-01	9.25172E-01	8.36424E-01	7.76544E-01	7.82952E-01	7.94157E-01
+ 1	6.61682E-01	6.42139E-01	7.33637E-01	5.61326E-01	6.65760E-01	7.51640E-01	6.80408E-01	7.32950E-01	7.23037E-01	8.73548E-01
+ 2	9.74934E-01	9.85130E-01	9.93821E-01	8.07230E-01	6.98625E-01	9.76555E-01	9.87838E-01	9.70734E-01	8.73983E-01	6.52696E-01
+ 3	9.30279E-01	9.91114E-01	9.93502E-01	9.89998E-01	9.94645E-01	9.91922E-01	9.76020E-01	9.32120E-01	8.43259E-01	9.13903E-01
+ 4	8.71707E-01	9.89243E-01	9.89845E-01	9.88632E-01	9.90806E-01	9.81214E-01	9.62139E-01	9.63141E-01	9.93654E-01	6.19187E-01
+ 5	8.30503E-01	9.83907E-01	9.83380E-01	9.83573E-01	9.80597E-01	9.67756E-01	9.73597E-01	9.94935E-01	9.00071E-01	6.67394E-01
+ 6	7.78709E-01	9.47569E-01	9.70719E-01	9.71324E-01	9.67987E-01	9.74418E-01	9.91481E-01	9.54307E-01	9.62191E-01	7.15266E-01
+ 7	7.10368E-01	8.48228E-01	9.48709E-01	9.55968E-01	9.72320E-01	9.89297E-01	9.67979E-01	9.72426E-01	9.67662E-01	9.35549E-01
+ 8	7.07303E-01	7.76260E-01	9.26750E-01	9.55941E-01	9.85542E-01	9.68681E-01	9.76676E-01	9.94214E-01	9.92196E-01	9.94736E-01
+ 9	7.97793E-01	7.68467E-01	9.07549E-01	9.68547E-01	9.71042E-01	9.72540E-01	9.94893E-01	9.99000E-01	9.99366E-01	9.99236E-01
+ 10	8.44036E-01	8.70794E-01	7.24614E-01	9.63417E-01	9.93167E-01	9.84541E-01	9.99883E-01	9.99983E-01	9.99931E-01	9.99081E-01
+ 11	8.83147E-01	9.22470E-01	6.56377E-01	9.93940E-01	9.99738E-01	9.98408E-01	9.99985E-01	9.99997E-01	9.99937E-01	9.99339E-01
+ 12	9.39707E-01	9.45240E-01	6.35614E-01	9.99647E-01	9.99927E-01	9.99653E-01	9.99995E-01	9.99986E-01	9.99963E-01	9.99611E-01
+ 13	9.67904E-01	9.55921E-01	8.68714E-01	9.99806E-01	9.99912E-01	9.99749E-01	9.99991E-01	9.99994E-01	9.99970E-01	9.98574E-01
+ 14	9.85418E-01	9.35976E-01	9.93876E-01	9.99725E-01	9.99885E-01	9.99672E-01	9.99990E-01	9.99991E-01	9.99984E-01	9.91785E-01
+ 15	9.79772E-01	8.91079E-01	9.96993E-01	9.99627E-01	9.99873E-01	9.99532E-01	9.99985E-01	9.99993E-01	9.99979E-01	9.98598E-01
+ 16	9.62902E-01	8.33433E-01	9.96695E-01	9.99582E-01	9.99852E-01	9.99959E-01	9.99978E-01	9.99976E-01	9.99984E-01	9.99710E-01
+ 17	9.38641E-01	7.85458E-01	9.96433E-01	9.99531E-01	9.99827E-01	5.46613E-01	9.99972E-01	9.99984E-01	9.99992E-01	7.12480E-01
+ 18	9.16097E-01	7.68083E-01	9.96072E-01	9.99448E-01	9.99780E-01	5.00000E-01	9.99977E-01	9.99990E-01	9.99989E-01	5.73969E-01
+ 19	9.06503E-01	7.46499E-01	9.95475E-01	9.99295E-01	9.99717E-01	5.00000E-01	9.99981E-01	9.99980E-01	9.99987E-01	9.99722E-01
+ 20	8.95327E-01	7.19385E-01	9.94159E-01	9.99068E-01	9.99623E-01	5.00000E-01	9.99990E-01	9.99990E-01	9.99988E-01	9.99834E-01
+ 21	8.79638E-01	6.81374E-01	9.92628E-01	9.98835E-01	9.99544E-01	5.00000E-01	9.99990E-01	9.99990E-01	9.99992E-01	9.99838E-01
+ 22	8.53267E-01	6.26865E-01	9.90636E-01	9.98530E-01	9.99416E-01	5.00000E-01	9.99989E-01	9.99976E-01	9.99990E-01	9.99972E-01
+ 23	8.09288E-01	5.86469E-01	9.88328E-01	9.98349E-01	9.99398E-01	5.00000E-01	9.99989E-01	9.99979E-01	9.99995E-01	9.99842E-01
+ 24	7.80579E-01	5.47729E-01	9.86504E-01	9.98291E-01	9.99373E-01	5.00000E-01	9.99988E-01	9.99973E-01	9.99989E-01	9.99823E-01
+ 25	7.44193E-01	5.27592E-01	9.85396E-01	9.98231E-01	9.99428E-01	5.00000E-01	9.99987E-01	9.99979E-01	9.99989E-01	9.99938E-01
+ 26	7.14137E-01	5.22467E-01	9.85560E-01	9.98126E-01	9.99387E-01	5.00000E-01	9.99987E-01	9.99983E-01	9.99991E-01	5.00000E-01
+ 27	7.08161E-01	5.16981E-01	9.84701E-01	9.97999E-01	9.99353E-01	5.00000E-01	9.99985E-01	9.99982E-01	9.99991E-01	5.00000E-01
+ 28	7.01701E-01	5.08103E-01	9.83686E-01	9.97898E-01	9.99320E-01	5.00000E-01	9.99982E-01	9.99973E-01	9.99988E-01	5.00000E-01

Fig. 8.1.178: Example of macroscopic probabilities in HTML output.

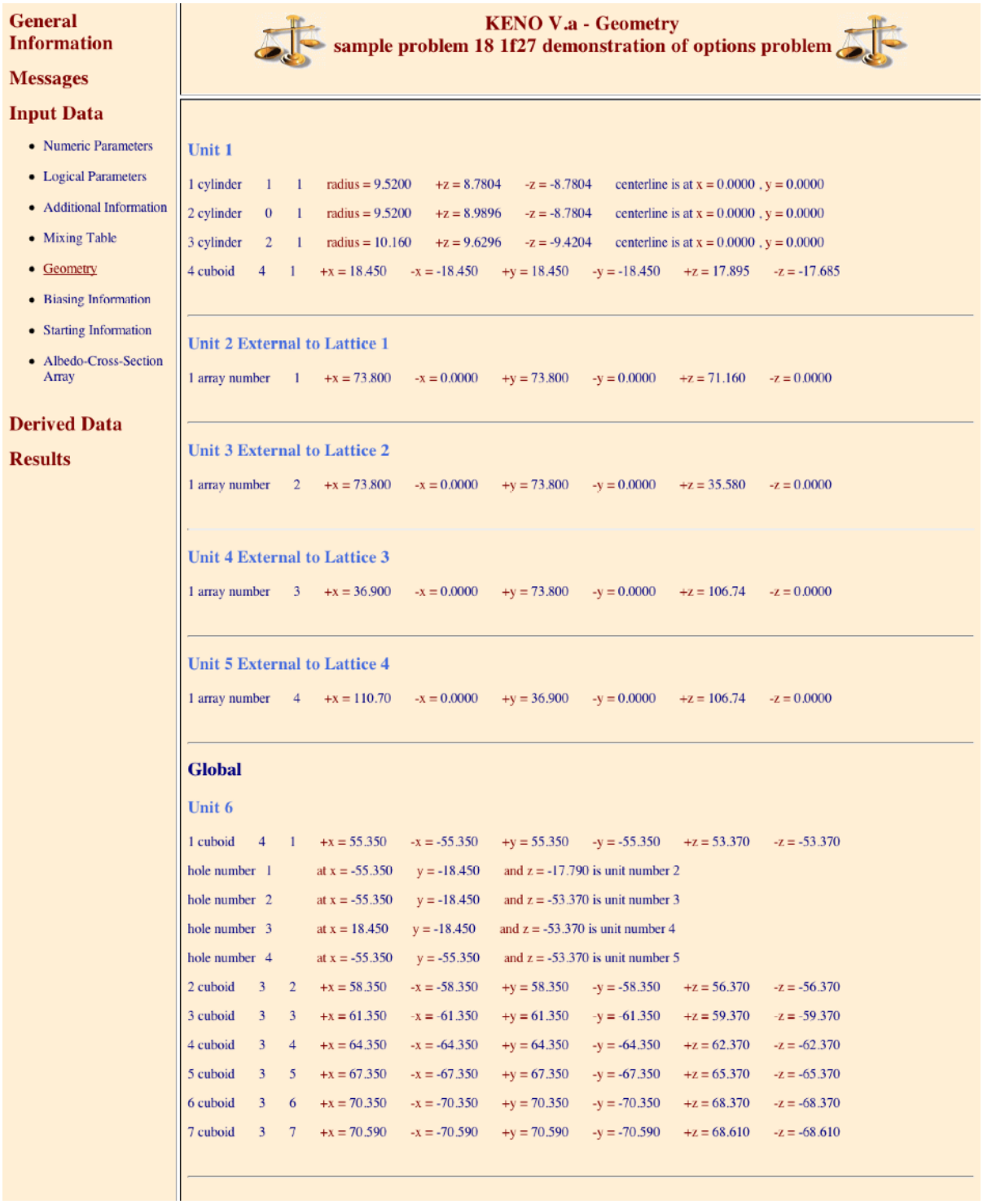


From to Group	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6	Group 7	Group 8	Group 9	Group 10
+ 0	9.00466E-01	9.09010E-01	8.87870E-01	8.50610E-01	8.88801E-01	9.22530E-01	8.88012E-01	8.53759E-01	8.32766E-01	8.18780E-01
+ 1	8.15490E-01	8.31144E-01	8.09725E-01	8.15581E-01	8.34844E-01	8.59485E-01	8.13793E-01	8.12298E-01	7.49822E-01	6.66307E-01
+ 2	7.81563E-01	7.48812E-01	7.15334E-01	8.21374E-01	8.37083E-01	8.18519E-01	7.48644E-01	6.49131E-01	5.05428E-01	3.76234E-01
+ 3	6.56417E-01	6.24330E-01	6.71306E-01	7.69588E-01	7.76540E-01	6.92276E-01	5.67989E-01	4.12011E-01	2.55083E-01	1.98451E-01
+ 4	5.81241E-01	5.86366E-01	6.26593E-01	6.77567E-01	6.54195E-01	5.23696E-01	3.56988E-01	2.03994E-01	1.39054E-01	1.54019E-01
+ 5	5.59772E-01	5.48450E-01	5.51594E-01	5.70642E-01	4.94862E-01	3.29606E-01	1.78840E-01	1.16146E-01	1.00961E-01	6.37242E-02
+ 6	5.53458E-01	5.15595E-01	4.67710E-01	4.33467E-01	3.12185E-01	1.66096E-01	1.02879E-01	8.22664E-02	4.31635E-02	2.68168E-02
+ 7	4.20227E-01	4.59019E-01	3.62758E-01	2.78309E-01	1.58149E-01	9.58220E-02	7.21237E-02	3.42439E-02	1.83656E-02	1.10673E-02
+ 8	4.01897E-01	3.72377E-01	2.43253E-01	1.44409E-01	9.15379E-02	6.70517E-02	3.01759E-02	1.43038E-02	7.91878E-03	5.92443E-03
+ 9	3.07038E-01	2.76675E-01	1.36509E-01	8.40101E-02	6.37001E-02	2.81829E-02	1.25767E-02	6.33208E-03	4.27022E-03	3.98946E-03
+ 10	2.26761E-01	1.29411E-01	1.62575E-01	5.78994E-02	2.59531E-02	1.18904E-02	5.55926E-03	3.51355E-03	2.87663E-03	3.52594E-03
+ 11	1.25801E-01	7.36116E-02	1.66514E-01	2.28922E-02	1.07916E-02	5.20195E-03	3.08075E-03	2.38554E-03	2.53851E-03	3.07746E-03
+ 12	6.79967E-02	4.54777E-02	1.02926E-01	9.41919E-03	4.84593E-03	2.87260E-03	2.09204E-03	2.10812E-03	2.21525E-03	2.60480E-03
+ 13	4.22243E-02	1.46343E-02	2.89257E-02	4.21425E-03	2.68713E-03	1.95430E-03	1.84700E-03	1.83931E-03	1.87814E-03	2.06626E-03
+ 14	1.48279E-02	2.82589E-03	5.26219E-03	2.32646E-03	1.82203E-03	1.72503E-03	1.61380E-03	1.56034E-03	1.47116E-03	1.58487E-03
+ 15	4.95792E-03	-4.33247E-03	2.24513E-03	1.56877E-03	1.60520E-03	1.52128E-03	1.33561E-03	1.20000E-03	8.64477E-04	1.21980E-03
+ 16	2.99135E-04	-1.04366E-02	1.31595E-03	1.36745E-03	1.40244E-03	1.23794E-03	9.99757E-04	4.56768E-04	6.24875E-04	9.56518E-04
+ 17	-2.75728E-03	-1.54219E-02	4.82316E-04	1.19465E-03	1.09536E-03	1.58330E-03	3.19047E-04	4.43081E-06	4.65414E-04	1.34460E-03
+ 18	-4.89460E-03	-1.86851E-02	6.78873E-04	8.94049E-04	-8.17605E-06	-1.20321E-03	6.29549E-06	1.63864E-06	3.03578E-06	1.43806E-03
+ 19	-6.02976E-03	-2.11932E-02	1.82303E-04	-3.47875E-05	2.23587E-05	-6.09573E-04	4.68669E-06	5.23980E-06	3.86665E-06	7.29021E-04
+ 20	-7.26537E-03	-2.45810E-02	-3.57752E-04	-3.68126E-05	-8.59895E-06	-4.97715E-04	-3.29831E-07	1.06459E-06	3.58671E-06	6.22411E-04
+ 21	-8.43190E-03	-3.32157E-02	-2.92318E-04	-6.91726E-05	-1.99372E-05	-1.00007E-02	-2.70091E-07	1.04884E-06	1.85870E-06	5.83043E-04
+ 22	-1.02988E-02	-3.36933E-02	-5.88616E-04	-8.20851E-05	1.18584E-05	-4.93810E-04	1.78572E-07	6.42181E-06	2.90973E-06	5.48062E-04
+ 23	1.39611E-04	-5.18451E-02	-7.59042E-04	-1.07570E-04	1.14640E-05	-4.93812E-04	-1.12128E-07	5.13631E-06	7.87091E-07	4.64784E-04
+ 24	-1.58823E-02	-6.28656E-02	-7.56608E-04	-1.09434E-04	1.36297E-05	-3.23275E-04	-1.24095E-07	7.47155E-06	2.87335E-06	4.19869E-04
+ 25	-2.09027E-02	-7.08387E-02	-5.32529E-04	-1.14174E-04	-3.40139E-05	-1.00007E-02	1.58147E-07	5.09399E-06	3.09216E-06	2.86938E-04
+ 26	-1.31040E-02	-7.27628E-02	-8.06804E-04	-1.22021E-04	-3.51869E-05	-5.27905E-04	-1.48729E-07	2.92542E-06	2.05590E-06	-8.30587E-04
+ 27	-1.35001E-02	-7.47823E-02	-8.56426E-04	-1.30255E-04	-3.54604E-05	-1.00007E-02	-1.53626E-07	2.85728E-06	1.68725E-06	-1.15515E-03
+ 28	-1.39352E-02	-7.83385E-02	-9.14311E-04	-1.36734E-04	-3.91336E-05	-3.23275E-04	6.02032E-07	7.32591E-06	2.34181E-06	-9.62229E-04

Fig. 8.1.179: Example of macroscopic angles in HTML output.

HTML output: Geometry data

The KENO V.a HTML version of the geometry region printout is shown in Fig. 8.1.180. The HTML edits of geometry are accessed with the *Geometry* link in the *Input Data* section.



General Information

Messages

Input Data

- Numeric Parameters
- Logical Parameters
- Additional Information
- Mixing Table
- Geometry
- Biasing Information
- Starting Information
- Albedo-Cross-Section Array

Derived Data

Results

KENO V.a - Geometry
sample problem 18 1f27 demonstration of options problem

Unit 1

1 cylinder 1 1 radius = 9.5200 +z = 8.7804 -z = -8.7804 centerline is at x = 0.0000 , y = 0.0000
2 cylinder 0 1 radius = 9.5200 +z = 8.9896 -z = -8.7804 centerline is at x = 0.0000 , y = 0.0000
3 cylinder 2 1 radius = 10.160 +z = 9.6296 -z = -9.4204 centerline is at x = 0.0000 , y = 0.0000
4 cuboid 4 1 +x = 18.450 -x = -18.450 +y = 18.450 -y = -18.450 +z = 17.895 -z = -17.685

Unit 2 External to Lattice 1

1 array number 1 +x = 73.800 -x = 0.0000 +y = 73.800 -y = 0.0000 +z = 71.160 -z = 0.0000

Unit 3 External to Lattice 2

1 array number 2 +x = 73.800 -x = 0.0000 +y = 73.800 -y = 0.0000 +z = 35.580 -z = 0.0000

Unit 4 External to Lattice 3

1 array number 3 +x = 36.900 -x = 0.0000 +y = 73.800 -y = 0.0000 +z = 106.74 -z = 0.0000

Unit 5 External to Lattice 4

1 array number 4 +x = 110.70 -x = 0.0000 +y = 36.900 -y = 0.0000 +z = 106.74 -z = 0.0000

Global

Unit 6

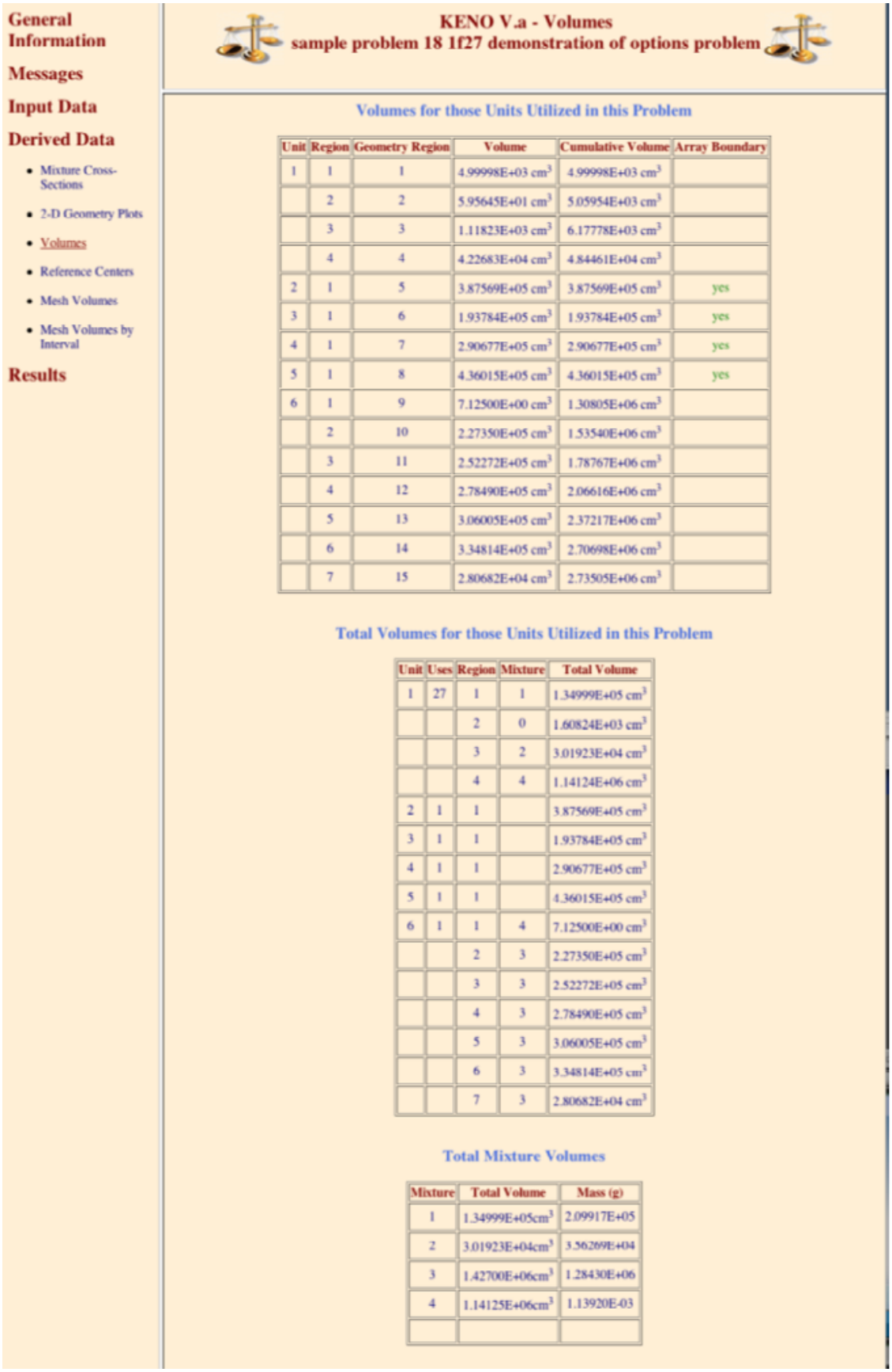
1 cuboid 4 1 +x = 55.350 -x = -55.350 +y = 55.350 -y = -55.350 +z = 53.370 -z = -53.370
hole number 1 at x = -55.350 y = -18.450 and z = -17.790 is unit number 2
hole number 2 at x = -55.350 y = -18.450 and z = -53.370 is unit number 3
hole number 3 at x = 18.450 y = -18.450 and z = -53.370 is unit number 4
hole number 4 at x = -55.350 y = -55.350 and z = -53.370 is unit number 5
2 cuboid 3 2 +x = 58.350 -x = -58.350 +y = 58.350 -y = -58.350 +z = 56.370 -z = -56.370
3 cuboid 3 3 +x = 61.350 -x = -61.350 +y = 61.350 -y = -61.350 +z = 59.370 -z = -59.370
4 cuboid 3 4 +x = 64.350 -x = -64.350 +y = 64.350 -y = -64.350 +z = 62.370 -z = -62.370
5 cuboid 3 5 +x = 67.350 -x = -67.350 +y = 67.350 -y = -67.350 +z = 65.370 -z = -65.370
6 cuboid 3 6 +x = 70.350 -x = -70.350 +y = 70.350 -y = -70.350 +z = 68.370 -z = -68.370
7 cuboid 3 7 +x = 70.590 -x = -70.590 +y = 70.590 -y = -70.590 +z = 68.610 -z = -68.610

Fig. 8.1.180: Example of geometry region data in HTML output.

HTML output: Volume information

KENO V.a

Three tables of volumes are always printed. An example of the volume printout is given in Fig. 8.1.181. The HTML edit is accessed with the *Volumes* link in the *Derived Data* section.



General Information

Messages

Input Data

Derived Data

- Mixture Cross-Sections
- 2-D Geometry Plots
- Volumes
- Reference Centers
- Mesh Volumes
- Mesh Volumes by Interval

Results

KENO V.a - Volumes
sample problem 18 1f27 demonstration of options problem

Volumes for those Units Utilized in this Problem

Unit	Region	Geometry Region	Volume	Cumulative Volume	Array Boundary
1	1	1	4.99998E+03 cm^3	4.99998E+03 cm^3	
	2	2	5.95645E+01 cm^3	5.05954E+03 cm^3	
	3	3	1.11823E+03 cm^3	6.17778E+03 cm^3	
	4	4	4.22683E+04 cm^3	4.84461E+04 cm^3	
2	1	5	3.87569E+05 cm^3	3.87569E+05 cm^3	yes
3	1	6	1.93784E+05 cm^3	1.93784E+05 cm^3	yes
4	1	7	2.90677E+05 cm^3	2.90677E+05 cm^3	yes
5	1	8	4.36015E+05 cm^3	4.36015E+05 cm^3	yes
6	1	9	7.12500E+00 cm^3	1.30805E+06 cm^3	
	2	10	2.27350E+05 cm^3	1.53540E+06 cm^3	
	3	11	2.52272E+05 cm^3	1.78767E+06 cm^3	
	4	12	2.78490E+05 cm^3	2.06616E+06 cm^3	
	5	13	3.06005E+05 cm^3	2.37217E+06 cm^3	
	6	14	3.34814E+05 cm^3	2.70698E+06 cm^3	
	7	15	2.80682E+04 cm^3	2.73505E+06 cm^3	

Total Volumes for those Units Utilized in this Problem

Unit	Uses	Region	Mixture	Total Volume
1	27	1	1	1.34999E+05 cm^3
		2	0	1.60824E+03 cm^3
		3	2	3.01923E+04 cm^3
		4	4	1.14124E+06 cm^3
2	1	1		3.87569E+05 cm^3
3	1	1		1.93784E+05 cm^3
4	1	1		2.90677E+05 cm^3
5	1	1		4.36015E+05 cm^3
6	1	1	4	7.12500E+00 cm^3
		2	3	2.27350E+05 cm^3
		3	3	2.52272E+05 cm^3
		4	3	2.78490E+05 cm^3
		5	3	3.06005E+05 cm^3
		6	3	3.34814E+05 cm^3
		7	3	2.80682E+04 cm^3

Total Mixture Volumes

Mixture	Total Volume	Mass (g)
1	1.34999E+05cm^3	2.09917E+05
2	3.01923E+04cm^3	3.56269E+04
3	1.42700E+06cm^3	1.28430E+06
4	1.14125E+06cm^3	1.13920E-03

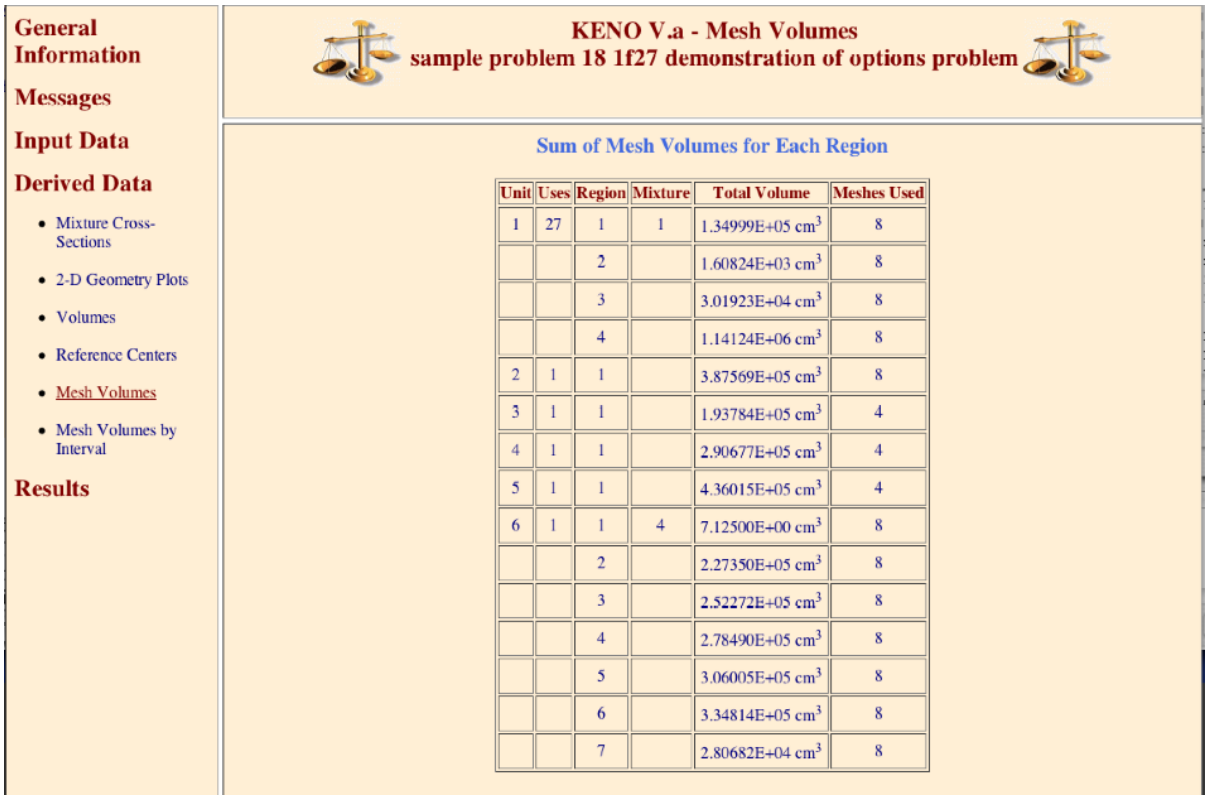
Fig. 8.1.181: Sample volume information in HTML output.

KENO-VI

Two tables of volumes are always printed. The first table will list the number of times each unit is used in the problem and the total volume of each region throughout the entire problem description. The second table will list the total volume and mass of each mixture used in the problem.

Mesh volumes

With the mesh flux accumulator, fluxes are tabulated for each region of each unit in a cubic mesh. For models in which the mesh flux accumulator is activated by setting **MFX**=YES and the mesh size set to a positive value with **MSH**=, the cumulative volume of all mesh intervals for a given region and the number of meshes used in each region are printed as shown in Fig. 8.1.182. The HTML edit is accessed with the *Mesh Volumes* link in the *Derived Data* section.



Unit	Uses	Region	Mixture	Total Volume	Meshes Used
1	27	1	1	1.34999E+05 cm^3	8
		2		1.60824E+03 cm^3	8
		3		3.01923E+04 cm^3	8
		4		1.14124E+06 cm^3	8
2	1	1		3.87569E+05 cm^3	8
3	1	1		1.93784E+05 cm^3	4
4	1	1		2.90677E+05 cm^3	4
5	1	1		4.36015E+05 cm^3	4
6	1	1	4	7.12500E+00 cm^3	8
		2		2.27350E+05 cm^3	8
		3		2.52272E+05 cm^3	8
		4		2.78490E+05 cm^3	8
		5		3.06005E+05 cm^3	8
		6		3.34814E+05 cm^3	8
		7		2.80682E+04 cm^3	8

Fig. 8.1.182: Sample sum of mesh volumes in HTML output.

An optional edit of volumes for each mesh interval for each region can be activated by entering **PMV**=YES in the parameter input. This edit can be very large, especially if a small mesh size is used with a large model. A sample of the volume by mesh for each region is shown in Fig. 8.1.183. This edit is found under *Derived Data* with the *Mesh Volumes by Interval* link. For the HTML output, the links on the right side of the screen control what data are displayed in the center pane. Clicking on the unit number in the pane labeled **UNIT** places the pane labeled Region at that unit. Clicking on a region number in the region pane places the volume by mesh for the selected unit in the center pane.

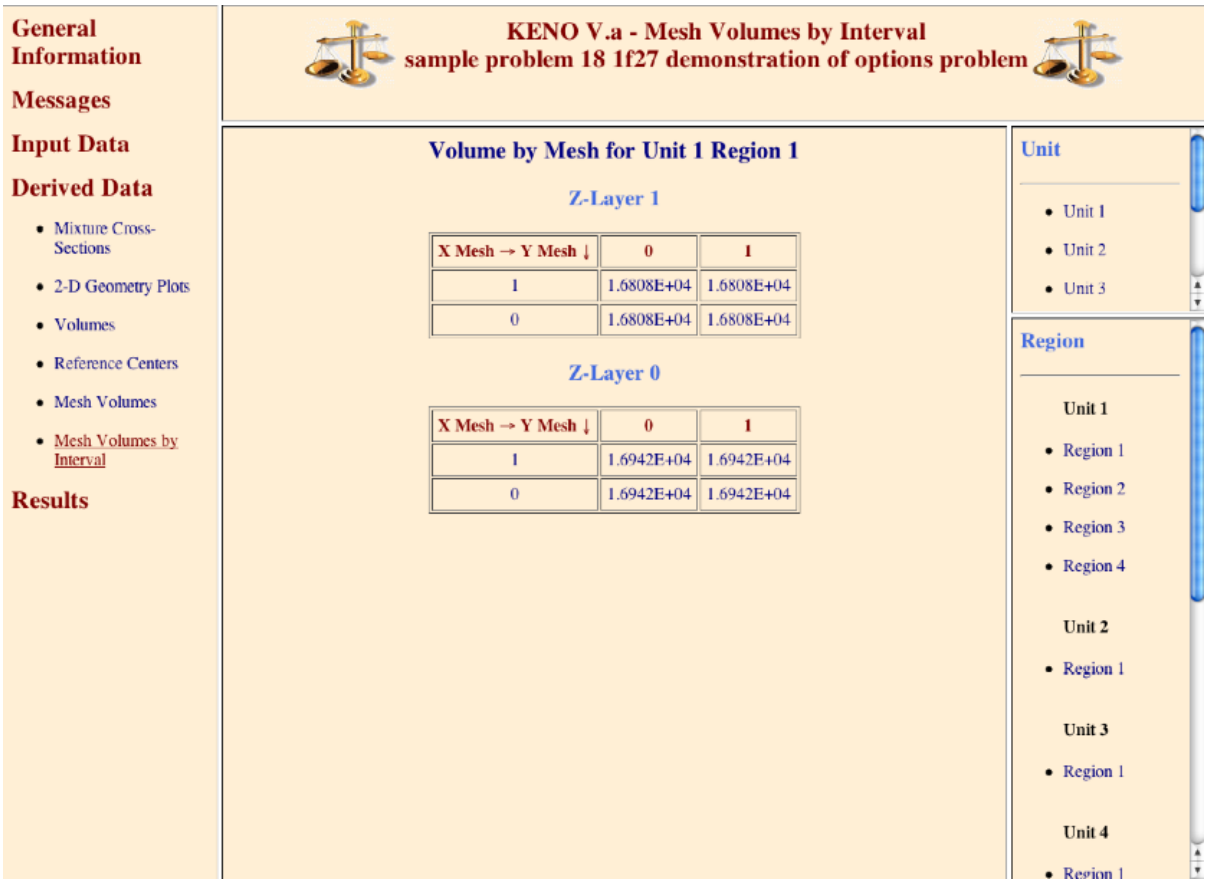


Fig. 8.1.183: Sample of volume by mesh for each region in HTML output.

HTML output: Biasing information

This table specifies the weighting or biasing data to be used in the problem. An example of biasing information is given in Fig. 8.1.184.

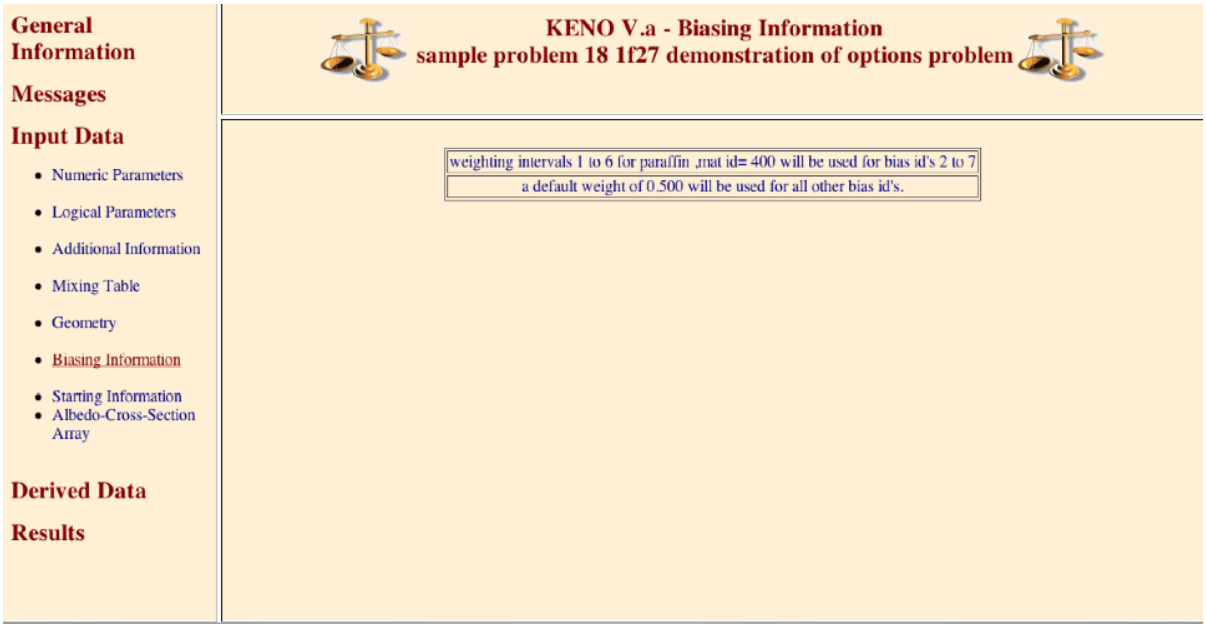


Fig. 8.1.184: Biasing information in HTML output.

HTML output: Group-dependent weights

Printing the group-dependent weights is optional. They are printed if **PWT**=YES is entered in the parameter data. This edit is available in the HTML output under *Results* in the *Average Weight* link, as shown in Fig. 8.1.185.

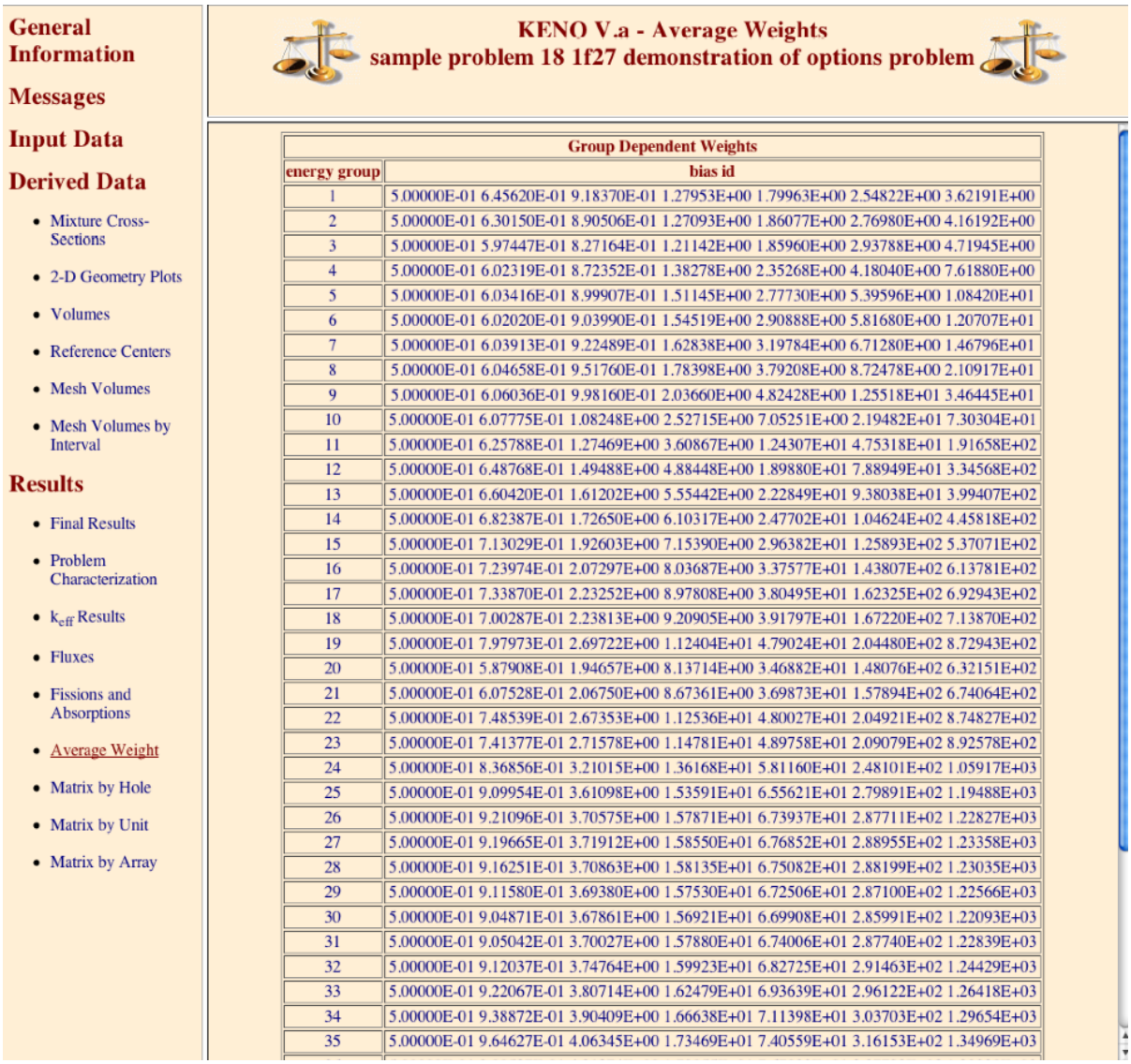


Fig. 8.1.185: Example of biasing data in HTML output.

HTML output: Plot representation

Plots representing 2-D slices through the geometrical description of the problem are optional. They are created if plot data are entered unless **PLT**=NO is specified either in the plot data or the parameter data. Character plots are not available in the HTML output, but the color plots are easily viewed as shown in Fig. 8.1.186. The user-requested plots can be viewed under *Derived Data* in the *2-D Geometry Plots* link.

Fig. 8.1.186 shows two color plots in a single HTML page. The color plots do not appear in the KENO text printout, but they are an integral part of the HTML output. They are generated from PNG files created when a color plot has been specified in the KENO input data. Any number of plots can be made in one problem. The color plots can be valuable tools to assist the user in verifying that a problem is described correctly.

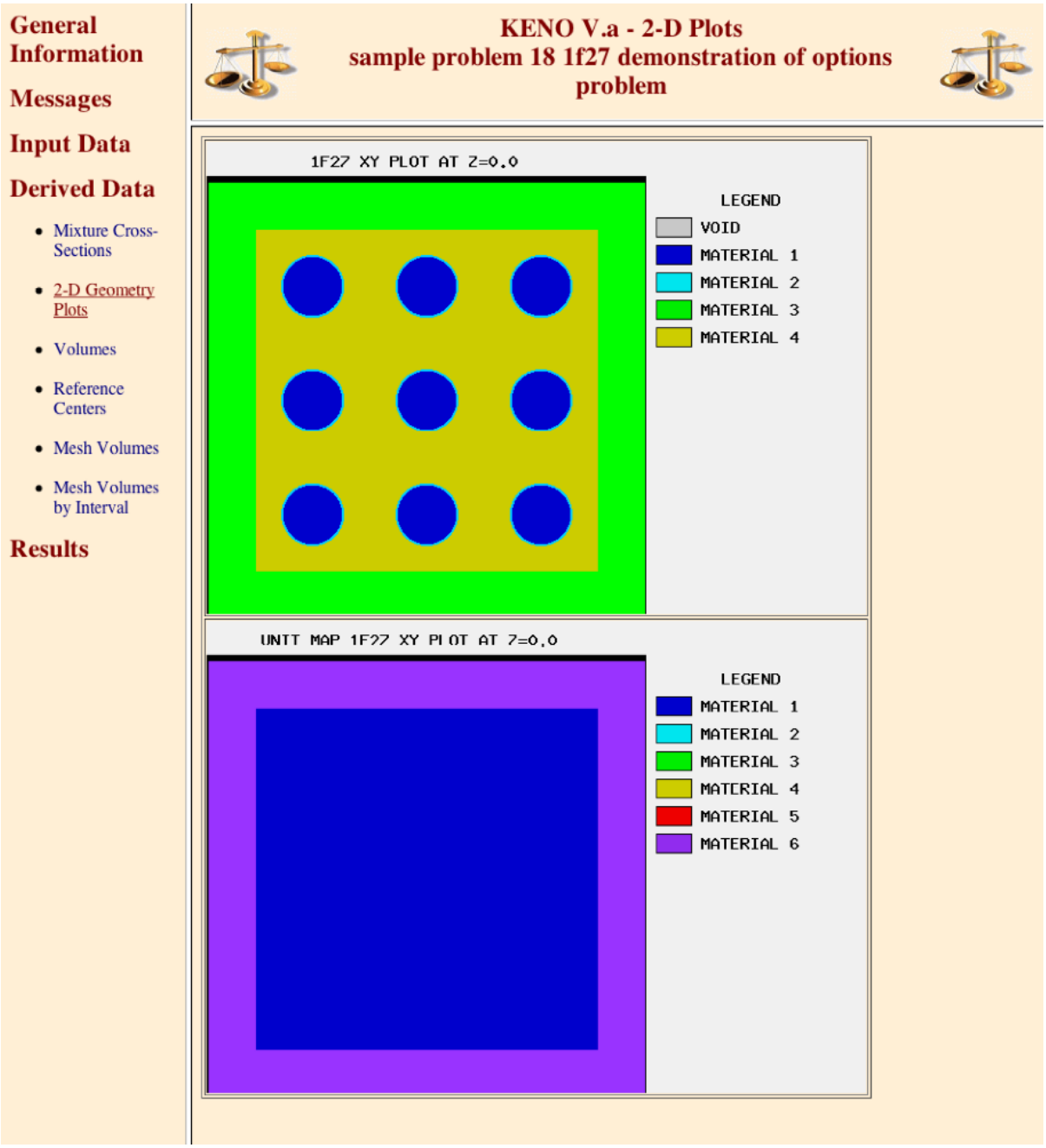


Fig. 8.1.186: Sample color plots in HTML output.

HTML output: Initial source and final pretracking edits

Prior to calculating the k-effs for each generation, KENO prints the final input data edit. The starting information is available in the HTML output under *Input Data* with the *Starting Information* link. An example of the HTML edit is shown in Fig. 8.1.187.

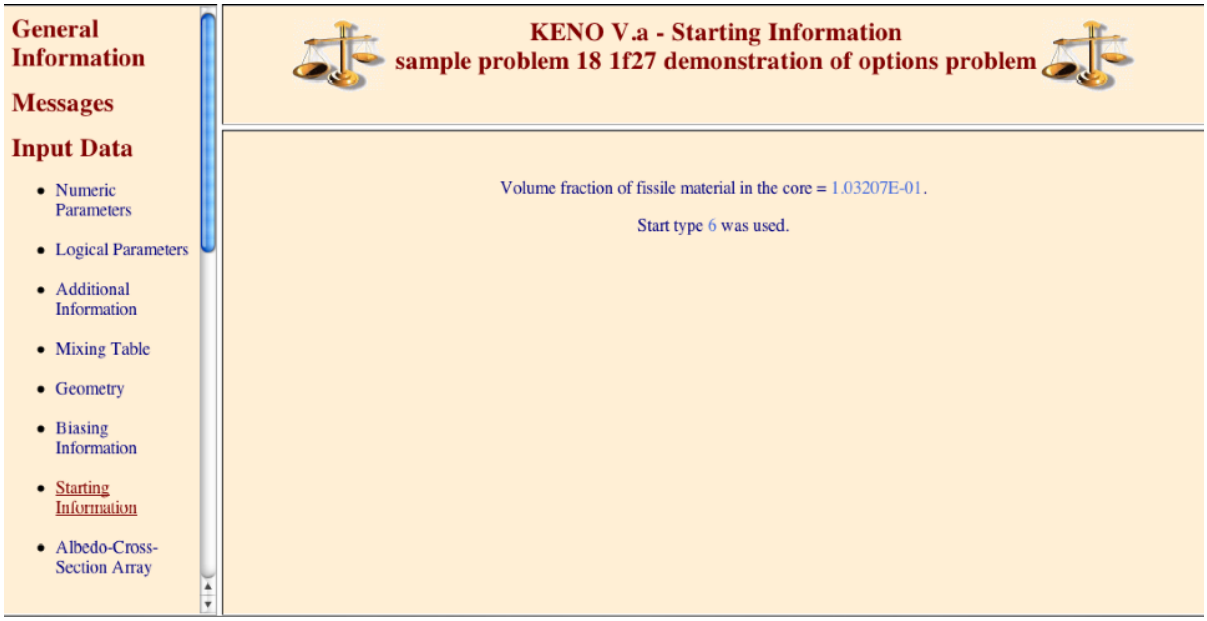


Fig. 8.1.187: Example of initial source information for start type 6 in HTML output.

HTML output: Reference center for flux moment/angular flux tranform

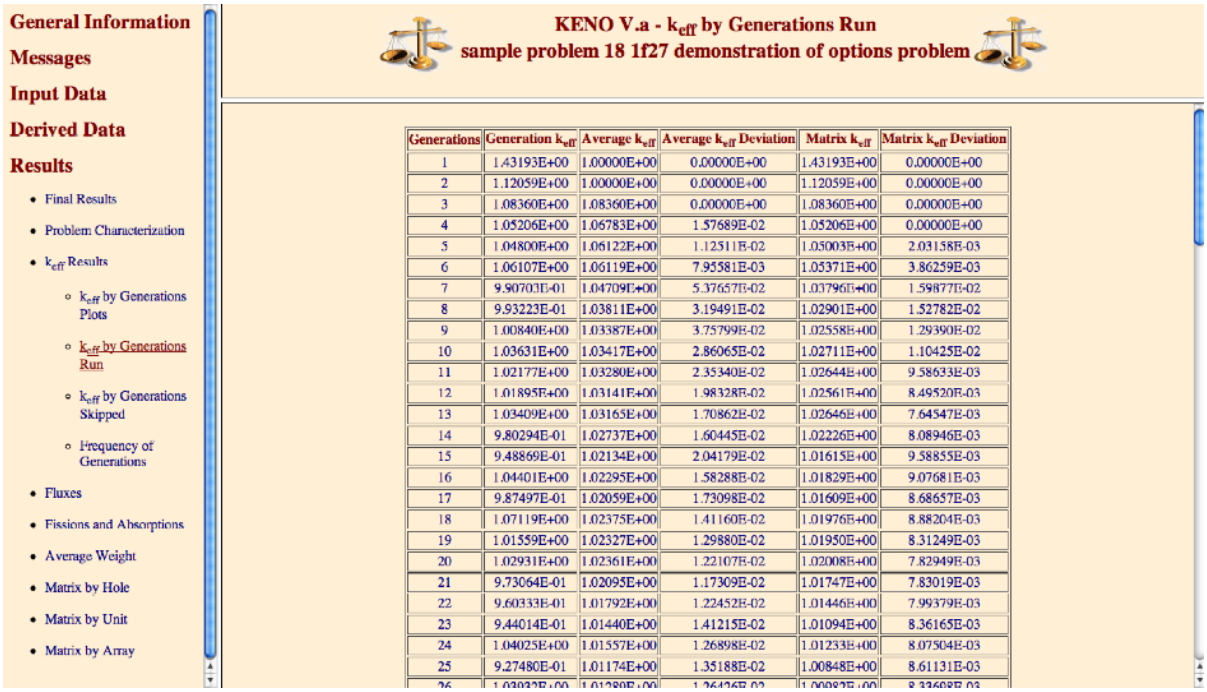
When **TFM**=YES is entered, the flux moments and angular fluxes are computed in a transformed coordinate system relative to a reference center point. Edits of the fuel center and the position of the reference center for each region are shown in Fig. 8.1.188. The HTML output edit is available under *Derived Data* in the *Reference Centers* link.

Fig. 8.1.188: Example of fuel center and reference center edits in HTML output.

HTML output: K-effectives by generation

At the completion of each generation, KENO prints the k-effective for that generation and associated information. An example of this printout is given in Fig. 8.1.188 for HTML output. This HTML output edit is available under *Results* in the section of *keff Results* in the *keff by Generations Run* link.

Example fuel center and reference center edits in HTML output.



Generations	Generation k_{eff}	Average k_{eff}	Average k_{eff} Deviation	Matrix k_{eff}	Matrix k_{eff} Deviation
1	1.43193E+00	1.00000E+00	0.00000E+00	1.43193E+00	0.00000E+00
2	1.12059E+00	1.00000E+00	0.00000E+00	1.12059E+00	0.00000E+00
3	1.08360E+00	1.08360E+00	0.00000E+00	1.08360E+00	0.00000E+00
4	1.05206E+00	1.06783E+00	1.57689E-02	1.05206E+00	0.00000E+00
5	1.04800E+00	1.06122E+00	1.12511E-02	1.05003E+00	2.03158E-03
6	1.06107E+00	1.06119E+00	7.95581E-03	1.05371E+00	3.86259E-03
7	9.90703E-01	1.04709E+00	5.37657E-02	1.03796E+00	1.59877E-02
8	9.93223E-01	1.03811E+00	3.19491E-02	1.02901E+00	1.52782E-02
9	1.00840E+00	1.03387E+00	3.75799E-02	1.02558E+00	1.29390E-02
10	1.03631E+00	1.03417E+00	2.86065E-02	1.02711E+00	1.10425E-02
11	1.02177E+00	1.03280E+00	2.35340E-02	1.02644E+00	9.58633E-03
12	1.01895E+00	1.03141E+00	1.98328E-02	1.02561E+00	8.49520E-03
13	1.03409E+00	1.03165E+00	1.70862E-02	1.02646E+00	7.64547E-03
14	9.80294E-01	1.02737E+00	1.60445E-02	1.02226E+00	8.08946E-03
15	9.48869E-01	1.02134E+00	2.04179E-02	1.01615E+00	9.58855E-03
16	1.04401E+00	1.02295E+00	1.58288E-02	1.01829E+00	9.07681E-03
17	9.87497E-01	1.02059E+00	1.73098E-02	1.01609E+00	8.68657E-03
18	1.07119E+00	1.02375E+00	1.41160E-02	1.01976E+00	8.88204E-03
19	1.01559E+00	1.02327E+00	1.29880E-02	1.01950E+00	8.31249E-03
20	1.02931E+00	1.02361E+00	1.22107E-02	1.02008E+00	7.82949E-03
21	9.73064E-01	1.02095E+00	1.17309E-02	1.01747E+00	7.83019E-03
22	9.60333E-01	1.01792E+00	1.22452E-02	1.01446E+00	7.99379E-03
23	9.44014E-01	1.01440E+00	1.41215E-02	1.01094E+00	8.36165E-03
24	1.04025E+00	1.01557E+00	1.26898E-02	1.01233E+00	8.07504E-03
25	9.27480E-01	1.01174E+00	1.35188E-02	1.00848E+00	8.61131E-03
26	1.03032E+00	1.01289E+00	1.26426E-02	1.00982E+00	8.33698E-03

Fig. 8.1.189: Example of k-effective by generation in HTML output.

The problem title is printed at the top of the page. A descriptive heading is printed at the top of each column of data. The data printed include (1) the generation number, (2) the k-effective calculated for the generation, (3) the average value of k-effective through the current generation (excluding the first two generations), (4) the deviation associated with the average k-effective, and (5) the matrix k-effective for the generation and the deviation associated with the matrix k-effective. The last two columns are filled with zeros if the user did not specify matrix k-effective calculations. The matrix k-effective is the largest eigenvalue of the fission production matrix. Matrix information can be calculated based on (1) position index, (2) unit number, (3) hole number, and (4) array number. The matrix k-effective printed in the sixth column is based on this order. If the matrix k-effective is calculated by position index, it is the one printed in the sixth column. The matrix k-effective by unit number is given second preference, followed by hole number and then array number. After the last generation, a message is printed to indicate why execution was terminated. If matrix k-effectives were calculated, this is followed by a message stating the method used to determine the matrix k-effective.

The user should examine this portion of the results to ensure that the two methods of calculating k-effective are in acceptable agreement and to verify that the average value of k-effective has become relatively stable. If the k-effectives appear to be oscillating or drifting significantly, the user should consider rerunning the problem with a larger number of histories per generation.

If a problem is restarted, the generation numbers and k-effectives are printed and the words FROM RESTART UNIT are printed in the elapsed time column. All other columns are blank. When the generation at which the problem is to be restarted is reached, the print reverts to the normal format as shown in Fig. 8.1.137.

HTML output: Problem characterization edit

The problem characterization edit is shown in Fig. 8.1.190 for HTML output.

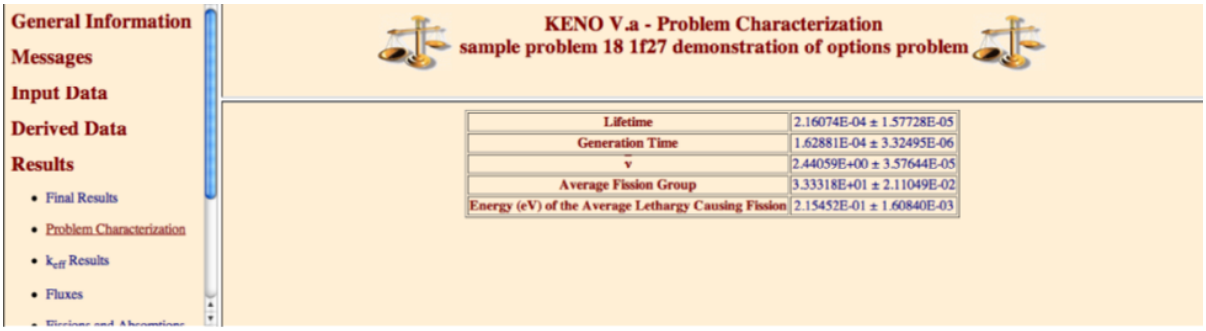
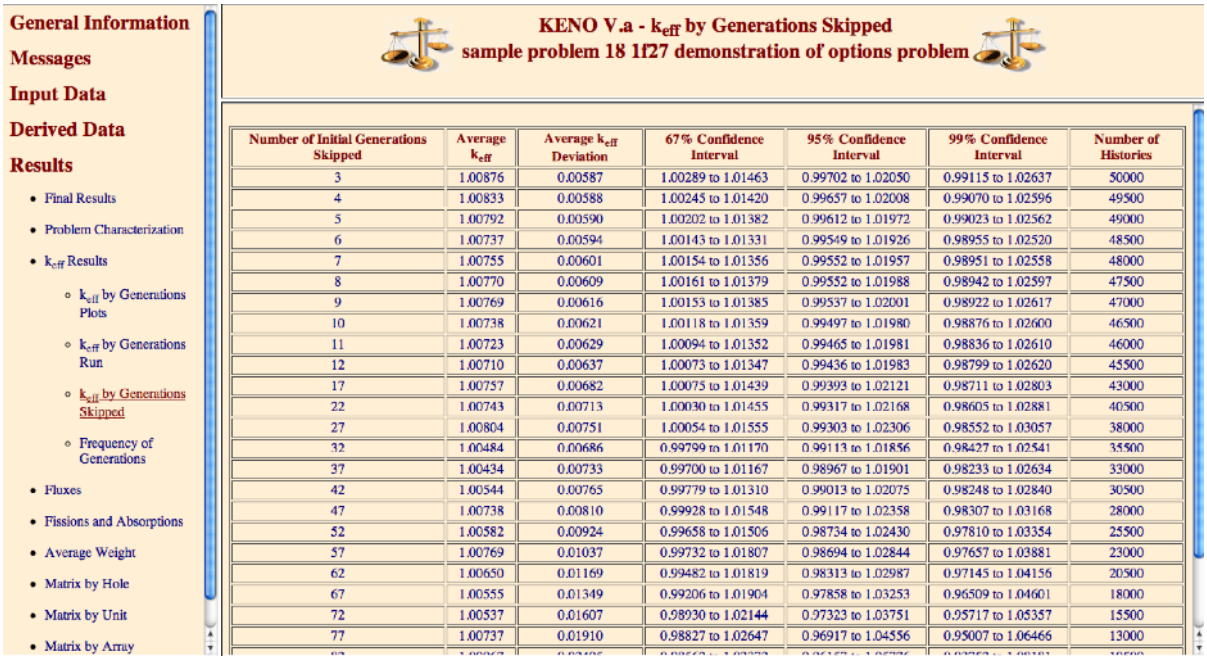


Fig. 8.1.190: Example problem characterization edit in HTML output.

HTML output: Final k-effective edit

The final k-effective edit prints the average k-effective and its associated deviation and the limits of k-effective for the 67, 95, and 99% confidence intervals. The final k-effective edit is printed as shown in Fig. 8.1.191. This HTML output edit is available under *Results* in the section of *keff Results* in the *keff by Generations Skipped* link.



Number of Initial Generations Skipped	Average k_{eff}	Average k_{eff} Deviation	67% Confidence Interval	95% Confidence Interval	99% Confidence Interval	Number of Histories
3	1.00876	0.00587	1.00289 to 1.01463	0.99702 to 1.02050	0.99115 to 1.02637	50000
4	1.00833	0.00588	1.00245 to 1.01420	0.99657 to 1.02008	0.99070 to 1.02596	49500
5	1.00792	0.00590	1.00202 to 1.01382	0.99612 to 1.01972	0.99023 to 1.02562	49000
6	1.00737	0.00594	1.00143 to 1.01331	0.99549 to 1.01926	0.98955 to 1.02520	48500
7	1.00755	0.00601	1.00154 to 1.01356	0.99552 to 1.01957	0.98951 to 1.02558	48000
8	1.00770	0.00609	1.00161 to 1.01379	0.99552 to 1.01988	0.98942 to 1.02597	47500
9	1.00769	0.00616	1.00153 to 1.01385	0.99537 to 1.02001	0.98922 to 1.02617	47000
10	1.00738	0.00621	1.00118 to 1.01359	0.99497 to 1.01980	0.98876 to 1.02600	46500
11	1.00723	0.00629	1.00094 to 1.01352	0.99465 to 1.01981	0.98836 to 1.02610	46000
12	1.00710	0.00637	1.00073 to 1.01347	0.99436 to 1.01983	0.98799 to 1.02620	45500
17	1.00757	0.00682	1.00075 to 1.01439	0.99393 to 1.02121	0.98711 to 1.02803	43000
22	1.00743	0.00713	1.00030 to 1.01455	0.99317 to 1.02168	0.98605 to 1.02881	40500
27	1.00804	0.00751	1.00054 to 1.01555	0.99303 to 1.02306	0.98552 to 1.03057	38000
32	1.00484	0.00686	0.99799 to 1.01170	0.99113 to 1.01856	0.98427 to 1.02541	35500
37	1.00434	0.00733	0.99700 to 1.01167	0.98967 to 1.01901	0.98233 to 1.02634	33000
42	1.00544	0.00765	0.99779 to 1.01310	0.99013 to 1.02075	0.98248 to 1.02840	30500
47	1.00738	0.00810	0.99928 to 1.01548	0.99117 to 1.02358	0.98307 to 1.03168	28000
52	1.00582	0.00924	0.99658 to 1.01506	0.98734 to 1.02430	0.97810 to 1.03354	25500
57	1.00769	0.01037	0.99732 to 1.01807	0.98694 to 1.02844	0.97657 to 1.03881	23000
62	1.00650	0.01169	0.99482 to 1.01819	0.98313 to 1.02987	0.97145 to 1.04156	20500
67	1.00555	0.01349	0.99206 to 1.01904	0.97858 to 1.03253	0.96509 to 1.04601	18000
72	1.00537	0.01607	0.98930 to 1.02144	0.97323 to 1.03751	0.95717 to 1.05357	15500
77	1.00737	0.01910	0.98827 to 1.02647	0.96917 to 1.04556	0.95007 to 1.06466	13000

Fig. 8.1.191: Example of the final k-effective edit in HTML output.

HTML output: Final edit of fissions, absorptions, and leakage

The final edit of fissions, absorptions, and leakage follows the final k-effective edit and includes the fission fraction for each group and the fission production, absorptions, and leakage, each with its associated percent deviation.

Tables corresponding to those available in the text output are also available in the HTML output. These data are accessed with the *Fissions and Absorptions Data* link under *Fissions and Absorptions* in the *Results* section, as illustrated in Fig. 8.1.192, where the region and group data (**FAR**=YES and **GAS**=YES) are present. To allow quicker loading of the large amounts of data in the HTML output, the data have been subdivided into several separate tables, which are accessed with the menu on the right side of the window.

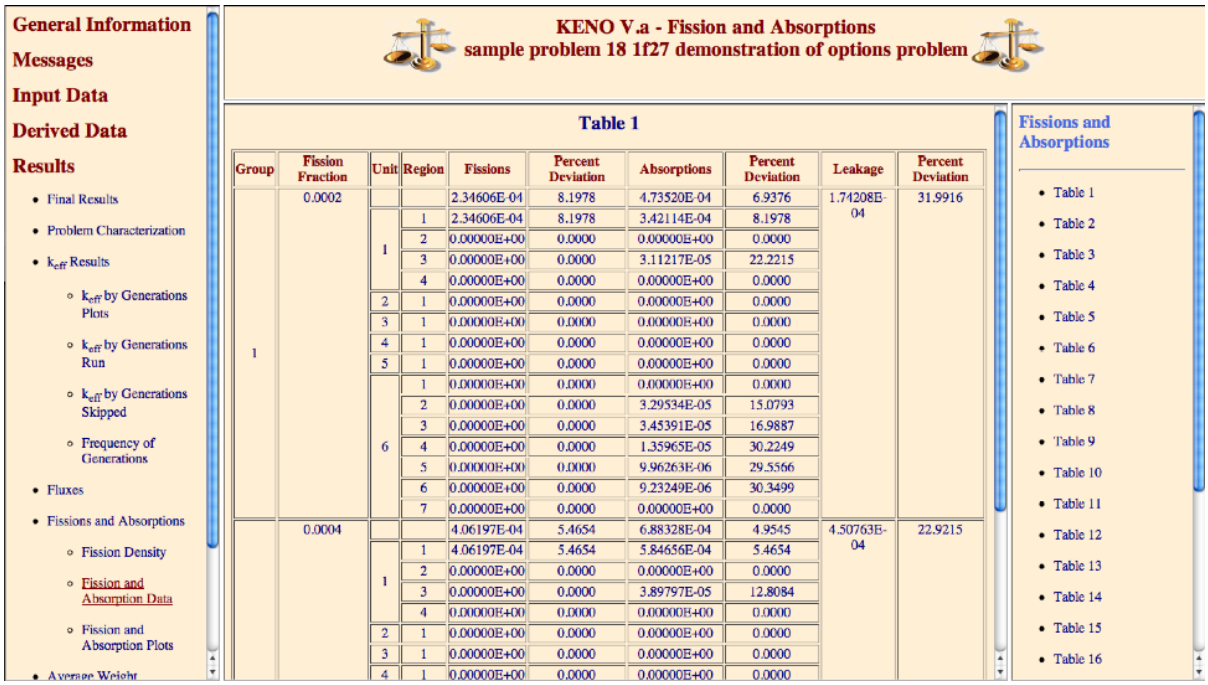
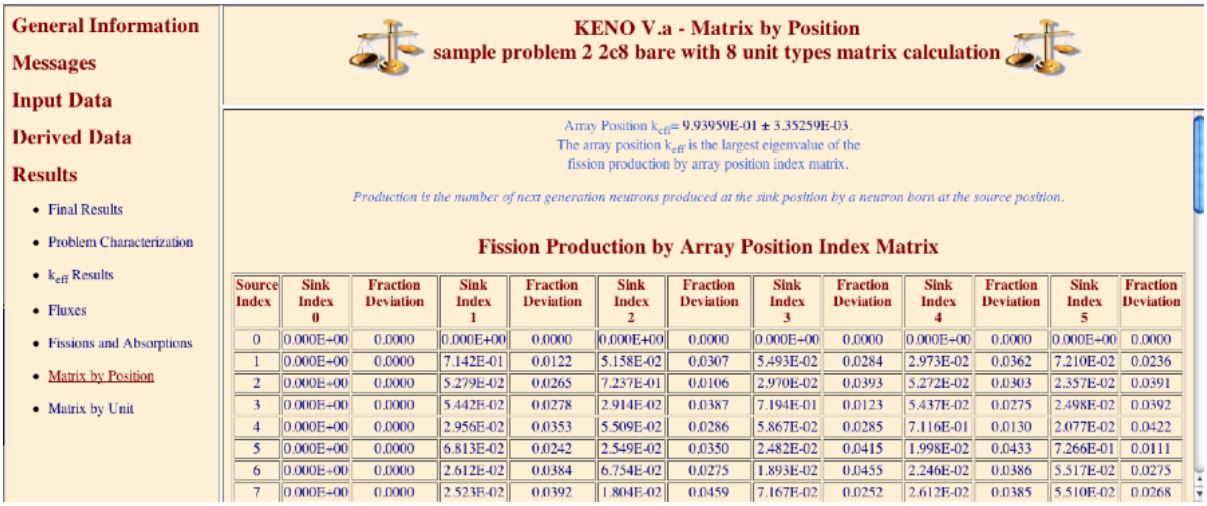


Fig. 8.1.192: Sample fissions, absorptions and leakage data in HTML output.

HTML output: Matrix k-effective by position index

The matrix k-effective by unit location (also referred to as array position or position index) is calculated if **MKP**=YES is specified in the parameter data and a global array is present in the model. An example of the matrix k-effective by position is given in Fig. 8.1.193. In the HTML output, the k-effective by position data is available in the *Matrix by Position* link in the *Results* section.



General Information
Messages
Input Data
Derived Data
Results
- Final Results
- Problem Characterization
- k_{eff} Results
- Fluxes
- Fissions and Absorptions
- Matrix by Position
- Matrix by Unit

KENO V.a - Matrix by Position
sample problem 2 2c8 bare with 8 unit types matrix calculation

Array Position k_{eff}= 9.93959E-01 ± 3.35259E-03.
The array position k_{eff} is the largest eigenvalue of the fission production by array position index matrix.

Production is the number of next generation neutrons produced at the sink position by a neutron born at the source position.

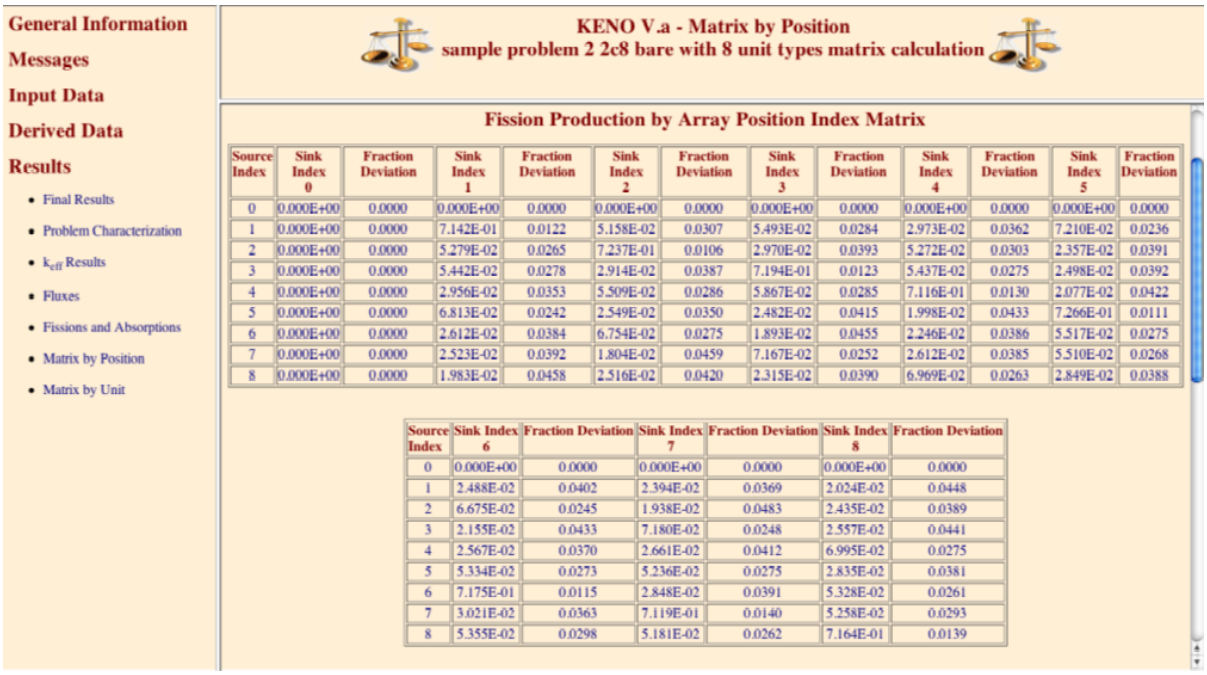
Fission Production by Array Position Index Matrix

Source Index	Sink Index 0	Fraction Deviation	Sink Index 1	Fraction Deviation	Sink Index 2	Fraction Deviation	Sink Index 3	Fraction Deviation	Sink Index 4	Fraction Deviation	Sink Index 5	Fraction Deviation
0	0.000E+00	0.0000	0.000E+00	0.0000	0.000E+00	0.0000	0.000E+00	0.0000	0.000E+00	0.0000	0.000E+00	0.0000
1	0.000E+00	0.0000	7.142E-01	0.0122	5.158E-02	0.0307	5.493E-02	0.0284	2.973E-02	0.0362	7.210E-02	0.0236
2	0.000E+00	0.0000	5.279E-02	0.0265	7.237E-01	0.0106	2.970E-02	0.0393	5.272E-02	0.0303	2.357E-02	0.0391
3	0.000E+00	0.0000	5.442E-02	0.0278	2.914E-02	0.0387	7.194E-01	0.0123	5.437E-02	0.0275	2.498E-02	0.0392
4	0.000E+00	0.0000	2.956E-02	0.0353	5.509E-02	0.0286	5.867E-02	0.0285	7.116E-01	0.0130	2.077E-02	0.0422
5	0.000E+00	0.0000	6.813E-02	0.0242	2.549E-02	0.0350	2.482E-02	0.0415	1.998E-02	0.0433	7.266E-01	0.0111
6	0.000E+00	0.0000	2.612E-02	0.0384	6.754E-02	0.0275	1.893E-02	0.0455	2.246E-02	0.0386	5.517E-02	0.0275
7	0.000E+00	0.0000	2.523E-02	0.0392	1.804E-02	0.0459	7.167E-02	0.0252	2.612E-02	0.0385	5.510E-02	0.0268

Fig. 8.1.193: Example of matrix k-effective by position index edit in HMTL output.

HTML output: Fission production by position index matrix

To obtain fission production by position information, the user must specify **MKP**=YES and **FMP**=YES in the parameter data. An example of the fission production matrix by position index for a $2 \times 2 \times 2$ array is shown in Fig. 8.1.194.



General Information
Messages
Input Data
Derived Data
Results
- Final Results
- Problem Characterization
- k_{eff} Results
- Fluxes
- Fissions and Absorptions
- Matrix by Position
- Matrix by Unit

KENO V.a - Matrix by Position
sample problem 2 2c8 bare with 8 unit types matrix calculation

Fission Production by Array Position Index Matrix

Source Index	Sink Index 0	Fraction Deviation	Sink Index 1	Fraction Deviation	Sink Index 2	Fraction Deviation	Sink Index 3	Fraction Deviation	Sink Index 4	Fraction Deviation	Sink Index 5	Fraction Deviation
0	0.000E+00	0.0000	0.000E+00	0.0000	0.000E+00	0.0000	0.000E+00	0.0000	0.000E+00	0.0000	0.000E+00	0.0000
1	0.000E+00	0.0000	7.142E-01	0.0122	5.158E-02	0.0307	5.493E-02	0.0284	2.973E-02	0.0362	7.210E-02	0.0236
2	0.000E+00	0.0000	5.279E-02	0.0265	7.237E-01	0.0106	2.970E-02	0.0393	5.272E-02	0.0303	2.357E-02	0.0391
3	0.000E+00	0.0000	5.442E-02	0.0278	2.914E-02	0.0387	7.194E-01	0.0123	5.437E-02	0.0275	2.498E-02	0.0392
4	0.000E+00	0.0000	2.956E-02	0.0353	5.509E-02	0.0286	5.867E-02	0.0285	7.116E-01	0.0130	2.077E-02	0.0422
5	0.000E+00	0.0000	6.813E-02	0.0242	2.549E-02	0.0350	2.482E-02	0.0415	1.998E-02	0.0433	7.266E-01	0.0111
6	0.000E+00	0.0000	2.612E-02	0.0384	6.754E-02	0.0275	1.893E-02	0.0455	2.246E-02	0.0386	5.517E-02	0.0275
7	0.000E+00	0.0000	2.523E-02	0.0392	1.804E-02	0.0459	7.167E-02	0.0252	2.612E-02	0.0385	5.510E-02	0.0268
8	0.000E+00	0.0000	1.983E-02	0.0458	2.516E-02	0.0420	2.315E-02	0.0390	6.969E-02	0.0263	2.849E-02	0.0388

Source Index	Sink Index 6	Fraction Deviation	Sink Index 7	Fraction Deviation	Sink Index 8	Fraction Deviation
0	0.000E+00	0.0000	0.000E+00	0.0000	0.000E+00	0.0000
1	2.488E-02	0.0402	2.394E-02	0.0369	2.024E-02	0.0448
2	6.675E-02	0.0245	1.938E-02	0.0483	2.435E-02	0.0389
3	2.155E-02	0.0433	7.180E-02	0.0248	2.557E-02	0.0441
4	2.567E-02	0.0370	2.661E-02	0.0412	6.995E-02	0.0275
5	5.334E-02	0.0273	5.236E-02	0.0275	2.835E-02	0.0381
6	7.175E-01	0.0115	2.848E-02	0.0391	5.328E-02	0.0261
7	3.021E-02	0.0363	7.119E-01	0.0140	5.258E-02	0.0293
8	5.355E-02	0.0298	5.181E-02	0.0262	7.164E-01	0.0139

Fig. 8.1.194: Sample fission production matrix by position index edit in HMTL output.

For each position index in the array, the number of next-generation neutrons produced at position index J per neutron born at position index I is determined. The fission production matrix by position index is used to determine the matrix k-effective, cofactor k-effective and source vector by position index.

HTML output: Source vector by position index

Source vector by position index information is printed if **MKP**=YES is specified in the parameter data. An example of the source vector by position index is shown in Fig. 8.1.195. The average self-multiplication by array position is the overall average of the self-multiplication of all units used in the problem.

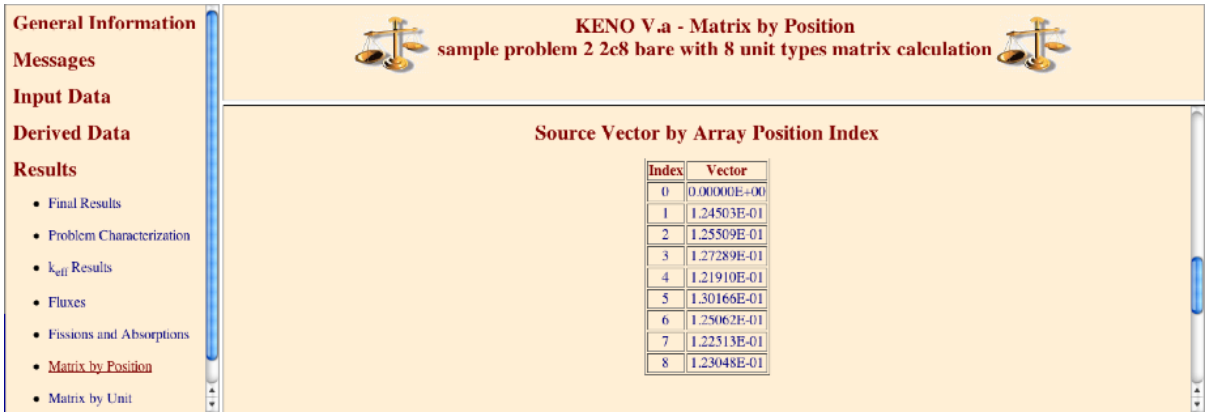


Fig. 8.1.195: Example of source vector by position index edit in HTML output.

HTML output: Cofactor k-effective by position index

The cofactor k-effective by position index edit is printed if **MKP**=YES is specified in the parameter data. An example of the cofactor k-effective by position index is shown in Fig. 8.1.196 and an example of the source vector by position index is shown in Fig. 8.1.197.

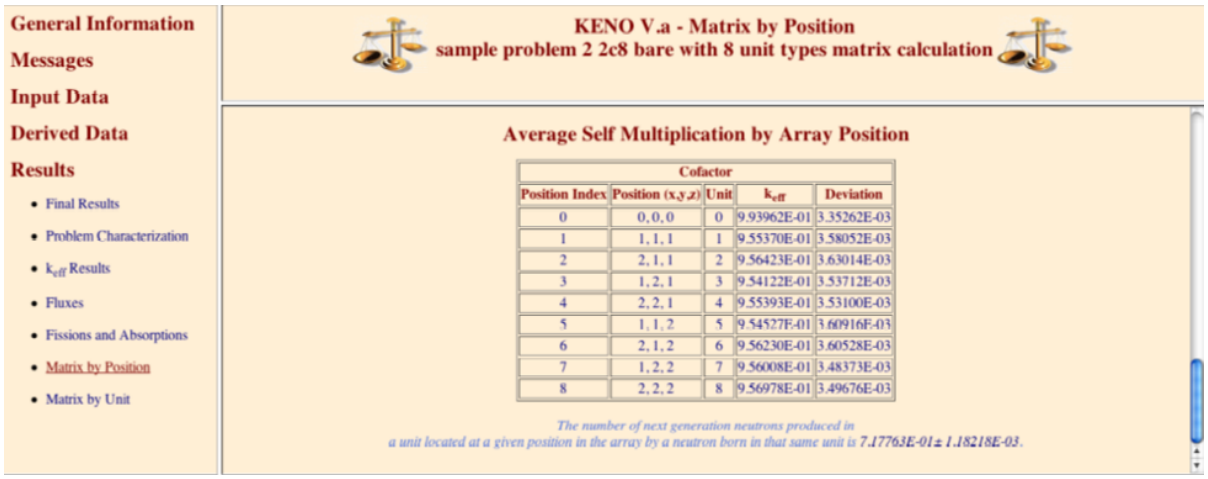


Fig. 8.1.196: Example cofactor k-effective by position index edit in HTML output.

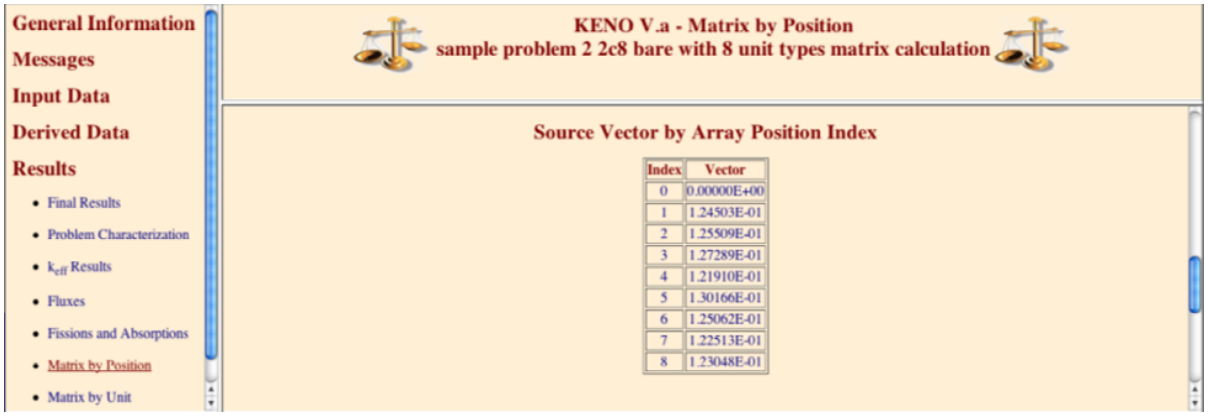


Fig. 8.1.197: Example of source vector by position index edit in HTML output.

HTML output: Matrix k-effective, fission production, source vector and cofactor k-effective by unit number

The matrix k-effective by **UNIT** number (unit k-effective) is the largest eigenvalue of the fission production by unit matrix. It is calculated only if **MKU**=YES is specified in the parameter data. An example of the matrix k-effective by **UNIT** is given in Fig. 8.1.198. Fission production, source vector and average self-multiplication by **UNIT** are all printed on the same page as the matrix k-effective by **UNIT**. The matrix k-effective HTML data by unit is available in the *Matrix by Unit* link in the *Results* section.

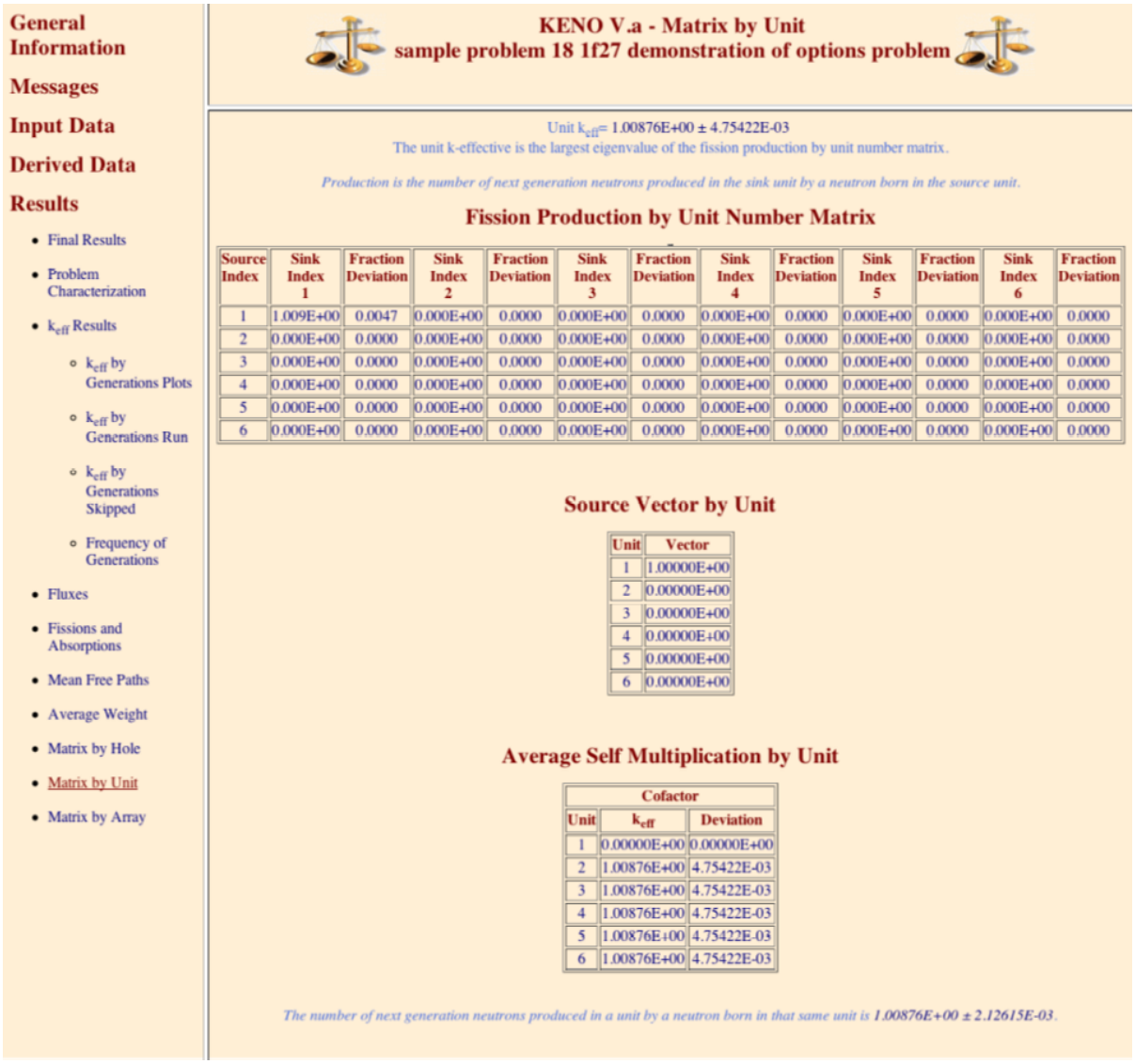


Fig. 8.1.198: Example matrix by unit edit in HTML output.

HTML output: Matrix k-effective, fission production, source vector and cofactor k-effective by hole number

The matrix k-effective by **HOLE** number is the largest eigenvalue of the fission production matrix collected by **HOLE** number and is calculated if **MKH**=YES was specified in the parameter data. An example of the matrix k-effective by **HOLE** number is given in Fig. 8.1.199. Fission production, source vector and average self-multiplication by **HOLE** are all printed on the same page as the matrix k-effective by **HOLE**. The matrix k-effective HTML data by **HOLE** is available in the *Matrix by Hole* link in the *Results* section.

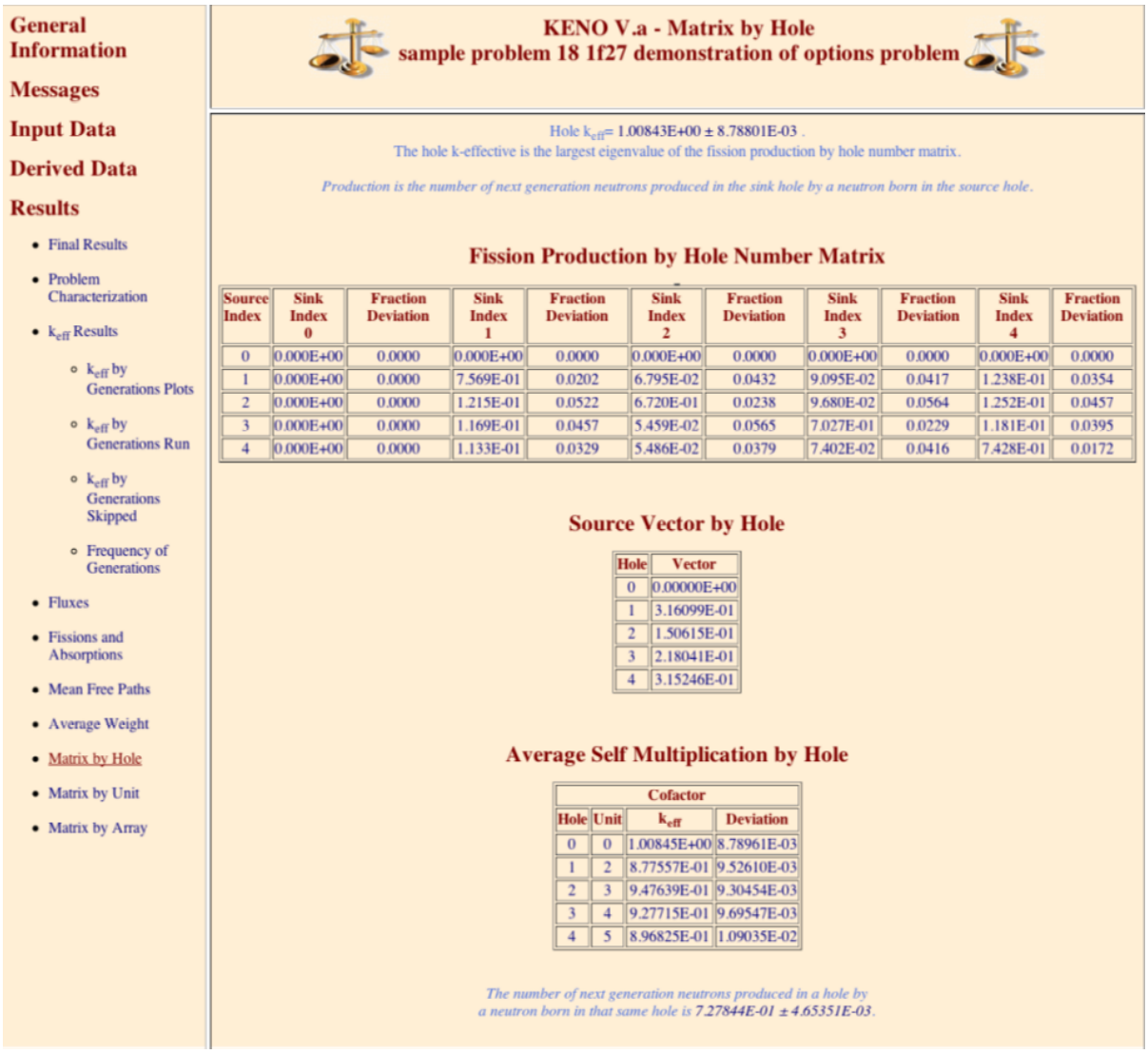


Fig. 8.1.199: Example matrix k-effective by hole edit in HTML output.

HTML output: Matrix k-effective, fission production, source vector and cofactor k-effective by array number

The matrix k-effective by array number is the largest eigenvalue of the fission production matrix collected by array number and is calculated if **MKA**=YES is entered in the parameter data. The number of next generation neutrons produced in array number J by a neutron born in array number I is given in this fission production matrix.

The matrix k-effective HTML data by array is available in the *Matrix by Array* link in the *Results* section, with an example shown in Fig. 8.1.200. Fission production, source vector and average self-multiplication by **ARRAY** number are all printed on the same page as the matrix k-effective by **ARRAY** number.

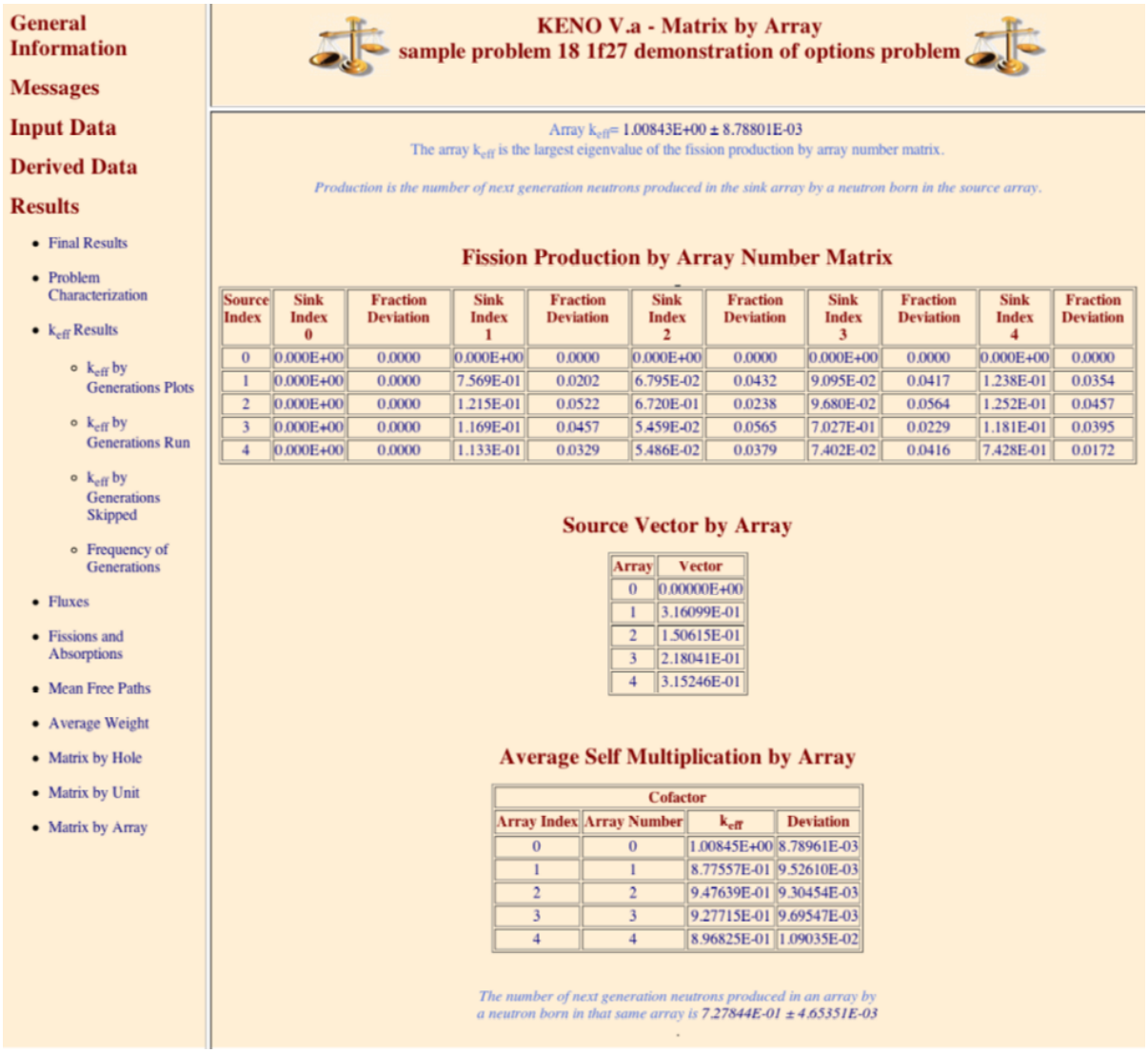
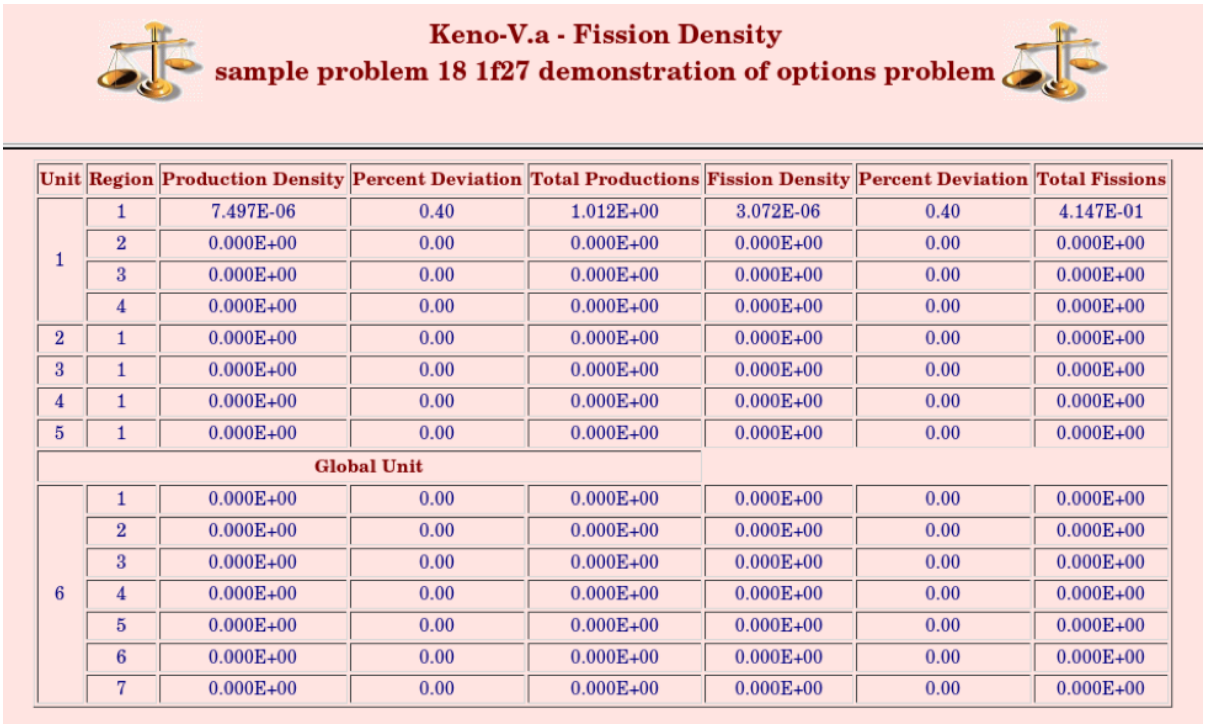


Fig. 8.1.200: Example matrix by array edit in HTML output.

HTML output: Fission density edit

The fission density edit is optional. In the HTML output, the fission density edit is available in the *Fission Density* link of *Fissions and Absorptions* under the *Results* section, as shown in Fig. 8.1.201.



Keno-V.a - Fission Density
sample problem 18 1f27 demonstration of options problem

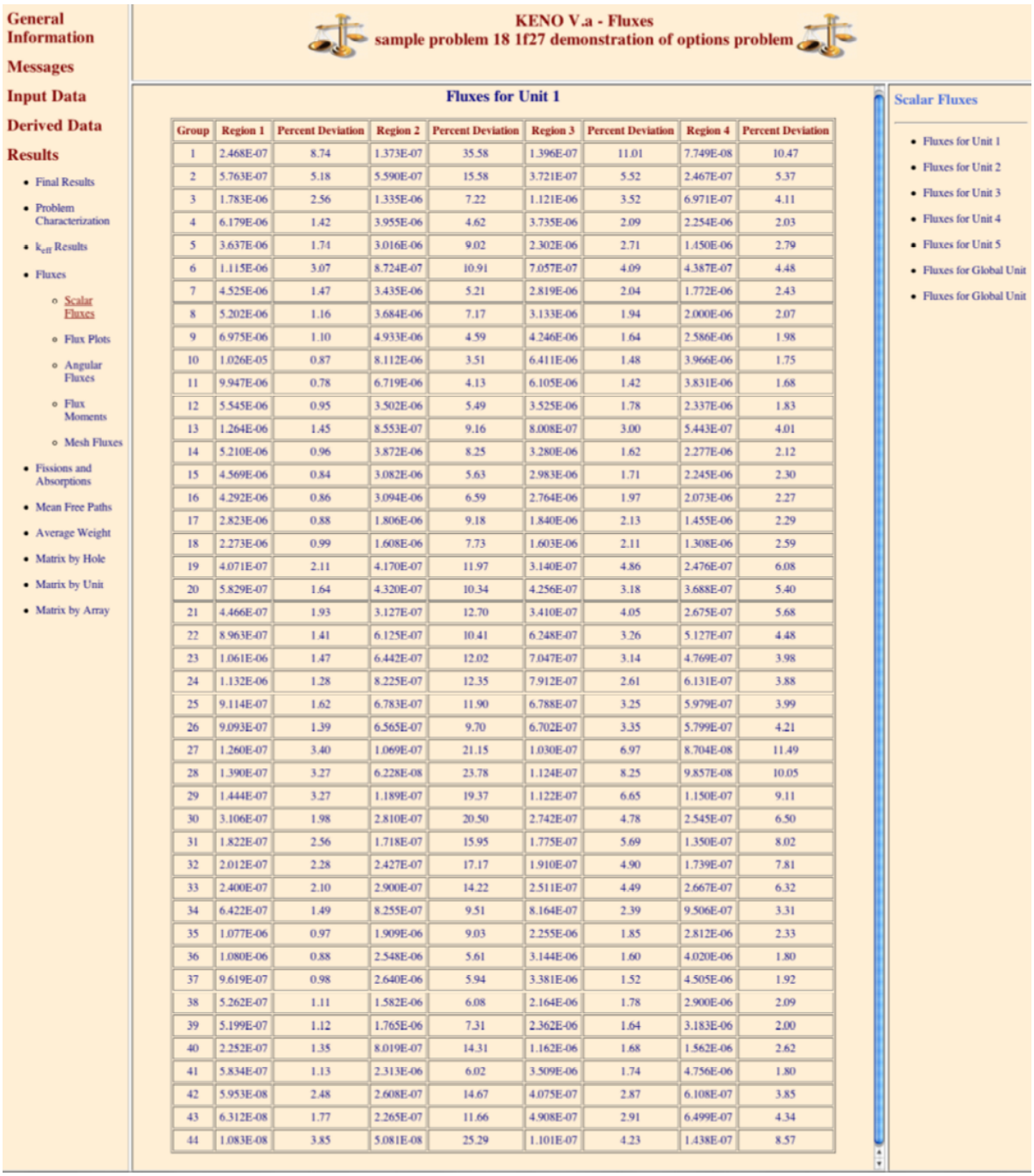
Unit	Region	Production Density	Percent Deviation	Total Productions	Fission Density	Percent Deviation	Total Fissions
1	1	7.497E-06	0.40	1.012E+00	3.072E-06	0.40	4.147E-01
	2	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
	3	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
	4	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
2	1	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
3	1	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
4	1	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
5	1	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
Global Unit							
6	1	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
	2	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
	3	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
	4	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
	5	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
	6	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00
	7	0.000E+00	0.00	0.000E+00	0.000E+00	0.00	0.000E+00

Fig. 8.1.201: Example fission density edit in HTML output.

HTML output: Flux edit

Printing the fluxes is optional; they are only printed if **FLX**=YES is specified in the parameter data. The fluxes are printed for each unit and each geometry region in the unit for every energy group.

In the HTML output, a table of flux data can be accessed with the *Scalar Fluxes* under *Fluxes* in the *Results* section, as shown in Fig. 8.1.202. The fluxes for each unit in the problem are accessed from links on the right side of the page. Separate links are presented for *Angular Fluxes* and *Flux Moments* if they are requested in the problem.



Group	Region 1	Percent Deviation	Region 2	Percent Deviation	Region 3	Percent Deviation	Region 4	Percent Deviation
1	2.468E-07	8.74	1.373E-07	35.58	1.396E-07	11.01	7.749E-08	10.47
2	5.763E-07	5.18	5.590E-07	15.58	3.721E-07	5.52	2.467E-07	5.37
3	1.783E-06	2.56	1.335E-06	7.22	1.121E-06	3.52	6.971E-07	4.11
4	6.179E-06	1.42	3.955E-06	4.62	3.735E-06	2.09	2.254E-06	2.03
5	3.637E-06	1.74	3.016E-06	9.02	2.302E-06	2.71	1.450E-06	2.79
6	1.115E-06	3.07	8.724E-07	10.91	7.057E-07	4.09	4.387E-07	4.48
7	4.525E-06	1.47	3.435E-06	5.21	2.819E-06	2.04	1.772E-06	2.43
8	5.202E-06	1.16	3.684E-06	7.17	3.133E-06	1.94	2.000E-06	2.07
9	6.975E-06	1.10	4.933E-06	4.59	4.246E-06	1.64	2.586E-06	1.98
10	1.026E-05	0.87	8.112E-06	3.51	6.411E-06	1.48	3.966E-06	1.75
11	9.947E-06	0.78	6.719E-06	4.13	6.105E-06	1.42	3.831E-06	1.68
12	5.545E-06	0.95	3.502E-06	5.49	3.525E-06	1.78	2.337E-06	1.83
13	1.264E-06	1.45	8.553E-07	9.16	8.008E-07	3.00	5.443E-07	4.01
14	5.210E-06	0.96	3.872E-06	8.25	3.280E-06	1.62	2.277E-06	2.12
15	4.569E-06	0.84	3.082E-06	5.63	2.983E-06	1.71	2.245E-06	2.30
16	4.292E-06	0.86	3.094E-06	6.59	2.764E-06	1.97	2.073E-06	2.27
17	2.823E-06	0.88	1.806E-06	9.18	1.840E-06	2.13	1.455E-06	2.29
18	2.273E-06	0.99	1.608E-06	7.73	1.603E-06	2.11	1.308E-06	2.59
19	4.071E-07	2.11	4.170E-07	11.97	3.140E-07	4.86	2.476E-07	6.08
20	5.829E-07	1.64	4.320E-07	10.34	4.256E-07	3.18	3.688E-07	5.40
21	4.466E-07	1.93	3.127E-07	12.70	3.410E-07	4.05	2.675E-07	5.68
22	8.963E-07	1.41	6.125E-07	10.41	6.248E-07	3.26	5.127E-07	4.48
23	1.061E-06	1.47	6.442E-07	12.02	7.047E-07	3.14	4.769E-07	3.98
24	1.132E-06	1.28	8.225E-07	12.35	7.912E-07	2.61	6.131E-07	3.88
25	9.114E-07	1.62	6.783E-07	11.90	6.788E-07	3.25	5.979E-07	3.99
26	9.093E-07	1.39	6.565E-07	9.70	6.702E-07	3.35	5.799E-07	4.21
27	1.260E-07	3.40	1.069E-07	21.15	1.030E-07	6.97	8.704E-08	11.49
28	1.390E-07	3.27	6.228E-08	23.78	1.124E-07	8.25	9.857E-08	10.05
29	1.444E-07	3.27	1.189E-07	19.37	1.122E-07	6.65	1.150E-07	9.11
30	3.106E-07	1.98	2.810E-07	20.50	2.742E-07	4.78	2.545E-07	6.50
31	1.822E-07	2.56	1.718E-07	15.95	1.775E-07	5.69	1.350E-07	8.02
32	2.012E-07	2.28	2.427E-07	17.17	1.910E-07	4.90	1.739E-07	7.81
33	2.400E-07	2.10	2.900E-07	14.22	2.511E-07	4.49	2.667E-07	6.32
34	6.422E-07	1.49	8.255E-07	9.51	8.164E-07	2.39	9.506E-07	3.31
35	1.077E-06	0.97	1.909E-06	9.03	2.255E-06	1.85	2.812E-06	2.33
36	1.080E-06	0.88	2.548E-06	5.61	3.144E-06	1.60	4.020E-06	1.80
37	9.619E-07	0.98	2.640E-06	5.94	3.381E-06	1.52	4.505E-06	1.92
38	5.262E-07	1.11	1.582E-06	6.08	2.164E-06	1.78	2.900E-06	2.09
39	5.199E-07	1.12	1.765E-06	7.31	2.362E-06	1.64	3.183E-06	2.00
40	2.252E-07	1.35	8.019E-07	14.31	1.162E-06	1.68	1.562E-06	2.62
41	5.834E-07	1.13	2.313E-06	6.02	3.509E-06	1.74	4.756E-06	1.80
42	5.953E-08	2.48	2.608E-07	14.67	4.075E-07	2.87	6.108E-07	3.85
43	6.312E-08	1.77	2.265E-07	11.66	4.908E-07	2.91	6.499E-07	4.34
44	1.083E-08	3.85	5.081E-08	25.29	1.101E-07	4.23	1.438E-07	8.57

Fig. 8.1.202: Example scalar flux edit in HTML output.

HTML output: Final results table

The final results table contains a summary of the most important physics parameters of the system and the number of error and warning messages generated during execution.

In the HTML output, the final results table is accessed with the *Final Results* link in the *Results* section, as shown in Fig. 8.1.203.

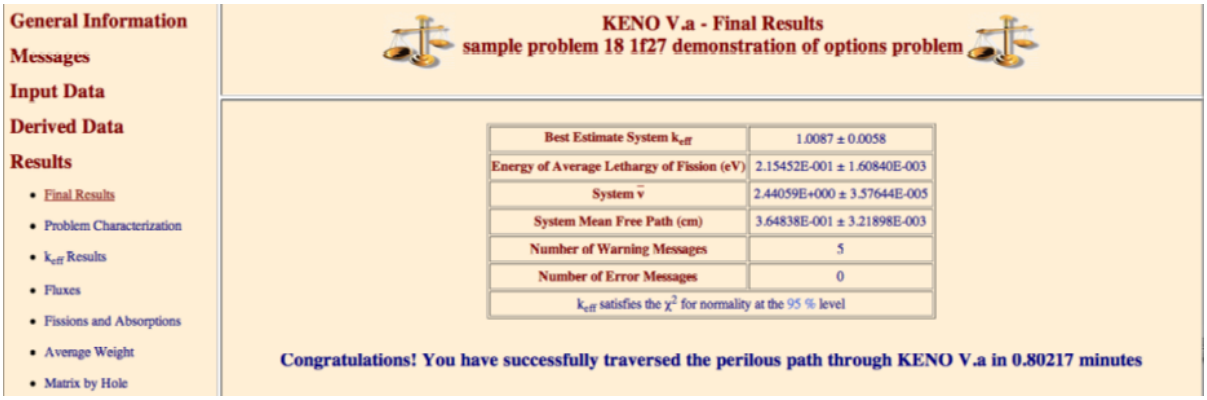


Fig. 8.1.203: An example of the final results table in HTML output.

8.1.6 WARNING MESSAGES AND ERROR MESSAGES

KENO prints warning and error messages that are identified by K5- or K6- for KENO V.a and KENO-VI, respectively, followed by a unique number (i.e., K5-001 is the identifier of the first message for KENO V.a). The generic symbol for messages originating in KENO V.a and KENO-VI is K?, where the question mark stands for 5 or 6. For additional information concerning the message, refer to the identifier number in this section.

Warning messages appear when a possible error is encountered. If the code alters data, that fact is stated in the message. It is the responsibility of the user to verify correct usage when a warning message is printed.

When an error is encountered, a global error flag (MFLAG) is set true and an error message is printed. The code immediately stops or execution is terminated with an error code if the error is too severe to continue. The warning and error messages in this section may show an underscore ____ or a numbered underscore (1) where data will be printed by the code. The explanation of the message will show an underscore or a numbered underscore to indicate the corresponding data.

8.1.6.1 Messages

KMSG001 *** WARNING *** K?-001 FOLLOWS: READ FLAG NOT FOUND. ASSUME PARAMETER DATA FOLLOWS.

> This message occurs in subroutine INITAL. It indicates that the word `READ` was not the first word of data encountered after the title card. If a parameter data block is to be entered, the code expects the words `READ PARAMETERS` to precede the parameter input data. If the word `READ` is not the first word, the code expects parameter input data immediately.

KMSG002 ** ERROR ** K?-002 FOLLOWS: THE NUMBER OF ENERGY GROUPS IS OUT OF RANGE FOR THE CROSS SECTION LIBRARY ON UNIT ___ THE INPUT MAY NOT SPECIFY A VALID DATA SET ON THIS UNIT OR THE MODULE THAT WAS TO CREATE THE CROSS SECTION LIBRARY ON THIS UNIT MAY HAVE FAILED

This message occurs in subroutine INITAL after subroutine PARAM has been executed. Unit number ____ should be checked to ensure that it was properly specified in the job control language. The data set name associated with this unit number should also be verified. This information is given in the printout in the third table. The module that generated the cross sections should be verified as having executed properly and that the data were saved or passed correctly. When this message is printed for an AMPX working format library, a STOP 108 is executed. When this message is printed for a mixed cross section format library, a STOP 109 is executed.

KMSG003 *** WARNING *** K?-003 FOLLOWS: MAT ID=___ WAS NOT FOUND IN THE WEIGHTS LIBRARY FOR THE SPECIFIED ENERGY GROUP STRUCTURE.

This message is printed in subroutine LODWTS when loading biasing factors from the direct access scratch file. It indicates that weights library did not contain biasing factors for this ID for the group structure that corresponds to the cross sections (multigroup), or to the binning structure (continuous energy). Default biasing factors will be substituted. The location of this set of biasing parameters should be checked to verify that the substitution does not cause a problem.

KMSG004 *** ERROR *** K?-004 FOLLOWS: INVALID INPUT PARAMETER NAME ___

This message comes from subroutine PARAM and indicates that the keyword for entering parameter data was misspelled or in error in some other way. A list of allowed keywords is given in Table 8.1.1 in the KENO input outline.

KMSG005 *** ERROR *** K?-005 FOLLOWS: AN ERROR WAS ENCOUNTERED IN THE ALPHANUMERIC PARAMETER DATA. THE DATA WAS ___

This message comes from subroutine PARAM and indicates that the keyword for the alphanumeric parameter data was entered correctly, but the data associated with it were not YES or NO as is required. The____ ____ in the error message could be something like `FLX`=YEX instead of `FLX`=YES.

KMSG006 ** WARNING ** K?-006 FOLLOWS: READ FLAG FOUND WHEN LOOKING FOR END FLAG. PARAMETER INPUT ASSUMED COMPLETE

This message occurs in subroutine PARAM. It indicates that the keywords `END PARAMETERS` were not found. The keywords `READ`____ were found instead. The code assumes the parameter data are complete and proceeds normally.

KMSG007 ** ERROR ** K?-007 FOLLOWS: ATTEMPT TO FIND END PARAMETER FLAG WAS UNSUCCESSFUL.

This message from subroutine PARAM occurs during the reading of the parameter data if the word `END` is found and it is not followed by the word `PARAMETERS`. A STOP 118 may be executed when this message is printed.

KMSG008 ** ERROR ** K?-008 FOLLOWS: AN END OF FILE WAS ENCOUNTERED WHILE ATTEMPTING TO READ PARAMETER DATA

This message is from subroutine PARAM. A STOP 118 may be executed when this message is printed.

KMSG009 ** WARNING ** K?-009 FOLLOWS: DUE TO INCONSISTENCIES BETWEEN INPUT AND RESTART DATA, FISSIONS AND ABSORPTIONS BY REGION WILL BE CALCULATED BUT NOT PRINTED. INPUT DATA SET FAR=NO, BUT DATA FROM THE RESTART UNIT SPECIFIED YES.

This message occurs in subroutine PARTBL and is mostly self-explanatory. The original problem (parent case) that wrote the restart data specified data inconsistent with the parameter data input to the restarted problem. The title of the parent case is given at the end of the parameter tables. The specification of the restart unit RST is given in the third table of the KENO output.

KMSG010 *** WARNING *** K?-010 FOLLOWS: DUE TO INCONSISTENCIES BETWEEN INPUT AND RESTART DATA, FLUXES WILL BE CALCULATED BUT NOT PRINTED. INPUT DATA SET FLX=NO, BUT DATA FROM THE RESTART UNIT SPECIFIED YES.

This message occurs in subroutine PARAM. The original problem (parent case) that wrote the restart data specified data did not agree with the parameter data input to the restarted problem. The title of the parent case is given at the end of the parameter tables. The specification of the restart unit RST is given in the third table of the KENO output.

KMSG011 ** ERROR ** K6-011 FOLLOWS: A BOUNDARY GEOMETRY WORD WAS NOT SPECIFIED FOR UNIT ___

This KENO-VI message is from subroutine SGGM_KENOG. A `BOUNDARY` card must be associated with each unit identifying the outermost region of the unit.

KMSG012 ** WARNING ** K?-012 FOLLOWS: INPUT PARAMETER NBK WAS ENTERED AS ___. IT WAS CHANGED TO ___. AT LEAST ___ POSITIONS ARE NECESSARY TO ACCOMMODATE THE NEUTRON BANK DATA.

This self-explanatory message is from subroutine PARTBL. `NBK` should not be entered as input data unless it is known that the default value is inadequate.

KMSG013 ** ERROR ** K6-013 FOLLOWS: MEDIA DATA MUST BE ENTERED FOR UNIT ___

This KENO-VI message is from subroutine SGGM_KENOG. Either another `UNIT` card or an `END GEOM` card was encountered prior to inputting `MEDIA` data for the current unit.

KMSG014 ** ERROR ** K?-014 FOLLOWS: ERROR - KEYWORD ___ ___ IS NOT A VALID MIXING TABLE KEYWORD.

This message is from subroutine MIXING_TABLE. It can only be encountered if a mixing table is expected (i.e., `READ MIX` or `READ MIXT` has been entered as data). At this point the only valid keywords are `MIX=`, `EPS=`, `SCT=` or `NCM=`. The keyword that was entered is printed in the message. See Sect. 8.1.3.10 for assistance in setting up the mixing table data.

KMSG015 ** ERROR ** K5-015 FOLLOWS: AN ERROR WAS FOUND IN THE HEMISPHERE DESIGNATION

This KENO V.a message was printed by subroutine K5GM_KENOG. It signifies that the geometry keyword has been destroyed after it was read. This indicates a code bug.

KMSG016 *** ERROR *** K?-016 FOLLOWS: ** ERROR ** A VALUE MUST BE ENTERED FOR LIB IN THE PARAMETER INPUT SO CROSS SECTIONS CAN BE MIXED.

This message is from subroutine DATAIN. It occurs when a mixing table has been read but the unit number for the AMPXS working library is undefined. This is corrected by entering LIB=____ in the parameter input data and making sure the desired AMPXS working library is properly defined as being on that unit in the job control language.

KMSG017 *** ERROR *** K?-017 FOLLOWS: UNRECOGNIZABLE GEOMETRY WORD ___

This message is from subroutines K5GM_KENOG, SGGM_KENOG, K5GM_CENTER or SGGM_CENTER. In the process of reading the geometry data, the word ____ was encountered when a geometry word was expected. Several of these messages may be generated. A message is generated for each word of data that is read, until a valid geometry word is found. The data are out of phase or the geometry word is misspelled. The previous geometry word should be checked to ensure that it has the correct entries. See Sect. 8.1.8.1 and Sect. 8.1.8.2 for lists of accepted geometry words.

KMSG018 ***** ERROR ***** K6-018 FOLLOWS: INVALID INDEX INTO KEYWRD ARRAY. INP=___ CHECK SUBROUTINE KENOG FOR A CODE ERROR.

This KENO-VI message is from subroutine SGGM_KENOG. This message usually means that a code error was introduced when changes were made to the program. Subroutine KENOG, function LCOMPR, and KEYWRD data block should be checked.

KMSG019 ***** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR **** K6-019 FOLLOWS: A GLOBAL UNIT HAS NOT BEEN SPECIFIED. **** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ******** ERROR ****

This KENO-VI message from subroutine SGGM_KENOG indicates that a GLOBAL UNIT has not been specified. The outermost unit in the problem must be specified as the GLOBAL UNIT. See Sect. 8.1.3.4 to determine the correct specification.

KMSG020 *** ERROR *** K6-020 FOLLOWS: GEOMETRY WORD INDEX OUT OF RANGE ON SCRATCH UNIT ___ FOR GEOMETRY WORD NUMBER ___ INP=___ GWRD=___

This KENO-VI message from subroutine SGGM_READGM means that INP is negative or larger than NP when the geometry data are read from the scratch unit, SKRT. This usually means that a code error was introduced when changes were made to the program. True geometry errors should be detected when the scratch unit, SKRT, is written. A STOP 125 is executed when this message is printed.

KMSG021 *** ERROR *** K?-021 FOLLOWS: ___ IS AN INVALID PARAMETER NAME FOR BIASING DATA. ID= OR WT= OR WTS= SHOULD HAVE BEEN ENTERED.

This message is from subroutine RDBIAS. See Sect. 8.1.3.7 for assistance in determining the proper procedure for entering biasing data.

KMSG022 *** ERROR *** K?-022 FOLLOWS: PROBLEM IN READGM, IGEO=___ IS AN INVALID GEOMETRY TYPE

If this message has not been preceded by other geometry error messages, it indicates a code bug.

KMSG023 *** ERROR *** K5-023 FOLLOWS: REFLECTION OR PERIODIC BOUNDARY CONDITIONS ARE ILLEGAL ON A CURVED BOUNDARY.

This self-explanatory KENO V.a message from K5GM_JOMCHK indicates that only a vacuum or a white boundary condition is allowed on a curvilinear surface.

KMSG024 K?-024 FOLLOWS: INCORRECT FLAG RETURNED FROM AREAD. IRET=___

This message from subroutine RDBIAS indicates that an error was encountered while reading the biasing data. The biasing data were not entered properly. See Sect. 8.1.3.7 for assistance.

KMSG025 *** ERROR *** K5-025 FOLLOWS: WHITE BOUNDARY CONDITION IS ALLOWED ONLY ON CUBOIDS, CYLINDERS, OR SPHERES

This self-explanatory KENO V.a message from K5GM_JOMCHK indicates that WHITE albedo boundary conditions cannot be applied to the outermost geometry if it is a hemisphere or a hemicylinder.

KMSG026 K?-026 - ARRAY DESCRIPTION ERROR MESSAGES K?-026 FOLLOWS: SET NUMBER ___ OF THE UNIT ORIENTATION DATA CONTAINS ___ ERROR(S).

This message from subroutine RDBOX is triggered when input errors are recognized in the unit orientation data. A set of unit orientation data consists of 10 numbers as shown in the companion message KMSG027. The number of errors printed in this message is a lower bound. More errors may actually exist. This message often means that a number was omitted or a blank was omitted when entering the unit orientation data.

KMSG027 K?-027 FOLLOWS: LTYPE=___ IX1=___ IX2=___ INCX=___ IY1=___ IY2=___ INCY=___ IZ1=___ IZ2=___ INCZ=___

This message is a companion to KMSG026. It indicates how the unit orientation data description for the set named in KMSG026 was entered. See Sect. 8.1.3.5.8 for information pertaining to unit orientation data.

KMSG028 K?-028 FOLLOWS: THE ABOVE UNIT ORIENTATION CARD(S) CONTAIN(S) AT LEAST ONE OF THE FOLLOWING ERRORS. 1. IX1, IY1, IZ1, INCX, INCY, OR INCZ IS LESS THAN OR EQUAL TO ZERO 2. IX2 IS LESS THAN IX1, IY2 IS LESS THAN IY1, OR IZ2 IS LESS THAN IZ1 3. IX2 IS GREATER THAN NBXMAX, IY2 IS LARGER THAN NBYMAX, OR IZ2 IS LARGER THAN NBZMAX 4. LTYPE IS LESS THAN 1 OR GREATER THAN NBOX

This self-explanatory message is from subroutine RDBOX. It pertains to the input orientation data for LOOP. See Sect. 8.1.3.5 for input instructions.

KMSG029 *** ERROR *** K?-029 FOLLOWS: THE ARRAY SIZE HAS BEEN SPECIFIED INCORRECTLY. NBXMAX=___ NBYMAX=___ NBZMAX=___ UNIT ORIENTATION DATA CANNOT BE READ UNLESS NBXMAX, NBYMAX AND NBZMAX ARE GREATER THAN ZERO.

This message from subroutine ARAYIN indicates that the array definition data were incorrectly specified. It occurs only if one or more of NBXMAX, NBYMAX, or NBZMAX is less than 1. In the array information data these are entered in the form `NUX=____ NUY=____ NUZ=____`. See Sect. 8.1.3.5. If a unit orientation data description is to be entered, NBXMAX, NBYMAX, and NBZMAX must all be greater than zero.

KMSG030 *** ERROR *** K?-030 FOLLOWS: END ___ FLAG WAS NOT FOUND. ___ ___ WAS READ INSTEAD.

This message from subroutine ARAYIN occurs if the unit orientation data description is terminated with the incorrect `END` flag.

KMSG031 *** ERROR *** K?-031 FOLLOWS: ___ IS AN INVALID PARAMETER NAME IN THE ARRAY DATA.

This message is written from subroutine ARAYIN if the array data block contains an incorrect keyword. The allowed keywords include NUX=, NUY=, NUZ=, FILL, LOOP, ARA``* =, ``TYP =. See Sect. 8.1.3.5 for additional assistance. A STOP 101 is executed when this message is printed.

KMSG032 `` ERROR `` K?-032 FOLLOWS: AN ERROR EXISTS IN UNIT ORIENTATION ARRAY NUMBER ___

This message from subroutines K5_SORTA or K6_SORTA is printed when an error is recognized in the array description. The type of error that will trigger the message is for a position in the unit orientation array to be undefined, zero, negative or greater than NBOX, the largest unit number. KMSG033 is a companion message.

KMSG033 *** ERROR *** K?-033 FOLLOWS: UNIT ___ IS INVALID AT X INDEX =___ Y INDEX =___ Z INDEX =___

This message comes from subroutines K5_SORTA or K6_SORTA. It is printed for each position in the unit orientation array that is in error. The message is printed a maximum of 10 times. Refer to Sect. 8.1.3.5 for assistance in correcting the error(s).

KMSG034 *** ERROR *** K?-034 FOLLOWS: THE NUMBER OF MIXTURES REQUESTED IN THE GEOMETRY IS ___ THE NUMBER OF MIXTURE CROSS SECTIONS IS ___

This message from subroutines K5_FLDATA, K6_FLDATA or MFLDATA occurs if the number of mixture cross sections from the restart unit, RSTRT, does not equal the number of mixtures requested in the geometry for a restarted problem.

KMSG035 *** ERROR *** K?-035 FOLLOWS: IN THE ALBEDO INPUT DATA ___ IS AN INVALID FACE CODE NAME.

This message is from subroutine RDREF. It occurs if an invalid face code name was entered in the albedo data. See Table 8.1.20 in Sect. 8.1.3.6 for a list of acceptable face code names.

KMSG036 *** ERROR *** K6-036 FOLLOWS: A BOUNDARY BODY ___ WAS SPECIFIED THAT IS NOT ONE OF THE BODIES IN THE GLOBAL UNIT BOUNDARY.

This message is from subroutines K6_RDBNDS or LOADGEOM1. It indicates that a boundary body was specified to assign boundary conditions that do not exist in the global unit. The bodies specified for boundary conditions should be carefully matched with those specified in the boundary definition vector of the global unit.

KMSG037 **** ERROR ***** K?-037 FOLLOWS: AVERAGE NU-BAR AND AVG. FISSION GROUP WAS SPECIFIED, BUT THE FISSION XSEC ID (18) WAS NOT FOUND IN THE EXTRA 1-D ARRAY (MT).

This message is from subroutine IDX1D. It indicates that the parameter data contained NUB= YES, but the corresponding necessary type of data was absent from the extra 1-D array. This can be due to a code error or an error concerning the extra 1-D data (X1D= in the parameter data).

KMSG038 **** ERROR *** K?-038 FOLLOWS: INPUT DATA INDICATED NO EXTRA 1-D XSEC IDS TO BE READ, BUT A READ FLAG WAS ENCOUNTERED.

This message from subroutine DATAIN is printed when the parameter data did not specify X1D= and the words READ X1DS were encountered later in the data. If extra 1-D data are to be used, X1D= must be entered in the parameter data and appropriate code modifications must be made to properly use the extra 1-D data.

KMSG039 *** ERROR *** K?-039 FOLLOWS: INVALID START PARAMETER NAME ___

This message is from subroutine RDSTRT. It indicates that an invalid start parameter name was encountered when the start data block was being read. A list of allowed start parameter names is contained in Sect. 8.1.3.8.

KMSG040 *** WARNING *** K?-040 FOLLOWS: LNU FOR START TYPE 6 WAS ENTERED AS ___. THE LARGEST VALUE NEEDED FOR LNU IS NPG; ENTRIES BEYOND NPG WILL BE IGNORED.

This self-explanatory message is from subroutine RDSTRT. See Sect. 8.1.3.8 and Sect. 8.1.4.8 for assistance in determining a valid value for LNU. NPG, the number of histories per generation, starting neutrons are needed to fill the initial neutron bank. KENO discards the starting points beyond NPG and notifies user with this message about this action.

KMSG041 *** ERROR *** K?-041 FOLLOWS: ALPHANUMERIC START DATA MUST BE ENTERED AS YES OR NO. THE DATA READ WAS ___ ___

This self-explanatory message is from subroutine RDSTRT. See Sect. 8.1.3.8 for assistance concerning start data.

KMSG042 *** WARNING *** K?-042 FOLLOWS: END ___ FLAG WAS NOT FOUND.___ ___ WAS READ INSTEAD

This message occurs when the READ ____ and END ____ do not match. When entering data blocks, each block must start with READ ____ and end with END ____.

KMSG043 *** ERROR *** K?-043 FOLLOWS: AN END OF FILE WAS ENCOUNTERED BEFORE AN END DATA WAS FOUND. THE PROBLEM WILL NOT RUN.

This message is from subroutine DATAIN. It occurs when an end of file is encountered while reading data.

KMSG044 *** ERROR *** K?-044 FOLLOWS: A PLOT OF ZERO HEIGHT AND/OR WIDTH HAS BEEN SPECIFIED. THE HEIGHT IS ___ AND THE WIDTH IS ___ THE DELTA X IS ___, THE DELTA Y IS ___, AND THE DELTA Z IS ___ THE DIRECTION COSINES DOWN THE PAGE ARE ___ ___ ___ THE DIRECTION COSINES ACROSS THE PAGE ARE ___ ___ ___

This message is from subroutine RDPLOT. At least one of the X, Y, or Z components that make up the upper left corner and lower right corner is inconsistent or one of the direction cosines UAX, VAX, WAX, UDN, VDN, or WDN have the wrong sign. Check the plot data for consistency.

KMSG045 *** ERROR *** K?-045 FOLLOWS: ILLEGAL DATA BLOCK IDENTIFIER, ___ ___

This message from subroutine DATAIN is printed whenever an invalid data block identifier is encountered. This can be caused by having the data out of order, by omitting data or by misspelling data. A block identifier consists of the words READ XXXX where XXXX is a keyword identifying the type of data to be read. Acceptable keywords when running KENO stand-alone are listed in Sect. 8.1.3.1.

KMSG046 *** ERROR *** K?-046 FOLLOWS: IPT=___ IS OUTSIDE THE ALLOWABLE LIMIT OF ___

This message is from subroutine DATAIN. It is indicative of a code error. IPT is the index into the LPOINT array, which contains the direct access pointers for the various types of data. LPOINT(1) is the pointer for the geometry region data. LPOINT(2) is the pointer for the array description (unit orientation) data. LPOINT(3) is the pointer for the mixing table data. LPOINT(4) is the pointer for volume data. LPOINT(5) is the pointer for the biasing or weighting data. LPOINT(6) is the pointer for the start data. LPOINT(7) is the pointer for the albedo data. LPOINT(8) is the pointer for the mixed cross section data. LPOINT(9) is the pointer for the energy and inverse velocity data. LPOINT(10) is the pointer for the plot data. LPOINT(11) is the pointer for the biasing input data. LPOINT(12) is the pointer for the grid data for fluxes. LPOINT(13) is the pointer for importance mapping data. LPOINT(14) is the pointer for reactions data. LPOINT(15) is the pointer for definitions data. LPOINT(16) is the pointer for system responses data. LPOINT(17) is the pointer for tallies data. LPOINT(18) is the pointer for the albedo-xsec energy correspondence data.

A STOP 152 is executed when this message is printed.

KMSG047 ***** ERROR ***** K6-047 FOLLOWS: SECTOR DATA WAS NOT FOUND FOR ARRAY ___

This KENO-VI message is from subroutine SGGM_ARRAY. This message usually means that sector data was left off the `ARRAY` contents record. See Sect. 8.1.3.4 for assistance in specifying an `ARRAY`.

KMSG048 *** ERROR *** K?-048 FOLLOWS: UNABLE TO ALLOCATE ___ DUE TO ERROR STATUS CODE ___ IN SUBROUTINE ___

This message is associated with allocating memory used to store problem dependent data. The message states the type of data memory was allocated for the error type returned when memory could not be allocated and the current subroutine. This message could be caused by insufficient system memory or a code problem.

KMSG050 *** ERROR *** K6-050 FOLLOWS: THE ___ POSITION OF ARRAY ___ IN UNIT ___ WAS SPECIFIED AS ___ WHICH IS LESS THAN 1 OR GREATER THAN ___, THE NUMBER OF UNITS IN THAT DIRECTION.

This self-explanatory KENO-VI message from subroutine SGGM_CORSIZ indicates that an array was not properly placed in a unit. This is usually due to the incorrect definition of one of the array indices in one dimension (X, Y, or Z) such that it was specified outside the array bounds in that dimension.

KMSG051 *** ERROR *** K?-051 FOLLOWS: SURFACE ___ OF BOUNDARY LABEL ___ REQUESTED A DIFFERENTIAL ALBEDO THAT IS NOT ON THE ALBEDO DATA SET. THE REQUESTED ALBEDO NAME IS ___

This message is from subroutine ALBRD. A list of the albedos that are on the standard albedo data set is given in Table 8.1.23 (Sect. 8.1.3.6).

KMSG052 *** ERROR *** K?-052 FOLLOWS: INTERPOLATION CODE SHOULD BE IN THE RANGE 1 TO 5 INT: ___

This message is from subroutine XSEC_INTERP. This indicates an error in the continuous energy point data. INT is the interpolation code from the data. A STOP 104 is executed in conjunction with this message, and a traceback may be printed from subroutine STOP.

KMSG053 *** ERROR *** K?-053 FOLLOWS: LOG INTERPOLATION SHOULD NOT BE USED WITH NEG OR ZERO ARGUMENT XP: ___

This message is from subroutine XSEC_INTERP and indicates that a LIN-LOG interpolation was requested with a non-positive value. A STOP 104 is executed in conjunction with this message, and a traceback may be printed from subroutine STOP.

KMSG054 *** ERROR *** K?-054 FOLLOWS: A SURFACE NUMBER ___ WAS SPECIFIED THAT IS LARGER THAN NUMBER OF SURFACES IN THE INDICATED GLOBAL BODY. THE PROBLEM WILL NOT BE RUN.

This message is from subroutines K5_RDBNDS or K6_RDBNDS. It indicates that an incorrect surface was specified to which a boundary condition was to be assigned. The surfaces of the bodies specified for boundary conditions should be carefully matched with those specified in the bounds of the global unit. Corresponding albedo surface numbers for all geometric shapes supported by KENO V.a and KENO-VI are listed in Table 8.1.21 and Table 8.1.22, respectively.

KMSG056 *** ERROR *** K?-056 FOLLOWS: MIXTURE ___ LACKS EITHER NU*FISSION OR CHI DATA

This message from subroutines NORM1D or XSEC1D indicates that a mixture that contains fissile material is missing the nu-fission cross section or the fission spectrum. A STOP 115 is executed when this message is printed.

KMSG057 *** ERROR *** K5-057 FOLLOWS: INVALID BIAS ID ___ IN REGION ___ ABOVE.

This KENO V.a message is from subroutines K5GM_PRTJOM or K5GM_READGM. It is printed if a negative or zero bias ID is entered for the specified geometry region. Review Sect. 8.1.3.4 for correct geometry data specification information.

KMSG058 *** ERROR *** K6-058 FOLLOWS: INVALID BIAS ID IN REGION ABOVE.

This KENO-VI message is from subroutine SGGM_PRTJOM. It is printed if a negative or zero bias ID is entered for the specified geometry region. Review Sect. 8.1.3.4 for correct geometry data specification information.

KMSG059 *** ERROR *** K?-059 FOLLOWS: THE CALCULATION WAS TERMINATED BECAUSE OF EXCESSIVE SPLITTING.

This message from subroutines K5_GUIDE or K6_GUIDE is printed only if message K?-128 is printed 50 or more times for a given generation. This indicates that the problem and/or the code is incapable of achieving a reasonable solution. If changes have been made in the code, they should be carefully scrutinized. If a biasing data block has been entered (Sect. 8.1.3.7), it should be checked carefully.

KMSG060 *** WARNING *** K?-060 FOLLOWS: THE ANGULAR SCATTERING DISTRIBUTION FOR MIXTURE ___ HAS BAD MOMENTS FOR THE TRANSFER FROM GROUP ___ TO GROUP ___ ___ MOMENTS WERE ACCEPTED

This message from subroutine BADMOM indicates that the moments from the cross section data are incorrect for the group transfer shown. The code replaces the incorrect moments with acceptable moments and proceeds normally. The moments printed in the last line of the message should match those printed in the eighth line. The user can suppress these messages by entering an appropriate value for the cross section message cutoff parameter, `EPS=` in the mixing table data, Sect. 8.1.3.10. See Sect. 8.1.4.4.4 for assistance in determining an appropriate value.

KMSG061 *** ERROR *** K6-061 FOLLOWS: ___ IS NOT A VALID UNIT NUMBER, A UNIT NUMBER MUST BE A POSITIVE INTEGER.

This KENO-VI message is from subroutine SGGM_KENOG. A unit number must be an integer greater than 0.

KMSG062 *** ERROR *** K6-062 FOLLOWS: INSUFFICIENT DATA FOLLOWED THE KEYWORD ARRAY.

This KENO-VI message is from subroutine SGGM_ARRAY. The data following the keyword ARRAY are misspelled or incomplete. See Sect. 8.1.3.5 for assistance in determining the correct method for inputting array data.

KMSG063 *** WARNING *** K?-063 FOLLOWS: NUCLIDE ___ ___ HAS NO GAMMA PRODUCTION DATA. *** WARNING ***

This message from subroutine MIXMIX indicates that somehow KENO has been told to run a coupled neutron-gamma problem, and the indicated nuclide has no neutron to gamma transfer data. This means this nuclide cannot produce any gammas. This message should never occur in KENO, as it should have ignored all gamma data from a coupled library.

KMSG064 *** WARNING *** K?-064 FOLLOWS: THE ANGULAR SCATTERING DISTRIBUTION FOR MIXTURE ___ HAS BAD MOMENTS FOR THE TRANSFER FROM GROUP ___ TO GROUP ___ ___ MOMENTS WERE ACCEPTED. THE P0 COEFFICIENT IS ___

This message from subroutine BADMOM is printed to inform the user that the cross sections were altered by the code because the moments from the cross section data were incorrect for the group transfer shown. The P_0 coefficient was larger than the cross section message cutoff parameter `EPS`, but the relative change in the moments was smaller than `EPS`.

KMSG065 *** WARNING *** K?-065 FOLLOWS: AN AMPX WORKING LIBRARY WAS SPECIFIED ON UNIT ___ BUT NO MIXING DATA WAS READ. MIXED CROSS SECTIONS FROM UNIT ___ WILL BE USED.

This message from subroutine ICEMIX occurs if the parameter data specified `LIB=____` but no cross section mixing data block was entered. The cross section mixing data block begins with `READ MIXT`. See Sect. 8.1.3.3 for parameter data and Sect. 8.1.3.10 for mixing table information.

KMSG066 *** WARNING *** K?-066 FOLLOWS: ___ MIXTURES WERE REQUESTED IN THE GEOMETRY, BUT ONLY ___ MIXTURES ARE ON THE MIXED CROSS SECTION LIBRARY.

This message from subroutine ICEMIX indicates that more mixtures were requested in the geometry region data than were available on the mixed cross section library. See Sect. 8.1.3.3 for the specification of the unit number of the mixed cross section library (**XSC**=), Sect. 8.1.3.10 for the specification of the mixing table, and Sect. 8.1.3.4 to determine the mixtures used in the geometry region data.

KMSG067 *** ERROR *** K?-067 FOLLOWS: THE ADJOINT INPUT PARAMETER WAS _ BUT THE ADJOINT INDICATOR FROM THE MIXTURE CROSS SECTION LIBRARY WAS _. KENO WILL NOT EXECUTE.

This message from subroutine ICEMIX occurs if the adjoint input parameter `ADJ=` specified a forward calculation and the cross sections were adjointed, or the adjoint input parameter specified an adjoint calculation and the cross sections were not adjointed. T indicates true, F indicates false.

KMSG069 ___ IS AN INVALID GEOMETRY TYPE INDEX

This KENO V.a message from K5GM_HOLEXT is a companion message to KMSG091, and will follow it.

KMSG070 *** ERROR *** K?-070 FOLLOWS: ERROR IN SUBROUTINE WRTRST. NDX=___

This message from subroutines K5_WRTRST or K6_WRTRST occurs only if the type of data to be written on the restart (`WSTRT`) unit is undefined (i.e., `NDX` is greater than 15). `NDX` is the index in the LPOINT array as described in messages KMSG046 and KMSG058. This error is usually caused by code errors that were introduced when changes were made to the code. A STOP 133 is executed when this message is printed.

KMSG071 THIS INTERSECTION MAY BE DUE TO ROUNDOFF IN CALCULATING THE ARRAY BOUNDARY DIMENSIONS. FOR ROUNDOFF ERRORS, INCREASE THE DIMENSIONS OF ALL SUBSEQUENT REGIONS SHARING THAT BOUNDARY. +X -X +Y -Y +Z -Z ARRAY ___ ___ ___ ___ ___ ___ ___

This KENO V.a message from K5GM_JOMCHK is a companion message to KMSG092 and will follow it.

KMSG072 CUBOID ___ ___ ___ ___ ___ ___

This KENO V.a message from K5GM_JOMCHK is a companion message to KMSG092 and will follow it.

KMSG073 *** ERROR *** K?-073 FOLLOWS: ___ MIXTURES WERE REQUESTED IN THE GEOMETRY DATA, BUT ONLY ___ OF THESE WERE FOUND IN THE MIXED CROSS SECTIONS.

This message from subroutines K5_MASTER or K6_MASTER indicates a mixture number used in the KENO model was not included in the cross sections defined in the READ MIXT block.

KMSG074 *** ERROR *** K6-074 FOLLOWS: INVALID GEOMETRY TYPE IN VOLCUB. IGEO =___

This KENO-VI message from subroutine SGGM_VOLCUB is probably the result of a code error.

KMSG075 *** ERROR *** THE PROBLEM WILL NOT RUN

This KENO V.a message from subroutines K5GM_HOLE or K5GM_HOLCHK is a companion message to KMSG188.

KMSG076 *** ERROR *** K5-076 FOLLOWS: POSIT ERROR — ILLEGAL GEOMETRY TYPE X =___ Y =___ Z =___ K1 =___ K2 =___ K =___ IGEO =___

This KENO V.a message from subroutine K5GM_POSIT indicates that the history at X,Y,Z in region K in the unit with regions K1 to K2 has a geometry type IGEO which is not valid geometry type. This is caused by a code error.

KMSG077 *** INFORMATION WARNING *** K5-077 FOLLOWS: POSIT WARNING — POINT NOT IN GEOMETRY X =___ Y =___ Z =___ K1 =___ K2 =___

This KENO V.a message from subroutine K5GM_POSIT indicates that point X,Y,Z is not within any region between K1 and K2. This is usually caused from specifying start data that tries to start histories outside the geometry.

KMSG078 *** ERROR *** K5-078 FOLLOWS: INVALID GEOMETRY ENCOUNTERED FOR THE LAST GEOMETRY REGION. IGEO=___

This KENO V.a message from subroutine K5GM_CORSIZ says the geometry type of the last geometry region of the global unit is invalid. This indicates a code bug.

KMSG079 *** ERROR *** K6-079 FOLLOWS: UNABLE TO DETERMINE IF QUADRATIC IS CONVEX OR CONCAVE. CHECK FUNCTION VEXCAV CALLED FROM QUAD.

This self-explanatory KENO-VI message is from function SGGM_VEXCAV. It indicates a problem with an arbitrary quadratic. KENO-VI is missing the data indicating whether a surface is convex or concave and so was unable to determine this for the current arbitrary quadratic.

KMSG080 *** ERROR *** K5-080 FOLLOWS: UNRECOGNIZED GEOMETRY TYPE. IGEOM =___ IN SUBROUTINE PRTJOM

This KENO V.a message from subroutine K5GM_PRTJOM indicates a code bug. This error should be reported to scalehelp@ornl.gov.

KMSG081 *** ERROR *** K5-081 FOLLOWS: ERROR IN HEMISPHERE DESIGNATION. ISET =___ IN SUBROUTINE PRTJOM

This KENO V.a message from subroutine K5GM_PRTJOM indicates a code bug. This error should be reported to scalehelp@ornl.gov.

KMSG082 *** ERROR *** K5-082 FOLLOWS: ERROR IN PRTJOM WITH HEMICYLINDER DESIGNATION. NHCYL =___

This KENO V.a message from subroutine K5GM_PRTJOM indicates a code bug. This error should be reported to scalehelp@ornl.gov.

KMSG083 *** ERROR *** K5-083 FOLLOWS: FOR ARRAY ___, THE _ DIMENSIONS OF UNIT ___ AT (___ ___ ___) DO NOT MATCH THOSE OF UNIT ___ AT (___ ___ ___) FOR UNIT ___ +_ =___ AND -_ =___ WHILE FOR UNIT ___ +_ =___ AND -_ =___

This KENO V.a message is from subroutine K5GM_ARASIZ. The common faces of adjacent units must be the same size and shape. This message occurs whenever this requirement is not met. One or more of the dimensions of the units specified in the message may be incorrect, or the array definition data may be incorrect. Carefully check the input data relating to the geometry region data and the array definition data as described in Sect. 8.1.3.4 and Sect. 8.1.3.5.

KMSG084 *** ERROR *** K?-084 FOLLOWS: UNIT ___, IN UNIT ORIENTATION ARRAY NUMBER ___ IS UNDEFINED IN THE INPUT DATA.

This message from subroutines K5GM_ARASIZ or SGGM_ARASIZ occurs when the array description data block specifies a unit that was not defined in the geometry region data. Verify the array definition data and the geometry region data as described in Sect. 8.1.3.4 and Sect. 8.1.3.5.

KMSG085 *** ERROR *** K?-085 FOLLOWS: UNIT ___ AT POSITION X=___ Y=___ Z=___ IN UNIT ORIENTATION ARRAY NUMBER ___ IS INVALID BECAUSE IT IS LESS THAN ZERO OR LARGER THAN THE LARGEST UNIT NUMBER IN THE INPUT DATA.

This message from subroutines K5GM_ARASIZ or SGGM_ARASIZ occurs if the unit number named in the message is less than or equal to zero or greater than NBOX (the number of different units). The position of the offending unit is also given. This error usually results from leaving some positions undefined in the unit orientation array or from erroneous data in the unit orientation data. (This includes extra data, mistyped data and omitted data.) See Sect. 8.1.3.5 for additional information.

KMSG086 *** ERROR *** K6-086 FOLLOWS: GEOMETRY WORD INDEX OUT OF RANGE. IGEOM =___ IN SUBROUTINE PRTJOM

This KENO-VI message from subroutine K6_PRTJOM is self-explanatory. The printed value of IGEOM must be greater than zero and less than 24 in KENO V.a or less than 27 in KENO-VI to be valid. If the geometry words (see Sect. 8.1.3.4) are correct, this message is due to a code error that has been introduced when changes were made to the code.

KMSG087 *** ERROR *** K6-087 FOLLOWS: INSUFFICIENT DATA FOLLOWED THE KEYWORD MEDIA.

This KENO-VI message from subroutine K6_MEDIA. Either the material, bias ID, or sector data were not included on the media card, or non-integer data were inadvertently entered. See Sect. 8.1.3.4 for additional information on the type of data required on a MEDIA card.

KMSG088 *** ERROR *** K6-088 FOLLOWS: SECTOR DATA WAS NOT FOUND FOR MEDIA ___

This KENO-VI message from subroutine K6_MEDIA. Either the material, bias ID, or sector data were not included on media card _____ or non-integer data were inadvertently entered. See Sect. 8.1.3.4 for additional information on the type of data required on a MEDIA card.

KMSG089 *** ERROR *** K6-089 FOLLOWS: VOLUME HAS BEEN MULTIPLY DEFINED. GENERATION = ___ NEUTRON = ___ UNIT =___ PARTICLE AT POSITION X=___ Y=___ Z=___ IS LOCATED INSIDE THE FOLLOWING REGIONS IN THE UNIT:

This KENO-VI message from subroutine K6_PRTPOS out of either subroutine SGGM_TRACE or K6_TRACK. It is caused by incorrectly specifying the region definition vector in the media cards of a unit. It may be caused by a volume defined as part of more than one media record or incorrectly specifying regions that share a common boundary so the boundary between regions is contained in more than one region. See Sect. 8.1.3.4 for more information concerning vector definition arrays on MEDIA records.

KMSG090 *** ERROR *** K?-090 FOLLOWS: THE PROBLEM HAS TERMINATED BECAUSE THE PARTICLE HAS ENTERED AN INFINITE LOOP. THIS IS POSSIBLY DUE TO A PROBLEM WITH THE CODE. UNIT ___ ARRAY ___ HOLE ___

This self-explanatory message indicates that a history has become totally lost, and this message prevents the code from wasting time in an infinite loop. This message should never occur. This error should be reported to scalehelp@ornl.gov.

KMSG091 *** ERROR *** K5-091 FOLLOWS: UNIT ___ CONTAINS THE FOLLOWING GEOMETRY INCONSISTENCIES:

This message from subroutines K5GM_JOMCHK, K5GM_HOLEXT or K5GM_HOLHOL indicates that one or more intersecting geometry regions were encountered in unit____. KMSG092 is the companion message from subroutine JOMCHK and specifies the regions that intersect. KMSG069 is the companion message from subroutine HOLEXT. KMSG163 and KMSG179 are the companion messages from subroutine HOLHOL. The geometry region data should be corrected and the problem resubmitted. Sect. 8.1.3.4 may provide assistance in correctly specifying the data.

KMSG092 *** ERROR *** K5-092 FOLLOWS: REGION NUMBER ___ INTERSECTS REGION NUMBER ___.

This KENO V.a message from subroutine K5GM_JOMCHK is a companion message to K5-91 and K5-94. It specifies the intersecting regions. The user must determine which region is incorrectly specified or if the data are out of order. KENO V.a REQUIRES THAT EACH SUCCESSIVE GEOMETRY REGION MUST COMPLETELY ENCLOSE THE PREVIOUS REGION. THIS DOES ALLOW COMMON FACES AND TANGENCY.

KMSG093 *** ERROR *** K5-093 FOLLOWS: INVALID GEOMETRY TYPE. IGEO=___

This KENO V.a message is from subroutine K5GM_JOMCHK. IGEO must be greater than zero and less than 20. If it does not fall in this range, a code error is the probable cause. K5GM_KENOG or K5GM_READGM are the subroutines likely to have the code error. Verify that all the geometry words (`fgeom`, Sect. 8.1.3.4) are correct.

KMSG094 *** ERROR *** K5-094 FOLLOWS: THE REFLECTOR DIMENSIONS ARE INCONSISTENT.

This KENO V.a message is from subroutine K5GM_JOMCHK. It is a companion to message K?-095 and is printed whenever one or more intersecting geometry regions are encountered in the external reflector.

KMSG095 *** ERROR *** K5-095 FOLLOWS: REGION NUMBER ___ IN UNIT NUMBER ___ CONTAINS AN ERROR IN THE DIMENSIONS.

This KENO V.a message from subroutine K5GM_VOLUME indicates an error in the geometry input data such that the negative dimension specification for a cube or cuboid is larger than the positive dimension specification (i.e., the -X dimension is greater than the +X dimension, or the -Y dimension is greater than the +Y dimension or the -Z dimension is greater than the +Z dimension). This message is also printed if the magnitude of the chord for a hemicylinder or hemisphere is larger than the radius. See Sect. 8.1.3.4 for assistance in specifying the geometry correctly.

KMSG096 *** ERROR *** K5-096 FOLLOWS: THE VOLUME DEFINED BY REGION ___ IN UNIT ___ IS NEGATIVE.

This KENO V.a message from subroutine K5GM_VOLUME is printed whenever a negative volume is calculated. This can be caused by the positive dimension being smaller than the negative dimension on a face of a geometry region. It can also be caused by having intersecting regions, or it may be the result of round-off when the volumes are calculated. Either the geometry regions are incorrectly specified, the data are out of order, or the dimensions are so tight fitting that round-off causes the net volume of the region to be

negative. If the error is caused by round-off, adjust the appropriate dimensions slightly. See Sect. 8.1.3.4.

KMSG097 *** ERROR *** K5-097 FOLLOWS: THE VOLUME FOR UNIT ___ IS NEGATIVE.

This KENO V.a message is from subroutine K5GM_VOLUME. A negative volume for a unit can be caused by having intersecting regions within the unit, or by having a unit consisting of one region and having a positive dimension smaller than the negative dimension on one or more faces. Message K5-95 or K5-96 may accompany this message. See Sect. 8.1.3.4 for assistance in specifying the geometry data correctly.

KMSG098 *** ERROR *** K6-098 FOLLOWS: PROBLEM IN UNIT ___ THE ARRAY TYPE SPECIFIED IN THIS UNIT IS UNDEFINED. SPECIFY IF THE ARRAY TYPE IS CUBOID, HEXPRISM, OR DODECAHEDRON.

This KENO-VI message is from subroutine K6_ARASIZ. The array type specified in this unit in an unknown type. See Sect. 8.1.3.5 for how to enter array definitions.

KMSG099 *** ERROR *** K?-099 FOLLOWS: PROBLEM WITH UNIT ___ THE OUTERMOST GEOMETRY REGION OF A UNIT UTILIZED IN THE UNIT ORIENTATION DESCRIPTION OF THE ARRAY DATA MUST BE A ___

This message is from subroutines K5_ARASIZ or K6_ARASIZ. It can occur when the boundary region of a unit in an array is not consistent with the array type. The unit specified should be corrected so that the outer boundary is of the type required, or it should be replaced in the unit orientation data with a unit that has the correct outer boundary.

KMSG100 *** ERROR *** K?-100 FOLLOWS: THIS PROBLEM WILL NOT BE RUN BECAUSE ERRORS WERE ENCOUNTERED IN THE INPUT DATA.

This message is from subroutines K5_MASTER or K6_MASTER, and indicates that other error messages were printed in the problem output. These messages should be located and the data corrected accordingly. Execution is terminated with error code 129.

KMSG101 *** ERROR *** K?-101 FOLLOWS: NO FISSILE MATERIAL WAS FOUND IN SUBROUTINE START

This message is from subroutine START. It indicates that none of the mixtures used in this problem has an associated fission spectrum. Either the geometry data did not specify a fissionable mixture number, the mixing table is incorrect, the wrong mixed cross section data set was mounted, or the mixed cross section data set was incorrectly or incompletely made.

KMSG102 *** ERROR *** K?-102 FOLLOWS: THE START DATA SPECIFIES THAT NEUTRONS CAN BE STARTED IN THE REFLECTOR. HOWEVER, NEUTRONS WILL NOT BE STARTED BECAUSE THE OUTER REGION OF THE REFLECTOR IS NOT A CUBE OR CUBOID. NEUTRONS CAN BE STARTED FOR THE EXISTING GEOMETRY IF XSM,XSP,YSM,YSP,ZSM AND ZSP ARE ENTERED AS START DATA. XSM,XSP,YSM,YSP,ZSM AND ZSP MUST FIT WITHIN THE OUTER REGION OF THE REFLECTOR.

This message is from subroutine START. Start type 0 allows starting points throughout noncuboidal regions. If a start type other than 0 or 6 is desired and the outermost region of the reflector is not a cube or cuboid, then data must be entered to specify an imaginary cube or cuboid within this outer region. See Sect. 8.1.3.8 for assistance in specifying this data.

KMSG103 *** ERROR *** K?-103 FOLLOWS: START TYPE ___ IS OUT OF RANGE.

This message from subroutines K5_START or K6_START indicates that the start type was less than zero or greater than 9. The start type is defined by entering the keyword NST= followed by the desired start type in the start data. The available starting options are given in Table 8.1.26, Sect. 8.1.3.8.

KMSG104 *** ERROR *** K?-104 FOLLOWS: A POSIT ERROR INDICATES THAT THE POINT X=___ Y=___ Z=___ DOES NOT OCCUR WITHIN UNIT ___ IF XSM, XSP, YSM, YSP, ZSM, ZSP WERE ENTERED IN THE START DATA, VERIFY THAT THEY FIT WITHIN THE OVERALL COORDINATES OF THE SYSTEM. THE OVERALL COORDINATES MAY NOT BE PRINTED FOR A BARE ARRAY.

If XSM, XSP, YSM, YSP, ZSM, ZSP were entered in the start data, they should fit within the overall coordinates of the system. The overall coordinates may not be printed for a bare array. This message from subroutines K5_START or K6_START may result from precision difficulties. It is allowed to occur a maximum of five times before being considered fatal. A code error may be the cause of this message if it becomes fatal.

KMSG105 K?-105 *** WARNING, ONLY ___ INDEPENDENT STARTING POSITIONS WERE GENERATED. ***

This message is from subroutines K5_START or K6_START. KENO must have *npb* (NPG=, see parameter data, Sect. 8.1.3.3) starting positions. This message is to inform the user that fewer than *npb* starting positions were generated. The remaining starting positions are randomly selected from those that were generated, thus giving duplicate starting positions. If the number of independent starting positions is nearly *npb*, the starting distribution is probably acceptable. If it is much smaller than *npb*, a different start type should be used to give a better starting distribution (see Sect. 8.1.3.8). The amount of time allowed to generate the starting positions is controlled by parameter TBA= (see Sect. 8.1.3.3). If the start data are appropriate, it may be necessary to increase the value of TBA to ensure generating *npb* starting positions.

KMSG106 *** ERROR *** K6-106 FOLLOWS: POSIT ERROR — UNIT ___ HAS MULTIPLY DEFINED SPACE AT X = ___ Y = ___ Z = ___ SECTOR INSIDE

This KENO-VI message from subroutine SGGM_POSIT is usually the result of an incorrectly specified starting point for the initial source distribution when NST=3, 4 or 6 is specified in the start data. The starting point may not be consistent with the unit's position in the global array. The message can also be caused by improperly specifying media records. MEDIA records in the specified units should be checked for multiply defined volumes. X, Y, Z is the location of the neutron, and SECTOR INSIDE is the sectors which contain that point. See Sect. 8.1.3.4 for correct geometry words.

KMSG107 *** ERROR *** K7-107 FOLLOWS: POSIT ERROR — UNIT ___ HAS AN UNDEFINED SPACE AT X = ___ Y = ___ Z = ___ SECTOR INSIDE

This KENO-VI message from subroutine SGGM_POSIT may result from not specifying all volumes in the unit using media records. It could also occur from start data if a particle is attempting to start outside the global unit or in an undefined space. Geometry should be checked for the specified unit.

KMSG108 *** ERROR *** K?-108 FOLLOWS: POSITION (___,___,___) IS NOT VALID FOR THE POSITION OF THE SPIKE FOR START TYPE 2.

This message from subroutine START2 indicates that NXS, NYS or NZS was entered as zero. See Sect. 8.1.3.8 for the correct start data specification.

KMSG109 *** ERROR *** K?-109 FOLLOWS: INVALID GEOMETRY TYPE IN START. IGEO =___

This message from subroutines K5_STRTSU or K6_STRTSU is likely the result of a code error. This error should be reported to scalehelp@ornl.gov.

KMSG110 *** ERROR *** K?-110 FOLLOWS: THE PROBLEM WILL NOT BE EXECUTED BECAUSE NO FISSILE MATERIAL WAS FOUND.

This message from subroutines K5_VOLFIS or K6_VOLFIS occurs when the volume of fissile material is found to be zero. It should be verified that the fissile material was correctly specified in the geometry data, and the volume of the fissile material in the printout should be nonzero. The mixing table should be verified as being correct or that the correct Monte Carlo mixed cross section data set has been used if a mixing table is not used. If the fissile region is small compared to the rest of the problem and the region volumes are calculated using TRACE or RANDOM types with default number of traces or particles, the fissile region volumes may be incorrect (even zero). If the volumes calculate to be zero, this message may be issued. In this case, the volume calculation (see Sect. 8.1.3.13) should be corrected. A STOP 131 is executed when this message is printed.

KMSG111 ******* WARNING ******* K?-111 FOLLOWS: RESTART DATA IS NOT AVAILABLE FOR RESTARTING WITH GENERATION ___ AS SPECIFIED IN THE INPUT DATA. **** **** HOWEVER, AVAILABLE RESTART DATA HAS ALLOWED RESTARTING WITH GENERATION ___

This message from subroutine RDCALC indicates that *nbas* (BEG= in Sect. 8.1.3.3) was not consistent with the set of restart data that was to be used. A set of restart data is written every *nrstrt* (RES= in Sect. 8.1.3.3). The value of *nbas* should be 1 greater than one of these generations.

KMSG112 ******* ERROR ******* K?-112 FOLLOWS: ERROR IN RESTART. PARAMETER DATA AND RESTART DATA DO NOT AGREE NUMBER PER GENERATION FROM RESTART, NPBT=___NUMBER PER GENERATION FROM INPUT DATA, NPB=___ NUMBER OF ENERGY GROUPS FROM RESTART, NGPT=___ NUMBER OF ENERGY GROUPS FROM INPUT DATA, NGP=___

This message is from subroutine RDCALC. A restarted problem MUST use the same number per generation and the same number of energy groups as the parent problem that wrote the restart data. It should be verified that the correct data set is mounted on unit *rstrt*. (RST= in Sect. 8.1.3.3). This message can also be caused by a code error.

KMSG113 *** ERROR *** K?-113 FOLLOWS: ERROR IN RESTART. PARAMETER DATA AND RESTART DATA DO NOT AGREE. FISSION DENSITIES, FLUXES, OR REGION DEPENDENT FISSIONS AND ABSORPTIONS WERE REQUESTED, BUT THE GEOMETRY DATA IS INCONSISTENT. NUMBER OF GEOMETRY REGIONS FROM RESTART, KMAXT=___ NUMBER OF GEOMETRY REGIONS FROM INPUT DATA, KMAX=___

This message is from subroutine RDCALC. It should be verified that the correct data set is mounted on unit *rstrt* (RST= in Sect. 8.1.3.3). A code error can also cause this message.

KMSG114 *** ERROR *** K?-114 FOLLOWS: PARAMETER DATA SPECIFIED FLUXES BUT THE RESTART DATA DID NOT INCLUDE FLUXES.

This message is from subroutine RDCALC. The restarted problem can turn off fluxes if the parent case that wrote the restart data set calculated fluxes. However, if the parent case did not calculate fluxes, the restarted problem cannot calculate fluxes either. If the correct restart data set was mounted on *rstrt* (RST= in Sect. 8.1.3.3), the parameter data FLX=YES must be removed from the input data or FLX=NO must be entered later in the parameter data of the restarted problem.

KMSG115 *** ERROR *** K?-115 FOLLOWS: PARAMETER DATA SPECIFIED REGION DEPENDENT FISSIONS AND ABSORPTIONS, BUT THEY WERE NOT INCLUDED ON RESTART

This message is from subroutine RDCALC. The restarted problem specified FAR=YES in the parameter data block but the parent case that wrote the restart data set did not calculate region-dependent fissions and absorptions. The restarted problem can turn off region-dependent data if the parent case calculated them, but cannot turn them on if they were not calculated by the parent case. It should be verified that the correct restart data set is mounted on *rstrt* (RST= in Sect. 8.1.3.3). FAR=YES should be removed from the parameter data of the restarted problem, or FAR=NO should be added later in the parameter data. Sect. 8.1.3.3 illustrates methods of changing the parameter input data.

KMSG116 *** ERROR *** K?-116 FOLLOWS: EXECUTION IS TERMINATED.

This message from subroutine RDCALC is a companion to messages K?-112 through K?-115. A STOP 121 is executed when this message is printed.

KMSG117 *** ERROR *** K?-117 FOLLOWS: THE CALCULATION WAS TERMINATED BECAUSE ERRORS WERE ENCOUNTERED IN THE START DATA.

This message is from subroutines K5_GUIDE or K6_GUIDE. It will be accompanied by one or more of messages K?-101 through K?-104 or K?-106 through K?-110. A STOP 130 is executed when this message is printed.

KMSG118 *** ERROR *** K?-118 FOLLOWS: EXECUTION TERMINATED. RAKBAR HAS BECOME ZERO OR NEGATIVE.

This message is from subroutines K5_GUIDE or K6_GUIDE. If this message appears without other error messages, a code error is the probable cause. For such a case, this error should be reported to scalehelp@ornl.gov.

KMSG119 k?-119 JOB PULLED GENERATION= ___ NEUTRON = ___

This message from subroutine REPORTIT indicates that the problem is looping, or the time allotted for each generation, *tbtch* (TBA= in Sect. 8.1.3.3) is too small. If *tbtch* (TBA=) is increased significantly and the message occurs again for the same generation and the same neutron, it is due to a code error. Without a functional system-dependent routine (PULL) to interrupt execution, the problem will loop indefinitely, and this message will not be printed.

KMSG121 *** ERROR *** K?-121 FOLLOWS: EXECUTION TERMINATED DUE TO INSUFFICIENT TIME IN THE JOB STEP. ___ SECONDS ARE NEEDED PER GENERATION, BUT ONLY ___ REMAIN IN THE JOB STEP.

This message is from subroutines K5_GUIDE or K6_GUIDE. If more histories are desired, the job control language should be changed to allow adequate time and resubmit the problem. This message is inaccessible on computers lacking job control language to specify the maximum execution time.

KMSG122 *** ERROR *** K?-122 FOLLOWS: EXECUTION TERMINATED DUE TO EXCEEDING THE TIME SPECIFIED FOR THE PROBLEM.

This message is from subroutines K5_GUIDE or K6_GUIDE. If more histories are desired, *tmax* (`TME=` in Sect. 8.1.3.3) should be increased to allow computation of the desired number of histories.

KMSG123 K?-123 EXECUTION TERMINATED DUE TO COMPLETION OF THE SPECIFIED NUMBER OF GENERATIONS.

This message from subroutines K5_GUIDE or K6_GUIDE states that the requested number of histories has been completed. If more histories are desired, the number of generations (`GEN=` in Sect. 8.1.3.3) should be increased.

KMSG124 *** ERROR *** K?-124 FOLLOWS: THE OPTION TO USE EXTRA 1-DS WAS SPECIFIED, BUT ID NO.___ WAS NOT FOUND IN THE EXTRA 1-D ARRAY.

This self-explanatory message is from function INDX. If extra 1-Ds are specified in the parameter data (`X1D=`, Sect. 8.1.3.3), extra 1-D IDs must be entered as data. See Sect. 8.1.3.9. A STOP 107 is executed when this message is printed.

KMSG126 *** ERROR *** K5-126 FOLLOWS: CROSS ERROR ___ ___ ___ ___ ___ ___ ___ ___

This KENO V.a message from subroutine K5GM_CROS indicates a code error. The printed data, left to right, are as follows: IGEO,K,X,Y,Z,X1,Y1,Z1. IGEO is the geometry type, K is the region number, X,Y,Z is the current position, and X1,Y1,Z1 is the end point of the path. A STOP 103 is executed when this message is printed.

KMSG127 *** ERROR *** K5-127 FOLLOWS: NHCYL=___

This KENO V.a message from subroutines K5GM_CROS or K5GM_KENOG indicates invalid hemicylinder information as the result of a code error. A STOP 104 is executed when this message is printed.

KMSG128 *** ERROR *** K?-128 FOLLOWS: NEUTRON BANK IS FULL. SPLITTING NOT ALLOWED

This message from subroutine TALLYONGRID indicates that the neutron bank is too small to allow additional splitting. This can occur if the bank size, *nbank* (`NBK=` in Sect. 8.1.3.3 is too small, if the biasing or weighting data are incorrect (Sect. 8.1.3.7), or if the biasing data are incorrectly utilized in the geometry description (Sect. 8.1.3.4).

KMSG129 *** ERROR *** K?-129 FOLLOWS: ** ERROR IN SUBROUTINE ALBEDO ** FACE NUMBER ___ USES ___ ALBEDO NUMBER=___ INCIDENT XSEC ENERGY GROUP=___ INCIDENT ALBEDO ENERGY GROUP=___ INCIDENT ANGLE INDEX=___ RANDOM NUMBER=___

This message from subroutines K5GM_ALBEDO or SGGM_ALBEDO indicates that a code error was encountered when determining the output energy group during the albedo treatment.

KMSG130 *** ERROR *** K?-130 FOLLOWS: PROBLEM IN SUBROUTINE ALBEDO: FACE NUMBER ___ USES ___ ALBEDO NUMBER=___ RETURNING XSEC ENERGY GROUP=___ INCIDENT ALBEDO ENERGY GROUP=___ INCIDENT ANGLE INDEX=___ RETURNING ALBEDO ENERGY GROUP=___ RANDOM NUMBER=___

This message from subroutines K5GM_ALBEDO or SGGM_ALBEDO indicates that a code error was encountered while computing the returning angle in the albedo treatment.

KMSG131 *** ERROR *** K?-131 FOLLOWS: NO FISSIONS WERE GENERATED

This message from subroutine NSTART indicates that none of the histories in the generation encountered a fissile material, so no fission points were generated.

KMSG132 K?-132 FOLLOWS: ONLY ___ INDEPENDENT FISSION POINTS WERE GENERATED FOR GENERATION ___

This message from subroutine NSTART indicates that fewer than *npb* (`NPG=` in Sect. 8.1.3.3) fission points were generated during the previous generation. Because *npb* fission points are required by the code, the remaining fission points are randomly selected from those that were generated, thus using duplicate fission points. If the k-effective of the system is significantly less than 1.0, several of these messages should be expected in the first few generations. The code attempts to set RAKBAR so the message can be expected to occur about once every 100 generations. The message may occur more frequently in a correctly modeled problem. However, if the number of fission points is considerably less than *npb* for most of the generations, the answer can be affected.

KMSG133 *** WARNING *** K?-133 FOLLOWS: THE CALCULATION CONCLUDED WITHOUT PRODUCING RESULTS BECAUSE NO ACTIVE GENERATIONS WERE RUN. THE NUMBER OF GENERATIONS RUN WAS NOT GREATER THAN THE NUMBER OF GENERATIONS SKIPPED.

This message from subroutine KEDIT occurs if the number of generations completed is less than *nskip*+1 (`NSK=`, Sect. 8.1.3.3). In this instance, the summaries for the problem cannot be printed.

KMSG136 *** ERROR *** K5-136 FOLLOWS: GEOMETRY TYPE - IGEO = ___ - OUT OF RANGE IN ___

This message from subroutines K5GM_SRMAX, K5GM_CRMIN or K5GM_CRMAX occurs if the geometry indicator is invalid as the result of a code error. A STOP 127 is executed in conjunction with this message and a traceback may be printed from subroutine STOP.

KMSG137 *** ERROR *** K?-137 FOLLOWS: THERE ARE MISSING CROSS SECTIONS. CHECK THE OUTPUT FOR OTHER ERROR MESSAGES.

This message from subroutine FILLSG says that not all the cross sections requested have been found. There should be other previous messages giving more detail on what cross sections are missing.

KMSG138 *** WARNING *** K?-138 FOLLOWS: A WEIGHT OF 0.0 INDICATES THAT WEIGHTS WERE NOT READ OR GENERATED FOR THE BIAS ID. WEIGHTS OF 0.0 WILL BE DEFAULTED TO 0.5 PRIOR TO EXECUTION.

This message is from subroutine PRTWTS. It is printed to alert the user that weights were not entered, defaulted, or generated. This message may appear as the result of a code error.

KMSG139 *** ERROR *** K?-139 FOLLOWS: NO VALID MIXTURES WERE FOUND IN THE GEOMETRY DESCRIPTION.

This message from subroutine ICEMIX indicates that the geometry data did not specify any valid mixtures. The geometry data (Sect. 8.1.3.4) should be checked and any errors that are found should be corrected. This message can also be triggered if the unit orientation data description is not properly entered for geometry having more than one unit.

KMSG140 *** WARNING *** THE UNIT THE HOLE IS IN IS NOT USED IN THE MODEL, SO THE PROBLEM WILL NOT BE TERMINATED BECAUSE OF THIS ERROR.

This KENO V.a message from subroutine K5GM_HOLCHK is companion message to KMSG188.

KMSG141 *** ERROR *** K?-141 FOLLOWS: DATA CANNOT BE CHANGED WHEN A PROBLEM IS RESTARTED AT A GENERATION GREATER THAN ONE.

This message from subroutine DATAIN is printed if data other than parameter data are entered for a problem being restarted at a generation greater than 1. If data other than certain parameter data are to be changed, the problem must be restarted with the first generation. The error flag is set so the problem will not execute.

KMSG142 *** ERROR *** K?-142 FOLLOWS: NO GEOMETRY DATA HAS BEEN SPECIFIED IN THE INPUT DATA.

This message from subroutine DATAIN indicates that a geometry data block was not entered for the problem either as input data or from the restart unit. The data needs to be corrected and the problem resubmitted.

KMSG143 *** ERROR *** K?-143 FOLLOWS: UNIT ORIENTATION DATA IS REQUIRED IF MORE THAN ONE UNIT TYPE IS SPECIFIED IN THE GEOMETRY DATA.

This self-explanatory message is from subroutine DATAIN. Enter an array i or unit orientation data block as described in Sect. 8.1.3.5 and Sect. 8.1.4.6.

KMSG144 *** WARNING *** K?-144 FOLLOWS: DUE TO INCONSISTENCIES BETWEEN INPUT AND RESTART DATA, MATRIX INFORMATION BY UNIT TYPE WILL BE CALCULATED BUT NOT PRINTED. INPUT DATA SET MKU=NO, BUT DATA FROM THE RESTART UNIT SPECIFIED YES.

This self-explanatory warning message is from subroutine PARTBL. The matrix information by unit type cannot be eliminated if it was calculated by the original problem (parent case) that wrote the restart data. However, printing it can be avoided. It should be verified that the correct problem is being used for restarting the problem (the title is printed at the bottom of the parameter tables). The specification of the restart unit, RST, in the third table of the output should also be verified.

KMSG145 *** WARNING *** K?-145 FOLLOWS: DUE TO INCONSISTENCIES BETWEEN INPUT AND RESTART DATA, MATRIX INFORMATION BY UNIT LOCATION WILL BE CALCULATED BUT NOT PRINTED. INPUT DATA SET MKP=NO, BUT DATA FROM THE RESTART UNIT SPECIFIED YES.

This self-explanatory warning message is from subroutine PARTBL. The matrix information by unit location (also called array position or position index) cannot be eliminated if it was calculated by the original problem (parent case) that wrote the restart data. However, printing it can be avoided. It should be verified that the correct problem is being used for restarting the problem (the title is printed at the bottom of the parameter tables). The specification of the restart unit, RST, in the third table of the output should also be verified.

KMSG146 *** ERROR *** K?-146 FOLLOWS: PARAMETER DATA SPECIFIED MATRIX INFORMATION BY UNIT TYPE BUT IT WAS NOT FOUND ON THE RESTART UNIT.

This message from subroutine RDCALC is printed if a restarted problem requests matrix information by unit type when it was not requested and calculated by the original problem (parent case) that wrote the restart data. It should be verified that the correct restart data are being used and that the restart unit (RST) is correctly specified. The request for matrix information by unit type (MKU=, in the parameter data) should be eliminated if it is not necessary. The problem must be restarted with the first generation (BEG=1, in the parameter data) if matrix information by unit type is required and was not calculated by the parent case. A STOP 121 is executed in conjunction with this message.

KMSG147 *** ERROR *** K?-147 FOLLOWS: PARAMETER DATA SPECIFIED MATRIX INFORMATION BY UNIT LOCATION BUT IT WAS NOT FOUND ON THE RESTART UNIT.

This message from subroutine RDCALC is printed if a restarted problem requests matrix information by unit location (also called array position or position index) when it was not requested and calculated by the original problem (parent case) that wrote the restart data. It should be verified that the correct restart data are being used and that the restart unit (RST) is correctly specified in the first table following the parameter table. The request for matrix information by unit location (MKP=, in the parameter data) should be eliminated if it is not necessary. If matrix information by unit location is required and it was not calculated by the parent case, the problem must be restarted with the first generation (BEG=1, in the parameter data). A STOP 121 is executed in conjunction with this message.

KMSG148 *** ERROR *** K?-148 FOLLOWS: AN ERROR WAS ENCOUNTERED WHILE ATTEMPTING TO READ RESTART DATA FROM UNIT ___ NDX =___ NREC =___

This message from subroutines RDRST or RDRST_ERROR is printed if a problem is encountered while reading restart data from the restart file. This may be caused by the specified section of the restart file not being present or the file section being the wrong length. The restart file is corrupted. It should be verified that the problem ran to completion and a restart file was generated. Problems in the input should be corrected, and the problem should be run again. If the problem persists, contact the SCALE helpline.

KMSG149 *** ERROR *** K?-149 FOLLOWS: A CROSS SECTION LIBRARY WAS SPECIFIED FOR A RESTARTED PROBLEM, BUT MIXING TABLE DATA WAS NOT AVAILABLE. THE PROBLEM WILL NOT EXECUTE.

This message from subroutine READ_MIXED_XSECS means that LIB= was entered in the parameter data block and a mixing table data block was not available. A flag is set to terminate execution when the data reading has been completed. If cross sections are to be used from the restart unit (RST=), the LIB= or XSC= should be eliminated from the parameter data. If new cross sections are to be mixed, LIB= must be specified in the parameter data. The IDs in the mixing table must be available on the cross section library specified by LIB= . A problem can be restarted using a new mixed cross section library by specifying XSC= in the parameter data.

KMSG150 *** ERROR *** K5-150 FOLLOWS: TOO FEW ENTRIES WERE SUPPLIED IN THE REFLECTOR GEOMETRY DESCRIPTION.

This KENO V.a message from subroutine K5GM_KENOG indicates that too few data entries were supplied for the geometry word REFLECTOR. The mixture ID, bias ID, one of

the thickness/region specifications or the number of regions to be generated was omitted or incorrectly specified. Each REFLECTOR entry requires (1) a mixture ID, (2) a bias ID, (3) N entries for the thickness/region specifications, and (4) the number of regions to be generated. The thickness/region can be obtained from the Increment Thickness column of Table 8.1.25 for the material to be used in the regions generated by the REFLECTOR specification. N is the number of thickness/region specifications required by the geometry shape: N=1 for spheres or hemispheres, N=3 for cylinders and hemicylinders, and N=6 for cubes, cuboids, and cores. A flag is set to terminate the problem when the input data reading is completed.

KMSG151 *** ERROR *** K?-151 FOLLOWS: ___ IS AN INVALID ARRAY TYPE IN THE ARRAY DEFINITION DATA.

This message from subroutine ARAYIN indicates that a parameter name was misspelled or the data were out of order. See Sect. 8.1.3.5 for a list of the array parameter names. A stop 137 is executed in conjunction with this message.

KMSG152 *** ERROR *** K?-152 FOLLOWS: IRET=___ A PREMATURE TERMINATION WAS ENCOUNTERED WHILE READING ARRAY DATA. IRET=1 INDICATES AN END WAS FOUND. IRET=2 INDICATES AN END OF FILE.

This message from subroutine ARAYIN indicates that an array number was specified without entering the corresponding UNIT ORIENTATION DESCRIPTION. See Sect. 8.1.3.5 and Sect. 8.1.4.6 for assistance.

KMSG153 *** ERROR *** IF THE UNIT THIS HOLE IS IN IS USED IN THE MODEL, THE PROBLEM WILL NOT BE RUN.

This KENO V.a message from subroutine K5GM_READGM is a companion message to KMSG188.

KMSG154 *** ERROR *** K?-154 FOLLOWS: ARRAY LEVEL ___ IS LARGER THAN ___, THE NUMBER OF ARRAYS. CHECK THE ARRAY AND GEOMETRY INPUT FOR AN ARRAY OR UNIT THAT IS RECURSIVELY NESTED.

This message from subroutines K5GM_SORTA or SGGM_SORTA indicates that the array data specified in the problem are recursively nested. An example of this follows:

array 1 contains array 3 array 2 contains array 1 array 3 contains array 2

Thus, the definition of array 1 and array 3 are intertwined in a never-ending loop. The array data (Sect. 8.1.3.5) should be corrected and the problem resubmitted. If the input data did not specify recursive nesting, a code error has occurred. A STOP 139 is executed in conjunction with this message.

KMSG155 *** ERROR *** K?-155 FOLLOWS: THE NESTING FLAG OR NUMBER OF ARRAY LEVELS HAS BEEN DESTROYED BY A CODE ERROR. THE ORIGINAL NESTING FLAG WAS SET _ IT IS NOW SET _. THE ORIGINAL NESTING LEVEL WAS ___, IT IS NOW ___

This self-explanatory message is from subroutines K5GM_LODARA or SGGM_LOADARA. A STOP 140 is executed in conjunction with this message.

KMSG156 *** ERROR *** K?-156 FOLLOWS: MIXTURE ___ CONTAINS AT LEAST ONE ZERO VALUE FOR THE TOTAL CROSS SECTION.

This message is from subroutine XSEC1D. All the total cross sections must be positive. Zero total cross sections can occur if all the components of a mixture are mixed with a zero-number density. Correct the mixing table for the specified mixture and resubmit the problem.

KMSG157 *** ERROR *** K5-157 FOLLOWS: THE FIRST HOLE IN A UNIT MUST FOLLOW A VALID GEOMETRY REGION.

A STOP 141 accompanies this KENO V.a message from subroutine K5GM_READGM. If holes are to be used in the geometry region data (Sect. 8.1.3.4), they must follow the region in which they are to be placed. This message indicates that HOLE was the first geometry description in a unit or was placed inside an ARRAY description. Correct the geometry region data and resubmit the problem.

KMSG158 *** ERROR *** K5-158 FOLLOWS: THE NUMBER OF HOLES IS INCORRECT. IHOL=___ NUMHOL=___

This KENO V.a message is from subroutine K5GM_READGM. A code error is the probable cause of this error.

KMSG159 *** ERROR *** K?-159 FOLLOWS: ARRAY NUMBER ___ SPECIFIED IN THE GEOMETRY REGION DATA WAS NOT ENTERED IN THE ARRAY DATA.

This message from subroutines K5GM_SORTA or K5GM_HOLE or subroutine SGGM_SORTA occurs if the array number specified for an ARRAY region description of the EXTENDED GEOMETRY data (Sect. 8.1.3.4) did not have the corresponding UNIT ORIENTATION DATA entered in the ARRAY DATA (Sect. 8.1.3.5). A STOP 142 is executed in conjunction with this message when it is printed from subroutine SORTA. The data should be corrected and the problem resubmitted.

KMSG160 *** ERROR *** K?-160 FOLLOWS: THE HOLES ARE RECURSIVELY NESTED.

This message from subroutines K5GM_HOLE or SGHM_HOLE indicates that the geometry region data description (Sect. 8.1.3.4) specifies holes that are recursively nested. This can occur if a unit contains a hole whose definition traces back to the same unit or are defined in terms of each other. A simple example of recursive nesting is

KENO V.a:

```
  UNIT 1      CUBE  0  1  2P10.0    HOLE  2  3*0.0
  UNIT 2      CUBE  0  1  2P10.0    HOLE  1  3*0.0

KENO-VI:
```

```
  UNIT 1 CUBOID  10 6P10.0 BOUNDARY 10 HOLE 2 1  3*0.0
  UNIT 2 CUBOID  10 6P10.0 BOUNDARY 10 HOLE 1 1  3*0.0

Thus ``UNIT`` 1 contains ``UNIT`` 2 and ``UNIT`` 2 contains ``UNIT`` 1.
The geometry region data should be checked for recursive nesting. In the
absence of recursive nesting, a code error is the probable cause of this
message. A STOP 143 is executed when this message occurs and a traceback
is printed.
```

KMSG161 *** ERROR *** K5-161 FOLLOWS: THE GLOBAL ARRAY SPECIFIED IN THE GEOMETRY REGION DATA IS ___ BUT THE GLOBAL ARRAY SPECIFIED IN THE ARRAY DATA IS ___. EXECUTION IS TERMINATED.

This KENO V.a message is from subroutine K5_FLDATA. A STOP 144 is executed in conjunction with it. The global array specified in the array data (Sect. 8.1.3.5) was entered using the keyword GBL=. It was not consistent with the implied global array from the geometry region data (Sect. 8.1.3.4). The global array number in the geometry region data is defined to be the array number in the global unit or of the last ARRAY description that does not immediately follow a UNIT definition (i.e., other geometry definitions occur between UNIT and ARRAY). The data should be corrected, and the problem resubmitted.

KMSG162 *** ERROR *** K5-162 FOLLOWS: THE GLOBAL ARRAY WAS NOT CONSISTENTLY SPECIFIED. THE ARRAY DATA SPECIFIED ARRAY NUMBER ___ AND THE GEOMETRY DATA SPECIFIED ARRAY NUMBER ___.

This KENO V.a message from subroutine K5GM_SORTA occurs if GBL= in the array data (Sect. 8.1.3.5) does not agree with the global array number implicitly set in the geometry region data (Sect. 8.1.3.4). The geometry region data define the global array number in the global unit or to be the array number of the last ARRAY description that does not immediately follow a UNIT definition (i.e., other geometry definitions occur between UNIT and ARRAY). A flag is set to terminate execution when the data reading is completed.

KMSG163 *** ERROR *** K5-163 FOLLOWS: HOLE NUMBER ___ (UNIT NUMBER ___) INTERSECTS REGION NUMBER ___

This self-explanatory KENO V.a message is from subroutine K5GM_HOLHOL.

KMSG164 *** ERROR *** K5-164 FOLLOWS: HOLE NUMBER ___ (UNIT NUMBER ___) INTERSECTS REGION NUMBER ___.

This self-explanatory KENO V.a message is from subroutine K5GM_HOLEXT.

KMSG165 *** ERROR *** K5-165 FOLLOWS: ** ERROR ** GEOMETRY TYPE OUT OF RANGE IN SUBROUTINE ADJUST. IGEOH = ___

This KENO V.a message from subroutine K5GM_ADJUST indicates that the geometry type, IGEO, falls outside the allowable range of 1 through 19. This can occur if a HOLE references an undefined unit number. If all the unit numbers referenced by the holes are valid, a code error is the probable cause.

KMSG166 *** ERROR *** K5-166 FOLLOWS: THE PLUS CUBE FACE IS NOT GREATER THAN THE MINUS CUBE FACE PLUS FACE = ___ MINUS FACE = ___

This self-explanatory KENO V.a message is from subroutine K5GM_KENOG.

KMSG167 *** ERROR *** K5-167 FOLLOWS: ONE OR MORE PLUS FACES OF A CUBOID ARE NOT GREATER THAN THE CORRESPONDING MINUS FACES +X = ___ -X = ___ +Y = ___ -Y = ___ +Z = ___ -Z = ___

This self-explanatory KENO V.a message is from subroutine K5GM_KENOG.

KMSG168 *** ERROR *** K5-168 FOLLOWS: THE CYLINDER RADIUS IS NOT GREATER THAN ZERO, OR THE + HEIGHT IS NOT GREATER THAN THE - HEIGHT RADIUS = ___ +H = ___ -H = ___

This self-explanatory KENO V.a message is from subroutine K5GM_KENOG.

KMSG169 *** ERROR *** K5-169 FOLLOWS: THE SPHERE RADIUS IS NOT GREATER THAN ZERO - RADIUS = ___

This self-explanatory KENO V.a message is from subroutine K5GM_KENOG.

KMSG170 *** ERROR *** K5-170 FOLLOWS: A REPLICATE THICKNESS IS LESS THAN ZERO SURFACE NUMBER = ___ THICKNESS = ___

This KENO V.a message from subroutine K5GM_KENOG specifies that a thickness on a replicate was specified as a negative number. This is not allowed, as it would lead to an intersection.

KMSG171 *** ERROR *** K?-171 FOLLOWS: DIRECTION COSINES DOWN THE PAGE WERE ALL INPUT AS ZERO.

This message from subroutine RDPLOT indicates that the values for UDN=, VDN- and WDN= were all zero. A zero value vector does not define a direction, so an error has occurred. See Sect. 8.1.3.11 for information concerning direction cosines down the page.

KMSG172 *** ERROR *** K?-172 FOLLOWS: DIRECTION COSINES ACROSS THE PAGE WERE ALL INPUT AS ZERO.

This message from subroutine RDPLOT indicates that the values for UAX=, VAX= and WAX= were all zero. This is an error because a zero-value vector does not define a direction. See Sect. 8.1.3.11 for assistance in defining direction cosines across the page.

KMSG173 *** ERROR *** K?-173 FOLLOWS: ERRORS WERE DETECTED IN THE INPUT DATA FOR PLOT ___. IT WILL NOT BE DRAWN

This message from subroutine RDPLOT is a companion to messages K?-171, K?-172, K?-174, and K?-180. The error that triggered the companion message should be corrected, and the problem resubmitted.

KMSG174 *** ERROR *** K?-174 FOLLOWS: ERROR IN KENO PLOT DATA- ___ ___ SHOULD BE ENTERED AS ___YES, OR ___NO

This self-explanatory message is from subroutine RDPLOT. The error should be corrected, and the problem resubmitted. See Sect. 8.1.3.11 for assistance.

KMSG175 *** WARNING *** K?-175 FOLLOWS: DUE TO INCONSISTENCIES BETWEEN INPUT AND RESTART DATA, MATRIX INFORMATION BY HOLE WILL BE CALCULATED BUT NOT PRINTED. INPUT DATA SET MKH=NO, BUT DATA FROM THE RESTART UNIT SPECIFIED YES.

This self-explanatory warning message is from subroutine PARTBL. The matrix information by hole cannot be eliminated if it was calculated by the original problem (parent case) that wrote the restart data. However, printing it can be avoided. It should be verified that the correct problem is being used for restarting the problem (the title is printed at the bottom of the parameter tables). The specification of the restart unit, RST, in the third table of the computer output should be verified.

KMSG176 *** WARNING *** K?-176 FOLLOWS: DUE TO INCONSISTENCIES BETWEEN INPUT AND RESTART DATA, MATRIX INFORMATION BY ARRAY WILL BE CALCULATED BUT NOT PRINTED. INPUT DATA SET MKA=NO BUT DATA FROM THE RESTART UNIT SPECIFIED YES.

This self-explanatory warning message is from subroutine PARTBL. The matrix information by array cannot be eliminated if it was calculated by the original problem (parent case) that wrote the restart data. However, printing it can be avoided. It should be verified that the correct problem is being used for restarting the problem (the title is printed at the bottom of the parameter tables). Specification of the restart unit, RST, in the third table in the computer output should also be verified.

KMSG177 *** ERROR *** K?-177 FOLLOWS: PARAMETER DATA SPECIFIED MATRIX INFORMATION BY HOLE BUT IT WAS NOT FOUND ON THE RESTART UNIT.

This message from subroutine RDCALC is printed if a restarted problem requests matrix information by hole when it was not requested and calculated by the original problem (parent case) that wrote the restart data. It should be verified that the correct restart data file is being used and that the restart unit (RST) is correctly specified. The request for matrix information by hole (MKH=, in the parameter data) should be eliminated if it is not necessary. The problem must be restarted with the first generation (BEG=1, in the parameter data) if matrix information by hole is required and was not calculated by the parent case. A STOP 121 is executed in conjunction with this message.

KMSG178 *** ERROR *** K?-178 FOLLOWS: PARAMETER DATA SPECIFIED MATRIX INFORMATION BY ARRAY BUT IT WAS NOT FOUND ON THE RESTART UNIT.

This message from subroutine RDCALC is printed if a restarted problem requests matrix information by array (also called array position or position index) when it was not requested and calculated by the original problem (parent case) that wrote the restart data. It should be verified that the correct restart data are being used and that the restart unit (RST) is correctly specified in the first table following the parameter tables. The request for matrix information by array (MKA=, in the parameter data) should be eliminated if it is not necessary. If matrix information by array is required and it was not calculated by the parent case, the problem must be restarted with the first generation (BEG=1, in the parameter data). A STOP 121 is executed in conjunction with this message.

KMSG179 *** ERROR *** K5-179 FOLLOWS: HOLE NUMBER ___ (UNIT NUMBER ___) INTERSECTS HOLE NUMBER ___ (UNIT NUMBER ___) IN REGION NUMBER ___.

This KENO V.a message from subroutine K5GM_HOLHOL indicates that the specified holes intersect. Check the dimensions and origins of the units being placed in the region.

KMSG180 *** ERROR *** K?-180 FOLLOWS: PROBLEM IN KENO PLOT DATA - KEYWORD ___ IS NOT VALID

This message from subroutine RDPLOT indicates that the plot data are out of order or a keyword is incorrectly spelled. See Sect. 8.1.3.11 for a list of correct keywords.

KMSG181 *** ERROR *** K?-181 FOLLOWS: LPIC IS OUT OF RANGE. LPIC=___

This message from subroutines K5GM_PRTPLT or K6GM_PRTPLT indicates that a code error has occurred or the type of plot (PIC=) was not properly specified. *lpic*=l for a mixture map, *lpic*=2 for a unit map, and *lpic*=3 for a bias ID map. Any other values of *lpic* are invalid.

KMSG182 *** ERROR *** K5-182 FOLLOWS: AN ARRAY WAS SPECIFIED IN THE GLOBAL UNIT, BUT ARRAY DATA WAS NOT ENTERED.

This KENO V.a message from subroutine K5GM_SORTA indicates that the global unit contains an array specification, but the array was not entered in the array data block. A typo may need to be corrected in the `GLOBAL UNIT` or in the `ARRAY` data block.

KMSG183 *** WARNING *** K5-183 FOLLOWS: UNIT ___ WAS NOT CHECKED FOR GEOMETRIC CONSISTENCY. IT CONTAINS ARRAY ___ BUT WAS NOT USED IN THE PROBLEM.

This KENO V.a warning message is from subroutine K5GM_JOMCHK. It indicates that a unit whose first region is an array was described in the extended geometry data, but that unit was not referenced in the unit orientation data (see Sect. 8.1.3.4 and Sect. 8.1.3.5). KENO usually checks all the geometry region data to be sure it is correct, even when the unit is not used in the problem. The code is unable to make these checks when a unit containing an array is not used in the problem. It is not necessarily an error, but the user should double check to be sure that unit was intentionally omitted from all the arrays.

KMSG184 *** ERROR *** K5-184 FOLLOWS: ALBEDOS WERE SPECIFIED FOR A PROBLEM WHOSE OUTER BOUNDARY IS NOT A CUBE OR CUBOID. THE PROBLEM WILL NOT EXECUTE.

This KENO V.a message from subroutine K5GM_JOMCHK indicates that a non-vacuum albedo boundary condition was entered for a curvilinear outer surface. This is not allowed, and the problem will not be run.

KMSG185 *** ERROR *** K?-185 FOLLOWS: ** ERROR ** THE NUMBER OF SETS OF BIASING CORRELATION DATA EXCEEDS THE NUMBER THAT WAS WRITTEN WHEN THE BIASING DATA WAS READ. ___ WERE WRITTEN, BUT AN ATTEMPT WAS MADE TO READ ___.

This message from subroutine WAITIN is accompanied by a STOP 148. It indicates a code error unless it is accompanied by error messages related to the biasing input data.

KMSG186 *** ERROR *** K?-186 FOLLOWS: THE NUMBER OF SETS OF BIASING AUXILIARY DATA EXCEEDS THE NUMBER THAT WAS WRITTEN WHEN THE BIASING DATA WAS READ. ___ WERE WRITTEN, BUT AN ATTEMPT WAS MADE TO READ ___.

This message from subroutine WAITIN is accompanied by a STOP 149. It indicates a code error unless it is accompanied by error messages related to the biasing input data.

KMSG187 *** WARNING *** K?-187 FOLLOWS: INTERVALS IN THE ABOVE RANGE WERE NOT USED. THIS COULD LEAD TO IMPROPER BIASING.

This message from subroutine LODWTS is printed to remind the user that at least one of the specified intervals was not used in the problem. This can result in improper biasing. Biasing should not be used between fissile units. When biasing is used, it should be flat or increasing as distance from the fissile material increases and flat or decreasing as a history moves toward fissile material. See Sect. 8.1.3.7 for additional assistance.

KMSG188 *** ERROR *** K?-188 FOLLOWS: HOLE NUMBER ___ REFERENCES UNDEFINED UNIT NUMBER ___

This message is printed by subroutines K5GM_HOLCHK, K5GM_HOLE, K5GM_READGM, or SGGM_HOLE if the unit number referenced by the hole is less than 1 or greater than the largest unit number in the geometry data. The message is printed by subroutine HOLE if the unit number referenced by the hole is larger than zero and not greater than the largest unit number in the geometry data but is a unit number for

which all data are missing. The message is printed by subroutine HOLCHK if the unit number referenced by the hole is undefined. Message K?-165 may accompany this message. Specify a valid unit number (*lhole* — see EXTENDED GEOMETRY DESCRIPTION, Sect. 8.1.3.4) and resubmit the problem.

KMSG189 *** ERROR *** K6-189 FOLLOWS: HOLE NUMBER ___ REFERENCES UNDEFINED UNIT NUMBER ___

This self-explanatory KENO-VI message is from subroutine SSGM_READGM.

KMSG190 *** ERROR *** K?-190 ERROR IN PLOT DATA - OPTION ___ IS NOT VALID FOR KEYWORD PIC=.

This message from subroutine RDPLOT indicates an incorrect option associated with the keyword `PIC=`. See Sect. 8.1.3.11. Acceptable options include MAT, MIX, MIXT, MEDI, UNT, UNIT, IMP, BIAS, WTS, WGT, WGTS, or WEIGH.

KMSG191 *** ERROR *** K?-191 FOLLOWS: START TYPE ___ IS INVALID FOR A PROBLEM THAT DOES NOT HAVE A GLOBAL ARRAY.

This message from subroutine DATAIN occurs if the start type (`NST=` in the start data, Sect. 8.1.3.8) is 2, 3, 4, or 5. A global array is required to use the specified start type.

KMSG192 *** WARNING *** K?-192 FOLLOWS: MATRIX INFORMATION BY ARRAY WAS SPECIFIED AS YES IN THE PARAMETER DATA (MKA=), BUT IS NOT OF USE UNLESS ARRAYS ARE SPECIFIED.

This warning message from subroutine DATAIN is self-explanatory. The code redefines the problem so matrix information will not be collected by array number.

KMSG193 *** WARNING *** K?-193 FOLLOWS: MATRIX INFORMATION BY HOLE WAS SPECIFIED AS YES IN THE PARAMETER, DATA (MKH=), BUT IS NOT OF USE UNLESS HOLES ARE SPECIFIED.

This warning message from subroutine DATAIN is self-explanatory. The code redefines the problem so matrix information will not be collected by hole number.

KMSG194 *** WARNING *** K?-194 FOLLOWS: MATRIX INFORMATION BY UNIT LOCATION WAS SPECIFIED AS YES IN THE PARAMETER DATA (MKP=), BUT IS NOT ALLOWED BECAUSE A GLOBAL ARRAY WAS NOT SPECIFIED.

This warning message from subroutine DATAIN is self-explanatory. The code redefines the problem so matrix information will not be collected by unit location.

KMSG195 *** ERROR *** K?-195 FOLLOWS: CHARACTER STRING EXCEEDS THE SPECIFIED LENGTH. CHECK FOR ENDING DELIMITER.

This error message is from subroutine RCHRS. It indicates that either the character string exceeds 132 characters or the ending delimiter was omitted for `TTL=` (plot title, Sect. 8.1.3.11) or for `COM=` (unit comment, Sect. 8.1.3.4, or array comment, Sect. 8.1.3.5). A STOP 153 is executed when this message is printed.

KMSG196 *** ERROR *** K?-196 FOLLOWS: A PROBLEM CANNOT BE RESTARTED WHEN RESTART DATA DO NOT EXIST FOR THE SPECIFIED GENERATION AND THE NEXT GENERATION FOR WHICH RESTART DATA ARE AVAILABLE IS LARGER THAN THE REQUESTED NUMBER OF GENERATIONS. ** ** EXECUTION IS TERMINATED.

This message from subroutine RDCALC indicates that a problem was to be restarted but the requested number of generations (GEN= in Sect. 8.1.3.3) was smaller than the beginning generation number (BEG= in Sect. 8.1.3.3). The beginning generation number for a restarted problem is the generation at which the calculation of k-effectives and associated information is resumed. Therefore, the number of generations to be run must be larger than the beginning generation number. The data should be corrected and the problem resubmitted. A STOP 154 is executed when this message is printed.

KMSG197 *** ERROR *** K5-197 FOLLOWS: A VALID GEOMETRY REGION MUST PRECEDE A REPLICATE REGION.

This KENO V.a message from subroutine K5GM_KENOG indicates that a replicate specification follows an invalid geometry specification (for example, REPLICATE immediately follows a UNIT specification).

KMSG198 *** ERROR *** K?-198 FOLLOWS: ARRAY ___ CONTAINS AN ERROR IN THE INPUT DATA.

This message from subroutine ARAYIN is printed as the result of an error in the FILL input data for the specified array. One or more messages from the library routine YREAD should immediately precede this message and indicate the nature of the error. The data should be corrected and the problem resubmitted. Messages K?-32, K?-33, and/or K?-85 may also print as a result of this error.

KMSG199 *** ERROR *** K?-199 FOLLOWS: THERE IS NO MIXING TABLE

This self-explanatory message comes from subroutine PREMIX-SETUP. A continuous energy problem requires a mixing table, but none was entered.

KMSG200 *** ERROR *** K?-200 FOLLOWS: START TYPE 6 WAS SPECIFIED IN THE START DATA, BUT THE STARTING POINTS WERE NOT SPECIFIED.

This message from subroutine RDSTRT indicates that start type 6 was specified but the corresponding starting points were not included in the START data block. The corresponding starting points must be entered or the start type must be changed. See Sect. 8.1.3.8.

KMSG201 *** WARNING *** K?-201 FOLLOWS: NEUTRON ___ SPECIFIED A POSITION IN THE GLOBAL ARRAY. THE GLOBAL UNIT DID NOT CONTAIN AN ARRAY SO THE POSITION WAS IGNORED ** WARNING ******** WARNING ***** WARNING ******** WARNING ******** WARNING ******** WARNING ******** WARNING ******** WARNING ******** WARNING ******** WARNING **

This warning message from subroutine K_STDATA indicates that extraneous data were specified in the start type 6 data. It should be verified that the correct global unit is specified.

KMSG202 *** ERROR *** K?-202 FOLLOWS: ONLY START TYPES 0,1,6 OR 7 ARE VALID FOR A PROBLEM WITHOUT AN ARRAY IN THE GLOBAL UNIT.

This message from subroutines K5_START or K6_START indicates that the specified start type is not valid for the problem. An appropriate start type must be chosen.

KMSG203 *** ERROR *** K?-203 FOLLOWS: UNIDENTIFIED KEYWORD ___ IN MIXING TABLE

This message from subroutine MIXING_TABLE indicates that when reading the mixing table, an unrecognized keyword was found. See Sect. 8.1.3.10 for the acceptable keywords.

KMSG204 K?-204 EXECUTION TERMINATED DUE TO ACHIEVING THE STANDARD DEVIATION SPECIFIED FOR THE PROBLEM. NOTE THAT AT LEAST 50 OR NSK*2 ACTIVE GENERATIONS ARE ALWAYS RUN.

This message from subroutines K5_GUIDE or K6_GUIDE indicates that the specified standard deviation has been achieved and the problem will therefore terminate at the current generation. If a lower standard deviation is desired the requested standard deviation (SIG=, Sect. 8.1.3.3) must be decreased.

KMSG205 *** ERROR *** K5-205 FOLLOWS: A GLOBAL UNIT MUST BE SPECIFIED FOR A SINGLE UNIT PROBLEM.

This message from subroutine K5_FLDATA is printed if a global unit is not specified for a single unit problem. A STOP 164 is executed when the message is printed. If the input data do not specify the global unit, it is defaulted to Unit 1.

KMSG206 *** ERROR *** K?-206 FOLLOWS: THE UNIT SPECIFIED FOR STARTING IS NOT USED IN THE PROBLEM OR IS UNDEFINED.

This message from subroutines K5_START or K6_START indicates that the unit in which neutrons are to be started is undefined. It should be verified that the global unit or array is correctly specified. The start data should be checked for start types 4 and 5 (Sect. 8.1.3.8) to be sure that NBX= is correctly specified.

KMSG207 *** ERROR *** K?-207 FOLLOWS: THE STARTING ARRAY POSITION IS INVALID.

This message from subroutines K5_START or K6_START indicates that the array position NXS, NYS, or NZS is not valid for start types 3 or 6. NXS, NYS, and NZS must be larger than zero and no larger than NBXMAX, NBYMAX, and NBZMAX of the global array, respectively. It should be verified that the GLOBAL UNIT or ARRAY is correctly specified. The start data (Sect. 8.1.3.8) should be corrected and the problem resubmitted.

KMSG208 *** WARNING *** K?-208 FOLLOWS: THE FRACTION OF NEUTRONS STARTED AS A SPIKE WAS LESS THAN ZERO. IT HAS BEEN RESET TO ZERO.

This message from subroutines K5_START or K6_START indicates that FCT= was incorrectly specified in the start data for start type 2. The resultant starting distribution is a cosine distribution throughout the volume of a cuboid defined by XSM, XSP, YSM, YSP, ZSM, and ZSP (see Sect. 8.1.3.8). If a spike was desired, FCT= should be set to a positive number between 0.0 and 1.0. If FCT =0.0 is specified, a cosine distribution without a spike is used as the starting distribution. If FCT=1.0 is specified, all the neutrons are started as a spike (i.e., they are started uniformly in the unit located at position NXS, NYS, NZS in the global array) as noted in Sect. 8.1.3.8

KMSG209 *** WARNING *** K?-209 FOLLOWS: THE FRACTION OF NEUTRONS STARTED AS A SPIKE WAS GREATER THAN ONE. IT HAS BEEN RESET TO ONE.

This message from subroutines K5_START or K6_START indicates that FCT= was incorrectly specified in the start type 2 data. The code reset FCT=1, so all the neutrons are started as a spike (i.e., they are started uniformly in the unit located at NXS, NYS, NZS in the global array) as noted in Sect. 8.1.3.8

KMSG210 *** ERROR *** K?-210 FOLLOWS: THE UNIT SPECIFIED FOR STARTING IS NOT IN THE GLOBAL ARRAY.

This message from subroutines K5_START or K6_START indicates that the unit specified by NBX= does not occur in the global array. It should be verified that the global array is correctly specified and that the unit specified by NBX= is correct (see Sect. 8.1.3.8).

KMSG211 *** ERROR *** K?-211 FOLLOWS: THE NUMBER OF SETS OF BIAS FACTORS FROM CARDS EXCEEDS THE NUMBER THAT WAS WRITTEN WHEN THE BIASING DATA WAS READ. ___ WERE WRITTEN, BUT AN ATTEMPT WAS MADE TO READ ___.

This message from subroutine WAITIN indicates a code error. A STOP 161 is executed when this message is printed.

KMSG212 *** ERROR *** K?-212 FOLLOWS: THE BIASING DATA SPECIFIED IBGN=___ AND IEND=___. IBGN MUST BE LARGER THAN ZERO AND IEND MUST BE AT LEAST AS LARGE AS IBGN. THE PROBLEM WILL NOT BE RUN.

This message from subroutine RDBIAS indicates an error in the biasing data. The biasing correlation data is order dependent. The order of data entry is ID=*nn ibgn iend*, where *nn* is an ID number from Table 8.1.25 and *ibgn* is the beginning BIAS ID and *iend* is the ending BIAS ID (see Sect. 8.1.3.7 and Sect. 8.1.3.7). To continue checking the input data, if *ibgn* is less than or equal to zero, it is set to 1. Similarly, if *iend* is less than *ibgn*, it is set to *ibgn*.

KMSG214 ***** ERROR ***** K6-214 FOLLOWS: THE BASE OF A TRIANGULAR FACE OF A WEDGE CANNOT BE ZERO.

This KENO-VI message is from subroutine SGGM_WEDGE. The input data for the wedge following the keyword WEDGE are incorrect. See Sect. 8.1.8.2.

KMSG215 ***** ERROR ***** K6-215 FOLLOWS: THE Y COORDINATE OF A TRIANGULAR FACE OF A WEDGE CANNOT BE ZERO.

This KENO-VI message is from subroutine SGGM_WEDGE. The input data for the wedge following the keyword WEDGE are incorrect. See Sect. 8.1.8.2.

KMSG216 ***** ERROR ***** K5-216 FOLLOWS: UNIT ___ CONTAINS ARRAY ___ WHICH WAS NOT DEFINED IN THE INPUT DATA.

This message from subroutine K5GM_JOMCHK indicates that the unit specified an array, but the array number was not entered in the array data block. The array number specified in the unit should be checked to verify that it is correct, or the unit orientation data should be entered in the array data block.

KMSG217 ***** ERROR ***** K?-217 FOLLOWS: *** ERROR *** DIFFERENTIAL ALBEDOS CANNOT BE USED IN AN ADJOINT PROBLEM.

This self-explanatory message is from subroutines MFLDATA, K5_FLDATA or K6_FLDATA. Reflector material in the mixing table and the geometry should be described instead of using differential albedos, or the problem should be run in the forward mode.

KMSG218 ***** ERROR ***** K?-218 FOLLOWS: AN INPUT DATA ERROR HAS BEEN ENCOUNTERED IN THE ___ DATA ENTERED FOR THIS PROBLEM.

This message is from subroutines INITAL, DATAIN, or KENOBUILDER. Additional error messages should be found and corrected, and the data in the named data block should be checked.

KMSG219 ***** ERROR ***** K?-219 FOLLOWS: THE START DATA SPECIFIED ___ STARTING POINTS CHOSEN FROM A COSINE DISTRIBUTION BUT NONE WERE FOUND.

This message from subroutines K5_START or K6_START is printed if start type 2 was specified and the code was unable to start any neutrons in the cuboid defined by XSM, XSP, YSM, YSP, ZSM, ZSP. It should be verified that fissile material exists within that cuboid. If it does not, the starting cuboid should be respecified to contain fissile material, or a different start type should be chosen. If message KMSG105 states that only 0 independent starting points were generated, it indicates that the code was unable to start any neutrons in the spike specified by start type 2. It should be verified that the unit specified for the spike contains fissile material. If only a very small fraction of the volume of this unit is fissile, it may be necessary to enter a larger value for the KENO parameter **TBA**= or a different start type may need to be selected. The problem will not be run if message KMSG219 is printed.

KMSG220 ***** ERROR ***** K?-220 FOLLOWS: ERROR IN PLOT DATA. IF THE COORDINATE OF THE LOWER RIGHT-HAND CORNER IS ENTERED, ONE OF THE PLOT PARAMETERS DLX, DLD, NAX OR NDN MUST BE ENTERED. CURRENT VALUES ARE LISTED BELOW. TITLE: _____ UPPER LEFT LOWER RIGHT COORDINATES COORDINATES X _____ _____ Y _____ _____ Z _____ _____ U AXIS V AXIS (DOWN) (ACROSS) X _____ _____ Y _____ _____ Z _____ _____ NDN=___ NAX=___ DLD=___ DLX=___

This message from subroutine RDPLOT indicates that the coordinates of the lower right-hand corner of the plot were specified in the input data without specifying one of the following plot parameters: (1) NDN, the number of characters down the page, (2) NAX, the number of characters across the page, (3) DLD, the vertical spacing between points, or (4) DLX, the horizontal spacing between points. The problem will not be run. To correct the error, NDN, NAX, DLD, or DLX should be specified in the plot data and the problem resubmitted. See Sect. 8.1.3.11 and Sect. 8.1.4.10 for assistance.

KMSG221 ***** ERROR ***** K6-221 FOLLOWS: THE LENGTH OF THE EDGE ALONG THE BASE OF THE X AXIS FOR A RHOMBOID MUST BE GREATER THAN 0.0. CHECK GEOMETRY WORD ___

This KENO-VI message is from subroutine SGGM_RHOMB. The data following the keyword RHOMB should be checked. See Sect. 8.1.8.2.

KMSG222 ***** WARNING ***** K?-222 FOLLOWS: ___ TRANSFERS FOR MIXTURE ___ WERE CORRECTED FOR BAD MOMENTS.

This message from subroutine MAKANG indicates moments were corrected to eliminate negative probabilities for calculated angles. If the moment was changed by more than EPS times the moment, a K?-60 or K?-64 message is printed. If EPS is very small, any change without an accompanying message is trivial. The K?-60 or K?-64 messages can be used to determine if the affected transfer and correction are significant. Most messages are caused by the cross sections being in single precision and the moments calculations being done in double precision. Generally, as the number of energy groups increase, so does the number of corrected transfers for a given mixture.

KMSG224 ***** ERROR ***** K6-224 FOLLOWS: THE HEIGHT OF A WEDGE CANNOT BE ZERO.

This KENO-VI message is from subroutine SGGM_WEDGE. The input data for the wedge following the keyword WEDGE are incorrect. See Sect. 8.1.8.2.

KMSG225 ***** ERROR ***** K6-225 FOLLOWS: TO START IN A GLOBAL PLANE GEOMETRY, IGEO =___ VALUES FOR XSM, XSP, YSM, YSP, ZSM, AND ZSP MUST BE ENTERED AS START DATA.

This error message is from subroutine K6_STRTSU. A GLOBAL PLANE GEOMETRY implies an infinite media. The boundary of the starting points must be entered. See Sect. 8.1.3.8.

KMSG226 ***** ERROR ***** K6-226 FOLLOWS: XDIST, YDIST, AND ZDIST SPECIFIED FOR A PPIPED MUST BE GREATER THAN 0.0. CHECK GEOMETRY WORD ___

This KENO-VI message is from subroutine SGGM_PPIPED. The data must be checked following the keyword PPIPED. See Sect. 8.1.8.2.

KMSG227 ***** ERROR ***** K6-227 FOLLOWS: PSI, THETA, AND PHI SPECIFIED FOR A PARALLELEPIPED MUST BE GREATER THAN OR EQUAL TO 0.0 AND LESS THAN 90.0. CHECK GEOMETRY WORD ___

This KENO-VI message is from subroutine SGGM_PPIPED. The data following the keyword PPIPED should be checked. See Sect. 8.1.8.2.

KMSG228 ***** ERROR ***** K5-228 FOLLOWS: ___ ___ IS INVALID. UNITS MUST BE GREATER THAN 0. THE PREVIOUS UNIT NUMBER IS ___. PRINTING THE GEOMETRY AS READ WILL BE TURNED ON

This message from subroutine K5GM_KENOG is printed if a unit number, *nn*, is read which is not greater than 0. This unit followed unit *mm* in the input (if *mm* is 1, the unit *nn* may be the first unit in the geometry). The geometry following this will be printed as read as an aid in locating the error. The error must be corrected, and all unit numbers must be greater than 0.

KMSG229 ***** WARNING ***** K?-229 FOLLOWS: THE NUMBER OF START TYPE 6 STARTING POINTS (LNU=___) SPECIFIED IN START DATA DOES NOT MATCH THE NUMBER OF NEUTRONS PER GENERATION (NPG=___). SPECIFIED STARTING POINTS WILL BE USED FOR FIRST ___ NEUTRONS, AND THE STARTING POINTS FOR THE REMAINING NEUTRONS WILL BE RANDOMLY SELECTED FROM THESE (LNU=___) POINTS.

This message from subroutine START6 is printed if the number of starting points specified (`LNU`) is less than the number of histories per generation (`NPG`). The remaining unspecified starting points will be randomly selected from the already specified starting points.

KMSG230 ***** ERROR ***** K?-230 FOLLOWS: MATRIX K-EFFECTIVE WAS NOT CONVERGED. K-EFFECTIVE IS ___ CONVERGENCE ERROR IS ___

This message is printed by subroutines K5_GUIDE, K6_GUIDE, or MATK if the matrix eigenvalue equation did not converge to 0.1 % after a maximum number of iterations. This is likely a code error, but it might be caused by too much variance in the matrix terms.

KMSG231 ***** ERROR ***** K6-231 FOLLOWS: THE CALCULATION WAS TERMINATED BECAUSE ARRAY ___ IS NOT PROPERLY POSITIONED IN REGION ___ OF UNIT ___. THE ARRAY BOUNDARY IS OUTSIDE POSITION ___ ___ ___ ON THE ___ ARRAY FACE.

This KENO-VI message from subroutine SGGM_PNTCHK is printed only if an array is placed in a region so that there is space in the region not occupied by the array. The array needs to be repositioned.

KMSG233 ***** ERROR ***** K?-233 FOLLOWS: MIXTURE ___ HAS A NEGATIVE CROSS SECTION FOR REACTION ___ - GROUP ___ THE VALUE IS ___

This error message from subroutine NORM1D is printed if an MT number for a specified mixture has a negative cross section value in any group. This is a library or cross section processing problem. This error should be reported to scalehelp@ornl.gov.

KMSG234 ***** ERROR ***** K?-234 FOLLOWS: MIXTURE ___ HAS NEGATIVE CROSS SECTIONS FOR REACTION ___ SET XS1=YES AND P1D=YES IN THE PARAMETERS TO SEE THE VALUE(S).

This error message from subroutine FILLSG is printed if an MT number for a specified mixture has a negative cross section value in any group. This is a library or cross section processing problem. This error should be reported to scalehelp@ornl.gov.

KMSG236 ***** ERROR ***** K?-236 FOLLOWS: EITHER NEUTRON OR GAMMA SHOULD HAVE BEEN ENTERED TO SELECT THE TYPE OF 1-D CROSS SECTIONS, BUT ___ WAS ENTERED INSTEAD.

This error message from subroutine IDX1D indicates the user specified extra 1-D cross sections but specified an invalid type. Valid types are either neutron or gamma. The data must be corrected in the `READ X1DS` data block.

KMSG237 ***** ERROR ***** K?-237 FOLLOWS: NUMBER OF SCATTERING ANGLES ASKED FOR, ___, DO NOT MATCH THE NUMBER ON THE MIXED CROSS SECTION LIBRARY ,___. UNABLE TO PROCESS FURTHER.

This error message from subroutine ICEMIX indicates the user requested a number of angles to be used in the problem, NSCT, that is inconsistent with the premixed cross section library specified for this problem. Either `SCT=` should not be specified, or it should be specified by creating a new mixed library.

KMSG238 ***** ERROR ***** K?-238 FOLLOWS: DELTA ___-COORDINATE ___ DIRECTION COSINE ACROSS ___ DIRECTION COSINE DOWN ___ ARE INCONSISTENT. IF DELTA IS ZERO, THEN BOTH DIRECTION COSINES SHOULD BE ZERO. IF DELTA IS NOT ZERO, THEN AT LEAST ONE DIRECTION COSINE SHOULD BE NONZERO, AND THE NONZERO VALUES SHOULD ALL HAVE THE SAME SIGN.

The error message from subroutine RDPLOT indicates an error in the `READ PLOT` input data. The direction cosines entered using `UAX`, `VAX`, `WAX`, `UDN`, `VDN`, and `WDN` are inconsistent with one of the following: (1) the lower right-hand corner and upper left-hand corner specified; (2) the delta across (`DLX`) and delta down (`DLD`); or (3) the number of points across (`NAX`) and the number of points down (`NDN`).

KMSG239 ***** WARNING ***** K6-239 FOLLOWS: THE VOLUMES FOR ALL REGIONS ARE NOT SET. SOME REGION VOLUMES ARE NOT SET BY INPUT, NOT IN THE SPACE FOR WHICH VOLUMES ARE CALCULATED, MISSED BY THE VOLUME CALCULATION, OR ACTUALLY 0.0. FOR THESE REGIONS, VOLUMES ARE SET TO -1.0 AND THE FLUXES AND FISSION DENSITIES ARE DIVIDED BY -1.0 INSTEAD OF THE REAL VOLUME.

This error message is from subroutine SGGM_VOLUME. If fluxes or other densities are needed, the real volumes are required. These can be calculated, with some error, or entered with as much precision as needed and the user knows.

KMSG240 ****** ERROR *****K6-240 FOLLOWS: GEOMETRY WORD UNIT IS REQUIRED BUT ___ WAS READ INSTEAD.

This error message is from subroutine SGGM_KENOG. Prior to specifying geometry or media data in the GEOMETRY DATA BLOCK, a unit must be specified.

KMSG241 ****** ERROR *****K6-241 FOLLOWS: NEUTRON ___ OF GENERATION ___ HAS LEAKED WITHOUT BEING ON THE SURFACE OF THE BOUNDARY REGION OF UNIT ___ AT X = ___ Y = ___ Z = ___ U = ___ V = ___ W = ___

This message from subroutine K6_TRACK indicates a history exited the system from somewhere internal to the geometry. The geometry of the unit at the specified point should be checked to ensure that it is correctly defined.

KMSG242 ****** ERROR *****K6-242 FOLLOWS: NEUTRON ___ IN GENERATION ___ HAS LEAKED FROM UNIT ___ AT X = ___ Y = ___ Z = ___ BUT IS STILL WITHIN THE VOLUME DEFINED BY THE UNIT BOUNDARY RECORD.

This error message from subroutines K6_TRACK or ADVANCETRACKTONEXTBOUNDARY indicates that the specified particle is inside the volume defined by the unit's boundary record but is not in any region. The media record descriptions should be checked for the specified unit. Also, specifying a screen plot of the problem may help identify the problem.

KMSG243 ***** ERROR ***** K6-243 FOLLOWS: IN UNIT ___ THE BOUNDARY RECORD CONTAINS LABEL ___ WHICH DOES NOT CORRESPOND TO A GEOMETRY RECORD.

This KENO-VI error message from subroutine SGGM_KLANGA indicates that in the specified unit the boundary record contains a geometry record label that is not linked to a geometry record in the unit.

KMSG244 ***** ERROR ***** K?-244 FOLLOWS: UNRECOGNIZABLE VOLUME CALCULATION TYPE ___ SPECIFIED

This error message from subroutine RDVOL indicates an invalid input. TYPE = parameter was encountered in the READ volume data block.

KMSG245 ***** ERROR ***** K?-245 FOLLOWS: UNIT ___ REGION ___ REFERENCES UNIT ___. AN INSTANCE OF THIS REGION HAS OCCURRED WITHOUT FINDING THE REFERENCED UNIT AT ANY NESTING LEVEL.

This error message from subroutines K5_GEOMETRY or K6_GEOMETRY indicates that the fuel center for the unit and region printed is specified with respect to the referenced unit, but the referenced unit could not be found in the geometry. A STOP 245 is executed in conjunction with this message.

KMSG246 ***** ERROR ***** K?-246 FOLLOWS: THE RESTART FILE WAS A ___ PROBLEM, BUT THE CURRENT PROBLEM IS ___. THE CURRENT PROBLEM WILL NOT BE RUN

This error message from subroutine PARAM indicates that a restart file to be read for this problem was either forward or adjoint, which conflicted with what was specified in the input. Only a forward restart file can be used for a forward problem, and only an adjoint restart file can be used for an adjoint problem.

KMSG247 ***** ERROR ***** K?-247 FOLLOWS: INVALID KEYWORD ___ IN THE VOLUME INPUT DATA ***** ERROR *****

This error message from subroutine RDVOL indicates the keyword printed is not valid in the `READ VOLUME` data block. See Sect. 8.1.3.13 for guidance on `VOLUME` inputs.

KMSG249 ***** ERROR ***** K6-249 FOLLOWS: UNDEFINED SPACE IN THE REGION CONTAINING HOLE ___.

This error message from subroutine SGGM_TRACE is caused by a unit improperly being placed in a hole, leaving a gap between crossing into a hole and entering the unit in the hole. The hole geometry must be checked.

KMSG250 K?-250 EXECUTION TERMINATED ON USER SIGNAL.

This message from subroutines K5_GUIDE or K6_GUIDE indicates that the user has set the flag to KENO so that it will not run further generations and wrap up execution. The flag is set by creating a file named stop_keno in the working directory.

KMSG251 ***** ERROR ***** K6-251 FOLLOWS: INVALID VOLUME CALCULATION TYPE ___

This error message from subroutine SGGM_VOLUME indicates that the variable VCALC has been overwritten. This indicates a code error.

KMSG253 ***** ERROR ***** K6-253 FOLLOWS: CANNOT SELECT A FACE FOR VOLUME INTEGRATION.

This error message from subroutine SGGM_PATHVOLS indicates an invalid face was selected for the volume integration. This indicates a code error.

KMSG254 ***** ERROR ***** K6-254 FOLLOWS: THE ANGLE BETWEEN THE Y AXIS AND THE Y EDGE OF THE BASE MUST BE BETWEEN 0 AND 90 DEGREES. CHECK GEOMETRY WORD ___

This KENO-VI error message from subroutine SGGM_RHOMB indicates an invalid angle has been entered. The angle must be between 0 and 90 degrees.

KMSG256 ***** ERROR ***** K6-256 FOLLOWS: RAY TRACE NUMBER ___ HAS LEAKED FROM UNIT ___ AT X= ___ Y= ___ Z= ___ BUT IS STILL WITHIN THE VOLUME DEFINED BY THE UNIT BOUNDARY RECORD.

This KENO-VI message from subroutine SGGM_TRACE occurs when volumes are calculated using ray tracing and an undefined region was encountered by a ray.

KMSG257 ***** ERROR ***** K6-257 FOLLOWS: ***** ERROR ***** KEYWORD PLACE NOT FOUND FOR ARRAY ___

This KENO-VI error message from subroutine SGGM_ARRAY means an array was improperly placed within a unit. The keyword `PLACE`, which should immediately follow the array's vector definition array, is missing from the array data. See Sect. 8.1.3.5 for more information about array placement.

KMSG258 ***** ERROR ***** K6-258 FOLLOWS: THE CALCULATION WAS TERMINATED BECAUSE A GEOMETRY WORD IN REGION ___ OF UNIT ___ PRODUCES AN IMAGINARY BOUNDARY.

This KENO-VI message from subroutine SGGM_PNTCHK should only be produced by the geometry word QUADRATIC when used as part of an array boundary. The boundary produces imaginary results. The coefficients of the geometry word should be checked.

KMSG259 ***** ERROR ***** K6-259 FOLLOWS: NEUTRON ___ FROM GENERATION ___ LOCATED IN UNIT ___ AT POSITION X = ___ Y = ___ Z = ___ IN ARRAY ___ BOUNDARY UNIT ___ GLOBAL REGION ___ AT BOUNDARY POSITION XARY = ___ YARY = ___ ZARY = ___ HAS A NEGATIVE DISTANCE TO CROSS THE ARRAY BOUNDARY THIS COULD BE DUE TO AN ARRAY BOUNDARY OVERLAPPING A HOLE.

This message from subroutines K6_TRACK, SGGM_TRACE, or NEWUNITINTERSECTIONTABLE occurs when a history computes a negative distance to cross an array boundary when it is inside the array. This can be caused by the array boundary intersecting a hole, of by round off in computing the coordinates of the point in the unit containing the array boundary for large dodecahedral arrays. For the latter case, KENO-VI will silently allow up to 5 such events per generation before giving this message and terminating.

KMSG260 ***** ERROR ***** K6-260 FOLLOWS: NEUTRON ___ FROM GENERATION ___ LOCATED IN UNIT ___ A PARTICLE AT POSITION X = ___ Y = ___ Z = ___ TRAVELING IN DIRECTION U = ___ V = ___ W = ___ TRAVELED___ CM INSIDE HOLE ___ BEFORE CROSSING THE BOUNDARY. THIS COULD BE DUE TO THE HOLE OVERLAPPING ANOTHER HOLE, UNIT, OR ARRAY BOUNDARY.

This message from subroutines K6_TRACK, SGGM_TRACE, and CROSSINTOAHOLE occurs when a history checks whether it is crossing into a hole and finds it is already inside the hole.

KMSG261 ***** ERROR ***** K6-261 FOLLOWS: NEUTRON ___ FROM GENERATION ___ LOCATED IN UNIT ___ A PARTICLE AT POSITION X = ___ Y = ___ Z = ___ TRAVELING IN DIRECTION U = ___ V = ___ W = ___ TRAVELED___ CM OUTSIDE THE UNIT BEFORE CROSSING THE BOUNDARY. THIS COULD BE DUE TO A HOLE OVERLAPPING THE UNIT BOUNDARY.

This message from subroutines K6_TRACK, SGGM_TRACE, or ADVANCETRACKTONEXTBOUNDARY occurs when a history goes to check crossing out of a unit and finds that it does not occur at the unit boundary.

KMSG262 ***** ERROR ***** K6-262 FOLLOWS: NEUTRON ___ FROM GENERATION ___ LOCATED IN UNIT ___ A PARTICLE AT POSITION X = ___ Y = ___ Z = ___ TRAVELING IN DIRECTION U = ___ V = ___ W = ___ IS OUTSIDE THE ARRAY BOUNDARY WITHOUT CROSSING OUT OF THE ARRAY THIS COULD BE DUE TO A HOLE OVERLAPPING THE ARRAY BOUNDARY.

This message from subroutine SGGM_CRSARA occurs when a history computes a negative distance to cross an array boundary when it is inside the array. This can be caused by the array boundary intersecting a hole.

KMSG263 ***** ERROR ***** K6-263 FOLLOWS: MULTIPLY DEFINED POINT - LL = ___ X = ___ Y = ___ Z = ___

This KENO-VI message from subroutine SGGM_TRACE indicates incorrect sector definitions in unit LL at point X, Y, Z.

KMSG264 ***** ERROR ***** K6-264 FOLLOWS: HOLE ___ NOT DEFINED IN REGION ___

This message from subroutine SGGM_TRACE indicates incorrect geometry specification of holes. See Sect. 8.1.4.6.1 for more information on the correction specification of `HOLE`s.

KMSG265 ***** ERROR ***** K6-265 FOLLOWS: BOUNDARY SECTOR DATA WAS NOT FOUND FOR UNIT ___

This KENO-VI message from subroutine SGGM_BNDRY indicates that the region definition vector for the unit is missing in the input.

KMSG266 ***** ERROR ***** K6-266 FOLLOWS: A VECTOR DEFINITION ARRAY IN UNIT ___ REFERENCES GEOMETRY RECORD ___ WHICH HAS NOT BEEN SPECIFIED.

This KENO-VI message from subroutine SGGM_KLANGA indicates that a region definition vector in the unit references an undefined geometry input record.

KMSG267 ***** ERROR ***** K6-267 FOLLOWS: IN UNIT ___ THE USE OF GEOMETRY LABEL ___ IN THE BOUNDARY DEFINITION VECTOR CONFLICTS WITH VECTOR DEFINITON GEOMETRY LABEL ___

This KENO-VI message from subroutine SGGM_KLANGA indicates that a geometry word referenced in a region definition vector in the unit defines the region to be on the wrong side of the unit boundary (the same geometry record is referenced with opposite signs).

KMSG268 ***** ERROR ***** K6-268 FOLLOWS: INVALID KEYWORD ___ IN THE VOLUME INPUT FILE

This self-explanatory message is from subroutine SGGM_GTVOLS, and it occurs when reading a separate volume file. See Sect. 8.1.3.13 for more information regarding the volume input file.

KMSG269 ***** ERROR ***** K6-269 FOLLOWS: FOR UNIT NUMBER ___ TOO MANY MEDIA ENTRIES WERE SPECIFIED.

This KENO-VI error message results from subroutine SGGM_GTVOLS, and it indicates that the separate volume file is inconsistent with the geometry input.

KMSG270 ***** ERROR ***** K6-270 FOLLOWS: CYLINDERS/ECYLINDERS/CONES IN THE GLOBAL UNIT BOUNDARY DEFINITION VECTOR NEED TO BE ORIENTED ALONG A MAJOR AXIS.

This KENO-VI message from subroutine SGGM_VOLCUB indicates that a bounding cuboid for the global unit cannot be calculated because one of the referenced bodies has been rotated off axis. The user can bypass this message by specifying a bounding cuboid in the input in the VOLUME data block.

KMSG271 ***** ERROR ***** K6-271 FOLLOWS: THE FIRST NON-NEGATIVE BODY SPECIFIED IN THE BOUNDARY DEFINITION VECTOR OF THE GLOBAL UNIT MUST SPECIFY FINITE VOLUME BODY.

This KENO-VI message from subroutine SGGM_VOLCUB indicates that a bounding cuboid for the global unit cannot be calculated because the first body in the boundary definition vector does not define a finite volume body. The user can bypass this message by specifying a bounding cuboid in the input in the VOLUME data block.

KMSG273 ***** ERROR ***** K?-273 FOLLOWS: THE NUMBER OF GROUPS ON THE RESTART LIBRARY DOES NOT MATCH WHAT WAS SET PREVIOUSLY. THE NUMBER OF GROUPS SET PREVIOUSLY ___ THE NUMBER OF GROUPS FROM THE RESTART UNIT ___

This self-explanatory message from subroutine READ_MIXED_XSECS indicates an inconsistency between the number of groups for cross sections on the restart unit and the number of groups specified elsewhere.

KMSG274 ***** ERROR ***** K?-274 FOLLOWS: ___ ERROR(S) OCCURED WHILE READING ___. THE MESSAGES RECEIVED FROM ___ INPUT PROCESSOR ARE AS FOLLOWS:

This error message comes from subroutines READ_DEFINITIONS_INPUT, READ_STARMESH_INPUT, or MGRID type's CHECKGRIDDATA procedure, and it indicates an error in specifying the mesh grid for the problem. See Sect. 8.1.3.14 for more information about the specification of the mesh grid geometry.

KMSG275 ***** ERROR ***** K?-275 FOLLOWS: ADJOINT SOLUTIONS ARE CURRENTLY NOT AVAILABLE IN CONTINUOUS ENERGY PROBLEMS. IF ADJOINT SOLUTION IS DESIRED, THE USER SHOULD CHOOSE THE MULTIGROUP ENERGY TREATMENT

This message from subroutine DATAIN indicates the user turned on adjoint calculation (see parameter `ADJ` in Sect. 8.1.3.3) for a continuous energy problem. The current version of KENO does not have adjoint continuous energy cross sections. If adjoint solution is desired, the user should choose the multigroup energy treatment.

KMSG276 ***** ERROR ***** K?-276 FOLLOWS: DIFFERENTIAL ALBEDOS ARE CURRENTLY NOT AVAILABLE IN CONTINUOUS ENERGY PROBLEMS.

This message from subroutine DATAIN indicates the user entered differential albedos for a continuous energy problem. The current version of KENO does not have differential albedos for continuous energy treatment. The user should replace the albedo data with appropriate materials.

KMSG277 ***** ERROR ***** K6-277 FOLLOWS: THE UNIT ___ BOUNDARY RECORD CONTAINS MORE THAN ONE POSITIVE GEOMETRY RECORD LABEL

This KENO-VI message from subroutine SGGM_JOMCHK indicates that the unit reported has multiple geometry records defining the boundary. This unit is used in an array, so it is only allowed one boundary geometry record.

KMSG278 ***** ERROR ***** K6-278 FOLLOWS: TOO MANY ERRORS TRACING THROUGH THE GEOMETRY WHILE INTEGRATING VOLUMES.

This self-explanatory message is from subroutine SGGM_PATHVOLS, and it indicates that the problem is terminated because of errors. There should be error messages before this indicating what the problems are. Errors should be corrected and the problem resubmitted.

KMSG280 ***** ERROR ***** K6-280 FOLLOWS: ERROR IN PERIODIC BOUNDARY CONDITION - DENOMINATOR 0 A = ___ B = ___ C = ___ D = ___ E = ___ F = ___ G = ___ H = ___ I = ___ J = ___ U = ___ V = ___ W = ___ X = ___ Y = ___ Z = ___

This KENO-VI error message comes from subroutine PERIOD, and it indicates the problem with periodic boundary conditions. A-J are the coefficients of the relevant quadratic equation, U-W are the direction cosines, and X-Z is the point on the boundary. The denominator is $AU^2+BV^2+CW^2+DUV+EUW+FVW$.

KMSG283 ***** ERROR ***** K?-283 FOLLOWS: USER DID NOT PROVIDE A CROSS SECTION DIRECTORY FILE (CE_XXXX). A DIRECTORY FILE THAT CONTAINS THE DATA PATH AND THE CROSS SECTION ID RECORDS MUST BE PROVIDED.

This message from subroutines NUCLIDE_SETUP or PREMIX_SETUP indicates that the filename specified in the library name record is not valid (i.e., the file does not exist in the working directory) for calculations in the continuous energy mode. Links to SCALE-provided libraries are automatically created in the working directory. If one of the default files is being used, the file may not be accessible due to network problems, or access permissions. If it is a user-supplied directory file, it should be verified to exist in the SCALE temporary working directory.

KMSG287 ***** ERROR ***** K?-287 FOLLOWS: OUTGOING ANGLE COSINE ___ IS GREATER THAN 1

This message from subroutine ROTASZ indicates a problem with the angle cosine when calculating the particle velocity in the continuous energy mode. This error usually indicates a coding error. Please contact scalehelp@ornl.gov.

KMSG288 ***** ERROR ***** K?-288 FOLLOWS: SAMPLING PROBABILITY TABLE FOR NUCLIDE:___ ENERGY (EV): ___

This message from subroutine CETRANSPORT_TOTAL_XS_FROM_PTABLE is issued if the energy of the particle that is being tracked is higher than the maximum energy of the probabilities available in the probability table that is being sampled. This error may be due to data problems. Please contact scalehelp@ornl.gov.

KMSG290 ***** WARNING ***** K?-290 FOLLOWS: COULD NOT FIND TEMP:___ ON CROSS SECTION FILE. USING CLOSEST TEMPERATURE:___

This warning message from subroutine READ_XSEC is issued to indicate that the temperature selected was not found for the current nuclide and the continuous energy cross sections that are at the closest temperature will be used. This should only be issued with `DBX=0`, so the user has disabled problem-dependent Doppler broadening. See Sect. 8.1.3.3 or Sect. 8.1.7.2.19 for more information.

KMSG292 ***** ERROR ***** K?-292 FOLLOWS: PROBLEM IN SUBROUTINE SAMPLE_COLLISION AT: NEUTRON: ___ IZA: ___ MT: ___ EIN: ___ MINIMUM ENERGY:___ MAXIMUM ENERGY:___ LOCATION:___

This message from subroutine SAMPLE_COLLISION indicates the energy of the particle being tracked is outside the energy range of the selected reaction for the selected nuclide. The code will reset the particle's energy to correspond to one of the boundaries and will continue tracking. If this message is issued, it usually indicates a problem with the continuous energy cross sections. The problem is usually not severe and may be the result of round off. Please contact scalehelp@ornl.gov.

KMSG293 ***** ERROR ***** K?-293 FOLLOWS: PROBLEM IN SUBROUTINE SAMPLE_ENERGY AT: MT: ___ EIN: ___ EOUT: ___ LOCATION:___

This message from subroutines SAMPLE_COLLISION or SAMPLE_BETA indicates the energy of the particle being tracked is outside the energy range of the selected reaction for the selected nuclide. If this message is issued, it usually indicates a problem with the continuous energy cross sections. Please contact scalehelp@ornl.gov.

KMSG294 ***** WARNING ***** K?-294 FOLLOWS: UNIT ___ REGION ___ CONTAINS FISSIONABLE MATERIAL, BUT NO FISSIONS OCCURRED IN IT. THIS MAY MEAN THE PROBLEM WAS INADEQUATELY SAMPLED.

This message from subroutine KEDIT indicates some of the fissile regions are not adequately sampled. Depending on the problem, the calculated keff value may be too small. The problem should be run again with enough histories or start data to ensure that all regions are sampled adequately.

KMSG295 ***** WARNING ***** K?-295 FOLLOWS: THERE WERE ___ COLLISIONS THAT PRODUCED A NEUTRON WITH AN ENERGY OUTSIDE THE GROUP STRUCTURE YOU SELECTED. THESE WERE TALLIED INTO THE APPROPRIATE END GROUP. IF YOU WISH TO SEPARATE THESE NEUTRONS, ADD A GROUP ON EACH END WITH LIMITS CORRESPONDING TO THE CROSS SECTIONS.

This warning message from subroutine SAMPLE_ENERGY indicates the energy boundaries of the groups specified for tallying in the continuous energy mode does not cover the possible energy ranges of all reactions for all nuclides in the problem. Group energy boundaries must be adjusted.

KMSG296 ***** ERROR ***** K?-296 FOLLOWS: IT IS NOT POSSIBLE TO TURN OFF WRITING ARRAYS TO DIRECT ACCESS IN WRTICE.

This error message from subroutine WRITE_ICE indicates that the arrays must be saved in the direct access file.

KMSG297 ***** ERROR ***** K?-297 FOLLOWS: ENERGY BOUNDARIES MUST BE IN DESCENDING ORDER (EV). EB (___) = ___ IS GREATER THAN EB (___) = ___.

This error message from subroutine RDENER indicates the upper energy boundaries specified in the ENERGY data block are not in descending order. The upper energy boundary of the highest energy group (i.e., group 1) must be specified first and then successive energy groups with descending upper energy group boundaries must be specified until all groups are specified.

KMSG298 ***** ERROR ***** K?-298 FOLLOWS: CROSS SECTION FILE FOR NUCLIDE ___ HAS NO 2-D KINEMATICS DATA FOR THERMAL RANGE. THIS IS OK IF FREEGAS TREATMENT IS NOT TURNED OFF. FOR THIS PROBLEM, FREEGAS TREATMENT HAS BEEN TURNED OFF (FRE=NO IN PARAMETER DATA BLOCK). EITHER PROVIDE DATA THAT CONTAINS 2-D KINEMATICS FOR THERMAL RANGE OR TURN ON FREEGAS TREATMENT.

This message from subroutine READ_KINEMATICS indicates that no thermal scattering has been specified for the given nuclide. Data must be provided or free-gas scattering must be allowed.

KMSG299 ***** ERROR ***** K?-299 FOLLOWS: INVALID KEYWORD ___ IN THE ENERGY BOUNDARIES INPUT DATA.

This message from subroutine RDENER says that an invalid keyword was read while reading the energy boundaries for a continuous energy problem. The energy data block must be checked to correct the error. See Sect. 8.1.3.12 for more information.

KMSG300 ***** ERROR ***** K?-300 FOLLOWS: INVALID THERMAL CUTOFF VALUE ___ IN THE PARAMETERS BLOCK. THE CUTOFF VALUE MUST BE GREATER THAN ZERO.

The thermal cutoff value for a continuous energy problem must be greater than zero. The data must be corrected, and the problem rerun.

KMSG304 K?-304 ALTHOUGH THE STANDARD DEVIATION SPECIFIED FOR THE PROBLEM IS ACHIEVED, AT LEAST 50 OR NSK*2=___ ACTIVE GENERATIONS HAVE NOT BEEN RUN. EXECUTION WILL CONTINUE.

This message from GUIDE indicates that although the standard deviation asked for in the input has been achieved, too few generations have been run to get a fair value for the standard deviation, and more generations will be run.

KMSG305 ***** ERROR ***** K6-305 FOLLOWS: MESH FLUXES HAVE BEEN SPECIFIED, BUT THE MESH DOES NOT COMPLETELY COVER THE GEOMETRY. THE POINT X=___ Y=___ Z=___ LIES OUTSIDE THE MESH. THE PROBLEM WILL NOT BE RUN. FIX THE MESH AND RESUBMIT THE CASE.

This self-explanatory message comes from subroutines ACCUMULATETEMPVOLUMES and from PNTVOLS.

KMSG306 ***** ERROR ***** K?-306 FOLLOWS: UNIT ___ HAS BEEN DEFINED ___ TIMES. THIS BEHAVIOR IS NO LONGER ALLOWED AS IT COULD POTENTIALLY MISLEAD THE USERS. NOTE THAT DEFINING UNITS THAT ARE NOT USED IN THE PROBLEM IS STILL ALLOWED. IF DUPLICATE UNITS ARE KEPT FOR CONVENIENCE (E.G., SCOPING CALCULATIONS), THEN SIMPLY USE A UNIT NUMBER THAT IS UNIQUE AND IS NOT UTILIZED IN THE PROBLEM. THE PROBLEM WILL NOT BE RUN. REMOVE DUPLICATE DEFINITIONS AND RESUBMIT.

This self-explanatory message comes from subroutines K5GM_READGM and SGGM_READGM.

KMSG307 ***** ERROR ***** K?-307 FOLLOWS: INVALID KEYWORD ___ IN THE REACTIONS INPUT DATA.

This message comes from subroutine READ_REACTIONS_INPUT, and indicates that an invalid keyword was entered in the input. See Sect. 8.1.3.15 for valid reaction block input. The data must be corrected, and the problem resubmitted.

KMSG308 ***** ERROR ***** K?-308 FOLLOWS: THE USER ENTERED ___ FOR THE ___ NUMBER. ONLY LEGAL NUMBERS (>0) AND * (WILDCARD) ARE ALLOWED. THE PROBLEM WILL NOT BE RUN.

This message results from subroutine READENTRY, and it indicates invalid reaction input. See Sect. 8.1.3.15 for valid reaction block input; data must be corrected and resubmitted.

KMSG309 ***** ERROR ***** K?-309 FOLLOWS: REACTION TALLY CALCULATIONS ARE CURRENTLY ALLOWED FOR ONLY CE TRANSPORT.

This self-explanatory message printed by subroutine READ_REACTIONS_INPUT indicates that reaction tally calculations were requested when running KENO with multigroup transport mode. Currently, this capability is only available with continuous-energy transport mode.

KMSG312 ***** ERROR ***** K?-312 FOLLOWS: OBSOLETE PARAMETER (CXM). USE REACTION DATA BLOCK TO SPECIFY PARAMETERS FOR REACTION TALLY CALCULATIONS.

This self-explanatory message printed by subroutine PARAM indicates that `CXM` is no longer supported as a *user parameter*.

KMSG313 ***** ERROR ***** K?-313 FOLLOWS: INVALID COMPUTATIONAL MODE (CXM=___) FOR TALLYING REACTION CROSS SECTIONS FOR CE-DEPLETION. AVAILABLE MODES ARE: (1) ALL REACITONS, NGP GROUP, (2) TRANSMUATION REACTIONS, NGP GROUP, (3) ALL REACTIONS, 1 GROUP, (4) TRANSMUTATION REACTIONS, 1 GROUP.

This self-explanatory message printed by subroutine TESTREACTIONFLAGS indicates that invalid `CXM` parameter was provided by the sequence that runs KENO. This rare case might be encountered if the KENO restart file is corrupted. This error should be reported to scalehelp@ornl.gov.

KMSG314 ***** ERROR ***** K?-314 FOLLOWS: NO MESH FOUND FOR ___. A MESH GRID MUST BE DEFINED EITHER WITH THE PARAMETER MSH IN PARAMETER INPUT OR WITH A GRID SPECIFIED IN GRIDGEOMETRY DATA INPUT.

This self-explanatory message printed by subroutine ASSIGNGRIDTOMESHQUANTITY indicates that no mesh was entered with any input methods. The mesh quantity was activated with either `CDS=yes`, `GFX=yes`, `FIS=yes`, or `MFX=yes` parameters, but a mesh was not defined either with parameter `MSH` in the `parameter data` block or in the `gridgeometry data` block. The user should define a mesh using either input method for the tally requested and rerun the calculation.

KMSG315 ***** ERROR ***** K?-315 FOLLOWS: MESH < ___ > SPECIFIED FOR ___ IS NOT FOUND IN INPUT.

This self-explanatory message is printed by subroutine ASSIGNGRIDTOMESHQUANTITY. It indicates that the mesh assigned to the mesh quantity activated with either `CDS`, `GFX`, `FIS`, or `MFX` parameters was not entered in the `gridgeometry data` block. The user should either correct the mesh assignment or define a mesh for the tally requested, and then rerun the calculation.

KMSG316 ***** WARNING ***** K?-316 FOLLOWS: NO MESH PROVIDED FOR SOURCE CONVERGENCE DIAGNOSTICS (SCD). CONTINUE WITH DEFAULT MESH.

This self-explanatory message is printed by subroutine SETUPMESHPOINTERS if no mesh was assigned to Shannon entropy tally used for fission source convergence diagnostics tests with parameter `SCD` in the `parameter data` input block. KENO constructs a DEFAULT mesh and tally fission points on this mesh for Shannon entropy calculations.

KMSG319 ***** WARNING ***** K?-319 FOLLOWS: CFP WAS EITHER NOT SPECIFIED OR A NON-POSITIVE VALUE WAS ENTERED! THE CE TSUNAMI IFP METHOD REQUIRES A POSITIVE NUMBER OF LATENT GENERATIONS. CFP=5 WILL BE USED FOR THIS CALCULATION.

KMSG320 ***** WARNING ***** K?-320 FOLLOWS: CFP>10 DETECTED! THIS CE TSUNAMI IFP RUN MAY PRODUCE A LARGE MEMORY FOOTPRINT!

KMSG321 ***** WARNING ***** K?-321 FOLLOWS: CFP<0 DETECTED! AN F*(R) MESH WILL NOT BE USED FOR THIS CALCULATION. THIS CE TSUNAMI CLUTCH CALCULATION MAY NOT PRODUCE ACCURATE SENSITIVITIES.

KMSG322 ***** ERROR ***** K?-322 FOLLOWS: CFP IS GREATER THAN THE NUMBER OF ACTIVE GENERATIONS! CE TSUNAMI IFP SENSITIVITY TALLIES WILL NOT BE PERFORMED. CFP SHOULD BE << THE NUMBER OF ACTIVE GENERATIONS.

KMSG323 ***** ERROR ***** K?-323 FOLLOWS: CFP IS GREATER THAN THE NUMBER OF INACTIVE GENERATIONS! CE TSUNAMI F*(R) TALLIES CANNOT BE PERFORMED. CFP SHOULD BE << THE NUMBER OF INACTIVE GENERATIONS.

KMSG324 ***** ERROR ***** K?-324 FOLLOWS: NO F*(R) MESH GRID WAS SPECIFIED!

KMSG325 ***** WARNING ***** K?-325 FOLLOWS: ONLY ___ PARTICLE HISTORIES ARE BEING USED TO POPULATE F*(R) TALLIES PER F*(R) MESH INTERVAL. THIS MAY LEAD TO POORLY-CONVERGED F*(R) ESTIMATES AND INACCURATE SENSITIVITY COEFFICIENTS! AT LEAST 10-100 INACTIVE PARTICLE HISTORIES SHOULD BE SIMULATED PER MESH INTERVAL TO OBTAIN ACCURATE F*(R) SCORES.

KMSG326 ***** ERROR ***** K?-326 FOLLOWS: THE CE TSUNAMI IFP METHOD CANNOT BE EXECUTED IN PARALLEL.

This self-explanatory message is printed by subroutine INITIALIZE_SENSITIVITIES if CE TSUNAMI calculation with IFP sensitivity method was run in parallel. Execution is immediately terminated since MPI parallelism is not available with IFP sensitivity method.

KMSG327 ***** WARNING ***** K?-327 FOLLOWS: USER SET MAP-TO-UNION (M2U) TO ON. DOPPLER BROADENING IS ON BY DEFAULT (DBX=2), BUT CANNOT FUNCTION WITH MAP-TO-UNION. DOPPLER BROADENING HAS BEEN DISABLED.

This self-explanatory message is printed by subroutine PARAM if `M2U=yes` and `DBX=2` were set together. Note that `DBX` parameter is defaulted to 2; therefore, entering only `M2U=yes` also satisfies the condition that controls emitting this message. For such a case, KENO disables the Doppler Broadening temperature correction calculations when loading the nuclide cross section data in continuous-energy mode and continues the calculation. If data at the actual problem temperature are required in the calculations for a more realistic modeling, then it is recommended that the map-to-union capability be disabled by entering `M2U=no`.

KMSG328 ***** ERROR ***** K?-328 FOLLOWS: NO SOURCE ENTROPY CALCULATIONS! ALL FISSION SOURCE POINTS ARE OUTSIDE THE USER-DEFINED MESH.

This self-explanatory message is printed by subroutine MESHSRCTEST if no fission points were tallied on the user-defined mesh for three consecutive generations. This results in 0.0 entropy values for these generations. These results indicate that user-defined mesh covers only a part of geometry; i.e., the fissile regions are not covered properly with the user-defined mesh. Therefore, execution is terminated with this error message. To obtain better diagnostics information for the fission source convergence, it is highly recommended to define a mesh grid for Shannon entropy tally, and to be sure that defined mesh grid covers all fissile regions.

KMSG329 ***** ERROR ***** K?-329 FOLLOWS: NO SOURCE ENTROPY CALCULATIONS! ALL FISSION SOURCE POINTS ARE SCORED IN THE SAME MESH VOXEL. USER-DEFINED MESH FOR SOURCE CONVERGENCE DIAGNOSTICS DOES NOT ENCLOSE THE FISSILE REGIONS PROPERLY.

This self-explanatory message is printed by subroutine MESHSRCTEST if all fission points were tallied in a single mesh voxel for three consecutive generations. This results in 0.0 entropy values for these generations. These results indicate that the user-defined mesh does not enclose the fissile regions properly. Therefore, execution is terminated with this error message. To obtain better diagnostics information for the fission source convergence, it is highly recommended to define a mesh grid for Shannon entropy tally, and to be sure that defined mesh grid covers all fissile regions.

KMSG330 ***** WARNING ***** K?-330 FOLLOWS: DISABLING SOURCE ENTROPY CALCULATIONS. ALL FISSION SOURCE POINTS ARE SCORED IN THE SAME MESH VOXEL. THE DEFAULT MESH FOR SOURCE CONVERGENCE DIAGNOSTICS DOES NOT ENCLOSE THE FISSILE REGIONS PROPERLY.

This self-explanatory message from subroutine MESHSRCTEST indicates that for three consecutive generations, all fission points were tallied in a single mesh voxel; therefore, entropy for these three generations was calculated as 0.0. For such a case, KENO disables the Shannon entropy calculations and source convergence diagnostics tests and continues calculation.

The default mesh constructed by KENO ideally covers the geometry completely. This result implies that either all fissile regions are confined in a single mesh voxel (i.e., fissile regions are relatively small compared with the entire problem geometry), or fissile regions are weakly coupled, and source neutrons have been started in one side. To obtain better diagnostics information for the fission source convergence, it is highly recommended to define a mesh grid for Shannon entropy tally, and to be sure that defined mesh grid covers the entire fissile regions.

KMSG331 ***** WARNING ***** K?-331 FOLLOWS: ALL FISSION SOURCE POINTS ARE OUTSIDE OF THE USER-DEFINED MESH.

This self-explanatory message is printed by subroutine MESHSRCTEST if none of the fission points have been tallied on the user-defined mesh for the current generation. This indicates that user-defined mesh does not cover the problem geometry completely (especially all fissile regions), and Shannon entropy for the current generation is calculated as 0.0. Although the same message may not be printed for other generations if there are some fission points scored on this mesh, this does not change the fact that the mesh was not defined properly for Shannon entropy calculations. This will result in incorrect interpretation of the fission source convergence tests.

KMSG332 ***** WARNING ***** K?-332 FOLLOWS: MESH FLUXES HAVE BEEN SPECIFIED, BUT CE TSUNAMI DOES NOT ALLOW MESH FLUX USAGE. CONTINUE CALCULATIONS WITH DISABLING MESH FLUXES.

KMSG333 ***** ERROR ***** K?-333 FOLLOWS: THE NUMBER OF GENERATIONS SKIPPED MUST BE LESS THAN THE NUMBER OF GENERATIONS TO BE RUN. NSK=___ - GEN=___. THE CASE WILL NOT BE RUN.

This self-explanatory message from subroutine PARAM indicates that the number of generations to be run (`GEN`) is always greater than the number of generations skipped (`NSK`). If only `GEN` is entered in the parameter data block, then its value must be greater than the default `NSK` value, which is 3. If only `NSK` is entered in the parameter data block, then its value must be less than the default `GEN` value, which is 203.

KMSG334 ***** ERROR ***** K?-334 FOLLOWS: UNABLE TO CREATE A WEIGHTED LIBRARY. CODE COULD NOT INITIALIZE AMPX WORKING LIBRARY TO WRITE REACTION CROSS SECTIONS IN CE DEPLETION CALCULATIONS. CE DEPLETION CALCULATIONS WILL BE TERMINATED.

Internal code error when running KENO as part of TRITON sequence. This error should be reported to scalehelp@ornl.gov.

KMSG335 ***** ERROR ***** K?-335 FOLLOWS: UNABLE TO ADD NUCLIDE ___ TO THE WEIGHTED LIBRARY. CE DEPLETION CALCULATIONS WILL BE TERMINATED.

Internal code error when running KENO as part of TRITON sequence. This error should be reported to scalehelp@ornl.gov.

KMSG336 ***** ERROR ***** K?-336 FOLLOWS: UNABLE TO GET NEUTRON 1D CROSS SECTION FOR THE REACTION, MT=___. CE DEPLETION CALCULATIONS WILL BE TERMINATED.

Internal code error when running KENO as part of TRITON sequence. This error should be reported to scalehelp@ornl.gov.

KMSG337 ***** ERROR ***** K?-337 FOLLOWS: UNABLE TO CREATE A NEUTRON 1D CROSS SECTION FOR THE WEIGHTING LIBRARY. CE DEPLETION CALCULATIONS WILL BE TERMINATED.

Internal code error when running KENO as part of TRITON sequence. This error should be reported to scalehelp@ornl.gov.

KMSG338 ***** ERROR ***** K?-338 FOLLOWS: ERROR OCCURED WHILE WRITING WEIGHTED LIBRARY, RETURN CODE=___. CE DEPLETION CALCULATIONS WILL BE TERMINATED.

Internal code error when running KENO as part of TRITON sequence. This error should be reported to scalehelp@ornl.gov.

KMSG339 ***** ERROR ***** K?-339 FOLLOWS: UNABLE TO INITIALIZE OBJECT FOR ENERGY BOUNDS FOR THE WEIGHTING LIBRARY. CE DEPLETION CALCULATIONS WILL BE TERMINATED.

Internal code error when running KENO as part of TRITON sequence. This error should be reported to scalehelp@ornl.gov.

KMSG340 ***** ERROR ***** K?-340 FOLLOWS: UNABLE TO INITIALIZE HEADER OBJECT FOR THE WEIGHTING LIBRARY. CE DEPLETION CALCULATIONS WILL BE TERMINATED.

Internal code error when running KENO as part of TRITON sequence. This error should be reported to scalehelp@ornl.gov.

KMSG341 ***** ERROR ***** K?-341 FOLLOWS: DATA ENTERED TO THE REACTION BLOCK WITHOUT CALCULATION MODE. CALCULATION MODE SHOULD BE ENTERED IN REACTION BLOCK WITH ONE OF THE FOLLOWING KEYWORDS: (1) XS/XSTALLY=YES, (2) RX/RRTALLY=YES, (3) CEDEPL=YES, AND/OR, (4) MIXFLX=YES.

This continuous-energy–specific self-explanatory message is printed by subroutine READ_REACTIOS_INPUT if the data entered in `reaction data` block do not include any information for the calculation mode. The valid keywords to set the calculations modes are `XS` or `XSTALLY` for micro reaction cross section calculations, `RX` or `RRTALLY` for reaction tally calculations, `CEDEPL` for both mixture flux and reaction cross section calculations, and `MIXFLX` for mixture flux calculations. At least one of the listed calculation modes above must be entered for a successful calculation.

KMSG342 ***** WARNING ***** K?-342 FOLLOWS: REACTION BLOCK IS CURRENTLY ALLOWED FOR ONLY CE TRANSPORT. SKIP READING DATA FROM REACTION BLOCK.

This self-explanatory message is printed by subroutine READ_REACTIOS_INPUT if a `reaction data` block was entered in a multigroup problem. The `reaction data` block will not function when KENO is run in multigroup mode. Use KMART5 or KMART6 to generate reaction rates when running KENO in multigroup mode.

KMSG346 ***** ERROR ***** K?-346 FOLLOWS: ERROR IN MIXTURE ENTRY IN REACTION BLOCK. MIXTURE NUMBER (MIX=___) SPECIFIED IN REACTION BLOCK IS NOT USED IN THE PROBLEM.

This self-explanatory message is printed by CHECKDATA procedure of REACTIONFILTERS type if the user set up a reaction filter for a non-existing MIXTURE in the problem.

KMSG347 ***** ERROR ***** K?-347 FOLLOWS: ERROR IN NUCLIDE ENTRY IN REACTION BLOCK. NUCLIDE ___[MIXTURE = ___] SPECIFIED IN REACTION BLOCK IS NOT FOUND IN THE MIXING TABLE.

This self-explanatory message is printed by CHECKDATA procedure of REACTIONFILTERS if the user set up a reaction filter with a NUCLIDE and MIXTURE pair where the MIXTURE does not contain this NUCLIDE. Defining a reaction filter with non-existing nuclide results in a code termination.

KMSG348 ***** WARNING ***** K?-348 FOLLOWS: SPECIFIED REACTION IS NOT AVAILABLE IN CE DATA LIBRARY FOR THE GIVEN NUCLIDE. PERFORM REACTION TALLY CALCULATIONS BY SKIPPING THIS REACTION MT=___ FOR THE GIVEN NUCLIDE, ___.

This message is printed by CHECKDATA procedure of REACTIONFILTERS if the nuclide reaction data requested by the NUCLIDE and MT pairs have not been found in the continuous-energy data library. Note that KENO loads 1D neutron reaction data for all nuclides defined in the problem (MT < 150). This message implies that these reaction data are not a valid reaction for the given nuclide; therefore, it is not provided as part of ENDF data, and it is also not included in the AMPX processed continuous-energy data libraries. If data for this nuclide reaction exist in ENDF and the nuclide cross section file being used is a standard file that is part of an official and current SCALE system, then potentially please contact scalehelp@ornl.gov.

KMSG349 ***** WARNING ***** K?-349 FOLLOWS: A CE TSUNAMI F*(R) MESH PRINT (FST=YES) WAS REQUESTED BUT AN F*(R) MESH IS NOT BEING USED.

KMSG350 ***** ERROR ***** K?-350 FOLLOWS: A FISSION NEUTRON WAS BORN OUTSIDE OF THE F*(R) MESH AT POSITION: X =___ Y =___ Z =___ THE F*(R) MESH MUST ENCOMPASS ALL FISSILE MATERIAL IN THE SYSTEM.

KMSG351 ***** ERROR ***** K?-351 FOLLOWS: MISSING REACTION MT SPECIFICATION IN REACTION BLOCK (MT).

This message is printed by subroutine READ_REACTIONS_INPUT for only continuous-energy calculations. It indicates that reaction MT was not defined in one of the reactions filter entries. See Sect. 8.1.3.15 for valid reaction block input. The data must be corrected, and the problem resubmitted.

KMSG352 ***** ERROR ***** K?-352 FOLLOWS: MISSING NUCLIDE SPECIFICATION IN REACTION BLOCK (NUC).

This message is printed by subroutine READ_REACTIONS_INPUT for only continuous-energy calculations. It indicates that nuclide(s) were not defined in one of the reactions filter entries. See Sect. 8.1.3.15 for valid reaction block input. The data must be corrected, and the problem resubmitted.

KMSG353 ***** ERROR ***** K?-353 FOLLOWS: MISSING MIXTURE SPECIFICATION IN REACTION BLOCK (MIX).

This message is printed by subroutine READ_REACTIONS_INPUT for only continuous-energy calculations. It indicates that mixture(s) were not defined in one of the reactions filter entries. See Sect. 8.1.3.15 for valid reaction block input. The data must be corrected, and the problem resubmitted.

KMSG354 ***** ERROR ***** K?-354 FOLLOWS: ERROR IN REACTION BLOCK. NUMBER OF ENERGY GROUPS (___) AND ENERGY BOUNDS (___) DO NOT AGREE. NUMBER OF ENERGY GROUPS ____ = ___, NUMBER OF ENERGY INTERVAL = ___.

This continuous-energy mode–specific message from subroutine READ_REACTIONS_INPUT indicates that either `ngp_xs` and entries in `ener_xs`, or `ngp_flx` and entries in `ener_flx` are not consistent.

KMSG355 ***** ERROR ***** K?-355 FOLLOWS: CE TSUNAMI GPT SYSTEMRESPONSES MAY ONLY CONTAIN ONE NUMERATOR RESPONSE.

KMSG356 ***** ERROR ***** K?-356 FOLLOWS: CE TSUNAMI GPT SYSTEMRESPONSES MAY ONLY CONTAIN ONE DENOMINATOR RESPONSE.

KMSG357 ***** ERROR ***** K?-357 FOLLOWS: DEFINITION NUMBER ___ IN SYSTEMRESPONSE ___ WAS NOT FOUND.

KMSG358 ***** ERROR ***** K?-358 FOLLOWS: NO GPT DEFINITIONS WERE READ FOR THIS CE TSUNAMI GPT CALCULATION.

KMSG359 ***** ERROR ***** K?-359 FOLLOWS: NO GPT SYSTEMRESPONSES WERE READ FOR THIS CE TSUNAMI GPT CALCULATION.

KMSG360 ***** ERROR ***** K?-360 FOLLOWS: DEFINITION NUMBER ___ CONTAINS AN EHIGH THAT IS NOT LESS THAN ITS ELOW.

KMSG361 ***** WARNING ***** K?-361 FOLLOWS: CFP WAS NOT SPECIFIED! FOR THE BEST RESULTS, THE CE TSUNAMI CLUTCH F*(R) METHOD REQUIRES A POSITIVE NUMBER OF LATENT GENERATIONS. CFP=5 WILL BE USED FOR THIS CALCULATION.

KMSG362 ***** WARNING ***** K?-362 FOLLOWS: TEMPERATURE:___ IS WITHIN 4 K OF CLOSEST LIBRARY TEMPERATURE. USING REFERENCE TEMPERATURE: ___.

This message from subroutine READ_XSEC indicates that when Doppler Broadening temperature correction is applied, the nuclide cross section data are loaded at the library reference temperature, which is within 4 K of the given temperature.

KMSG363 ***** WARNING ***** K?-363 FOLLOWS: TEMPERATURE:___ IS ___ REFERENCE LIBRARY TEMPERATURE. USING REFERENCE TEMPERATURE:___.

This message from subroutine READ_XSEC indicates that when Doppler Broadening temperature correction is applied, (1) minimum library temperature is always used if the given temperature is less than this minimum, or (2) maximum library temperature is used if the given temperature is greater than this maximum.

KMSG364 ***** WARNING ***** K?-364 FOLLOWS: AN ENERGY GRID WAS ENTERED THAT DOES NOT COVER ALL POSSIBLE NEUTRON ENERGIES, AS IS REQUIRED BY CE TSUNAMI-3D. THE MIN AND MAX BOUNDS OF THIS ENERGY GRID WERE AUTOMATICALLY EXTENDED TO ___ EV AND ___ EV.

KMSG365 ***** WARNING ***** K?-365 FOLLOWS: DEFINITION NUMBER ___ REFERENCES A REACTION THAT IS NOT SUPPORTED BY EHIGHTRANSFER OR ELOWTRANSFER.

KMSG366 ***** ERROR ***** K?-366 FOLLOWS: DEFINITION NUMBER ___ CONTAINS AN EHIGHTRANSFER THAT IS NOT LESS THAN ITS ELOWTRANSFER.

KMSG367 ***** ERROR ***** K?-367 FOLLOWS: DBRC PARAMETER (DBR=___) IS NONZERO BUT COULD NOT FIND DBRC DATA IN CE DATA LIBRARY FOR ANY NUCLIDE.

This message is printed by subroutine NUCLIDE_SETUP if no DBRC data are available in the given continuous-energy data library. Access permissions, network problems, and typographical errors in the input file should be verified. If the nuclide cross section file being used is a standard file that is part of an official and current SCALE system, then please contact scalehelp@ornl.gov.

KMSG368 ***** ERROR ***** K?-368 FOLLOWS: PARAMETER, THC, IS NO LONGER SUPPORTED IN THE ENERGY DATA BLOCK. PLEASE MOVE THIS PARAMETER TO THE PARAMETERS BLOCK.

This self-explanatory message is printed by subroutine RDENER if `THC` parameter was entered in the energy data block. This parameter is no longer supported as part of the energy data block. This issue must be corrected by defining `THC` parameter in the parameter block rather than in the energy block, and then rerun the problem.

KMSG369 ***** WARNING ***** K?-369 FOLLOWS: PARAMETER THC SPECIFIED BUT WILL BE IGNORED SINCE THIS IS NOT A CE CALCULATION.

This self-explanatory message is from subroutine KENOBUILDER, and it notifies that the defined `THC` parameter is not a valid multigroup parameter; therefore, it will be ignored in the multigroup calculations.

KMSG370 ***** WARNING ***** K?-370 FOLLOWS: MESH VOLUME CALCULATION TYPE=___ IS NOT YET AVAILABLE. THE TYPE IS CHANGED TO RANDOM. BE SURE TO VERIFY THAT THE VOLUMES HAVE BEEN CALCULATED ADEQUATELY.

This self-explanatory message is from subroutine RDVOL, and it indicates that mesh volume calculation is always performed with `RANDOM` estimates. Methods other than `RANDOM` estimates defined in input are always ignored by KENO codes, and the volume calculation method is defaulted to `RANDOM` estimates.

KMSG371 ***** ERROR ***** K6-371 FOLLOWS: KENO-VI GEOMETRY PROCESSOR ALLOWS MAXIMUM SIX CHORD MODIFIER OF ANY TYPE WHEN TRUNCATING A PRE-DEFINED SHAPE. THERE ARE MORE THAN SIX CHORD MODIFIERS ON THE SHAPE, ___ ___, IN UNIT ___.

This self-explanatory KENO-VI message is printed by subroutine SGGM_KENOG if more than the supported number of `CHORD` modifiers were defined on the given shape in the specified unit.

KMSG372 ***** WARNING ***** K?-372 FOLLOWS: ___ POSITIONS HAVE F*(R) MESH TALLIES OF 0.0; A VALUE OF 1.0 WILL BE USED IN THE SENSITIVITY CALCULATIONS.

KMSG373 ***** ERROR ***** K?-373 FOLLOWS: INVALID NGP PARAMETER, (NGP=___). NGP MUST BE A POSITIVE INTEGER.

This self-explanatory continuous-energy mode–specific message from subroutine DATAIN is caused by an invalid **NGP** entry in the parameter data block. The number of energy groups is always a positive integer number.

KMSG374 ***** WARNING ***** K?-374 FOLLOWS: NO ENERGY BOUNDARIES SPECIFIED. DEFAULT ENERGY GROUP BOUNDARIES USED FOR TALLYING WILL BE OBTAINED FROM ___-GROUP SCALE MULTI GROUP LIBRARY.

This self-explanatory message from subroutine DATAIN indicates no energy boundaries were specified with any input method in the continuous-energy calculations. KENO codes always need a set of energy boundaries for tallying. For such a case, KENO codes acquire the energy group boundaries from the 252-group SCALE multigroup library and continue calculation.

KMSG375 ***** WARNING ***** K?-375 FOLLOWS: NGP PARAMETER WILL BE IGNORED SINCE THE ENERGY GROUP BOUNDARIES HAS ALREADY BEEN READ FROM ENERGY DATA BLOCK. NGP VALUE PRINTED IN NUMERIC PARAMETERS EDIT IN THE OUTPUT REFLECTS THIS UPDATE.

This self-explanatory message from subroutine DATAIN indicates that data provided by another input method (energy data block) will override the number of energy group value entered by `NGP` parameter.

KMSG376 ***** WARNING ***** K?-376 FOLLOWS: NGP PARAMETER WILL BE IGNORED SINCE IT IS NOT A VALID PARAMETER FOR MULTI GROUP TRANSPORT.

This self-explanatory message coming from subroutine DATAIN indicates that the multigroup calculation does not process `NGP` parameter that is dedicated for continuous-energy calculations.

KMSG377 ***** WARNING ***** K?-377 FOLLOWS: ENERGY DATA BLOCK IS CURRENTLY ALLOWED FOR ONLY CONTINUOUS ENERGY TRANSPORT. SKIP READING DATA FROM ENERGY DATA BLOCK.

This self-explanatory message from RDENER indicates that the multigroup calculation does not process data in the energy data input block since it is dedicated for continuous-energy calculations.

KMSG378 ***** WARNING ***** K?-378 FOLLOWS: IN MULTI GROUP MODE, DEFAULT ENERGY BOUNDARIES USED FOR TALLYING ARE ALWAYS OBTAINED FROM THE MULTI GROUP LIBRARY USED BY TRANSPORT.

This self-explanatory message from subroutine UPDATE_EBOUNDS_MG_MODE reminds that multigroup mode acquires the energy group structure from the multigroup library used by transport process, and it uses these energy boundaries in all tallies if not otherwise requested.

KMSG379 ***** WARNING ***** K?-379 FOLLOWS: NGP PARAMETER WILL BE IGNORED SINCE ENERGY GROUP BOUNDARIES HAS ALREADY BEEN READ FROM ENERGY DATA BLOCK. NGP VALUE PRINTED IN NUMERIC PARAMETERS EDIT IN THE OUTPUT MAY NOT REFLECT THIS UPDATE. FINAL NGP VALUE IS CORRECTLY PRINTED IN ADDITIONAL INFORMATION EDIT (SHOWN AS NUMBER OF ENERGY GROUPS).

This self-explanatory message from subroutine DATAIN reminds that energy boundaries were read from the energy data block and **NGP** value printed in different output edits might not be updated accordingly. Update methods for data used in the output edits, and/or order of output editing might show some minor differences depending on how KENO is run (e.g., running KENO as stand-alone, running KENO through a SCALE sequence, running KENO through a SCALE sequence with parm=check).

KMSG380 ***** WARNING ***** K6-380 FOLLOWS: CALCULATED VOLUME FRACTION OF FISSILE MATERIAL IN THE SYSTEM REPORTED BELOW MAY NOT REFLECT THE ACTUAL VALUE SINCE SOME REGION VOLUMES WERE NEITHER SPECIFIED NOR CALCULATED. EVEN NEGATIVE VOLUME FRACTION FOR FISSILE MATERIAL MAY BE REPORTED SINCE ALL UNDEFINED VOLUMES WERE SET TO -1.0.

This self-explanatory KENO-VI message is printed by subroutine K6_VOLFIS. It notifies that the fraction of the fissile material might be calculated as negative numbers if there is a missing region volume data. When volumes are not entered, not calculated, or calculated as zero, KENO resets their values to -1.0. Therefore, missing volumes might result in incorrect volume fraction calculation for the fissile materials.

KMSG381 ***** WARNING ***** K?-381 FOLLOWS: IN MULTI GROUP MODE, DEFAULT ENERGY BOUNDARIES USED FOR TALLYING ARE ALWAYS OBTAINED FROM THE MULTI GROUP LIBRARY USED BY TRANSPORT. NGP VALUE PRINTED IN NUMERIC PARAMETERS EDIT IN THE OUTPUT MAY NOT REFLECT THIS. FINAL NGP VALUE IS CORRECTLY PRINTED IN ADDITIONAL INFORMATION EDIT (SHOWN AS NUMBER OF ENERGY GROUPS).

This self-explanatory message is multigroup mode–specific and comes from subroutine DATAIN. It indicates that the default energy group structure used for all tallies are always acquired from the library used by transport process in multigroup calculations. The `NGP` value printed in different output edits might not be updated accordingly. Update methods for data used in the output edits, and/or order of output editing, might show some minor differences depending on how KENO is run (e.g., running KENO as standalone, running KENO through a SCALE sequence, running KENO through a SCALE sequence with parm=check).

KMSG383 ***** WARNING ***** K?-383 FOLLOWS: ENERGY GROUP BOUNDARIES READ FROM ENERGY DATA BLOCK OVERRIDE THE DEFAULT NGP VALUE (252). NGP VALUE PRINTED IN NUMERIC PARAMETERS EDIT IN THE OUTPUT MAY NOT REFLECT THIS UPDATE. THE FINAL NGP VALUE IS CORRECTLY PRINTED IN ADDITIONAL INFORMATION EDIT (SHOWN AS NUMBER OF ENERGY GROUPS).

This self-explanatory message is specific to continuous-energy mode, and it is printed by subroutine DATAIN when running KENO standalone or through a sequence with PARM=CHECK. The `NGP` value printed in different output edits might not be updated accordingly.

KMSG384 ***** ERROR ***** K?-384 FOLLOWS: SPECIFICATION OF ENERGY BOUNDARIES WITH BOTH ENERGY DATA BLOCK AND DEFINITIONS DATA BLOCK IS NOT ALLOWED.

This self-explanatory message is printed by subroutine READ_DEFINITIONS_INPUT when running KENO as part of a SCALE sequence (CSAS, TRITON). It indicates that energy boundaries were specified with both `CSAS definitions data` and `KENO energy data` blocks, and this is illegal.

KMSG385 ***** ERROR ***** K?-385 FOLLOWS: ___ IS NOT ALLOWED TO BE USED TOGETHER WITH THE DATA SPECIFIED IN ___ BLOCK.

This message from subroutine READ_DEFINITIONS_INPUT is caused by using both old and new input definition methods when running KENO as part of either CSAS or TRITON sequence to activate some of the KENO capabilities. For example, specifying the fission rate mesh tally with `FIS` parameter in KENO parameter data block (old method, activate

capability with a single parameter) together with specifying fission source accumulation in `CSAS tallies data` block is not allowed.

KMSG386 ***** ERROR ***** K?-386 FOLLOWS: INCONSISTENT ENERGY BOUNDARIES. THE SIZE OF ENERGY BOUNDARIES SPECIFIED IN DEFINITIONS DATA WITH THE IDENTIFICATION NUMBER, ___, MUST BE LESS THAN THE SIZE OF THE DEFAULT ENERGY BOUNDARIES ACQUIRED FROM THE MULTI GROUP LIBRARY USED BY TRANSPORT, (___ > ___).

This message is specific to multigroup mode and emitted from subroutine VALIDATE_ENERGY_INTERVALS to explain that energy boundaries specified in `CSAS definitions data` block are not consistent with the DEFAULT energy group structure (or energy boundaries) obtained from the library used by transport process. Multiple energy group structures for tallying are supported in KENO, even for multigroup mode. However, in multigroup mode, each group structure must be a subset of the DEAFAULT energy boundaries acquired from the multigroup library used in the neutron transport. Therefore, the size of a group structure defined in `CSAS definition data` block cannot be larger than the DEFAULT energy boundaries.

KMSG387 ***** ERROR ***** K?-387 FOLLOWS: INCONSISTENT ENERGY INTERVALS. ALTHOUGH THEIR SIZE MATCH, ENERGY BOUNDARIES SPECIFIED IN DEFINITIONS DATA WITH IDENTIFICATION NUMBER, ___, DEFINE DIFFERENT ENERGY INTERVALS COMPARED TO THE DEFAULT ENERGY BOUNDARIES ACQUIRED FROM MULTI GROUP LIBRARY USED BY TRANSPORT.

This message is specific to multigroup mode and emitted from subroutine VALIDATE_ENERGY_INTERVALS to explain that energy boundaries specified in `CSAS definitions data` block are not consistent with the DEFAULT energy group structure (or energy boundaries) obtained from the library used by transport process. Multiple energy group structures for tallying are supported in KENO, even for multigroup mode. However, in multigroup mode, each group structure must be a subset of the DEFAULT energy boundaries acquired from the multigroup library used in the neutron transport. Therefore, the energy intervals defined in `CSAS definition data` block must be the same as the DEFAULT energy boundaries if their sizes are equal.

KMSG388 ***** ERROR ***** K?-388 FOLLOWS: INCONSISTENT ENERGY INTERVALS. ENERGY BOUNDARIES SPECIFIED IN DEFINITIONS DATA WITH THE IDENTIFICATION NUMBER ___ MUST BE A SUBSET OF THE DEFAULT ENERGY BOUNDARIES ACQUIRED FROM THE MULTI GROUP LIBRARY USED BY TRANSPORT.

This message is specific to multigroup mode and emitted from subroutine VALIDATE_ENERGY_INTERVALS to indicate that energy boundaries specified in `CSAS definitions data` block are not consistent with the DEFAULT energy group structure (or energy boundaries) obtained from the library used by transport process. Multiple energy group structures for tallying are supported in KENO. However, in multigroup mode, each group structure must be a subset of the DEFAULT energy boundaries acquired from the multigroup library used in the neutron transport.

KMSG389 ***** ERROR ***** K?-389 FOLLOWS: ENERGY BOUNDARIES SPECIFIED IN DEFINITIONS DATA WITH THE IDENTIFICATION NUMBER ___ MUST BE IN DESCENDING ORDER (EV). EB (___) = ___ IS GREATER THAN OR EQUAL TO EB(___) = ___.

This message is emitted from subroutine VALIDATE_ENERGY_INTERVALS to indicate that energy boundaries defined either with **KENO energy data** or **CSAS definitions data** blocks were not in descending order. KENO always expects the energy points entered in descending order.

KMSG390 ***** ERROR ***** K?-390 FOLLOWS: IDENTIFICATION NUMBER ___ WAS USED MORE THAN ONCE WHEN SPECIFYING ENERGY BOUNDARIES IN DEFINITIONS DATA.

This self-explanatory message is printed by eBounds_type's ADD procedure when running KENO as part of CSAS or TRITON sequences. It indicates that identification numbers must be unique entries, and using the same identification number multiple times for different energy boundaries definitions is an illegal operation.

KMSG391 ***** ERROR ***** K?-391 FOLLOWS: MESH TALLY IDENTIFICATION NUMBER ___ WAS USED MORE THAN ONCE WHEN SPECIFYING MESH TALLY IN TALLIES BLOCK.

This self-explanatory message is printed by subroutine TALLIES_TYPE_ADD_MESH_TALLY when running KENO as part of CSAS or TRITON sequences. It indicates that identification numbers must be unique entries and using the same identification number multiple times for different mesh tallies is an illegal operation.

KMSG392 ***** ERROR ***** K?-392 FOLLOWS: MESH TALLIES ___ AND ___ SPECIFIED IN TALLIES BLOCK HAVE IDENTICAL DEFINITIONS.

This self-explanatory message is printed by subroutine TALLIES_TYPE_ADD_MESH_TALLY when running KENO as part of CSAS or TRITON sequences. It indicates that the same mesh tally was defined two times in the tallies data block. This is considered a potential user error: therefore, calculation is terminated.

KMSG393 ***** ERROR ***** K?-393 FOLLOWS: BAD VALUE FOR KEYWORD MESH TALLY, NEEDS AN IDENTIFICATION NUMBER.

This message from subroutine READ_MESH_TALLY_INPUT when running KENO as part of CSAS or TRITON sequences indicates that the mesh tally identification number either was not entered or was defined as a non-integer number.

KMSG394 ***** ERROR ***** K?-394 FOLLOWS: INVALID MESHTALLY IDENTIFIER, ___, IN TALLIES BLOCK. IDENTIFICATION NUMBER MUST BE A POSITIVE INTEGER.

This message from subroutine READ_MESH_TALLY_INPUT when running KENO as part of CSAS or TRITON sequences indicates that mesh tally identification number is not a positive integer number.

KMSG395 ***** ERROR ***** K?-395 FOLLOWS: ___ – ENDED WITH – END ___ –.

This message is printed by subroutine READ_MESH_TALLY_INPUT when running KENO as part of CSAS or TRITON sequences if `READ TALLIES` is not ended with `END TALLIES`, or `READ MESH` is not ended with `END MESH`.

KMSG396 ***** ERROR ***** K?-396 FOLLOWS: INVALID RESPONSE TYPE, ___. ONLY [FLUX, FISSION_DENSITY, FISSION_SOURCE] ARE THE VALID RESPONSE TYPES.

This message printed by subroutine READ_MESH_TALLY_INPUT when running KENO as part of CSAS or TRITON sequences if the mesh tally calculation was requested with an invalid response type.

KMSG397 ***** ERROR ***** K?-397 FOLLOWS: INVALID ENERGY IDENTIFICATION NUMBER (ENERGY= ___) IN MESH TALLY ___ IN TALLIES BLOCK. IDENTIFICATION NUMBER MUST BE A POSITIVE INTEGER OR DEFAULT.

This message printed by subroutine READ_MESH_TALLY_INPUT when running KENO as part of CSAS or TRITON sequences is caused by an invalid identification number that was entered as the energy identifier in tallies data block.

KMSG398 ***** ERROR ***** K?-398 FOLLOWS: SPECIFICATION OF ENERGY BOUNDARIES WITH BOTH NGP PARAMETER AND DEFINITIONS DATA IS NOT ALLOWED.

This self-explanatory message printed by subroutine READ_DEFINITIONS_INPUT when running KENO as part of CSAS or TRITON sequences indicates that energy boundaries cannot be specified by using the old input specification method (`NGP` parameter) and new input specification methods (`energybounds` in CSAS definition data block) together.

KMSG399 ***** ERROR ***** K?-399 FOLLOWS: INVALID ENERGY BOUNDS SPECIFICATIONS WITH IDENTIFICATION NUMBER ___; ALL ENERGY POINTS ARE IDENTICAL.

This self-explanatory message is printed by eBounds_type's ADD procedure when running KENO as part of CSAS or TRITON sequences. It indicates that defined energy boundaries cannot be used for tallying since all energy points were entered as the same.

KMSG400 ***** WARNING ***** K?-400 FOLLOWS: IGNORE READING ENERGY GROUP BOUNDARIES FROM DEFINITIONS DATA THAT ARE MARKED AS DEFAULT SINCE THIS IS A MULTI GROUP CALCULATION.

This self-explanatory message is printed by eBounds_type's ADD procedure when running KENO as part of CSAS or TRITON sequences. It notifies that DEFAULT energy group structure acquired from the library used in the neutron transport cannot be overridden by the energy boundaries data in **CSAS definitions data** block.

KMSG401 ***** WARNING ***** K?-401 FOLLOWS: IDENTICAL ENERGY BOUNDARIES WERE SPECIFIED IN DEFINITIONS DATA WITH IDENTIFICATION NUMBERS ___ AND ___. ONLY ONE OF THEM WILL BE PROCESSED AND STORED FOR THE CALCULATIONS.

This self-explanatory message is printed by eBounds_type's ADD procedure when running KENO as part of CSAS or TRITON sequences. It notifies that two sets of energy boundaries defined in **CSAS definitions data** block have the identical specifications. KENO only processes one of them and stores it in energy data container for tallying. It updates energy bound pointers for the tallies if they are defined to use the energy bounds not stored.

KMSG403 ***** ERROR ***** K?-403 FOLLOWS: ENERGY BOUNDARIES (ENERGY=___) SPECIFIED IN ___ ___ IN TALLIES BLOCK IS NOT FOUND IN THE DEFINITIONS DATA INPUT BLOCK.

This self-explanatory message is printed by VALIDATE_ENERGY_BOUNDS_ID procedure when running KENO as part of CSAS or TRITON sequences. It indicates that the energy boundaries were not entered in **CSAS definitions data** block even though it was intended to be used with a mesh tally calculation defined in **CSAS tallies data** block.

KMSG404 ***** ERROR ***** K?-404 FOLLOWS: SPECIFICATION OF MESH GRIDS WITH BOTH GRIDGEOMETRY DATA INPUT AND GRIDGEOMETRY IN DEFINITIONS DATA IS NOT ALLOWED.

This self-explanatory message is emitted by subroutine READ_DEFINITIONS_INPUT when running KENO as part of a SCALE sequence (CSAS, TRITON). It indicates that mesh grids were specified with both **CSAS definitions data** and **KENO gridgeometry data** blocks, and this is not allowed.

KMSG405 ***** WARNING ***** K?-405 FOLLOWS: WARNING MESSAGES EMITTED BY ___ INPUT PROCESSOR ARE AS FOLLOWS:

KENO and MONACO use the same legacy input parser to process the data entered in **definitions data** block. This common utility has its own messaging method, which accumulates all user error and warning messages encountered while processing data, then forwards them to the caller who runs this utility. Warning messages from this utility are filtered and printed with this message by READ_DEFINITIONS_INPUT to notify user about the possible issues in the input. Note that this message is only emitted when running KENO as part of a SCALE sequence (CSAS, TRITON).

KMSG406 ***** ERROR ***** K?-406 FOLLOWS: GRID IDENTIFICATION NUMBER ___ WAS USED MORE THAN ONCE WHEN SPECIFYING GRIDGEOMETRY.

This self-explanatory message is printed by CHECKGRIDDATA procedure of MGRID type procedure, and it indicates that identification numbers must be unique entries and using the same identification number multiple times for different mesh grid definitions is an illegal operation.

KMSG407 ***** WARNING ***** K?-407 FOLLOWS: IDENTICAL GRIDGEOMETRIES WERE SPECIFIED WITH IDENTIFICATION NUMBERS ___ AND ___. ONLY ONE OF THEM WILL BE PROCESSED AND STORED FOR THE CALCULATIONS.

This self-explanatory message is printed by REMOVEDUPLICATES procedure of MGRID type, and it notifies that two sets of grid geometry either defined in **CSAS definitions data** or in **KENO gridgeometry data** blocks have the identical specifications. KENO only processes one of them and stores it in grid data container for tallying. It updates the grid data pointers for the tallies if they are defined to use the mesh grids not stored.

KMSG408 ***** ERROR ***** K?-408 FOLLOWS: GRIDGEOMETRY (GRID=___) SPECIFIED IN ___, ___ IN TALLIES BLOCK IS NOT FOUND IN THE DEFINITIONS DATA INPUT BLOCK.

This self-explanatory message is printed by VALIDATE_GRID_ID procedure when running KENO as part of CSAS or TRITON sequences. It indicates that the mesh grid was not entered in **CSAS definitions data** block even though it was intended to be used with a mesh tally calculation defined in **CSAS tallies data** block.

KMSG409 ***** ERROR ***** K?-409 FOLLOWS: INVALID GRID IDENTIFICATION NUMBER (GRID= ___) IN MESH TALLY ___ IN TALLIES BLOCK. IDENTIFICATION NUMBER MUST BE A POSITIVE INTEGER.

This message comes from subroutine READ_MESH_TALLY_INPUT when running KENO as part of CSAS or TRITON sequences. It indicates that the mesh grid identification number defined in **CSAS tallies data** block is not a positive integer number.

KMSG410 ***** WARNING ***** K?-410 FOLLOWS: ENERGY BOUNDARIES MARKED AS DEFAULT WILL BE USED IN ALL TALLY CALCULATIONS IF OTHERWISE NOT REQUESTED.

This message is printed immediately after the energy boundaries output edit to indicate that the DEFAULT energy boundaries listed in this edit will be used in all tally calculations, unless otherwise requested. Note that the default energy boundaries can be defined with different input data or can be read by the code from the relevant SCALE library depending on the computation mode.

KMSG411 ***** WARNING ***** K?-411 FOLLOWS: ENERGY GROUP BOUNDARIES READ FROM ENERGY BLOCK OVERRIDE THE DEFAULT NGP VALUE.

This self-explanatory message printed by subroutine DATAIN indicates that data read from energy data block will override the default `NGP` value.

KMSG412 ***** WARNING ***** K?-412 FOLLOWS: NO ENERGY BOUNDARIES SPECIFIED AS DEFAULT IN DEFINITIONS DATA. THE DEFAULT ENERGY GROUP BOUNDARIES USED FOR TALLYING ARE OBTAINED FROM ___-GROUP SCALE MULTI GROUP LIBRARY.

This self-explanatory message printed by subroutine DATAIN when running KENO as part of CSAS or TRITON sequences indicates that **CSAS definition data** block does not have a definition for the DEFAULT energy boundaries. For this continuous-energy–specific case, KENO obtains the DEFAULT energy boundaries from the SCALE 252-group multigroup library and utilizes it for tallying.

KMSG413 ***** WARNING ***** K?-413 FOLLOWS: DEFAULT ENERGY GROUP BOUNDARIES READ FROM DEFINITIONS DATA OVERRIDE THE DEFAULT NGP VALUE. NGP VALUE PRINTED IN NUMERIC PARAMETERS EDIT IN THE OUTPUT MAY NOT REFLECT THIS UPDATE. THE FINAL NGP VALUE IS CORRECTLY PRINTED IN ADDITIONAL INFORMATION EDIT (SHOWN AS NUMBER OF ENERGY GROUPS).

This self-explanatory message is specific to continuous-energy mode, and it is printed by subroutine DATAIN when running KENO through a sequence with PARM=CHECK. This message reminds that the DEFAULT NGP value will be overridden by this data read from the definitions data block, and the updated NGP value can be shown in Additional information output edit.

KMSG414 ***** WARNING ***** K?-414 FOLLOWS: DEFAULT ENERGY GROUP BOUNDARIES READ FROM DEFINITIONS DATA OVERRIDE THE DEFAULT NGP VALUE. NGP VALUE PRINTED IN NUMERIC PARAMETERS EDIT IN THIS OUTPUT REFLECTS THIS UPDATE.

This self-explanatory message is specific to continuous-energy mode, and it is printed by subroutine DATAIN when running KENO through a sequence. This message indicates that the DEFAULT NGP value printed in numeric parameters output edit was also updated accordingly.

KMSG415 ***** WARNING ***** K?-415 FOLLOWS: SPECIFIED NGP VALUE (NGP=___) DOES NOT CORRESPOND TO ONE OF THE STANDARD ENERGY GROUP STRUCTURES AVAILABLE. CONSTRUCT ___ EQUAL LETHARGY BINS AND SET THEM AS DEFAULT ENERGY GROUP BOUNDARIES WHICH WILL BE USED FOR TALLYING.

This self-explanatory message is specific to continuous-energy mode, and it is printed by subroutine SETUP_DEFAULT_EBOUNDS_CE_MODE. It indicates that specified `NGP` parameter does not match any SCALE group structure; therefore, KENO constructs the DEFAULT energy boundaries with `NGP` equal lethargy bins and stores this in the ebounds data container. The DEFAULT energy boundaries will be used for all tallies.

KMSG416 ***** ERROR ***** K6-416 FOLLOWS: BOUNDARY CONDITIONS OTHER THAN VACUUM IS NOT ALLOWED ON THE BOUNDARIES WHEN THE BOUNDARY DEFINITION VECTOR OF THE GLOBAL UNIT HAS NEGATIVE BODY LABEL(S).

This self-explanatory KENO-VI message from subroutine K6_RDBNDS indicates that a non-vacuum albedo boundary condition was defined illegally. KENO-VI allows constructing global unit with combinations of multiple bodies whose labels listed in the boundary definition vector of the global unit. However, only VACUUM albedo boundary condition is to be applied to the global boundaries if the boundary definition vector of the global unit has negative body label(s). Therefore, execution is terminated with this error message. See Sect. 8.1.3.6 for specific examples for albedo boundary condition specifications for KENO-VI geometry.

KMSG417 ***** ERROR ***** K6-417 FOLLOWS: BOUNDARY CONDITION SPECIFICATION WITH SURFACE AND FACE CODES WITHOUT CORRESPONDING BODY IS NOT ALLOWED WHEN MULTIPLE BODIES USED TO DEFINE THE GLOBAL UNIT BOUNDARY.

This self-explanatory KENO-VI message from subroutine K6_RDBNDS is caused by a missing `BODY` parameter in the boundary condition specification. When multiple bodies define the global boundaries, boundary conditions of each body should be defined with `BODY` parameter following by the corresponding face codes. Therefore, the execution is terminated with this error message. See Sect. 8.1.3.6 for specific examples for albedo boundary condition specifications for KENO-VI geometry.

KMSG418 ***** ERROR ***** K?-418 FOLLOWS: FACE CODES AND SURFACE KEYWORDS ARE NOT ALLOWED TO BE USED TOGETHER IN BOUNDARY DATA SPECIFICATION FOR THE SAME BODY (BODY=___).

This self-explanatory message from subroutines K6_RDBNDS or K5_RBNDS indicate that there is illegal input data in bounds data block. KENO does not allow using `SURFACE` and cuboidal `FACE CODE` s together for albedo boundary condition specifications.

KMSG419 ***** ERROR ***** K5-419 FOLLOWS: FACE CODES ARE NOT ALLOWED TO BE USED WITH NON-CUBOIDAL BOUNDING SHAPE WHICH IS A ___.

This self-explanatory message from subroutine K5_JOMCHK indicates that `FACE CODE` s were used illegally when specifying albedo boundary conditions. Cuboidal `FACE CODE` s cannot be used for non-cuboidal boundary shape when defining boundary conditions.

KMSG420 ***** ERROR ***** K5-420 FOLLOWS: ___ BOUNDARY CONDITION IS NOT ALLOWED ON HEMISPHERES AND HEMICYLINDERS.

This message from subroutine K5GM_JOMCHK indicates that non-vacuum boundary conditions were applied to the outermost geometry region, which is a hemisphere or a hemicylinder. KENO V.a geometry does not allow boundary conditions other than VACUUM on these boundary shapes. Execution is terminated.

KMSG421 ***** ERROR ***** K5-421 FOLLOWS: BODY IS AN INVALID PARAMETER NAME IN BOUNDARY CONDITION DATA SPECIFICATION.

This message emitted from subroutine K5GM_RDBNS indicates that an invalid bounds data parameter was used when defining albedo boundary conditions. `BODY` parameter is a KENO-VI specific parameter used to define albedo boundary conditions to the boundary shape(s) which constructs the global unit. Therefore, execution is terminated.

KMSG422 ***** ERROR ***** K6-422 FOLLOWS: FACE CODES ARE NOT ALLOWED TO BE USED WITH NON-CUBOIDAL BOUNDING SHAPE (BOUNDARY BODY ___ IN THE GLOBAL UNIT BOUNDARY IS A ___).

This self-explanatory message emitted from subroutine K6_RDBNDS indicates that FACE CODEs were used illegally when specifying albedo boundary conditions on the non-cuboidal boundary shape(s). Cuboidal FACE CODE s cannot be used with non-cuboidal boundary shape(s) when defining the boundary conditions. Therefore, execution is terminated.

KMSG430 ***** ERROR ***** K?-430 FOLLOWS: ILLEGAL PARAMETER VALUE (___ , ___), IT MUST BE A POSITIVE NUMBER.

This message printed by subroutine RDSTRT is caused by an illegal parameter value in START data block. START data parameters LNU, NXS, NYS, and NZS must be a positive integer value; otherwise, execution is terminated with this error message.

KMSG431 ***** ERROR ***** K?-431 FOLLOWS: LNU IN EACH SUCCESSIVE DATA SET SHOULD BE LARGER THAN THE PREVIOUS VALUE OF LNU (CURRENT LNU=___, PREVIOUS LNU=___).

This message printed by subroutine RDSTRT if an illegal LNU parameter value is in START data block. LNU parameter is used in START type 6 data. This start type allows neutrons to be started at the arbitrary starting points defined by a set of data. The last entry for each start 6 data set must be LNU, and the LNU value of each successive set of data must be larger than the last. Otherwise, execution is terminated with this error message.

KMSG432 ***** WARNING ***** K?-432 FOLLOWS: ONLY ___ STARTING POINT WILL BE READ FROM THE ASCII START DATA FILE (RDU=___).

This message emitted by subroutine RDSTRT indicates that starting points read from the ASCII start data file is less than the number of neutrons per generations. If there are no other start data sets defined in the START data block, KENO uniformly samples the remaining starting points from the set of these arbitrary starting points already read from the ascii start data file. See Sect. 8.1.4.8 for specific examples for start type 6.

KMSG433 ***** ERROR ***** K?-433 FOLLOWS: START TYPE 6 PARAMETER ___ HAS BEEN USED MORE THAN ONCE IN THE SAME DATA SET.

This self-explanatory message from subroutine RDSTRT indicates that one of the START type 6 parameter was defined more than once in the same start type 6 data set. Starting parameters NXS, NYS, NZS, TFX, TFY, and TFZ should be used once in the same start type 6 data set ended by LNU parameter.

KMSG434 ***** ERROR ***** K?-434 FOLLOWS: NO LNU HAS BEEN SPECIFIED WITH THE ___ SET IN START TYPE 6.

This self-explanatory message printed by subroutine RDSTRT is caused by a missing LNU parameter in one of the start type 6 data set which may include either NXS, NYS, and NZS, or TFX, TFY, and TFZ parameters.

KMSG435 ***** ERROR ***** K?-435 FOLLOWS: RDU (RDU=___) CANNOT BE USED TOGETHER WITH ___ IN THE SAME START TYPE 6 DATA SET ENDED WITH (___ , ___).

This self-explanatory message from subroutine RDSTRT is caused by using RDU parameter together with any NXS, NYS, NZS, TFX, TFY, and TFZ options in the same start type 6 data set. RDU cannot be used together with these start type 6 options in the same data set; therefore, execution is terminated.

KMSG436 ***** ERROR ***** K?-436 FOLLOWS: SPECIFIED ASCII START DATA FILE (RDU=___) DOES NOT EXIST.

This self-explanatory message is printed by subroutine RDSTRT if the defined ASCII start data file could not be found. Execution is terminated with this error message for such a case.

KMSG437 ***** ERROR ***** K?-437 FOLLOWS: INSUFFICIENT NUMBER OF START DATA POINTS (SPECIFIED LNU=___ >> MAX. LNU READ FROM RDU FILE, LNU=___) FOUND IN SPECIFIED ASCII START DATA FILE (RDU=___).

This self-explanatory message is printed by subroutine RDSTRT if the ASCII start data file contains starting points less than the defined LNU value. There must be at least LNU number of starting points in the ASCII start data file for a successful operation. See Sect. 8.1.4.8 for specific examples for start type 6.

KMSG438 ***** WARNING ***** K?-438 FOLLOWS: ___ SEQUENCE DOES NOT SUPPORT PARAMETER (___) SPECIFIED IN THE PARAMETER DATA. THIS PARAMETER WILL BE IGNORED IN THIS CALCULATION.

This self-explanatory message is printed by subroutine PARAM if an unsupported parameter is entered in the **parameter data** input block. KENO by design has several parameters, some of them dedicated to the sequence running KENO. For example, parameters CET, CFP, etc. are allocated for sensitivity calculations performed by KENO when running KENO as part of a TSUNAMI sequence. These parameters are not valid for other sequences running KENO such as CSAS and TRITON. If there is an inconsistency between such sequence-specific parameters and the sequence, which is running KENO, then execution is terminated with this error message to prevent excess resource allocation for the sequence-specific capabilities activated by these parameters.

KMSG439 ***** ERROR ***** K?-439 FOLLOWS: ILLEGAL VALUE FOR START TYPE 6 PARAMETER (___=___). ___ CANNOT BE GREATER THAN THE NUMBER OF UNITS IN THE ___-DIRECTION OF THE GLOBAL ARRAY (___=___).

This message from subroutine K_STDATA indicates that the values entered with the NXS, NYS, and NZS start type 6 parameters are not consistent with the definition of the global array. Valid NXS, NYS, and NZS values must be less than or equal to the NUX, NUY, and NUZ entries in the global array definitions in the `array data` block. Otherwise, execution is terminated with this error message.

KMSG440 ***** WARNING ***** K?-440 FOLLOWS: ___ IS NOT A VALID DATA FOR START TYPE ___. THIS PARAMETER WILL BE IGNORED IN THIS CALCULATION.

This self-explanatory message is printed by subroutine RDSTRT if a start type parameter is defined with another start type which does not support this parameter. For example, LNU parameter is a start type 6-specific parameter, and it is not allowed with start type 0. For such a case, execution is terminated with this error message.

KMSG441 ***** WARNING ***** K?-441 FOLLOWS: ___ IS NO LONGER SUPPORTED BY START DATA (OBSOLETE PARAMETER). THIS PARAMETER WILL BE IGNORED IN THIS CALCULATION.

This self-explanatory message from subroutine RDSTRT indicates that an obsolete parameter was used in the **start data** input block. KENO development is an evolving process; sometimes due to new limitations as part of modernization or other needs, some legacy parameters are no longer supported. KENO ignores processing this parameter and continues the calculation by emitting this warning message.

KMSG442 ***** INFORMATION WARNING ***** K6-442 FOLLOWS: POSIT WARNING — POINT MAY NOT BE IN GEOMETRY X = ___ Y = ___ Z = ___

This message from subroutine K6_START indicates that starting point sampled/entered might be outside the geometry. Usually, this warning message is followed by an error message if the starting point is verified that it is not located inside the geometry by subroutine SGGM_POSIT. If the number of points outside the geometry is less than 5, then calculation will continue by resampling these points from the set of successful starting points (which are inside the geometry). Otherwise, execution is terminated. For such a case, the user should ensure that all starting points are located inside the geometry that are either defined with start type 6 data or defined in ASCII start data file(s).

8.1.7 THEORY AND TECHNIQUES

8.1.7.1 The transport equation

The equation KENO solves may be derived in the following manner, starting with the Boltzmann neutron transport equation which may be written as

$$\begin{aligned}\frac{1}{\mathrm{v}}\frac{\partial\Phi}{\partial t}(X,E,\Omega,t)+\Omega\bullet\nabla\Phi(X,E,\Omega,t)+\Sigma_t(X,E,\Omega,t)\,\Phi(X,E,\Omega,t)\\ =S(X,E,\Omega,t)\\ +\int_{E'}\int_{\Omega'}\Sigma_s\left(X,E'\rightarrow E,\Omega'\rightarrow\Omega,t\right)\Phi\left(X,E',\Omega',t\right)d\Omega'dE'\,,\end{aligned}\tag{8.1.3}$$

where

$\Phi(\mathrm{X},\mathrm{E},\Omega,\mathrm{t})$ = neutron flux (neutrons/cm^2/s) per unit energy at energy E per steradian about direction Ω at position X at time t moving at speed v corresponding to E;

$\Sigma_t(\mathrm{X},\mathrm{E},\Omega,\mathrm{t})$ = macroscopic total cross section of the media (cm^{-1}) at position X, energy E, direction Ω and time t;

$\Sigma_s(\mathrm{X},\mathrm{E}'\rightarrow\mathrm{E},\Omega'\rightarrow\Omega,\mathrm{t})$ = macroscopic differential cross section of the media (cm^{-1}) per unit energy at energy E' per steradian about direction Ω' at position X, and time t, for scattering to energy E and direction Ω;

$\mathrm{S}(\mathrm{X},\mathrm{E},\Omega,\mathrm{t})$ = neutrons/cm^3/s born at position X and time t per unit energy at energy E per steradian about direction Ω (excludes scatter source).

Defining $\mathrm{q}(\mathrm{X},\mathrm{E},\Omega,\mathrm{t})$ as the total source resulting from the external source, scattering, fission, and all other contributions, the following relationship can be written.

$$\begin{aligned}q(X,E,\Omega,t)=\\ S(X,E,\Omega,t)+\int_{E'}\int_{\Omega'}\Sigma_s\left(X,E'\rightarrow E,\Omega'\rightarrow\Omega,t\right)\Phi\left(X,E',\Omega',t\right)d\Omega'dE'\,.\end{aligned}\tag{8.1.4}$$

Combining Eq. (8.1.3) and Eq. (8.1.4), assuming media to be stationary and ignoring time-dependence, yields

$$\Omega\bullet\nabla\Phi(X,E,\Omega)+\Sigma_t(X,E,\Omega)\,\Phi(X,E,\Omega)=q(X,E,\Omega)\tag{8.1.5}$$

8.1.7.2 Continuous energy mode solution procedure

Using the relationship $X' = X - R\Omega$, using an integrating factor on both sides of Eq. (8.1.5), and defining

$$T(R) = \int_0^R \Sigma_t(X - R^{'}\Omega, E)dR, \tag{8.1.6}$$

the following equation can be written.

At this point, the problem becomes an eigenvalue problem. If there is no external source, the source may be defined as

$$\Phi(X, E, \Omega) = \int_O^\infty q(X - R\Omega, E, \Omega)e^{-T(R)}dR \tag{8.1.7}$$

At this point, the problem becomes an eigenvalue problem. If there is no external source, the source may be defined as

$$\begin{aligned} q\,(X, E, \Omega) = \int \int dE^{'} d\Omega^{'}\ \Phi\left(X, E^{'}, \Omega^{'}\right)\ \Sigma_{s}\left(X, E^{'} \rightarrow E, \Omega^{'} \cdot \Omega\right) \\ +\ \frac{1}{k} Q^{'}(X, E, \Omega), \end{aligned} \tag{8.1.8}$$

where

k is the largest eigenvalue of the integral equation,

$Q'(X, E, \Omega)$ is the fission source at position X for energy E and direction Ω (all fission contributions to point E from all energy points in the previous generation),

$\Sigma_s\,(X, E' \rightarrow E, \Omega' \bullet \Omega)$ is the scattering cross section for scattering at position X from energy point E' and direction Ω' to energy point E and direction Ω.

Assuming the fission neutrons to be isotropic, the fission source $Q'(X, E, \Omega)$ can be written as

$$Q^{'}(X, E, \Omega) = \frac{1}{4\pi} \int_{E'} \int_{\Omega'} dE^{'} d\Omega^{'} \Phi(X, E^{'}, \Omega^{'}) \chi(X, E^{'} \rightarrow E) \nu(X, E^{'}) \Sigma_f(X, E^{'}), \tag{8.1.9}$$

where

- $\chi\,(X, E' \rightarrow E)$ **is the fraction of neutrons born at energy point E from** fission at energy point E' in the media at position X,
- $\nu\,(X, E')$ **is the number of neutrons resulting from a fission at energy** point E' at position X,
- $\Sigma_f\,(X, E')$ **is the macroscopic fission cross section of the** material at position X for a neutron at energy point E'.

Substituting Eq. (8.1.8) into Eq. (8.1.7) yields the following equation:

$$\begin{aligned} \Phi\,(X, E, \Omega) = \int_0^\infty \mathrm{dR}\ e^{-T(R)} \left\{ \frac{1}{k} Q^{'}\,(X - R\Omega, E, \Omega) \right. \\ \left. +\ \int_{E'} \int_{\Omega'} dE^{'} d\Omega^{'} \Phi\left(X - R\Omega, E^{'}, \Omega^{'}\right) \Sigma_{s}\left(X - R\Omega, E^{'} \rightarrow E, \Omega^{'} \cdot \Omega\right) \right\} \end{aligned} \tag{8.1.10}$$

The definition of k may be given as the ratio of the number of neutrons produced in the (n + 1)*th* generation to the number of neutrons produced in the n*th* generation or the largest eigenvalue of the integral equation. Using Eq. (8.1.9), Eq. (8.1.7) can be written as

$$\Phi(X,E,\Omega)=\int_0^\infty dRe^{-T(R)}\left\{\frac{1}{k}\int_{E'}\int_{\Omega'}\nu\left(X-R\Omega,E'\right)\Sigma_f\left(X-R\Omega,E'\right)\chi\left(X-R\Omega,E'\rightarrow E\right)\Phi\left(X-R\Omega,E',\Omega'\right)d\mathrm{E}'\frac{d\Omega'}{4\pi}\right.$$
$$\left.+\int_{E'}\int_{\Omega'}d\mathrm{E}'d\Omega'\Phi\left(X-R\Omega,E',\Omega'\right)\Sigma_s\left(X-R\Omega,E'\rightarrow E,\Omega'\cdot\Omega\right)\right\} \tag{8.1.11}$$

Writing Eq. (8.1.11) in generation notation, multiplying and dividing certain terms by $\Sigma_t(X,E)$ and multiplying both sides of the equation by $\nu(X,E)\Sigma_f(X,E)$, yields the following equation, which is solved by KENO V in the continuous energy mode:

$$\frac{\nu(X,E)\Sigma_f(X,E)}{\Sigma_t(X,E)}\Sigma_t(X,E)\Phi_n(X,E,\Omega)=\frac{\nu(X,E)\Sigma_f(X,E)}{\Sigma_t(X,E)}\Sigma_t(X,E)\int_0^\infty dRe^{-T(R)}$$
$$\left\{\frac{1}{k}\int_{E'}\int_{\Omega'}\frac{\nu\left(X-R\Omega,E'\right)\Sigma_f\left(X-R\Omega,E'\right)}{\Sigma_t\left(X-R\Omega,E'\right)}\chi\left(X-R\Omega,E'\rightarrow E\right)\Sigma_t\left(X-R\Omega,E'\right)\Phi_{n-1}\left(X-R\Omega,E',\Omega'\right)d\mathrm{E}'\frac{d\Omega'}{4\pi}\right.$$
$$\left.+\int_{E'}\int_{\Omega}\frac{\Sigma_S\left(X-R\Omega,E'\rightarrow E,\Omega'\cdot\Omega\right)}{\Sigma_t\left(X-R\Omega,E'\right)}\Sigma_t\left(X-R\Omega,E'\right)\Phi_n\left(X-R\Omega,E',\Omega'\right)dE'd\Omega'\right\} \tag{8.1.12}$$

where n indicates the n*th* generation and n - 1 is the (n - 1)*th* generation. Note that the left-hand side of the equation, $\nu(X,E)\Sigma_f(X,E)\phi_n(X,E,\Omega)$ is the fission production for the n*th* generation.

The solution strategy used by KENO solves Eq. (8.1.12) by using an iterative procedure. The fission production at point X at energy point E due to neutrons in the (n - 1)*th* generation, normalized to the system multiplication, is

$$\frac{1}{k}\int_{E'}\int_{\Omega'}\frac{\nu(X,E')\Sigma_f(X,E')}{\Sigma_t(X,E')}\chi(X,E'\rightarrow E)\Sigma_t(X,E)\Phi_{n-1}(X,E'\Omega')dE'\frac{d\Omega'}{4\pi} \tag{8.1.13}$$

The collision points used in KENO are chosen by selecting path lengths from the distribution

$e^{-T(R)}$,

which is the probability of transport from any position $X-R\Omega$ to position X.

The first collision density of neutrons at energy E per unit solid angle about Ω resulting from the fission source produced by the (n - 1) generation, normalized to the system multiplication, is

$$\Sigma_t(X,E)\int_0^\infty dRe^{-T(R)}\frac{1}{k}\int_{\Omega'}\int_{E'}\frac{\nu(X-R\Omega,E')\Sigma_f(X-R\Omega,E')}{\Sigma_t(X-R\Omega,E')}$$
$$\chi(X-R\Omega,E'\rightarrow E)\Sigma_t(X-R\Omega,E')\Phi_{n-1}(X-R\Omega,E',\Omega')dE'\frac{d\Omega'}{4\pi} \tag{8.1.14}$$

The scattering source at position X emerging at energy E and direction Ω resulting from previous collisions in the same generation, is

$$\int_{E'}\int_{\Omega'}\frac{\Sigma_s\left(X,E'\rightarrow E,\Omega'\cdot\Omega\right)}{\Sigma_t\left(X,E'\right)}\Sigma_t\left(X,E'\right)\Phi_n\left(X,E',\Omega'\right)d\Omega'dE' \tag{8.1.15}$$

The collision density at energy E, per solid angle about Ω is

$$\Sigma_t(X,E)\int_0^{\infty} \mathrm{dR}\; e^{-T(R)} \int_{E'}\int_{\Omega'} \frac{\Sigma_s\left(X-R\Omega, E' \rightarrow E, \Omega'\cdot\Omega\right)}{\Sigma_t\left(X-R\Omega,E'\right)} \Sigma_t\left(X-R\Omega,E'\right)\Phi_n\left(X-R\Omega,E',\Omega'\right)d\Omega' dE' \tag{8.1.16}$$

The total collision density times $\frac{\nu_g(X)\Sigma_{fg}(X)}{\Sigma_{tg}(X)}$ is the relationship from which KENO picks the source points for the next generation.

Problem initialization

In order to use continuous energy cross sections in the random walk, various initialization tasks must be addressed for each problem. Based on the mixture specifications for a problem, KENO reads the microscopic cross section data for each nuclide and dynamically allocates storage for the particular problem. For continuous energy problems, if **UUM=**no then KENO will not allocate macroscopic cross sections for each material, and will instead use the master set of microscopic cross section data (on a non-unionized energy grid). This is the default behavior. Storing cross sections on a material-based unionized energy grid (**UUM=**yes) results in smaller cross section lookup times and faster KENO runtimes; however, storing unionized cross section data for every material can require a prohibitively large amount of memory for problems with a large number of materials. Users should therefore only set **UUM**=yes for relatively small models; experience is the best guide as to what "small" means with respect to available system resources. After finishing the cross section processing, KENO reads the user-specified KENO data (See Sect. 8.1.5) and stores the problem-dependent data for retrieval during the random walk. After the data initialization tasks are complete, the Monte Carlo random walk can be performed according to the procedures that are documented in the subsequent sections.

Initial source distribution

Before the Monte Carlo simulation can be performed, the initial source distribution of neutrons must be sampled. Typically, each mixture in a problem is composed of multiple isotopes, and each mixture must be checked for the presence of fissionable material. For each fissionable mixture, the volume fraction of fissionable material must be calculated on a per isotope basis.

The first source distribution is comprised of the initial spatial coordinates, direction cosines and energy for each neutron within the problem. Regarding the coordinate values, both continuous energy and multigroup KENO use the same start types, and the initial angular distribution is sampled from an isotropic distribution. For the initial energy distribution, the energy of each neutron must be sampled from the continuous energy fission spectrum, $\chi(E)$, of a fissionable isotope within the mixture. If more than one fissionable isotope is present, the i^{th} isotope can be selected using the following relation:

$$\sum_{j=1}^{i-1}\Sigma_f^j < R\sum_{j=1}^{N}\Sigma_f^j \le \sum_{j=1}^{i}\Sigma_f^j \tag{8.1.17}$$

where

$\sum_f^j$ = macroscopic fission cross section for the j^{th} isotope,

R = random number [0, 1),

N = total number of fissionable isotopes in the mixture.

Note that the relation in Eq. (8.1.17) requires the knowledge of the fission cross section at a particular energy. Therefore, an initial energy of 0.025 eV is assumed for selecting the appropriate isotope to sample. Once the i^{th} isotope is selected, the initial energy is sampled from the corresponding $\chi_{\text{i}}(E)$.

Collision site selection

Each neutron must be tracked until the history is terminated via leakage from the system or the particle is "killed" via roulette. The selection of the next collision site is governed by the following probability density function (PDF):

$$f(x)dx = \Sigma_t^m(E)e^{-\Sigma_t^m(E)x}dx, \tag{8.1.18}$$

where

$\Sigma_t^m(E)$ = macroscopic total cross section for mixture m at energy E,

x = spatial variable.

The PDF in Eq. (8.1.18) describes the probability that a neutron will have an interaction between x and $x + dx$ along its flight path. Integrating Eq. (8.1.18) over the spatial variable yields the following cumulative distribution function (CDF):

$$F(x) = \int_0^x \Sigma_t^m(E)e^{-\Sigma_t^m(E)x'}dx' = 1 - e^{-\Sigma_t^m(E)x}, \tag{8.1.19}$$

and the next collision site is determined by setting the CDF in Eq. (8.1.19) equal to a random number on the interval [0, 1) and solving for x.

Note that Eq. (8.1.19) requires the total cross section for mixture m at energy E in order to calculate the next collision site. As mentioned in the Sect. 8.1.7.2.1, KENO has two approaches to calculate the macroscopic total cross section for each mixture; (1) use a unionized energy grid (**UUM=**yes) — once KENO reads the microscopic data for each isotope/nuclide in a mixture, KENO calculates and stores the macroscopic total cross sections for the nuclides in each mixture (2) on-the-fly mixture macroscopic cross section calculation (**UUM=**no) — KENO does not store macroscopic cross sections and instead calculates mixture cross sections upon request during particle tracking. Selecting **UUM**=yes results in substantially increased memory usage for problems with a large number of materials, making it impossible to perform CE calculations for some detailed problems. Thus, it is highly recommended to use **UUM**=no, the default setting, for the large problems with multiple mixture definitions. For most cases selecting **UUM**=no increases the runtime for KENO CE calculations by approximately 17 %, but this feature also expands the code's capability to simulate very large problems with multiple mixture configurations.

Just as the `UUM` option allows the user to prevent KENO from storing mixture cross sections on a unionized energy grid, the `M2U` option toggles whether or not KENO stores cross sections on a unionized energy grid for each individual nuclide. The default for this option is `M2U=no`, which prevents the storage of all the transport cross sections (i.e. inelastic scattering levels) for each nuclide in addition to the major reactions that are already on the unionized energy grid (i.e. total, fission, capture, and scatter) on an energy grid that is unionized for that nuclide, and setting `M2U`=yes activates this unionization and storage. `M2U`=no reduces the nuclear data memory footprint of large models by several gigabytes, but it also increases the runtime of these problems by several percent. It is encouraged to investigate both `UUM` and `M2U` options in continuous energy problems to optimize the runtime and memory depending on the user's models and applications. In the continuous energy depletion calculations, `UUM` parameter is intentionally forced to "no" to minimize the memory requirement of internal data storage. The cross section storage and treatment options should be used consistently in both validation and analysis calculations.

If any of the isotopes in the mixture have unresolved-resonance data and corresponding probability-table data, KENO must determine if the neutron energy is in the unresolved resonance range (URR) for each isotope

during the random walk. If the neutron energy is in the URR for an isotope, the appropriate probability table must be sampled to obtain the total cross section for each isotope (refer to Sect. 8.1.7.2.1 for sampling probability tables). Subsequently, the macroscopic total cross section for mixture m is adjusted to account for sampling the probability tables. Once the revised $\Sigma_t^m(E)$ is determined, Eq. (8.1.19) can be used to select the next collision site within mixture m at energy E.

Collision treatment

Once the collision site is determined, the collision can be modeled, and the post collision parameters can be calculated. In the continuous energy approach, each collision is modeled with an individual isotope/nuclide. If a mixture is defined by multiple isotopes and/or nuclides, the target must be selected at each collision site. If there are N different isotopes/nuclides present in a mixture, the following equation can be used to select the i^{th} target for interaction:

$$\sum_{j=1}^{i-1} \Sigma_t^j < R \sum_{j=1}^{N} \Sigma_t^j \leq \sum_{j=1}^{i} \Sigma_t^j, \tag{8.1.20}$$

where,

Σ_t^j = macroscopic total cross section for the j^{th} isotope/nuclide.

As noted in Sect. 8.1.7.2.2 and Sect. 8.1.7.2.3, the collision energy may be in the URR of one or more isotopes within the mixture. Consequently, the total cross section for each isotope that has unresolved-resonance data is sampled from the probability table information prior to selecting the next collision site (see discussion in Sect. 8.1.7.2.1). Subsequently, the sampled values for the total and partial reactions are stored for retrieval. Therefore, the macroscopic total cross section values that are used in Eq. (8.1.20) are adjusted to account for sampling the probability table data by retrieving the appropriate microscopic total cross section value at energy E. Once the corrected values for $\Sigma_t^j(E)$ are determined, Eq. (8.1.20) can be used to select the target for interaction.

After selecting the collision target, the neutron's weight is reduced by the nonabsorption probability:

$$w = \frac{\sigma_s^i(E)}{\sigma_t^i(E)} w_b = P_{nabs}(E) w_b, \tag{8.1.21}$$

where

$\sigma_s^i(E)$ = microscopic scattering cross section for the i^{th} isotope/nuclide at energy E,

$\sigma_t^i(E)$ = microscopic total cross section for the i^{th} isotope/nuclide at energy E,

wb = weight before collision.

The absorption and fission weights are calculated using the following relations, respectively:

$$w_a = \frac{\sigma_a^i(E)}{\sigma_t^i(E)} w_b, \tag{8.1.22}$$

where

$\sigma_a^i(E)$ = microscopic absorption cross section for the i^{th} isotope/nuclide at energy E,

and

$$w_f = \frac{\bar{\nu}^i(E)\sigma_f^i(E)}{\sigma_t^i(E)} w_b, \tag{8.1.23}$$

$\bar{\nu}^i$ = average number of neutrons released per fission at energy E,

$\sigma_f^i(E)$ = microscopic fission cross section for the i^{th} isotope/nuclide at energy E.

If the collision isotope is in the URR at energy E and probability table data are available, the collision probabilities must be adjusted for sampling the partial reactions from the appropriate probability table. In particular, the revised or sampled values for scattering, absorption and fission must be used to calculate the collision weights as defined by Eq. (8.1.21) through Eq. (8.1.23). Depending on the neutron's weight, splitting and/or Roulette are performed as necessary. Once the appropriate weights are calculated, the collision can be processed to determine the type of interaction and the corresponding exiting energy and angle if secondary neutrons are generated.

Because an explicit collision treatment is dictated by the point cross section data, the type of reaction must be modeled explicitly in the continuous energy version of KENO. Fig. 8.1.204 summarizes the cross section hierarchy and can be used as an aide to understanding the collision treatment in KENO in continuous energy mode. After selecting the isotope/nuclide for interaction according to Eq. (8.1.20) and calculating the weights using Eq. (8.1.21) through Eq. (8.1.23), the collision is modeled using the following procedures:

1. At each collision site, ensure that the type of collision is selected based on the cross section data. Moreover, the k^{th} reaction can be randomly selected using the following relation:

$$\sum_{j=1}^{k-1} \sigma_j < R \sum_{j=1}^{NI} \sigma_j < \sum_{j=1}^{k} \sigma_j \tag{8.1.24}$$

   where

   $\sigma_j(E)$ = cross section for the j^{th} reaction,

   NI = number of elastic and nonelastic reactions (i.e., excludes neutron disappearance reactions).

   Note that the fission reactions (e.g., first chance fission, second chance fission, etc.) are excluded from Eq. (8.1.24) because an implicit approach is used to treat fission. The details of the fission treatment are provided in Sect. 8.1.7.2.5. If a neutron production reaction with multiple exit neutrons such as (n, 2n), (n, 3n) is selected and the reaction is not a fission reaction, then the weight of the neutron is adjusted by the multiplicity to account for the additional neutrons, such as $w = 2w$ for (n, 2n), $w = 3w$ for (n, 3n).

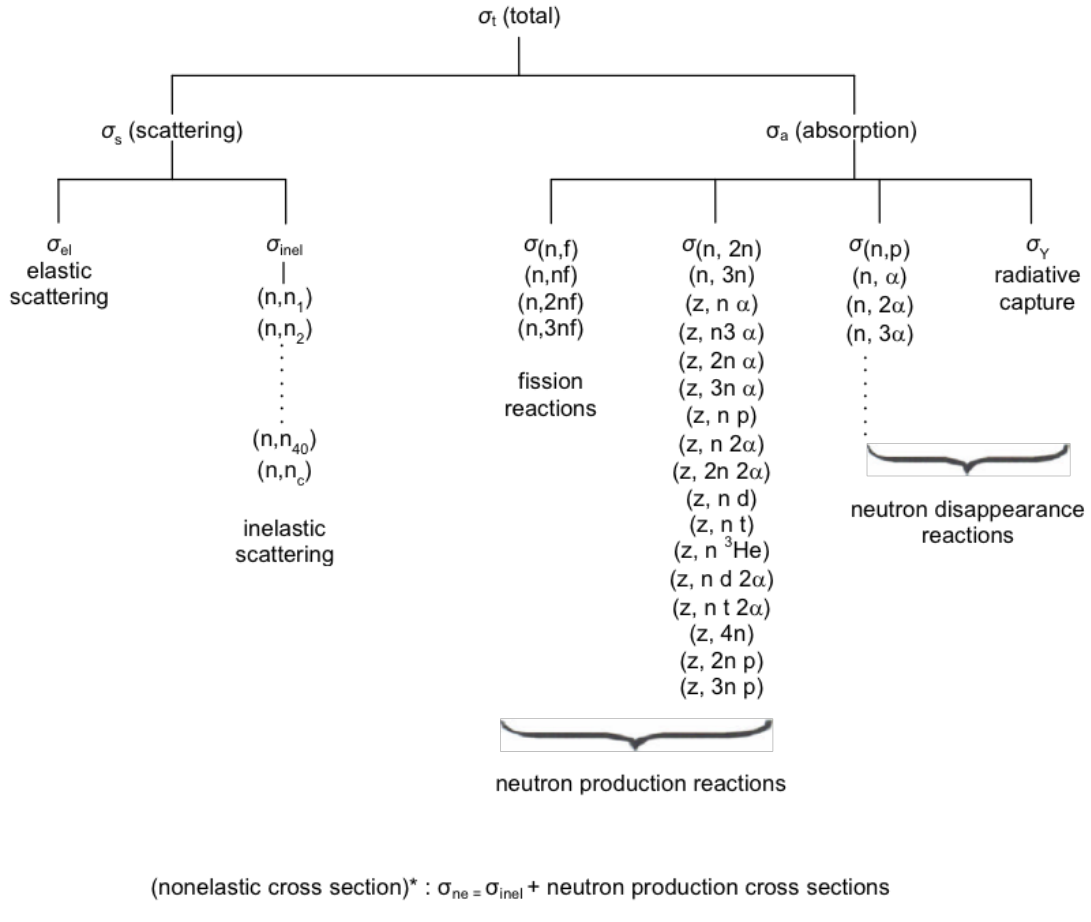


Fig. 8.1.204: Cross section hierarchy for particle interactions.

2. Select the exiting angle for the collision. The cross section library provides a kinematics data block that provides the angle and energy distributions of secondary particles for a specified reaction. Each record for each section of a reaction provides a flag (NPU) indicating the number of secondary angles for that energy. If the NPU flag is 1 and the corresponding cosine value is -2 (set as a programming flag), then the reaction is isotropic. The exiting angle for the collision can be obtained as follows:

   i. Isotropic Emission

   If the LAND flag is 0 for the secondary angular distribution, the exiting angle cosine is sampled uniformly between -1 and +1:

$$\mu = 2R - 1 \tag{8.1.25}$$

   If the LAND flag is 1 for the secondary angular distribution, the exiting angle cosine must be sampled from an anisotropic distribution provided in the cross section library for the specified reaction. Details for sampling the exiting angle from the angular distribution data are provided in Sect. 8.1.7.2.6.

   The secondary angular distribution data are provided as a function of incident energy. The initial record of each section for a reaction provides the number of incident energies (*NE*) for the secondary angular

and energy distributions. Corresponding to each incident energy is a secondary angular distribution. Therefore, there will be *NE* records used to describe the secondary angular data for each section.

For each angular distribution record, there is an *LMU* flag and a *NPU* flag that describes the format of the data. The *LMU* flag designates whether the secondary distribution is provided in equiprobable cosine bins or in nonequiprobable cosine bins. The procedures for sampling the two types of data are provided in Sect. 8.1.7.2.10. The *LMU* flag must be the same for all *NE* records within a section. The *NPU* flag specifies the number of cosine values in the distribution.

Based on the incident energy of the neutron, the angular distribution data are sampled according to the procedures of Sect. 8.1.7.2.10 to obtain the exiting angle cosine, μ, in the lab system.

3. Select the exiting energy for the collision. In addition to the angular distributions of the kinematics data, the cross section library provides the energy distributions of secondary particles for a specified reaction.

   The secondary energy distributions are provided as a function of incident energy and secondary angle cosine in each section. Once the exiting angle cosine is selected, the exiting energy is selected from the energy distribution that corresponds to the (E, μ) pair. The exit energy is determined using the procedures of Sect. 8.1.7.2.10.

   For each energy distribution record, there is an *LE* flag and an *NPE* flag describing the format of the data. The *LE* flag designates whether the secondary distribution is provided in equiprobable energy bins or in nonequiprobable energy bins. The procedures for sampling the two types of data are provided in Sect. 8.1.7.2.10. Note that the *LE* flag must be the same for all *NE* records within a section for the secondary energy distribution data. The *NPE* flag specifies the number of secondary energy values in the distribution. For elastic scattering and discrete-level inelastic scattering, the *NPE* flag will equal the *NPU* flag due to the one-to-one correspondence between angle and energy.

4. Select the azimuthal angle η uniformly between 0. and 2π (i.e., $\eta = 2R\pi$.

5. Calculate the new direction cosines in the lab system using the following equations:

$$u' = u\mu - \sqrt{v^2 + w^2}\sqrt{1-\mu^2}\cos(\eta) \tag{8.1.26}$$

$$v' = v\mu + \frac{uv}{\sqrt{v^2 + w^2}}\cos(\eta)\sqrt{1-\mu^2} - \frac{w}{\sqrt{v^2 + w^2}}\sqrt{1-\mu^2}\sin(\eta) \tag{8.1.27}$$

$$w' = w\mu + \frac{uw}{\sqrt{v^2 + w^2}}\cos(\eta)\sqrt{1-\mu^2} + \frac{v}{\sqrt{v^2 + w^2}}\sqrt{1-\mu^2}\sin(\eta) \tag{8.1.28}$$

where

u, v, and w = initial direction cosines,

$u', v',$ and w' = exiting direction cosines,

μ = cosine of the scattering angle, and

η = cosine of the azimuthal angle.

Fission treatment

As noted in the previous section, an implicit approach is used to treat a fission event. During the collision treatment, the fission weight is calculated using Eq. (8.1.23). After processing the collision, the fission weight is evaluated to determine if the fission treatment should be applied. If the fission weight is greater than zero, the collision occurred in fissile material. During the random walk, several fission source points must be generated to provide an adequate representation of the true source distribution. A minimum production factor is defined at the beginning of each generation to ensure that enough fission points are generated:

$$mpf = \frac{3.0\bar{k}}{\sqrt{FG}}, \tag{8.1.29}$$

where

$\bar{k}$ = running average of k_{eff} through the current generation, and

FG = number of histories per generation.

Eq. (8.1.29) represents an estimate of the 99% lower confidence interval for the distribution of the generation k-effective.

If the fission weight is greater than zero, a pseudo-fission weight is calculated as follows:

$$pfw = \frac{w_f}{R} = \frac{\nu^{-i}(E)\sigma^i_f(E)}{R\sigma^i_t(E)} w_b. \tag{8.1.30}$$

If the pseudo-fission weight is less than the minimum production factor given by Eq. (8.1.29) fission points are not generated and tracking of the particle continues. However, if the pseudo-fission weight is greater than the minimum production factor, *pfw* is redefined to be the minimum production factor divided by a random number:

$$pfw = \frac{mpf}{R} \tag{8.1.31}$$

Once the pseudo-fission weight is redefined, a fission point can be generated. Each time a fission point is generated, the pseudo-fission weight is stored with the point in the fission bank. When a new point is stored in the fission bank, the energy and angular cosine must be selected from the kinematics data. Once the kinematics data are sampled for the new fission point, the fission weight of the history is decremented by the minimum production factor. If the remaining fission weight is greater than zero, the fission treatment procedures are repeated until the fission weight of the history has been exhausted (i.e., $w_f \leq 0$).

Sampling details

The preceding sections describe the procedures for the continuous energy random walk for KENO. During the random walk, KENO must sample various tables of data that may include probability tables and angle-energy distributions for secondary particles.

Probability tables

For each isotope with unresolved resonance data, multiple probability tables may be used to describe the URR. The header block for each isotope has an LPTAB flag that provides the number of probability tables for an isotope. Each table is defined for a range of incident energies between E_i and E_{i+1}, and the energy range for a table should not overlap with another table for the isotope. For a particle with energy E, a search must be performed to find the probability table with energy bounds that bracket the particle energy, $E_i \leq E < E_{i+1}$.

Once the appropriate table is identified, the table can be sampled to obtain the total, elastic scattering, fission and capture cross section values in the URR.

The probability table block provides four separate records that correspond to each reaction within a table; however, the probability table construction is based on the total cross section. Therefore, the probabilities in each table refer to the total cross section band values, and the bands should be sampled based on the total cross section. Once the band is sampled, the corresponding partial reaction cross section values are selected from the same band as the total cross section. The cross section format permits the probability table to be expressed in equiprobable or nonequiprobable cross section bands (i.e., LBND = 0 or 1, respectively). The procedures for sampling both types of tables are provided in the subsequent subsections.

Equiprobable cross section bands

Each probability record has an NB parameter that designates the number of cross section bands for a table. If the table is constructed with equiprobable cross section bands, the k^{th} band can be selected as follows:

$$k = NB\ ^{*}R + 1 \tag{8.1.32}$$

where

R = random number [0., 1).

Based on the sampled cross section band, the total cross section corresponding to k^{th} band is extracted from the table. Likewise, the elastic scattering, fission and capture cross section values that correspond to the k^{th} band are also extracted from their respective records in the probability-table block.

Nonequiprobable cross section bands

If the LBND flag is 1, the cross section bands in the table are not equiprobable, and a different procedure must be used to sample the appropriate cross section band. For this case, the probability values in the table must be constructed as a cumulative distribution function. As noted previously, each probability record has an NB parameter that designates the number of cross section bands for a table, and the k^{th} band can be selected as follows:

$$\sum_{j=1}^{k-1} P_j < R \sum_{j=1}^{NB} P_j \leq \sum_{j=1}^{k} P_j, \tag{8.1.33}$$

where

Pj = probability corresponding to the j^{th} cross section band.

Based on the sampled cross section band, the total cross section corresponding to the k^{th} band is extracted from the table. Likewise, the elastic scattering, fission and capture cross section values that correspond to the k^{th} band are also extracted from their respective records in the probability-table block.

Kinematics data

The kinematics data in the KENO library are provided in the lab or target-at-rest system as opposed to the center-of-mass system. By adhering to the lab coordinate system, KENO does not have to transform between different coordinate systems during the random walk; however, the energy and angle representations for elastic and discrete-level inelastic scattering become more complex in the lab system. For example, an angular distribution that is isotropic in the center-of-mass system is anisotropic in the lab system. Moreover, the secondary energy distribution as a function of exit angle in the lab system can be double valued (i.e.,

two possible exit energies with respect to a single angle) for energies above the threshold for the reaction. Likewise, a similar double valued distribution is observed for elastic scattering with hydrogen in the lab system if the actual mass ratio is used (i.e., $A = 0.99928$ as opposed to $A = 1.0$). Consequently, special care must be exercised in the construction of the secondary angle and energy distributions in the lab system.

The subsequent sections address the general procedures for sampling the kinematics data with emphasis on the special cases that must be addressed during the random walk. In Sect. 8.1.7.2.11, the general procedures for sampling the kinematics data are provided, Sect. 8.1.7.2.14 discusses isotropic angular data with energy coupling, while Sect. 8.1.7.2.15 discusses coherent and incoherent elastic scattering. A discussion is also provided in Sect. 8.1.7.2.16 to describe the elastic and discrete-level inelastic treatment.

General procedures

The kinematics data format is designed to accommodate coupled angle energy distributions of secondary particles. The following discussion provides the procedures for sampling the coupled distributions. Special cases such as elastic and discrete level inelastic scattering are discussed after the "General Procedures" section.

The sampling procedures in the following sections assume that the angle and energy distributions are continuous in nature. As a result, interpolation procedures can be used to obtain intermediate angle or energy values between the tabulated angle or energy values. During the course of KENO development, there may be a need to provide an average angle or energy value for a bin. The kinematics format can be modified as needed to accommodate additional angle and energy representations. Therefore, the following procedures represent the anticipated approach for sampling the current form of the kinematics data.

Exit angle cosine

The first record for each reaction provides the number of sections (NSECT) used to describe the kinematics for the reaction. Within each section, the first record defines the incident energy range for the section. After the first record for a section, the first block of data provides the secondary angular distributions for *NE* different incident energies within the energy range of the section. Consequently, there will be *NE* different angular distribution records in the secondary angle cosine block. The kinematics format permits the anisotropic angular distributions to be expressed in either equiprobable or nonequiprobable cosine bins (i.e., LMU = 0 or 1, respectively). Moreover, the formats permit the number of cosine bins to vary as a function of incident energy. In particular, each cosine distribution record can have *NPU* secondary angles that correspond to *NPU* - 1 cosine bins.

Equiprobable Cosine Bins. If the incident energy, E, is between E_{i} and $E_{\mathrm{i+1}}$, the angle cosine is sampled in both tables, and the exit angle cosine is obtained by interpolating between the two tables. If the angular distribution is provided in equiprobable cosine bins, the a^{th} bin is selected from the i^{th} table as follows:

$$a = (NPU_i - 1)\ {}^{*}R_1 + 1 = NA_i * R_1 + 1. \tag{8.1.34}$$

In Eq. (8.1.34) a denotes an integer quantity, and NA_{i} is the number of cosine bins for the i^{th} table. Using a similar procedure, the b^{th} bin is selected from the i+1$^{\mathrm{st}}$ table:

$$b = NA_{i+1}\ {}^{*}R_1 + 1 \tag{8.1.35}$$

As noted for the i^{th} table, b is an integer quantity in Eq. Eq. (8.1.35) and $NA_{\mathrm{i+1}}$ is the number of cosine bins for the i+1$^{\mathrm{st}}$ table. Once the cosine bins are selected, the cosine of the exiting angle from the i^{th} and i+1$^{\mathrm{st}}$ tables is calculated with Eq. (8.1.36) and Eq. (8.1.37), respectively:

$$\dot{\mu}_i = \mu_{ia} + (a - NA_i * R_1)\,(\mu_{ia+1} - \mu_{ia})\,, \tag{8.1.36}$$

$$\dot{\mu}_{i+1} = \mu_{i+1b} + (b - NA_{i+1} * R_1)\,(\mu_{i+1b+1} - \mu_{i+1b})\,. \tag{8.1.37}$$

The exiting angle cosine is obtained by interpolating between i and i+1 based on the incident energy grid:

$$\mu = \dot{\mu}_i + \left(\frac{E - E_i}{E_{i+1} - E_i}\right)(\dot{\mu}_{i+1} - \dot{\mu}_i)\,. \tag{8.1.38}$$

Nonequiprobable Cosine Bins. If the angular distribution is in the form of nonequiprobable cosine bins, the a^{th} bin is selected from the cumulative distribution function for the i^{th} table:

$$C_{ia-1} < R_1 \leq C_{ia}\ , \tag{8.1.39}$$

where

C_{ia-1} = cumulative probability corresponding to the $a - 1^{st}$ cosine bin,

C_{ia} = cumulative probability corresponding to the a^{th} cosine bin,

Using a similar procedure, the b^{th} bin is selected from the i+1$^{\text{st}}$ table:

$$C_{i+1b-1} < R_1 \leq C_{i+1b}\ , \tag{8.1.40}$$

where

C_{ib-1} = cumulative probability corresponding to the $b - 1^{st}$ cosine bin,

C_{ib} = cumulative probability corresponding to the b^{th} cosine bin.

If the cosine bins are not equiprobable and the PDF is represented by a continuous distribution, the bins are selected so that linear interpolation can be performed in the PDF. Because the CDF is obtained by integrating the PDF, the interpolation procedure for the CDF has a quadratic form. The value for $\dot{\mu}_i$ is obtained with the following equation:

$$\dot{\mu}_i = \mu_{ia} + \frac{\sqrt{P_{ia}^2 + 2s_i(R_1 - C_{ia-1})} - P_{ia}}{s_i}\ , \tag{8.1.41}$$

and

$$s_i = \frac{P_{ia+1} - P_{ia}}{\mu_{ia+1} - \mu_{ia}}\ , \tag{8.1.42}$$

where

P_{ia} = value of the PDF corresponding to the lower boundary of the a^{th} cosine bin in the distribution for E_i, and

P_{ia+1} = value of the PDF corresponding to the upper boundary of the a^{th} cosine bin in the distribution for E_i.

Likewise, the equation for $\dot{\mu}_{i+1}$ is obtained using an equation that is similar to Eq. (8.1.41)

$$\dot{\mu}_{i+1} = \mu_{i+1b} + \frac{\sqrt{P_{i+1b}^2 + 2s_{i+1}(R_1 - C_{i+1b-1})} - P_{i+1b}}{s_{i+1}}, \tag{8.1.43}$$

and

$$s_{i+1} = \frac{P_{i+1b+1} - P_{i+1b}}{\mu_{i+1b+1} - \mu_{i+1b}}, \tag{8.1.44}$$

where

P_{i+1b} = value of the PDF corresponding to the lower value of the b^{th} cosine bin in the distribution for E_{i+1}, and

P_{i+1b+1} = value of the PDF corresponding to the upper value of the b^{th} cosine bin in the distribution for E_{i+1}.

The exiting angle cosine is obtained by interpolating between $\dot{\mu}_i$ and $\dot{\mu}_{i+1}$ based on the incident energy using Eq. (8.1.38).

Exit energy

In the kinematics data block, each incident energy has a secondary distribution of *NPU* angle cosines. Therefore, there are *NPU* (E, μ) pairs associated with each incident energy. For each (E, μ) pair, there is a corresponding exit energy distribution that can have *NPE* exit energies. Because the kinematics data are tabulated in a coupled angle-energy format, the exit energy is obtained by a 2-D interpolation as shown in Fig. 8.1.205.

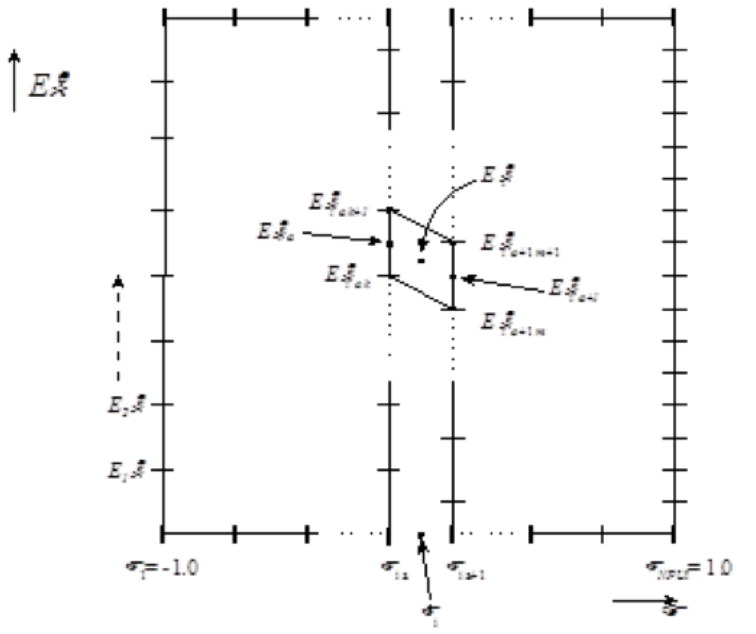
The objective of the sampling procedure is to sample the exit energy *E'* in conjunction with the sampled angle cosine. Therefore, the interpolation for the exit energy is performed with an equation that is analogous to Eq. (8.1.38):

$$E' = E'_i + \left(\frac{E - E_i}{E_{i+1} - E_i}\right)(E'_{i+1} - E'_i) \tag{8.1.45}$$

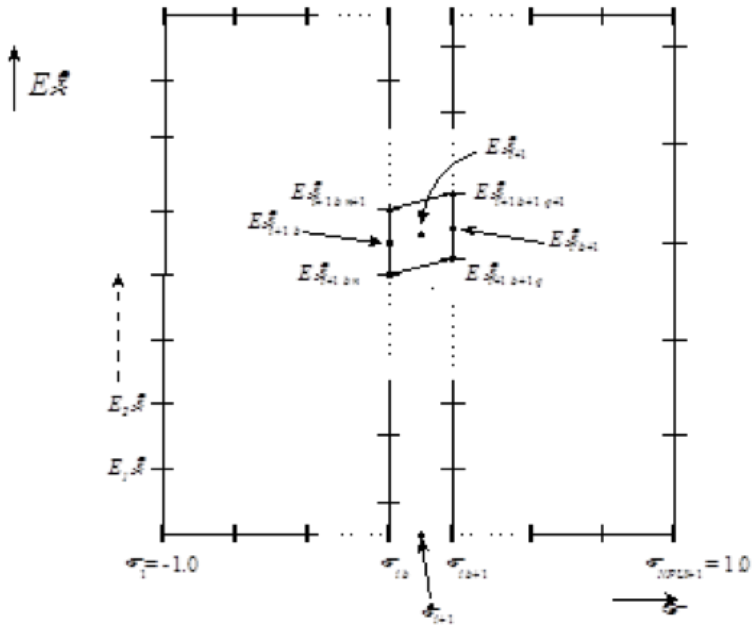
Based on Eq. (8.1.45), the values of E'_i and E'_{i+1} must be obtained in order to calculate the exit energy for the collision. The a^{th} and b^{th} cosine bins are selected according to the procedures in Sect. 8.1.7.2.11 from tables i and i+1, respectively. As a result, the secondary energy distributions corresponding to the a^{th} and b^{th} cosine bins must be used to sample the exit energies from tables i and i+1. The details for sampling the energy bins are deferred for the moment. Once the appropriate energy bins are sampled from the two tables, the interpolation for E'_i and E_{i+1} is obtained with the following equations:

$$E'_i = E'_{ia} + \left(\frac{\dot{\mu}_i - \mu_{ia}}{\mu_{ia+1} - \mu_{ia}}\right)\left(E'_{ia+1} - E'_{ia}\right) , \tag{8.1.46}$$

$$E'_{i+1} = E'_{i+1b} + \left(\frac{\dot{\mu}_{i+1} - \mu_{i+1b}}{\mu_{i+1b+1} - \mu_{i+1b}}\right)\left(E'_{i+1b+1} - E'_{i+1b}\right). \tag{8.1.47}$$



Secondary Angle-Energy Table i for E_i



Secondary Angle-Energy Table $i+1$ for E_{i+1}

Fig. 8.1.205: Interpolation diagram for secondary angle-energy data.

Based on Eq. (8.1.46) and Eq. (8.1.47) the values for E'_{ia}, E'_{ia+1}, and $E'_{ia+1b+1}$ and E'_{i+1}. The values for E'_{ia} and E'_{ia+1} correspond to secondary energies that are coupled with the a^{th} and $a + 1^{st}$ angle cosines in Table i. Likewise, the values for E'_{i+1b} E'_{i+1b+1} correspond to the secondary energies that are coupled with the b^{th} and $b + 1^{st}$ angle cosines in Table i+1. Therefore, the exit energies from Table i are sampled from the secondary energy distributions that correspond to (E, μ_{ia}) and (E, μ_{ia+1}), and the energies from Table i+1 are sampled from the energy distributions for $(E, \mu_{i+1}b)$ and (E, μ_{i+1b+1}).

As with the angular data, the secondary energy distributions may be provided in equiprobable or nonequiprobable energy bins. The following discussion provides the procedures for sampling the different distributions.

Equiprobable Energy Bins. In Table i, the k^{th} and m^{th} energy bins are sampled from the energy distributions for $(E, \mu_{i\,a})$ and $(E, \mu_{i\,a+1})$, respectively. The number of energy bins in a distribution is one less than the number of energies in the table (i.e., *NPE* - 1). If the energy distributions are provided in equiprobable bins, the k^{th} and m^{th} bins are selected as follows:

$$k = (NPE_{ia} - 1) * R_2 + 1, \tag{8.1.48}$$

$$m = (NPE_{ia+1} - 1) * R_2 + 1. \tag{8.1.49}$$

The exiting energy values for $E'_{i\,a}$ $E'_{i\,a+1}$ are interpolated as follows:

$$E'_{ia} = E'_{iak} + \Big[k - (NPE_{ia} - 1) * R_2\Big]\Big(E'_{iak+1} - E'_{iak}\Big), \tag{8.1.50}$$

$$E'_{ia+1} = E'_{ia+1m} + \Big[m - (NPE_{ia+1} - 1) * R_2\Big]\Big(E'_{ia+1m+1} - E'_{ia+1m}\Big). \tag{8.1.51}$$

In Table i+1, the n^{th} and q^{th} energy bins are sampled from the energy distributions for $(E, \mu_i b)$ and (E, μ_{ib+1}), respectively. As presented for the i^{th} table, the n^{th} and q^{th} bins are selected from an equiprobable distribution in a manner analogous with Eq. (8.1.48) and Eq. (8.1.49):

$$n = (NPE_{i+1b} - 1) * R_2 + 1, \tag{8.1.52}$$

$$q = (NPE_{i+1b+1} - 1) * R_2 + 1. \tag{8.1.53}$$

The exiting energy values for $E_{i+1\ b}$ and E_{i+1b+1} are interpolated as follows:

$$E'_{i+1b} = E'_{i+1bn} + \Big[n - (NPE_{i+1b} - 1) * R_2\Big]\Big(E'_{i+1bn+1} - E'_{i+1bn}\Big), \tag{8.1.54}$$

$$E'_{i+1b+1} = E'_{i+1b+1q} + \Big[q - (NPE_{i+1b+1} - 1) * R_2\Big]\Big(E'_{i+1b+1q+1} - E'_{i+1b+1q}\Big). \tag{8.1.55}$$

The exit energy from the i^{th} table (i.e., E') is obtained by substituting the values for E'_{ia} and E'_{ia+1} into Eq. (8.1.46), and the exit energy from the i+1$^{\text{st}}$ table (i.e., E'_{i+1}) is calculated by substituting the values for E'_{ib} and E'_{ib+1} into Eq. (8.1.47). Finally, the exit energy for the collision is obtained by inserting the calculated values for E'_i and E'_{i+1} into Eq. (8.1.45).

Nonequiprobable Energy Bins. If the energy distributions are provided in nonequiprobable bins, the k^{th} and m^{th} bins are sampled according to the cumulative distribution values:

$$C_{iak-1} < R_2 \leq C_{iak}, \tag{8.1.56}$$

$$C_{ia+1m-1} < R_2 \leq C_{ia+1m}, \tag{8.1.57}$$

where

C_{iak-1} = cumulative probability corresponding to the k-1st energy bin in the (E, μ_{ia}) distribution,

C_{iak} = cumulative probability corresponding to the k^{th} energy bin in the (E, μ_{ia}) distribution,

$C_{ia+1m-1}$ = cumulative probability corresponding to the m-1st energy bin in the $(E, \mu_i a + 1)$ distribution, and

C_{ia+1m} = cumulative probability corresponding to the m$^{\text{th}}$ energy bin in the (E, μ_{ia+1}) distribution.

For the nonequiprobable bins, the bins for the secondary energy are selected so that linear interpolation can be performed in the PDF. Since the CDF is obtained by integrating the PDF, the interpolation procedure for the CDF is quadratic in form. As a result, the value for $E'_{i\ a}$ is obtained with the following equation:

$$E'_{ia} = E'_{iak} + \frac{\sqrt{P^2_{iak} + 2s_{ia}(R_2 - C_{iak-1})} - P_{iak}}{s_{ia}}, \tag{8.1.58}$$

and

$$s_{ia} = \frac{\sqrt{P_{iak+1} - P_{iak}}}{E'_{iak+1} - E'_{iak}}, \tag{8.1.59}$$

where

P_{iak} = value of the PDF corresponding to the lower boundary of the k^{th} energy bin in the (E, $\mu_{i\ a}$) distribution, and

P_{iak+1} = value of the PDF corresponding to the upper boundary of the k^{th} energy bin in the (E, $\mu_{i\ a}$) distribution.

Likewise, the value for $E_{i\ a+1}$ is obtained with the following equation:

$$E'_{ia+1} = E'_{ia+1m} + \frac{\sqrt{P^2_{ia+1m} + 2s_{ia+1}(R_2 - C_{ia+1m-1})} - P_{ia+1m}}{s_{ia+1}}, \tag{8.1.60}$$

and

$$s_{ia+1} = \frac{P_{ia+1m+1} - P_{ia+1m}}{E'_{ia+1m+1} - E'_{ia+1m}} \tag{8.1.61}$$

where

P_{ia+1m} = value of the PDF corresponding to the lower boundary of the m^{th} energy bin in the (E, μ_{ia+1}) distribution, and

$P_{ia+1m+1}$ = value of the PDF corresponding to the upper boundary of the m^{th} energy bin in the (E, μ_{ia+1}) distribution.

For nonequiprobable distributions in the i+1st table, the n^{th} and q^{th} bins are sampled according to the cumulative distribution values:

$$C_{i+1bn-1} < R_2 \leq C_{i+1bn}, \tag{8.1.62}$$

$$C_{i+1b+1q-1} < R_2 \leq C_{i+1b+1q}, \tag{8.1.63}$$

where

$C_{i+1bn-1}$ = cumulative probability corresponding to the n-1st energy bin in the (E, $\mu_{i+1\ b}$) distribution,

C_{i+1bn} = cumulative probability corresponding to the n^{th} energy bin in the (E, $\mu_{i+1\ b}$) distribution,

$C_{i+1b+1q-1}$ = cumulative probability corresponding to the q-1st energy bin in the (E, $\mu_{i+1\ b+1}$) distribution, and

$C_{i+1b+1q}$ = cumulative probability corresponding to the q^{th} energy bin in the (E, $\mu_{i+1\ b+1}$) distribution.

As observed for the ith table, the interpolation procedure for the CDF is quadratic in form, and the value for $E'_{i+1\ b}$ is obtained as follows:

$$E'_{i+1b} = E'_{i+1bn} + \frac{\sqrt{P^2_{i+1bn} + 2s_{i+1b}(R_2 - C_{i+1bn-1})} - P_{i+1bn}}{s_{i+1b}}, \tag{8.1.64}$$

and

$$s_{i+1b} = \frac{P_{i+1bn+1} - P_{i+1bn}}{E'_{i+1bn+1} - E'_{i+1bn}}, \tag{8.1.65}$$

where

P_{i+1bn} = value of the PDF corresponding to the lower boundary of the n^{th} energy bin in the (E, μ_{i+1b}) distribution, and

$P_{i+1bn+1}$ = value of the PDF corresponding to the upper boundary of the n^{th} energy bin in the (E, μ_{i+1b}) distribution.

Likewise, the value for E'_{i+1b+1} is obtained with the following equation:

$$E'_{i+1b+1} = E'_{i+1b+1q} + \frac{\sqrt{P^2_{i+1b+1q} + 2s_{i+1b+1}(R_2 - C_{i+1b+1q-1})} - P_{i+1b+1q}}{s_{i+1b+1}}, \tag{8.1.66}$$

and

$$s_{i+1b+1} = \frac{P_{i+1b+1q+1} - P_{i+1b+1q}}{E'_{i+1b+1q+1} - E'_{i+1b+1q}}, \tag{8.1.67}$$

where

$P_{i+1b+1q}$ = value of the PDF corresponding to the lower boundary of the q^{th} energy bin in the (E, μ_{i+1b+1}) distribution, and

$P_{i+1b+1q+1}$ = value of the PDF corresponding to the upper boundary of the q^{th} energy bin in the (E, μ_{i+1b+1}) distribution.

To obtain the exit energy for the collision, the energy from the ith table (i.e., E'_i) is calculated by substituting the values for E'_{ia} and E'_{ia+1} into Eq. (8.1.46), and the exit energy from the i+1st table (i.e., E'_{i+1}) is calculated by substituting the values for E'_{ib} and E'_{ib+1} into Eq. (8.1.47). Finally, the exit energy for the collision is obtained by inserting the calculated values for E'_i and E'_{i+1} into Eq. (8.1.45).

Isotropic Angular Distributions With Energy Coupling

The kinematics format can accommodate isotropic angular distributions in the coupled angle energy format. The following discussion describes the special case with the appropriate sampling procedures. If the secondary angular distribution is isotropic at an incident energy E, a single exit cosine with a value of -2.0 is specified in the (E, μ) block. Therefore, the exit angle cosine is sampled uniformly between -1.0 and 1.0:

$$\mu = 2R_1 - 1. \tag{8.1.68}$$

Because there is only one exit cosine specified in the (E, μ) block, there is a single energy distribution record specified for the (E, μ) pair. The general structure (i.e., material identifiers and data flags are omitted for clarity) of the kinematics data within a section for a reaction is presented in Table 8.1.31. As a result, the sampling procedure for the exit energy is based on the incident energy.

Table 8.1.31: Kinematics data structure for isotropic angular distributions

(E, μ) Data Block					
E	C_2	μ	C_μ	P_μ	
E_1	0	-2.0	1.0	1.0	
E_2	0	-2.0	1.0	1.0	
.	.	.	.	.	
.	.	.	.	.	
.	.	.	.	.	
E_{NE}	0	-2.0	1.0	1.0	
(E, μ, $E¢$) Data Block					
E	μ	$E¢$(j)	$C_{E¢}$(j)	$P_{E¢}$(j)	j=1, NPE
E_1	-2.0	$E¢$(j)	$C_{E¢}$(j)	$P_{E¢}$(j)	j=1, NPE
E_2	-2.0	$E¢$(j)	$C_{E¢}$(j)	$P_{E¢}$(j)	j=1, NPE
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
E_{NE}	-2.0	$E¢$(j)	$C_{E¢}$(j)	$P_{E¢}$(j)	j=1, NPE

Once the angle is sampled according to Eq. (8.1.68), the exit energy is sampled in a manner that is analogous to the procedures of Sect. 8.1.7.2.13. If the incident energy, E, is between E_i and $E_{\mathrm{i}+1}$, the k^{th} energy bin is sampled from the distribution corresponding to E_i using Eq. (8.1.69) or Eq. (8.1.70) for equiprobable or nonequiprobable distributions, respectively.

$$k = (NPE_i - 1) * R_2 + 1, \tag{8.1.69}$$

or

$$C_{ik-1} < R_2 \leq C_{ik}, \tag{8.1.70}$$

where

NPE_i = number of exit energies corresponding to E_i,

C_{ik-1} = cumulative probability corresponding to the k-1$^{\mathrm{st}}$ energy bin in the distribution for E_i, and

C_{ik} = cumulative probability corresponding to the k$^{\mathrm{th}}$ energy bin in the distribution for E_i.

For equiprobable energy bins, the exit energy corresponding to E_i is calculated as follows:

$$E_i^{'} = E_{ik}^{'} + \left[k - (NPE_i - 1) * R_2\right]\left(E_{ik+1}^{'} - E_{ik}^{'}\right). \tag{8.1.71}$$

If the secondary energy distributions are provided in nonequiprobable bins the exit energy is calculated using the following equation:

$$E_i^{'} = E_{ik}^{'} + \frac{P_{ik}^2 + 2s_i(R_2 - C_{ik-1}) - P_{ik}}{s_i}, \tag{8.1.72}$$

and

$$s_i = \frac{P_{ik+1} - P_{ik}}{E_{ik+1}^{'} - E_{ik}^{'}}, \tag{8.1.73}$$

where

P_{ik} = value of the PDF corresponding to the lower boundary of the k^{th} energy bin in the distribution for E_i, and

P_{ik+1} = value of the PDF corresponding to the upper boundary of the k^{th} energy bin in the distribution for E_i.

By replacing i with i+1 in Eq. (8.1.69) through Eq. (8.1.73), the exit energy $E_{i+1}^{'}$ can be calculated from the secondary energy distribution corresponding to the incident energy E_{i+1}. Subsequently, Eq. (8.1.45) can be used to calculate the exit energy.

Coherent and incoherent elastic scattering

If thermal scattering law data are available for a material, the elastic scattering mechanism may be specified as either coherent or incoherent elastic scattering. Consequently, there is no change in energy resulting from either collision. The secondary energy distribution block for each (E, μ) pair only has one exit energy with a value equal to the incident energy E. Therefore, the procedure for coherent or incoherent elastic scattering reduces to sampling the exit angle cosine. The procedures detailed in Sect. 8.1.7.2.12 are used to sample the exit angle cosine. The general structure of the kinematics data within a section for coherent or incoherent elastic scattering is presented in Table 8.1.32 with the material identifiers and data flags omitted for clarity. Once the angle cosine is selected, the exit energy is set equal to the incident energy.

Table 8.1.32: Kinematics data structure for coherent and incoherent elastic scattering

(E, μ) Data Block					
E	C_2	μ(j)	C_μ(j)	P_μ(j)	j=1,*NPU*
E_1	0	μ(j)	C_μ(j)	P_μ(j)	j=1,*NPU*
E_2	0	μ(j)	C_μ(j)	P_μ(j)	j=1,*NPU*
.	.	.	.	.	
.	.	.	.	.	
.	.	.	.	.	
E_{NE}	0	μ(j)	C_μ(j)	P_μ(j)	j=1,*NPU*
$(E, \mu, E\acute{e}\,)$ Data Block					
E	μ	$E\acute{e}$	$C_{E\acute{e}}$	$P_{E\acute{e}}$	
E_1	$\mu(1)$	E_1	1.0	1.0	
E_1	$\mu(2)$	E_1	1.0	1.0	
.	.	.	.	.	
.	.	.	.	.	
.	.	.	.	.	
E_1	$\mu(NPU)$	E_1	1.0	1.0	
.					
.					
.					
E_{NE}	$\mu(1)$	E_{NE}	1.0	1.0	
E_{NE}	$\mu(2)$	E_{NE}	1.0	1.0	
.	.	.	.	.	
.	.	.	.	.	
.	.	.	.	.	
E_{NE}	$\mu(NPU)$	E_{NE}	1.0	1.0	

Elastic and discrete-level inelastic scattering

There is a one-to-one correspondence between the exiting angle and energy for elastic and discrete level inelastic scattering . Once the exiting angle is selected, the exiting energy is already determined based on the kinematics equations documented in most conventional reactor theory text books. However, an obscure fact is that the exiting energy distributions for discrete-level inelastic reactions and hydrogen elastic scattering can be double valued in the lab system. For discrete level inelastic scattering in the lab system, the range of incident energies that can have a double valued exit energy is given by the following expression:

$$\Delta E_{double} = \frac{Q}{A(1-A)}, \tag{8.1.74}$$

where

Q = the excess of the kinetic energy of the product particles over that of the original particles, and

A = atomic mass ratio of the target mass to the mass of a neutron.

Eq. (8.1.74) provides the size in energy of the double valued region above the threshold energy for the reaction. As indicated by Eq. (8.1.74), the range of energies for a double valued region is inversely proportional to the target mass. Using Eq. (8.1.74) and ENDF/B data, the values of ΔE_{double} can be calculated for all discrete level inelastic scattering reactions for all isotopes. A plot of ΔE_{double} as a function of atomic mass ratio is provided in Fig. 8.1.206 for all possible discrete level inelastic scattering collisions for all isotopes. As shown in Fig. Fig. 8.1.206, the size of the double valued region approaches 1 MeV as the mass ratio decreases. Because ΔE_{double} can be relatively large, the double valued anomaly cannot be ignored in the collision

treatment. The following discussion outlines the properties of the kinematics data and the procedures for treating elastic and discrete level inelastic scattering.

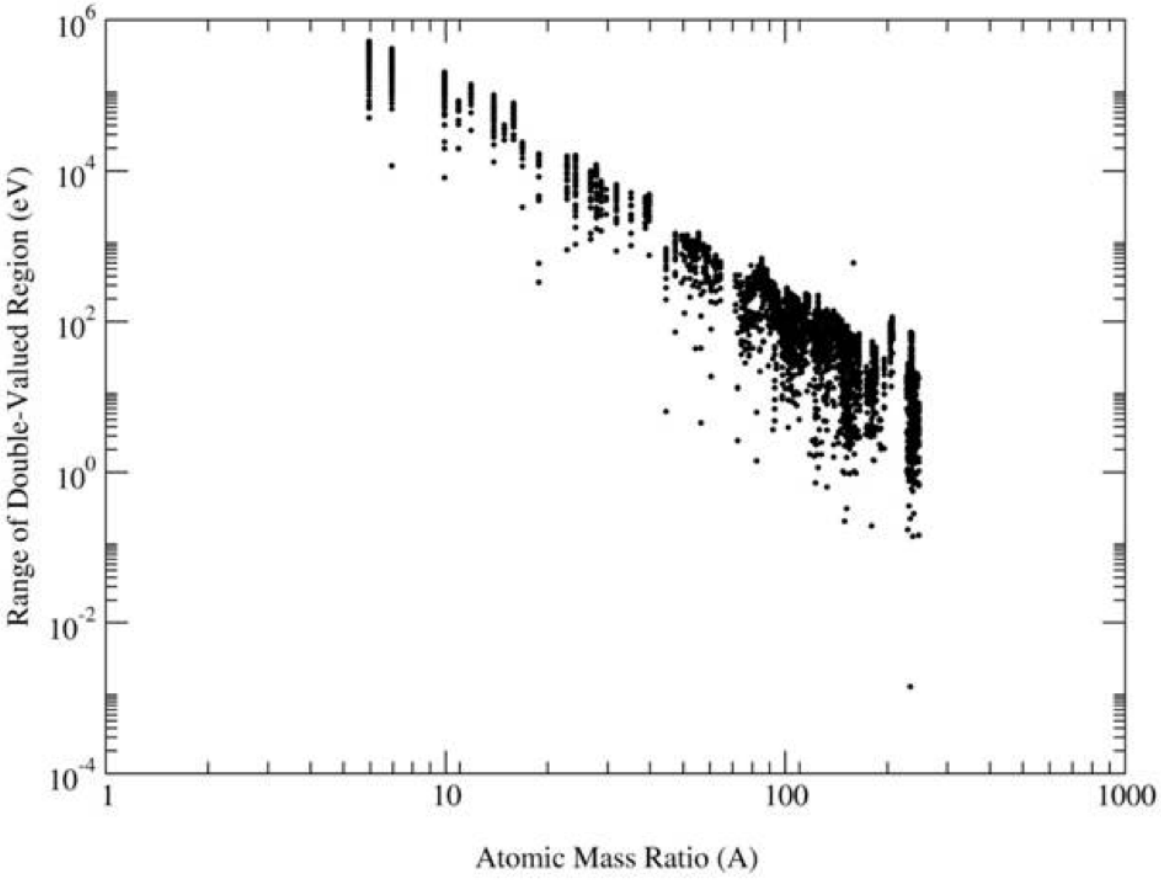


Fig. 8.1.206: Range of incident energies (ΔE_{double}) for double valued region as a function of mass ratio.

As noted previously, *NPU* secondary angle cosines are provided for each incident energy for a reaction, and the cosine distribution can be represented with either equiprobable or nonequiprobable distributions. Depending on the type of distribution, procedures similar to those of Sect. 8.1.7.2.13 are used to sample an equiprobable or nonequiprobable angular distribution for the exit cosine in the lab system.

The angular data blocks for discrete level inelastic scattering and hydrogen elastic scattering are similar in form to other reactions; however, the exit lab cosines for discrete level inelastic scattering are greater than zero in the double-valued region, and an exit lab cosine may appear twice in the distribution with different probabilities. As an example, discrete level inelastic scattering for ^{7}Li is characterized by a double valued exit energy distribution as a function of exit cosine in the lab system. Fig. 8.1.200 provides the exit energy distribution for an inelastic collision that leaves the ^{7}Li nucleus in the first excited state (i.e., MT = 51) in the lab system. For ^{7}Li, the threshold for MT = 51 is 5.4672×10^5 eV, and the double valued region extends from the threshold energy up to 5.58259×10^5 eV; therefore, the incident energy range of the double valued region is ~1.15×10^4 eV. As shown in Fig. 8.1.200, the exit energy distributions are provided for different incident energies within the double valued region and an incident energy above the double valued region. As the incident energy exceeds the maximum energy for which two exiting energies can be produced, the secondary energy distribution becomes single valued (i.e., one exit energy for an exit cosine).

Note that the double valued anomaly does not occur in the center-of-mass system. Therefore, the two exit energies possible for the lab system actually correspond to two different center-of-mass exit cosines. During the preparation of the KENO library, the angular distribution tables for discrete level inelastic scattering and elastic scattering for hydrogen can be constructed in the center-of-mass system and subsequently converted to the lab system. In order to properly construct the angular distribution tables in the lab system, the minimum lab cosine, $\mu^{\min}_{\text{lab}}$, must be determined because exit angles below the minimum lab cosine are not possible.

For discrete level inelastic scattering and elastic scattering, the exit cosine in the lab system can be expressed as a function of the center-of-mass scattering angle, μ_{cm}:

$$\mu = \frac{\gamma + \mu_{cm}}{(1 + 2\gamma\mu_{cm} + \gamma^2)^{1/2}}, \tag{8.1.75}$$

where

$$\frac{1}{\gamma} = \left[A^2 + \frac{A(A+1)Q}{E}\right]^{1/2} \tag{8.1.76}$$

Physically, the quantity $1/\gamma$ represents the ratio of the exit velocity of the neutron in the lab system to the center-of-mass velocity. Moreover, the quantity $1/\gamma$ reduces to A for elastic scattering (i.e., $Q = 0$). The minimum value for μ can be obtained by taking the derivative of Eq. (8.1.77) with respect to μ_{cm} which yields the following expression:

$$\frac{d\mu}{d\mu_{cm}} = \frac{1 + \gamma\mu_{cm}}{(1 + 2\gamma\mu_{cm} + \gamma^2)^{3/2}}. \tag{8.1.77}$$

Setting Eq. (8.1.77) equal to 0 reveals that the minimum lab cosine ($\mu^{\min}_{\text{lab}}$) occurs when $\mu_{\text{cm}} = -1/\gamma$. For elastic scattering, the minimum lab cosine occurs when $\mu_{\text{cm}} = -A$. Because hydrogen is the only nuclide with a mass ratio below 1, hydrogen is the only nuclide that has a double valued exit energy distribution in the lab system for elastic scattering.

During the preparation of the KENO library, the lab distributions for discrete level inelastic scattering and hydrogen elastic scattering will be constructed so that $\mu^{\min}_{\text{lab}}$ is a boundary for an angular bin, and no angular cosines in the lab system will be permitted below $\mu^{\min}_{\text{lab}}$. As shown in Fig. 8.1.207 for ^{7}Li, the exit cosines greater than $\mu^{\min}_{\text{lab}}$ will have two possible exit energies in the double valued region. As a result, the angular distribution table in the library will have angles that appear twice with different probabilities. For example, an angle cosine μ may be present in the table with probabilities P_{m} and P_{n}. Although the value of the angle cosine is the same for both probabilities, the corresponding exit energy will be different for each (μ, *P)* pair. In other words, the probability for an exit cosine determines the location in the exit energy distribution table for selecting the outgoing energy.

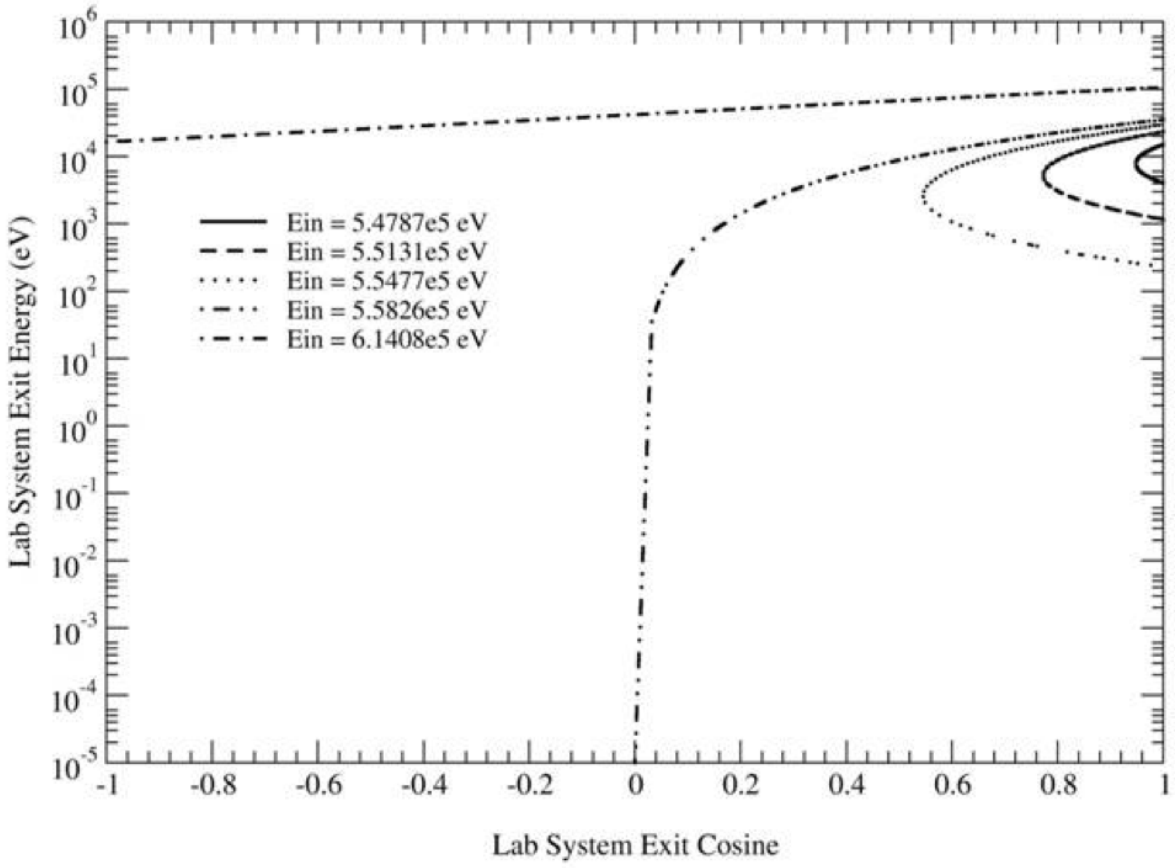


Fig. 8.1.207: ^{7}Li exit energy distribution for the first discrete level inelastic scattering reaction (MT=51). $E_{\text{threshold}} = 5.4672 \times 10^5$ eV; $\Delta E_{\text{double}} = 1.1539 \times 10^4$ eV.

Based on the format of the kinematics data, angle cosine distributions are provided as a function of incident energy. If the incident energy, E, is between E_{i} and $E_{\text{i+1}}$, then the i^{th} table is selected with probability P_{i} and the i+1 table is selected with probability $P_{\text{i+1}}$:

$$P_i = \frac{E_{i+1} - E}{E_{i+1} - E_i}, \tag{8.1.78}$$

$$P_{i+1} = \frac{E - E_i}{E_{i+1} - E_i}. \tag{8.1.79}$$

Once the angular distribution corresponding to the incident energy is selected, the exit angle cosine can be selected using procedures like those in Sect. 8.1.7.2.12 For example, if the i^{th} incident energy is selected, either Eq. (8.1.34) is used to select the a^{th} cosine bin for equiprobable bins, or Eq. (8.1.39) is used for nonequiprobable bins. Subsequently, either Eq. (8.1.36) or Eq. (8.1.41) is used to calculate the exit angle cosine, depending on the type of angular distribution provided.

The coupled secondary angle-energy data blocks (E, μ, E') follow the angular data block in the kinematics data. For elastic scattering and discrete level inelastic scattering, there is one possible exit energy. Therefore, in the exit energy distribution block, a single exit energy ($NPE = 1$) is provided for each (E, μ) pair, and the general structure (i.e., material identifiers and data flags are omitted for clarity) of the secondary energy distribution data block is presented in Table 8.1.33. Because only one exit energy is provided for each (E, μ)

pair, the probability for the exit energy is 1.0. As indicated in Table 8.1.33, a cumulative probability of 1.0 is stored for each exit energy; however, the parameters stored in the PDF location are the power-interpolation parameters needed to interpolate the exit energy.

Table 8.1.33: Secondary energy distribution format for elastic and discrete-level inelastic scattering

E	μ	E'	$C_{E'}$	$p_{E'}$
E_1	μ_1	E'_1	1.0	p_1
E_1	μ_2	E'_2	1.0	p_2
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
E_1	μ_{NPU}	E'_{NPU}	1.0	p_{NPU}
		.		
		.		
		.		
E_{NE}	μ_1	E'_1	1.0	p_1
E_{NE}	μ_2	E'_2	1.0	p_2
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
E_{NE}	μ_{NPU}	E'_{NPU}	1.0	p_{NPU}

When an exiting angle $\dot{\mu}_i$ is sampled between μ_{a} and $\mu_{\text{a+1}}$, the exiting energy values that correspond to μ_{a} and $\mu_{\text{a+1}}$ for the incident energy E_{i} can be used to interpolate the exiting energy E' that corresponds to$\dot{\mu}_i$. Physically, the interpolation procedures correspond to an interpolation along one of the incident energy curves as depicted in Fig. 8.1.205. In order to perform the interpolation, the a^{th} and a+1$^{\text{st}}$ (E, μ) pairs are located in the secondary energy distribution block corresponding to E_{i}. Subsequently, the corresponding a^{th} and a+1 exit energy values (i.e., E'_a and E'_{a+1}) are read from the secondary energy distribution record.

The exiting energy is interpolated with the following relation:

$$E'(\dot{\mu}_i) = E'_a + \left(\frac{\dot{\mu}_i^{P_a} - \mu_a^{P_a}}{\mu_{a+1}^{P_a} - \mu_a^{P_a}} \right) \left(E'_{a+1} - E'_a \right) , \tag{8.1.80}$$

where

P = is a "power-interpolation" factor for the a^{th} cosine bin that is obtained from a fit to the E' distribution as a function of μ.

The interpolation procedure in Eq. (8.1.80) is analogous to the interpolation scheme that has been used successfully to interpolate Bondarenko factors as a function of temperature in the BONAMI module. If the curve is monotonic in nature, the power-interpolation scheme can be used to represent the curvature with a minimum number of points. For elastic and discrete level inelastic scattering, the $E'(\mu)$ distribution is monotonic and is suited for the power interpolation scheme. Note that if p is 1, Eq. (8.1.80) reduces to a linear interpolation in μ-E' space.

Thermal scattering effects

A collision between a neutron and nucleus can be affected by the thermal motion of the target nucleus. Moreover, ENDF provides thermal scattering law data to account for the thermal effects of scattering with a material below 10 eV; however, the scattering law data are only available for a select number materials as shown in Table 8.1.34. If thermal scattering law data are not available, elastic scattering is treated with the free gas approximation in the KENO library. For heavy nuclides with 0 K scattering libraries available, the Doppler Broadening Rejection Correction (DBRC) method is used instead. For DBRC-enabled nuclides, thermal scattering effects are taken into account up to 210 eV (although this parameter can be controlled by the user). More information on DBRC is given in Sect. 8.1.7.2.18.

Table 8.1.34: ENDF/B thermal scattering law data.

Symbol	MAT	Principal Scatterer S(α,β)	Coherent Elastic	Incoherent Elastic	Incoherent Inelastic S(α,β)
Be	26	Be	✓		✓
Bemetal	26	Be	✓		✓
benzine (C_6H_6)	40	H			✓
beo	27	BeO	✓		✓
d_d2o	11				✓
graphite	31	C	✓		✓
h_ch2	37	H		✓	✓
h_h2o	1	H			✓
h_zrh	7	ZrH		✓	✓
lch4	33	H			✓
orthod	13				✓
orthoh	3	H			✓
parad	12				✓
parah	2	H			✓
sch4	34	H		✓	✓
zr_zrh	58	ZrH		✓	✓
o-beo*	28	BeO	✓		✓
o-uo2*	75	O	✓		✓
u-uo2*	76	U	✓		✓

*ENDF/B-VII only.

The continuous energy cross section library is processed to account for the appropriate scattering effects. As a result, the KENO cross section library inherently accounts for thermal scattering using either the thermal scattering law data or the free gas approximation. In other words, the kinematics data based on thermal effects are combined with the fast data to represent the collision kinematics for the material. As noted previously, the kinematics data block for a reaction is divided into sections that describe the collision kinematics for a range of incident energies. As a result, the kinematics data structure has a modular format that is beneficial for data library production. Because of the kinematics data structure, the thermal scattering law data, which are typically applied up to 10 eV, can be processed independently with AMPX to obtain the angle energy probability distributions. Subsequently, one or more sections in the kinematics data block can be used to represent the thermal angle -energy distributions for energies up to 10 eV. Moreover, the fast kinematics data can be represented with one or more sections for energies above 10 eV. As part of the library preparation,

the thermal kinematics data are combined with the fast kinematics data block to complete the collision representation for the entire range of incident energies. Because the thermal scattering effects are treated during the processing of the cross section data for a material, the thermal effects are inherently treated in KENO. The 10 eV cutoff energy for thermal scattering can be input by the parameter **THC** in the parameter input. Using a cutoff value above 10 eV increases the energy range of free gas approximation because AMPX processed thermal scattering law data are available up to 10 eV, and this may increase the runtime in the sampling process while improving the accuracy in the results for some casees. During the Monte Carlo random walk, the procedures provided in Sect. 8.1.7.2.12 and Sect. 8.1.7.2.13 are used to sample the exit angle and energy for the collision.

Doppler broadening rejection correction method

When colliding with a heavy nuclide, the thermal motion of the target nuclide can significantly affect the exit angle and energy of the neutron. The thermal motion of the target nuclide was previously ignored, but this assumption has been proven inadequate by Dagan and Becker [BDL09].

A new sampling equation has been implemented to allow for Doppler broadening of the scattering kernel. Removing this approximation can have a significant impact on the critical eigenvalue of systems by allowing neutrons to be upscattered into absorption resonances. Currently, DBRC is enabled in KENO only for ^{238}U, and this feature can be activated by setting the parameter **DBR**=1.

Doppler broadening methods

CE KENO calculations will perform temperature adjustment of neutron cross sections by default. The cross sections provided with SCALE are typically supplied in 300 K intervals, and additional cross section temperature resolution may be desired to model non-room temperature systems or systems with significant temperature effects.

The capability to perform problem-dependent Doppler broadening of nuclear cross sections has been implemented in KENO. This feature is controlled by the **DBX** parameter, which causes KENO to perform Doppler broadening on the cross sections in all compositions in a problem before simulating particle histories. When this occurs, the cross sections for each material are broadened to the temperature assigned to that material in the `READ COMP` block, as long as the requested temperature is more than 4 K from the library temperature. Within this ± 4 K band, the library temperature is used to avoid difficulties with numerical instabilities in small cross section adjustments. The default for this feature is `DBX=2`, which performs Doppler broadening for both the 1-D and 2-D cross sections. Current options available are as follows:

- `DBX` = 0 – Doppler broadening is not performed. KENO selects the library at the closest temperature. If desired, the user can use the `TTL` parameter to force a job abort if the library temperature is too far from the requested temperature. By default, `TTL=-1`, so the closest library will be used regardless of the temperature difference.
- `DBX=1` – Doppler broadening is performed using a finite difference method on the one-dimensional cross sections that are temperature dependent. Logarithmic interpolation is also used to broaden the probability table data. When this option is used, the collision probabilities are recalculated after the Doppler broadening is complete to ensure consistency.
- `DBX=2` – The same actions as `DBX=1` are performed, and Doppler broadening is also performed on the thermal moderator data by conducting interpolation on the double differential probability data present on the cross section library.

Nuclide level energy unionization is automatically disabled (`M2U=NO`) when Doppler broadening is enabled in order to reduce the memory usage. At this writing, research is in progress to allow KENO to perform on-the-fly cross section Doppler broadening as particle histories are being simulated.

8.1.7.3 Multigroup mode solution procedure

$$\frac{1}{v_g}\frac{\partial \Phi_g}{\partial t}(X,\Omega,t) + \Omega \bullet \nabla \Phi_g(X,\Omega,t) + \Sigma_{\mathrm{tg}}(X)\ \Phi_g(X,\Omega,t) = \mathrm{q_g}(X,\Omega,t), \tag{8.1.81}$$

where

- g is the energy group of interest,
- v_g is the average velocity of the neutrons in group g,
- $\Phi_g(X,\Omega,t)$ is the angular flux of neutrons having their energies in group g, at position X and time t, and
- $\Sigma_{tg}(X)$ is the macroscopic total cross section of the media at position X for group g, corresponding to

$$\Sigma_{tg}(X) = \frac{\int_{\Delta E_g} \Sigma_t(X,E)\Phi(X,E,\Omega,t)dE}{\int_{\Delta E_g} \Phi(X,E,\Omega,t)dE}, \tag{8.1.82}$$

where

- ΔE_g defines group g, and
- $q_g(X,\Omega,t)$ is the total source contributing to energy group g at position X, and time t in direction Ω.

Using the relationship $X' = X - R\Omega$, defining the problem to be time independent using an integrating factor on both sides of Eq. (8.1.81), and defining

$$T(R) = \int_0^R \Sigma_{tg}(X - R'\Omega)dR', \tag{8.1.83}$$

the following equation can be written:

$$\Phi_g(X,\Omega) = \int_O^\infty q_g(X - R\Omega,\Omega)e^{-T(R)}dR. \tag{8.1.84}$$

At this point, the problem becomes an eigenvalue problem. If there is no external source, the source may be defined as

$$q_g(X,\Omega) = \sum_{g'} \int d\Omega' \Phi_{g'}\left(X,\Omega'\right)\ \Sigma_s\left(X, g' \to g, \Omega' \cdot \Omega\right) + \frac{1}{k} Q'_g(X,\Omega), \tag{8.1.85}$$

where

- k is the largest eigenvalue of the integral equation,
- $Q'_g(X,\Omega)$ is the fission source at position X for energy group g and direction Ω (all fission contributions to group g from all energy groups in the previous generation), and
- $\Sigma_s(X, g' \to g, \Omega' \to \Omega)$ is the scattering cross section for scattering at position X from group g' and direction Ω' to group g and direction Ω.

In terms of energy, the scatter can be defined as

$$\Sigma_s\left(X, g' \to g, \Omega' \cdot \Omega\right) = \frac{\int_{\Delta E_g}\int_{\Delta E_{g'}} \Sigma_s\left(X, E' \to E, \Omega' \cdot \Omega\right)\ \Phi\left(X, E', \Omega'\right) \mathrm{d}E' \mathrm{d}E}{\int_{\Delta E_{g'}} \Phi\left(X, E', \Omega'\right) \mathrm{d}E'}\ , \tag{8.1.86}$$

where

ΔE_g is the energy-range-defining energy group g, and

$\Delta E'_g$ is the energy-range-defining energy group g'.

Assuming the fission neutrons to be isotropic, the fission source $Q_{g'}(X, \Omega)$ can be written as

$$Q'_g(X, \Omega) = \frac{1}{4\pi} \sum_{g'} \int_{\Omega'} d\Omega' \Phi_{g'}(X, \Omega') \chi(X, g' \rightarrow g) \nu_{g'}(X) \Sigma_{fg'}(X), \tag{8.1.87}$$

where

- $\chi(X, g' \rightarrow g)$ is the fraction of neutrons born in energy group g from fission in energy group g in the media at position X,
- $\nu_{g'}(X)$ is the number of neutrons resulting from a fission in group g' at position X, and
- $\Sigma_{fg'}(X)$ is the macroscopic fission cross section of the material at position X for a neutron in energy group g'.

Substituting Eq. (8.1.85) into Eq. (8.1.84) yields the following equation:

$$\Phi_g(X, \Omega) = \int_0^\infty dR e^{-T(R)} \left\{ \frac{1}{k} Q'_g(X - R\Omega, \Omega) + \sum_{g'} \left[\int_{\Omega'} d\Omega' \Phi_{g'}(X - R\Omega, \Omega') \Sigma_s(X - R\Omega, g' \rightarrow g, \Omega' \cdot \Omega) \right] \right\} \tag{8.1.88}$$

The definition of k may be given as the ratio of the number of neutrons in the (n + 1)*th* generation to the number of neutrons in the n*th* generation or the largest eigenvalue of the integral equation. Using Eq. (8.1.87), Eq. (8.1.88) can be written as

$$\Phi_g(X, \Omega) = \int_0^\infty dR e^{-T(R)} \left\{ \sum_{g'} \frac{1}{k} \int_{\Omega'} \nu_{g'}(X - R\Omega) \Sigma_{fg'}(X - R\Omega) \chi(X - R\Omega, g' \rightarrow g) \Phi_g(X - R\Omega, \Omega') \frac{d\Omega'}{4\pi} + \sum_{g'} \left[\int_{\Omega'} d\Omega' \Sigma_{tg'}(X - R\Omega, \Omega') \Sigma_s(X - R\Omega, g' \rightarrow g, \Omega' \cdot \Omega) \right] \right\} \tag{8.1.89}$$

Writing Eq. (8.1.89) in generation notation, multiplying and dividing certain terms by $\Sigma_t(X)$, and multiplying both sides of the equation by $\nu_g(X)\Sigma_{fg}(X)$ yields the following equation, which is solved by KENO:

$$\frac{\nu_g(X)\Sigma_{fg}(X)}{\Sigma_{tg}(X)} \Sigma_{tg}(X) \Phi_{g,n}(X, \Omega) = \frac{\nu_g(X)\Sigma_{fg}(X)}{\Sigma_{tg}(X)} \Sigma_{tg}(X) \int_0^\infty dR e^{-T(R)} \left\{ \frac{1}{k} \sum_{g'} \left[\int_{\Omega'} \frac{\nu_{g'}(X - R\Omega)\Sigma_{fs'}(X - R\Omega)}{\Sigma_{tg'}(X - R\Omega)} \chi(X - R\Omega, g' \rightarrow g) \Sigma_{tg'}(X - R\Omega) \Phi_{g',n-1}(X - R\Omega, \Omega') \frac{d\Omega'}{4\pi} \right] + \sum_{g'} \left[\int_{\Omega'} \frac{\Sigma_s(X - R\Omega, g' \rightarrow g, \Omega' \cdot \Omega)}{\Sigma_{tg'}(X - R\Omega)} \Sigma_{tg'}(X - R\Omega) \Phi_{g',n}(X - R\Omega, \Omega') d\Omega' \right] \right\} \tag{8.1.90}$$

where n indicates the n*th* generation and n - 1 is the (n - 1)*th* generation. Note that the left-hand side of the equation — $\nu_g(X) \sum_{fg}(X) \Phi_{g,n}(X, \Omega)$ — is the fission production for the n*th* generation.

The solution strategy used by KENO solves Eq. (8.1.90) by using an iterative procedure. The fission production at point X in energy group g due to neutrons in the (n - 1)*th* generation, normalized to the system multiplication, is

$$\frac{1}{k}\sum_{g'}\int_{\Omega'}\frac{\nu_{g'}(X)\Sigma_{fg'}(X)}{\Sigma_{tg'}(X)}\chi(X,g'\rightarrow g)\Sigma_{tg'}(X)\Phi_{g',n-1}(X,\Omega')\frac{d\Omega'}{4\pi}. \tag{8.1.91}$$

The collision points used in KENO are chosen by selecting path lengths from the distribution

$$e^{-T(R)},$$

which is the probability of transport from any position X - RΩ to position X.

The first collision density of neutrons in group g per unit solid angle about Ω resulting from the fission source produced by the (n - 1) generation, normalized to the system multiplication, is

$$\frac{\Sigma_{tg}(X)\int_0^\infty dRe^{-T(R)}\frac{1}{k}\int_{\Omega'}\sum_{g'}\frac{\nu_{g'}(X-R\Omega)\Sigma_{fg'}(X-R\Omega)}{E_{tg'}(X-R\Omega)}}{\chi(X-R\Omega,g'\rightarrow g)\Sigma_{tg'}(X-R\Omega)\Phi_{g',n-1}(X-R\Omega,\Omega')\frac{d\Omega'}{4\pi}}. \tag{8.1.92}$$

The scattering source at position X emerging in group g and direction Ω resulting from previous collisions in the same generation, is

$$\sum_{g'}\int_{\Omega'}\frac{\Sigma_s(X,g'\rightarrow g,\Omega'\cdot\Omega)}{\Sigma_{tg'}(X)}\Sigma'_{tg}(X)\Phi_{g',n}(X,\Omega)d\Omega' \tag{8.1.93}$$

The collision density in group g, per solid angle about Ω is

$$\Sigma_{tg}\int_0^\infty dRe^{-T(R)}\sum_{g'}\int_{\Omega'}\frac{\Sigma_s(X-R\Omega,g'\rightarrow g,\Omega'\cdot\Omega)}{\Sigma_{tg'}(X-R\Omega)}\Sigma_{tg'}(X-R\Omega)\Phi_{g',n}(X-R\Omega,\Omega')\,d\Omega' \tag{8.1.94}$$

The total collision density times $\frac{\nu_g(X)\Sigma_{fg}(X)}{\Sigma_{tg}(X)}$ is the relationship from which KENO picks the source points for the next generation.

Collision treatment in KENO

A collision occurs in a geometrical region when a history exhausts its mean-free-path length within the boundaries of the region. For each collision, the absorbed weight and the fission weight are tabulated, then the weight is modified by the nonabsorption probability. This new weight is checked for splitting and Russian roulette, and if it survives, the history is scattered. A new energy group is selected from the cumulative transfer probability distribution. This group-to-group transfer determines an angular scattering distribution, usually expressed as a Legendre expansion of the cross section transfer array. A set of discrete angles and probabilities are generated by a generalized Gaussian quadrature procedure, preserving the moments of the Legendre expansion of the angular scattering distribution. KENO treats P_0 and P_1 Legendre expansions as special cases. If the scattering distribution is isotropic, a flag is set to randomly select new direction cosines from an isotropic distribution, instead of using discrete scattering angles. If the distribution is a P_1 expansion, KENO randomly selects the cosine of the scattering angle according to

(1) $|\bar{\mu}| < \frac{10^{-10}}{3}$: scattering distribution is isotropic,

(2) $|\bar{\mu}| \leq 1/3 : \mu = \left(\sqrt{1+6\zeta\bar{\mu}+(3\bar{\mu})^2}-1\right)/3\bar{\mu}$,

or

(3) $|\bar{\mu}| > 1/3 : \mu = \zeta(1 - |\bar{\mu}|) + \bar{\mu}$

where ζ is a uniform random variable between -1 and +1 and

$\bar{\mu}$ is the mean cosine of the scattering angle.

Otherwise, KENO randomly selects one of the discrete scattering angles (μ). New direction cosines are then calculated according to the following relationships where u, v, and w are the initial direction cosines and u', v', and w' are the direction cosines after the collision:

$$u' = u\cos\Psi - \sqrt{v^2 + w^2}\sin\Psi\cos\eta$$

$$v' = v\cos\Psi + \frac{uv}{\sqrt{v^2+w^2}}\cos\eta\sin\Psi - \frac{w}{\sqrt{v^2+w^2}}\sin\Psi\sin\eta$$

$$w' = w\cos\Psi + \frac{uw}{\sqrt{v^2+w^2}}\cos\eta\sin\Psi + \frac{v}{\sqrt{v^2+w^2}}\sin\Psi\sin\eta$$

where

$\sin\psi = \sqrt{1-\mu^2}$,

$\cos\psi = \mu$ = cosine of the scattering angle,

η = a random azimuthal angle between 0 and 2π.

Fission point selection

In order for a fission to occur, a neutron must first have a collision. The fission weight, fisw, is defined as the neutron weight, wt, times the ν-fission probability, fnfp:

$$fisw = wt \times fnfp \tag{8.1.95}$$

Two important variables used in the processing of fission points are (1) FWR, which is defined as the fission weight, fisw, divided by a random number, and (2) RAKBAR, which is defined as a factor times the running average value of k-effective, AKBAR. This factor is a function of the square root of the number of neutrons per generation and was chosen because it usually produces an adequate number of independent fission points and does not produce so many that an excessive amount of time is spent choosing from the fission points produced.

The following procedure for generating fission points is repeated until FWR is less than RAKBAR. A fission point is generated only if FWR is greater than RAKBAR. Multiple fissions at the same point are allowed only if fisw is greater than RAKBAR. If fisw is greater than RAKBAR, a fission point is stored with FWR set equal to RAKBAR divided by a random number and fisw is decremented by RAKBAR. Then the energy group of fission is chosen randomly from the fission spectrum of the mixture in which the fission occurred. The energy group of fission, the X, Y, and Z position, the location of the unit within the array, the region number, the value of FWR, the region number of the array boundary, and the nesting data for holes and/or arrays are stored in the fission bank. The quantity of fission points kept to be used as fission positions for the next generation is limited to the number of positions in the fission bank (input parameter `NFB=`). Typically `NFB` is equal to the input parameter `NPG`, the number of neutrons per generation. If a fission occurs and the fission counter is less than `NFB`, the fission point information is stored in the fission bank. If a fission occurs and the fission counter is greater than or equal to the number per generation, a search is made to find the smallest stored value of FWR. If FWR of the newly fissioned neutron is less than the smallest FWR in the table, it is discarded. Otherwise, the information from the newly fissioned neutron replaces that associated with the smallest value of FWR found in the table.

When the next generation is ready to be processed, data are transferred from the fission bank into the neutron bank to be used as starting positions for the fission neutrons. If more than NPG neutrons are saved in the fission bank, `NPG` of those having the highest values of FWR will be used. If too few fission positions were stored (less than the number per generation), a warning message to that effect (K?-132) is printed, and additional fission points are randomly chosen from those that were stored until the number of fission points available to start the next generation is equal to the number of neutrons per generation.

Biasing or weighting

In order to minimize the statistical deviation of k-effective per unit computer time, KENO uses weighted tracking rather than analog tracking. Weighted tracking accounts for absorption by reducing the neutron weight rather than allowing the neutron history to be terminated by absorption. To prevent expending excessive computer time tracking low-weight neutrons, Russian roulette is played when the weight of the neutron drops below a preset weight, WTLOW. Neutrons that survive Russian roulette are assigned a weight, WTAVG. The value of WTLOW and WTAVG can be assigned as a function of position and energy. The values used by KENO are

DWTAV = 0.5, the default value of WTAVG,

WTAVG = DWTAV, the weight given a neutron that survives Russian roulette, and

WTLOW = WTAVG/3.0, the value of weight at which Russian roulette is played.

A study [Hof82] by Hoffman shows these default values to be reasonable for bare critical assemblies. Fig. 8.1.208 from this study shows the analytic relationship between the variance and WTLOW when WTAVG is 0.5. Note that the default value of 0.167 for WTLOW is very close to the minimum point on the curve. Experimental results of actual Monte Carlo calculations[7] provide further assurance that 0.167 is an optimum choice for WTLOW when WTAVG is 0.5.

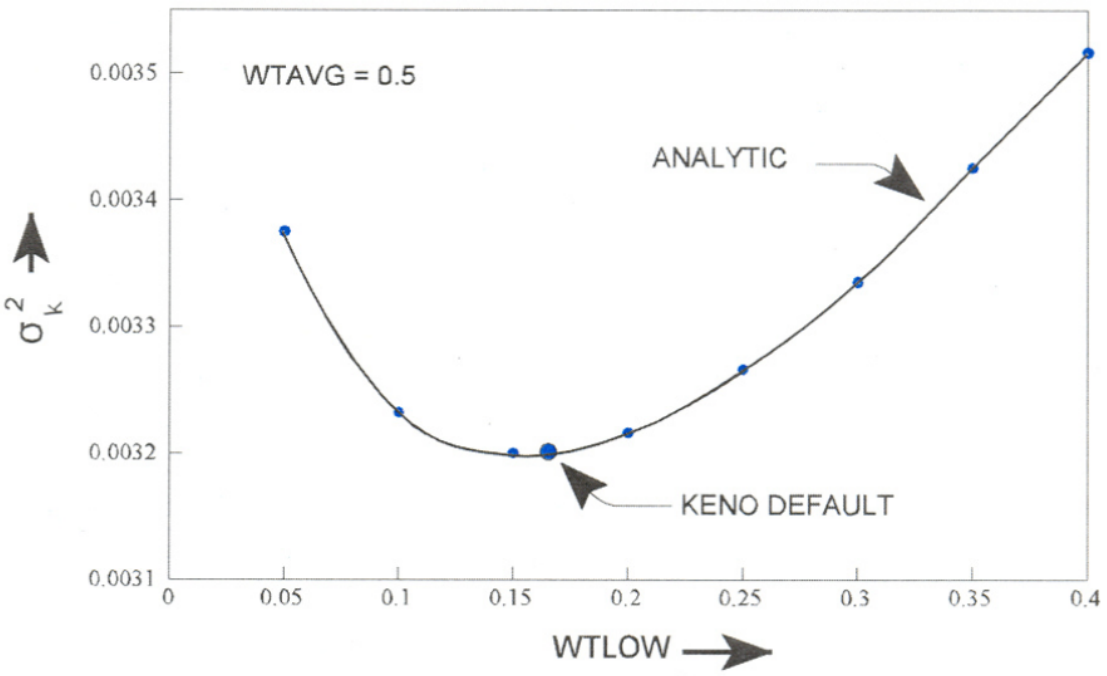


Fig. 8.1.208: Analytic estimate of the relationship between WTLOW and the variance, σ_k^2, when WTAVG is 0.5.

Fig. 8.1.209, also from the Hoffman study, shows the analytic relationship between the variance and the value chosen for WTAVG for a value of WTLOW = 0.167. Although the KENO default value for WTAVG is not the optimum, a close examination of the data shows that the variance was changing relatively slowly as a function of WTAVG. Even though this study shows a value near 0.26 to be optimum for this system, further studies of other systems are needed before changing the default value of WTAVG from 0.5 as it has been used in previous versions of KENO.

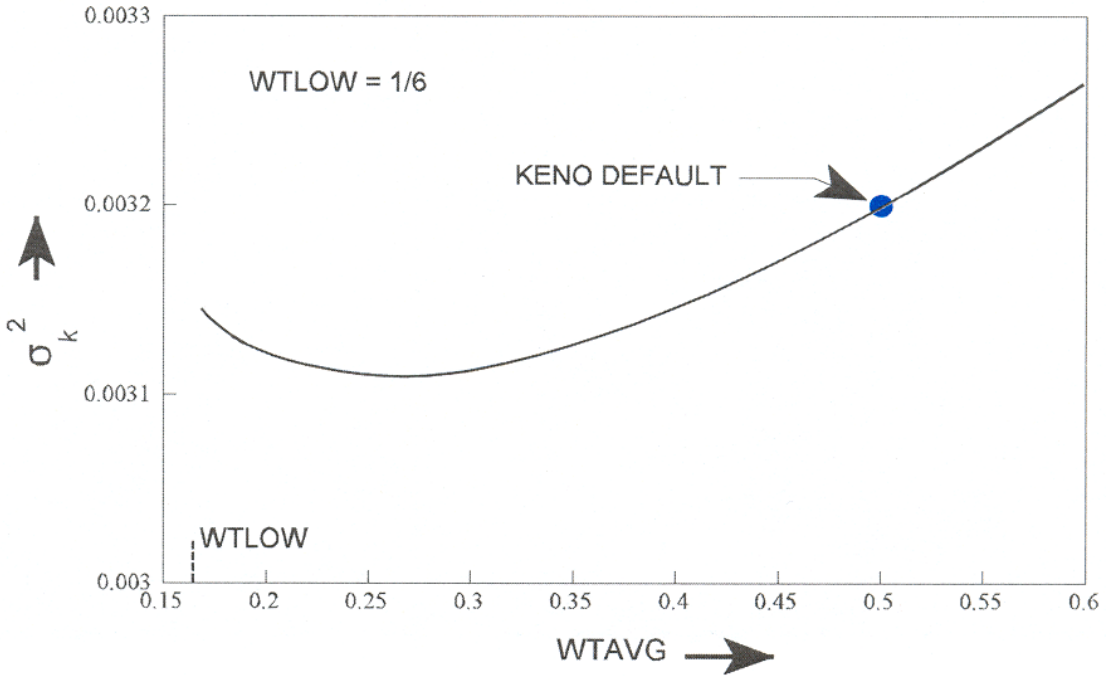


Fig. 8.1.209: Analytic estimate of the relationship between WTAVG and the variance, σ_k^2, when WTLOW is $^1/_6$.

Inside a fissile core, the importance of a neutron is a slowly varying function in terms of energy and position. Hence, for many systems, the standard defaults for WTLOW and WTAVG are good values to use. For reflectors, however, the worth of a neutron varies as a function of distance from the fissile material and as a function of energy. As a neutron in the reflector becomes less important relative to a neutron in the fissile region, it becomes desirable to spend less time tracking it. Therefore a space- and energy-dependent weighting or biasing function is used in KENO to allow the user to minimize the variance in k-effective per unit of tracking time. When a biasing function is used in a reflector, it becomes possible for a neutron to move from one importance region into another in which the WTLOW is greater than the weight of the neutron. When this occurs, Russian roulette is played to reduce the number of neutrons tracked. When the reverse occurs, that is, when the neutron moves to a region of higher importance, its weight may be much higher than WTAVG for that region. When the weight of the neutron is greater than a preset value, WTHI, the neutron is split into two neutrons, each having a weight equal to one-half the weight of the original neutron. This procedure is repeated until the weight of the split neutron is less than WTHI. The default value for WTHI is WTAVG*3.0. WTHI is the weight at which splitting occurs.

The weighting or biasing function for a given core material and reflector material can be obtained by using the adjoint solution from S_n type programs for a similar (usually simplified) problem. This adjoint flux gives the relative contribution of a neutron at a given energy and position to the total fissions in the system. The weighting function for KENO is thus proportional to the reciprocal of the adjoint flux. Although such a function can be difficult to obtain, the savings gained makes the effort worthwhile for many of the materials that are frequently used as reflectors. Biasing functions [ODellA87] have been prepared for several reflector materials commonly used in KENO calculations. The use of biasing to minimize the variance in k-effective per unit of computer time will usually increase the variance in other parameters such as leakage or absorption

in the reflector.

Differential albedos

Arrays reflected by thick layers of material having a small absorption to the scattering ratio may require large amounts of computer time to determine k-effective[Dem99a] because of the relatively long time a history may spend in the reflector. A differential albedo technique was developed for use with the KENO codes to eliminate tracking in the reflector. This involves returning a history at the point it impinges on the reflector and selecting an emergent energy and polar angle from a joint density function dependent upon the incident energy and polar angle. The weight of the history is adjusted by the functional return from the reflector, which is also based on the incident energy and angle.

The characteristics of a differential albedo emulate the attributes of the reflector material and are independent of the material or materials adjacent to the reflector. Thus, a differential albedo that is generated for a given reflector material can be used with any array, regardless of the type of fuel or fissile material contained within the array.

For many calculations involving reflected arrays of fissile material, the differential albedo treatment is a powerful tool that can significantly reduce the computing time required to determine k-effective. The savings will vary depending on the importance of the reflector to the system. A substantial effort is required to generate a differential albedo, but the savings gained were well worth the effort for commonly used reflector materials. The savings are not worth the applicability questions on modern computing platforms; the differential albedo capability is maintained only for backwards compatibility.

To generate the differential albedo information for a material, a fixed-source calculation must be made for each incident energy and angle. The data presently available for use with KENO were generated by 1-D discrete ordinates calculations for slab geometry representing infinite slabs. Consequently, for a finite reflector, these data will not correctly treat histories that enter the reflector near an edge. Past experience with differential albedo reflectors indicates that k-effective appears to be conservative for small faces and will tend toward the correct result as the face becomes large relative to the area near the corners. Care must be taken to ensure that any surface to which a differential albedo is applied is large enough that the errors at the edges can be ignored.

Because differential albedos are expensive and time consuming to generate, those corresponding to the Hansen-Roach 16-energy-group structure are the only differential albedos available for use with KENO at this writing. In the past, their use was limited to problems using cross sections having the Hansen–Roach 16-energy-group structure. KENO extends the use of differential albedos to other energy-group structures by allowing appropriate energy transfers. This is accomplished by creating lethargy boundary tables for the albedo group structure and the cross section group structure and determining the lethargy interval corresponding to the desired transfer (cross section group structure to albedo group structure or vice versa) based on a uniform lethargy distribution over the interval. When the energy-group boundaries of the cross sections and albedos are different, the results should be scrutinized by the user to evaluate the effects of the approximations.

8.1.7.4 KENO Geometries

KENO V.a geometry is restricted to the use of specific shapes. These shapes are called geometry regions or regions. Allowed shapes in KENO V.a are cubes, cuboids (rectangular parallelepipeds), spheres, cylinders, hemispheres, and hemicylinders. These shapes must be oriented along orthogonal axes and cannot be rotated in KENO V.a. They can be translated. Hemispheres and hemicylinders are not limited to half spheres and half cylinders; the definitive plane can be positioned by entering a chord. The value of this chord can range from the positive magnitude of the radius (giving a complete sphere or cylinder) to the negative magnitude of the radius (giving a zero volume, nonexistent sphere or cylinder).

KENO-VI geometry can model any geometric shape that can be described using quadratic equations. These geometric shapes are stacked together forming regions. The set of regions is then used to build units. A set of predefined shapes that include cones, cuboids (rectangular parallelepipeds), cylinders, dodecahedrons, ecylinders (extruded elliptical cylinders), ellipsoids, hexprisms, hoppers, parallelepipeds, planes, rhombohedrons, spheres, wedges (triangular prisms), as well as others is used to construct regions. In addition, the keyword `QUADRATIC` is provided which allows additional shapes to be constructed by specifying the quadratic equations that describe the shape. These shapes can be rotated and translated to any orientation and position within their respective units. Hemispheres and hemicylinders can be constructed using spheres and cylinders with a chord. Regions are rotated by providing the nonzero angles associated with the Euler X-convention.

A major restriction applied to KENO V.a geometry is that intersections are not allowed. Furthermore, each successive geometry region must completely enclose the preceding region. Tangency and shared faces are allowed. The volume of a region is the volume of the specified shape minus the volume of the preceding region shape and any holes contained in the region. To alleviate the complete enclosure restriction, KENO V.a allows multiple sets of geometry regions, with each set independently governed by this restriction. Each set of these multiple geometry regions is called a *unit*. Units can be stacked together in a 3-D rectangular parallelepiped called an *array* or *lattice*, just as children's blocks can be stacked. Units that are to be stacked together in this manner must have a rectangular parallelepiped outer region, and the adjacent faces of adjacent units must be the same size and shape. An array can be treated as a building block and used as a unit within another array.

A major improvement in KENO-VI is the ability to intersect regions. Region volumes are no longer calculated due to the complexity involved with intersecting regions. Each set of multiple geometry regions is called a unit. KENO-VI allows multiple sets of geometry regions (i.e., units), and each set has an independent coordinate system. A global unit must be specified for every problem, including single-unit problems. Units with cuboidal outer boundaries where the adjoining faces have the same dimensions can be stacked together in a 3-D rectangular parallelepiped called an *array* or *lattice*, just as children's blocks can be stacked. Unlike KENO V.a, units having hexagonal or dodecahedral outer boundaries where the adjoining faces have the same dimensions can also be stacked together in an array. An array boundary must be specified that either coexists with the outermost edge of the array or that is entirely within the array. The array boundary can be any shape that is definable using quadratic equations. An array can be treated as a building block and used as a unit within another array.

The use of holes in KENO allows a unit to be emplaced within another unit. This feature allows the addition of a complex structure, previously defined as a unit, to be directly placed within another unit. In KENO V.a a hole is not allowed to intersect other holes or regions, but holes may intersect multiple regions in KENO-VI.

Multiple arrays can be described in KENO. The global array in an unreflected problem is the outermost array in the problem geometry description. The global array in a reflected problem is the array referenced by surrounding geometry regions following the last array placement description that does not immediately follow a unit number description. In KENO-VI, the outermost boundary is always specified as the global unit boundary. If the outermost boundary is to be the array boundary, a global unit must still be specified with the global unit boundary coinciding with the array boundary. Unlike past versions of KENO, KENO-VI cannot run a single unit problem without specifying a global unit.

8.1.7.5 Fluxes

Fluxes are computed in KENO with a track length estimator. The scalar flux in region z for energy group g for a single generation is computed as

$$\Phi_{g,z} = \frac{\sum_{k=1}^{K} W_{k,z} l_{k,z}}{V_z \sum_{k=1}^{K} W_{k,0}}, \tag{8.1.96}$$

where

$l_{k,z}$ = distance traversed by particle k while within region z and energy group g,

$W_{k,z}$ = weight of particle k while traversing region z,

V_z = volume of region z,

$W_{k,0}$ = initial weight of particle k, and

K = total number of histories in the generation.

The average fluxes for all active generations and the standard deviation in the averages are also computed. Scalar fluxes computed by KENO are reported in units of neutrons per cm^2 per initial source neutron.

KENO is also capable of computing the angular flux for a level symmetric quadrature set. The angular flux for energy group g in region z for quadrature direction n is computed as

$$\Phi_{g,z}^{n} = \frac{\sum_{k=1}^{K} W_{k,z} l_{k,z,n}}{V_z \sum_{k=1}^{K} W_{k,0}}, \tag{8.1.97}$$

where

$l_{\mathrm{k,z,n}}$ = the distance traversed by particle k while within region z and energy group g within the quadrature direction n.

The angular flux can be expanded to flux moments using an appropriate spherical harmonics expansion. KENO has even number symmetric level quadrature sets S_2-S_{16}.

Flux moments can also be computed directly in KENO. The *jth* moment, which corresponds to real valued spherical harmonics functions [Dem99b] for a single generation for energy group g in region z, is computed as

$$\phi_{g,z}^{j} = \frac{\sum_{k=1}^{K} R_k^j W_{k,z} l_{k,z}}{V_z \sum_{k=1}^{K} W_{k,0}}, \tag{8.1.98}$$

where

R_k^j = real valued spherical harmonics function for moment index j corresponding to the direction of particle k.

KENO offers the option of computing the angular fluxes and flux moments using a transformed coordinate system so that the moments are based on a polar rather than a Cartesian position vector. This is a 3-D extension of the 1-D method for calculating the flux moments in terms of Legendre polynomials based only on the direction cosine with respect to the spatial coordinate.

Here, $\hat{i}$, $\hat{j}$, and $\hat{k}$ represent the directional coordinate system axes, μ, η, ξ represent the direction cosines, and θ and *rho*represent the polar and azimuthal angles of the "normal" coordinate system. The same symbols

primed represent the transformed coordinate system. The transformed polar or $\hat{k}'$ axis is co-linear, with the position vector $\vec{r}$ directed from the center of the region for which moments are desired to the point at which the flux tally occurs. By using the center of the region as a reference point, consistency of the moment calculation is assured with differing models of the same system. The $\tilde{i}'$ and $\tilde{j}'$ axes are chosen to form an orthogonal coordinate system, with $\tilde{i}'$ held in the plane formed by $\hat{i}$ and $\hat{j}$. The use of constraints other than the restriction of $\tilde{i}'$ to this plane may be explored in future studies. If an additional constraint is not placed on either $\tilde{i}'$ or $\tilde{j}'$, the transform would be able to rotate about $\hat{k}'$, and the consistency of consecutive transformations of the same direction could not be assured. With the transform computed, the position and direction of travel of the particle remain unchanged, but the quadrature directions and/or spherical harmonics terms are calculated using the transformed coordinate system. With the direction cosines consistently transformed for each history, the new polar and azimuthal angles can be computed and the spherical harmonics functions can be calculated for each history.

KENO offers a mesh flux tally in which the fluxes are computed in a user-defined cubic mesh superimposed on the geometry model. The mesh is defined from the origin of the global unit and is oriented with the Cartesian axes. At the end of each particle track, appropriate track lengths are assigned to the mesh intervals that were crossed for the region in which the track occurred. The length of the actual particle track is equal to the sum of the lengths assigned to the meshes encountered by the particle track. Because the coordinates of the mesh are defined in terms of the global unit, fluxes for regions in repeated units (i.e., units in arrays or holes) may be stored separately for each occurrence of the unit. The volumes of each region in each mesh interval are computed so that the fluxes are appropriately normalized on a per unit volume basis, consistent with Eq. (8.1.96). When the mesh flux option is activated, all requested fluxes, scalar, angular and/or moment, are computed for each mesh interval.

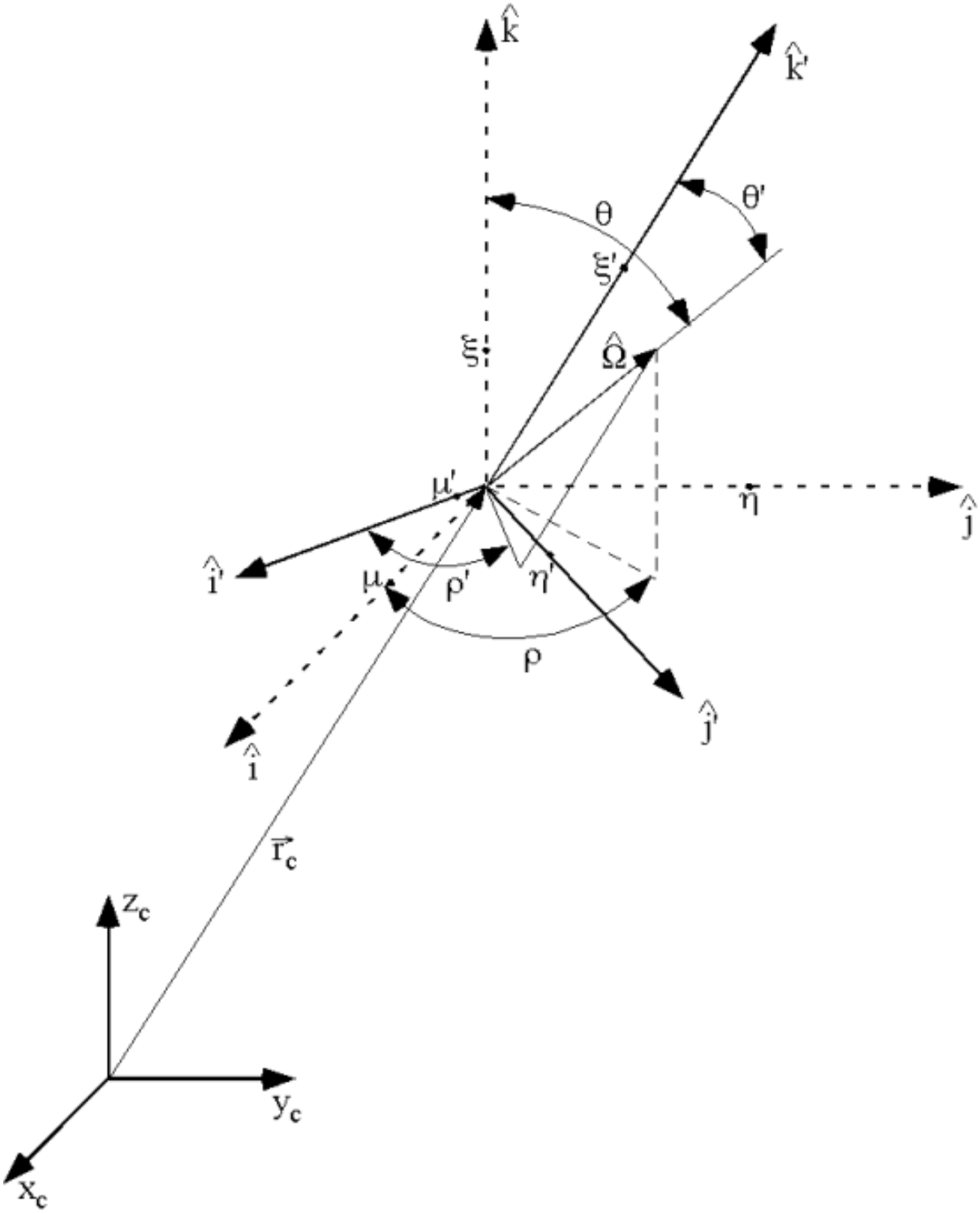


Fig. 8.1.210: Coordinate transform used for quadrature direction and spherical harmonics terms.

8.1.7.6 Reaction Rate and Few Group Micro Cross Section Calculations

A few-group microscopic reaction cross section calculation capability is included in the KENO codes to provide this data especially for the CE depletion calculations in SCALE. This new method produces multigroup cross sections and reaction rates directly in CE mode calculations rather than using a post-processing approach. In each generation, KENO uses track length estimators for the reaction rate tallies for all isotopes in each specified regions. At the end of each generation, a subsequent calculation is performed to compute few group microscopic reaction cross sections for all isotopes in a region as the ratio of the computed reaction rates to the flux averaged over this cell. Finally, KENO computes mean values and statistical uncertainties for all these quantities, and saves them in a file, which could be used by the depletion modules in SCALE.

Reaction tally calculations can be enabled by entering the required data in the **reaction data** block described in Sect. 8.1.3.15.

8.1.7.7 Source Convergence Diagnostics

Fission source convergence detection techniques are implemented in KENO to provide improved confidence in the computed k_{eff} and tally results. The dominance ratio of a system is equal to the ratio of the eigenvalue of the first nonfundamental mode to that of the fundamental mode, k_1/k_0. The dominance ratio is strongly related to the convergence rate of the fission source for systems, and problems with larger dominance ratios will require more inactive generations to produce reliable k_{eff} and reaction rate estimates. A converged k_{eff} estimate does not necessarily guarantee a converged fission source distribution, and the fission source and flux distribution may continue to evolve well after k_{eff} convergence is reached. Convergence of the fission source distribution is necessary to ensure that all regions containing fissile material in a model are adequately represented in the final k_{eff} estimate. Typically, the k_{eff} estimate converges much faster than the fission source and neutron flux distributions, and simulations may undersample important regions and produce inaccurate flux and reaction rate estimates if the user includes only enough inactive generations for the eigenvalue to converge. This is particularly important in models where one region is physically small but substantially more reactive than others (i.e. the k-effective of the World problem [Whi71]). The highly reactive region may not be adequately sampled, which may result in an underprediction of k_{eff}. Thus, it is essential to monitor the convergence of both the fission source distribution and k_{eff}; versions of KENO included in SCALE versions prior to SCALE 6.2 only monitored the convergence of k_{eff}, but diagnostic tools to identify unconverged fission source distributions are included in SCALE 6.2 and beyond.

Note: Due to limited experience with these tests in KENO, the Shannon entropy test results should be used as a supplement to guarantee that a problem's fission source is converged. Failure of Shannon entropy tests typically indicates insufficient convergence of fission site source distributions. Whenever Shannon entropy tests report failure, users should review the plots of k_{eff} as a function of generation skipped and k_{eff} as a function of generation completed (Sect. 8.1.5.29 and Sect. 8.1.5.30) to ensure that the k_{eff} value is adequately converged. If it is not clear that the k_{eff} value is converged, users should rerun the calculation using a different number of neutrons per generation (using the `NPG` parameter) or using a different random number seed value (using the `RND` parameter) or a different starting source term distribution (using the `READ START` block).

It may also be possible to pass the Shannon entropy tests by changing the source convergence mesh (using the `SCD` parameter and the `READ GRID` block) or changing the number skipped generations (using the `NSK` parameter).

For KENO calculations that are used for other purposes (such as producing neutron flux and reaction rate distributions, fuel depletion calculations, or fission distributions for radiation transport calculations) in which

the spatial dependence of the neutron flux and fission source may be more important, KENO calculations with failed Shannon entropy tests should be rerun because these spatially dependent distributions may not be adequately converged.

Shannon Entropy Statistics

The fission source distribution, S, is measured at the beginning of each generation using given spatial meshes. The state of source distribution at the j^{th} generation can then be characterized by Shannon entropy:

$$H(S^j) = -\sum_i^M [S_i^j \log_2(S_i^j)], \tag{8.1.99}$$

where S_i^j is the fraction of the fission source distribution from the j^{th}' generation tallied in the spatial bin i, and M is the total number of spatial bins (typically represented using a spatial mesh). The value of H ranges from a maximum value of $\log_2 M$ when the source distribution is uniform, and to a minimum value of zero when the whole source is located in a single bin. By computing H for each generation, the distribution of the fission source is used to generate a score for H for each generation, and the convergence of the fission source can be assessed by tracking the convergence and randomness of the values of H over each successive generation. Random fluctuations in the generation-to-generation Shannon entropy tallies make it difficult to determine if and when the Shannon entropy of a system has converged, so several metrics and tests are typically applied to examine the Shannon entropy and identify convergence in a system [Uek05].

The first test for fission source convergence is a comparison between the mean square of posterior relative entropy, and the centered mean square Shannon entropy. The posterior relative entropy, $D(S^j \parallel T)$, is defined as the statistical distance between the binned fission source $S_i^j$and the average fission source over the second half of the active generations, T, and is given by

$$D(S^j \parallel T) = \sum_i^M [S_i^j \log_2(S_i^j / T_i)], \tag{8.1.100}$$

where T is defined by

$$T_i = \frac{2}{N} \sum_{j=N/2+1}^{N} S_i^j, \tag{8.1.101}$$

where N is the number of active generations. The posterior relative entropy is non-negative and becomes zero only when $S^j = T$ and achieves its maximum value when S^j is uniform.

The mean square posterior relative entropy is defined as

$$msq(D) = \frac{2}{N} \sum_{j=N/2+1}^{N} D(S_i^j \parallel T_i)^2, \tag{8.1.102}$$

and the centered mean square Shannon entropy is defined as

$$cmsq(H) = \frac{2}{N-2} \sum_{j=N/2+1}^{N} [H(S_i^j) - \bar{H}]^2, \tag{8.1.103}$$

where $\bar{H}$ is defined as

$$\bar{H} = \frac{2}{N} \sum_{j=N/2+1}^{N} H(S_i^j). \tag{8.1.104}$$

This test compares the measure of fluctuations of Shannon entropy (*cmsq(H)*) with the measure of the magnitude of the penalty incurred by assuming T (*msq(D)*) and states that the fission source is converged if *msq(D) < cmsq(H)*. This test is useful to determine whether the source is converged by the end of the calculation. Failure of this test indicates that the fission source was still moving during the active generations, which indicates that the number of skipped generations was not sufficient to converge the fission source.

The second test is given by

$$D(S^j \parallel T) - [H(T - H(S^j))] \leq \varepsilon \tag{8.1.105}$$

where $\mathcal{E}$ has been set to 0.1. This test verifies that the Shannon entropy of each active generation does not vary significantly from the average Shannon entropy of the system and should be valid over all active generations. For example, this test could detect an unconverged fission source for a problem with an undersampled, highly fissile fuel region, as any neutrons that reached this region would produce an outlier in the Shannon entropy tally for that generation. This test is especially useful for reporting the generation at which the source converged. Once the converged generation is determined, the user can repeat the calculation with the number of inactive generations (skipped generations) set to be the identified generation or greater to ensure that only contributions from generations with a converged source contribute to flux and reaction rate tallies. This test complements the first test since it will not be affected by whether or not the binned source is uniform.

The last test calculates whether

$$H_\eta = H(S^j) - H_b \leq \eta, \tag{8.1.106}$$

where H_b is the average Shannon Entropy over the second half of active generations, and η has been set to 0.1. This test verifies that the average Shannon entropy of all active generations does not differ significantly from the Shannon entropy of the last half of the active generations. This test is useful for detecting fission source convergence in problems where an inadequate number of inactive generations was sampled, as the Shannon entropy would continue to change during the active generations until it eventually converges.

Source Convergence Diagnostic Input

Because the processing of the source tallies for fission source convergence calculations is quite fast and produces small memory requirements, these source convergence diagnostics have been permanently enabled in KENO and require no additional input parameters. However, the KENO input specifications were updated to allow users to specify the spatial mesh data for the source convergence tests via the `SCD` parameter and `READ GRID` data block.

8.1.8 APPENDICES

8.1.8.1 KENO V.a Shape Descriptions

The geometry **shape**s allowed in KENO V.a geometry description are:

CUBE, CUBOID, SPHERE, CYLINDER, ZCYLINDER, XCYLINDER, YCYLINDER, HEMISPHERE, HEMISPHE+X, HEMISPHE-X, HEMISPHE+Y,

HEMISPHE-Y, HEMISPHE+Z, HEMISPHE-Z, XHEMICYL+Y, XHEMICYL-Y, XHEMICYL+Z, XHEMICYL-Z, YHEMICYL+X, YHEMICYL-X, YHEMICYL+Z, YHEMICYL-Z, ZHEMICYL+X, ZHEMICYL-X, ZHEMICYL+Y, ZHEMICYL-Y

CUBE specifies a cube. It sets +X = +Y = +Z and -X = -Y = -Z. Note that the +X dimension need not equal the -X dimension of the cube (i.e., the origin need not be at the center of the cube).

CUBOID is a rectangular parallelepiped and may be described anywhere relative to the origin.

SPHERE specifies a sphere that is centered about the origin, unless otherwise specified by the optional region origin data.

CYLINDER specifies a cylinder that has its length described along the Z axis. Its centerline must lie on the Z axis, unless otherwise specified by the optional region origin data.

ZCYLINDER specifies a cylinder that has its length described along the Z axis. Its centerline must lie on the Z axis, unless otherwise specified by the optional region origin data.

XCYLINDER specifies a cylinder that has its length described along the X axis. Its centerline must lie on the X axis, unless otherwise specified by the optional region origin data.

YCYLINDER specifies a cylinder that has its length described along the Y axis. Its centerline must lie on the Y axis, unless otherwise specified by the optional region origin data.

HEMISPHERE is used to specify a spherical segment of one base whose spherical surface exists in the positive Z direction. The base or flat portion of the spherical segment is centered about a point that may be specified in the optional region origin data. By default, the center of the spherical surface is the origin and the distance to the base from the center of the spherical surface is zero.

HEMISPHE*bc* is used to specify a spherical segment of one base whose spherical surface exists in the *bc* direction (b = + or -, c = x, y, or z). The base or flat portion of the spherical segment is located a distance ρ from the center of the spherical surface, and the center may be specified in the optional region origin data. **HEMISPHE+Z** is the same as the previously described **HEMISPHERE** and **HEMISPHE-Z** is the mirror image of **HEMISPHE+Z**, therefore existing only in the negative Z direction. By default the center of the spherical surface is the origin and the distance of the base from the center of the spherical surface is zero.

b*HEMICYL*cd is used to specify a cylindrical segment whose axis is in the *b* direction (b = x, y, or z) and whose cylindrical surface exists only in the *c* direction from a plane perpendicular to the *d* axis (c = + or -, d = x, y, or z). The position of the plane (cut surface) can be specified in the optional region chord data. This plane cuts the cylinder parallel to the axis at some distance, ρ, from the axis. By default, the axis passes through the origin and ρ is zero. (Examples: **ZHEMICYL+X, YHEMICYL-Z, XHEMICYL+Y**)

Up to 6 dimensions follow the shape keyword [*xx(1)* through *xx(6)*]. These entries are separated by one or more blanks and define the size of the region. Dimensions must be given in cm.

xx(1): Radius for a sphere, cylinder, hemisphere or hemicylinder,

+X dimension for a cube or cuboid.

xx(2): -X dimension for cube or cuboid,

+Z for cylinder or Z cylinder,

+X for X cylinder,

+Y for Y cylinder,

+length for hemicylinder,

omit *xx(2)* for a sphere or hemisphere.

xx(3): +Y dimension for cuboid,

-Z for cylinder or Z cylinder,

-X for X cylinder,

-Y for Y cylinder,

-length for hemicylinder,

omit *xx(3)* for a sphere, hemisphere, or cube.

xx(4): -Y dimension for cuboid,

omit for all other geometry types.

xx(5): +Z dimension for cuboid,

omit for all other geometry types.

xx(6): -Z dimension for cuboid,

omit for all other geometry types.

8.1.8.2 KENO VI Shape Descriptions

The geometry **shape**s allowed in KENO-VI geometry description are:

CONE, **CUBOID**, **CYLINDER**, **DODECAHEDRON**, **ECYLINDER**, **ELLIPSOID**, **HEXPRISM**, **HOPPER**, **PARALLELEPIPED**, **PPIPED**, **PENTAGON**, **PLANE**, **QUADRATIC**, **RHEXPRISM**, **RHOMBOID**, **SPHERE**, **WEDGE**, **XCYLINDER**, **XPPLANE**, **YCYLINDER**, **YPPLANE**, **ZCYLINDER**, **ZPPLANE**

CONE specifies a body consisting of one nappe of a right circular cone. It is defined by specifying the top radius of the cone, R_t, the Z coordinate of the top face, Z_t, the bottom radius of the cone, R_b, and the Z coordinate of the bottom face, Z_b. Fig. 8.1.211 shows the correct input sequence for a cone.

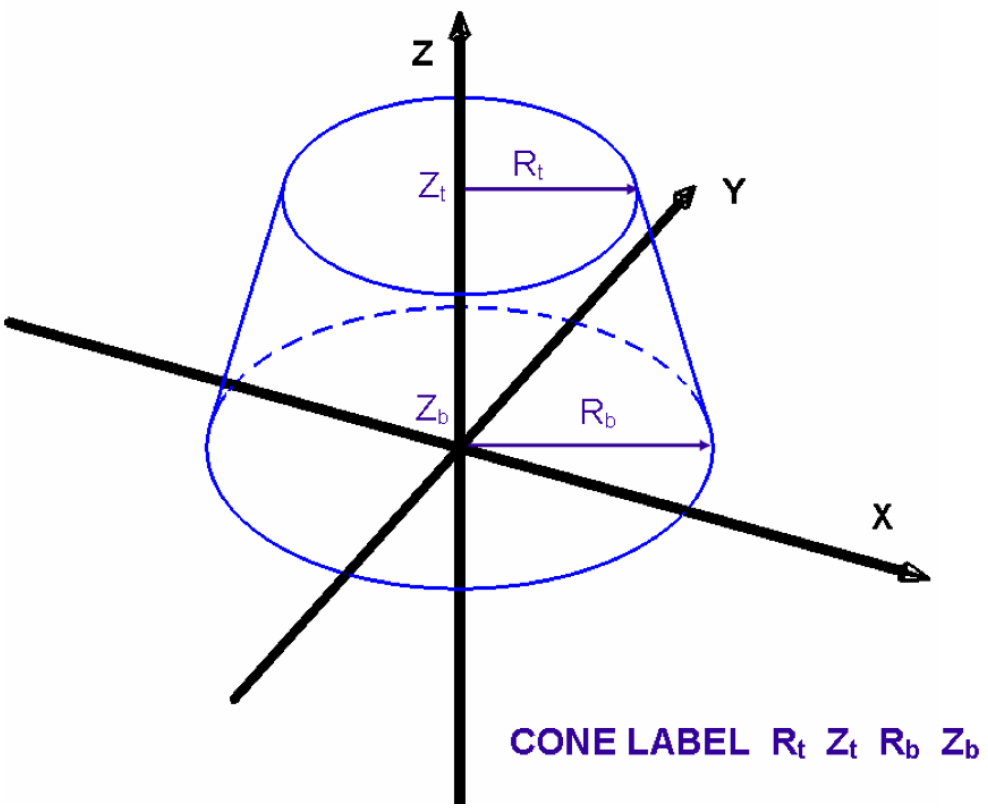


Fig. 8.1.211: Example of cone construction.

CUBOID specifies a rectangular parallelepiped. It is defined by specifying the +X dimension, -X dimension, +Y dimension, -Y dimension, +Z dimension, -Z dimension. It is perpendicular to the X, Y, and Z axes unless otherwise specified by the option geometry modification data. Fig. 8.1.212 shows the correct input sequence for a cuboid.

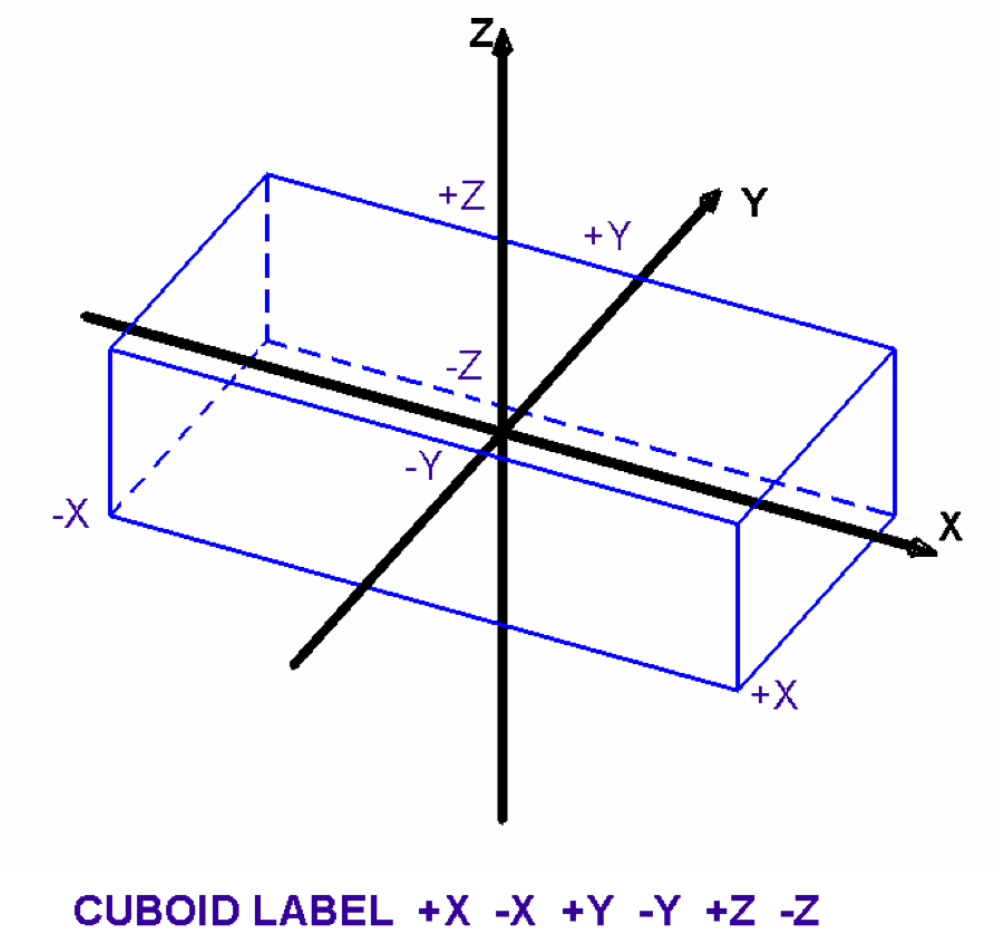


Fig. 8.1.212: Example of cuboid construction.

CYLINDER specifies a right circular cylinder. It is defined by specifying the radius of the cylinder, R, the Z coordinate of the top face, Z_t, and the Z coordinate of the bottom face, Z_b. Its centerline must lie on the Z axis, unless otherwise specified by the optional geometry modification data. Fig. 8.1.213 shows the correct input sequence for a cylinder.

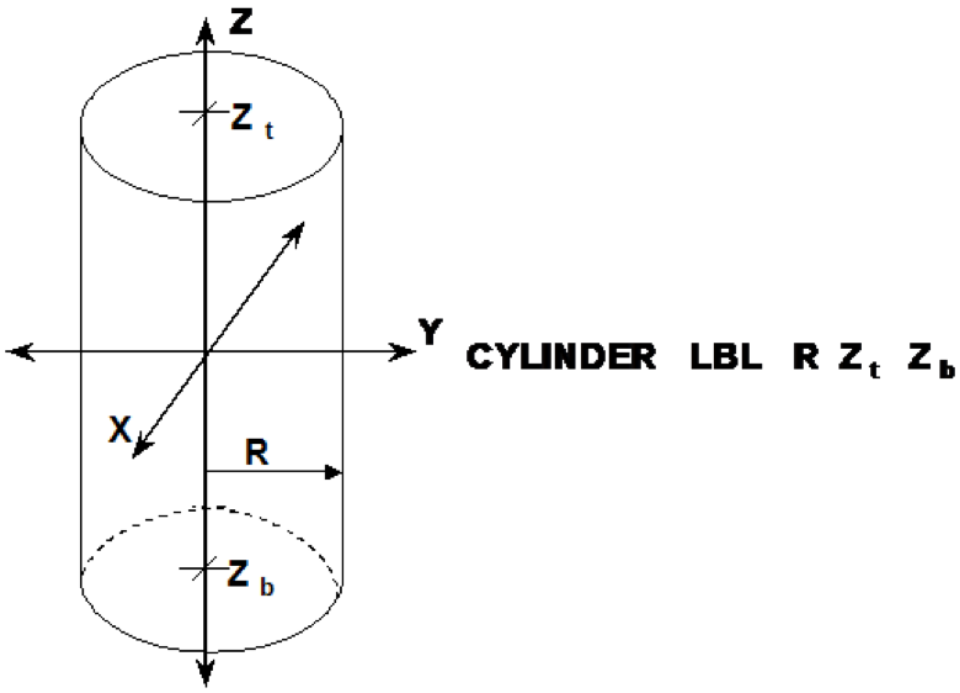


Fig. 8.1.213: Example of cylinder construction.

DODECAHEDRON specifies a body whose surface consists of 12 rhombuses of the same size and shape. It is defined by specifying the radius of the inscribed sphere, R. It is centered on the origin in a fixed orientation unless otherwise specified by the optional geometry modification data. Fig. 8.1.214 shows the correct input sequence for a dodecahedron.

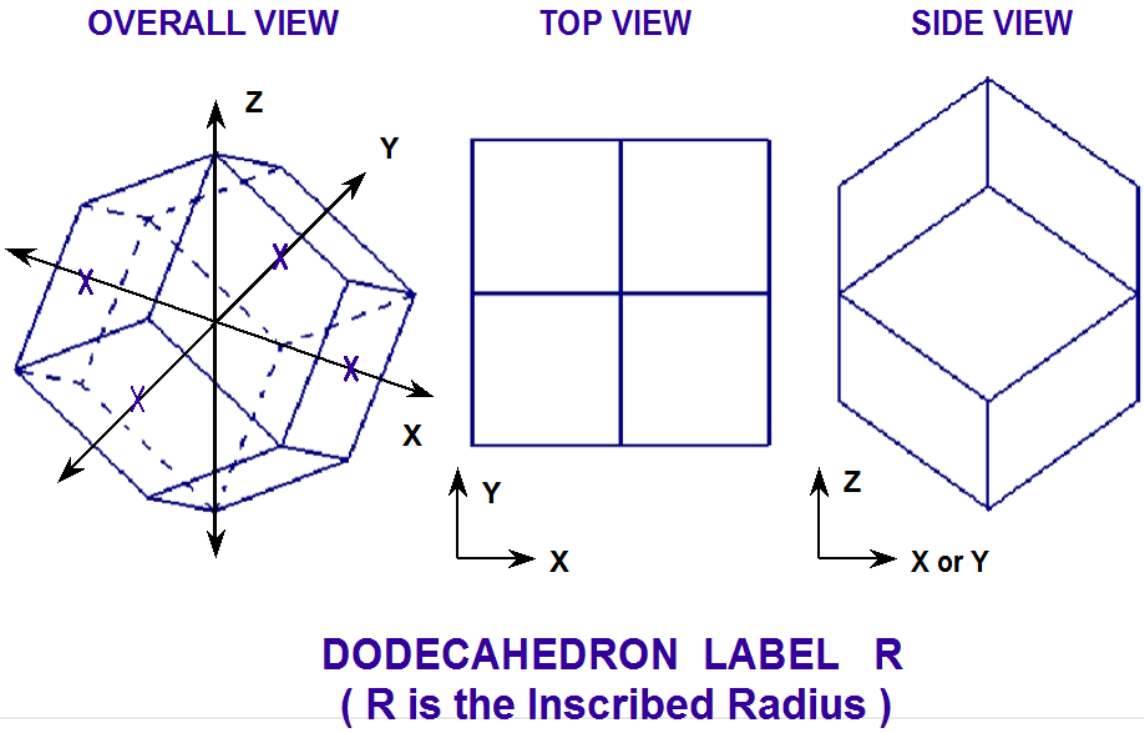


Fig. 8.1.214: Example of dodecahedron construction.

ECYLINDER specifies a right cylinder with an elliptical cross section. It is defined by specifying the semiradius along the X-axis, R_x, the semiradius along the Y-axis, R_y, the Z coordinate of the top face, Z_t, and the Z coordinate of the bottom face, Z_b. Its centerline must lie on the Z axis, unless otherwise specified by the optional geometry modification data. Fig. 8.1.215 shows the correct input sequence for an elliptical cylinder.

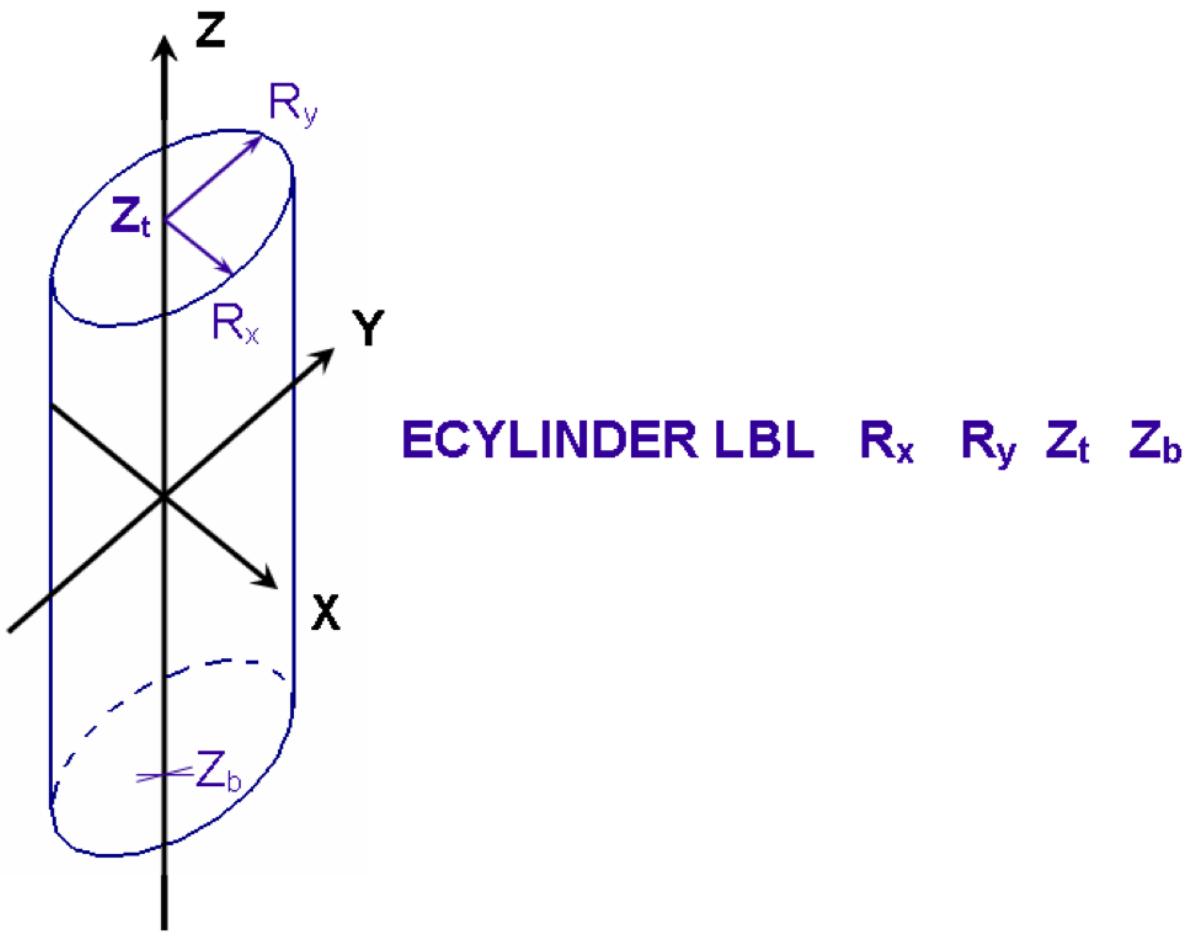


Fig. 8.1.215: Example of elliptical cylinder construction.

ELLIPSOID specifies a body whose cross-section slices parallel to each of the coordinate axes are ellipses. It is defined by specifying the semiradius along the X-axis, R_x, the semiradius along the Y-axis, R_y, and the semiradius along the Z-axis, R_z. It is centered about the origin, unless otherwise specified by the optional geometry modification data. Fig. 8.1.216 shows the correct input sequence for an ellipsoid.

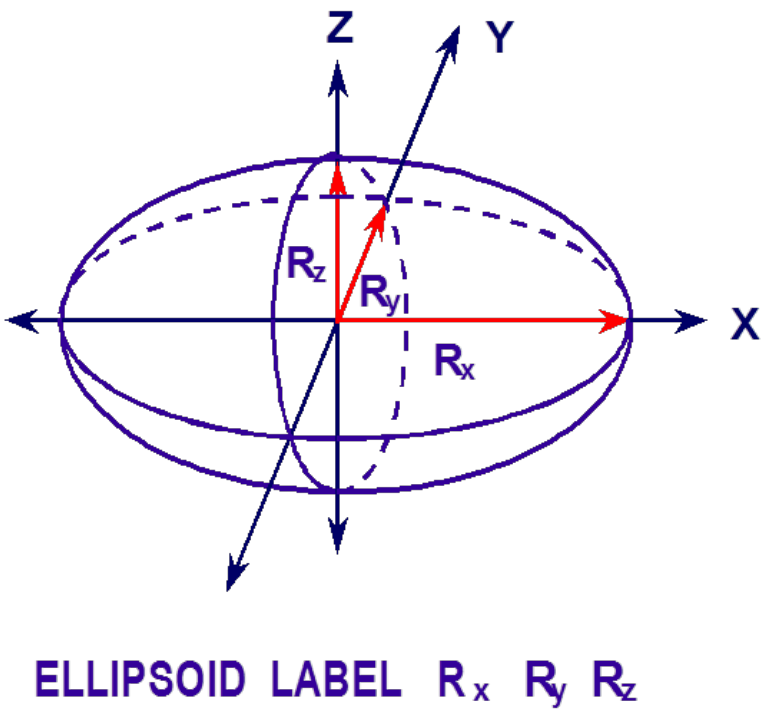


Fig. 8.1.216: Example of ellipsoid construction.

HEXPRISM specifies a body whose top and bottom faces are hexagons that have the same orientation and are perpendicular to the Z axis. It is defined by specifying the inscribed radius, R, the Z coordinate of the top face, Z_t, and the Z coordinate of the bottom face, Z_b. Fig. 8.1.217 is an example input for a hexprism.

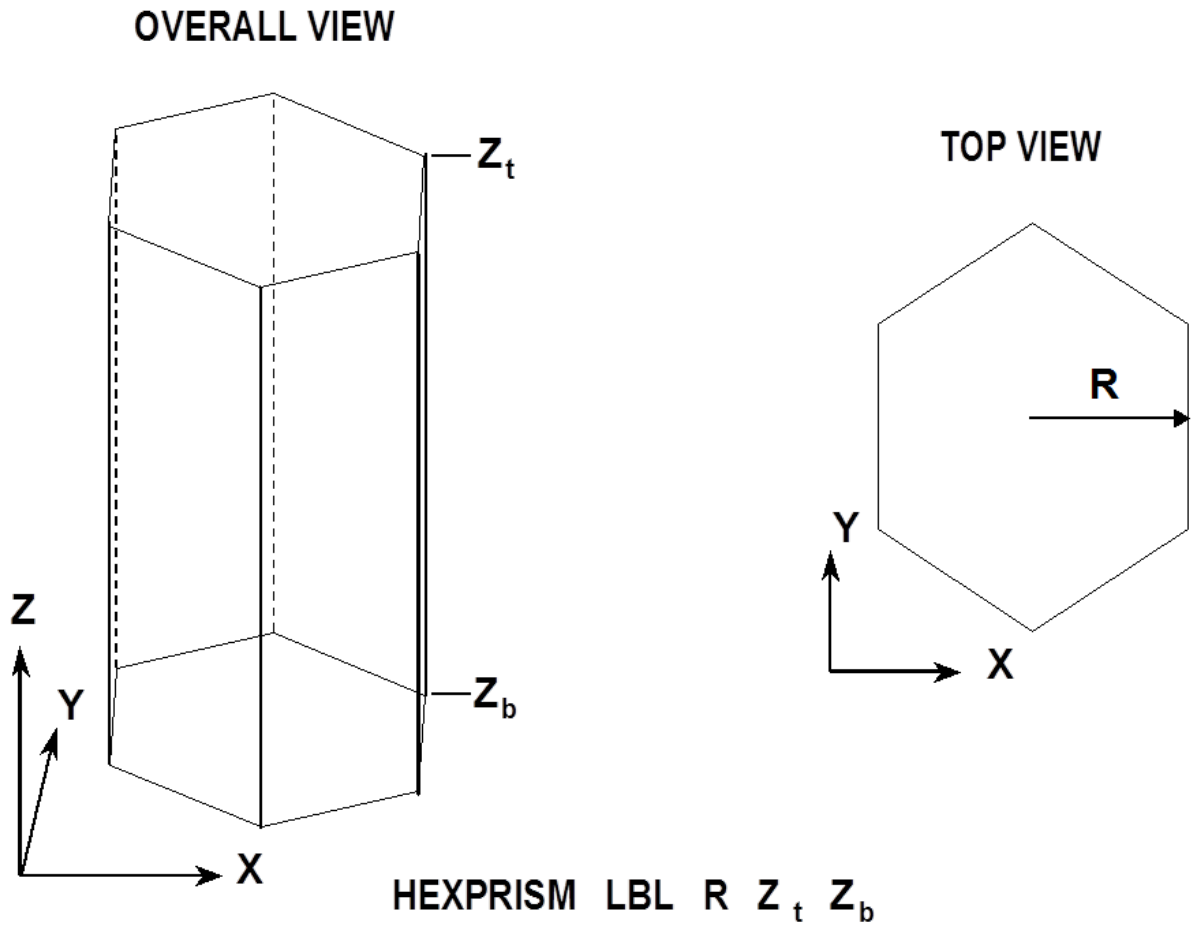


Fig. 8.1.217: Example of hexprism construction.

HOPPER specifies a body whose top and bottom faces are rectangular parallelepipeds centered about the Z-axis and parallel to the X and Y axes. It is defined by specifying the half-length of the top face along the X-axis, X_t, the half-length of the top face along the Y-axis, Y_t, the Z coordinate of the top face, Z_t, the half-length of the bottom face along the X-axis, X_b, the half-length of the bottom face along the Y-axis, Y_b, and the Z coordinate of the bottom face, Z_b. Its centerline must lie on the Z axis unless otherwise specified by the optional geometry modification data. Fig. 8.1.218 shows the correct input sequence for a hopper.

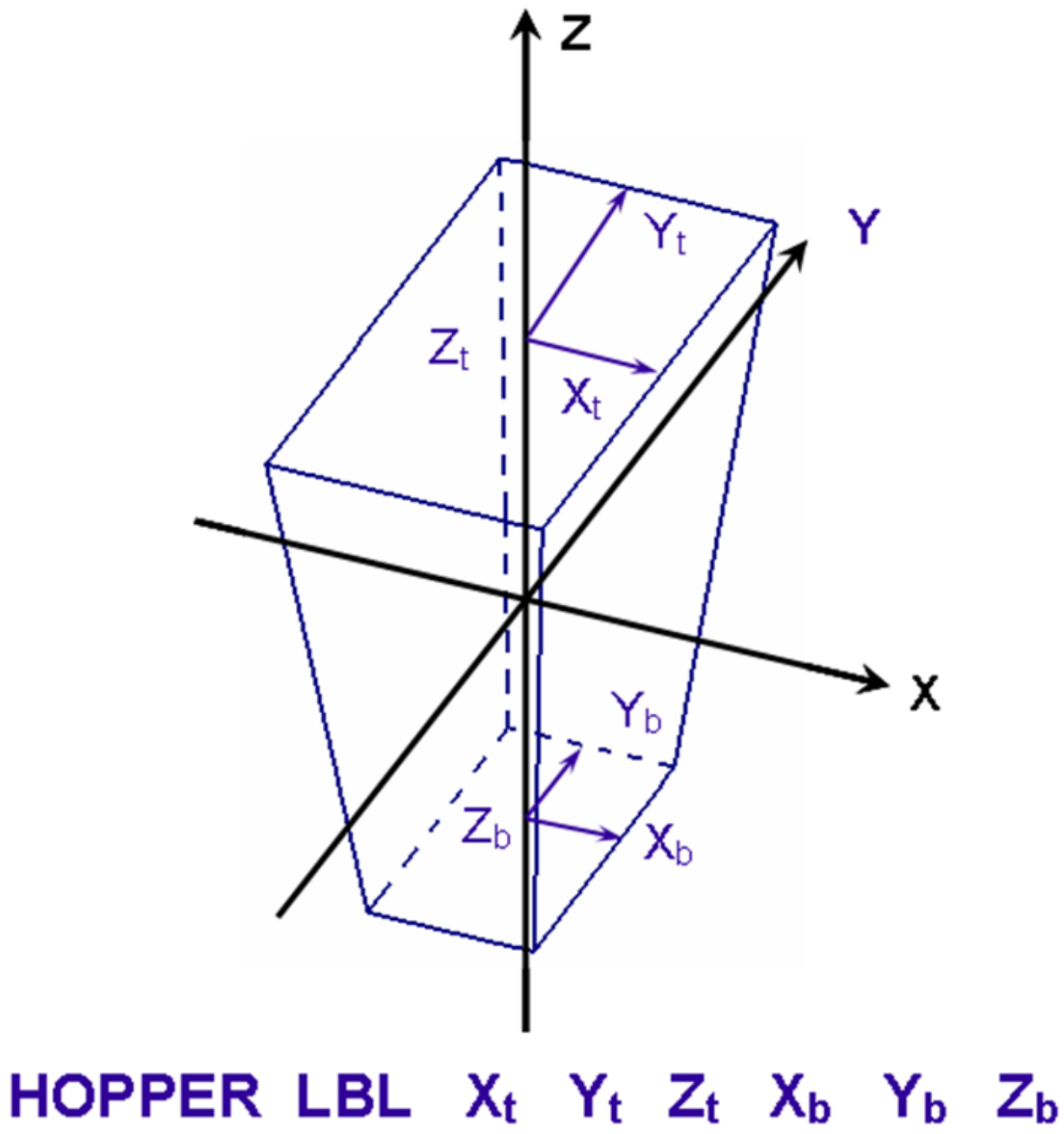


Fig. 8.1.218: Example of hopper construction.

PARALLELEPIPED or PPIPED is a body with six faces composed of parallelograms, whose opposing faces are parallel. It is defined by specifying the length of the faces in the X direction, XDIST, the length of the faces in the Y direction, YDIST, the length of the faces in the Z direction, ZDIST, the angle between the X-face and the Y-axis, PSI, the angle between the Y-face and the Z-axis, THETA, and the angle between the projection of the top corner nearest the Z-axis onto the X-Y plane and the X-axis, PHI. The bottom face must lie on the X-Y plane at Z = 0 with a corner at the origin unless otherwise specified by the optional geometry modification data. Fig. 8.1.219 shows the correct input sequence for a parallelepiped. The angles psi, theta, and phi must be in the range 0 to 90°.

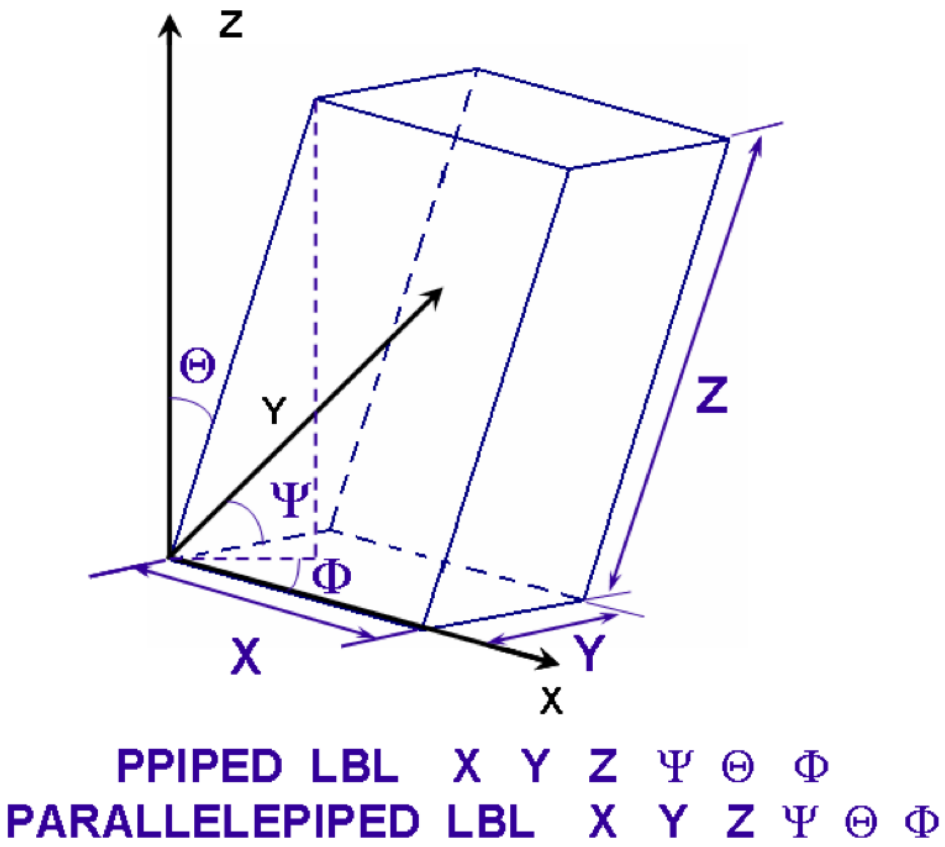


Fig. 8.1.219: Example of parallelepiped construction.

PENTAGON specifies a body whose top and bottom faces are pentagons that have the same orientation and are perpendicular to the Z axis. It is defined by specifying the inscribed radius, R, the Z coordinate of the top face, Z_t, and the Z coordinate of the bottom face, Z_b. Fig. 8.1.220 is an example input for a pentagon.

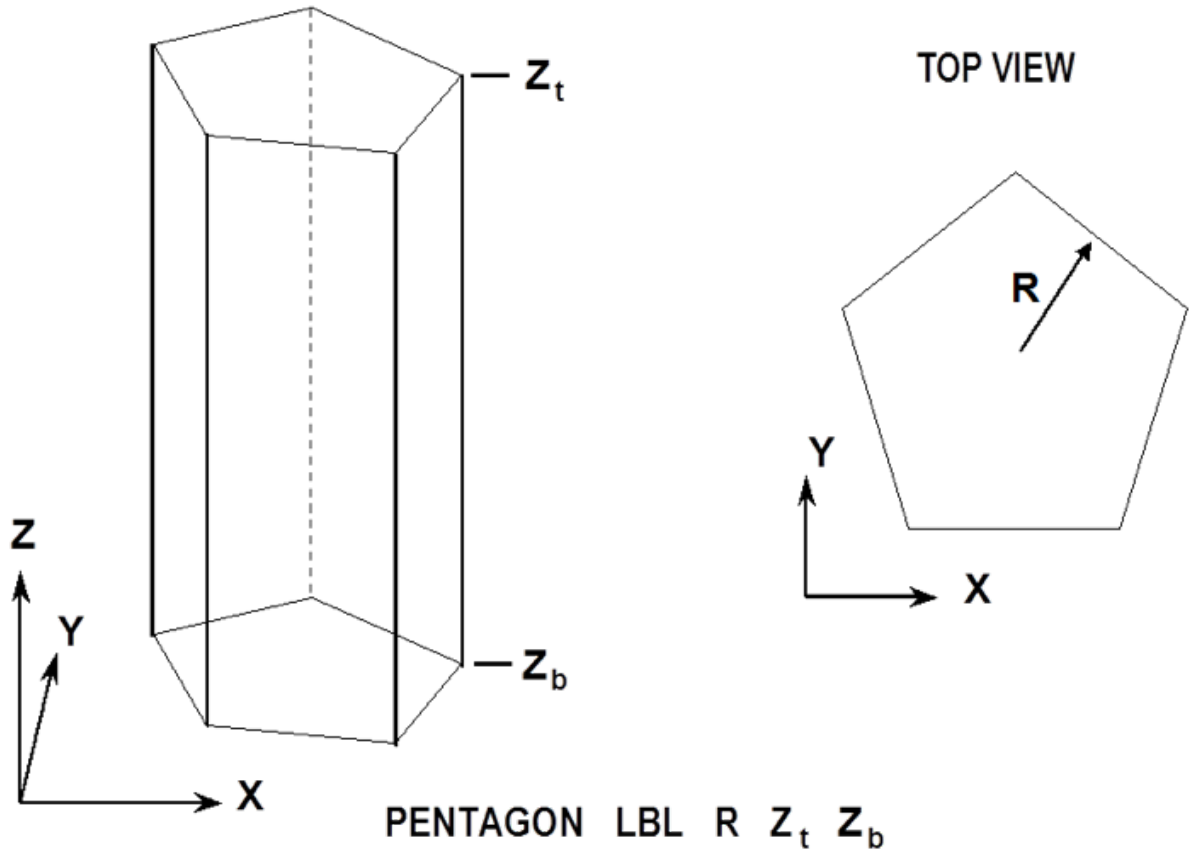
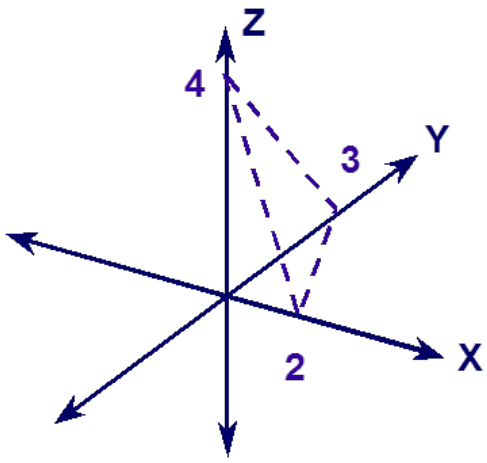


Fig. 8.1.220: Example of pentagon construction.

PLANE is a surface where any two points can be connected by a straight line entirely contained within a plane that divides all space into two regions. The positive side of the plane is the side the normal points to or where the equation $aX + bY + cZ + d > 0$. It is defined by specifying the coefficients of the equation $aX + bY + cZ + d = 0$ using the keywords XPL=a, YPL=b, ZPL=c, and CON=d. Only the nonzero coefficients of the equation need to be specified. Fig. 8.1.221 shows the correct input sequence for a plane.



PLANE Label XPL=a YPL=b ZPL=c CON=d

aX + bY + cZ + d = 0

6X + 4Y + 3Z - 12 = 0

Fig. 8.1.221: Example of plane construction.

QUADRATIC specifies a surface using a quadratic equation of the form:

$$aX^2 + bY^2 + cZ^2 + dXY + eXZ + fYZ + gX + hY + iZ + j = 0.$$

It is defined by specifying the coefficients of the above equation using the keywords AQU=a, BQU=b, CQU=c, DQU=d, EQU=e, FQU=f, GQU=g, HQU=h, IQU=i, and JQU=j. Only the nonzero coefficients of the equation need to be specified.

RHEXPRISM specifies a body whose top and bottom faces are rotated hexagons that have the same orientation and are perpendicular to the Z axis. It is defined by specifying the inscribed radius, R, the Z coordinate of the top face, Z_t, and the Z coordinate of the bottom face, Z_b. Fig. 8.1.222 is an example input for a rotated hexprism.

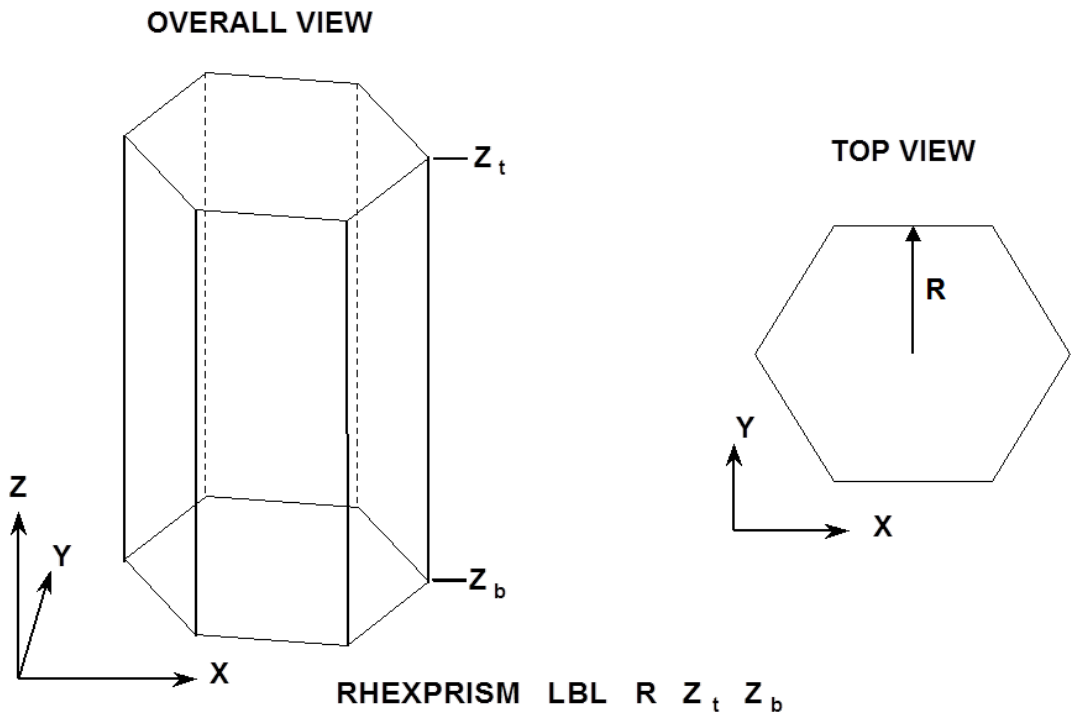


Fig. 8.1.222: Example of rotated hexprism construction.

RING is a body composed of the space between 2 concentric cylinders. It is defined by specifying the radius Rin of the inner cylinder and Rout of the outer cylinder, and the coordinate Zt of the top and Zb of the bottom of the annulus. Its center line lies on the Z axis unless specified by the optional geometry modification data. Fig. 8.1.223 shows the correct input sequence for a ring.

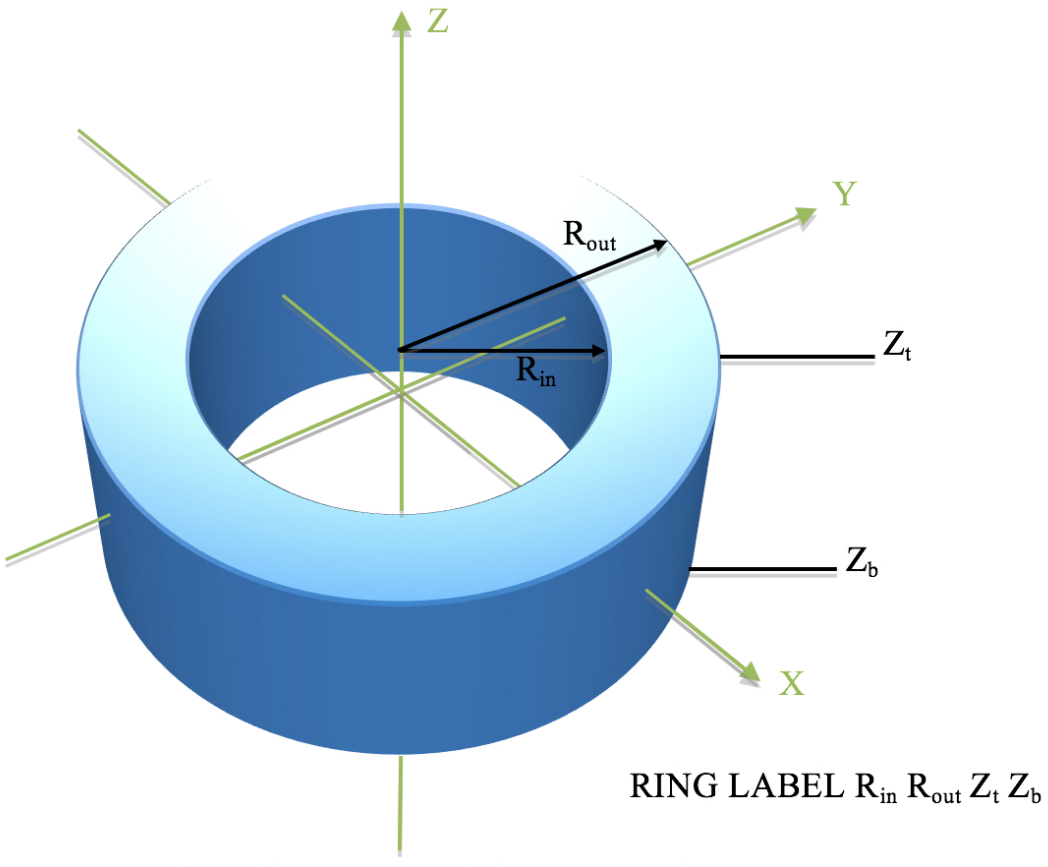


Fig. 8.1.223: Example of ring construction.

RHOMBOID is a body composed of six identical faces, each one a rhombus. It is defined by specifying the length of the edge of the base along the X-axis, DX and the angle between Y edge of the base and the Y-axis, Ψ. Its base is in the XY plane at Z = 0, with a corner at the origin unless otherwise specified by the optional geometry modification data. Fig. 8.1.224 shows the correct input sequence for a rhomboid.

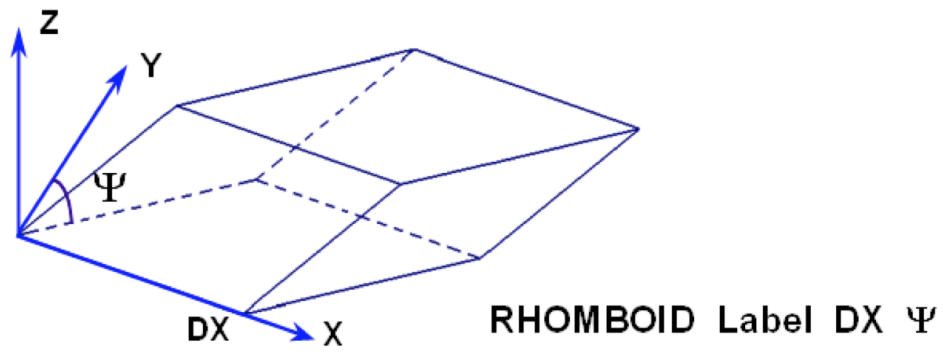


Fig. 8.1.224: Example of rhomboid construction.

SPHERE specifies a sphere. It is defined by specifying the radius, R. It is centered about the origin, unless otherwise specified by the optional geometry modification data. Fig. 8.1.225 shows the correct input

sequence for a sphere.

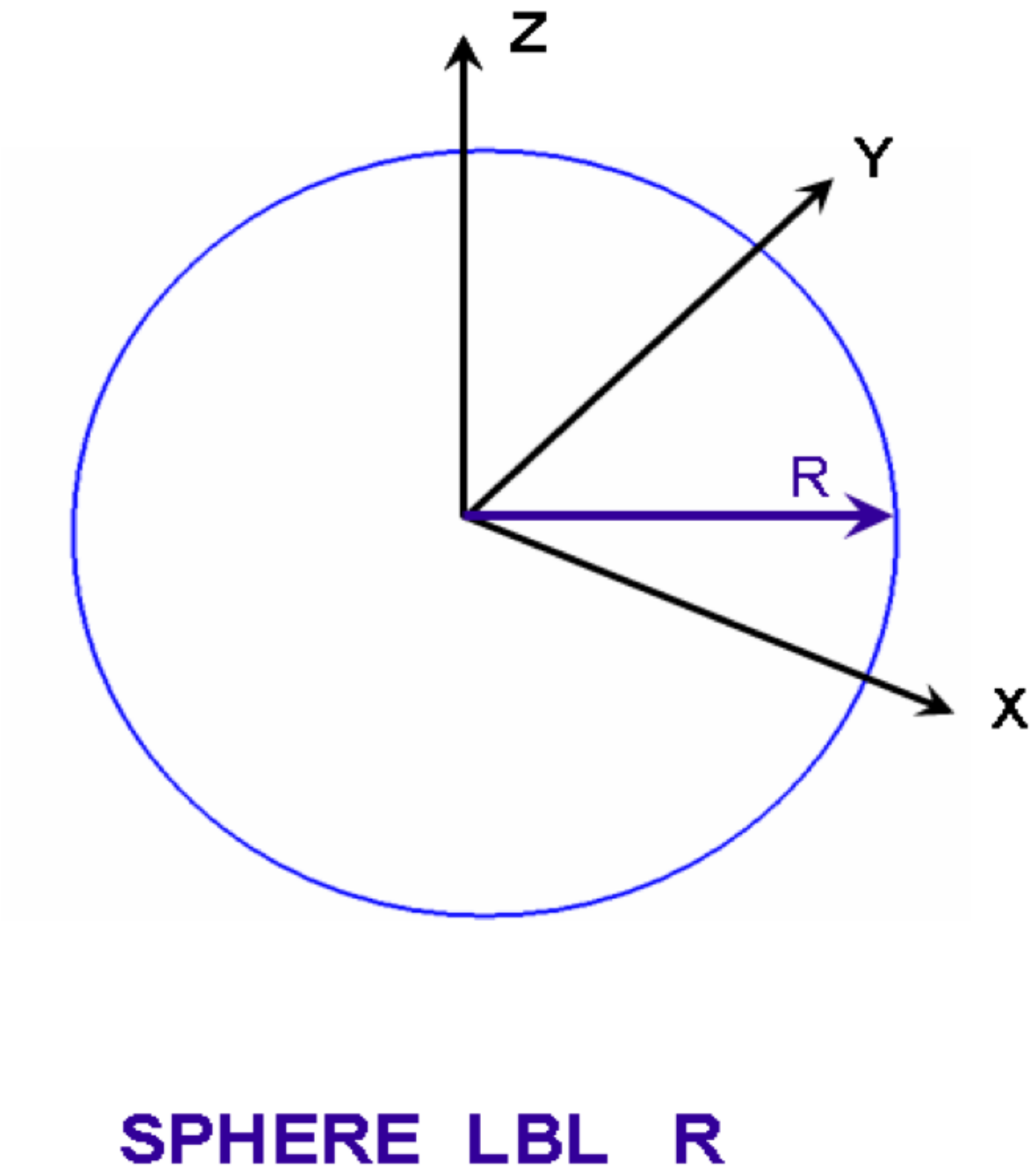


Fig. 8.1.225: Example of sphere construction.

WEDGE is a right-triangular prism having five faces. The two ends are triangles, and the three sides are rectangles. It is defined by specifying the length of the base along the X-axis, XBASE, the X and Y coordinate where the other two sides meet, Xpt and Ypt, and the length along the Z-axis, ZLNG. One side is in the XZ plane at Y = 0, and the bottom face is in the XY plane at Z = 0, with a corner at the origin unless otherwise specified by the optional geometry modification data. Fig. 8.1.226 shows

the correct input sequence for a wedge.

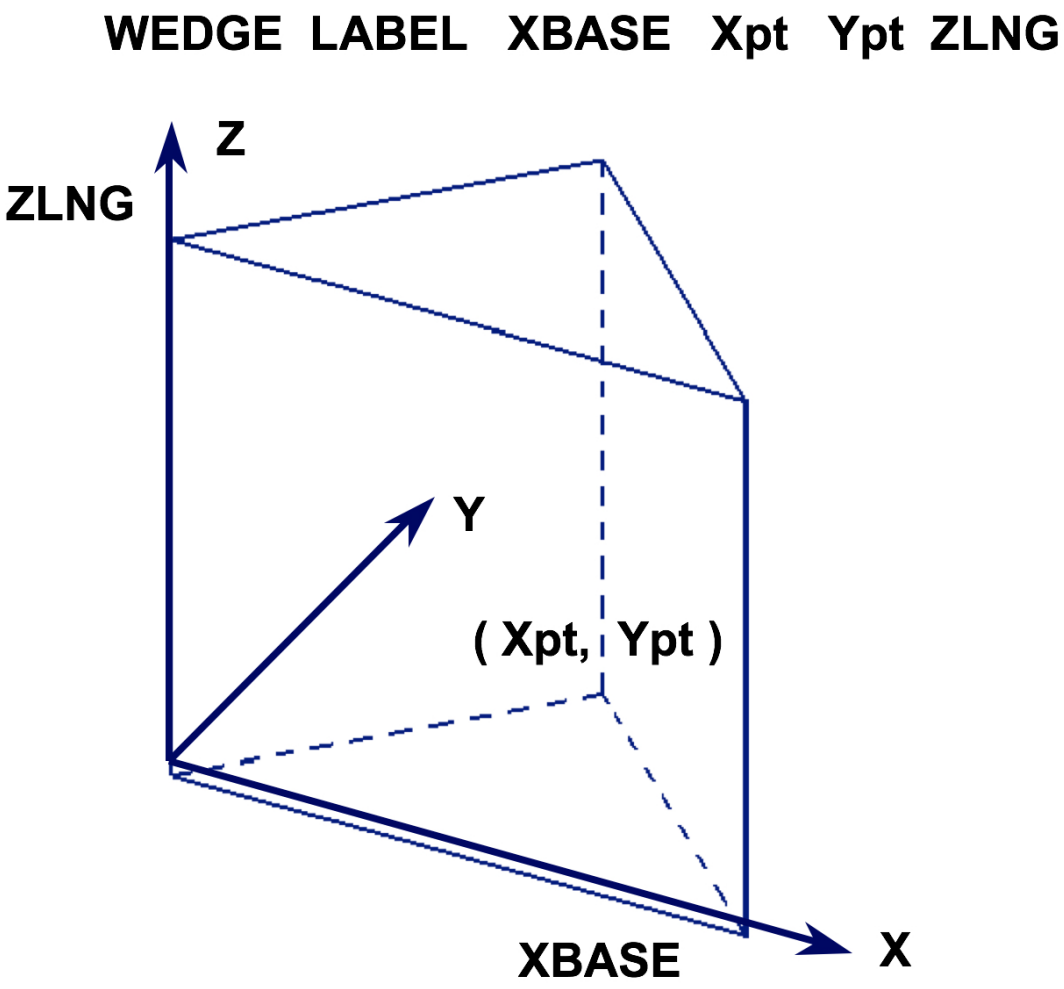


Fig. 8.1.226: Example of wedge construction.

XCYLINDER specifies a right circular cylinder oriented about the X-axis. It is defined by specifying the radius of the cylinder, R, the X coordinate of the top face, X_t, and the X coordinate of the bottom face, X_b. Its centerline must lie on the X axis, unless otherwise specified by the optional geometry modification data. Fig. 8.1.227 shows the correct input sequence for a cylinder.

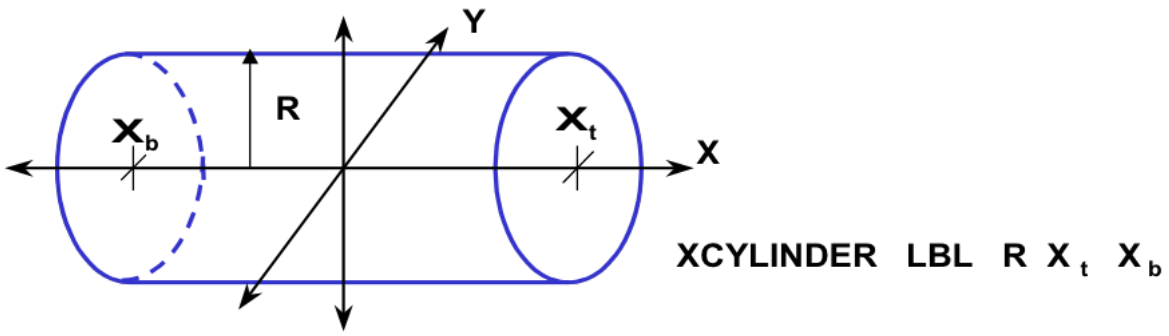


Fig. 8.1.227: Example of xcylinder construction.

XPPLANE is a set of flat parallel surfaces where any two points in one of the surfaces can be connected by a straight line entirely contained within that surface. These surface planes divide space into three sections; one section between the two planes which is considered inside the surfaces, one section on the negative side of the negative plane which is considered outside the surfaces, and one section on the positive side of the positive plane which is considered outside the surfaces. The set of parallel planes are defined by the keyword XPPLANE, which places the planes perpendicular to the X-axis, the X-intercept between the more positive plane and the X-axis (X_+) and the X-intercept between the more negative plane and the X-axis (X_-). Fig. 8.1.228 shows the correct input sequence for the set of paired planes.

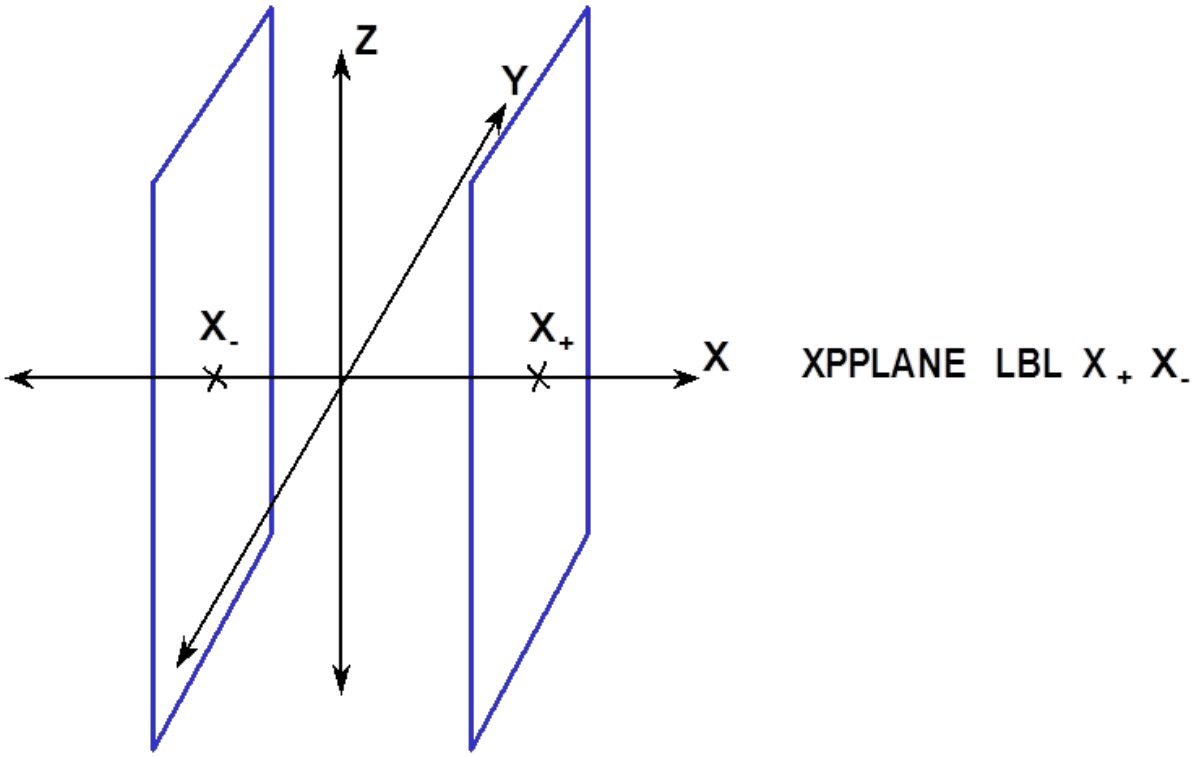
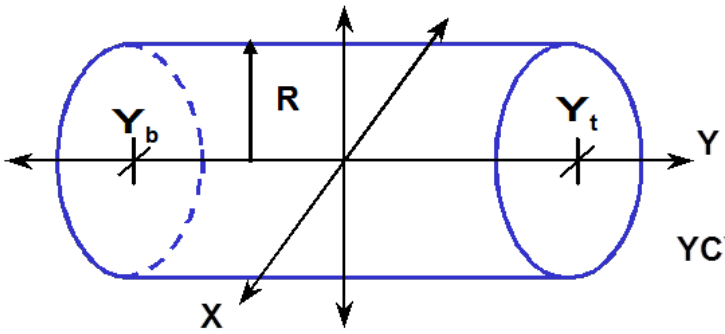


Fig. 8.1.228: Example of x-paired plane construction.

YCYLINDER specifies a right circular cylinder oriented about the Y-axis. It is defined by specifying the radius of the cylinder, R, the Y coordinate of the top face, Y_t, and the Y coordinate of the bottom

face, Y_b. Its centerline must lie on the Y axis, unless otherwise specified by the optional geometry modification data. Fig. 8.1.229 shows the correct input sequence for a cylinder.



YCYLINDER LBL R Y $_t$ Y $_b$

Fig. 8.1.229: Example of ycylinder construction.

YPPLANE is a set of flat parallel surfaces where any two points in one of the surfaces can be connected by a straight line entirely contained within that surface. These surface planes divide space into three sections; one section between the two planes which is considered inside the surfaces, one section on the negative side of the negative plane which is considered outside the surfaces, and one section on the positive side of the positive plane which is considered outside the surfaces. The set of parallel planes are defined by the keyword YPPLANE, which places the planes perpendicular to the Y-axis, the Y-intercept between the more positive plane and the Y-axis (Y_+) and the Y-intercept between the more negative plane and the Y-axis (Y_-). Fig. 8.1.230 shows the correct input sequence for the set of paired planes.

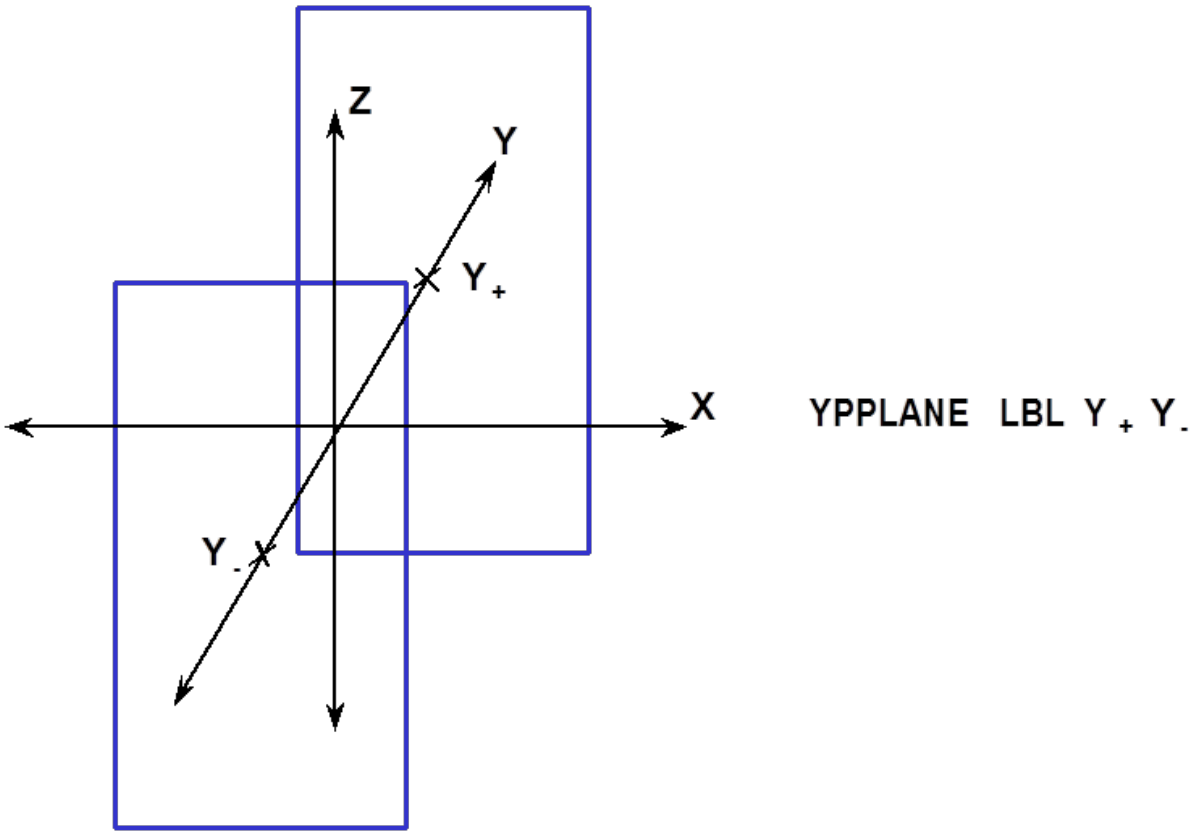


Fig. 8.1.230: Example of y-paired plane construction.

ZCYLINDER specifies a right circular cylinder oriented about the Z-axis. It is defined by specifying the radius of the cylinder, R, the Z coordinate of the top face, Z_t, and the Z coordinate of the bottom face, Z_b. Its centerline must lie on the Z-axis, unless otherwise specified by the optional geometry modification data. The keyword ZCYLINDER is the same as CYLINDER. It is included to be consistent with the XCYLINDER and YCYLINDER keywords. Fig. 8.1.231 shows the correct input sequence for a zcylinder.

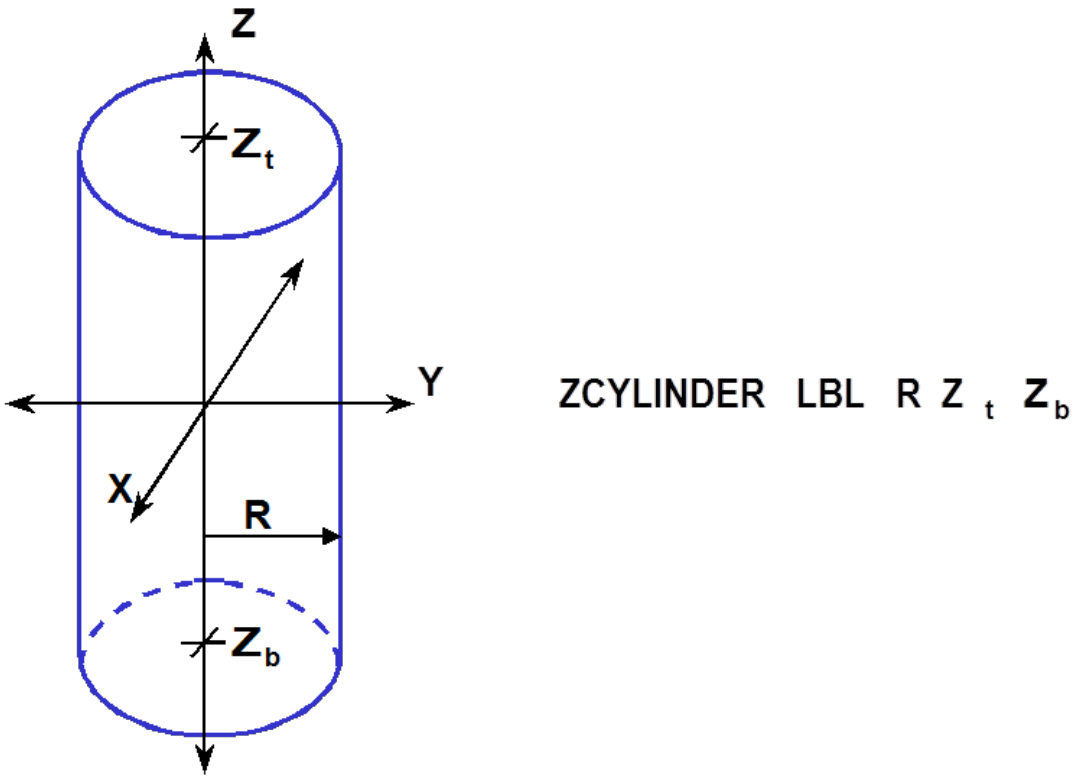


Fig. 8.1.231: Example of zcylinder construction.

ZPPLANE is a set of flat parallel surfaces where any two points in one of the surfaces can be connected by a straight line entirely contained within that surface. These surface planes divide space into three sections; one section between the two planes which is considered inside the surfaces, one section on the negative side of the negative plane which is considered outside the surfaces, and one section on the positive side of the positive plane which is considered outside the surfaces. The set of parallel planes are defined by the keyword ZPPLANE, which places the planes perpendicular to the Z-axis, the Z-intercept between the more positive plane and the Z-axis (Z_+) and the Z-intercept between the more negative plane and the Z-axis (Z_-). Fig. 8.1.232 shows the correct input sequence for the set of paired planes.

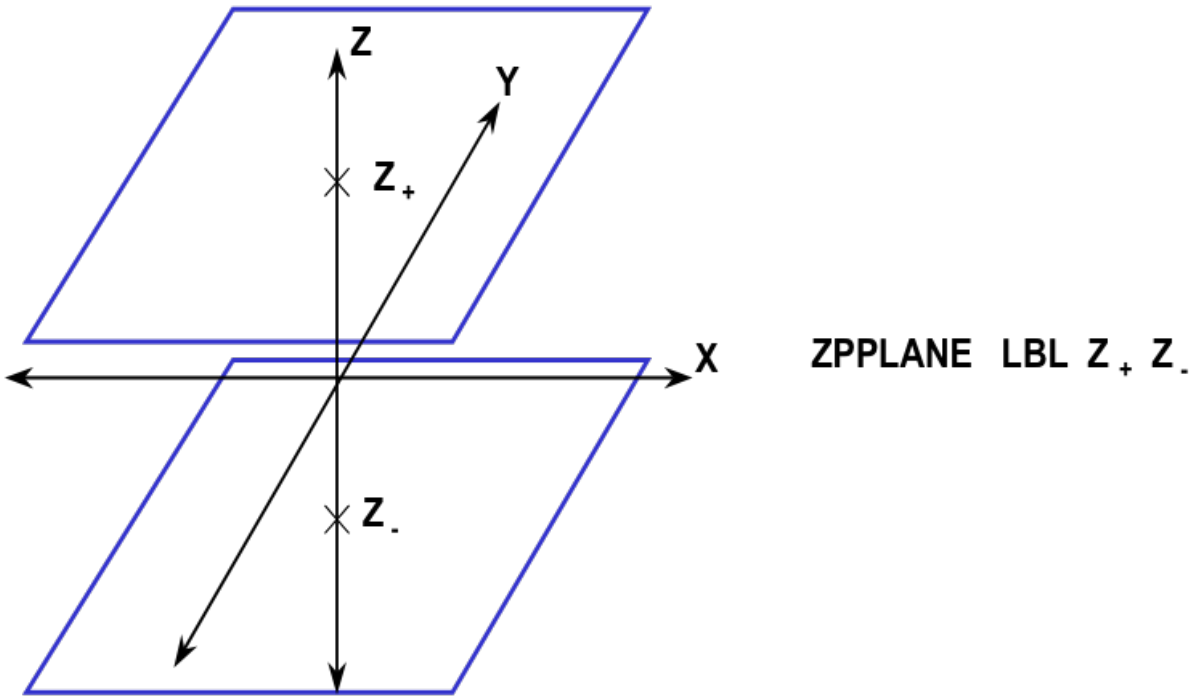


Fig. 8.1.232: Example of z-paired plane construction.

8.1.8.3 KENO Sample Problems

This section contains sample problems to demonstrate some of the options available in KENO in stand-alone mode. Because stand-alone KENO has no means to read standard composition information and process for use, the problem-dependent cross section library must be prepared before executing KENO in the multigroup mode. The **MIXTURE** data block (See Sect. 8.1.3.1) is used to provide the mixing table. In the continuous energy mode, the cross sections are directly used and therefore no problem-dependent library is needed. The mixing table is required in the continuous energy mode as well. If KENO is executed as part of CSAS5 or CSAS6 sequence, generation of the problem-dependent library (for the multigroup mode) and the mixing table is automatically performed by the sequence.

A total of 33 KENO V.a different case inputs and 27 KENO-VI inputs are provided as multigroup mode KENO sample problems in a single input file "kenova.inp" and "kenovi.inp" for KENO V.a and KENO-VI, respectively. This input file contains an initial CSAS-MG input to create the problem-dependent cross section library to be used in the sample problems in the input file. Although KENO does not run stacked cases, when KENO is run as part of SCALE, the driver allows KENO to be executed each time it encounters an **=KENOVA**, respectively **=KENOVI**. The ".inp" file contains all 33/27 problems one after the other. A similar input file "cekenova.inp", respectively "cekenovi.inp" is also provided for continuous energy mode of calculations. The changes required to create the continuous energy mode input file from the multigroup mode input file are simple. The continuous energy mode file does not have (or need) the CSAS-MG input at the beginning. In addition, all "lib=4" entries in the PARAMETER data block are changed to **cep=ce_v7.1_endf** to indicate the mode of calculation is continuous energy and the continuous energy cross section directory file is "ce_v7.1_endf" indicating ENDF/B-VII.1-based cross sections. The mixing table entry SCT is not applicable in the continuous energy mode, so it has been deleted from the continuous energy input file. Finally, material-specific albedos have also been removed from the continuous-energy input because they are not supported with continuous-energy mode.

The same 33/27 problems are also executed as individual cases with filenames “k5smp??.inp”, respectively “k6smp??.inp”, where ?? stands for sample problem number (01 through 33 or 27). Since each one of these sample problems needs a problem-dependent cross section library (multigroup mode only) and a mixing table, these problems have been converted to run as CSAS5/6 problems. Similar input files are also provided to be run in the continuous energy mode and the files are named “cek5smp??.inp”, respectively “cek6smp??.inp”, where ?? again stands for sample problem number (01 through 33 or 27). The change required to create the continuous energy mode inputs from the multigroup mode inputs is very simple: the cross-section library name is changed from “v7.1-252” to “ce_v7.1”.

In the following section the input for each case is listed assuming the multigroup mode of calculation in KENO. The KENO input is also listed in the file corresponding “.inp” file. The CSAS-MG input file for these cases is in the next section.

Warning: Some input might show differences in **NPG**, **NSK** and **GEN** parameters between multigroup and continuous energy input files.

CSAS-MG data

CSAS-MG can (1) be run alone with problem-dependent working library on logical Unit 4 saved for later use with the KENO sample problems, or (2) be placed in front of the KENO sample problems.

The CSAS-MG SCALE control module calculates the necessary resonance data required to create the problem-dependent AMPX working format library using SCALE standard composition input.

The multigroup mode KENO sample problem input data are independent of energy group structure. To use a different energy group structure, simply supply the desired master cross-section library name in the CSAS-MG or CSAS5/6 data. See XSProc, Standard Composition and CSAS5/6 chapters for additional information and examples. See the XSLib chapter for information about the master format cross-section libraries that are available in SCALE.

Data for CSAS-MG are provided to create a problem-dependent AMPX working format cross-section library suitable for use with the multigroup mode KENO sample problems. These data include all of the mixtures used in the KENO sample problems and will create an AMPX working format cross-section library with nuclide IDs matching those in the KENO sample problem mixing tables. This cross-section library is problem-specific and is not appropriate for use with other problems.

The CSAS-MG input data to produce an AMPX working format cross-section library for the multigroup mode KENO V.a sample problems are given below. Note that “kenovi.inp” also has its own CSAS-MG data which is identical to the below input with some minor updates in the comment lines.

```
=csas-mg
xsproc to prepare 252 group working format xsec lib for kenova smp prbs
v7.1-252
read composition
' uranium metal for smp prbs 1,2,3,4,5,6,7,8,9,10,11,12,19,22,23,24,25,26,27,28
 uranium     1 den=18.76 1 300
                                92234 1
                                92235 93.2
                                92236 0.2
                                92238 5.6   end
' uranyl nitrate solution for smp prbs 12,18,19   spg=1.555
 solution   mix=2  rho[uo2(no3)2]=415 92235 92.6
                                     92238 5.9
```

(continues on next page)

(continued from previous page)

```
                                    92234 1
                                    92236 0.5
                molar[hno3]=0.009783
                density=? temperature=300 vol_frac=1
  end solution
' uranium metal for smp prbs 13,14
 uranium     3 den=18.69 1 300
                                  92234 1
                                  92235 93.2
                                  92236 0.2
                                  92238 5.6   end
' uranium metal for smp prb 15
 uranium     4 den=18.794 1 300
                                  92234 1.09
                                  92235 97.67
                                  92236 0.21
                                  92238 1.03   end
' uranyl fluoride solution for smp prb 16
 solution    mix=5  rho[uo2f2]=578.7 92235 93.2
                                     92238 6.8
                density=? temperature=300 vol_frac=1
  end solution
' borated uranyl fluoride solution for smp prb 16
 solution    mix=6  rho[uo2f2]=578.7 92235 93.2
                                     92238 6.8
                density=? temperature=300 vol_frac=1
  end solution
 boron       6 den=0.0266 1 300   end
' uranyl fluoride solution for smp prb 17
 solution    mix=7  rho[uo2f2]=133 92235 93
                                   92238 7
                density=? temperature=300 vol_frac=1
  end solution
' uranyl fluoride solution for smp prb 20
 solution    mix=8  rho[uo2f2]=576.87 92235 93.2
                                      92238 6.8
                density=? temperature=300 vol_frac=1
  end solution
' uranyl fluoride solution for smp prb 21     spg= 1.56
 solution    mix=9  rho[uo2f2]=494 92235 4.89
                                   92238 95.09
                                   92234 0.02
                density=? temperature=300 vol_frac=1
  end solution
' paraffin for smp prbs 3,4,18
 paraffin    10 1 300   end
' plexiglas for smp prbs 12,15,18,19
 plexiglas   11 1 300   end
' water for smp prbs 15
 h2o         12 1 300   end
' pyrex glass for smp prb 16
 pyrex       13 1 300   end
' aluminum for smp prb 20,21
 al          14 1 300   end
' low density water for smp prb 18
 h2o         15 1e-09 300   end
' uranium metal for smp prbs 29 - 32
 uranium     16 den=18.747 1 300
                                  92234 0.9844
                                  92235 93.21
                                  92236 0.0359
                                  92238 5.7697   end
' uranium metal for water moderated portion of smp prb 33
 uranium     17 den=19 1 300
                                  92234 0.002
```

(continues on next page)

(continued from previous page)

```
                                92235 1.95
                                92236 0.006
                                92238 98.042   end
' internal (2nd) moderator water for smp prb 33
 h2o          18 1 300   end
' external moderator water and reflector for smp prb 33
 h2o          19 1 300   end
' uranium metal for bare portion of smp prb 33
 uranium      20 den=19 1 300
                                92234 0.002
                                92235 1.95
                                92236 0.006
                                92238 98.042   end
end composition
read celldata
'latticecell data for samp prb 33
  latticecell atriangpitch imodr=3.302 18 fuelr=9.144 17 hpitch=10.414 19 end
  latticecell atriangpitch imodr=3.302 0 fuelr=9.144 20 hpitch=10.414 0 end
end celldata
end
```

KENO V.a sample problem data

A brief problem description and the associated input data are included for each multigroup mode KENO sample problems. Different options may be easily activated by making changes in the data. These problems are set up using an AMPX working format library which was created by a CSAS-MG case just prior to the KENO V.a/KENO-VI cases. The nuclide identifiers for this library are consistent with the SCALE identifiers created by CSAS-MG. Input data to create this library are given in Sect. 8.1.8.3.1. The unit number is defined by the parameter **LIB=** in the parameter data.

Sample Problem 1 2C8 BARE

This is a simple 2 × 2 × 2 array of uranium metal cylinders as described in the article "Critical Three-Dimensional Arrays of U(93.2)-Metal Cylinders," [KENO-Appendix-CTho73] by J. T. Thomas. This critical experiment is designated in Table II of that article as cylinder index 11 and reflector index 1. Fig. 8.1.233 shows the critical experiment.

Input Data

KENO V.a

```
=kenova
sample problem 1  case 2c8 bare
read parameters
  flx=yes fdn=yes far=yes gas=no lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    cylinder 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
end geometry
read array
  nux=2 nuy=2 nuz=2 fill f1 end fill
end array
end data
end
```

```
=kenovi
kenovi  sample problem 1  case 2c8 bare
read parameters
  flx=yes fdn=yes far=yes gas=no lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='single 2c8 unit centered'
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  4p6.87 2p6.505
    media  1 1 10       vol=8938.968624
    media  0 1 20 -10   vol=10710.044784
    boundary 20
  global unit 2
    cuboid 10  4p13.74 2p13.01
    com='2x2x2  2c8 array'
    array 1 +10 place 1 1 1 2r-6.87 -6.505
    boundary  10
end geometry
read array
  ara=1 nux=2 nuy=2 nuz=2  fill f1  end fill
end array
end data
end
```

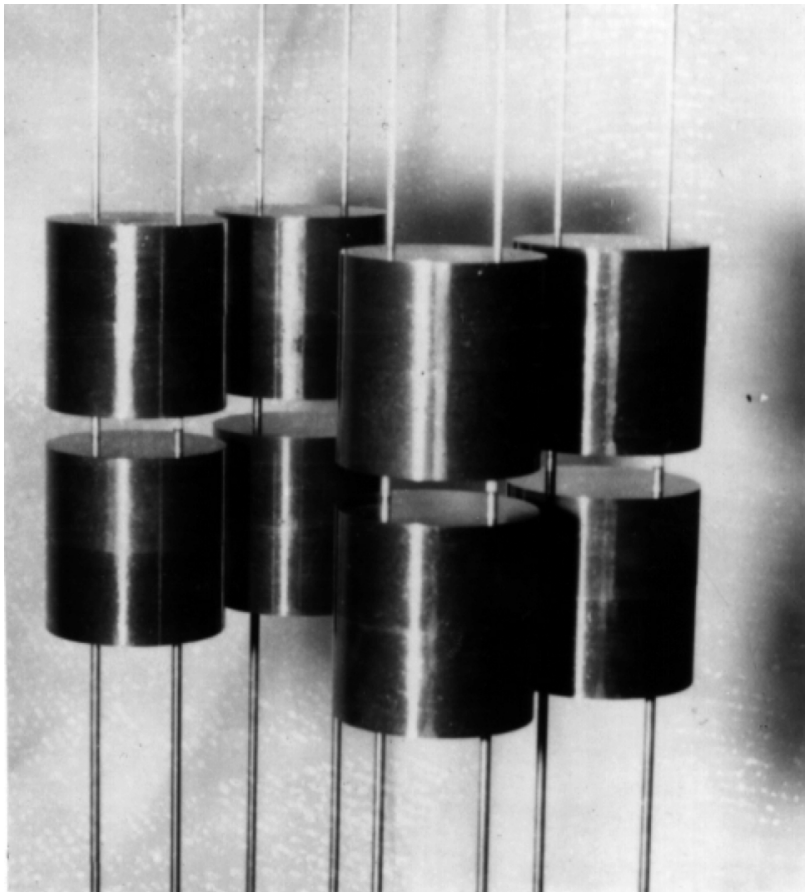


Fig. 8.1.233: Critical 2C8 bare assembly.

Sample Problem 2 CASE 2C8 BARE WITH 8 UNIT TYPES MATRIX CALCULATION

This problem is the same as sample problem 1 except it is set up as a mixed unit problem with each unit of the array defined as a different unit type. Matrix k-effectives will be calculated for this problem by both unit type and array position. The print flags are set to print all matrix data.

Input Data

KENO V.a

```
=kenova
sample problem 2  2c8 bare with 8 unit types matrix calculation
read param
  lib=4  flx=yes fdn=yes
  mku=yes fmu=yes mkp=yes fmp=yes
  htm=no
end param
read geometry
  unit 1
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit 2
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit 3
```

(continues on next page)

(continued from previous page)

```
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit 4
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit  5
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit 6
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit 7
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit 8
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
end geom
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read array
  nux=2 nuy=2 nuz=2  loop
    10*1
    3*2 7*1
    3 1 1 1 2 2 1 1 1 1
    4 2 2 1 2 2 1 1 1 1
    5 6*1 2 2 1
    6 2 2 1 1 1 1 2 2 1
    7  1 1 1 2 2 1 2 2 1
    8 2 2 1 2 2 1 2 2 1    end loop
end array
end data
end
```

KENO-VI

```
=kenovi
kenovi sample problem 2  case 2c8 bare with 8 unit types matrix cal
read param
  lib=4 flx=yes fdn=yes mku=yes cku=yes fmu=yes mkp=yes ckp=yes fmp=yes
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    cylinder 10 5.748 5.3825 -5.3825
    cuboid   20 4p6.87 2p6.505
    media  1 1 10     vol=1117.371078
    media  0 1 20 -10 vol=1338.755598
    boundary   20
  unit 2
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  4p6.87 2p6.505
    media  1 1 10     vol=1117.371078
    media  0 1 20 -10 vol=1338.755598
    boundary   20
  unit 3
    cylinder 10  5.748 5.3825 -5.3825
```

(continues on next page)

(continued from previous page)

```
    cuboid   20  4p6.87 2p6.505
    media  1 1 10     vol=1117.371078
    media  0 1 20 -10 vol=1338.755598
    boundary  20
  unit 4
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  4p6.87 2p6.505
    media  1 1 10     vol=1117.371078
    media  0 1 20 -10 vol=1338.755598
    boundary 20
  unit 5
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  4p6.87 2p6.505
    media  1 1 10     vol=1117.371078
    media  0 1 20 -10 vol=1338.755598
    boundary  20
  unit 6
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20 4p6.87 2p6.505
    media  1 1 10     vol=1117.371078
    media  0 1 20 -10 vol=1338.755598
    boundary  20
  unit 7
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20 4p6.87 2p6.505
    media  1 1 10     vol=1117.371078
    media  0 1 20 -10 vol=1338.755598
    boundary 20
  unit 8
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20 4p6.87 2p6.505
    media  1 1 10     vol=1117.371078
    media  0 1 20 -10 vol=1338.755598
    boundary  20
  global unit 9
    cuboid  10 4p13.74 2p13.01
    com='2x2x2  2c8 array'
    array 1 +10 place 1 1 1 2r-6.87 -6.505
    boundary  10
end geometry
read array
 ara=1 nux=2 nuy=2 nuz=2  gbl=1
 loop  10*1
       3*2 7*1
       3 1 1 1 2 2 1 1 1 1
       4 2 2 1 2 2 1 1 1 1
       5 6*1 2 2 1
       6 2 2 1 1 1 1 2 2 1
       7  1 1 1 2 2 1 2 2 1
       8 2 2 1 2 2 1 2 2 1 end loop
end array
end data
end
```

Sample Problem 3 2C8 15.24-CM PARAFFIN REFL

A 2 × 2 × 2 array of uranium metal cylinders is reflected by 6 in. of paraffin on all faces (Fig. 8.1.233). This critical experiment[1] is designated as cylinder index 11 and reflector index 5 in Table II of Ref. 1. Fig. 8.1.234 shows half of the critical experiment, which consisted of the half shown and the mirror image of it. These two assemblies were moved together to achieve criticality. The top reflector is missing in Fig. 8.1.234, but was present when criticality was achieved.

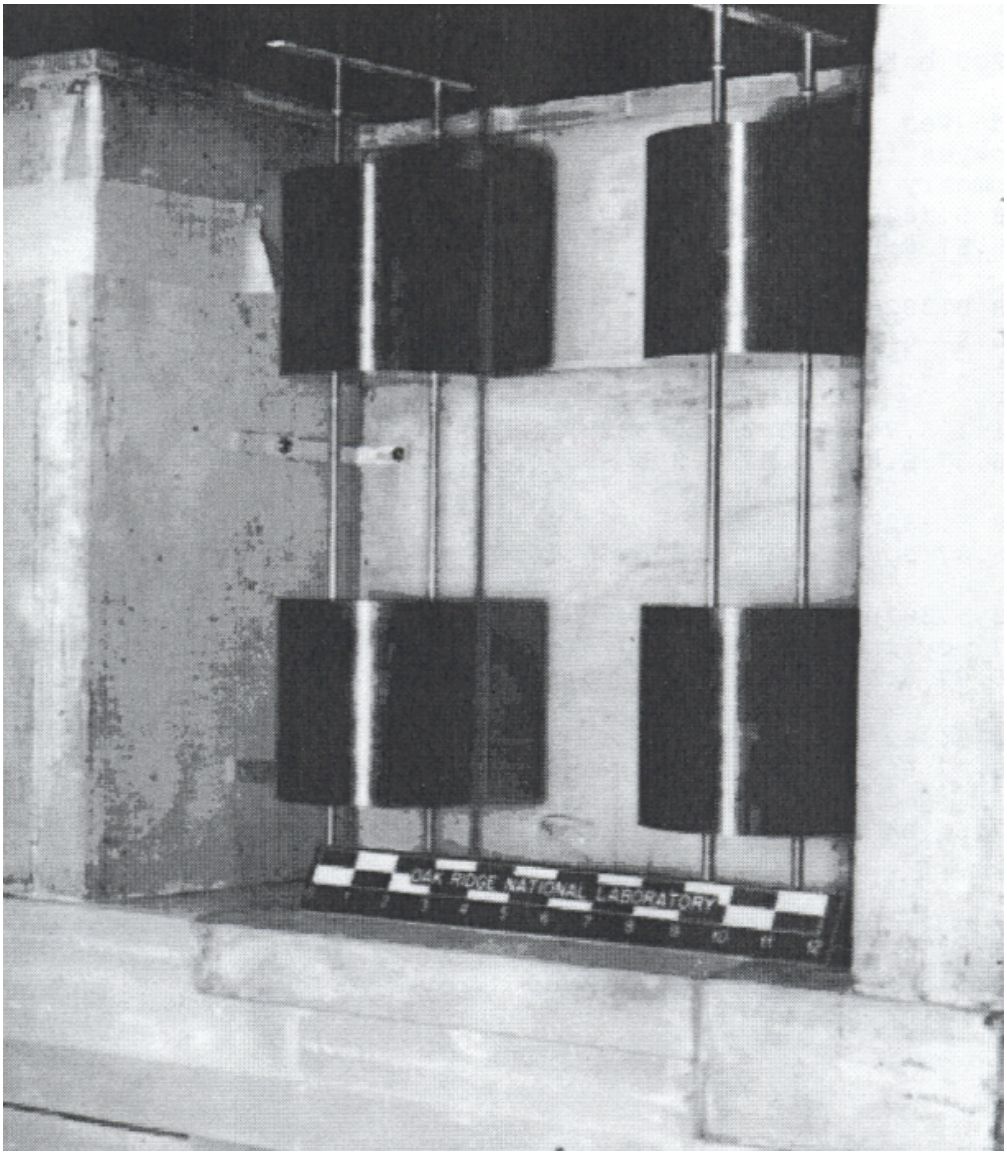


Fig. 8.1.234: Half of the paraffin reflected 2C8 assembly before the top reflector was added.

Input Data

KENO V.a

```
=kenova
sample problem 3  2c8  15.24 cm paraffin refl
```

(continues on next page)

(continued from previous page)

```
read param
  lib=4  flx=yes fdn=yes pwt=yes
  htm=no
end param
read array
  nux=2 nuy=2 nuz=2 fill f1 end fill
end array
read mixt
  mix=1  ncm=1
    92234 4.82717e-04   92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
  mix=2  ncm=10
     6000 3.84193e-02 9001001 7.99120e-02
  sct=2
end mixt
read geom
  unit 1
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 11.74 -11.74 11.74 -11.74 11.375 -11.375
  global unit 2
    array   1 -23.48 -23.48 -22.75
    cuboid  2 2 26.48 -26.48 26.48 -26.48 25.75 -25.75
    cuboid  2 3 29.48 -29.48 29.48 -29.48 28.75 -28.75
    cuboid  2 4 32.48 -32.48 32.48 -32.48 31.75 -31.75
    cuboid  2 5 35.48 -35.48 35.48 -35.48 34.75 -34.75
    cuboid  2 6 38.72 -38.72 38.72 -38.72 37.99 -37.99
end geom
read bias
  id=400 2 6
end bias
end data
end
```

KENO-VI

```
=kenovi
keno-vi sample problem 3  2c8  15.24 cm paraffin refl
read param
  lib=4 flx=yes fdn=yes pwt=yes
  htm=no gen=300 nsk=10 npg=2000
end param
read mixt
  mix=1  ncm=1
    92234 4.82717e-04   92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
  mix=2  ncm=10
     6000 3.84193e-02 9001001 7.99120e-02
  sct=2
end mixt
read geometry
  unit 1
    com='single 2c8 unit centered'
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  4p11.74 2p11.375
    media  1 1 10      vol=8938.968624
    media  0 1 20 -10  vol=10710.044784
    boundary  20
  global unit 2
    com='2x2x2  2c8 array with reflector'
    cuboid 10 4p23.48 2p22.75
    cuboid 20   26.48 -26.48 26.48 -26.48 25.75 -25.75
    cuboid 30   29.48 -29.48 29.48 -29.48 28.75 -28.75
    cuboid 40   32.48 -32.48 32.48 -32.48 31.75 -31.75
    cuboid 50   35.48 -35.48 35.48 -35.48 34.75 -34.75
    cuboid 60   38.72 -38.72 38.72 -38.72 37.99 -37.99
    array 1 +10 place 1 1 1 2r-11.74 -11.375
    media  2 2 -10 +20  vol=4.41067E+04
```

(continues on next page)

(continued from previous page)

```
    media  2 3 -20 +30  vol=5.54410E+04
    media  2 4 -30 +40  vol=6.80712E+04
    media  2 5 -40 +50  vol=8.19974E+04
    media  2 6 60 -50   vol=1.05694E+05
    boundary  60
end geometry
read bias
  id=400 2 6
end bias
read array
  ara=1 nux=2 nuy=2 nuz=2 fill f1 end fill
end array
end data
end
```

Sample Problem 4 2C8 15.24-CM PARAFFIN REFL AUTOMATIC REFL

This problem is the same as sample problem 3 except it is set up using more reflector regions.

Input Data

KENO V.a

```
=kenova
sample problem 4  2c8 15.24 cm paraffin refl automatic refl
read param
  pwt=yes lib=4  flx=yes fdn=yes
  htm=no
end param
read geometry
  unit 1
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 11.74 -11.74 11.74 -11.74 11.375 -11.375
  global unit 2
    array      1 -23.48 -23.48 -22.75
    reflector  2 2 6*3.0 5
    reflector  2 7 6*.24 1
end geom
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04   92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
  mix=2  ncm=10
     6000 3.84193e-02 9001001 7.99120e-02
end mixt
read arra
  nux=2 nuy=2 nuz=2 fill f1 end fill
end array
read bias
  id=400 2 7
end bias
end data
end
```

KENO-VI

```
=kenovi
keno-vi sample problem 4  2c8  15.24 cm paraffin refl
read param
  lib=4 flx=yes fdn=yes pwt=yes
  htm=no
end param
read mixt
```

(continues on next page)

(continued from previous page)

```
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04   92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
  mix=2  ncm=10
     6000 3.84193e-02 9001001 7.99120e-02
end mixt
read geometry
  unit 1
    com='single 2c8 unit centered'
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  4p11.74 2p11.375
    media  1 1 10
    media  0 1 20 -10
    boundary  20
  global unit 2
    com='2x2x2  2c8 array with reflector'
    cuboid 10 4p23.48 2p22.75
    cuboid 20   26.48 -26.48 26.48 -26.48 25.75 -25.75
    cuboid 30   29.48 -29.48 29.48 -29.48 28.75 -28.75
    cuboid 40   32.48 -32.48 32.48 -32.48 31.75 -31.75
    cuboid 50   35.48 -35.48 35.48 -35.48 34.75 -34.75
    cuboid 60   38.48 -38.48 38.48 -38.48 37.75 -37.75
    cuboid 70   38.72 -38.72 38.72 -38.72 37.99 -37.99
    array 1 +10 place 1 1 1 2r-11.74 -11.375
    media  2 2 -10 +20
    media  2 3 -20 +30
    media  2 4 -30 +40
    media  2 5 -40 +50
    media  2 6  60 -50
    media  2 7  70 -60
    boundary 70
end geometry
read volume
   type=trace
end volume
read bias
  id=400 2 7
end bias
read array
  ara=1 nux=2 nuy=2 nuz=2 fill f1 end fill
end array
end data
end
```

Sample Problem 5 2C8 12-INCH PARAFFIN ALBEDO REFLECTOR

This problem is the same as samples problems 3 and 4 except the reflector is represented by a 12 in. paraffin albedo. Note the decrease in execution time when using an albedo reflector instead of doing actual tracking. Note also that k-effective is somewhat higher for this system, probably due to the small edge size of the system [KENO-Appendix-CWT69].

Input Data

KENO V.a

```
=kenova
sample problem 5  2c8 12 inch paraffin albedo reflector
read para
  flx=yes far=yes gas=no fdn=yes lib=4
  htm=no
end para
read array
  nux=2 nuy=2 nuz=2 fill f1 end fill
```

(continues on next page)

(continued from previous page)

```
end array
read mixt
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
  sct=2
end mixt
read bounds
  all=paraffin
end bounds
read geom
  unit 1
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 11.74 -11.74 11.74 -11.74 11.375 -11.375
end geom
end data
end
```

KENO-VI

```
=kenovi
kenovi sample problem 5  2c8 12 inch paraffin albedo reflector
read para
  flx=yes far=yes gas=no fdn=yes lib=4
  htm=no
end para
read mixt
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
  sct=2
end mixt
read bounds
  all=paraffin
end bounds
read geometry
  unit 1
    com='single 2c8 unit centered'
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20 4p11.74 2p11.375
    media  1 1 10
    media  0 1 20 -10
    boundary  20
  global unit 2
    cuboid 10  4p23.48 2p22.75
    com='2x2x2  2c8 array'
    array 1 +10 place 1 1 1 2r-11.74 -11.375
    boundary 10
end geometry
read array
  ara=1 nux=2 nuy=2 nuz=2  fill f1 end fill
end array
read volume
  type=random
end volume
end data
end
```

Sample Problem 6 ONE 2C8 UNIT (SINGLE UNIT)

One of the 2C units[1] is described and run as a single-unit problem, and its k-effective is calculated.

Input Data

KENO V.a

```
=kenova
sample problem 6  one 2c8 unit (single unit)
read para
  lib=4  flx=yes fdn=yes far=yes gas=no
  htm=no
end para
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    cylinder  1 1 5.748 5.3825 -5.3825
end geometry
end data
end
```

KENO-VI

```
=kenovi
kenovi sample problem 6  one 2c8 unit (single unit)
read para
  lib=4 flx=yes fdn=yes far=yes gas=no
  htm=no
end para
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  global unit 1
    com='single 2c8 unit centered'
    cylinder 10  5.748 5.3825 -5.3825
    media  1 1 10  vol=1117.3710776
    boundary  10
end geometry
end data
end
```

Sample Problem 7 BARE 2C8 USING SPECULAR REFLECTION

One of the 2C units[1] is described and the $2 \times 2 \times 2$ array is simulated by using specular reflection on the positive X, Y, and Z faces of the unit. This is a simulation of sample problem 1.

Input Data

KENO V.a

```
=kenova
sample problem 7  bare 2c8 using specular reflection
read para
  lib=4  flx=yes fdn=yes far=yes gas=no
  htm=no
```

(continues on next page)

(continued from previous page)

```
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geom
  unit 1
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
end geom
read bounds
  +fc=specular
end bounds
end data
end
```

KENO-VI

```
=kenovi
keno-vi  sample problem 7  bare 2c8 using specular reflection
read para
  flx=yes fdn=yes far=yes gas=no lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  global unit 1
    com='single 2c8 unit centered'
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  4p6.87 2p6.505
    media  1 1 10      vol=1117.371078
    media  0 1 20 -10  vol=1338.755598
    boundary  20
end geometry
read bounds
  +fc=specular
end bounds
end data
end
```

Sample Problem 8 INFINITELY LONG CYLINDER FROM 2C8 UNIT

The fuel and cylinder radius from sample problem 1 is used. The length of the cylinder is arbitrarily chosen to be 20 cm, and the unit is specularly reflected on the top and bottom to create an infinitely long cylinder.

Input Data

KENO V.a

```
=kenova
sample problem 8  infinitely long cylinder from 2c8 unit
read param
  lib=4
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=1
```

(continues on next page)

(continued from previous page)

```
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    cylinder  1 1 5.748 10.0 -10.0
    cuboid  0 1 6.87 -6.87 6.87 -6.87 10.0 -10.0
end geometry
read bounds
  zfc=mirror
end bounds
end data
end
```

KENO-VI

```
=kenovi
keno-vi  sample problem 8 infinitely long cylinder from 2c8 unit
read parameters
  lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  global unit 1
    com='single 2c8 unit centered'
    cylinder 10  5.748 2p10.0
    cuboid   20  4p6.87 2p10.0
    media  1 1 10
    media  0 1 20 -10
    boundary  20
end geometry
read bounds
  zfc=mirror
end bounds
read volume
  type=trace  iface=zface
end volume
end data
end
```

Sample Problem 9 INFINITE ARRAY OF 2C8 UNITS

Input Data

KENO V.a

```
=kenova
sample problem 9  infinite array of 2c8 units
read param
  lib=4  gen=103
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read boun
  all=mir
```

(continues on next page)

(continued from previous page)

```
end boun
read geom
  unit 1
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
end geom
end data
end
```

KENO-VI

```
=kenovi
keno-vi  sample problem 9  infinite array of 2c8 units
read parameters
  lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  global unit 1
    com='single 2c8 unit centered'
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  4p6.87 2p6.505
    media  1 1 10      vol=1117.371078
    media  0 1 20 -10  vol=1338.755598
    boundary  20
end geometry
read bounds
  all=mirror
end bounds
end data
end
```

Sample Problem 10 2C8 BARE WRITE RESTART

The geometry description from sample problem 1 is used, and the cuboid is specularly reflected on all faces to create an infinite array of 2C8 units having an edge-to-edge spacing of 2.244 cm in the X and Y directions and 2.245 cm in the Z direction.

Input Data

KENO V.a

```
=kenova
sample problem 10  case 2c8 bare  write restart
read parameters
  flx=yes fdn=yes far=yes gas=no lib=4  res=5 wrs=94
  app=yes
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    cylinder 1 1 5.748 5.3825 -5.3825
```

(continues on next page)

(continued from previous page)

```
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
end geometry
read array
  nux=2 nuy=2 nuz=2 fill f1 end fill
end array
end data
end
```

KENO-VI

```
=kenovi
sample problem 10  case 2c8 bare  write restart
read parameters
  flx=yes fdn=yes far=yes gas=no lib=4 res=5 wrs=94 app=yes
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='single 2c8 unit centered'
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  4p6.87 2p6.505
    media  1 1 10      vol=8938.968624
    media  0 1 20 -10  vol=10710.044784
    boundary  20
  global unit 2
    cuboid 10  4p13.74 2p13.01
    com='2x2x2  2c8 array'
    array 1 +10 place 1 1 1 2r-6.87 -6.505
    boundary  10
end geometry
read array
  ara=1 nux=2 nuy=2 nuz=2  fill f1  end fill
end array
end data
end
```

Sample Problem 11 2C8 BARE READ RESTART DATA

This problem is a restart of sample problem 10. The problem is restarted from the tenth set of restart data that was written by sample problem 10 (i.e., it restarts with the fifty-first generation).

Input Data

KENO V.a

```
=kenova
sample problem 11  2c8 bare  read restart data
read param
  beg=51  rst=94 res=0
  htm=no
end param
end data
end
```

KENO-VI

```
=kenovi
sample problem 11  2c8 bare  read restart data
read param
  beg=51 rst=94 res=0
  htm=no
end param
end data
end
```

Sample Problem 12 4 AQUEOUS 4 METAL

This problem is a critical experiment consisting of a composite array[1] of four highly enriched uranium metal cylinders and four cylindrical Plexiglas containers filled with uranyl nitrate solution. The metal units in this experiment are designated in Table II of Ref. 1 as cylinder index 11 and reflector index 1. A photograph of the experiment is given in Fig. 8.1.235.

Fig. 8.1.235: Critical assembly of 4 solution units and 4 metal units.

KENO V.a

```
=kenova
sample problem 12 4 aqueous 4 metal mixed units
read param
  lib=4 fdn=yes nub=yes smu=yes mkp=yes
  mku=yes fmp=yes fmu=yes
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
  mix=2  ncm=2
     1001 5.77931e-02    7014 2.13092e-03    8016 3.74114e-02
    92234 1.06784e-05   92235 9.84602e-04   92236 5.29386e-06
    92238 6.19414e-05
  mix=3  ncm=11
     1001 5.67873e-02    6000 3.54921e-02    8016 1.41968e-02
end mixt
read geom
  unit 1
    cylinder  2 1 9.525 8.89 -8.89
    cylinder  3 1 10.16 9.525 -9.525
    cuboid  0 1 10.875 -10.875 10.875 -10.875 10.24 -10.24
  unit  2
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.59 -15.16 6.59 -15.16 6.225 -14.255
  unit  3
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.59 -15.16 15.16 -6.59 6.225 -14.255
  unit  4
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.59 -15.16 6.59 -15.16 14.255 -6.225
  unit  5
    cylinder  1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.59 -15.16 15.16 -6.59 14.255 -6.225
end geom
read array
  gbl=1 ara=1 nux=2 nuy=2 nuz=2  loop
    1 3r2 1 2 1 1 2 1
    2 9r1
    3 3r1 2 2 1 3r1
    4 6r1 2 2 1
    5 3r1 2 2 1 2 2 1  end loop
end array
end data
end
```

KENO-VI

```
=kenovi
sample problem 12 4 aqueous 4 metal mixed units
read param
  lib=4  flx=yes fdn=yes nub=yes smu=yes mku=yes fmp=yes fmu=yes
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
  mix=2  ncm=2
     1001 5.77931e-02    7014 2.13092e-03    8016 3.74114e-02
    92234 1.06784e-05   92235 9.84602e-04   92236 5.29386e-06
```

(continues on next page)

(continued from previous page)

```
    92238 6.19414e-05
  mix=3  ncm=11
     1001 5.67873e-02    6000 3.54921e-02    8016 1.41968e-02
end mixt
read geom
  unit 1
    cylinder 10  9.525 8.89 -8.89
    cylinder 20  10.16 9.525 -9.525
    cuboid   30  10.875 -10.875 10.875 -10.875 10.24 -10.24
    media 2 1 10       vol=20270.8327
    media 3 1 -10 20   vol=4440.27764
    media 0 1 30 -20   vol=14042.16966
    boundary  30
  unit      2
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  6.59 -15.16 6.59 -15.16 6.225 -14.255
    media 1 1 10       vol=1117.371078
    media 0 1 20 -10   vol=8570.948922
    boundary  20
  unit      3
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  6.59 -15.16 15.16 -6.59 6.225 -14.255
    media 1 1 10       vol=1117.371078
    media 0 1 20 -10   vol=8570.948922
    boundary  20
  unit      4
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  6.59 -15.16 6.59 -15.16 14.255 -6.225
    media 1 1 10       vol=1117.371078
    media 0 1 20 -10   vol=8570.948922
    boundary  20
  unit      5
    cylinder 10  5.748 5.3825 -5.3825
    cuboid   20  6.59 -15.16 15.16 -6.59 14.255 -6.225
    media 1 1 10       vol=1117.371078
    media 0 1 20 -10   vol=8570.948922
    boundary  20
  global
  unit  6
    cuboid 10  43.5 0.0 43.5 0.0 40.96 0.0
    array 1 +10 place 1 1 1 15.16 15.16 14.255
    boundary  10
end geom
read array
  gbl=1 ara=1 nux=2 nuy=2 nuz=2  loop
  1 3r2 1 2 1 1 2 1
  2 9r1
  3 3r1 2 2 1 3r1
  4 6r1 2 2 1
  5 3r1 2 2 1 2 2 1  end loop
end array
end data
end
```

Sample Problem 13 TWO CUBOIDS IN A CYLINDRICAL ANNULUS

This critical experiment [KENO-Appendix-CIM64] consists of two assemblies of 93.2% ^{235}U-enriched uranium metal (ρ = 18.69^{g}/cc) stacked vertically. The bottom assembly contains a uranium metal cuboid offset to the left within a uranium metal cylindrical annulus. The top assembly contains a uranium metal cuboid offset to the right within a uranium metal cylindrical annulus. The cuboid extends above the annulus. A drawing of the two sections and the total assembly is given in Fig. 8.1.236.

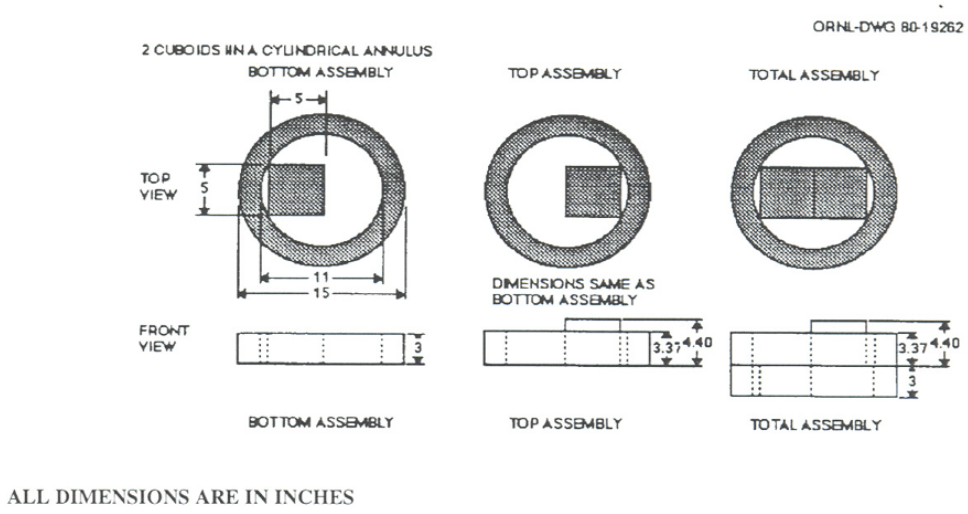


Fig. 8.1.236: Drawing of two cuboids in an annulus critical assembly.

Input Data

KENO V.a

```
=kenova
sample problem 13  two cuboids in a cylindrical annulus
read param
   lib=4
  htm=no
end param
read geom
  unit 1
    cuboid  1 1 6.35 -6.35 6.35 -6.35 7.62 0.0
    cylinder  0 1 13.97 7.62 0.0 orig -6.0934 0.0
    cylinder  1 1 19.05 7.62 0.0 orig -6.0934 0.0
    cuboid  0 1 12.9566 -25.1434 19.05 -19.05 7.62 0.0
  unit 2
    cuboid  1 1 6.35 -6.35 6.35 -6.35 8.56 0.0
    cylinder  0 1 13.97 8.56 0.0 origin 6.0934 0.0
    cylinder  1 1 19.05 8.56 0.0 origin 6.0934 0.0
    cuboid  0 1 25.1434 -12.9566 19.05 -19.05 8.56 0.0
  unit 3
    cuboid  1 1 6.35 -6.35 6.35 -6.35 2.616 0.0
    cuboid  0 1 25.1434 -12.9566 19.05 -19.05 2.616 0.0
end geom
read mixt
  sct=2
  mix=1  ncm=3
    92234 4.80916e-04  92235 4.46300e-02  92236 9.53661e-05 92238 2.64776e-03
end mixt
read array
```

(continues on next page)

(continued from previous page)

```
  gbl=1 nux=1 nuy=1 nuz=3  fill 1 2 3 end fill
end array
end data
end
```

KENO-VI

```
=kenovi
sample problem 13  two cuboids in a cylindrical annulus
read param
  lib=4
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=3
    92234 4.80916e-04  92235 4.46300e-02  92236 9.53661e-05 92238 2.64776e-03
end mixt
read geom
  unit 1
    cuboid    10 6.35 -6.35 6.35 -6.35 7.62 0.0
    cylinder  20 13.97 7.62 0.0 orig x=-6.0934
    cylinder  30 19.05 7.62 0.0 orig x=-6.0934
    cuboid    40 12.9566 -25.1434 19.05 -19.05 7.62 0.0
    media  1 1 10     vol=1229.0298
    media  0 1 20 -10 vol=3442.914497898
    media  1 1 30 -20 vol=4015.555429598
    media  0 1 40 -30 vol=2373.768472504
    boundary  40
  unit 2
    cuboid   10 6.35 -6.35 6.35 -6.35 8.56 0.0
    cylinder 20  13.97 8.56 0.0 origin x=6.0934
    cylinder 30  19.05 8.56 0.0 origin x=6.0934
    cuboid   40 25.1434 -12.9566 19.05 -19.05 8.56 0.0
    media  1 1 10     vol=1380.6424
    media  0 1 20 -10 vol=3867.630984515
    media  1 1 30 -20 vol=4510.912661071
    media  0 1 40 -30 vol=2666.595554414
    boundary  40
  unit 3
    cuboid 10  6.35 -6.35 6.35 -6.35 2.616 0.0
    cuboid 20 25.1434 -12.9566 19.05 -19.05 2.616 0.0
    media  1 1 10     vol=421.93464
    media  0 1 20 -10 vol=3375.47712
    boundary  20
  global unit 4
    cuboid  10 12.9566 -25.1434 2p19.05 18.796 0.
    array 1 10 place 1 1 1 3r0.
    boundary  10
end geom
read array
  ara=1 nux=1 nuy=1 nuz=3  fill 1 2 3 end fill
end array
end data
end
```

Sample Problem 14 U METAL CYLINDER IN AN ANNULUS

This critical experiment[3] consists of a 93.2 ^{235}U-enriched uranium metal cylinder within a cylindrical annulus of the same material as shown in Fig. 8.1.237. The uranium metal specification is identical to that used in sample problem 13.

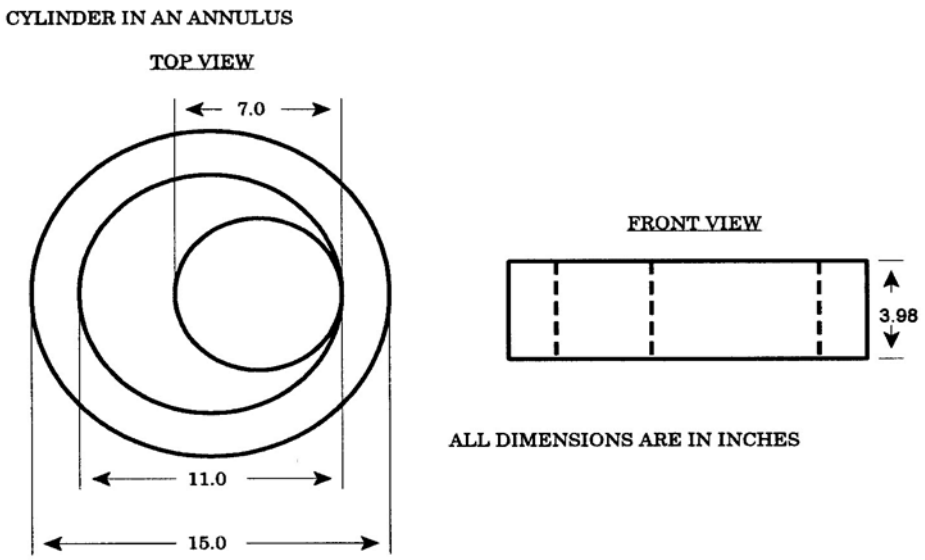


Fig. 8.1.237: Drawing of the cylinder in an annulus critical assembly.

Input Data

KENO V.a

```
=kenova
sample problem 14  u metal cylinder in an annulus
read param
  lib=4
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=3
    92234 4.80916e-04  92235 4.46300e-02  92236 9.53661e-05 92238 2.64776e-03
end mixt
read geom
  global unit 1
    cylinder  1 1 8.89 10.109 0.0 orig 5.0799 0.0
    cylinder  0 1 13.97 10.109 0.0
    cylinder  1 1 19.05 10.109 0.0
end geom
end data
end
```

KENO-VI

```
=kenovi
sample problem 14  u metal cylinder in an annulus
read param
```

(continues on next page)

(continued from previous page)

```
  lib=4
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=3
    92234 4.80916e-04  92235 4.46300e-02  92236 9.53661e-05 92238 2.64776e-03
end mixt
read geom
  global unit 1
    cylinder 10  8.89 10.109 0.0 orig x=5.08
    cylinder 20  13.97 10.109 0.0
    cylinder 30  19.05 10.109 0.0
    media  1 1 10          vol=2509.929894
    media  0 1 20 -10      vol=3688.060252
    media  1 1 30 -20 -10  vol=5327.198142
    boundary  30
end geom
end data
end
```

Sample Problem 15 SMALL WATER REFLECTED SPHERE ON PLEXIGLAS COLLAR

This critical experiment [KENO-Appendix-CBKH+77] is a small highly enriched uranium sphere supported by a Plexiglas doughnut in a tank of water. The sphere extends down through the hole of the doughnut. However, the KENO geometry package cannot rigorously describe a doughnut (torus) with either KENO V.a or KENO-VI. Therefore, the KENO mockup of this problem describes the doughnut as an annular cylindrical plate and the sphere is supported by it. Both are contained in a cylindrical tank of water. A drawing of the experiment is given in Fig. 8.1.238. This drawing shows the sphere above the cylindrical collar for the sake of clarity. The sphere is actually supported by the collar and extends into the opening in its center. The actual experiment utilized a torus or doughnut instead of a cylindrical collar.

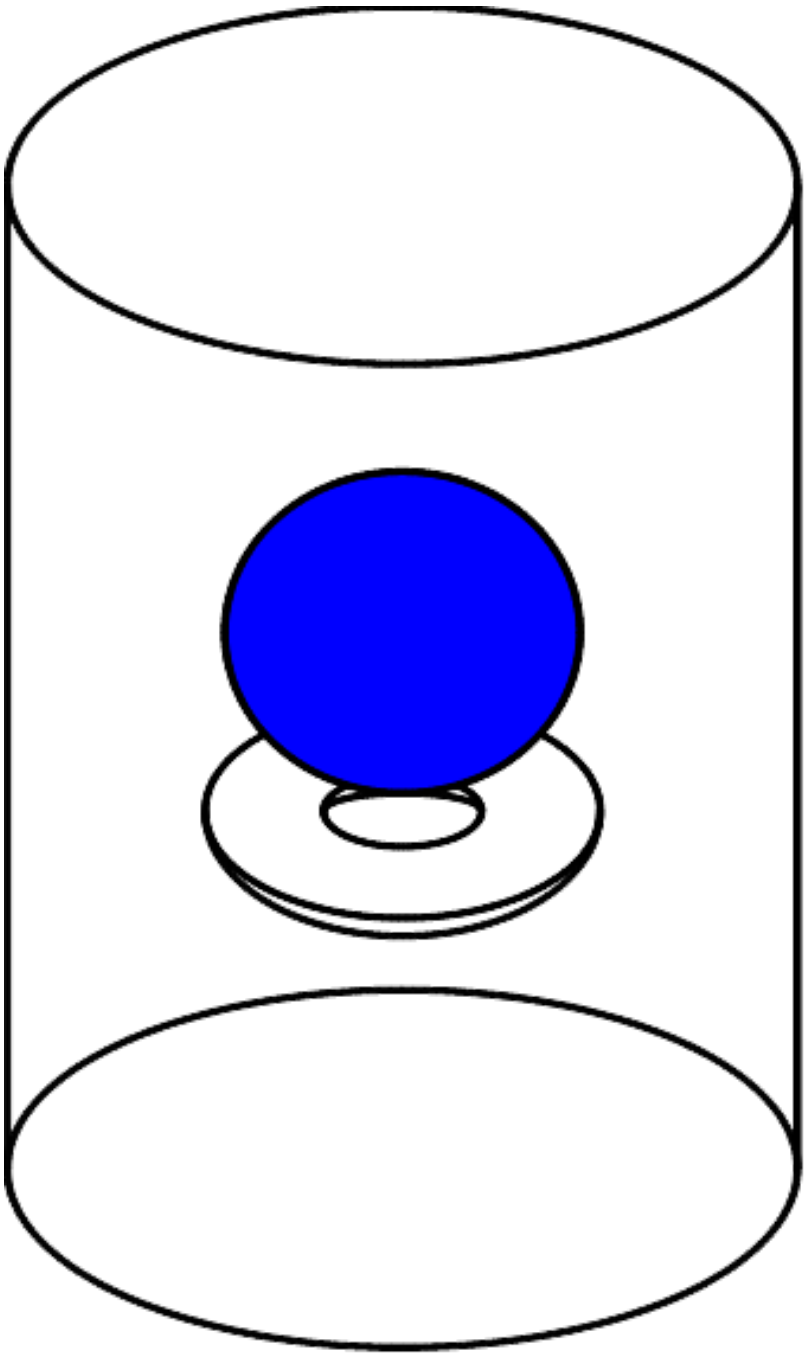


Fig. 8.1.238: Drawing of a critical assembly consisting of a uranium sphere on a Plexiglas collar with a cylindrical water reflector.

Input Data

KENO V.a

```
=kenova
sample problem 15  small water reflected sphere on plexiglas collar
read param
  lib=4  flx=yes fdn=yes
```

(continues on next page)

(continued from previous page)

```
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=4
     92234 5.27115e-04   92235 4.70308e-02   92236 1.00692e-04  92238 4.89708e-04
  mix=2  ncm=11
      1001 5.67873e-02    6000 3.54921e-02    8016 1.41968e-02
  mix=3  ncm=12
      1001 6.67554e-02
  mix=3  ncm=12
      8016 3.33757e-02
end mixt
read geom
  unit 1
    hemisphe-z 1 1 6.5537 chord -5.09066
    cylinder  3 1 4.1275 -5.09066 -7.63065
    cylinder  2 1 12.7   -5.09066 -7.63065
    cuboid  3 1 4p12.7 -5.09066 -7.63065
  unit 2
    hemisphe+z 1 1 6.5537 chord 5.09066
    cuboid  3 1 4p12.7 6.5537 -5.09066
  global unit 3
    array     1 -12.7 -12.7 -7.092175
    cylinder  3 1 17.97 2p7.0922
    replicate 3 2 3*3.0 5
end geom
read bias
  id=500 2 6
end bias
read array
  nux=1 nuy=1 nuz=2  fill 1 2 end fill
end array
read plot
  scr=yes lpi=10
  ttl='x-z slice through the center of the sphere'
  xul=-20.0 zul=10.0 yul=0.0  xlr=20.0 ylr=0.0 zlr=-10.0
  uax=1.0 wdn=-1.0 nax=400
end plot
end data
end
```

KENO-VI

```
=kenovi
sample problem 15  small water reflected sphere on plexiglas collar
read param
  lib=4 flx=yes fdn=yes plt=yes
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=4
     92234 5.27115e-04   92235 4.70308e-02   92236 1.00692e-04  92238 4.89708e-04
  mix=2  ncm=11
      1001 5.67873e-02    6000 3.54921e-02    8016 1.41968e-02
  mix=3  ncm=12
      1001 6.67554e-02
  mix=3  ncm=12
      8016 3.33757e-02
end mixt
read geom
  global unit 1
    sphere   10  6.5537
    cylinder 20  4.1275 -5.09066 -7.63065
```

(continues on next page)

(continued from previous page)

```
    cylinder 30  12.7   -5.09066 -7.63065
    cylinder 40  21.5537 21.5537 -21.5537
    media  1 1 10  vol=1179.093598091
    media  3 1 20 -10  vol=95.1516
    media  2 1 30 -20 -10  vol=1151.089182028
    media  3 1 40 -30 -20 -10  vol=60488.221616778
    boundary  40
end geom
read plot
  scr=yes  lpi=10
  ttl='x-z slice through the center of the sphere'
  xul=-20.0 zul=10.0 yul=0.0  xlr=20.0 ylr=0.0 zlr=-10.0
  uax=1.0 wdn=-1.0 nax=400
end plot
end data
end
```

Sample Problem 16 UO2F2 INFINITE SLAB K-INFINITY

This problem solves for the k-infinity of an infinite number of slabs of uranyl fluoride solution contained in Pyrex glass and separated by borated uranyl fluoride solution. The uranyl fluoride slab is 4.958 cm thick, 93.2% enriched, and has a density of 578.7 g U/l. The Pyrex glass is 1.27 cm thick and is present on both faces of the uranyl fluoride solution. A total of 27.46 cm of borated solution separates the Pyrex glass of adjacent slabs of solution. 1.482×10^{-27} atoms of boron per milliliter are present in the borated solution.

Input Data

KENO V.a

```
=kenova
sample problem 16 uo2f2 infinite slab k-infinity
read parameters
  lib=4  amx=yes xap=no
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=5
     9019 2.96287e-03   1001 6.08125e-02   8016 3.33691e-02  92235 1.38188e-03  92238 9.95505e-05
  mix=2  ncm=13
    11023 2.39503e-03  13027 4.97720e-04  14028 1.66260E-02  14029 8.41845E-04  14030 5.58826E-04
          5010 9.14627e-04   5011 3.68149e-03   8016 4.49174e-02
  mix=3  ncm=6
     9019 2.96287e-03   1001 6.08125e-02   8016 3.33691e-02  92235 1.38188e-03  92238 9.95505e-05
          5010 2.94862e-04   5011 1.18686e-03
end mixt
read geometry
  global unit 1
    cuboid 1 1 2.479 -2.479 100 -100 100 -100
    cuboid 2 1 3.749 -3.749 100 -100 100 -100
    cuboid 3 1 17.479 -17.479 100 -100 100 -100
end geom
read bounds
  all=mirror
end bounds
end data
end
```

KENO-VI

```
=kenovi
sample problem 16 uo2f2 infinite slab k-infinity
read parameters
  lib=4 amx=yes xap=no
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=5
     9019 2.96287e-03   1001 6.08125e-02   8016 3.33691e-02  92235 1.38188e-03  92238 9.95505e-05
  mix=2  ncm=13
    11023 2.39503e-03  13027 4.97720e-04  14028 1.66260E-02  14029 8.41845E-04  14030 5.58826E-04
     5010 9.14627e-04   5011 3.68149e-03   8016 4.49174e-02
  mix=3  ncm=6
     9019 2.96287e-03   1001 6.08125e-02   8016 3.33691e-02  92235 1.38188e-03  92238 9.95505e-05
         5010 2.94862e-04   5011 1.18686e-03
end mixt
read geometry
  global unit 1
    cuboid 10  2.479 -2.479 100.0 -100.0 100.0 -100.0
    cuboid 20  3.749 -3.749 100.0 -100.0 100.0 -100.0
    cuboid 30  17.479 -17.479 100.0 -100.0 100.0 -100.0
    media 1 1 10
    media 2 1 20 -10
    media 3 1 30 -20 -10
    boundary  30
end geom
read bounds
  all=mirror
end bounds
read volume
  type=trace  iface=xface
end volume
end data
end
```

Sample Problem 17 93% UO2F2 SOLUTION SPHERE ADJOINT CALCULATION

A single 93% enriched uranyl fluoride sphere is run as an adjoint calculation. The result for the forward and adjoint k-effectives should be the same within statistical error when the problem is run both ways.

Input Data

KENO V.a

```
=kenova
sample problem 17 93% uo2f2 solution sphere  adjoint calculation
read parameters
  lib=4  npg=10000 nbk=10500 adj=yes amx=yes xap=no
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=7
     1001 6.55892e-02   8016 3.34755e-02   9019 6.80925e-04 92235 3.16910e-04  92238 2.35522e-05
end mixt
read geometry
  global unit 1
    sphere 1 1 16.0
end geom
end data
end
```

KENO-VI

```
=kenovi
sample problem 17 93% uo2f2 solution sphere  adjoint calculation
read parameters
  lib=4 amx=yes pwt=yes xap=no adj=yes npg=10000 nbk=10500 tba=0.5
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=7
     1001 6.55892e-02   8016 3.34755e-02   9019 6.80925e-04 92235 3.16910e-04  92238 2.35522e-05
end mixt
read geometry
  global unit 1
    sphere 10 16.0
    media 1 1 10  vol=17157.284678
    boundary  10
end geom
end data
end
```

Sample Problem 18 1F27 DEMONSTRATION OF OPTIONS

A reflected cubic array of 27 cylinders of aqueous uranyl nitrate in Plexiglas bottles [KENO-Appendix-CTho64]. The walls of the bottles were 0.64-cm thick, and each bottle was filled with 5 liters of 92.6% enriched solution at a concentration of 415 g/L, a specific gravity of 1.555 and 0.39 mg excess nitrate/g soln (From experimental facility documents. Not reported in ORNL/TM-719.) The $3 \times 3 \times 3$ array was surrounded by a 6-in. paraffin reflector. Most of the print options available in KENO are exercised in this problem. A perspective of this critical experiment is shown in Fig. 8.1.239. A photograph of one of the experiments utilized 27 of the Plexiglas bottles is shown in Fig. 8.1.240. Sample problem 18 has 15.24 cm of paraffin on all six faces rather than the 2.54-cm Plexiglas shown on five faces.

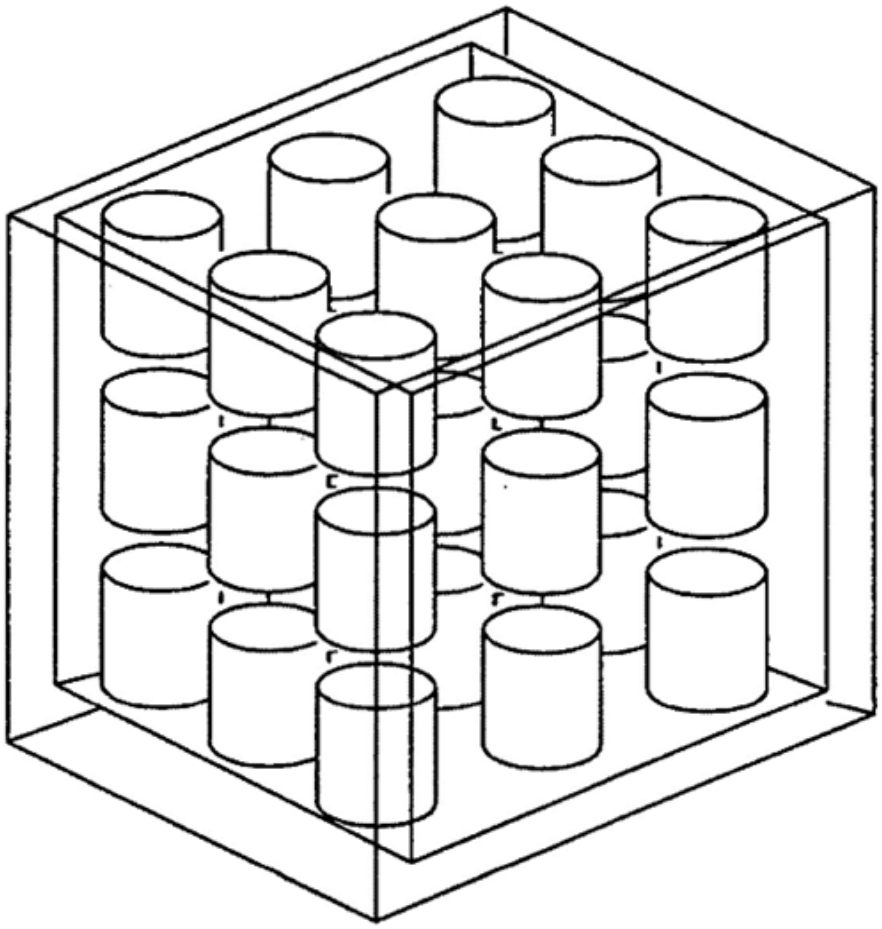


Fig. 8.1.239: Perspective of critical 1F27 experiment.

Fig. 8.1.240: View of a 27-unit array with 2.54-cm. thick Plexiglas reflector on five sides and a 15.24-cm. thick paraffin base.

Input Data

KENO V.a

```
=kenova
sample problem 18   1f27 demonstration of options problem
read para   gen=103 npg=1000 fdn=yes nub=yes lib=4
  mku=yes fmu=yes mkh=yes fmh=yes mka=yes fma=yes rnd=f12c09ed2195
  pwt=yes far=yes flx=yes amx=yes pax=yes pgm=yes
  htm=no
end para
read mixt
  sct=2
  mix=1  ncm=2
      1001 5.77931e-02    7014 2.13092e-03    8016 3.74114e-02
     92234 1.06784e-05   92235 9.84602e-04   92236 5.29386e-06
     92238 6.19414e-05
  mix=2  ncm=11
      1001 5.67873e-02    6000 3.54921e-02    8016 1.41968e-02
  mix=3  ncm=10
```

(continues on next page)

(continued from previous page)

```
      6000 3.84193e-02 9001001 7.99120e-02
  mix=4  ncm=15
      8016 3.33757e-11    1001 6.67515e-11
 end mixt
read bounds
  -zb= h2o
end bounds
read geom
  unit 1
    cylinder 1 1 9.52 8.7804 -8.7804
    cylinder 0 1 9.52 8.9896 -8.7804
    cylinder 2 1 10.16 9.6296 -9.4204
    cuboid 4 1 18.45 -18.45 18.45 -18.45 17.8946 -17.6854
  unit 2
    array 1 3*0.0
  unit 3
    array 2 3*0.0
  unit 4
    array 3 3*0.0
  unit 5
    array 4 3*0.0
  global
  unit 6
    cuboid 4 1 55.3501 -55.3501 55.3501 -55.3501 53.3701 -53.3701
     hole 2 -55.35   -18.45   -17.79
     hole 3 -55.35   -18.45   -53.3701
     hole 4  18.4501 -18.45   -53.3701
     hole 5 -55.3501 -55.3501 -53.3701
    replicate 3 2 6*3 5
    replicate 3 7 6*0.24 1
end geom
read bias
  id=400 2 7
end bias
read array
  ara=1 nux=2 nuy=2 nuz=2 fill f1 end fill
  ara=2 nux=2 nuy=2 nuz=1 fill f1 end fill
  ara=3 nux=1 nuy=2 nuz=3 fill f1 end fill
  ara=4 nux=3 nuy=1 nuz=3 fill f1 end fill
end array
read start
  nst=6 tfx=0.0 tfy=0.0 tfz=0.0
  lnu=1000 ps6=yes
end start
read plot
  scr=yes  plt=yes lpi=10
  ttl='  1f27 xy plot at z=0.0'
  xul=-71.0 yul= 71.0 zul=0.0
  xlr= 71.0 ylr=-71.0 zlr=0.0
  uax=1     vdn=-1    nax=400
  run=yes
  end plt1
  ttl='unit map 1f27 xy plot at z=0.0'
  pic=unit
end plot
end data
end
```

KENO-VI

```
=kenovi
sample problem 18   1f27  critical experiment
read para
  gen=103 npg=1000 fdn=yes nub=yes lib=4 plt=yes
  mku=yes cku=yes fmu=yes fmh=yes mka=yes cka=yes fma=yes pwt=yes
```

(continues on next page)

(continued from previous page)

```
  far=yes flx=yes amx=yes pax=yes pgm=yes rnd=f12c09ed2195
  htm=no
end para
read mixt
  sct=2
  mix=1  ncm=2
      1001 5.77931e-02    7014 2.13092e-03    8016 3.74114e-02
     92234 1.06784e-05   92235 9.84602e-04   92236 5.29386e-06
     92238 6.19414e-05
  mix=2  ncm=11
      1001 5.67873e-02    6000 3.54921e-02    8016 1.41968e-02
  mix=3  ncm=10
      6000 3.84193e-02 9001001 7.99120e-02
  mix=4  ncm=15
      8016 3.33757e-11    1001 6.67515e-11
end mixt
read bounds
  -zb=h2o
end bounds
read geom
  unit 1
    cylinder 10  9.52 8.7804 -8.7804
    cylinder 20  9.52 8.9896 -8.7804
    cylinder 30  10.16 9.6296 -9.4204
    cuboid   40  18.45 -18.45 18.45 -18.45 17.8946 -17.6854
    media 1 1 10
    media 0 1 -10 20
    media 2 1 -10 -20 30
    media 0 1 40 -20 -30
    boundary  40
  unit 2
    cuboid 10  18.45 -55.35  55.35 -18.45  53.37 -17.79
    cuboid 20  18.45 -55.35  55.35 -18.45 -17.79 -53.37
    cuboid 30  55.35  18.45  55.35 -18.45  53.37 -53.37
    cuboid 40  55.35 -55.35 -18.45 -55.35  53.37 -53.37
    cuboid 50  55.35 -55.35  55.35 -55.35  53.37 -53.37
    array  1 10 place 1 1 1 -36.90 0.0 -0.1046
    array  2 20 -10 place 1 1 1 -36.90 0.0 -35.6846
    array  3 30 -20 -10 place 1 1 1 36.90 0.0 -35.6846
    array  4 40 -30 -20 -10 place 1 1 1 -36.90 -36.90 -35.6846
    media 0 1 50 -40 -30 -20 -10
    boundary  50
  global unit 3
    cuboid 10  55.35 -55.35 55.35 -55.35 53.37 -53.37
    cuboid 20  58.35 -58.35 58.35 -58.35 56.37 -56.37
    cuboid 30  61.35 -61.35 61.35 -61.35 59.37 -59.37
    cuboid 40  64.35 -64.35 64.35 -64.35 62.37 -62.37
    cuboid 50  67.35 -67.35 67.35 -67.35 65.37 -65.37
    cuboid 60  70.59 -70.59 70.59 -70.59 68.61 -68.61
    array 5 10 place 1 1 1 3*0.0
    media 3 2 -10 20
    media 3 3 -20 30
    media 3 4 -30 40
    media 3 5 -40 50
    media 3 6 60 -50
    boundary  60
 end geom
read bias
  id=400 2 6
end bias
read volume
  type=random
end volume
read array
  ara=1 nux=2 nuy=2 nuz=2 fill f1 end fill
```

(continues on next page)

(continued from previous page)

```
  ara=2 nux=2 nuy=2 nuz=1 fill f1 end fill
  ara=3 nux=1 nuy=2 nuz=3 fill f1 end fill
  ara=4 nux=3 nuy=1 nuz=3 fill f1 end fill
  gbl=5 ara=5 nux=1 nuy=1 nuz=1 fill f2 end fill
end array
read plot
  scr=yes lpi=10
  ttl='  1f27 xy plot at z=0.0 '
  xul=-71.0 yul=71.0 zul=0.0 xlr=71.0 ylr=-71.0 zlr=0.0
  uax=1 vdn=-1 nax=400 end plt0
  ttl='unit map 1f27 xy plot at z=0.0'
  pic=unit
end plot
end data
end
```

Sample Problem 19 4 AQUEOUS 4 METAL ARRAY OF ARRAYS (SAMP PROB 12)

This critical experiment was described previously as SAMPLE PROBLEM 12. The input data given below utilize the array of arrays option. See Fig. 8.1.235.

Input Data

KENO V.a

```
=kenova
sample problem 19 4 aqueous 4 metal array of arrays (samp prob 12)
read param
  lib=4  flx=yes fdn=yes nub=yes smu=yes mkp=yes
  mku=yes fmp=yes fmu=yes
  htm=no
end param
read mixt
  mix=1  ncm=1
    92234 4.82717e-04   92235 4.47971e-02   92236 9.57233e-05   92238 2.65767e-03
  mix=2  ncm=2
     1001 5.77931e-02    7014 2.13092e-03    8016 3.74114e-02
    92234 1.06784e-05   92235 9.84602e-04   92236 5.29386e-06
    92238 6.19414e-05
  mix=3  ncm=11
     1001 5.67873e-02    6000 3.54921e-02    8016 1.41968e-02
  sct=2
end mixt
read geom
  unit 1
    com='uranyl nitrate solution in a plexiglas container'
    cylinder  2 1 9.525 2p8.89
    cylinder  3 1 10.16 2p9.525
    cuboid  0 1 4p10.875 2p10.24
  unit 2
    com='uranium metal cylinder'
    cylinder  1 1 5.748 2p5.3825
    cuboid  0 1 4p6.59 2p6.225
  unit 3
    com='1x2x2 array of solution units'
    array 1 3*0.0
  unit 4
    com='1x2x2 array of metal units padded to match solution array'
    array 2 3*0.0
    replicate 0 1 2*0.0 2*8.57 2*8.03 1
end geom
read array
  ara=1 nux=1 nuy=2 nuz=2 fill f1 end fill
```

(continues on next page)

(continued from previous page)

```
  ara=2 nux=1 nuy=2 nuz=2 fill f2 end fill
  gbl=3 ara=3 nux=2 nuy=1 nuz=1
    com='composite array of solution and metal units'
    fill 4 3 end fill
end array
end data
end
```

KENO-VI

```
=kenovi
sample problem 19 4 aqueous 4 metal array of arrays (samp prob 12)
read param
  lib=4 flx=yes fdn=yes nub=yes smu=yes mkp=yes mku=yes fmp=yes fmu=yes
  htm=no
end param
read mixt
  mix=1  ncm=1
    92234 4.82717e-04   92235 4.47971e-02   92236 9.57233e-05   92238 2.65767e-03
  mix=2  ncm=2
     1001 5.77931e-02    7014 2.13092e-03    8016 3.74114e-02
    92234 1.06784e-05   92235 9.84602e-04   92236 5.29386e-06
    92238 6.19414e-05
  mix=3  ncm=11
     1001 5.67873e-02    6000 3.54921e-02    8016 1.41968e-02
  sct=2
end mixt
read geometry
  unit 1
    com='uranyl nitrate solution in a plexiglas container'
    cylinder 10   9.525 2p8.89
    cylinder 20   10.16 2p9.525
    cuboid   30  4p10.875 2p10.24
    media 2 1 10      vol=20270.83270
    media 3 1 -10 20  vol=4440.27764
    media 0 1 30 -20  vol=14042.16966
    boundary  30
  unit 2
    com='uranium metal cylinder'
    cylinder 10  5.748 2p5.3825
    cuboid   20 4p6.59 2p6.225
    media 1 1 10      vol=4469.48431
    media 0 1 20 -10  vol=4181.39321
    boundary  20
  unit 3
    com='1x2x2 array of solution units'
    cuboid 10  21.75 0.0 43.5 0.0 40.96 0.0
    array 1 +10 place 1 1 1 10.875 10.875 10.240
    boundary  10
  unit 4
    com='1x2x2 array of metal units padded to match solution array'
    cuboid 10  13.18 0.0 26.36 0.0 24.9 0.0
    cuboid 20  13.18 0.0 34.93 -8.57 32.93 -8.03
    array 2 +10 place 1 1 1 6.59 6.59 6.225
    media 0 1 20 -10  vol=14830.750188
    boundary  20
  global unit 5
    com='global unit of arrays 1 and 2'
    cuboid 10 34.93 0.0 43.5 0.0 40.96 0.0
    array 3 +10 place 1 1 1 0 8.57 8.03
    boundary  10
end geom
read array
  ara=1 nux=1 nuy=2 nuz=2 fill f1 end fill
  ara=2 nux=1 nuy=2 nuz=2 fill f2 end fill
```

(continues on next page)

(continued from previous page)

```
  gbl=3 ara=3 nux=2 nuy=1 nuz=1
  com='composite array of solution and metal units'
  fill 4 3 end fill
end array
end data
end
```

Sample Problem 20 TRIANGULAR PITCHED ARRAY

This problem is a critical experiment14 consisting of seven cylinders in a triangular-pitched unreflected array. The central cylinder has six cylinders arranged around it. The surface-to-surface separation between the units is 0.15 in. Each unit consists of a 60-mil-thick aluminum can with an 8-in. inside diameter, filled with a solution of 93.2% enriched uranyl fluoride with a H/235U atomic ratio of 44.3 and a density of 576.87 g U/L. The apparatus for conducting this experiment is shown in Fig. 8.1.241.

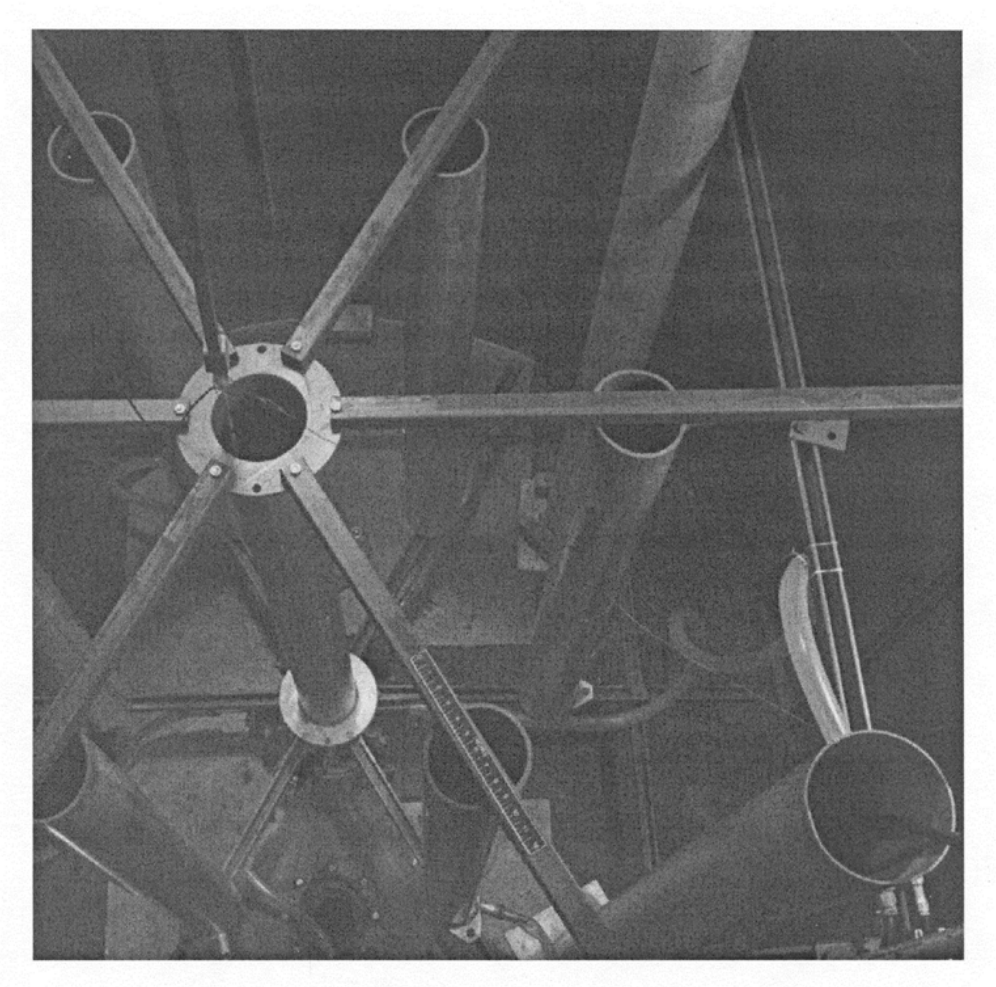


Fig. 8.1.241: Typical arrangement for critical experiments with interacting arrays of aluminum cylinders containing enriched ^{235}U solutions.

Input Data

KENO V.a

```
=kenova
sample problem 20 triangular pitched array
read param
  lib=4
  htm=no
end param
read mixt
  sct=2
  mix=1  ncm=8
    92235 1.37751e-03  92238 9.92357e-05   8016 3.33717e-02  9019 2.95350e-03   1001 6.08364e-02
  mix=2  ncm=14
    13027 6.03067e-02
end mixt
read geom
  unit 1
    cylinder   1 1 10.16 18.288 0
    cylinder   2 1 10.312 18.288 -.152
  unit 2
    cuboid     0 1 4p50 50 -.152
    hole       1 3r0
    hole       1 21.006 2r0
    hole       1 -21.006 2r0
    hole       1 10.503 18.192 0
    hole       1 -10.503 18.192 0
    hole       1 10.503 -18.192 0
    hole       1 -10.503 -18.192 0
end geom
read array
  gbl=1 nux=1 nuy=1 nuz=1 fill 2 end fill
end array
read plot
  ttl='hex array' pic=mix lpi=10 scr=yes
  xul=0   yul=100 zul=10
  xlr=100 ylr=0   zlr=10
  uax=1   vdn=-1  nax=400
end plot
end data
end
```

KENO-VI

```
=kenovi
sample problem 20  triangular pitched array 7 pins in a circle
read parameters
  lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=8
    92235 1.37751e-03  92238 9.92357e-05   8016 3.33717e-02  9019 2.95350e-03   1001 6.08364e-02
  mix=2  ncm=14
    13027 6.03067e-02
end mixt
read geometry
  unit 1
    com='single cell fuel can in hexprism'
    cylinder 10  10.16  18.288  0.0
    cylinder 20  10.312 18.288 -0.152
    hexprism 30  10.503 18.288 -0.152
    media  1 1 10      vol=41514.66537
    media  2 1 20 -10  vol=1606.91193
    media 0 1 30 -20   vol=6204.469507
    boundary  30
  unit 2
    com='empty cell'
```

(continues on next page)

(continued from previous page)

```
    hexprism 10  10.503 18.288 -0.152
    media  0 1 10  vol=8155.956715
    boundary  10
  global unit 3
    cylinder 10  31.500 18.288 -0.152
    com='7 cylinders in a circle with cylindrical boundary'
    array  1 10 place 3 3 1 3*0.0
    boundary  10
end geometry
read array
  ara=1 typ=triangular nux=5 nuy=5 nuz=1
  fill 7*2 2*1 2*2 3*1 2*2 2*1 7*2 end fill
end array
end data
end
```

Sample Problem 21 PARTIALLY FILLED SPHERE

This critical experiment consisted of a partially filled, unreflected spherical container. This aluminum container had an inside diameter of 27.244 in. and a wall thickness of 1/16 in. It is referred to in the report as the 27.3-in.-diameter vessel. The sphere was 98% filled with uranyl fluoride at an enrichment of 4.89% with an $H/^{235}U$ atomic ratio of 1099. The height of the solution in the sphere was 64.6 cm above the bottom of the sphere. A schematic diagram of the apparatus used in the experiment is given in Fig. 8.1.242. The steel tank was ignored.

Input Data

KENO V.a

```
=kenova
sample problem 21  partially filled sphere
read param
  lib=4
  htm=no
end param
read geom
  global unit 1
    hemisphe-z  1 1 34.6   chord 30.
    sphere      0 1 34.6
    sphere      2 1 34.759
end geom
read mixt
  sct=2
  mix=1  ncm=9
     1001 6.19770e-02   8016 3.34895e-02   9019 2.50098e-03
    92234 2.54224e-07  92235 6.18924e-05  92238 1.18835e-03
  mix=2  ncm=14
    13027 6.03067e-02
end mixt
end data
end
```

KENO-VI

```
=kenovi
sample problem 21  partially filled sphere
read param
  lib=4
  htm=no
end param
read mixt
```

(continues on next page)

(continued from previous page)

```
  sct=2
  mix=1  ncm=9
     1001 6.19770e-02   8016 3.34895e-02   9019 2.50098e-03
    92234 2.54224e-07  92235 6.18924e-05  92238 1.18835e-03
  mix=2  ncm=14
    13027 6.03067e-02
end mixt
read geom
  global unit 1
    sphere 10 34.6  chord -z=30.0
    sphere 20 34.6
    sphere 30 34.759
    media 1 1 10  vol=171309.
    media 0 1 20 -10  vol=2198.14
    media 2 1 30 -20 -10  vol=2403.00
    boundary  30
end geom
end data
end
```

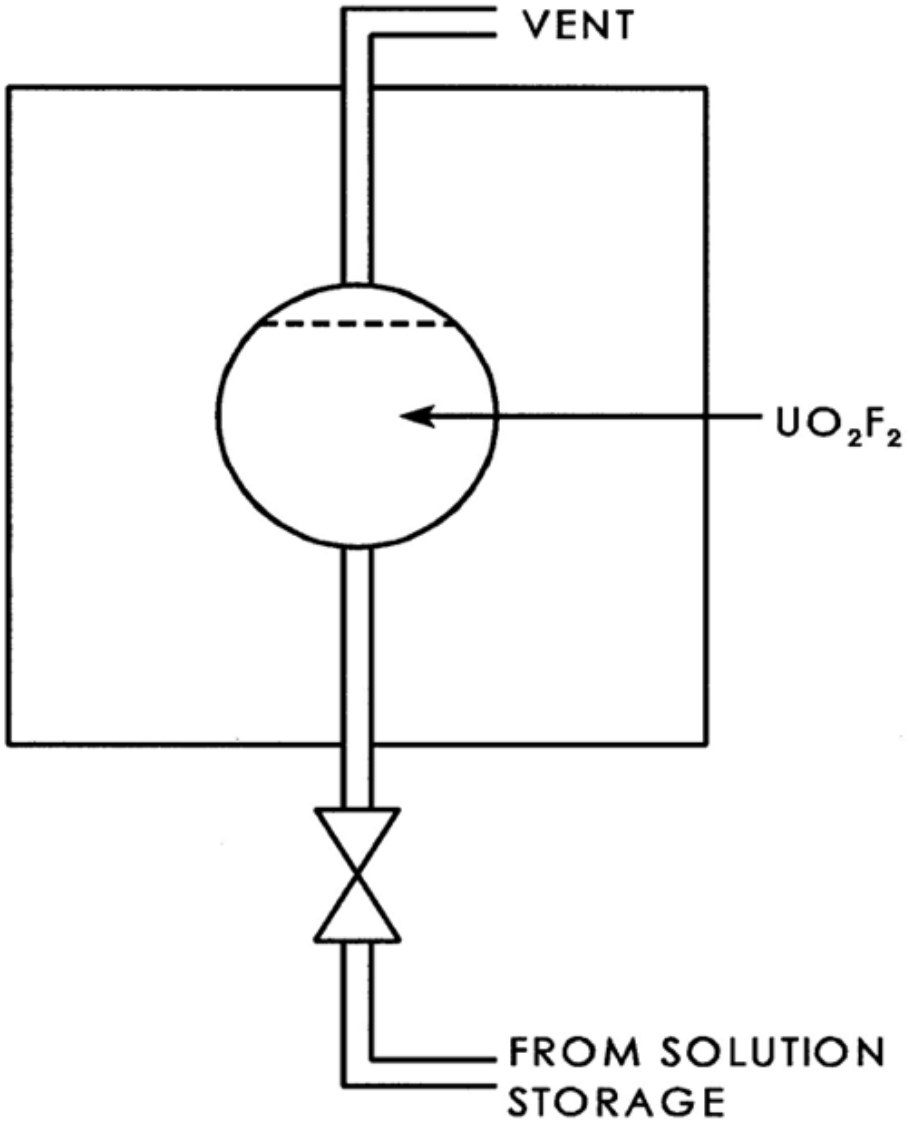


Fig. 8.1.242: Schematic of bare partially filled sphere experiment inside a 9.5-ft-diameter, 9-ft-high steel tank.

Sample Problem 22 CASE 2C8 BARE WITH 3 NESTED HOLES, EACH IS EQUAL VOLUME

The physical representation of this sample problem is the critical experiment described in sample problem 1. It is a simple $2 \times 2 \times 2$ array of 93.2% wt enriched uranium metal cylinders. This sample problem defines a uranium cylinder in a void spacing cuboid using nested holes. Eight of these units are stacked together in a $2 \times 2 \times 2$ array.

Input Data

KENO V.a

```
=kenova
sample problem 22   case 2c8 bare with 3 nested, equal volume holes
read parameters
  flx=yes fdn=yes far=yes gas=no lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry

  unit 1
    cylinder 1 1 3.621 2p3.3907

  unit 2
    cylinder 1 1 4.5622 2p4.2721
    hole 1 3*0.0

  unit 3
    cylinder 1 1 5.2224 2p4.8903
    hole 2 3*0.0

  unit 4
    cylinder 1 1 5.748 5.3825 -5.3825
    hole 3 3*0.0
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505

end geometry
read array
  nux=2 nuy=2 nuz=2 fill f4 end fill
end array
end data
end
```

KENO-VI

```
=kenovi
sample problem 22  case 2c8 bare with 3 nested, equal volume holes
read parameters
  flx=yes fdn=yes far=yes gas=no lib=4 mkh=yes ckh=yes fmh=yes
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    cylinder 10 3.621 2p3.3907
    media 1 1 10  vol=279.335597542
    boundary  10
  unit 2
    cylinder 20 4.5622 2p4.2721
    hole 1
    media 1 1 20  vol=279.353142545
    boundary 20
  unit 3
    cylinder 20 5.2224 2p4.8903
    hole 2
    media 1 1 20  vol=279.333676489
    boundary  20
  unit 4
    cylinder 20  5.748 2p5.3825
```

(continues on next page)

(continued from previous page)

```
    cuboid   30 6.87 -6.87 6.87 -6.87 6.505 -6.505
    hole 3
    media 1 1 20  vol=279.34866089
    media 0 1 30 -20  vol=1338.755598534
    boundary  30
  global unit 5
    cuboid 10 20.61 -6.87 20.61 -6.87 19.515 -6.505
    array 1 10 place 1 1 1 3*0.0
    boundary  10
end geometry
read array
  ara=1 nux=2 nuy=2 nuz=2 fill f4 end fill
end array
end data
end
```

Sample Problem 23 CASE 2C8 BARE AS STACKED CYLINDERS

The physical representation of this sample problem is the critical experiment described in sample problem 1. This sample problem describes each of the eight units in the critical $2 \times 2 \times 2$ array using Z hemicylinders (in KENO V.a) or hemicylinders with different chord sizes and directions (in KENO-VI).

Input Data

KENO V.a

```
=kenova
sample problem 23  case 2c8 bare as mixed zhemicylinders
read parameters
  npg=1000 fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='-x half of unit 3'
    zhemicyl-x 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 0.0  -6.87 6.87 -6.87 6.505 -6.505
  unit 2
    com='+x half of unit 3'
    zhemicyl+x 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87  0.0  6.87 -6.87 6.505 -6.505
  unit 3
    com='cylinder composed of equal halves (zhemicylinders with x radii)'
    array 1 3*0.0
  unit 4
    com='-x portion (more than half) of unit 6'
    zhemicyl-x 1 1 5.748 5.3825 -5.3825 chord 3.0
    cuboid  0 1 3.0  -6.87 6.87 -6.87 6.505 -6.505
  unit 5
    com='+x portion (less than half) of unit 6'
    zhemicyl+x 1 1 5.748 5.3825 -5.3825 chord -3.0
    cuboid  0 1 6.87  3.0  6.87 -6.87 6.505 -6.505
  unit 6
    com='cylinder composed of unequal halves (zhemicylinders with x radii)'
    array 2 3*0.0
  unit 7
    com='cylinder of a single zhemicylinder in the -x direction'
    zhemicyl-x 1 1 5.748 5.3825 -5.3825 chord 5.748
```

(continues on next page)

(continued from previous page)

```
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit 8
    com='cylinder of a single zhemicylinder in the +x direction'
    zhemicyl+x 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit 9
    com='-y half of unit 11'
    zhemicyl-y 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 0.0  -6.87 6.505 -6.505
  unit 10
    com='+y half of unit 11'
    zhemicyl+y 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.87  0.0  6.505 -6.505
  unit 11
    com='cylinder composed of equal halves (zhemicylinders with z radii)'
    array 3 3*0.0
  unit 12
    com='-y portion (more than half) of unit 14'
    zhemicyl-y 1 1 5.748 5.3825 -5.3825 chord 3.0
    cuboid  0 1 6.87 -6.87 3.0  -6.87 6.505 -6.505
  unit 13
    com='+y portion (less than half) of unit 14'
    zhemicyl+y 1 1 5.748 5.3825 -5.3825 chord -3.0
    cuboid  0 1 6.87 -6.87 6.87  3.0  6.505 -6.505
  unit 14
    com='cylinder composed of unequal halves (zhemicylinders with z radii)'
    array 4 3*0.0
  unit 15
    com='cylinder of a single zhemicylinder in the -y direction'
    zhemicyl-y 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
  unit 16
    com='cylinder of a single zhemicylinder in the +y'
    zhemicyl+y 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
end geometry
read array
  com='array 1 defines unit 3 (zhemicylinders with x radii)'
  ara=1 nux=2 nuy=1 nuz=1 fill 1 2 end fill
  com='array 2 defines unit 6 (zhemicylinders with x radii)'
  ara=2 nux=2 nuy=1 nuz=1 fill 4 5 end fill
  com='array 3 defines unit 11 (zhemicylinders with y radii)'
  ara=3 nux=1 nuy=2 nuz=1 fill 9 10 end fill
  com='array 4 defines unit 14 (zhemicylinders with y radii)'
  ara=4 nux=1 nuy=2 nuz=1 fill 12 13 end fill
  com='array 5 defines the total 2c8 problem'
  gbl=5 ara=5 nux=2 nuy=2 nuz=2 fill 3 7 6 8 11 15 14 16 end fill
end array
end data
end
```

KENO-VI

```
=kenovi
sample problem 23  case 2c8 bare as mixed unrotated zcylinders
read parameters
  npg=1000 fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
```

(continues on next page)

(continued from previous page)

```
unit 1
  com='-x half of unit 3'
  cylinder 10 5.748 5.3825 -5.3825 chord -x=0.0
  cuboid   20 0.0 -6.87 6.87 -6.87 6.505 -6.505
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 2
  com='+x half of unit 3'
  cylinder 10 5.748 5.3825 -5.3825 chord +x=0.0
  cuboid   20 6.87 0.0 6.87 -6.87 6.505 -6.505
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 3
  com='cylinder composed of equal halves (zhemicylinders with x radii)'
  cuboid 10 6.87 -6.87 6.87 -6.87 6.505 -6.505
  array 1 10 place 1 1 1 0.0 0.0 0.0
  boundary  10
unit 4
  com='-x portion (more than half) of unit 6'
  cylinder 10 5.748 5.3825 -5.3825 chord -x=3.0
  cuboid   20 3.0 -6.87 6.87 -6.87 6.505 -6.505
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 5
  com='+x portion (less than half) of unit 6'
  cylinder 10 5.748 5.3825 -5.3825 chord +x=3.0
  cuboid   20 6.87 3.0 6.87 -6.87 6.505 -6.505
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 6
  com='cylinder composed of unequal halves (zhemicylinders with x radii)'
  cuboid 10 6.87 -6.87 6.87 -6.87 6.505 -6.505
  array 2 10 place 1 1 1 3*0.0
  boundary  10
unit 7
  com='cylinder of a single zhemicylinder in the -x direction'
  cylinder 10 5.748 5.3825 -5.3825 chord -x=5.748
  cuboid   20 6.87 -6.87 6.87 -6.87 6.505 -6.505
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 8
  com='cylinder of a single zhemicylinder in the +x direction'
  cylinder 10 5.748 5.3825 -5.3825 chord +x=-5.748
  cuboid   20 6.87 -6.87 6.87 -6.87 6.505 -6.505
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 9
  com='-y half of unit 11'
  cylinder 10 5.748 5.3825 -5.3825 chord -y=0.0
  cuboid   20 6.87 -6.87 0.0 -6.87 6.505 -6.505
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 10
  com='+y half of unit 11'
  cylinder 10 5.748 5.3825 -5.3825 chord +y=0.0
  cuboid   20 6.87 -6.87 6.87 0.0 6.505 -6.505
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
```

(continues on next page)

(continued from previous page)

```
    boundary  20
  unit 11
    com='cylinder composed of equal halves (zhemicylinders with y radii)'
    cuboid 10 6.87 -6.87 6.87 -6.87 6.505 -6.505
    array 3 10 place 1 1 1 0.0 0.0 0.0
    boundary  10
  unit 12
    com='-y portion (more than half) of unit 14'
    cylinder 10 5.748 5.3825 -5.3825 chord -y=3.0
    cuboid   20 6.87 -6.87 3.0 -6.87 6.505 -6.505
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 13
    com='+y portion (less than half) of unit 14'
    cylinder 10 5.748 5.3825 -5.3825 chord +y=3.0
    cuboid   20 6.87 -6.87 6.87 3.0 6.505 -6.505
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 14
    com='cylinder composed of unequal halves (zhemicylinders with y radii)'
    cuboid 10 6.87 -6.87 6.87 -6.87 6.505 -6.505
    array 4 10 place 1 1 1 3*0.0
    boundary  10
  unit 15
    com='cylinder of a single zhemicylinder in the -y direction'
    cylinder 10 5.748 5.3825 -5.3825 chord -y=5.748
    cuboid   20 6.87 -6.87 6.87 -6.87 6.505 -6.505
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 16
    com='cylinder of a single zhemicylinder in the +y direction'
    cylinder 10 5.748 5.3825 -5.3825 chord +y=-5.748
    cuboid   20 6.87 -6.87 6.87 -6.87 6.505 -6.505
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  global unit 17
    cuboid 10 13.74 -13.74 13.74 -13.74 13.010 -13.010
    array 5 10 place 1 1 1 -6.87 -6.87 -6.505
    boundary  10
end geometry
read array
  com='array 1 defines unit 3 (zhemicylinders with x radii)'
  ara=1 nux=2 nuy=1 nuz=1 fill 1 2 end fill
  com='array 2 defines unit 6 (zhemicylinders with x radii)'
  ara=2 nux=2 nuy=1 nuz=1 fill 4 5 end fill
  com='array 3 defines unit 11 (zhemicylinders with y radii)'
  ara=3 nux=1 nuy=2 nuz=1 fill 9 10 end fill
  com='array 4 defines unit 14 (zhemicylinders with y radii)'
  ara=4 nux=1 nuy=2 nuz=1 fill 12 13 end fill
  com='array 5 defines the total 2c8 problem'
  gbl=5 ara=5 nux=2 nuy=2 nuz=2 fill 3 7 6 8 11 15 14 16 end fill
end array
end data
end
```

Sample Problem 24 CASE 2C8 BARE AS STACKED ROTATED CYLINDERS

The physical representation of this sample problem is the critical experiment described in sample problem 1. This sample problem describes each of the eight units in the critical $2 \times 2 \times 2$ array using hemicylinders whose axes are in the x direction. In KENO V.a this is realized using xhemicylinders, while in KENO-VI the hemycylinders with different chord sizes are rotated in the X-direction.

Input Data

KENO V.a

```
=kenova
sample problem 24  case 2c8 bare as mixed xhemicylinders
read parameters
  npg=1000 fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='-y half of unit 3'
    xhemicyl-y 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.505 -6.505 0.0 -6.87 6.87 -6.87
  unit 2
    com='+y half of unit 3'
    xhemicyl+y 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.505 -6.505 6.87 0.0  6.87 -6.87
  unit 3
    com='cylinder composed of equal halves (xhemicylinders with y radii)'
    array 1 3*0.0
  unit 4
    com='-y portion (more than half) of unit 6'
    xhemicyl-y 1 1 5.748 5.3825 -5.3825 chord 3.0
    cuboid  0 1 6.505 -6.505 3.0 -6.87 6.87 -6.87
  unit 5
    com='+y portion (less than half) of unit 6'
    xhemicyl+y 1 1 5.748 5.3825 -5.3825 chord -3.0
    cuboid  0 1 6.505 -6.505 6.87 3.0  6.87 -6.87
  unit 6
    com='cylinder composed of unequal halves (xhemicylinders with y radii)'
    array 2 3*0.0
  unit 7
    com='cylinder of a single xhemicylinder in the -y direction'
    xhemicyl-y 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.505 -6.505 6.87 -6.87 6.87 -6.87
  unit 8
    com='cylinder of a single xhemicylinder in the +y direction'
    xhemicyl+y 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.505 -6.505 6.87 -6.87 6.87 -6.87
  unit 9
    com='-z half of unit 11'
    xhemicyl-z 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.505 -6.505 6.87 -6.87 0.0 -6.87
  unit 10
    com='+z half of unit 11'
    xhemicyl+z 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.505 -6.505 6.87 -6.87 6.87 0.0
  unit 11
    com='cylinder composed of equal halves (xhemicylinders with z radii)'
    array 3 3*0.0
  unit 12
```

(continues on next page)

(continued from previous page)

```
    com='-z portion (more than half) of unit 14'
    xhemicyl-z 1 1 5.748 5.3825 -5.3825 chord 3.0
    cuboid  0 1 6.505 -6.505 6.87 -6.87 3.0 -6.87
  unit 13
    com='+z portion (less than half) of unit 14'
    xhemicyl+z 1 1 5.748 5.3825 -5.3825 chord -3.0
    cuboid  0 1 6.505 -6.505 6.87 -6.87 6.87 3.0
  unit 14
    com='cylinder composed of unequal halves (xhemicylinders with z radii)'
    array 4 3*0.0
  unit 15
    com='cylinder of a single xhemicylinder in the -z direction'
    xhemicyl-z 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.505 -6.505 6.87 -6.87 6.87 -6.87
  unit 16
    com='cylinder of a single xhemicylinder in the +z direction'
    xhemicyl+z 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.505 -6.505 6.87 -6.87 6.87 -6.87
end geometry
read array
  com='array 1 defines unit 3 (xhemicylinders with y radii)'
  ara=1 nux=1 nuy=2 nuz=1 fill 1 2 end fill
  com='array 2 defines unit 6 (xhemicylinders with y radii)'
  ara=2 nux=1 nuy=2 nuz=1 fill 4 5 end fill
  com='array 3 defines unit 11 (xhemicylinders with z radii)'
  ara=3 nux=1 nuy=1 nuz=2 fill 9 10 end fill
  com='array 4 defines unit 14 (xhemicylinders with z radii)'
  ara=4 nux=1 nuy=1 nuz=2 fill 12 13 end fill
  com='array 5 defines the total 2c8 problem'
  gbl=5 ara=5 nux=2 nuy=2 nuz=2 fill 3 7 6 8 11 15 14 16 end fill
end array
end data
end
```

KENO-VI

```
=kenovi
sample problem 24  case 2c8 bare as mixed x-rotated cylinders
read parameters
  rnd=4c6a61962572 npg=1000 fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='-y half of unit 3'
    cylinder 10 5.748 5.3825 -5.3825 chord -x=0.0 rotate a1=90 a2=90
    cuboid   20 6.505 -6.505 0.0 -6.87 6.87 -6.87
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 2
    com='+y half of unit 3'
    cylinder 10 5.748 5.3825 -5.3825 chord +x=0.0 rotate a1=90 a2=90
    cuboid   20 6.505 -6.505 6.87 0.0 6.87 -6.87
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 3
    com='cylinder composed of equal halves (xhemicylinders with y radii)'
    cuboid 10 6.505 -6.505 6.87 -6.87 6.87 -6.87
```

(continues on next page)

(continued from previous page)

```
  array 1 10 place 1 1 1 0.0 0.0 0.0
  boundary  10
unit 4
  com='-y portion (more than half) of unit 6'
  cylinder 10 5.748 5.3825 -5.3825 chord -x=3.0 rotate a1=90 a2=90
  cuboid   20 6.505 -6.505 3.0 -6.87 6.87 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 5
  com='+y portion (less than half) of unit 6'
  cylinder 10 5.748 5.3825 -5.3825 chord +x=3.0 rotate a1=90 a2=90
  cuboid   20 6.505 -6.505 6.87 3.0 6.87 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 6
  com='cylinder composed of unequal halves (xhemicylinders with y radii)'
  cuboid 10 6.505 -6.505 6.87 -6.87 6.87 -6.87
  array 2 10 place 1 1 1 3*0.0
  boundary  10
unit 7
  com='cylinder of a single xhemicylinder in the -y direction'
  cylinder 10 5.748 5.3825 -5.3825 chord -x=5.748 rotate a1=90 a2=90
  cuboid   20 6.505 -6.505 6.87 -6.87 6.87 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 8
  com='cylinder of a single xhemicylinder in the +y direction'
  cylinder 10 5.748 5.3825 -5.3825 chord +x=-5.748 rotate a1=90 a2=90
  cuboid   20 6.505 -6.505 6.87 -6.87 6.87 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 9
  com='-z half of unit 11'
  cylinder 10 5.748 5.3825 -5.3825 chord -y=0.0 rotate a1=90 a2=90
  cuboid   20 6.505 -6.505 6.87 -6.87 0.0 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 10
  com='+z half of unit 11'
  cylinder 10 5.748 5.3825 -5.3825 chord +y=0.0 rotate a1=90 a2=90
  cuboid   20 6.505 -6.505 6.87 -6.87 6.87 0.0
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 11
  com='cylinder composed of equal halves (xhemicylinders with z radii)'
  cuboid 10 6.505 -6.505 6.87 -6.87 6.87 -6.87
  array 3 10 place 1 1 1 0.0 0.0 0.0
  boundary  10
unit 12
  com='-z portion (more than half) of unit 14'
  cylinder 10 5.748 5.3825 -5.3825 chord -y=3.0 rotate a1=90 a2=90
  cuboid   20 6.505 -6.505 6.87 -6.87 3.0 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 13
  com='+z portion (less than half) of unit 14'
  cylinder 10 5.748 5.3825 -5.3825 chord +y=3.0 rotate a1=90 a2=90
  cuboid   20 6.505 -6.505 6.87 -6.87 6.87 3.0
```

(continues on next page)

(continued from previous page)

```
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 14
    com='cylinder composed of unequal halves (xhemicylinders with z radii)'
    cuboid 10 6.505 -6.505 6.87 -6.87 6.87 -6.87
    array 4 10 place 1 1 1 3*0.0
    boundary  10
  unit 15
    com='cylinder of a single xhemicylinder in the -z direction'
    cylinder 10 5.748 5.3825 -5.3825 chord -y=5.748 rotate a1=90 a2=90
    cuboid   20 6.505 -6.505 6.87 -6.87 6.87 -6.87
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 16
    com='cylinder of a single xhemicylinder in the +z direction'
    cylinder 10 5.748 5.3825 -5.3825 chord +y=-5.748 rotate a1=90 a2=90
    cuboid   20 6.505 -6.505 6.87 -6.87 6.87 -6.87
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  global unit 17
    cuboid 10 13.01 -13.01 13.74 -13.74 13.74 -13.74
    array 5 10 place 1 1 1 -6.505 -6.87 -6.87
    boundary  10
end geometry
read array
  com='array 1 defines unit 3 (xhemicylinders with y radii)'
  ara=1 nux=1 nuy=2 nuz=1 fill 1 2 end fill
  com='array 2 defines unit 6 (xhemicylinders with y radii)'
  ara=2 nux=1 nuy=2 nuz=1 fill 4 5 end fill
  com='array 3 defines unit 11 (xhemicylinders with z radii)'
  ara=3 nux=1 nuy=1 nuz=2 fill 9 10 end fill
  com='array 4 defines unit 14 (xhemicylinders with z radii)'
  ara=4 nux=1 nuy=1 nuz=2 fill 12 13 end fill
  com='array 5 defines the total 2c8 problem'
  ara=5 nux=2 nuy=2 nuz=2 fill 3 7 6 8 11 15 14 16 end fill
end array
end data
end
```

Sample Problem 25 CASE 2C8 BARE AS MIXED YHEMICYLINDERS

The physical representation of this sample problem is the critical experiment described in sample problem 1. This sample problem describes each of the eight units in the critical $2 \times 2 \times 2$ array using hemicylinders whose axes are in the y direction. This is realized in KENO V.a by using yhemicylinders, while in KENO-VI it is realized using hemicylinders with different chord sizes and directions whose long axes are rotated in the Y-direction.

Input Data

KENO V.a

```
=kenova
sample problem 25  case 2c8 bare as mixed yhemicylinders
read parameters
   npg=1000 fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
```

(continues on next page)

(continued from previous page)

```
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='-x half of unit 3'
    yhemicyl-x 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 0.0  -6.87 6.505 -6.505 6.87 -6.87
  unit 2
    com='+x half of unit 3'
    yhemicyl+x 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87  0.0  6.505 -6.505 6.87 -6.87
  unit 3
    com='cylinder composed of equal halves (yhemicylinders with x radii)'
    array 1 3*0.0
  unit 4
    com='-x portion (more than half) of unit 6'
    yhemicyl-x 1 1 5.748 5.3825 -5.3825 chord 3.0
    cuboid  0 1 3.0  -6.87 6.505 -6.505 6.87 -6.87
  unit 5
    com='+x portion (less than half) of unit 6'
    yhemicyl+x 1 1 5.748 5.3825 -5.3825 chord -3.0
    cuboid  0 1 6.87  3.0  6.505 -6.505 6.87 -6.87
  unit 6
    com='cylinder composed of unequal halves (yhemicylinders with x radii)'
    array 2 3*0.0
  unit 7
    com='cylinder of a single yhemicylinder in the -x direction'
    yhemicyl-x 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.87 -6.87 6.505 -6.505 6.87 -6.87
  unit 8
    com='cylinder of a single yhemicylinder in the +x direction'
    yhemicyl+x 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.87 -6.87 6.505 -6.505 6.87 -6.87
  unit 9
    com='-z half of unit 11'
    yhemicyl-z 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.505 -6.505 0.0 -6.87
  unit 10
    com='+z half of unit 11'
    yhemicyl+z 1 1 5.748 5.3825 -5.3825
    cuboid  0 1 6.87 -6.87 6.505 -6.505 6.87 0.0
  unit 11
    com='cylinder composed of equal halves (yhemicylinders with z radii)'
    array 3 3*0.0
  unit 12
    com='-z portion (more than half) of unit 14'
    yhemicyl-z 1 1 5.748 5.3825 -5.3825 chord 3.0
    cuboid  0 1 6.87 -6.87 6.505 -6.505 3.0 -6.87
  unit 13
    com='+z portion (less than half) of unit 14'
    yhemicyl+z 1 1 5.748 5.3825 -5.3825 chord -3.0
    cuboid  0 1 6.87 -6.87 6.505 -6.505 6.87 3.0
  unit 14
    com='cylinder composed of unequal halves (yhemicylinders with z radii)'
    array 4 3*0.0
  unit 15
    com='cylinder of a single  yhemicylinder in the -z direction'
    yhemicyl-z 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.87 -6.87 6.505 -6.505 6.87 -6.87
  unit 16
    com='cylinder of a single yhemicylinder in the +z direction'
    yhemicyl+z 1 1 5.748 5.3825 -5.3825 chord 5.748
    cuboid  0 1 6.87 -6.87 6.505 -6.505 6.87 -6.87
end geometry
```

(continues on next page)

(continued from previous page)

```
read array
  com='array 1 defines unit 3 (yhemicylinders with x radii)'
  ara=1 nux=2 nuy=1 nuz=1 fill 1 2 end fill
  com='array 2 defines unit 6 (yhemicylinders with x radii)'
  ara=2 nux=2 nuy=1 nuz=1 fill 4 5 end fill
  com='array 3 defines unit 11 (yhemicylinders with z radii)'
  ara=3 nux=1 nuy=1 nuz=2 fill 9 10 end fill
  com='array 4 defines unit 14 (zhemicylinders with z radii)'
  ara=4 nux=1 nuy=1 nuz=2 fill 12 13 end fill
  com='array 5 defines the total 2c8 problem'
  gbl=5 ara=5 nux=2 nuy=2 nuz=2 fill 3 7 6 8 11 15 14 16 end fill
end array
end data
end
```

KENO-VI

```
=kenovi
sample problem 25  case 2c8 bare as mixed y-rotated cylinders
read parameters
  npg=1000 fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='-x half of unit 3'
    cylinder 10 5.748 5.3825 -5.3825 chord -y=0.0 rotate a1=180 a2=90 a3=90
    cuboid   20 0.0 -6.87 6.505 -6.505 6.87 -6.87
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 2
    com='+x half of unit 3'
    cylinder 10 5.748 5.3825 -5.3825 chord +y=0.0 rotate a1=180 a2=90 a3=90
    cuboid   20 6.87 0.0 6.505 -6.505 6.87 -6.87
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 3
    com='cylinder composed of equal halves (yhemicylinders with x radii)'
    cuboid 10 6.87 -6.87 6.505 -6.505 6.87 -6.87
    array 1 10 place 1 1 1 0.0 0.0 0.0
    boundary  10
  unit 4
    com='-x portion (more than half) of unit 6'
    cylinder 10 5.748 5.3825 -5.3825 chord -y=3.0 rotate a1=180 a2=90 a3=90
    cuboid   20 3.0 -6.87 6.505 -6.505 6.87 -6.87
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 5
    com='+x portion (less than half) of unit 6'
    cylinder 10 5.748 5.3825 -5.3825 chord +y=3.0 rotate a1=180 a2=90 a3=90
    cuboid   20 6.87 3.0 6.505 -6.505 6.87 -6.87
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  unit 6
    com='cylinder composed of unequal halves (yhemicylinders with x radii)'
    cuboid 10 6.87 -6.87 6.505 -6.505 6.87 -6.87
```

(continues on next page)

(continued from previous page)

```
  array 2 10 place 1 1 1 3*0.0
  boundary  10
unit 7
  com='cylinder of a single yhemicylinder in the -x direction'
  cylinder 10 5.748 5.3825 -5.3825 chord -y=5.748 rotate a1=180 a2=90 a3=90
  cuboid   20 6.87 -6.87 6.505 -6.505 6.87 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 8
  com='cylinder of a single yhemicylinder in the +x direction'
  cylinder 10 5.748 5.3825 -5.3825 chord +y=-5.748 rotate a1=180 a2=90 a3=90
  cuboid   20 6.87 -6.87 6.505 -6.505 6.87 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 9
  com='-z half of unit 11'
  cylinder 10 5.748 5.3825 -5.3825 chord -x=0.0 rotate a1=180 a2=90 a3=90
  cuboid   20 6.87 -6.87 6.505 -6.505 0.0 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 10
  com='+z half of unit 11'
  cylinder 10 5.748 5.3825 -5.3825 chord +x=0.0 rotate a1=180 a2=90 a3=90
  cuboid   20 6.87 -6.87 6.505 -6.505 6.87 0.0
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 11
  com='cylinder composed of equal halves (yhemicylinders with z radii)'
  cuboid 10 6.87 -6.87 6.505 -6.505 6.87 -6.87
  array 3 10 place 1 1 1 0.0 0.0 0.0
  boundary  10
unit 12
  com='-z portion (more than half) of unit 14'
  cylinder 10 5.748 5.3825 -5.3825 chord -x=3.0 rotate a1=180 a2=90 a3=90
  cuboid   20 6.87 -6.87 6.505 -6.505 3.0 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 13
  com='+z portion (less than half) of unit 14'
  cylinder 10 5.748 5.3825 -5.3825 chord +x=3.0 rotate a1=180 a2=90 a3=90
  cuboid   20 6.87 -6.87 6.505 -6.505 6.87 3.0
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 14
  com='cylinder composed of unequal halves (yhemicylinders with z radii)'
  cuboid 10 6.87 -6.87 6.505 -6.505 6.87 -6.87
  array 4 10 place 1 1 1 3*0.0
  boundary  10
unit 15
  com='cylinder of a single yhemicylinder in the -z direction'
  cylinder 10 5.748 5.3825 -5.3825 chord -x=5.748 rotate a1=180 a2=90 a3=90
  cuboid   20 6.87 -6.87 6.505 -6.505 6.87 -6.87
  media 1 1 10     vol=2234.742156
  media 0 1 20 -10 vol=2677.511196
  boundary  20
unit 16
  com='cylinder of a single yhemicylinder in the +z direction'
  cylinder 10 5.748 5.3825 -5.3825 chord +x=-5.748 rotate a1=180 a2=90 a3=90
  cuboid   20 6.87 -6.87 6.505 -6.505 6.87 -6.87
```

(continues on next page)

(continued from previous page)

```
    media 1 1 10     vol=2234.742156
    media 0 1 20 -10 vol=2677.511196
    boundary  20
  global unit 17
    cuboid 10 13.74 -13.74 13.01 -13.01 13.74 -13.74
    array 5 10 place 1 1 1 -6.87 -6.505 -6.87
    boundary  10
end geometry
read array
  com='array 1 defines unit 3 (yhemicylinders with z radii)'
  ara=1 nux=2 nuy=1 nuz=1 fill 1 2 end fill
  com='array 2 defines unit 6 (yhemicylinders with z radii)'
  ara=2 nux=2 nuy=1 nuz=1 fill 4 5 end fill
  com='array 3 defines unit 11 (yhemicylinders with x radii)'
  ara=3 nux=1 nuy=1 nuz=2 fill 9 10 end fill
  com='array 4 defines unit 14 (yhemicylinders with x radii)'
  ara=4 nux=1 nuy=1 nuz=2 fill 12 13 end fill
  com='array 5 defines the total 2c8 problem'
  gbl=5 ara=5 nux=2 nuy=2 nuz=2 fill 3 7 6 8 11 15 14 16 end fill
end array
read volume
  type=random
end volume
end data
end
```

Sample Problem 26 (KENO V.a ONLY) CASE 2C8 BARE AS MIXED ZHEMICYLINDERS WITH ORIGINS

The physical representation of this sample problem is the critical experiment described in sample problem 1. This sample problem describes each of the eight units in the critical $2 \times 2 \times 2$ array using zhemicylinders with origins.

KENO V.a

```
=kenova
sample problem 26  case 2c8 bare as mixed zhemicylinders with origins
read parameters
  npg=1000 fdn=yes lib=4  run=yes
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='-x half of first cylinder'
    zhemicyl-x 1 1 5.748 5.3825 -5.3825 origin 6.87 0.0
    cuboid  0 1 6.87 0.0 6.87 -6.87 6.505 -6.505
  unit 2
    com='+x half of first cylinder'
    zhemicyl+x 1 1 5.748 5.3825 -5.3825 origin 6.87 0.0
    cuboid  0 1 13.74 6.87 6.87 -6.87 6.505 -6.505
  unit 3
    com='1st cylinder composed of equal portions (z hemicylinders with x radii)'
    array 1 3*0.0
  unit 4
    com='-x portion (more than half) of second cylinder'
    zhemicyl-x 1 1 5.748 5.3825 -5.3825 chord 3.0 origin 6.87 0.0
    cuboid  0 1 9.87 0.0 6.87 -6.87 6.505 -6.505
  unit 5
```

(continues on next page)

(continued from previous page)

```
    com='+x portion (less than half) of second cylinder'
    zhemicyl+x 1 1 5.748 5.3825 -5.3825 chord -3.0 origin 6.87 0.0
    cuboid  0 1 13.74 9.87 6.87 -6.87 6.505 -6.505
  unit 6
    com='2nd cylinder composed of unequal portions (z hemicylinders with x radii)'
    array 2 3*0.0
  unit 7
    com='3rd cylinder: described as a zhemicylinder in the -x direction'
    zhemicyl-x 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 6.87 0.0
    cuboid  0 1 13.74 0.0 6.87 -6.87 6.505 -6.505
  unit 8
    com='4th cylinder: described as a zhemicylinder in the +x direction'
    zhemicyl+x 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 6.87 0.0
    cuboid  0 1 13.74 0.0 6.87 -6.87 6.505 -6.505
  unit 9
    com='-y half of fifth cylinder'
    zhemicyl-y 1 1 5.748 5.3825 -5.3825 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 6.87 0.0 6.505 -6.505
  unit 10
    com='+y half of fifth cylinder'
    zhemicyl+y 1 1 5.748 5.3825 -5.3825 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 13.74 6.87  6.505 -6.505
  unit 11
    com='5th cylinder composed of equal portions (zhemicylinders with y radii)'
    array 3 3*0.0
  unit 12
    com='-y portion (more than half) of sixth cylinder'
    zhemicyl-y 1 1 5.748 5.3825 -5.3825 chord 3.0 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 9.87  0.0 6.505 -6.505
  unit 13
    com='+y portion (less than half) of sixth cylinder'
    zhemicyl+y 1 1 5.748 5.3825 -5.3825 chord -3.0 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 13.74  9.87  6.505 -6.505
  unit 14
    com='6th cylinder composed of unequal portions (zhemicylinders with y radii)'
    array 4 3*0.0
  unit 15
    com='7th cylinder: described as a zhemicylinder in the -y direction'
    zhemicyl-y 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 13.74 0.0 6.505 -6.505
  unit 16
    com='8th cylinder: described as a zhemicylinder in the +y direction'
    zhemicyl+y 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 13.74 0.0 6.505 -6.505
  global unit 17
    com='complete 2c8 bare configuration'
    array 5 3*0.0
end geometry
read array
  com='array 1: 1st cylinder (unit 3) equal x portions of zhemicylinders'
  ara=1 nux=2 nuy=1 nuz=1 fill 1 2 end fill
  com='array 2: 2nd cylinder (unit 6) unequal x portions of zhemicylinders'
  ara=2 nux=2 nuy=1 nuz=1 fill 4 5 end fill
  com='array 3: 5th cylinder (unit 11) equal y portions of zhemicylinders'
  ara=3 nux=1 nuy=2 nuz=1 fill 9 10 end fill
  com='array 4: 6th cylinder (unit 14) unequal y portions of zhemicylinders'
  ara=4 nux=1 nuy=2 nuz=1 fill 12 13 end fill
  com='array 5 defines the total 2c8 problem'
  ara=5 nux=2 nuy=2 nuz=2 fill 3 7 6 8 11 15 14 16 end fill
end array
end data
end
```

Sample Problem 27 (KENO V.a oONLY) CASE 2C8 BARE AS MIXED XHEMICYLINDERS WITH ORIGINS

The physical representation of this sample problem is the critical experiment described in sample problem 1. This sample problem describes each of the eight units in the critical $2 \times 2 \times 2$ array using hemicylinders whose axes are in the x direction. Origins are specified for each hemicylinder.

Input Data

KENO V.a

```
=kenova
sample problem 27  case 2c8 bare as mixed xhemicylinders with origins
read parameters
  npg=1000 fdn=yes lib=4  run=yes
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='-y half of first cylinder'
    xhemicyl-y 1 1 5.748 5.3825 -5.3825 origin 6.87 0.0
    cuboid  0 1 6.505 -6.505 6.87 0.0 6.87 -6.87
  unit 2
    com='+y half of first cylinder'
    xhemicyl+y 1 1 5.748 5.3825 -5.3825 origin 6.87 0.0
    cuboid  0 1 6.505 -6.505 13.74 6.87 6.87 -6.87
  unit 3
    com='1st cylinder composed of equal portions (xhemicylinders with y radii)'
    array 1 3*0.0
  unit 4
    com='-y portion (more than half) of second cylinder'
    xhemicyl-y 1 1 5.748 5.3825 -5.3825 chord 3.0 origin 6.87 0.0
    cuboid  0 1 6.505 -6.505 9.87 0.0 6.87 -6.87
  unit 5
    com='+y portion (less than half) of second cylinder'
    xhemicyl+y 1 1 5.748 5.3825 -5.3825 chord -3.0 origin 6.87 0.0
    cuboid  0 1 6.505 -6.505 13.74 9.87  6.87 -6.87
  unit 6
    com='2nd cylinder composed of unequal portions (xhemicylinders with y radii)'
    array 2 3*0.0
  unit 7
    com='3rd cylinder: described as a xhemicylinder in the -y direction'
    xhemicyl-y 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 6.87 0.0
    cuboid  0 1 6.505 -6.505 13.74 0.0 6.87 -6.87
  unit 8
    com='4th cylinder: described as a xhemicylinder in the +y direction'
    xhemicyl+y 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 6.87 0.0
    cuboid  0 1 6.505 -6.505 13.74 0.0 6.87 -6.87
  unit 9
    com='-z half of fifth cylinder'
    xhemicyl-z 1 1 5.748 5.3825 -5.3825 origin 0.0 6.87
    cuboid  0 1 6.505 -6.505 6.87 -6.87 6.87 0.0
  unit 10
    com='+z half of fifth cylinder'
    xhemicyl+z 1 1 5.748 5.3825 -5.3825 origin 0.0 6.87
    cuboid  0 1 6.505 -6.505 6.87 -6.87 13.74 6.87
  unit 11
    com='5th cylinder composed of equal portions (xhemicylinders with z radii)'
    array 3 3*0.0
  unit 12
```

(continues on next page)

(continued from previous page)

```
    com='-z portion (more than half) of sixth cylinder'
    xhemicyl-z 1 1 5.748 5.3825 -5.3825 chord 3.0 origin 0.0 6.87
    cuboid  0 1 6.505 -6.505 6.87 -6.87 9.87 0.0
  unit 13
    com='+z portion (less than half) of sixth cylinder'
    xhemicyl+z 1 1 5.748 5.3825 -5.3825 chord -3.0 origin 0.0 6.87
    cuboid  0 1 6.505 -6.505 6.87 -6.87 13.74 9.87
  unit 14
    com='6th cylinder composed of unequal portions (xhemicylinders with z radii)'
    array 4 3*0.0
  unit 15
    com='7th cylinder: described as a xhemicylinder in the -z direction'
    xhemicyl-z 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 0.0 6.87
    cuboid  0 1 6.505 -6.505 6.87 -6.87 13.74 0.0
  unit 16
    com='8th cylinder: de3scribed as a xhemicylinder in the +z direction'
    xhemicyl+z 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 0.0 6.87
    cuboid  0 1 6.505 -6.505 6.87 -6.87 13.74 0.0
  global unit 17
    com='complete 2c8 bare configuration'
    array 5 3*0.0
end geometry
read array
  com='array 1: 1st cylinder (unit 3) equal y portions of xhemicylinders'
  ara=1 nux=1 nuy=2 nuz=1 fill 1 2 end fill
  com='array 2: 2nd cylinder (unit 6) unequal y portions of xhemicylinders'
  ara=2 nux=1 nuy=2 nuz=1 fill 4 5 end fill
  com='array 3: 5th cylinder (unit 11) equal z portions of xhemicylinders'
  ara=3 nux=1 nuy=1 nuz=2 fill 9 10 end fill
  com='array 4: 6th cylinder (unit 14) unequal z portions of xhemicylinders'
  ara=4 nux=1 nuy=1 nuz=2 fill 12 13 end fill
  com='array 5 defines the total 2c8 problem'
  gbl=5 ara=5 nux=2 nuy=2 nuz=2 fill 3 7 6 8 11 15 14 16 end fill
end array
end data
end
```

Sample Problem 28 (KENO V.a oONLY) CASE 2C8 BARE AS MIXED YHEMICYLINDERS WITH ORIGINS

The physical representation of this sample problem is the critical experiment described in sample problem 1. This sample problem describes each of the eight units in the critical $2 \times 2 \times 2$ array using hemicylinders whose axes are in the y direction. Origins are specified for each hemicylinder.

Input Data

KENO V.a

```
=kenova
sample problem 28  case 2c8 bare as mixed yhemicylinders with origins
read parameters
  npg=1000 fdn=yes lib=4  run=yes
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=1
    92234 4.82717e-04 92235 4.47971e-02 92236 9.57233e-05 92238 2.65767e-03
end mixt
read geometry
  unit 1
    com='-x half of first cylinder'
    yhemicyl-x 1 1 5.748 5.3825 -5.3825 origin 6.87 0.0
```

(continues on next page)

(continued from previous page)

```
    cuboid  0 1 6.87 0.0 6.505 -6.505 6.87 -6.87
  unit 2
    com='+x half of unit 3'
    yhemicyl+x 1 1 5.748 5.3825 -5.3825 origin 6.87 0.0
    cuboid  0 1 13.74 6.87  6.505 -6.505 6.87 -6.87
  unit 3
    com='1st cylinder composed of equal portions (yhemicylinders with x radii)'
    array 1 3*0.0
  unit 4
    com='-x portion (more than half) of second cylinder'
    yhemicyl-x 1 1 5.748 5.3825 -5.3825 chord 3.0 origin 6.87 0.0
    cuboid  0 1 9.87 0.0 6.505 -6.505 6.87 -6.87
  unit 5
    com='+x portion (less than half) of second cylinder'
    yhemicyl+x 1 1 5.748 5.3825 -5.3825 chord -3.0 origin 6.87 0.0
    cuboid  0 1 13.74 9.87 6.505 -6.505 6.87 -6.87
  unit 6
    com='2nd cylinder composed of unequal portions (yhemicylinders with x radii)'
    array 2 3*0.0
  unit 7
    com='3rd cylinder: described as a single yhemicylinder in the -x direction'
    yhemicyl-x 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 6.87 0.0
    cuboid  0 1 13.74 0.0 6.505 -6.505 6.87 -6.87
  unit 8
    com='4th cylinder: described as a single yhemicylinder in the +x direction'
    yhemicyl+x 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 6.87 0.0
    cuboid  0 1 13.74 0.0 6.505 -6.505 6.87 -6.87
  unit 9
    com='-z half of fifth cylinder'
    yhemicyl-z 1 1 5.748 5.3825 -5.3825 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 6.505 -6.505 6.87 0.0
  unit 10
    com='+z half of sixth cylinder'
    yhemicyl+z 1 1 5.748 5.3825 -5.3825 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 6.505 -6.505 13.74 6.87
  unit 11
    com='5th cylinder composed of equal portions (yhemicylinders with z radii)'
    array 3 3*0.0
  unit 12
    com='-z portion (more than half) of sixth cylinder'
    yhemicyl-z 1 1 5.748 5.3825 -5.3825 chord 3.0 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 6.505 -6.505 9.87 0.0
  unit 13
    com='+z portion (less than half) of sixth cylinder'
    yhemicyl+z 1 1 5.748 5.3825 -5.3825 chord -3.0 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 6.505 -6.505 13.74 9.87
  unit 14
    com='6th cylinder composed of unequal portions (yhemicylinders with z radii)'
    array 4 3*0.0
  unit 15
    com='7th cylinder: described as a yhemicylinder in the -z direction'
    yhemicyl-z 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 6.505 -6.505 13.74 0.0
  unit 16
    com='8th cylinder: described as a yhemicylinder in the +z direction'
    yhemicyl+z 1 1 5.748 5.3825 -5.3825 chord 5.748 origin 0.0 6.87
    cuboid  0 1 6.87 -6.87 6.505 -6.505 13.74 0.0
  global unit 17
    com='complete 2c8 bare configuration'
    array 5 3*0.0
end geometry
read array
  com='array 1: 1st cylinder (unit 3) equal x portions of yhemicylinders'
  ara=1 nux=2 nuy=1 nuz=1 fill 1 2 end fill
  com='array 2: 2nd cylinder (unit 6) unequal x portions of yhemicylinders'
```

(continues on next page)

(continued from previous page)

```
  ara=2 nux=2 nuy=1 nuz=1 fill 4 5 end fill
  com='array 3: 5th cyllinder (unit 11) equal z portions of yhemicylinders'
  ara=3 nux=1 nuy=1 nuz=2 fill 9 10 end fill
  com='array 4: 6th cylinder (unit 14) unequal z portions of yhemicylinders'
  ara=4 nux=1 nuy=1 nuz=2 fill 12 13 end fill
  com='array 5 defines the total 2c8 problem'
  gbl=5 ara=5 nux=2 nuy=2 nuz=2 fill 3 7 6 8 11 15 14 16 end fill
end array
end data
end
```

Sample Problem 29 BARE CRITICAL SPHERE 3.4420-IN. RADIUS

This problem is a critical experiment [KENO-Appendix-CMLTH93] consisting of a critical Oralloy sphere. The density of the Oralloy is 18.747 g/cc, and the isotopic enrichment (wt %) is 93.21% ^{235}U, 5.7697% ^{238}U, 0.9844% ^{234}U, and 0.0359% ^{236}U. The critical radius was 8.74268 cm. A photograph of the experiment is given in Fig. 8.1.243. The support structure was ignored in the input data.

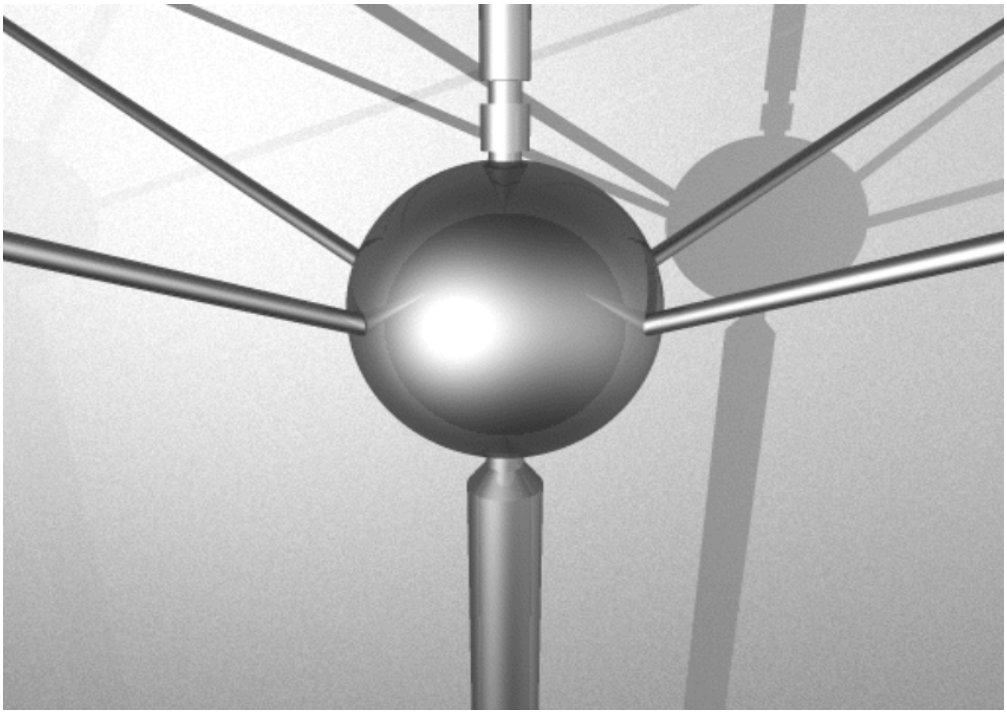


Fig. 8.1.243: Critical Oralloy sphere.

Input Data

KENO V.a

```
=kenova
sample problem 29  bare critical sphere   3.4420" radius
read parameters
    npg=1000 fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
```

(continues on next page)

(continued from previous page)

```
  mix=16  ncm=16
     92235   4.47709e-02  92238   2.73631e-03  92234   4.74858e-04  92236   1.71704e-05
end mixt
read geometry
  global unit 1
    sphere   16 1 8.74268
end geometry
read plot
  scr=yes lpi=10
  ttl='x-y slice at z=0.0'
  xul=-9 yul= 9 zul=0.0
  xlr= 9 ylr=-9 zlr=0.0
  uax=1  vdn=-1 nax=400 nch=' *'
end plot
end data
end
```

KENO-VI

```
=kenovi
sample problem 26  bare critical sphere   3.4420" radius
read parameters
  npg=1000 fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=1  ncm=16
     92235   4.47709e-02
     92238   2.73631e-03
     92234   4.74858e-04
     92236   1.71704e-05
end mixt
read geometry
  global unit 1
    sphere   10  8.74268
    media  1 1 10  vol=2799.1254126
    boundary 10
end geometry
end data
end
```

Sample Problem 30 (KENO V.a ONLY) BARE CRITICAL SPHERE Z HEMISPHERE MODEL 3.4420-IN. RADIUS

The physical representation of this sample problem is the critical experiment described in sample problem 29. This sample problem describes the sphere as two Z hemispheres, each with a chord and origin specified. One of the hemispheres is placed using the hole geometry option.

Input Data

KENO V.a

```
=kenova
sample problem 30   bare critical sphere    z hemisphere model 3.4420" radius
read parameters
    npg=1000 fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=16
```

(continues on next page)

(continued from previous page)

```
     92235   4.47709e-02  92238   2.73631e-03  92234   4.74858e-04  92236   1.71704e-05
end mixt
read geometry
   unit 1
     hemisphe+z   16 1 8.74268  chord +3.0 origin 8.9 8.9 8.9
   global unit 2
     hemisphe-z   16 1 8.74268  chord -3.0 origin 8.9 8.9 8.9
     cuboid       0 1 17.8 0.0 17.8 0.0 17.8 0.0
     hole 1 3*0.0
end geometry
read plot
  scr=yes lpi=10
  ttl='y-z slice at x=8.9   mixture map'
  xul=8.9 yul=-0.5 zul=18.5
  xlr=8.9 ylr=18.5 zlr=-0.5
  vax=1   wdn=-1   nax=400   end plt1
  ttl='y-z slice at x=8.9   unit map'
  pic=unit                   end plt2
end plot
end data
end
```

Sample Problem 31 (KENO V.a ONLY) BARE CRITICAL SPHERE X HEMISPHERE MODEL 3.4420-IN. RADIUS

The physical representation of this sample problem is the critical experiment described in sample problem 29. This sample problem describes the sphere as two X hemispheres, each with a chord and origin specified. One of the hemispheres is placed using the hole geometry option.

Input Data

KENO V.a

```
=kenova
sample problem 31   bare critical sphere    x hemisphere model 3.4420" radius
read parameters
    fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=16  ncm=16
     92235   4.47709e-02  92238   2.73631e-03  92234   4.74858e-04  92236   1.71704e-05
end mixt
read geometry
   unit 1
     hemisphe-x   16 1 8.74268  chord +3.0
   global unit 2
     hemisphe+x   16 1 8.74268  chord -3.0 origin 8.9 8.9 8.9
     cuboid       0 1 17.8 0.0 17.8 0.0 17.8 0.0
     hole 1 3*8.9
end geometry
read plot
  scr=yes lpi=10
  ttl='x-y slice at z=8.9     mixture map'
  xul=-0.5 yul=18.5 zul=8.9
  xlr=18.5 ylr=-0.5 zlr=8.9
  uax=1   vdn=-1    nax=400  end plt1
  ttl='y-z slice at x=8.9   unit map'
  pic=unit                   end plt2
end plot
end data
end
```

Sample Problem 32 (KENO V.a ONLY) BARE CRITICAL SPHERE Y HEMISPHERE MODEL 3.4420-IN. RADIUS

The physical representation of this sample problem is the critical experiment described in sample problem 29. This sample problem describes the sphere as two Y hemispheres, each with a chord and origin specified. One of the hemispheres is placed using the hole geometry option.

Input Data

KENO V.a

```
=kenova
sample problem 32   bare critical sphere    y hemisphere model 3.4420" radius
read parameters
    fdn=yes lib=4
  htm=no
end parameters
read mixt
  sct=2
  mix=16  ncm=16
     92235   4.47709e-02  92238   2.73631e-03  92234   4.74858e-04  92236   1.71704e-05
end mixt
read geometry
  unit 1
    hemisphe-y   16 1 8.74268  chord +3.0 origin 8.9 9.9 10.9
  global unit 2
    hemisphe+y   16 1 8.74268  chord -3.0 origin 8.9 8.9 8.9
    cuboid       0 1 17.8 0.0 17.8 0.0 17.8 0.0
    hole 1 0.0 -1.0 -2.0
end geometry
read plot
  scr=yes lpi=10
  ttl='x-y slice at z=8.9     mixture map'
  xul=-0.5 yul=18.5 zul=8.9
  xlr=18.5 ylr=-0.5 zlr=8.9
  uax=1    vdn=-1   nax=400  end plt1
  ttl='y-z slice at x=8.9   unit map'
  pic=unit                   end plt2
end plot
end data
end
```

Sample Problem 33 CRITICAL TRIANGULAR PITCHED ARRAY OF ANNULAR RODS

This sample problem represents a critical experiment [KENO-Appendix-CJoh66] that consists of a partially flooded array of 19 low enriched uranium metal cylindrical annuli billets arranged in a triangular pitched array. The density of the uranium metal was 19.0 g/cc, and the isotopic enrichment in weight percent was 1.95% ^{235}U, 98.02% ^{238}U, 0.006% ^{236}U, and 0.002% ^{234}U. The cylindrical annuli had an inside diameter of 6.604 cm, an outside diameter of 18.288 cm, and were placed with a pitch of 20.828 cm. Each billet was 101.6 cm long. The array was positioned in a very large tank. This configuration was critical when the tank was filled to a height of 47.7 cm on a scale whose zero point was 0.6 cm below the bottom of the billets. The bottom of the billets was 21.6 cm above the bottom of the tank. The tank and all support structures have been ignored in this model. The model utilizes only 15.24 cm of water reflector on all sides of the array. Fig. 8.1.244 and Fig. 8.1.245 provide a representation of the model. A photograph of a single annular billet is shown in Fig. 8.1.246.

Input Data

KENO V.a

```
=kenova
sample problem 33   critical triangular pitched array of annular rods
read parameters  fdn=yes nub=yes lib=4
  htm=no npg=2000
end parameters
read mixt
  sct=2
  mix=17  ncm=17
     92235   9.49270e-04  92238   4.71245e-02  92234   9.77785e-07  92236   2.90844e-06
  mix=18  ncm=18
      8016   3.33757e-02   1001   6.67515e-02
  mix=19  ncm=19
      1001   6.67515e-02   8016   3.33757e-02
  mix=20  ncm=20
     92235   9.49270e-04  92238   4.71245e-02  92234   9.77785e-07  92236   2.90844e-06
end mixt
read geom
  unit 1
    zhemicyl-x 18 1 3.302 47.7 0.6
    zhemicyl-x 17 1 9.144 47.7 0.6
  unit 2
    zhemicyl-y 18 1 3.302 47.7 0.6
    zhemicyl-y 17 1 9.144 47.7 0.6
  unit 3
    zhemicyl+x 18 1 3.302 47.7 0.6
    zhemicyl+x 17 1 9.144 47.7 0.6
  unit 4
    zhemicyl+y 18 1 3.302 47.7 0.6  origin 0.0 -18.03758
    zhemicyl+y 17 1 9.144 47.7 0.6  origin 0.0 -18.03758
    cuboid     19 1 2p10.414 2p18.03758 47.7 0.6
    hole      1       10.414    0.0      0.0
    hole      2        0.0     18.03758  0.0
    hole      3      -10.414    0.0      0.0
  unit 5
    cuboid     19 1 2p10.414 10.414 0.0 47.7 0.6
  unit 6
    zhemicyl-y 18 1 3.302 47.7 0.6
    zhemicyl-y 17 1 9.144 47.7 0.6
    cuboid     19 1 2p10.414 0.0 -10.414 47.7 0.6
  unit 7
    zhemicyl-y 18 1 3.302 47.7 0.6 origin 0.0 18.03758
    zhemicyl-y 17 1 9.144 47.7 0.6 origin 0.0 18.03758
    cuboid     19 1 2p10.414 2p18.03758 47.7 0.6
    hole     3      -10.414 0.0 0.0
  unit 8
    zhemicyl+y 18 1 3.302 47.7 0.6 origin 0.0 -18.03758
    zhemicyl+y 17 1 9.144 47.7 0.6 origin 0.0 -18.03758
    cuboid     19 1 2p10.414 2p18.03758 47.7 0.6
    hole     3      -10.414 0.0 0.0
  unit 9
    zhemicyl+y 18 1 3.302 47.7 0.6
    zhemicyl+y 17 1 9.144 47.7 0.6
    cuboid     19 1 2p10.414 10.414 0.0 47.7 0.6
  unit 10
    zhemicyl+y 18 1 3.302 47.7 0.6  origin 0.0 -18.03758
    zhemicyl+y 17 1 9.144 47.7 0.6  origin 0.0 -18.03758
    cuboid     19 1 2p10.414 2p18.03758 47.7 0.6
    hole     1       10.414 0.0 0.0
  unit 11
    zhemicyl-y 18 1 3.302 47.7 0.6  origin 0.0 18.03758
    zhemicyl-y 17 1 9.144 47.7 0.6  origin 0.0 18.03758
    cuboid     19 1 2p10.414 2p18.03758 47.7 0.6
    hole     1       10.414 0.0 0.0
  unit 21
    zhemicyl-x  0 1 3.302 102.2 47.7
    zhemicyl-x 20 1 9.144 102.2 47.7
```

(continues on next page)

(continued from previous page)

```
  unit 22
    zhemicyl-y  0 1 3.302 102.2 47.7
    zhemicyl-y 20 1 9.144 102.2 47.7
  unit 23
    zhemicyl+x  0 1 3.302 102.2 47.7
    zhemicyl+x 20 1 9.144 102.2 47.7
  unit 24
    zhemicyl+y  0 1 3.302 102.2 47.7  origin 0.0 -18.03758
    zhemicyl+y 20 1 9.144 102.2 47.7  origin 0.0 -18.03758
    cuboid      0 1 2p10.414 2p18.03758 102.2 47.7
    hole      21      10.414    0.0      0.0
    hole      22       0.0     18.03758  0.0
    hole      23     -10.414    0.0      0.0
  unit 25
    cuboid      0 1 2p10.414 10.414 0.0 102.2 47.7
  unit 26
    zhemicyl-y  0 1 3.302 102.2 47.7
    zhemicyl-y 20 1 9.144 102.2 47.7
    cuboid      0 1 2p10.414 0.0 -10.414 102.2 47.7
  unit 27
    zhemicyl-y  0 1 3.302 102.2 47.7  origin 0.0 18.03758
    zhemicyl-y 20 1 9.144 102.2 47.7  origin 0.0 18.03758
    cuboid      0 1 2p10.414 2p18.03758 102.2 47.7
    hole     23      -10.414 0.0 0.0
  unit 28
    zhemicyl+y  0 1 3.302 102.2 47.7  origin 0.0 -18.03758
    zhemicyl+y 20 1 9.144 102.2 47.7  origin 0.0 -18.03758
    cuboid      0 1 2p10.414 2p18.03758 102.2 47.7
    hole     23      -10.414 0.0 0.0
  unit 29
    zhemicyl+y  0 1 3.302 102.2 47.7
    zhemicyl+y 20 1 9.144 102.2 47.7
    cuboid      0 1 2p10.414 10.414 0.0 102.2 47.7
  unit 30
    zhemicyl+y  0 1 3.302 102.2 47.7  origin 0.0 -18.03758
    zhemicyl+y 20 1 9.144 102.2 47.7  origin 0.0 -18.03758
    cuboid      0 1 2p10.414 2p18.03758 102.2 47.7
    hole     21       10.414 0.0 0.0
  unit 31
    zhemicyl-y  0 1 3.302 102.2 47.7  origin 0.0 18.03758
    zhemicyl-y 20 1 9.144 102.2 47.7  origin 0.0 18.03758
    cuboid      0 1 2p10.414 2p18.03758 102.2 47.7
    hole     21       10.414 0.0 0.0
  unit 32
    com='flooded portion of array with 15.24 cm of water in x and y'
    array 1 2*0.0 0.6
    replicate 19 1 4r15.24 0.0 0.6 1
    replicate 19 2 5r0.0 3.0 7
  unit 33
    com='unflooded upper portion of array'
    array 2 3*0.0
    replicate 0 1 4r15.24 2*0.0 1
  global
  unit 34
    array 3 -67.31 -61.72916 -21.0
end geom
read bias
  id=500 2 8
end bias
read array
   ara=1 nux=5 nuy=4 nuz=1  fill  5 3r 6  5   11 3r 4  7    10 3r 4  8      5 3r 9  5  end fill
   ara=2 nux=5 nuy=4 nuz=1  fill 25 3r26 25   31 3r24 27    30 3r24 28     25 3r29 25  end fill
   ara=3 nux=1 nuy=1 nuz=2  fill 32 33 end fill
end array
read start
```

(continues on next page)

(continued from previous page)

```
  nst=1   xsm=-52 xsp=52   ysm=-47 ysp=47   zsm=0.6 zsp=47.7
end start
read plot
  scr=yes lpi=10
  clr=17 255 0 0
      18 128 255 255
      19 0 0 255
      20 255 0 128
  end color
  ttl='x-y plot of pins at z=45.0'
    xul=-52.0 yul= 47.0 zul=45.0
    xlr= 52.0 ylr=-47.0 zlr=45.0
    uax=  1.0 vdn=-1.0  nax=400
  end plt1
  ttl='x-z plot of pins at y=0.0'
    xul=-52.0 yul=0.0 zul=102.7
    xlr= 52.0 ylr=0.0 zlr=-3.0
    uax=  1.0 wdn=-1.0 nax=400
  end plt2
  ttl='x-z plot at y=0.0'
    xul=-68.0 yul=0.0 zul=102.7
    xlr= 70.0 ylr=0.0 zlr=-25.0
    uax=  1.0 wdn=-1.0 nax=400
  end plt3
end plot
end data
end
```

KENO-VI

```
=kenovi
sample problem 27   critical triangular pitched array of annular rods
read parameters
  fdn=yes nub=yes lib=4
  htm=no
  npg=4000
end parameters
read mixt
  sct=2
  mix=1  ncm=17
     92235   9.49270e-04
     92238   4.71245e-02
     92234   9.77785e-07
     92236   2.90844e-06
  mix=2  ncm=18
      8016   3.33757e-02
      1001   6.67515e-02
  mix=3  ncm=19
      1001   6.67515e-02
      8016   3.33757e-02
  mix=4  ncm=20
     92235   9.49270e-04
     92238   4.71245e-02
     92234   9.77785e-07
     92236   2.90844e-06
  mix=5  ncm=18
      8016   3.33757e-02
      1001   6.67515e-02
end mixt
read geom
  unit 1
    cylinder 10  3.302 102.2 0.6
    cylinder 20  9.144 102.2 0.6
    plane    30  zpl=1.0 con=-47.7
    hexprism 40 10.414 102.2 0.0
```

(continues on next page)

(continued from previous page)

```
    media  2 1 10 -30
    media  1 1 20 -10 -30
    media  3 1 40 -20 -30
    media  0 1 10  30
    media  4 1 20 -10  30
    media  0 1 40 -20  30
    boundary 40
  unit 2
    plane    10  zpl=1.0 con=-47.7
    hexprism 20 10.414 102.2 0.0
    media  3 1 -10  20
    media  0 1  10  20
    boundary 20
  global unit 3
    cylinder 10  52.42 102.2 0.0
    plane    20  zpl=1.0 con=-47.7
    cylinder 30  82.9  102.2 -21.0
    array  1 10 place 4 4 1 3*0.0
    media  0 1 30  20 -10
    media  5 1 30 -20 -10
    boundary 30
end geom
read array
  ara=1 nux=7 nuy=7 nuz=1 typ=tri fill
  2 2 2 2 2 2 2
   2 2 2 1 1 1 2
    2 2 1 1 1 1 2
     2 1 1 1 1 1 2
      2 1 1 1 1 2 2
       2 1 1 1 2 2 2
        2 2 2 2 2 2 2   end fill
end array
read volume
  type=random
end volume
read plot
  scr=yes lpi=10
  clr=1 255 0 0
      2 128 255 255
      3 0 0 255
      4 255 0 128
      5 200 200 200
  end color
  ttl='x-z plot of pins at y=0.0'
    xul=-68.0 yul= 0.0 zul=102.7
    xlr= 70.0 ylr= 0.0 zlr=-25.0
    uax=  1.0 wdn=-1.0
    nax=800
  end plt0
  ttl='x-y plot of pins and water at z=45.0'
    xul=-68.0 yul= 68.0 zul=45.0
    xlr= 68.0 ylr=-68.0 zlr=45.0
    uax=  1.0 vdn= -1.0
    nax=800
  end plt1
end plot
end data
end
```

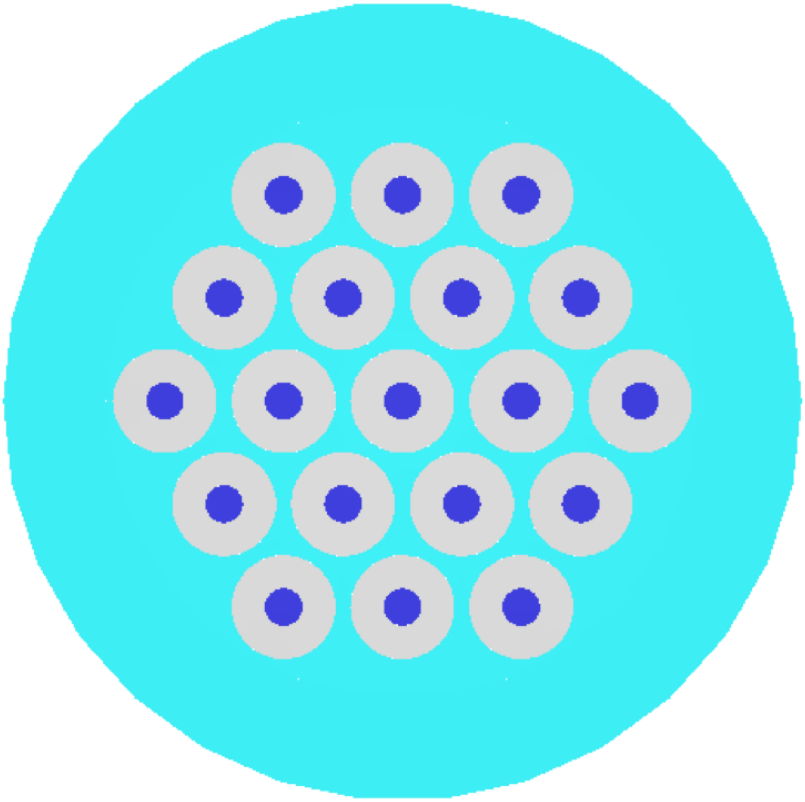


Fig. 8.1.244: Horizontal slice through a critical triangular pitched array of partially flooded 1.95% enriched uranium metal annular billets.

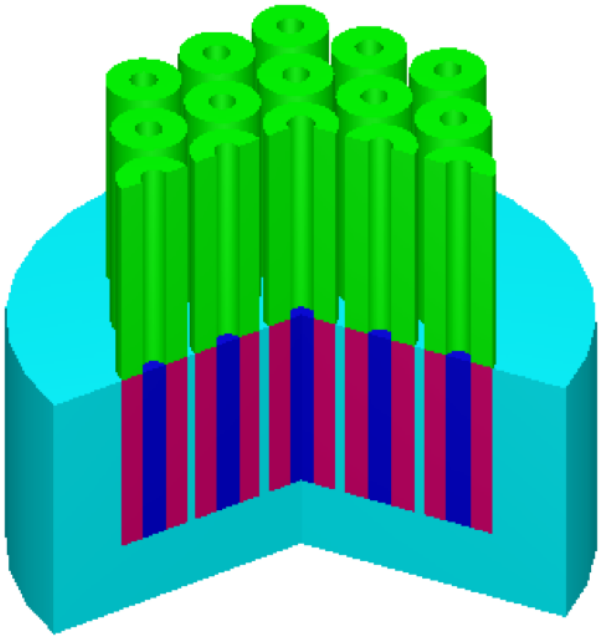


Fig. 8.1.245: Vertical slice through the center of a critical triangular-pitched array of partially flooded 1.9% enriched uranium metal annular billets.

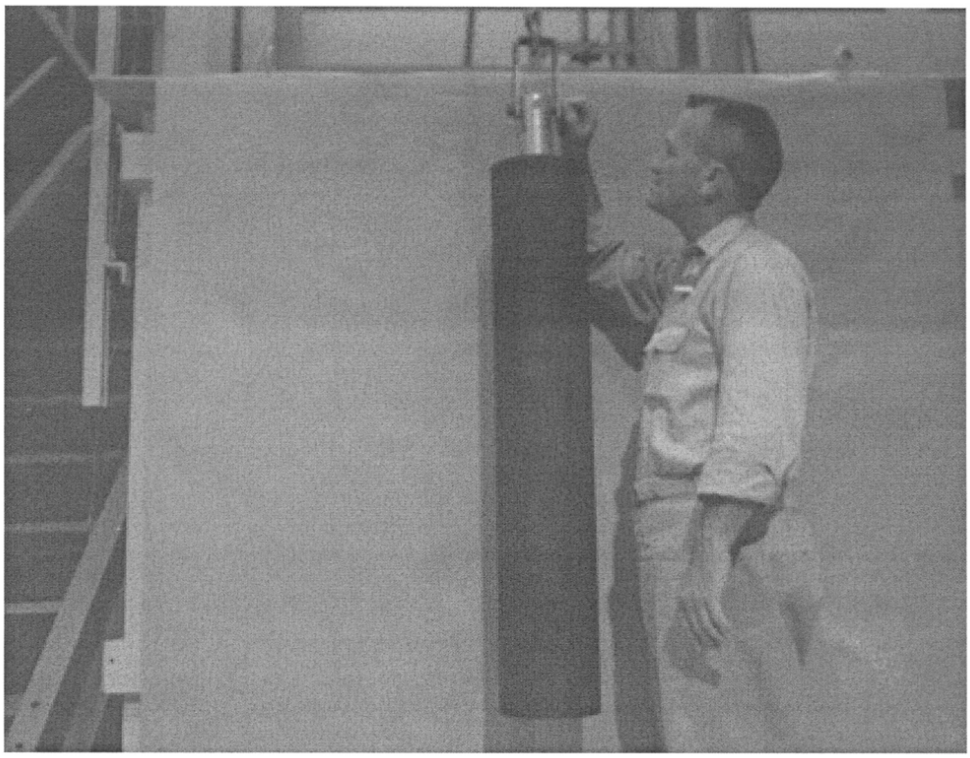


Fig. 8.1.246: 1.95% Enriched uranium metal annular billet used in critical experiments

8.2 MONACO: A FIXED-SOURCE MONTE CARLO TRANSPORT CODE FOR SHIELDING APPLICATIONS

D. E. Peplow and C. Celik

Monaco is a general-purpose, fixed-source, Monte Carlo shielding code for the SCALE package. It is a functional module that uses either AMPX cross sections or continuous energy libraries to calculate neutron and photon fluxes and responses to specific geometry regions, to point detectors and to mesh tallies. Basic multigroup transport methods are inherited from Monaco's predecessor, MORSE. Continuous energy physics has been incorporated into the code with a new physics package that uses the same CE data as CE-KENO-VI, with extensions for simulating photons. Variance reduction capabilities include source biasing and weight windows, either by geometry region or by using a mesh-based importance map. User input includes the cross section file unit number; the geometry description using the SCALE General Geometry Package; source description as a function of position, energy, and direction; tally descriptions (fluxes in which regions, at what point detectors, or over what mesh grids); and response functions (functions of energy). Output consists of tables detailing the region and point detector fluxes (and their responses), as well as files for mesh tallies.

8.2.1 INTRODUCTION

Monaco is a neutron/photon, fixed-source Monte Carlo shielding code for the SCALE code package. Monaco uses the SCALE General Geometry Package (SGGP)—the same geometry description as KENO-VI. Monaco has many options available to the user for specifying source distributions, many tally options, and many variance reduction capabilities. Monaco was originally based on the MORSE Monte Carlo code but has been extensively modified to modernize the coding, increase the number of capabilities in terms of sources and tallies, and allow for either multigroup or continuous energy (CE) transport through the use of the new SCALE CE Modular Physics Package (SCEMPP).

Monaco was developed to address a number of long-term goals for the Monte Carlo shielding capabilities in SCALE. The principal goals for this project included (1) unification of geometric descriptions between the SCALE shielding and criticality Monte Carlo codes, (2) implementation of a mesh-based importance map and mesh-based biased source distribution so that automated variance reduction could be used, and (3) establishment of a code using modern programming practices from which to continue future development. The addition of a continuous-energy transport capability is a significant change as well.

Monaco is the key component of the MAVRIC sequence, which also uses Denovo to create the mesh-based importance map and mesh-based biased source distribution for general 3-D automated variance reduction. See the MAVRIC chapter for more information.

8.2.2 MONACO CAPABILITIES

Monaco has a wide range of source descriptions and tallies for performing general radiation transport calculations. Note that Monaco can work with either the AMPX-based multigroup libraries or the newer AMPX-based CE libraries. Note that for CE calculations, tallies still employ a multigroup energy structure to store and report results.

8.2.2.1 Source Descriptions

Multiple sources can be defined for a Monaco calculation. Sampling of the different sources can be biased by the user. Each source is specified by its spatial distribution, its energy distribution, its directional distribution, and its strength. Distributions defined by the user can also be biased and can be used multiple times by different sources. The Monaco tallies assume that the sources all have units of particles/second. If the source strengths are given in other units, the user will have to incorporate the proper conversion to the tally results and remember to interpret the results accordingly.

Distributions

Two types of basic distributions are used by Monaco — binned histograms and a set of value/function pairs. The binned histogram type is defined by $n + 1$ bin boundaries and n values, representing the integrated amount in each bin. For the true distribution $f(x)$, the bin boundaries $[x_0,\ x_1,\ \ldots,\ x_n]$ and the integrated amounts $F_i = \int_{x_{i-1}}^{x_i} f(x)\,\mathrm{dx}$ are given. The distribution will be normalized by Monaco after reading. The user can optionally bias a binned histogram distribution by supplying one of the following: the biased sampling distribution amounts, $G_i = \int_{x_{i-1}}^{x_i} g(x)\,\mathrm{dx}$; the importance of each bin, I_i; or the suggested weight for each bin, w_i.

Based on what type of input is given, Monaco will compute a properly normalized probability distribution function for sampling. If the importances are given, the sampling distribution is computed as

$$G_i = \frac{I_i F_i}{\sum_i I_i F_i} \tag{8.2.1}$$

If suggested weights are given, then the sampling distribution is computed as

$$G_i = \frac{\frac{F_i}{w_i}}{\sum_i \frac{F_i}{w_i}} \tag{8.2.2}$$

for bins with non-zero weight. The sampling distribution for bins with a suggested weight of zero are set to $G_i = 0$. When sampled, particles are assigned a weight of $\frac{F_i}{G_i}$.

The second type of distribution that a user can define is for a series of point values of a function. For a set of $n + 1$ point pairs, (x_i, f_i) for $i \in [0 \ldots n]$, defining n intervals, a distribution can be made by linearly interpolating between adjacent point pairs. This type of distribution can also be biased by supplying one of the following: the biased sampling distribution function value g_i at each point, the importance of each point, I_i; or the suggested weight for each point, w_i. Similar to above, if importances or weights are given, Monaco computes the biased distribution for sampling. For the value/function point pairs type of distribution, the weight assigned to the sampled particle is a continuous function.

Some commonly used distributions are built into Monaco and can be used by simple keywords. Monaco can produce a graph of any distribution so that the user can verify that the input was entered correctly.

Spatial energy and directional attributes

Each Monaco source is described by three separable components: spatial, energy and directional.

The spatial component of a source in Monaco is simple but very flexible. First, the general shape of the source region is defined in global coordinates. The basic solid shapes and their allowed degenerate cases are listed in Table 8.2.1. The user can reference any of the defined distributions to describe the source distribution in any coordinate (x, y, and z for cuboids, r and z for cylinders and r for spheres) to use for sampling or leave the source distribution as uniform over each dimension for the solid shape. The source region can be limited by the underlying SGGP geometry variables of unit, media, and mixture. This way, source volumes (or planes, lines, or points) can be defined that are independent or dependent on the model geometry. A cylinder or cylindrical shell region can be oriented with its axis in any direction.

Table 8.2.1: Available source shapes and their allowed degenerate cases

Shape	**Allowable degenerate cases**
cuboid	rectangular plane, line, point
cylinder	circular plane, line, point
cylindrical shell	cylinder, planar annulus, circular plane, cylindrical surface, line, ring, point
sphere	point
spherical shell	sphere, spherical surface, point

Monaco samples the source position using either the given distributions or uniformly over the basic solid shape and then uses rejection if any of the optional SGGP geometry limiters have been specified. For sources that are confined to a particular unit, media, or mixture, users should make sure the basic solid shape tightly bounds the desired region for efficient sampling.

For the energy component of each source, either type of distribution described above can be used. Biasing can be used in the energy component of the source as well. The Watt spectrum is a built-in distribution which uses the Froehner and Spencer [MONACO-FS81] method for sampling. If the defined energy distribution has point(s) that are out of the problem's energy range for a CE problem, these points will be rejected in

the source energy sampling and an error message will be generated. The warnings will be suppressed if the number of rejected source points exceeds a pre-defined threshold (1000).

Distributions can be used to define the directional component of the source. A function of the cosine of the polar angle, with respect to some reference direction in global coordinates, can be used by Monaco. If no directional distribution is specified, the default is an isotropic distribution (one directional bin from $\mu = -1$ to $\mu = 1$). The default reference direction is the positive z-axis (<0,0,1>).

Monaco mesh source map files

An alternative to specifying the separate spatial and energy distributions, a Monaco mesh source file can be used. A mesh source consists of a 3D Cartesian mesh that overlay the geometry. Each mesh cell has some probability of emitting a source particle, and within each mesh cell, a different energy distribution can be sampled. Position within each mesh cell is sampled uniformly, and the emission direction is sampled from the standard directional distribution. Monaco mesh source files are typically produced by the MAVRIC sequence or by other Monaco calculations (see the mesh source saver option in the source input). For a source constructed from the separable spatial and energy distributions, Monaco can create a mesh source file which can then be visualized using the Mesh File Viewer. This is a convenient way to ensure that the source being used is what was intended.

8.2.2.2 Tallies

Monaco offers three tally types: point detectors, region tallies, and mesh tallies. Each is useful in determining quantities of interest in the simulation. Any number of each can be used, up to the limit of machine memory. The tallies will compute flux for each group, the total neutron and total photon fluxes, and any number of dose-like responses. A typical dose-like response, R, is the integral over energy of the product of a response function, $f(E)$, and the flux, $\phi(E)$.

$$R = \int f(E)\,\phi(E)\,dE \tag{8.2.3}$$

In multigroup calculations, the total response would be expressed as the sum over all groups $R = \sum f_g \phi_g$. For CE calculations, tallies can be segmented into energy and time bins which can be thought of as "groups". All three of the tally types can be scaled with a constant — for example, to account for units conversions.

Tally statistics

The three Monaco tallies are really just collections of simple and extended tallies for each group, each total, and each group contribution to a response or total response. The simple tally works in the following way: a history score h_i is zeroed out at the start of history i. During the course of the history, when an event occurs during substep j, a score consisting of some contribution c_{ij} weighted by the current particle weight w_{ij} is calculated and added to h_i. At the end of the history, the history score is the total weighted score for each substep j in the history.

$$h_i = \sum_j w_{\mathrm{ij}} c_{\mathrm{ij}} \tag{8.2.4}$$

Note that the values for the contribution c_{ij} and when it is added to the accumulator are determined by the tally type. At the end of the each history, the history score is added to two accumulators (power sums) - the first accumulator is for finding the tally average, S_1, and the second accumulator is for finding the uncertainty in the tally average, S_2.

$$S_1 = \sum_i h_i \tag{8.2.5}$$

$$S_2 = \sum_i h_i^2 \tag{8.2.6}$$

At the end of all N histories, the second sample central moment is found from the power sums

$$m_2 = \frac{S_2}{N} - \frac{S_1^2}{N^2} \tag{8.2.7}$$

and then the tally average is computed as $\overline{x} = \frac{S_1}{N}$ and the uncertainty in the tally average is $u = \sqrt{\frac{m_2}{N}}$.

The extended tally uses four accumulators — the first and second are the same as the simple tally — with the third and fourth accumulators used for finding the variance of the variance (VOV). These extra accumulators, S_3 and S_4, are calculated as

$$S_3 = \sum_i h_i^3 \tag{8.2.8}$$

$$S_4 = \sum_i h_i^4 \tag{8.2.9}$$

At the end of all N histories, the tally average $\overline{x}$ and uncertainty in the tally average u are found in the same way as a simple tally. For the VOV calculation, the third and fourth sample central moments are found as

At the end of all N histories, the tally average $\overline{x}$ and uncertainty in the tally average u are found in the same way as a simple tally. For the VOV calculation, the third and fourth sample central moments are found as

$$m_3 = \frac{S_3}{N} - \frac{3S_1S_2}{N^2} + \frac{2S_1^3}{N^3} \tag{8.2.10}$$

$$m_4 = \frac{S_4}{N} - \frac{4S_1S_3}{N^2} + \frac{6S_1^2S_2}{N^3} - \frac{3S_1^4}{N^4} \tag{8.2.11}$$

and then the VOV [MONACO-PFB97] and figure-of-merit (FOM) are found using

$$\text{VOV} = \frac{m_4 - m_2^2}{Nm_2^2} \tag{8.2.12}$$

$$\text{FOM} = \frac{1}{\left(\frac{u}{\overline{x}}\right)^2 T} \tag{8.2.13}$$

where T is the calculation time (in minutes).

Extended tallies are used for the total neutron flux, total photon flux and any responses for the Monaco tallies. Simple tallies are used for each group's flux and each group's contribution to a response.

Detailed, group-wise results for each tally are saved to separate files at the end of each batch of particles. Users can view these files (in the SCALE temporary directory) as the Monaco simulation progresses. Summaries of the extended tallies appear in the final Monaco output file.

Statistical tests

Statistical tests are performed on the extended tallies at the end of each batch. Results for each batch are stored in files and the results for the final batch are shown in the main output tally summary. The six tests are:

Table 8.2.2: Statistical tests used by tallies

	Quantity	**Test**	**Goal**	**Within**
1.	mean	relative slope of linear fit	= 0.00	±0.10
2.	standard deviation	exponent of power fit	= -0.50	$R^2 > 0.99$
3.	relative uncertainty	final value	< 0.05	
4.	relative VOV	exponent of power fit	= -1.00	$R^2 > 0.95$
5.	relative VOV	final value	< 0.10	
6.	figure-of-merit erit	relative slope of linear fit	= 0.00	±0.10

For the tests that are fit to a function with respect to batch (1, 2, 4, and 6), only the last half of the simulation is used. The basis for these tests is that in a well-behaved Monte Carlo, the mean should not increase or decrease as a function of the number of histories (N), the standard deviation should decrease with $\frac{1}{\sqrt{N}}$, the variance of the variance should decrease with $\frac{1}{N}$ and the figure-of-merit should neither

increase or decrease as a function of the number of histories (proportional to time). For tests 2 and 4, the coefficient of determination, R^2, from a forced fit to a function with the right exponent is used as the tally test.

Point detector tallies

Point detectors are a form of variance reduction in computing the flux or response at a specific point. At the source emission site and at every interaction in the particle's history, an estimate is made of the probability of the particle striking the position of the point detector. For each point detector, Monaco tallies the uncollided and total flux for each energy group, the total for all neutron groups, and the total for all photon groups. Any number of optional dose-like responses can be calculated as well.

Multigroup

After a source particle of group g is started, the distance R between the source position and the detector position is calculated. Along the line connecting the source and detector positions, the sum of the distance sj through each region j multiplied by the total cross section Σ_j^g for that region is also calculated. The contribution cg to the uncollided flux estimator is then made to the tally for group g.

$$c_g = \frac{1}{4\pi R^2} \exp\left(- \sum_j s_j \Sigma_j^g \right) \tag{8.2.14}$$

Continuous Energy

After a source particle with energy E is started, the distance R between the source position and the detector position is calculated. For each bin g of the tally energy structure, a specific energy E_g is sampled uniformly within the bin. Along the line connecting the source and detector positions, the sum of the distance sj through each region j multiplied by the total cross section $\Sigma_j\left(E_g\right)$ for that region. The contribution cg to the uncollided flux estimator is then made to the tally for group g. total cross section $\Sigma_j(E)$:

$$c_g = \frac{1}{4\pi R^2} \exp\left(- \sum_j s_j \Sigma_j(E) \right) \tag{8.2.15}$$

Only source particles contribute to the uncollided flux tally. At each interaction point during the life of the particle, similar contributions are made to each of the tallies. For each group g' that the particle could scatter into and reach the detector location, a contribution is made that also includes the probability to scatter from the current direction towards the detector and having the energy change from group g to group g'.

This type of tally is costly, since ray-tracing through the geometry from the current particle position to the detector location is required many times over the particle history. Point detectors should be located in regions made of void material, so that contributions from interactions arbitrarily close to the point detector cannot overwhelm the total estimated flux (as $\frac{1}{4\pi R^2 \to \infty}$).

Care must be taken in using point detectors in deep penetration problems to ensure that the entire phase space that could contribute has been well sampled-so that the point detector is not underestimating the flux by leaving out areas far from the source but close to the point detector position. One way to check this is by examining how the tally average and uncertainty change with each batch of particles used in the simulation. Large fluctuations in either quantity could indicate that the phase space is not being sampled well.

Region tallies

Region tallies are used for calculating the flux and/or responses over one of the regions listed in the SGGP geometry. Both the track-length estimate of the flux and the collision density estimate of the flux are calculated-and for each, the region tally contains simple tallies for finding flux in each group, the total neutron flux, and the total photon flux. For each of the optional response functions, the region tally also contains simple tallies for each group and the total response.

For the track-length estimate of flux, each time a particle of energy E moves through the region of interest, a contribution of l (the length of the step in the region) is made to the history score for the simple tally for flux for tally group g. The same contribution is made for the history score for the simple tally for total particle flux, neutron or photon, depending on the particle type.

If any optional response functions were requested with the tally, then the contribution of $\mathrm{lf}(E)$is made for the response group, where $f(E)$ is the response function value for energy E. The history score for the total response function is also incremented using $\mathrm{lf}(E)$.

At the end of all of the histories, the averages and uncertainties of all of the simple tallies for fluxes are found for every group and each total. These results then represent the average track-length over the region. To determine flux, these results are divided by the volume of the region. If the volume V of the region was not given in the geometry input nor calculated by Monaco, then the tally results will be just the average track lengths and their uncertainties. A reminder message is written to the tally detail file if the volume of the region was not set.

For the collision density estimate of the flux, each time a particle of energy E has a collision in the region of interest, a contribution of $\frac{1}{\Sigma}$ (the reciprocal of the total macroscopic cross section) is made to the history scores for the simple tally for flux for tally energy group g and for the total particle flux. At the end of the simulation, the averages and uncertainties of all of the simple tallies for every group flux and total flux are found and then divided by the region volume, if available.

Similar to the point detector tallies, region tallies produce a file listing the tally average and uncertainty at the end of each batch of source particles (a *.chart file). This file can be plotted using the simple 2-D plotter (ChartPlot) to observe the tally convergence behavior.

Mesh tallies

For a D Cartesian mesh or a cylindrical mesh (independent of the SGGP geometry), Monaco can calculate the track-length estimate of the flux. Since the number of cells (voxels) in a mesh can become quite large, the mesh tallies are not updated at the end of each history but are instead updated at the end of each batch of particles. This prevents the mesh tally accumulation from taking too much time but means that the estimate of the statistical uncertainty is slightly low.

Like the other tallies, mesh tallies can calculate optional response functions.

Since a mesh tally consists of many actual tallies, the statistical tests are a bit more complex than for the region and point detector tallies. Several statistical quantities and tests are used in Monaco similar to those in several recent studies [MONACO-KI11, MONACO-KS11] which look at a distribution of relative variances over the mesh tally. In Monaco, the basis of the statistical tests center on the distribution of relative uncertainties and its mean, $\bar{r}$, of the voxels (V) with score.

$$\bar{r} = \frac{1}{V} \sum R_i \tag{8.2.16}$$

where R_i is the relative uncertainty of the flux or dose in voxel i. If every voxel has been sampled well and its relative uncertainty $R_i \propto \frac{1}{\sqrt{N}}$, then the mean relative uncertainty of the voxels should also behave as $\frac{1}{\sqrt{N}}$. The variance of the mean relative uncertainty can be calculated and a figure of merit (FOM) for the mesh tally can be constructed using

$$FOM = \frac{1}{\bar{r}^2 T} \tag{8.2.17}$$

with the time T in minutes. The four tests measure over the simulation: 1) if ζ, the fraction of voxels with non-zero score, is constant; 2) if the mean relative uncertainty is decreasing as $\frac{1}{\sqrt{N}}$ (as measured by the coefficient of determination, R^2, of a fit to a curve with power of -0.5); 3) if the variance of the mean relative uncertainty is decreasing with $\frac{1}{N}$; and 4) if the FOM is constant.

Table 8.2.3: Mesh tally statistical tests

	Quantity	Test	Goal	Within
1.	ζ, fraction with score	relative slope of linear fit	= 0.00	±0.10
2.	$\bar{r}$, mean relative uncertainty	exponent of power fit	= -0.50	$R^2 > 0.99$
3.	variance of $\bar{r}$	exponent of power fit	= -1.00	$R^2 > 0.95$
4.	figure-of-merit	exponent of power fit	= 0.00	±0.10

For non-uniform meshes (especially cylindrical), these tests may not be the best measure of performance since different size voxels will have a wider variety of relative uncertainties. The user is also cautioned that if there are individual voxels within the mesh tally that have relative uncertainties that are not decreasing as $\frac{1}{\sqrt{N}}$, then the mesh tally statistical tests will not be meaningful. It is ultimately up to the user to decide if the mesh tally is performing well (is the goal of the mesh tally just to calculate dose, not flux?; are all spatial areas of the mesh tally equally important?; are all magnitudes of the flux or response values equally important?; etc.)

Mesh tallies can be viewed with the Mesh File Viewer, a Java utility that can be run from GeeWiz (on PC systems) or can be run separately (on any system). The Mesh File Viewer will show the flux for each group, the total flux for each type of particle and the optional responses. Uncertainties and relative uncertainties can also be shown for mesh tallies using the Mesh File Viewer. For more information on the Mesh File Viewer,

see its on-line documentation.

8.2.2.3 Continuous Energy Transport

Using multigroup data in Monte Carlo transport calculations is generally sufficient for most problems (both shielding and criticality). Many of the reaction cross sections vary slowly with energy, so energy "groups" can be made with one set of properties for the group. Multigroup treatments can further simplify radiation transport by combining the different types of reactions that can occur into a simple scattering matrix — particles then have certain probabilities to scatter from their current energy group to another energy group. If the user is not interested in knowing which specific type of interaction happened at each collision, this simplification can increase calculation efficiency.

One major drawback of the multigroup approach is in representing discrete gammas, such as the decay radiation from common isotopic sources. Consider a simple shielding simulation using cobalt-60. This isotope gives off two high-energy gamma rays when it decays (1173230 eV with intensity 99.85% and 1332490 eV with intensity 99.9826%). In the SCALE multigroup calculations, a cobalt-60 source spectrum is represented by a broad pdf, controlled by the group structure. This is shown in Fig. 8.2.1. for the fine 47-group structure and the broad 19-group structure.

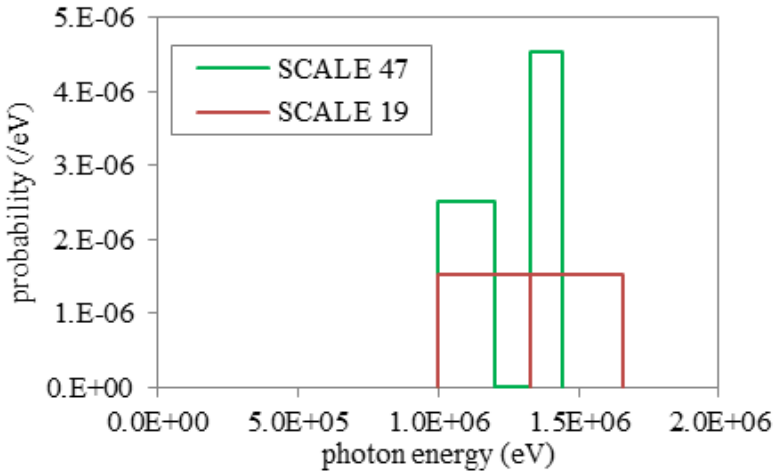


Fig. 8.2.1: The multigroup representation of a cobalt-60 source.

Note that in both group structures, 1.33 MeV is a group boundary, so the 1332490 eV line is represented by group that covers higher energies. The cross section for that group is lower than the cross section for the specific line, so multigroup transport calculations will tend to overestimate the number of photons penetrating a shield, which will overestimate dose rates.

Using CE and the two multigroup libraries, the total cross sections for the cobalt lines are listed in Table 8.2.4. Fig. 8.2.2. shows the total cross section of photons in tungsten, in both CE and the two SCALE multigroup structures. On the whole, the multigroup data represents the CE data well. Fig. 8.2.3. shows the same cross section information near the two cobalt lines, which shows how the multigroup cross sections average over quite large energy ranges.

Table 8.2.4: Total macroscopic cross section in tungsten (/cm).

	1173230 eV	1332490 eV
SCALE CE	1.03353	0.94864
SCALE 47	1.09066	0.92743
SCALE 19	1.05167	0.89289

The small differences in cross section can make large differences in the transport. Consider just 5 cm of tungsten. Using the cross sections in Table 8.2.4, the attenuation ($e^{-\mu x}$) of either line can vary by 30%.

In addition to source representation problems, multigroup transport is not adequate for applications where line spectra are measured. Because of the group structure, tally results will be averaged out within a group. With the fixed boundaries, specific lines in the tallies will not be able to be seen. For examples, in the 19-group library, there is no group around the 511 keV annihilation gammas — they are averaged in with other photons from 400 to 600 keV. No multigroup structure could contain thin groups around every line of interest.

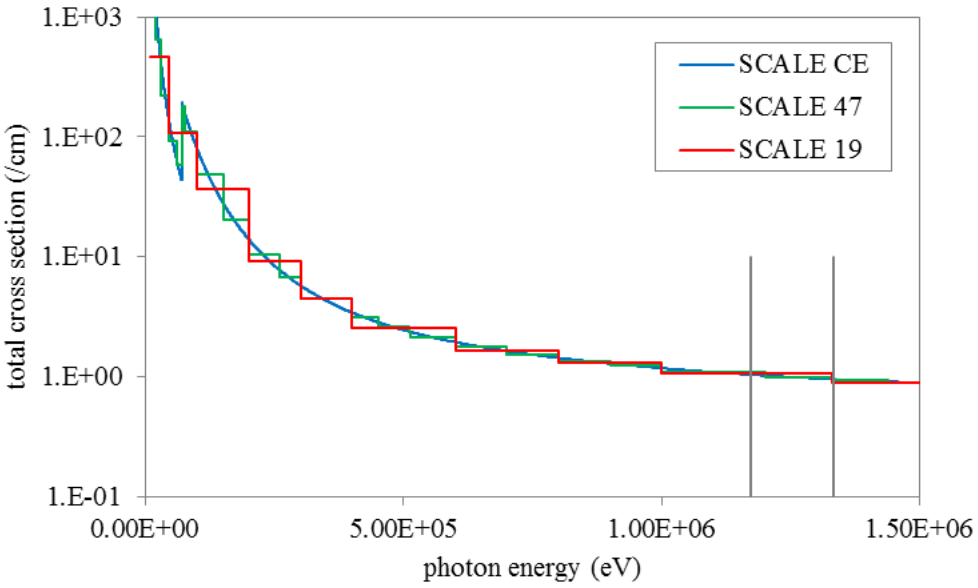


Fig. 8.2.2: Photon total cross section in tungsten. The energies of the cobalt-60 are displayed as lines at 1173230 and 1332490 eV.

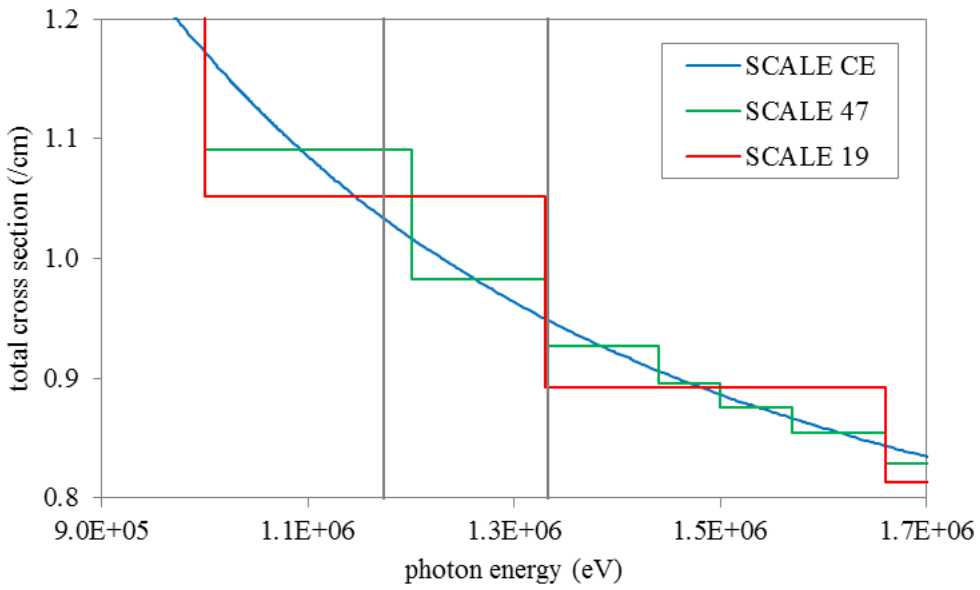


Fig. 8.2.3: Photon total cross section in tungsten, near the cobalt lines. The energies of the cobalt-60 are displayed as lines at 1173230 and 1332490 eV.

A sample problem involving a cobalt source and a slab of tungsten will compare the use of continuous-energy transport to multigroup transport, to demonstrate the large difference in results for single-line sources. For distributions, differences between multigroup and continuous-energy may not be very significant.

8.2.3 MONACO INPUT FILES

The input file for Monaco consists of two lines of text ("=monaco" command line and one for the problem title) and then several blocks, with each block starting with "read xxxx" and ending with "end xxxx". There are three blocks that are required and seven blocks that are optional. The cross section and geometry blocks must be listed first and in the specified order. Other blocks may be listed in any order.

Blocks (must be in this order):

- Cross Sections – (required) lists the cross-section file and the mixing table information
- Geometry – (required) SCALE general geometry description
- Array – optional addition to the above geometry description
- Volume – optional calculation or listing of region volumes
- Plot – create 2D slices of the SGGP geometry

Other Blocks (any order, following the blocks listed above):

- Definitions – defines locations, response functions, grid geometries, cylindrical mesh geometries, energy bin boundaries, time bin boundaries and various distributions used by other blocks
- Source – (required) description of multiple sources; with the spatial, energy, and directional distributions and particle type for each
- Tallies – description of what to calculate: point detector tallies, region tallies, or mesh tallies
- Parameters – how to perform the simulation (random number seed, how many histories, etc.)

- Biasing – data for reducing the variance of the simulation

The physical model blocks (Geometry, Array, Volume and Plot) follow the standard SCALE format. See the other SCALE references as noted in the following sections for details.

For the other six blocks, scalar variables are set by "keyword=value", fixed length arrays are set with "keyword $value_1$ ... $value_N$", variable length arrays are set with "keyword $value_1$ ... $value_N$ end", and some text and filenames are read in as quoted strings. Single keywords to set options are also used in some instances. The indention, comment lines, and upper/lower case shown in this document are not required-they are used in the examples only for clarity. Except for strings in quotes (like filenames), SCALE is not case sensitive.

After all of the blocks are listed, a single line with "end data" should be listed. A final "end" should also be listed, to signify the end of all Monaco input. See Example 8.2.1 for an overview of the Monaco input file structure.

8.2.3.1 Cross sections block

Monaco does its own mixing, so it needs a mixing table. For each element of each mixture, an identifier and a number density must be supplied. These can be found in the output of whatever sequence was used to make the cross-section file, such as CSAS-MG. Two coupled neutron/photon multigroup libraries were created specifically for shielding problems from ENDF/B-VII.0 data-the v7-200n47g fine-group and the v7-27n19g coarse-group libraries. CE libraries made from ENDF/BVII.0 are also available in SCALE.

Example 8.2.1: Overall input format for Monaco

```
=monaco                         %   name of sequence
Some title for this prob        %   title
read crossSections              %   List of isotopes/mixtures
   ...                          %       [required block]
end crossSections               %
read geometry                   %   SCALE SGGP geometry
   ...                          %       [required block]
end geometry                    %
read array                      %   SCALE SGGP arrays
   ...                          %       [optional block]
end array                       %
read volume                     %   SCALE SGGP volume calc
   ...                          %       [optional block]
end volume                      %
read plot                       %   SGGP Plots
   ...                          %       [optional block]
end plot                        %
read definitions                %   Definitions
   ...                          %       [possibly required]
end definitions                 %
read sources                    %   Sources definition
   ...                          %       [required block]
end sources                     %
read tallies                    %   Tally specifications
   ...                          %       [optional block]
end tallies                     %
read parameters                 %   Monte Carlo parameters
   ...                          %       [optional block]
end parameters                  %
read biasing                    %   Biasing information
   ...                          %       [optional block]
end biasing                     %
end data                        %   end of all blocks
end                             %   end of Monaco input
```

For example, if CSAS-MG was used to produce an AMPX file using the following input,

```
=csas-mg
Demonstration problem, three mixtures
v7-200n47g
read composition
    uo2   1 0.2 293.0 92234 0.0055 92235 3.5 92238 96.4945 end
    ss304 2 1.0 293.0 end
    h2o   4 1.0 293.0 end
end composition
end
```

in addition to creating an AMPX file, the output would include a tables similar to

```
m i x i n g   t a b l e    (THREAD = 00 )
entry   mixture   isotope   number density   new identifier   explicit temperature
    1       1       92234     2.73451E-07             92234             293.0
    2       1       92235     1.73272E-04             92235             293.0
    3       1       92238     4.71674E-03             92238             293.0
    4       1        8016     9.78057E-03              8016             293.0

m i x i n g   t a b l e    (THREAD = 00 )
entry   mixture   isotope   number density   new identifier   explicit temperature
    1       2        6000     3.18488E-04              6000             293.0
    2       2       14028     1.57010E-03             14028             293.0
    3       2       14029     7.97625E-05             14029             293.0
    4       2       14030     5.26416E-05             14030             293.0
    5       2       15031     6.94688E-05             15031             293.0
    6       2       24050     7.59178E-04             24050             293.0
    7       2       24052     1.46400E-02             24052             293.0
    8       2       24053     1.66006E-03             24053             293.0
    9       2       24054     4.13224E-04             24054             293.0
   10       2       25055     1.74072E-03             25055             293.0
   11       2       26054     3.42190E-03             26054             293.0
   12       2       26056     5.37166E-02             26056             293.0
   13       2       26057     1.24055E-03             26057             293.0
   14       2       26058     1.65094E-04             26058             293.0
   15       2       28058     5.26873E-03             28058             293.0
   16       2       28060     2.02951E-03             28060             293.0
   17       2       28061     8.82212E-05             28061             293.0
   18       2       28062     2.81288E-04             28062             293.0
   19       2       28064     7.16357E-05             28064             293.0

 m i x i n g   t a b l e    (THREAD = 00 )
 entry   mixture   isotope   number density   new identifier   explicit temperature
    1       4        1001     6.67531E-02              1001             293.0
    2       4        8016     3.33765E-02              8016             293.0
```

which can be used to construct the Monaco cross-section block mixing table.

```
read crossSections
    ampxFileUnit=4
    mixture 1
        element    92234 2.73451E-07
        element    92235 1.73272E-04
        element    92238 4.71674E-03
        element     8016 9.78057E-03
    end mixture
    mixture 2
        element     6000 3.18488E-04
        element    14028 1.57010E-03
        element    14029 7.97625E-05
        element    14030 5.26416E-05
        element    15031 6.94688E-05
        element    24050 7.59178E-04
```

(continues on next page)

(continued from previous page)

```
        element    24052 1.46400E-02
        element    24053 1.66006E-03
        element    24054 4.13224E-04
        element    25055 1.74072E-03
        element    26054 3.42190E-03
        element    26056 5.37166E-02
        element    26057 1.24055E-03
        element    26058 1.65094E-04
        element    28058 5.26873E-03
        element    28060 2.02951E-03
        element    28061 8.82212E-05
        element    28062 2.81288E-04
        element    28064 7.16357E-05
    end mixture
    mixture 4
        element     1001 6.67531E-02
        element     8016 3.33765E-02
    end mixture
end crossSections
```

For a CE calculation, instead of the keyword "ampxFileUnit=" (which refers to a given AMPX library), the keyword "ceLibrary=" should be used with a CE library name, enclosed in quotes. Also for CE, a default temperature can be set before any mixtures are defined using the "ceTempDefault=" temperature (in Kelvins). With each mixture, a specific temperature can be set using "temperature."

Other keywords that can be used in the cross-section block for multigroup problems include flags to turn on printing of different aspects of the cross-section mixing process ("printTotals", "printScatters", "printAngleProb", "printFissionChi", "printExtra", and "printLegendre"). The keyword "fullyCoupled" can be used to specify all groups to be treated as primary groups. These keywords do not work in CE problems since the point wise data contain an enormous number of points.

Users are encouraged to use Monaco by running the MAVRIC sequence, which creates the cross-section mixing table automatically, for both multigroup and CE calculations.

8.2.3.2 Geometry block

The geometry input uses the standard SGGP, similar to KENO-VI. Input instructions can be found in *Geometry Data* in the KENO-VI chapter of the SCALE manual.

Shielding calculations (Monaco, MAVRIC, SAS4) differ from their criticality cousins (KENO V.a, KENO-VI) in a very special way-sources and detectors can be located outside of the materials where the transport takes place. To accommodate this fact in Monaco and MAVRIC, make sure that a void region (a media record using mixture 0) surrounds the source area and any point detectors, if they are not located in a region of the actual geometry.

For example, if the objective is to calculate the effectiveness of a simple slab shield, the model geometry would consist of just one slab of material. The source would be on one side of the slab, and a detector would be on the other side of the slab. In Monaco (and the MAVRIC sequence), the input should list at least two regions: (1) the slab itself and (2) a void region outside of the slab containing both the source and detector positions.

Monaco tracks particles through the SGGP geometry as well as other geometries used for mesh tallies or mesh importance maps. Because Monaco must track through all of these geometries at the same time, users should not use the reflective boundary capability in the SGGP geometry.

The graphical user interfaces GeeWiz and Keno3D can be used on Windows platforms to develop and view the geometry.

8.2.3.3 Array, volume, and plot blocks

Geometry array input uses the standard SGGP, similar to KENO-VI. Input instructions can be found in KENO-VI chapter on *Array Data* of the SCALE manual.

Volumes of various geometry regions are used to calculate fluxes for those regions. Volumes can be input as part of the geometry input block above, or calculated by the SGGP using one of two different methods. See KENO-VI chapter on *Volume Data* for instructions.

The "read plot" block allows users to create a 2-D character or color plots of slices through a specified portion of the 3-D geometrical representation of the problem. These images can be saved as *.png files. For more information, see the KENO-VI chapter on *Plot Data*.

8.2.3.4 Definitions block

The definitions block defines different types of data (locations, detector response functions, grid geometries, cylindrical geometries, distributions, energy bin boundaries and time bin boundaries) that are used by some of the other blocks in Monaco. Individual data can be listed in any order. Identification numbers must be positive integers and unique within that type of data. Each type of data begins with a keyword and ends with an "end" and that same keyword. All of the different data types can have an optional title using the keyword "title=".

```
read definitions
    location 43
        ...
    end location
    response 45
        ...
    end response
    distribution 1
        ...
    end distribution
    response 12
        ...
    end response
end definitions
```

Locations

Locations ("location") require an identification number and the physical position in global coordinates using the "position" keyword (a fixed length array). A position is specified by listing its x, y, and z coordinates.

```
location  1
   title="Radial detector - close to surface"
    position 162.0 0.0 0.0
end location
location  2  position 0.0 0.0 295.6   end location
location  3
    title="Corner detector"
    position 162.0 0.0 295.6
end location
location 105 position   0.0 0.0 385.6   end location
location 106 position 252.0 0.0 385.6   end location
```

Response functions

Response functions ("response") require an identification number and information on how to build an energy dependent response function. There are three basic types of responses: 1) the general user-defined response, 2) a response based on cross-section data, and 3) a response based on a specific flux-to-dose conversion factor. For multigroup calculations, a fourth type of a response simply listing multigroup values is also available. Responses must be defined as either a neutron response or a photon response.

Type 1. A general user-defined response function can be either a binned histogram function (n+1 energies and n values) or a set of value/function pairs that will be linearly interpolated (n+1 energies and n+1 values). The energies (in eV) are set using the "bounds ... end" keyword. The response values are entered with the "values ... end" keyword. The energies can be entered from low energy to high energy order or the traditional high energy to low energy order but must be monotonic. The values array of the response is interpreted to correspond to the order of the bounds array. These two examples

```
response 11
    title="user-defined response, histogram"
    neutron
    bounds 1e7   8e6   6e6   4e6   2e6   1e5 end
    values    1.0   0.8   0.6   0.4   0.2    end
end response
response 12
    title="user-defined response, value/function pairs"
    photon
    bounds  1e5 2e6 4e6 6e6 8e6 1e7 end
    values 0.01 0.2 0.4 0.6 0.8 1.0 end
end response
```

are shown in Fig. 8.2.4 and Fig. 8.2.5.

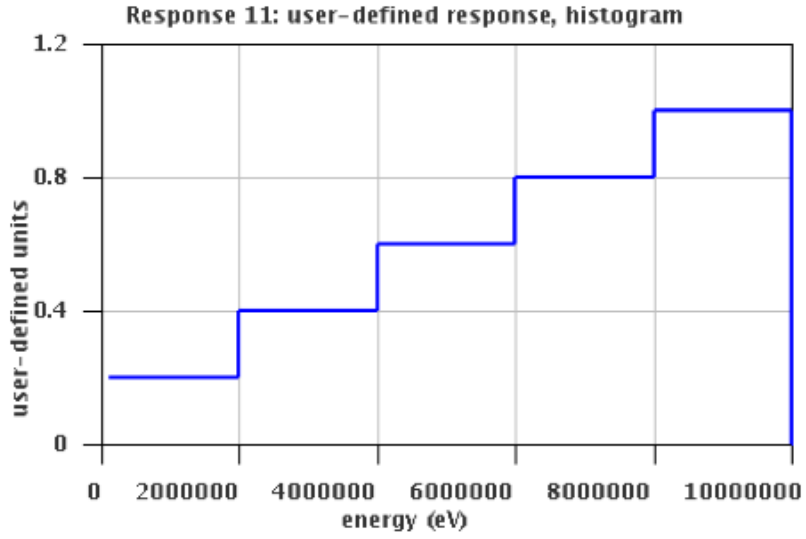


Fig. 8.2.4: Histogram-type response.

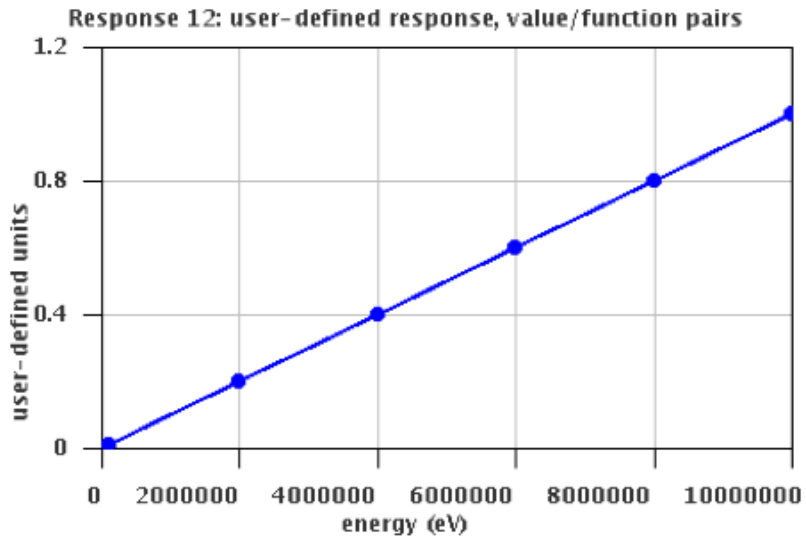


Fig. 8.2.5: Value/function pair response.

Type 2. Data from the cross-section library can also be used to define a response, for example in finding reaction rates. For the cross section (with units of barns) for a single isotope, the user specifies a material/ZAID/MT combination. The keyword "macro" can be used to multiply the cross section by the atom density of the ZAID in the material (which converts the units of the response from barns to /cm). Users can also specify just the material and MT numbers, to produce the macroscopic cross section of reaction MT for the entire material (with units of /cm). A partial list of common MT numbers is shown in Table F23.3.2 (the full list is in XSECLIB M04, Appendix B). To match some other sequences in SCALE, users can also use text strings to specify the ZAID and MT by using keywords "nuclide=" (for example, nuclide=U-235) and "reaction=" (for example reaction=fission). If the user requested a microscopic cross section response for a reaction in a CE problem, the response will be generated for the nuclide from the AMPX CE libraries even if the nuclide itself is not included in any of the material definitions in the problem. Available reaction lists depend on the nuclide and the list will be printed as a warning message in the output if a non-existing reaction is requested.

```
read composition
    uo2 7 1.0 293.0 end
end composition
...
read definitions
    response 41
        title="get the microscopic (b) for 235"
        neutron
        material=7 ZAID=92235 MT=18
    end response
    response 43
        title="get the macroscopic (/cm) for 235"
        neutron
        material=7 ZAID=92235 MT=18
        macro
    end response
    response 45
        title="get the macroscopic (/cm) for UO_2 (234, 235, 238)"
        neutron
```

(continues on next page)

(continued from previous page)

```
        material=7 MT=18
    end response
end definitions
```

For the examples above, response 41 is shown in Fig. 8.2.7. and Fig. 8.2.8. for both MULTIGROUP and CE.

MT	Description
1	Total cross section
18	Total fission cross section
27	Absorption cross section (MT=18 and 101)
101	Neutron disappearance
102	(n,γ) radiative capture cross section
103	(n,p) cross section
104	(n,^{2}H) cross section
105	(n,^{3}H) cross section
106	(n,^{3}He) cross section
107	(n,^{4}He) cross section
1452	Product of ν times the fission cross section

MT	Description
501	Total photon interaction cross section
502	Photon coherent scattering
504	Photon incoherent scattering
516	Pair production, nuclear and electron field
518	Photofission (γ,f)
522	Photoelectric

Fig. 8.2.6: Common MT (reaction) numbers for responses

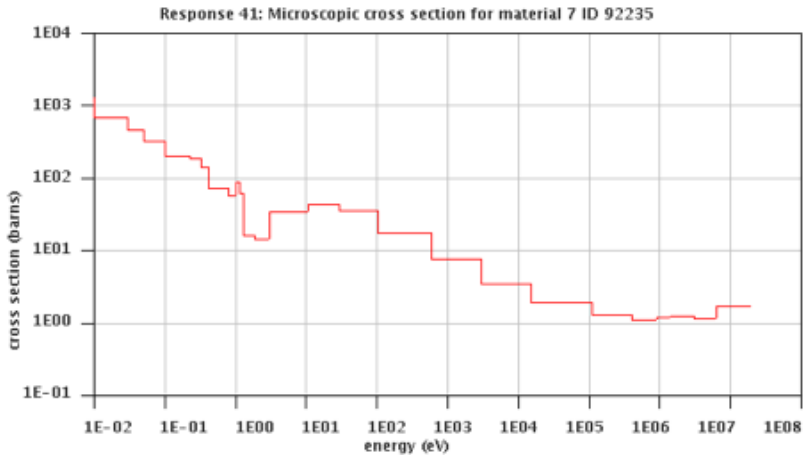


Fig. 8.2.7: Multigroup ^{235}U total fission cross section.

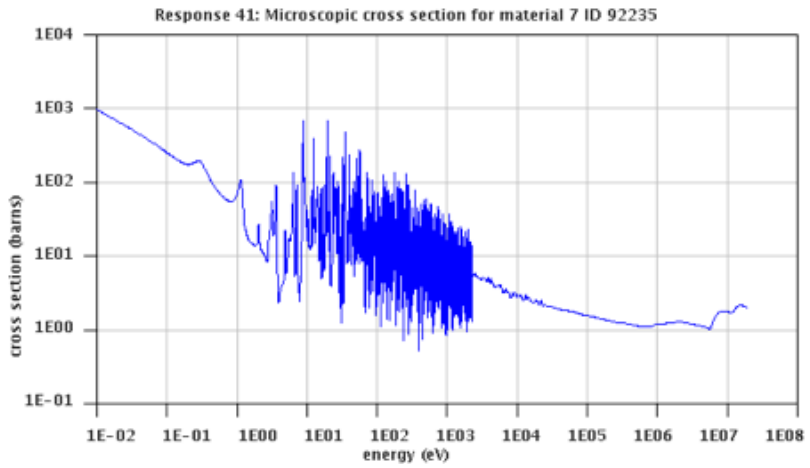


Fig. 8.2.8: CE ^{235}U total fission cross section.

Type 3. Flux-to-dose conversion factors are a little different in multigroup and continuous-energy implementations. The AMPX multigroup shielding libraries contain neutron and photon dose responses from several sources. These have been processed by the AMPX system (the jergens module). To form the multigroup values for the libraries, the original data was extrapolated to cover the entire energy range of the shielding libraries and was then collapsed into the group structures using a weighting spectrum. These dose responses can be accessed through Monaco/MAVRIC by defining a response object that uses the keyword "specialDose=" and then providing the MT number of the particular response. The dose responses available in the shielding libraries in are shown in Fig. 8.2.9. Note that the coupled responses in SCALE 6.1 are no longer used by Monaco, since responses are now defined to be either a neutron response or a photon response. When using the "specialDose=" keyword, the "neutron" or "photon" designation is ignored, since the particle type is inherent with the MT number.

```
read definitions
    response 1
        specialDose=9031
    end response
end definitions
```

Response		Neutron MT	Neutron Units	Photon MT	Photon Units
Henderson conversion factors		9027	(rad/h)/(n/cm^2/s)	9502	(rad/h)/(p/cm^2/s)
Claiborne-Trubey conversion factors				9503	(rad/h)/(p/cm^2/s)
ANSI standard (1977) flux-to-dose-rate factors		9029	(rem/h)/(n/cm^2/s)	9504	(rem/h)/(p/cm^2/s)
ANSI standard (1991) flux-to-dose-rate factors		9031	(rem/h)/(n/cm^2/s)	9505	(rem/h)/(p/cm^2/s)
ICRU-44 Table B.3 (air) Kerma		9032	(Gy/h)/(n/cm^2/s)		
		9033	(rad/h)/(n/cm^2/s)		
ICRU-57 Table A.21 (air) Kerma				9506	(Gy/h)/(p/cm^2/s)
				9507	(rad/h)/(p/cm^2/s)
Ambient dose equivalent (ICRU-57)		9034	(Sv/h)/(n/cm^2/s)	9508	(Sv/h)/(p/cm^2/s)
		9035	(rem/h)/(n/cm^2/s)	9509	(rem/h)/(p/cm^2/s)
Effective dose (ICRU-57)		9036	(Sv/h)/(n/cm^2/s)	9510	(Sv/h)/(p/cm^2/s)
		9037	(rem/h)/(n/cm^2/s)	9511	(rem/h)/(p/cm^2/s)
Dosimetry for Criticality Accidents	Personal absorbed dose	8001	(Gy/h)/(n/cm^2/s)	8511	(Gy/h)/(p/cm^2/s)
(ANSI/HPS N13.3-2013, Table B1 & B2)	Ambient absorbed dose	8002	(Gy/h)/(n/cm^2/s)	8512	(Gy/h)/(p/cm^2/s)
	(air) Kerma			8501	(Gy/h)/(p/cm^2/s)

Fig. 8.2.9: Flux-to-Dose conversion factor MT numbers

The standard flux-to-dose conversion factors have not been made part of the continuous-energy libraries. Routines have been added to the Monaco code base to generate data points to allow users to define responses based on the original references. Note that the responses in these references were defined over different energy ranges, as shown in Fig. 8.2.10.

Response	Neutron Energy Range (MeV)		Photon Energy Range (MeV)	
Henderson conversion factors	0.01	18	0.01	10
Claiborne-Trubey conversion factors			0.02	16
ANSI standard (1977) flux-to-dose-rate factors	2.5E-08	20	0.01	15
ANSI standard (1991) flux-to-dose-rate factors	2.5E-08	14	0.01	12
ICRU-44 Table B.3 (air) Kerma	2.5E-08	29		
ICRU-57 Table A.21 (air) Kerma			0.01	10
Ambient dose equivalent (ICRU-57)	1.0E-09	20.1	0.01	10
Effective dose (ICRU-57)	1.0E-09	18	0.01	10
Dosimetry for Criticality Accidents (ANSI/HPS N13.3-2013)	1.0E-09	20	0.01	10

Fig. 8.2.10: Energy ranges of the original Flux-to-Dose responses

The keyword "doseData=" can be used to create a response using the original, point-wise data (except for Claiborne-Trubey where the original data is a histogram). Data points are also extrapolated to cover the energy range of 10^{-5} to 2×10^7 eV for neutrons and up to 20 MeV for photons. (The optional keyword "noExtrapolation" can be used to get just the original data without the extrapolations.) The final response is formed by interpolating (lin-lin) between these points. For multigroup problems, these keywords will collapse the original data (with or without extrapolation) into a multigroup structure but without the weighting function

used to create the dose factors in the multigroup libraries. This will not match the multigroup responses in the those libraries.

```
read definitions
    response 1
        doseData=9031
    end response
    response 1
        doseData=9031  noExtrapolation
    end response
end definitions
```

As an example of the various forms of a flux-to-dose conversion factor, the ANSI 1991 values (MT=9031 and 9505) are shown in Fig. 8.2.11 through Fig. 8.2.14.

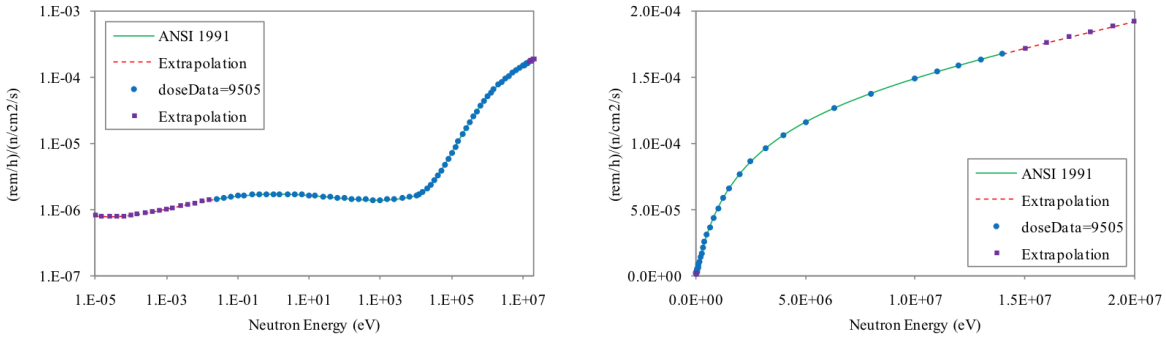


Fig. 8.2.11: ANSI 1991 neutron CE (left is log-log, right is linear-linear)

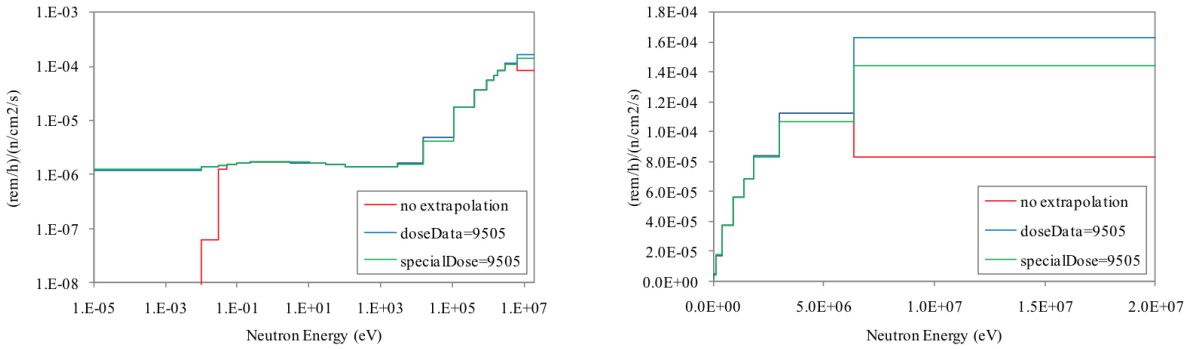


Fig. 8.2.12: ANSI 1991 neutron MULTIGROUP (left is log-log, right is linear-linear)

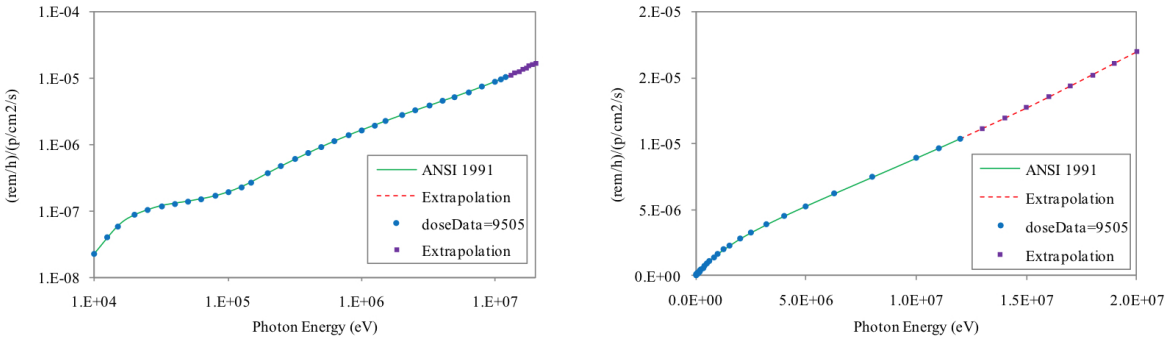


Fig. 8.2.13: ANSI 1991 photon CE (left is log-log, right is linear-linear)

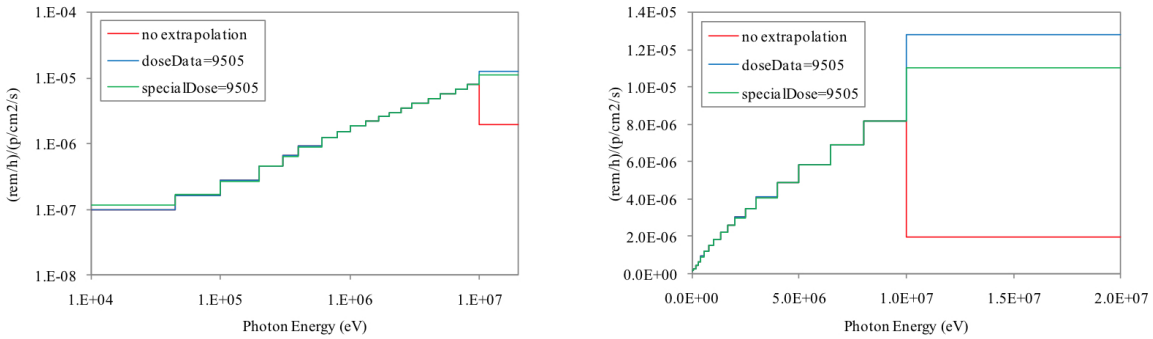


Fig. 8.2.14: ANSI 1991 photon MULTIGROUP (left is log-log, right is linear-linear)

The use of the `specialDose=` and `doseData=` keywords is summarized in Fig. 8.2.15. Users should understand that the only way to get the 'true' response described in the original references is to use the `doseData=` and `noExtrapolation` keywords. The traditional approach in SCALE has been to extrapolate the original data over the entire energy range of the problem, yielding higher dose rates than the 'true' response would.

keyword(s)	Multi-Group (MG)	Continuous Energy (CE)
specialDose=*MT*	uses pre-computed responses contained in the MG library: data points from reference, extrapolated to energy range of the library and collapsed to the library group structure using a weighting spectrum	this keyword is not allowed since there are no pre-computed dose responses contained within the CE libraries
doseData=*MT*	data points from reference, extrapolated over the range of 1.0d-11 to 20.0 MeV and collapsed to the library group structure without a weighting spectrum	data points from reference, extrapolated over the range of 1.0d-11 to 20.0 MeV
doseData=*MT* noExtrapolation	data points from reference and collapsed to the library group structure without a weighting spectrum	data points from reference using only the original energy range

Fig. 8.2.15: Use of the "specialDose=" and "doseData=" keywords.

Type 4. For multigroup calculations, since the energy structure is already known, a response can be defined by listing just the values for each group using the keyword "values ... end". The array length of this type of response should match the number of energy groups for that particle type in the cross-section library. Values should be entered in the standard multigroup order — from high energy to low energy. The shortcut keyword "unity" places a value of 1.0 as the response for each group.

```
response 19
    title="Total Photon Dose at Each Detector Point Location (ANSI 9504)"
    photon
    values                          1.16200E-05  8.74457E-06  7.45967E-06
         6.35058E-06  5.39949E-06  4.60165E-06  3.95227E-06  3.45885E-06
         3.01309E-06  2.62001E-06  2.19445E-06  1.82696E-06  1.51490E-06
         1.15954E-06  8.70450E-07  6.21874E-07  3.70808E-07  2.68778E-07
         5.93272E-07  end
end response
response 4
     title="total photon flux above 1 MeV, photons/(/cm2/sec)"
     photon
     values 11r1.0  8r0.0 end
end response
response 99
     title="put a 1 in every group"
     neutron
     unity
end response
```

The different response types all share some optional keywords. The keyword "makeChart" can be used to produce a *.chart file (called `\ *outputName*.resp\ *id*.chart`) so that the response can be plotted with the ChartPlot 2D plotter. To create files for every response, use the keyword `makeCharts` inside the definitions block but outside any particular response definition. The keyword `multiplier=` can be used with any type of response, which is useful for things such as units conversions. Multiple uses of the `multiplier=` keyword within one response definition will apply the product of all

multipliers to that response. Using the keyword `multiplier=` in the definitions block but outside any particular response will apply that multiplier to all responses. Keywords `eHigh=` and `eLow=` can be used to only keep the response values in a range between eHigh and eLow (both in eV). The keyword "lessOutput" can be used to suppress response data echoing in the output file and minimize output file size particularly for CE responses that can have fine point-wise data. It will cause to print only the first five and the last five points of the data if the number of bins is greater than twenty for binned histogram and value/function pairs type of responses.

The original flux-to-dose conversion factor references that were incorporated into Monaco are:

- ANSI/ANS-6.1.1-1977 (N666) "American National Standard Neutron and Gamma-Ray Flux-to-Dose-Rate Factors," Prepared by the American Nuclear Society Standards Committee Working Group ANS-6.1.1, Published by the American Nuclear Society, 555 North Kensington Avenue LaGrange Park, Illinois 60525, Approved March 17, 1977 by the American National Standards Institute, Inc.
- ANSI/ANS-6.1.1-1991, "American National Standard for Neutron and Gamma-Ray Fluence-to-Dose Factors," Prepared by the American Nuclear Society Standards Committee Working Group ANS-6.1.1, Published by the American Nuclear Society, 555 North Kensington Avenue LaGrange Park, Illinois 60525 USA, Approved August 26, 1991 by the American National Standards Institute, Inc.
- H. C. Claiborne and D. K. Trubey, "Dose Rates in a Slab Phantom from Monoenergetic Gamma Rays," *Nuclear Applications & Technology*, Vol. 8, May 1970.
- B. J. Henderson, "Conversion of Neutron or Gamma Ray Flux to Absorbed Dose Rate," ORNL Report No. XDC-59-8-179, August 14, 1959.
- International Commission of Radiation Units and Measurements, *ICRU Report 44: Tissue Substitutes in Radiation Dosimetry and Measurement*, Bethesda, MD, 1989.
- International Commission of Radiation Units and Measurements, *ICRU Report 57: Conversion Coefficients for use in Radiological Protection Against External Radiation*, Bethesda, MD, August 1, 1998.
- ANSI/HPS N13.3–2013, "Dosimetry For Criticality Accidents," Prepared by the Health Physics Society, 1313 Dolley Madison Blvd. Suite 402, McLean, VA, 2013

Grid geometries

Grid geometries ("gridGeometry") require an identification number and then a description of a 3D rectangular mesh by specifying the bounding planes of the cells in each of the x, y, and z dimensions. The keyword "xplanes ... end" can be used to list plane values (in any order). The keyword `xLinear n a b` can be used to specify n cells between a and b. The keywords "xplanes" and "xLinear" can be used together and multiple times — they will simply add planes to any already defined for that dimension. Any duplicate planes will be removed. Similar keywords are used for the y- and z-dimensions.

```
gridGeometry 3
    title="Boring uniform grid"
    xLinear 10 -100 100
    yLinear 10 -100 100
    zLinear 10 -100 100
end gridGeometry
gridGeometry 2
```

(continues on next page)

(continued from previous page)

```
    xplanes -100.0 -90.0 -99.0 -95.0 end
    xLinear  9 -90.0  0.0
    xLinear 18   0.0 90.0
    xplanes 95.0 100.0 99.0  end
    yLinear 20  100.0 -100.0
    zLinear 40  100.0 -100.0
end gridGeometry
```

When using multiple instances of the keyword *Linear and *planes for a given dimension, duplicates should be removed from the final list. In some cases, double precision math will leave two planes that are nearly identical but not removed (for example: 6.0 and 5.9999999). To prevent this, a default tolerance is set to remove planes that are within 10^{-6} cm of each other. The user is free to change this by using the keyword "tolerance=" and specifying something else. Note that the tolerance can be reset to a different value in between each use of *Linear or *planes.

The keyword "make3dmap" for a particular grid geometry definition will create a file called `outputName*.grid\ *id*.3dmap` which can be visualized using the Java Mesh File Viewer. Using the keyword `make3dmaps` in the definitions block but outside any particular gridGeometry definition will create a geometry file for each gridGeometry defined.

Cylindrical mesh geometries

Cylindrical geometries ("cylGeometry") require an identification number and then a description of a 3D cylindrical mesh by specifying the bounding planes of the cells in each of the r, θ, and z dimensions. The keywords `radii ... end`, `thetas ... end`, and `zplanes ... end` can be used to list the plane values in any order. The keywords `radiusLinear n a b`, `thetaLinear n a b`, and `zLinear n a b` can be used to specify n cells between a and b. Note that the keywords "thetas" and "thetaLinear" expect values between 0 and 2π. For entering values between 0 and 360°, use the keywords `degrees` and `degreeLinear` instead. The keywords for each dimension can be used together and multiple times — they will simply add planes to any already defined for that dimension. Any duplicate planes will be removed.

Cylindrical meshes are oriented along the positive z-axis by default. To change this, the user can specify the axis of the cylinder using the keyword `zaxis u v w` and specify the perpendicular direction where *theta* = 0 using `xaxis u v w`. To change the base position of the cylinder, use the keyword `position x y z`. Some examples of cylindrical mesh geometries include:

```
cylGeometry 12
    radiusLinear 20 100.0 168.0
    radiusLinear 10 168.0 368.0
    degreeLinear 12 0 360
    zLinear 25 255.2 -255.2
    zPlanes  -45.0 -40. -35.0 end
end cylGeometry
cylGeometry 13
    title="degenerate: only one angular bin"
    radiusLinear 10 168.0 368.0
    thetaLinear  1 0.0 6.2831853
    zLinear 25 255.2 -255.2
end cylGeometry
cylGeometry 14
    title="degenerate: emulate surface tally over partial angle range"
    radiusLinear 1 367.5 368.5
    degreeLinear 1 45 135
    zLinear 25 255.2 -255.2
    zaxis 0 0 1
    xaxis 0 -1 0
 end cylGeometry
```

Similar to the grid geometries, the user can use the keyword `tolerance=` to specify how close duplicate planes can be when being considered for removal. The keyword `makeCylMap` for a particular cylindrical geometry definition will create a file called `outputName*.cyl\ *id*.3dmap` which can be visualized using the Java Mesh File Viewer. Using the keyword "makeCylMaps" in the definitions block but outside any particular gridGeometry definition will create a geometry file for each gridGeometry defined. The Mesh File Viewer is written for rectilinear geometries and will not display circles. The only view that works in the Mesh File Viewer for cylindrical meshes is the *x*-*z* view, which will correctly show an *r*-*z* slice. The slider (marked "*y*") will control which *theta* value to display (from 0 to 2π).

Cylindrical meshes can only be used for tallies. They cannot be used for making mesh sources or for any importance calculations in MAVRIC.

Distributions

Distributions (`distribution`) require an identification number and several other keywords depending on the type of distribution. For a binned histogram distribution over n intervals, the keyword "abscissa ... end" is used to list the $n + 1$ bin boundaries and the keyword "truePDF ... end" is used to list the n values of the pdf integrated over those bins. For a pdf defined using a series of evaluated points over n intervals, use the keywords "abscissa ... end" and "truePDF ... end" listing the $n + 1$ values for each. The "truePDF" values should be the value of the pdf evaluated at the corresponding point in the abscissa array. The abscissa array should either be in increasing order or decreasing order — monotonic either way — with the truePDF array ordered accordingly.

For either the binned histogram or the value/function point pairs distributions, biasing can also be specified for a given distribution using the "biasedPDF ... end" keyword, the "weight ... end" keyword, or the "importance ... end" keyword, with a length that matches the truePDF array. Weights specify the suggested sampling weights for particles and importances specify the suggested importance. For biasing, the user only needs to specify just one of "biasedPDF", "weight" or "importance". The other arrays will be computed by Monaco.

For discrete distributions (such as gamma line sources), use the keyword "discrete ... end" to list the discrete abscissa values and use the keyword "truePDF ... end" to list the probabilities. The "biasedPDF ... end", "trueCDF ... end", and "biasedCDF ... end" keywords can also be used. Each array should have the same length–the number of discrete lines.

To visualize a distribution, add the keyword "runSampleTest" and a *.chart file will be produced showing the true pdf, the pdf used for sampling (the biased pdf) and the results of a sampling test using 10^6 samples. The file will be named using the output name of the SCALE job and the distribution identification number '*outputName*.dist*id*.chart' and can be viewed with the ChartPlot 2D Interactive Plotter. To perform a sampling test and create a *.chart file for all of the distributions in the definitions block, use the keyword "runSampleTests" inside the definitions block but outside any particular distribution.

Some example distribution inputs are listed below and shown in Fig. 8.2.16.

```
distribution 11
   title="a binned histogram"
   abscissa -5 -4 -3 -2 -1 0 1 2 3 4 5 end
   truePDF   1  2  3  4  5 4 3 2 2 2   end
end distribution
distribution 12
   title="value/function pairs"
   abscissa   -5 -4 -3 -2 -1 0 1 2 3 4 5  end
   truePDF     0  1  2  3  4 5 4 3 2 2 2  end
```

(continues on next page)

(continued from previous page)

```
end distribution
distribution 21
   title="a binned histogram with biasing"
   abscissa -5 -4 -3 -2 -1 0 1 2 3 4 5 end
   truePDF   1  2  3  4  5 4 3 2 2 2   end
   biasedPDF 3  2  1  1  1 1 1 2 2 2   end
end distribution
distribution 22
   title="value/function pairs with importances"
   abscissa   -5 -4 -3 -2 -1 0 1 2 3 4 5  end
   truePDF     0  1  2  3  4 5 4 3 2 2 2  end
   importance  4  3  2  1  1 1 1 1 2 2 2  end
end distribution
distribution 31
   title="a binned histogram using CDF's"
   abscissa -5 -4 -3 -2 -1  0  1  2  3  4 5 end
   trueCDF   1  3  6 10 15 19 22 24 26 28   end
end distribution
distribution 32
   title="a binned histogram with biasing using CDF's"
   abscissa -5 -4 -3 -2 -1  0  1  2  3  4 5 end
   trueCDF   1  3  6 10 15 19 22 24 26 28   end
   biasedPDF 3  5  6  7  8  9 10 12 14 16   end
end distribution
```

Other notes on distributions:

1) Binned histogram distributions can also be specified using cdf's (keywords "trueCDF" and "biased-CDF").

2) For distributions that will be used for source energy sampling, use abscissa values of eV.

3) For multigroup calculations using histograms, the keywords "neutronGroups" or "photonGroups" can be used instead of specifying the abscissa values. In this case, be sure to list the binned pdf values in order from the highest energy group to the lowest energy group.

4) For CE calculations, instead of specifying abscissa values, the bin boundaries of an energyBounds object (see next section) can be specified using "energyBoundsID=".

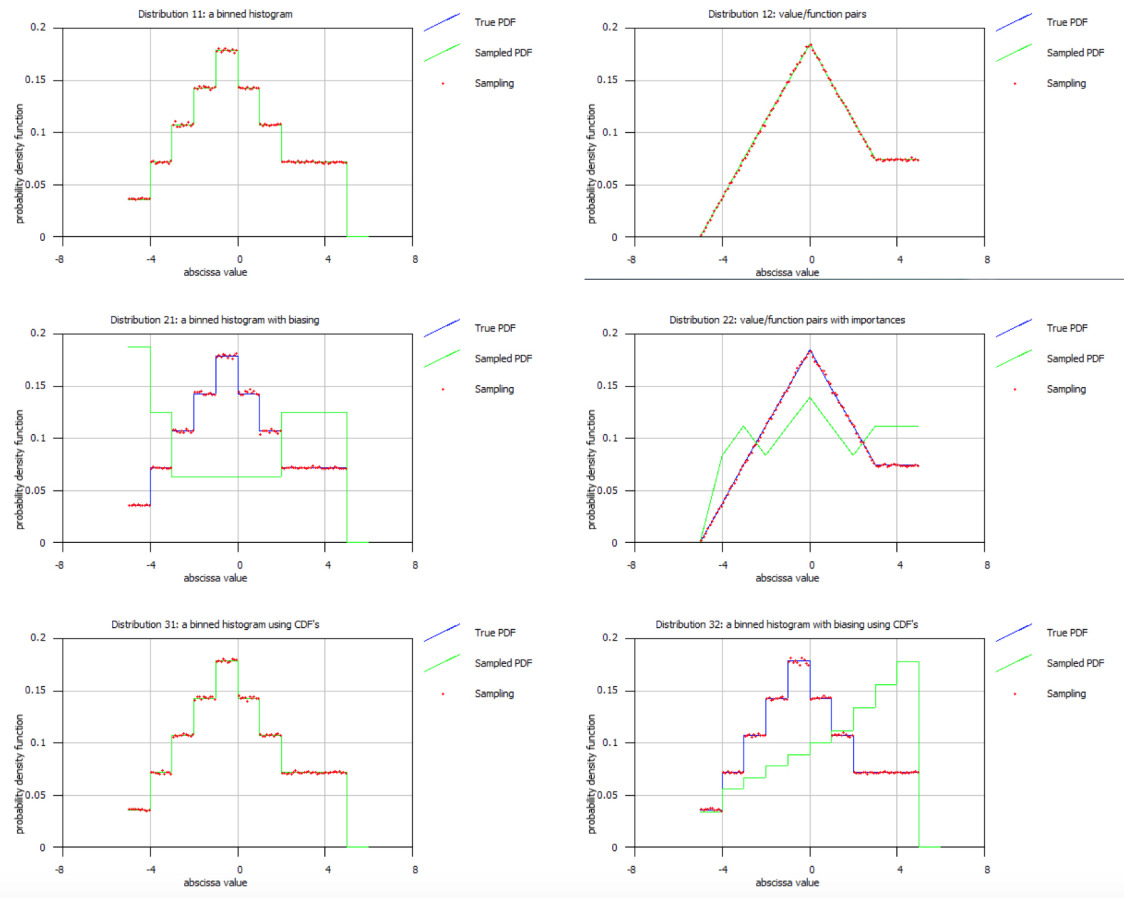


Fig. 8.2.16: Sampling tests for the distribution examples.

Several special (built-in) distributions are available in Monaco. To use one of these, specify the keyword "special=" with a distribution name in quotes and the keyword "parameters ... end" (if required) for that type of distribution. These special distributions are summarized in Table 8.2.8.

The Watt spectrum has the form

$$p(E) = ce^{-E/a}\sinh(\sqrt{bE}) \tag{8.2.18}$$

with the parameters a and b (with c as a normalization constant). For spontaneous fission of ^{252}Cf, values typically used are a=1.025 MeV and b=2.926/MeV. For thermal fission of ^{235}U, the parameters are a=1.028 MeV and b=2.249/MeV. For induced fission, the parameters a and b are, in general, functions of incident neutron energy. See Table 8.2.5 for an example. The Watt spectrum distribution will be displayed in the *.chart plot as a histogram distribution using the cross-section energy structure neutron groups but when sampled in Monaco, the continuous Froehner and Spencer[1] method is used to select an energy of source particles using a Watt spectrum distribution.

with the parameters a and b (with c as a normalization constant). For spontaneous fission of ^{252}Cf, values typically used are a=1.025 MeV and b=2.926/MeV. For thermal fission of ^{235}U, the parameters are a=1.028 MeV and b=2.249/MeV. For induced fission, the parameters a and b are, in general, functions of incident neutron energy. See Fig. 8.2.17 for an example. The Watt spectrum distribution will be displayed in the *.chart plot as a histogram distribution using the cross-section energy structure neutron groups but when

sampled in Monaco, the continuous Froehner and Spencer[1] method is used to select an energy of source particles using a Watt spectrum distribution.

Table 8.2.5: Special (built-in) distributions

Distribution	Parameters	Description
"wattSpectrum"	*a b n*	Watt spectrum distribution. Units are: *a* in MeV, *b* in /MeV. Optional parameter *n* specifies how many subintervals in each neutron group to use in integrating the pdf (default 100) for the histogram representation in the sampling test and mesh source representation.
"fissionNeutrons"	*m ZAID*	Spectrum of fission neutrons from the MULTIGROUP cross-section library for material *m* and nuclide *ZAID*.
"fissionPhotons"	*ZAID*	Spectrum of fission photons from nuclide *ZAID*.
"origensBinaryConcentrationFile"	*c s*	Spectrum from an ORIGEN-S binary concentration file case number *c*, spectra type *s*. For the spectra type *s*, values are: 1 - total neutron, 2 - spontaneous fission, 3 - (α,n), and 4 - delayed neutrons, 5 - photons. The ORIGEN-S filename should be supplied with the keyword filename= "..." and the path/filename in quotes.
"cosine"	*n*	Cosine function from -π/2 to π/2. Optional parameter *n* (default 100) is the number of value/function pairs to show in the sampling test.
"pwrNeutronAxialProfile"	none	Typical neutron PWR axial profile.
"pwrGammaAxialProfile"	none	Typical gamma PWR axial profile.
"pwrNeutronAxialProfileReverse"	none	Typical neutron PWR axial profile, reversed top to bottom.
"pwrGammaAxialProfile Reverse"	none	Typical gamma PWR axial profile, reversed top to bottom.
"exponential"	*a n*	Exponential function *eax* from -1 to 1. Optional parameter *n* (default 100) is the number of value/function pairs to show in the sampling test.
"origensDiscreteGammas"	*z a m*	Discrete gammas from the ORIGEN mpdkxgam database for isotope of atomic number *z*, mass *a* and metastable state *m*. (default is m=0)

For the ORIGEN-S binary concentration sources, the ORIGEN input file should be specified using the filename="..." with the path/filename in quotes. Note that the ORIGEN calculation has to be set to save the neutron or photon data will be used as a Monaco distribution. This can be done by specifying the number of photon or neutron groups on the 3$ (library integer constants) array and specifying the energy bin boundaries on the 83* and 84* (group structure) arrays. In Monaco, to show all of the cases in the binary concentration file, ask for case 0. To show what data is available for a particular case, ask for that case number and spectra type 0.

Other notes on special distributions: 1) Fission neutron distributions use MT=1018 for the specified ZAID of the specified isotope from the cross-section library. 2) Fission photon distributions are not read from the cross-section file but are instead read from a separate file containing only ENDF/B-VII.0 fission photon data. 3) The neutron and photon axial profile distributions come from the SCALE 5.1 SAS4 manual, Table S4.4.5. 4) Fission neutron distributions are not allowed in the CE problems, users are advised to use "wattSpectum" in order to get a similar distribution.

Energy (MeV)	a (MeV)	b (/MeV)
1.E-11	0.977	2.546
1.5	0.977	2.546
2	0.980	2.532
5	1.010	2.412
7	0.970	2.571
10	0.980	2.474
12.2	1.010	2.612
15	1.000	2.652
18	1.040	2.689
20	1.060	2.620
30	1.060	2.620

Fig. 8.2.17: Watt spectrum parameters for neutron induced fission of ^{233}U (From ENDF/B-VII.0)

Some example special distribution inputs are listed below and shown in Fig. 8.2.18.

```
distribution 11
    special="wattSpectrum"
    parameters 1.0 3.0 end
end distribution
distribution 12
    special="fissionNeutrons"
    parameters 1 92235 end
end distribution
distribution 21
    special="fissionPhotons"
    parameters 94239 end
end distribution
distribution 22
    special="origensBinaryConcentrationFile"
    filename="c:\\path\somefile.f71"
    parameters 9 5 end
end distribution
distribution 31
    special="origensBinaryConcentrationFile"
    filename="c:\\path\somefile.f71"
    parameters 9 1 end
end distribution
distribution 32
```

(continues on next page)

(continued from previous page)

```
    special="cosine"
    parameters 100 end
end distribution
distribution 41
    special="pwrNeutronAxialProfile"
end distribution
distribution 42
    special="exponential"
    parameters 1.0 100 end
end distribution
```

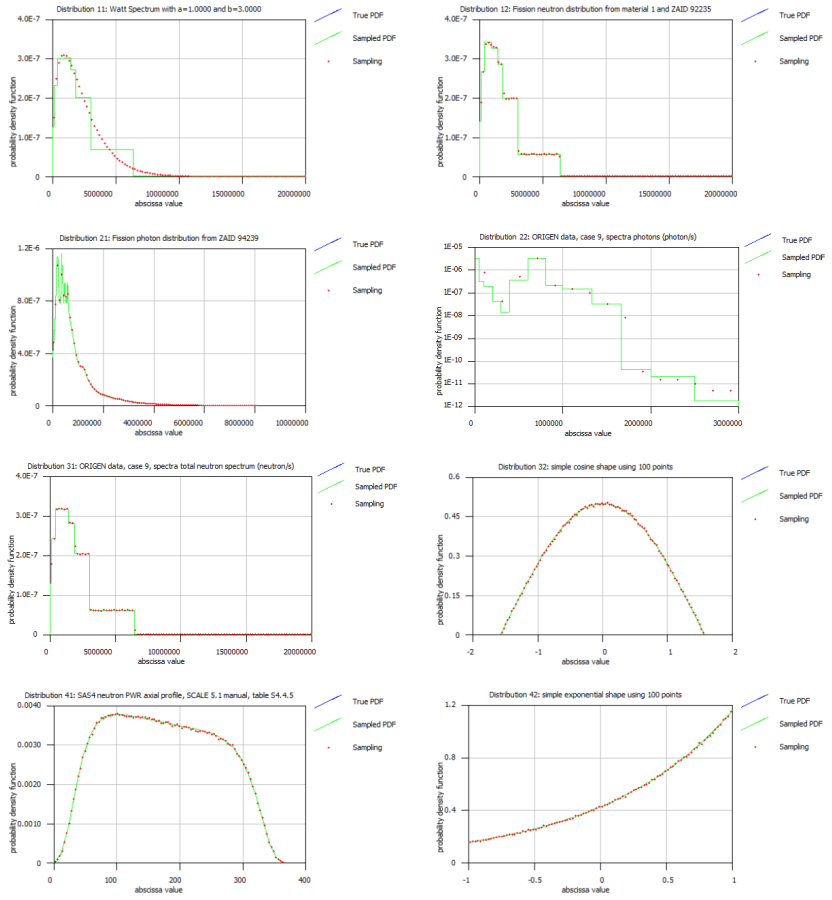


Fig. 8.2.18: Sampling tests for the special (built-in) distribution examples.

Energy boundaries

Energy boundaries ("energyBounds") require an identification number and a specification of a set of bin boundaries in energy (eV). Energy bounds objects are typically used in CE calculations for specifying and energy grid for tallies. The keyword "bounds ... end" can be used to list energy values (in eV, in any order). The keyword "linear *n a b*" can be used to specify *n* bins between *a* and *b*. Likewise, the keyword "logarithmic *n a b*" can be used for *n* bins logarithmically spaced between *a* and *b*. The keywords "bounds", "linear" and "logarithmic" can be used together and multiple times — they will simply add energy boundaries to any already defined. Any duplicate planes will be removed using the absolute tolerance, specified with the keyword "tolerance=". To specify one of the more common SCALE energy structures (handy for doing tallies one a standard structure in CE calculations), one of the following shortcut keywords can be used: "252n", "238n", "200n", "56n", "47p", "44n", "27n", or "19p".

These keywords will cause to load the energy structures from the MG cross-section libraries aliased in the "FileNameAliases.txt" with names of "xn252", "xn238", "xn200", "xn56", "xg47", "xn44", "xn27", and "xg19" relatively. If required energy structure is for neutrons and there is no alias for MG cross-section library or the library is missing, MG JEFF reaction data library will be searched as "n{NG}.reaction.data" to load the energy structure. These can be used in combination with the other keywords to use existing structures supplemented with extra boundaries.

```
energyBounds 1
    title="bounds command, check for duplicates"
    bounds 1 4 2 3 5 end
    bounds 7 6 10 5 9 8 7 end
end energyBounds
energyBounds 3
    title="logarithmic command"
    logarithmic  21  1.0 10000000.0
end energyBounds
energyBounds 11
    title="SCALE 19 group photon structure with extras"
    19p
    linear 10 6.0e6 7.0e6
end energyBounds
```

An energyBounds object can also be used to set the energy bin boundaries for a response (type1) instead of using the "bounds ... end" keyword. This is done by using with the keyword "energyBoundsID=" and referencing a defined energyBounds object. Likewise for distributions, instead of specifying the "abscissa ... end" keyword and listing abscissa values, an energyBounds object can be used. This allows the user to define a set of energy bin boundaries once and re-use them across multiple responses and definitions. When using the "energyBoundsID=" keyword, the data values should be entered in the standard multigroup order — from high energy to low energy. For a stand-alone multigroup Monaco calculation, do not use ID numbers of 1 or 2 for energyBounds objects — these ID numbers are reserved.

Time boundaries

Time boundaries ("timeBounds") are similar to energy bin boundaries but take values in seconds. These objects are only used in tallies in CE calculations.

```
timeBounds 2
    title="linear command"
    linear 10 0.0 10.0e-3
end timeBounds
timeBounds 7
    title="logarithmic command"
    logarithmic  6  1.0e-6 1.0
end timeBounds
```

8.2.3.5 Sources block

The sources block specifies what sources to use. Multiple sources are allowed and each is sampled according to its strength, relative to the total strength of all sources. Each source description must be contained with a "src *id*" and an "end src" (where the *id* is the source identification number). The sources block must contain at least one source.

For each user-defined source, the user can specify the spatial distribution, the energy distribution and the directional distribution separately. Many options for each distribution are available and defaults are used for most if the user does not specify anything. The source strength is set using the keyword "strength=" and the type of source is set using the keyword "neutron" or "photon". The "strength=" keyword is required for each source.

When using more than one source, the user can set the true strength of each using the keyword "strength=" and can also specify how often to sample each source using the keyword "biasedStrength=". The true strengths of the sources will be combined to form the true source distribution PDF. The biased strengths of sources will be combined to form a PDF from which to sample. The weights of the source particles will be properly weighted to account for the biased sampling strengths. For example, consider two sources of strengths 10^9 and 9×10^9 /sec that should be sampled in a ratio of 4:1. The biased sampling strengths are then set to 4 and 1. Monaco will sample the first source 80% of the time and the particles will be born with a weight of 0.125. The second source will be sampled 20% of the time and its particles will be born with weights of 4.5.

Spatial distribution

Keyword	Parameters	Possible degenerate cases
cuboid	x_{max} x_{min} y_{max} y_{min} z_{max} z_{min}	rectangular plane, line, point
xCylinder	*r* x_{max} x_{min}	circular plane, line, point
yCylinder	*r* y_{max} y_{min}	circular plane, line, point
zCylinder	*r* z_{max} z_{min}	circular plane, line, point
xShellCylinder	*r1 r2* x_{max} x_{min}	cyl., planar annulus, cyl. surface, line, ring, point
yShellCylinder	*r1 r2* y_{max} y_{min}	cyl., planar annulus, cyl. surface, line, ring, point
zShellCylinder	*r1 r2* z_{max} z_{min}	cyl., planar annulus, cyl. surface, line, ring, point
sphere	*r*	point
shellSphere	*r1 r2*	sphere, spherical surface, point

Note that other than the shell-type solids, the parameters are the same as the SGGP geometry specification of those solids. The SGGP keyword "origin" (followed by at least one of "x=", "y=", and/or"z=") is available for all of the different source solid bodies. For the cylinder based solid bodies, the direction of the axis of the cylinder can be set by using the keyword "cylinderAxis *u v w*", where *u*, *v*, and *w* are the direction cosines with respect to the global *x*-, *y*-, and *z*-directions.

The source can be limited to only be from the parts of the solid body that are inside a specific unit ("unit="), inside a specific region ("region=") within the specified unit, or made of a certain material ("mixture="). A mixture and a unit/region cannot both be specified since that would either be redundant or mutually exclusive.

If no source spatial information is provided by the user, the default is a point source located at the origin (in global coordinates). Like SGGP input, the geometry keywords used for the bounding shape are fixed lengths arrays and do not have an "end" terminator. They must be followed by the correct number of parameters.

The spatial distribution in each dimension of the cuboid shape is specified by using the keywords "xDistributionID=", "yDistributionID=", or "zDistributionID=" and pointing to a distribution defined in the definitions block. For the cylindrical shapes, "rDistributionID=" and "zDistributionID=" can be used. For spherical shapes, only the "rDistributionID=" can be specified. Distributions defined using abscissa values that are different than the length of the simple geometry bounding shape can still be used if the keyword "xScaleDist" (or "y", "z", or "r") is used. This linearly scales the distribution abscissa values to the length of the simple geometry bounding shape. Note that for cylindrical sources, since the axis can point in any direction, the z distribution is interpreted as the length along the axis, with the base position as z=0.

Energy distribution

"eDistributionID=" and pointing to one of the distributions defined in the definitions block. Energies will be sampled from the distribution in a continuous manner. For MULTIGROUP calculations, that energy will then be mapped onto the group structure of the cross-section library being used by Monaco. Each source should have an energy distribution that has abscissa values in units of eV. If no energy distribution is given, 1 MeV (translated to the current group structure if a multigroup problem) will be used.

To use the total of an energy distribution as the source strength, use the keyword "useNormConst" without either "strength=" or "fissions=". This will set the strength to be equal to the normalization constant of the distribution — the total of the distribution before it was normalized into a pdf. An optional "multiplier=" keyword can be used to increase or decrease that strength. For example, consider a case using the neutron spectrum information from a case of an ORIGEN-S binary concentration file that used a basis of an entire core. If the Monaco source was just one of the 200 assemblies, then the "multiplier=" keyword can be set to 0.005 so that the source strength is scaled appropriately.

Directional distribution

The directional distribution of the source is specified by using the keyword "dDistributionID=" and pointing to one of the distributions defined in the definitions block. The distribution will be used to sample the cosine of the polar angle, μ, from the reference direction. The reference direction, where $\mu = 1$, is set with the keyword "direction *u v w*", where *u*, *v*, and *w* are the direction cosines with respect to the global *x*-, *y*-, and *z*-directions. The default value for the reference direction is the positive *z*-axis (<0,0,1>). The keyword "dScaleDist" can be used to linearly scale the distribution abscissa values to the range of $\mu \in [-1, 1]$. If no directional distribution is specified with the keyword "dDistributionID=", then an isotropic directional distribution will be used.

Using a Monaco mesh source map file

The user can alternatively specify an existing Monaco mesh source map file-a binary file created by a previous MAVRIC or Monaco calculation. The mesh source map must be a binary file using the Monaco mesh source map format (a *.msm file). This option is specified with the "meshSourceFile=" keyword and the file name (and full path if necessary) in quotes.

```
read sources
    src 1
        meshSourceFile="c:\mydocu~1\previouslyMadeSource.msm"
    end src
end sources
```

If the "meshSourceFile=" keyword is used, all energy distribution keywords and most spatial distribution keywords will be ignored. Source keywords that can be used with a mesh source include "strength=" to override the source strength in the mesh source; "biasedStrength=" to set the sampling strength; "origin", "x=", "y=", and "z=" to place the origin of the mesh source file at a particular place in the current global

coordinate system; and the keywords for describing the directional distribution — "dDistributionID=", "direction *u v w*" and "dScaleDist".

Mesh sources are sampled using the following algorithm: First, a direction is sampled. Second, a voxel is sampled and a position is picked uniformly within the voxel. If that position does not match the optional limiters (unit, region, material specified in the mesh source), a new position is chosen within the voxel until a match is made. If a position cannot be found within the voxel after 10000 tries, Monaco will stop. (This can occur if the mesh voxel contained just a sliver of source volume when generated. For this case, the keyword "allowResampling" can be used to select a new voxel instead of stopping. In general, this keyword should not be used.)

Creating a mesh source

To create a mesh source out of the source definition, use the "meshSourceSaver" subblock inside the sources block. It is quite handy to visualize the sources and ensure they are what were intended. You must specify which one of the defined grid geometries to use (keyword "gridGeometryID=") and a filename for the resulting mesh source file (keyword "filename=" with the filename in quotes "*path\name*.msm"). For more than one source, each will be stored separately and the filename will include the source id number.

```
read sources
    src 1
        ...
        ...
    end src
    src 5
        ...
        ...
    end src
    meshSourceSaver
        gridGeometryID=7
        filename="meshSource.msm"
        subcells=3
    end meshSourceSaver
end sources
```

To create the mesh source, Monaco determines if the defined source exists within each cell. This is done by dividing each mesh cell into n × n × n subcells (from the keyword subCells=*n* with a default of *n*=2) and testing each subcell center. For every subcell center that is a valid source position (within the spatial solid and meets the optional unit, region, or mixture requirements), an amount of source proportional to the subcell volume is assigned to the mesh cell. The keyword subCells= can be used to better refine how much source is computed for the mesh cells at the boundary of a curved source region. Of course, more subcell testing takes more time. If a given source is degenerate in any dimension (point, line, or plane), that information will be stored in the resulting mesh source so that particles will not be sampled over the entire corresponding voxel but will have closer to the original spatial distribution. Likewise, if the original source had restrictions based on unit, region or mixture, those restrictions will be stored as part of the resulting mesh source.

The above process may miss small sources or degenerate sources (surfaces, lines, points) that do not lay on the tested subcell centers. If none of the mesh cells contain any source after the subcell method, then random sampling of the source is used. A number of source positions are sampled from the source (set by the "sourceTrials=" keyword, default of 1000000) and then placed into the proper mesh cell. If this method is used, the resulting mesh source file should be visualized to ensure that the statistical nature of the source trials method does not unduly influence the overall mesh source. To skip the subcell method and go directly to the source trials method, use "subCells=0".

The keyword "makeTotal" will make a single mesh source file which is the composite of all of the individual sources. Geometric degeneracies or restrictions to only sample particles from a specified unit, region or

material will only be kept if they are the same for all of the sources. For this reason, users may not wish to use a mesh source using the "makeTotal" keyword for transport but rather use it to verify that all of the sources have been input properly.

The keyword "reduce" can be used to only save the smallest rectangular portion of the mesh surrounding the voxels with non-zero source amounts. This can result in much smaller file sizes for sources that are small compared to the extents of the grid geometry.

Monaco mesh source files (*.msm) can be viewed with the Mesh File Viewer. Plots can be made showing the source values for each group (or total). The viewer can also show the geometry regions or material mixtures as well. Using the viewer is an easy way to confirm that the source definition was entered correctly. Note that the *.msm files actually only store the biased sampling distribution and the initial weights (to speed up the sampling process). So, in the viewer the "true" source is computed as the product of the sampled distribution and the weights. If groups with real source are set to zero importance, the viewer cannot recreate the original true source. The true source shown by the viewer is the amount of true source only in groups that have non-zero importance.

Mesh source advanced features

Two advanced features exist in the meshSourceSaver subblock — mainly used by the MAVRIC sequence when the importance map calculations use a different cross-section library than the final Monaco calculation.

The keyword "sampleFromMesh" can be used to tell Monaco to sample from the created mesh file(s) instead of the standard source definition. This can be useful in determining if the mesh source is fine enough to accurately represent the original source definition. If the "makeTotal" keyword was used, then Monaco will sample from the total mesh source file.

The keyword "meshBiasFile=" can be optionally be used when "sampleFromMesh" is on. This tells Monaco to sample from the mesh source file(s) version of the source definition that has been modified using just the importance information from the named mesh source file. For example, using a 27-group biased mesh source for a Watt spectrum source may not represent the high energy tail very well. In this case, it would be better to do a 200-group Monaco calculation but still use the importance information from 27-group mesh source file using "sampleFromMesh" and "meshBiasFile=".

8.2.3.6 Tallies block

The tallies block tells Monaco what to compute: fluxes at certain points in space (point detectors), fluxes in certain geometry regions, or fluxes in each voxel of a mesh grid. The computed fluxes can also be integrated with response functions to compute dose, reaction rate or some other dose-like quantity. Any number of optional response functions can be evaluated with each tally.

Each tally type begins with a keyword ("pointDetector", "regionTally", or "meshTally") and ends with an "end" and that same keyword. Individual tallies can be listed in any order. Identification numbers for each tally are required and must be positive integers and unique among the tally type. All three of the tally types can have an optional title using the keyword "title=" followed by the title enclosed in double quotes. Tallies should be defined as either a neutron tally or a photon tally.

```
read tallies
    pointDetector 1
        ...
    end pointDetector
    regionTally 9
        ...
```

(continues on next page)

(continued from previous page)

```
    end regionTally
    regionTally 19
        ...
    end regionTally
    meshTally 1
        ...
    end meshTally
end tallies
```

Each tally computes the fluxes in each tally bin and the total flux. For multigroup calculations, the multigroup cross section group structure is used for all tallies. For CE calculations, each tally can use a different "energyBoundsID=", which points to one of the energy bounds defined in the definitions block. CE calculations can also use the keyword "timeBoundsID=" to specify a set of time bin boundaries. For one response function to integrate the fluxes with, the keyword "responseID=" can be used, where the value corresponds to the identification number of one of the response functions defined in the definitions block. For multiple response functions, the keyword array "responseIDs ... end" can be used.

Each tally type can be multiplied by using the "multiplier=" keyword. This is useful for units conversions or other types of scaling. Multiple uses of the multiplier keyword within one tally definition will apply the product of all multipliers to that tally. Using the keyword "multiplier=" inside the tallies block but outside any particular tally will apply that multiplier to all tallies.

Point detector tallies

A point detector tally computes the uncollided and total flux at a given location in space. This tally requires exactly one location and can use any number of optional response functions. The "locationID=" keyword is used to specify one of the locations listed in the definitions block. Point detectors should only use locations that are in void regions of the geometry.

Because point detectors estimate the flux at the location using a ray-trace from every collision site during the life of the particle, they can be quite expensive. For particles very far away from the detector location, the contributions to the tally can be quite small. Point detectors can be made to use the importance of the current particle location/energy to decide whether or not to make a contribution to the tally using the keyword "minSampProb=" and a value such as 0.1 or 0.01. This keyword specifies the minimum sampling probability for a given point detector. As the particle is transported, the probability p of making a contribution to the point detector tally is set by using the current weight of the particle, w, to be

$$p = \left(\frac{w_{\min}}{w}\right)^a \tag{8.2.19}$$

where the power a for each point detector was determined at the start of the simulation using the minimum sampling probability $p_{\min}$ to be

$$a = \frac{\ln\left(p_{\min}\right)}{\ln\left(\frac{w_{\min}}{w_{\max}}\right)} \tag{8.2.20}$$

where the minimum and maximum target weights, $w_{\min}$ and $w_{\min}$, were determined from either the region-based weight targets or the mesh-based importance map weight targets. So, when the current particle weight approaches the minimum target weight of the problem (in very important areas), the point detector contribution is made nearly 100% of the time. When the particle weight approaches the maximum target weight (in very unimportant areas), the contribution to the point detector tally is made with probability $p_{\min}$, saving quite a bit of computer time.

This option should only be used if the point detector location is in the area of high importance. For point detectors in areas of low importance, using this option may severely undersample the point detector where contributions would actually be the most significant, causing an underestimation of the flux. The default value for the minimum sampling probability is $p_{\min} = 1$, giving a default value of $a = 0$ so that the point detector contribution is made at every collision, independent of the current particle weight.

The keyword "minSampProb" used in the tallies block but outside any particular point detector tally specification will be applied to all point detectors.

Each point detector is summarized in the main output file. The uncollided and total flux for each particle type is listed as well as the values for the optional integrated response functions. Along with each of these quantities is a list of the standard deviation of the quantity, the relative uncertainty, the figure-of-merit and a summary of list of the statistical checks (passed or not). Group-by-group values of the fluxes and responses are listed in a separate file named "*outputName*.pd*id*.txt" where *outputName* is the name the user chose for his output file and "*id*" is the identification number corresponding to the point detector tally specification. This file also contains more information about the six statistical checks for flux and each response — their pass/fail values with each batch of simulated particles and their final numerical values at the end of the simulation. A second file called "*outputName*.pd*id*.chart" is also created which can be displayed using the ChartPlot 2D Interactive Plotter, to visually check the convergence behavior of the tally. The total neutron flux, the total photon flux, and the total response function value for each response as a function of batch can be viewed.

Region tallies

A region tally computes both the track-length estimate and the collision density estimate of the flux over a given geometry region (an SGGP "media"). This tally uses the keywords "unit=", "region=", and "mixture=" to limit the tally to one or more of those aspects. For example, "unit=2" and "region=3" are used to specify a region tally for the 3rd media listed for unit 2 of the SGGP geometry input. A mixture and a unit/region cannot both be specified since that would either be redundant or mutually exclusive. If the volume of the region is not given (or calculated) in the SGGP input, then instead of flux, the tally will compute average track length and average collision density.

Each region tally is summarized in the main output file. The total flux for each particle type is listed as well as the values for the optional integrated response functions. Along with each of these quantities is a list of the standard deviation of the quantity, the relative uncertainty, the figure-of-merit and a summary of list of the statistical checks (passed or not). Group-by-group values of the fluxes and responses are listed in a separate file named "*outputName*.rt*id*.txt" where *outputName* is the name the user chose for his output file and "*id*" is the identification number corresponding to the region tally specification. This file also contains more information about the six statistical checks for flux and each response — their pass/fail values with each batch of simulated particles and their final numerical values at the end of the simulation. A second file called "*outputName*.rt*id*.chart" is also created which can be displayed using the ChartPlot 2D Interactive Plotter, to visually check the convergence behavior of the tally. The total neutron flux, the total photon flux, and the total response function value for each response as a function of batch can be viewed.

Mesh tallies

A mesh tally computes the track-length estimate of the flux for every cell in a grid (mesh) geometry. This tally requires exactly one grid geometry or cylindrical geometry and can use any number of optional response functions. The "gridGeometryID=" or "cylGeometryID=" keyword is used to specify one of the mesh geometries listed in the definitions block. The group-by-group flux values, the total flux values and the total for each response are kept in memory during the simulation. Group-by-group contributions to the responses are not tallied during the simulation.

Mesh tallies can be limited to only save contributions to the voxel flux from track lengths through a certain unit, region or material using the keywords "unit=", "region=", and "mixture=". For example, to compute a mesh tally of a reaction rate of a specific isotope, the response function should only be multiplied by the amount of flux in the voxel that resulted in contributions from the material that contained that isotope. In this case, the keyword "mixture=" should be used so that the fluxes in each voxel represent only the flux from that material that holds the desired isotope.

A mesh tally saves the flux for each group, as well as the integrated response functions for each listed response for every cell of the grid geometry to a file called "*outputName*.mt*id*.3dmap" where *outputName* is the name the user chose for his output file and "*id*" is the identification number corresponding to the mesh tally specification. This file contains the group flux values and their absolute uncertainties. If any response functions were specified, then the responses and their uncertainties will be computed and stored in the same file. Monaco mesh tally files can be viewed with the Mesh File Viewer.

With each mesh tally, files are also created with the statistical test information - "*outputName*.mt*id*.flux" and "*outputName*.mt*id*.resp*xx*.txt" where *xx* is the responseID. The statistical tests can be turned off with the keyword "noStatChecks".

The Mesh File Viewer can be used to show the value, uncertainty, or relative uncertainty of any of the group fluxes, total fluxes, or responses tallied. For characterizing the mesh tally, the viewer can be used to display histograms of the relative errors-showing what fraction of the mesh cells had less than some amount of relative uncertainty. Images from the Mesh File Viewer can be saved as *.jpg, *.bmp, *.gif or *.png files or exported to other applications, such as MS Word.

Mesh tally files can become quite large-if the group-by-group fluxes are not important for a given problem, the keyword "noGroupFluxes" can be added to the mesh tally input. Instead of the group fluxes, only the total neutron and total photon fluxes will be written to the mesh tally file. If the group-by-group values for the response functions are required, the keyword "saveRespDetails" can be used to create separate mesh tally files of each response called "*outputName*.mt*id*.resp*id*.3dmap" where the second "*id*" is the response identification number. Note that for CE calculations, these group-by-group results are formed using the response function mapped onto the energy grid of the tally, since separate tallies of response by group are not made during the simulation.

The mesh tally keyword "weightless" will instruct Monaco to not include the particle weight in the contribution to the mesh tally for track lengths that cross the voxels of the mesh. Instead of a flux tally, this will compute the Monte Carlo particle density — a measure of the number of particles simulated by Monaco in each mesh cell and in each energy group.

Mesh tallies can use a cylindrical mesh instead of a rectilinear mesh. Use the keyword "cylGeometryID=" instead of "gridGeometryID=" and reference one of the defined cylindrical meshes defined in the definitions block. Both cannot be specified at the same time. The Java Mesh File Viewer can only show the *r*-*z* view of a cylindrical mesh tally.

8.2.3.7 Parameter Block

The parameter block sets the Monte Carlo parameters used by Monaco. Items can be listed in any order. The initial random number ("randomSeed=") is given as a 16-digit hexadecimal number. The number of histories per batch ("perBatch=") and the number of batches ("batches=") can be specified. After every batch of source particles, the tally files are saved to disk. To prevent long run times, a maximum run time in minutes ("maxMinutes=") can also be specified. Defaults are 10 batches of 1000 histories each, with no time limit. The value of batches is used to allocate arrays for the tally statistical tests — so do not make this overly large, even when using maxMinutes to control termination.

For MULTIGROUP, the particles contained in the library will be transported, unless turned off using the keywords "noNeutron" or "noPhoton." In CE, to prevent loading large amounts of unneeded cross-section information into memory, the user should specify which particles to transport, using the keywords "neutron" and/or "photon." Monaco also supports Doppler pre-broadening of the CE neutron cross sections. This is controlled by the "dopplerBroaden=" parameter. Integer options are 0 (disabled, default), 1 (broaden 1D cross sections only), 2 (broaden 1D and 2D cross sections), and 3 (broaden 2D cross sections normally and broaden 1D cross sections using a less robust, but faster, interpolation method).

The default behavior for Monaco is to create neutrons from fission events and create secondary gammas from neutron collisions. To turn off the creation of fission neutrons in all multiplying media (for example, when the source already includes them), use the keyword "fissionMult=0". For problems where the library has photon data but none of the tallies require photons, use the keyword "secondaryMult=0" to stop the creation of secondary photons from neutrons. In CE problems, the number of fission neutrons and secondary gammas can be simulated as one particle of each type having a high weight ("fissionMult=1", "secondaryMult=1") or as many particles as the physical yield for the reaction would dictate ("secondaryMult=2"), each with a weight corresponding to their probability of emission. If the user requests option 2 and the particle bank becomes too large, Monaco will drop the settings back to option 1. In MULTIGROUP, values of 1 and 2 are both treated as 1.

```
read parameters
   randomSeed=003ecd7b4e3e8b
   perBatch=100000  batches=100  maxMinutes=1440.0
   fissionMult=0
end parameters
```

In complex geometries, particles can sometimes "get lost" due to round-off errors in the ray-tracing. This would normally result in the code stopping, since lost particles usually indicate an undefined or doubly defined region of space. In order to tolerate a few lost particles without stopping the code, use the keyword "maximumLost=". Care should be taken not to increase this just to get around poorly defined geometry. To aid in geometry testing, the keyword "voidAllRegions" can be used. This keyword sets every region material to void so that tracks can stream through without interacting (faster). A large source and this keyword can be used to test a geometry input for gaps and overlaps. When the "voidAllRegions" keyword is used, mesh tally files will not contain material information, only the unit and region information.

Note that for both CE and MULTIGROUP, using fissionMult=0 only turns off the creation of neutrons from fission. The fission photons are controlled by the secondaryMult setting because some of the ENDF data evaluations do not separate fission gammas from other neutron collision gammas.When using a mesh-based importance map, if a particle is outside the importance map, the code will stop with the message "Could not find particle importance. The particle is outside of the importance map." If the importance map does not cover the entire problem, then the tallies may be missing part of their final values. If the user intends to use an importance map that does not cover the entire geometry and wants particles outside the importance map to have zero importance (they will then be killed), then the keyword "allowShortImpMap" can be used to

allow the use of a "short" importance map. Users must be sure that areas outside the importance map are unimportant to the problem.

8.2.3.8 Biasing Block

The optional biasing block lists the parameters for the standard Monaco variance reduction tools: forced collisions, region-based weight windows, and path-length stretching. This block also allows for the use of a previously made Monaco mesh importance map, such as those produced by the MAVRIC sequence.

Forced collisions

Forced collisions are one of the simplest variance reduction techniques. This makes a particle have a collision along its current flight direction before leaving the geometry. The collision is forced and the particle weight is reduced by the true probability of having a collision within the geometry. This is helpful in small or low-density geometries where many particles leave without interacting but can add computation time to ordinary problems. To use forced collisions, specify the "forcedCollisions" keyword. This requires the use of Russian roulette ("targetWeights" and "lowerWeights", see below).

Weight windows

Monaco can use Russian roulette for preventing low particle weights from being followed and splitting to prevent the production of very high weight particles. Either of these requires the target weight values ("targetWeights ... end") for each energy group and for each region to be listed. For CE calculations, the energy bin boundaries are specified with the keywords "nEnergyBoundsID=" and "pEnergyBoundsID=". For Russian roulette, the lower weight bounds must be specified ("lowerWeights ... end") and for splitting the upper weight bounds are listed ("upperWeights ... end"). The different weight arrays can have a length matching (1) the product of the number of energy groups and the number of regions, (2) the number of regions, or (3) the number of energy groups. In case 2, the values are repeated for each energy group. In case 3, the values are repeated for each region. For example, to specify only Russian roulette in a coupled neutron-photon problem with target and lower weights the same in each region the following is used. An example using the 27/19 multigroup library would be

```
read biasing
    targetWeights 27r1.0 19r0.1  end
    lowerWeights  27r0.1 19r0.01 end
end biasing
```

and for a CE calculation

```
read definitions
    energyBounds 1 27n end energyBounds
    energyBounds 2 19p end energyBounds
end definitions

read biasing
    nEnergyBoundsID=1  pEnergyBoundsID=2
    targetWeights 27r1.0 19r0.1  end
    lowerWeights  27r0.1 19r0.01 end
end biasing
```

Alternatively, to use Russian roulette and splitting, the target weights and a window ratio ("windowRatio=") can be specified. The window ratio is simply the ratio of the weight window upper bound to the weight window lower bound, with the target weight being the average of the upper and lower. If target weights $\overline{w}$ and a window ratio r are supplied, then the lower and upper weight bounds are found by using

$$w_{\min} = \frac{2}{r+1}\overline{w} \tag{8.2.21}$$

$$w_{\max} = \frac{2r}{r+1}\overline{w} \tag{8.2.22}$$

If only the window ratio is supplied, both Russian roulette and splitting will be turned on with the target weights for every energy group and every region set to 1.

A target weight of 0 will prevent particles of that energy group in that region from being transported. For example, to perform a neutron-only calculation using a coupled neutron-photon library, simply set the target weight values for all of the photon groups in every region to 0. The user should be careful not to "turn off" energy groups or regions that may impact (bias incorrectly) the final tally results.

Monaco always uses the implicit capture technique-at collision sites absorption is not simulated but instead the particle weight is reduced by the ratio of the scatter probability to the total interaction probability. Particles only stop if they escape the defined geometry. This generally produces tally results with lower uncertainties in less time, but for highly scattering or very large geometries, particles with very low weights will be tracked until their weight reaches the lower limit of real numbers in double precision. This is not typically what the user wants. So, for problems that are not using any weight windows or importance map, Russian roulette and splitting are automatically turned on using the a target weight of 1 for every energy group and every region and a window ratio of 5.

Path-length stretching

Path-length stretching allows particles going a certain direction to travel farther (with reduced weight) before interacting. Seventeen different directions are available, as listed in the Table 8.2.6. One of the directions is specified by using the "direction=" keyword and one of the direction strings listed in Table 8.2.10, in quotes. The amount of stretching is specified using the "pathStretch ... end" array, with values between 0 (no stretching) and 1 (lots of stretching), for each energy group and region. Items can be listed in any order. Similar to the weight window arrays, the "pathStretch" array can have a length matching (1) the product of the number of energy groups and the number of regions, (2) the number of regions, or (3) the number of energy groups. Values are repeated to fill in all of the regions and groups.

```
read biasing
    pathStretch  46r0.5 46r0.6 46r0.75 end
    direction="localZp"
end biasing
```

When stretching toward a given location (direction="location"), then the location ID number must be specified using the "locationID=" keyword. For CE calculations, energy boundary objects need to be defined using the keywords "nEnergyBoundsID=" and "pEnergyBoundsID=".

Table 8.2.6: Directions available for path-length stretching

Direction	Comment
`localXp`	+x direction in local coordinates
`localYp`	+y direction in local coordinates
`localZp`	+z direction in local coordinates
`localXm`	-x direction in local coordinates
`localYm`	-y direction in local coordinates
`localZm`	-z direction in local coordinates
`localYZ`	cylindrically away from x-axis (local)
`localXZ`	cylindrically away from y-axis (local)
`localXY`	cylindrically away from z-axis (local)

Table 8.2.6 – continued from previous page

Direction	Comment
	`globalXp` | +x direction in global coordinates
`globalYp`	+y direction in
`globalZp`	+z direction in
`globalXm`	-x direction in g
`globalYm`	-y direction in g
`globalZm`	-z direction in g
`outward`	spherically outw
`location`	in the direction

Mesh-based importance map

The user can alternatively specify an existing Monaco mesh-based importance map-a binary file created by a previous MAVRIC calculation. The mesh importance map must be a binary file using the Monaco mesh importance map format (a *.mim file). This option is specified with the "meshImpMapFile=" keyword and the file name (and full path if necessary) in quotes.

```
read biasing
    meshImpMapFile="c:\mydocu~1\previouslyMadeImpMap.mim"
    windowRatio=10.0
end biasing
```

If the "meshImpMapFile=" keyword is used, most other biasing block keywords cannot be used. The keyword "windowRatio=" can be used and its default value is five.

If particles leave the importance map but are still in the defined geometry, the simulation will be stopped. If the user wants to allow importance maps that do not cover the entire problem, the keyword "allowShortImpMap" should be used in the parameters block. In that case, areas outside the mesh importance map will be treated as completely unimportant-particles will be killed outside the mesh.

Note that for the most effective use of an importance map, the source should be biased to match. This is what the MAVRIC sequence does — it produces a biased source such that sampled particles are born with a weight matching the target weight of the importance map.

Monaco input summary

Below are summaries of the Monaco blocks and their available keywords.

Table 8.2.7: Keywords for the cross section block

block	keyword	type	length	default	required	restrictions/comments
read crossSections						
	maxNumMixtures=	integer		100	no	only required if more than 100 mixtures
	maxNumElements=	integer		100	no	only required if more than 100 elements in any mixture
	For multi-group calculations					
	ampxFileUnit=	integer			a*	unit number of AMPX working library
	iceFileUnit=	integer			b*	unit number of ICE cross sections
						*required: either a) ampxFileUnit= or b) iceFileUnit=
	For continuous energy calculations					
	ceLibrary=	character			yes	required for CE, contained in double quotes
	ceTempDefault=	real		293	no	set default temperature for CE mixtures
	mixture *id*	integer			yes	non-negative integer
	element	integer and real			yes	integer=z*1000+a, real=atom density
	temperature=	real		293	no	temperature for this mixture (CE only)
	end mixture					
	For multi-group calculations					
	printTotals		not present		no	prints total cross sections for each mixture
	printScatters		not present		no	prints 2D cross sections for each mixture
	printAngleProb		not present		no	print angles and probabilities for each mixture
	printFissionChi		not present		no	print fission spectrum chi for each mixture
	printExtra		not present		no	print extra 1D cross sections for each mixture
	printLegendre		not present		no	print Legendre coefficients for each mixture
	fullyCoupled		not present		no	treat all particles as primary particles
end crossSections						

Table 8.2.8: Keywords for the definitions block

block	keyword	type	length	default	required	restrictions/comments
read definitions						
	location *id*	integer			yes	non-negative integer, unique among positions
	title=	character		none	no	contained in double quotes (title="yada yada yada")
	position	real	3		yes	global coordinates (x,y,z)
	end location					
	Type 1: generic user-defined response (n values is histogram, n+1 values is value/function pairs)					
	response id	integer			yes	non-negative integer, unique among responses
	neutron/photon				yes	indicate either neutron or photon for the response type
	bounds	real	n+1	none	yes*	energy bin boundaries (eV)
	energyBoundsID=	integer			yes*	use a defined energyBounds objects for bounds *either bounds or energyBoundsID is required
	values	real	n or n+1		yes	non-negative real numbers
	end response					
	Type 2: Using a cross-section from the library (usually with units of barns)					
	response id	integer			yes	non-negative integer, unique among responses
	neutron/photon				yes	indicate either neutron or photon for the response type
	material=	integer			yes	which material
	ZAID=	integer			no	which isotope (ZAID=Z*1000+A) (omit to use cross section of entire material)
	MT=	integer			yes	which MT reaction number
	macro			not present	no	convert units from barns to /cm using ZAID atom density
	end response					
	Type 3: Flux-to-Dose conversion factors					
	response id	integer			yes	non-negative integer, unique among responses
	specialDose=	integer			yes	special dose factor from a MG library
	end response					
	response id	integer			yes	non-negative integer, unique among responses
	doseData=	integer			yes	generate data points from original flux-to-dose reference
	noExtrapolation			not present	no	do not use SCALE energy extrapolation
	end response					
	Type 4: Standard multi-group response function (MG only)					
	response *id*	integer			yes	non-negative integer, unique among responses
	neutron/photon				yes	indicate either neutron or photon for the response type
	values	real	ng		yes	non-negative real numbers
	end response					
	Optional keywords for any type of response					
	response *id*	integer			yes	non-negative integer, unique among responses
	title=	character	256	none	no	contained in double quotes (title="yada yada yada")
	multiplier=	real		1.0	no	multiplier for units conversion, etc.
	makeChart			not present	no	makes a *.chart file showing the input values and the mutli-group representation (for MG)
	eHigh=	real			no	keep only responses values below eHigh (eV)
	eLow=	real			no	keep only responses values above eLow (eV)
	lessOutput			not present	no	only show first 5 and last 5 points of response
	end response					
	makeCharts			not present	no	makes a *.chart file for each response
	multiplier=	real		1.0	no	multiplier applied to all defined responses
	lessOutput			not present	no	only show first 5 and last 5 points of response
end definitions						

ng is the number of groups in the multi-group cross section library for that type of particle

Table 8.2.9: More keywords for the definitions block

block	keyword	type	length	default	required	restrictions/comments
read definitions						
	gridGeometry *id*				yes	non-negative integer, unique among grid geometries
	title=	character	256	none	no	contained in double quotes (title="yada yada yada")
	xplanes	real	any		no	a list of real numbers (order not important) in *x* dimension
	xLinear n min max	int real real			no	adds *n*+1 planes evenly from min to max in *x* dimension
	yplanes	real	any		no	a list of real numbers (order not important) in *y* dimension
	yLinear n min max	int real real			no	adds *n*+1 planes evenly from min to max in *y* dimension
	zplanes	real	any		no	a list of real numbers (order not important) in *z* dimension
	zLinear n min max	int real real			no	adds *n*+1 planes evenly from min to max in *z* dimension
	tolerance=	real		1.E-06	no	for removing duplicate planes from *planes or *Linear
	make3dmap			not present	no	makes a *.3dmap file showing the grid geometry
	xdivide=	integer		1	no	Once all xplanes are entered, further divide them by this
	ydivide=	integer		1	no	Once all yplanes are entered, further divide them by this
	zdivide=	integer		1	no	Once all zplanes are entered, further divide them by this
	end gridGeometry					
	make3dmaps			not present	no	makes a *.3dmap file for each of the grid geometries
	cylGeometry *id*	integer			yes	non-negative integer, unique among cylindrical geometries
	title=	character	256	none	no	contained in double quotes (title="yada yada yada")
	radii	real	any		no	list of radii
	thetas	real	any		no	list of angles (0 to 2_{π})
	degrees	real	any		no	list of angles (0 to 360°)
	zPlanes	real	any		no	list of zplanes (relative to position where z=0)
	radiusLinear *n min max*	int real real			no	adds *n*+1 radii evenly from min to max
	thetaLinear *n min max*	int real real			no	adds *n*+1 theta planes evenly from min to max (0 to 2π)
	degreeLinear *n min max*	int real real			no	adds *n*+1 theta planes evenly from min to max (0 to 360°)
	zLinear *n min max*	int real real			no	adds *n*+1 *z* -planes evenly from min to max
	position	real	3	(0, 0, 0)	no	position from which to start radii and zplanes
	zaxis *u v w*	real	3	<0, 0, 1>	no	set the axis of the cylindrical mesh
	xaxis *u v w*	real	3	<1, 0, 0>	no	set the reference position of the cylindrical mesh
	tolerance=	real		1.E-06	no	for removing duplicate planes from *planes or *Linear
	makeCylMap			not present	no	makes a *.3dmap file showing the cylindrical mesh
	end cylGeometry					
	makeCylMaps			not present	no	makes a *.3dmap file for each of the cylindrical geometries
end definitions						

Table 8.2.10: Even more keywords for the definitions block

block	keyword	type	length	default	required	restrictions/comments
read definitions						
	For a histogram-type distribution of n bins					
	distribution *id*	integer			yes	non-negative integer, unique among responses
	title=	character	256		no	contained in double quotes (title="yada yada yada")
	abscissa	real	n+1		yes*	bin boundaries
	truePDF	real	n			binned values of the true probability distribution function
	trueCDF	real	n			binned values of the true continuous distribution function
	biasedPDF	real	n		see text	binned values of the biased probability distribution function
	biasedCDF	real	n			binned values of the biased continuous distribution function
	weight	real	n			suggested weight for each bin
	importance	real	n			suggested importance value for each bin
	runSampleTest			not present	no	run a sampling test for this distribution (makes a *.chart file)
	end distribution					
	For distributions consisting of value/function pairs over n intervals					
	distribution *id*	integer			yes	non-negative integer, unique among responses
	title=	character	256		no	contained in double quotes (title="yada yada yada")
	abscissa	real	n+1		yes	abscissa values
	truePDF	real	n+1			point values of the true probability distribution function
	biasedPDF	real	n+1		see text	point values of the biased probability distribution function
	weight	real	n+1			point values of the suggested weights
	importance	real	n+1			point values of the suggested importance
	runSampleTest			not present	no	run a sampling test for this distributions (makes *.chart file)
	end distribution					
	For histograms or value/function pairs					
	distribution *id*					
	Use a defined energyBounds objects for the abscissa values					
	energyBoundsID=	integer			yes*	use boundaries from a defined energyBounds as abscissa
	For multi-group calculations					
	neutronGroups/photonGroups				yes*	use neutron/photon energy bin boundaries as abscissa value *required: abscissa, energyBoundsID, or neutron/photonGrou If not abscissa, n must match number of groups
	end distribution					
	For a list on n discrete values					
	distribution *id*	integer			yes	non-negative integer, unique among responses
	title=	character	256		no	contained in double quotes (title="yada yada yada")
	discrete	real	n		yes	discrete values
	truePDF	real	n			true probabilities
	trueCDF	real	n			true cumulative probabilities
	biasedPDF	real	n		see text	biased probabilities
	biasedCDF	real	n			biased cumulative probabilites
	weight	real	n			suggested weight for each bin
	importance	real	n			suggested importance for each value
	runSampleTest			not present	no	run a sampling test for this distribution (makes a *.chart file)
	end distribution					
end definitions						

Table 8.2.11: Special distribution keywords for the definitions block

block	keyword	type	length	default	required	restrictions/comments
read definitions						
	For special and built-in distributions					
	distribution *id*					
	special=	character	30		yes	"wattSpectrum", "cosine", "exponential", "pwrNeutronAxialProfile", "pwrGammaAxialProfile", "pwrNeutronAxialProfileReverse", "pwrGammaAxialProfileReverse", "fissionPhotons", "origensBinaryConcentrationFile", "fissionNeutrons"
	parameters	real	<=10		depends	each type of distribution could have parameters
	filename=	string		none	depends	legal file name for current system, in quotes required for special="origensBinaryConcentrationFile"
	runSampleTest			not present	no	run a sampling test for this distributions (makes *.chart file)
	end distribution					
	runSampleTests			not present	no	show all of the distributions and a sampling test of each
end definitions						

Table 8.2.12: Continuous energy keywords for the definitions block

block	keyword	type	length	default	required	restrictions/comments
read definitions						
	energyBounds *id*	integer			yes	non-negative integer, unique among energyBounds objects
	title=	character	256	none	no	contained in double quotes (title="yada yada yada")
	bounds	real	any		no	a list of energy bin boundaries (in eV, order not important)
	linear *n min max*	int real real			no	adds *n*+1 boundaries (eV) evenly from min to max
	logarithmic *n min max*	int real real			no	adds *n*+1 boundaries (eV) logarithmically from min to max
	tolerance=	real		1.E-06	no	for removing duplicate boundaries from various commands
	252n			not present	no	add energy bin boundaries from 252-group neutron library
	238n			not present	no	add energy bin boundaries from 238-group neutron library
	200n			not present	no	add energy bin boundaries from 200-group neutron library
	56n			not present	no	add energy bin boundaries from 56-group neutron library
	47p			not present	no	add energy bin boundaries from 47-group photon library
	44n			not present	no	add energy bin boundaries from 44-group neutron library
	27n			not present	no	add energy bin boundaries from 27-group neutron library
	19p			not present	no	add energy bin boundaries from 19-group photon library
	end energyBounds					
	timeBounds *id*	integer			yes	non-negative integer, unique among timeBounds objects
	title=	character	256	none	no	contained in double quotes (title="yada yada yada")
	bounds	real	any		no	a list of time bin boundaries (in sec, order not important)
	linear *n min max*	int real real			no	adds *n*+1 boundaries (sec) evenly from min to max
	logarithmic *n min max*	int real real			no	adds *n*+1 boundaries (sec) logarithmically from min to max
	tolerance=	real		1.E-10	no	for removing duplicate boundaries from various commands
	end timeBounds					
end definitions						

Table 8.2.13: Keywords for the sources block

block	keyword	type	length	default	required	restrictions/comments
read sources						
	Defining a source					
	src id	integer			yes	non-negative integer, unique among src objects
	title=	character	256	none	no	contained in double quotes (title="yada yada yada")
	neutron/photon				yes	indicate either neutron or photon for a source particle type
	strength=	real			yes	total source strength (particles/sec)
	biasedStrength=	real		0.0	no	strength to use for sampling this source
	(spatial dependence: at most one solid)					
	cuboid	real	6		no	parameters: x_{max} x_{min} y_{max} y_{min} z_{max} z_{min}
	xCylinder	real	3		no	parameters: r x_{max} x_{min}
	yCylinder	real	3		no	parameters: r y_{max} y_{min}
	zCylinder	real	3		no	parameters: r z_{max} z_{min}
	xShellCylinder	real	4		no	parameters: r_1 r_2 x_{max} x_{min}
	yShellCylinder	real	4		no	parameters: r_1 r_2 y_{max} y_{min}
	zShellCylinder	real	4		no	parameters: r_1 r_2 z_{max} z_{min}
	sphere	real	1		no	parameters: r
	shellSphere	real	2		no	parameters: r_1 r_2
	origin				no	change center point of geometry by adding x,y,z
	x=	real		0.0	no	
	y=	real		0.0	no	
	z=	real		0.0	no	
	cylinderAxis *u v w*	real	3	<0,0,1>	no	direction <*u*,*v*,*w*> in global coordinates of cylinder axis
	unit=	integer		-1	no	limit source to a specific unit
	region=	integer		-1	no	limit source to a specific region of a unit
	mixture=	integer		-1	no	limit source to a specific mixture
	xDistributionID=	integer		none	no	distribution for *x* component of cuboid
	xScaleDist			not present	no	scale the distribution to match *x* length
	yDistributionID=	integer		none	no	distribution for *y* component of cuboid
	yScaleDist			not present	no	scale the distribution to match *y* length
	zDistributionID=	integer		none	no	distribution for *z* comp. of cuboid or axial comp. of cylinder
	zScaleDist			not present	no	scale the distribution to match *z* length
	rDistributionID=	integer		none	no	distribution for *r* component of cylinder or sphere
	rScaleDist			not present	no	scale the distribution to match *r* length
	(energy dependence)					
	eDistributionID=	integer		none	no	distribution to use for energy sampling
	useNormConst			not present	no	set the source strength with the normalization constant of the energy distribution (do not use strength= with this)
	(directional dependence)					
	direction *u v w*	real	3	<0,0,1>	no	reference direction <*u*,*v*,*w*> in global coordinates
	dDistributionID=			none	no	distribution to use for energy sampling
	dScaleDist			not present	no	scale the distribution to match [-1,1] range
	(spatial & energy dependence together)					
	meshSourceFile=	string		not present	no	legal file name for current system, in quotes
	allowResampling			not present	no	see manual
	end src					
end sources						

Table 8.2.14: More keywords for the sources block

block	keyword	type	length	default	required	restrictions/comments
read sources						
	Define source(s)					
	src *id*					
	…					
	end src					
	src *id*					
	…					
	end src					
	Saving the defined source(s) as a mesh source file					
	meshSourceSaver					
	gridGeometryID=	integer			yes	matches one of the id numbers from gridGeometries
	filename=	string		"source.msm"	no	legal file name for current system, in quotes
	subCells=	integer		2	no	subcells per cell (each dimension)
	sourceTrials=	integer		1000000	no	how many source particles to sample
	reduce			not present	no	stores only the cuboid around the voxels with source
	makeTotal			not present	no	save a mesh version of the total source (if >1 source)
	For continuous energy calculations					
	nEnergyBoundsID=	integer		not present	yes*	neutron energy bins for CE-Monaco calculations
	pEnergyBoundsID=	integer		not present	yes*	photon energy bins for CE-Monaco calculations
	Advanced Options					
	sampleFromMesh			not present	no	replace the above source(s) with this mesh source
	meshBiasFile=	string		none	no	use the biasing info in a different mesh source
	end meshSourceSaver					
end sources						

Table 8.2.15: Keywords for the tallies block

block	keyword	type	length	default	required	restrictions/comments
read tallies						
	pointDetector *id*				yes	non-negative integer, unique among pointDetectors
	locationID=	integer			yes	matches one of the id numbers from locations
	minSampProb=	real		1.0	no	minimum sampling probability for pd calculation
	end pointDetector					
	regionTally *id*				yes	non-negative integer, unique among regionTallys
	unit=	integer		-1	see text	limit tally to a specific unit
	region=	integer		-1		limit tally to a specific region of a unit
	mixture=	integer		-1		limit tally to a specific mixture
	end regionTally					
	meshTally *id*				yes	non-negative integer, unique among meshTallys
	gridGeometryID=	integer			a*	matches one of the id numbers from gridGeometries
	cylGeometryID=	integer			b*	matches one of the id numbers from cylGeometries
						*required: either a) gridGeometryID= or b) cylGeometryID=
	unit=	integer		-1	see text	limit tally to a specific unit
	region=	integer		-1		limit tally to a specific region of a unit
	mixture=	integer		-1		limit tally to a specific mixture
	noGroupFluxes			not present	no	do not write group fluxes to mesh tally file
	noStatChecks			not present	no	do not perform statistical checks
	saveRespDetails			not present	no	save each response (group by group) to a separate file
	weightless			not present	no	without weight, this becomes an MC particle flux tally
	end meshTally					
	For all three tally types					
	****Tally *id*					
	title=	character	256	none	no	contained in double quotes (title="yada yada yada")
	neutron/photon				yes	indicate either neutron or photon for a tally type
	responseID=	integer		none	no	matches one of the id numbers from responses
	responseIDs	integer	any	none	no	list of id numbers from responses
	multiplier=	real		1.0	no	multiplier for units conversion, etc.
	For continuous energy calculations					
	energyBoundsID=	integer			no	default is one energy bin 0 to 100 MeV
	timeBoundsID=	integer			no	default is one time bin 0 to ?? Seconds
	end ****Tally					
	multiplier=	real		1.0	no	multiplier for all tallies (mulitplies individual multipliers)
	minSampProb=	real		1.0	no	minimum sampling probability for all point detectors
end tallies						

Table 8.2.16: Keywords for the parameters block

block	keyword	type	length	default	required	restrictions/comments
read parameters						
	randomSeed=	hex		0000000100000001	no	hexidecimal number
	perBatch=	integer		1000	no	positive integer
	batches=	integer		10	no	positive integer
	maxMinutes=	real		0.0	no	non-negative real number
	maximumLost=	integer		10	no	maximum number of lost particles to tolerate
	voidAllRegions			not present	no	replace all materials with voids for geometry testing
	allowShortImpMap			not present	no	particles outside importance map are killed
	noCheckAtBirth			not present	no	do not check the weight of source particles
	Particles to simulate					
	neutron			not present	no	For MG, the particles contained in the library will be
	photon			not present	no	transported unless turned off. In CE, the user should
	noNeutron			not present	no	specify which particles to transport.
	noPhoton			not present	no	
	fissionMult=	integer		1	no	production of fission neutrons (0-no, 1-one)
	secondaryMult=	integer		2	no	production of gammas form neutrons (0-no, 1-one, 2-multiple)
	For continuous energy calculations					
	nMinEnergy	real		1.0E-05	no	minimum energy (eV) for neutrons
	pMinEnergy	real		1.0E+04	no	minimum energy (eV) for photons
	nMaxEnergy	real		2.0E+07	no	maximum energy (eV) for neutrons
	pMaxEnergy	real		2.0E+07	no	maximum energy (eV) for photons
	nMaxAge	real		1	no	maximum age (sec) for neutrons
	pMaxAge	real		1	no	maximum age (sec) for photons
	noPTables			not present	no	turn off the use of probability tables
	db_xs_mode	integer		0	no	Doppler pre-broadening of cross sections (0-none, 1-1D cross sections, 2-1D and 2D cross sections)
	Legacy keywords					
	noFissions			not present	no	no production of fission neutrons (fissionMult=0)
	noSecondaries			not present	no	no production of gammas from neutrons (secondaryMult=0)
end parameters						

Table 8.2.17: Keywords for the biasing block

block	keyword	type	length	default	required	restrictions/comments
read biasing						
	Standard biasing techniques					
	forcedCollisions			not present	no	force particle to interact within geometry
	targetWeights	real	a*	none	no	non-negative real numbers
	lowerWeights	real	a*	none	no	non-negative real numbers
	upperWeights	real	a*	none	no	non-negative real numbers
	windowRatio=	real		none	no	real number greater than one
	pathStretch	real	a*	none	no	valid range of values: [0.0,1.0)
	direction=	string		none	no	one of 17 strings, in quotes
	locationID=	integer		none		required for direction="location"
						*a is one of {ng*nr, nr, ng}
	mapMultiplier=	real		1.0	no	multiply *Weights by this factor
	For continuous energy					
	nEnergyBoundsID=	integer		not present	no	neutron energy bins for CE-Monaco calculations
	pEnergyBoundsID=	integer		not present	no	photon energy bins for CE-Monaco calculations
	Using a Monaco mesh importance map file					
	meshImpMapFile=	string		not present	no	legal file name for current system, in quotes
	mapMultiplier=	real		1.0	no	multiply targetWeights in imp. Map
end biasing						

ng is the number of groups in the MG cross section library or in the energyBounds specified for CE
nr is the number of regions in the geometry

8.2.4 MONACO OUTPUT

8.2.4.1 Main text output file

The Monaco output file first reviews the input Monaco received. First is a review of the geometry-showing which materials are used in each region and the volume of that region, if input or calculated. Then there is a detailed list of other Monaco input: cross-section parameters, data definitions, the sources, the tallies, the Monte Carlo parameters, and the biasing parameters. For calculations using an importance map, its summary is also given. The "Mesh Importance Map Characterization" shows where the importance map may be changing too fast and may require more refinement.

For each batch of source particles simulated, the output file lists the batch time and the starting random number for the next batch, which may be useful in rerunning just a portion of a problem. Once all of the batches are completed, a list of the various tally files that have been created is given. Finally, the tallies are summarized in a section titled "Final Tally Results Summary". For each point detector, the total neutron and photon fluxes (uncollided and total) are given as well as the final response values for each response function. For each region tally, the total neutron and photon fluxes (both track-length and collision density estimates) are listed, followed by the final response values for each response function. Along with each of the final quantities are the standard deviation of the quantity, the relative uncertainty, the figure-of-merit and a summary of list of the statistical checks (passed or not).

8.2.4.2 Tally files

In addition to the summary of tallies contained in the Monaco text output file, many other files are created containing the group-by-group details of the final tally data. Each mesh tally produces a file "*outputName*.mt*id*.3dmap" where *outputName* is the name the user chose for his output file and "*id*" is the identification number corresponding to the tally specification. This file can be viewed using the Mesh File Viewer capability of Fulcrum. Point detector tallies and region tallies each create files "*outputName*.pd*id*.txt" or "*outputName*.rt*id*.txt" to list the group-by-group results. They also produce chart files, "*outputName*.pd*id*.chart" or "*outputName*.rt*id*.chart", which contain the total neutron flux, the total photon flux, and the total response function value calculated at the end of each batch. This data can be used to look at tally convergence and can be viewed with the Interactive Plotter capabilities of Fulcrum. Table 8.2.18 lists the output files, based on the name of the main output file (here called *outputName)*, that are available to the user. These files will be copied back to the directory where SCALE was executed.

8.2.4.3 Diagnostic files

Three of the data types defined in the definitions block can create files or add output to the main text output file to allow the user to ensure items were interpreted by Monaco as they were intended. Responses that included the "makeChart" keyword will each produce a file "*outputName*.resp*id*.chart", where "*id*" is the response identification number, which can be displayed using the Interactive Plotter. Grid geometries that include the keyword "make3dmap" will each produce a file called "*outputName*.grid*id*.3dmap", where "*id*" is the grid geometry identification number, which can be visualized using the Mesh File Viewer. Likewise, cylindrical geometries with the keyword "makeCylMap" will create a file called "*outputName*.cyl*id*.3dmap". Distributions using the "runSampleTest" keyword will produce a file "*outputName*.dist*id*.chart", where "*id*" is the response identification number. The "runSampleTest" results will also be displayed in the main text output with each distribution listed in the Monaco input review.

Table 8.2.18: Output files created by Monaco

File Type	Filename	Viewer	Description
Output Summary			
	outputName.out		main text output file, contains results summary
Diagnostic Files			
	outputName.resp*id*.chart	P	response input and MULTIGROUP representation for response *id*
	outputName.grid*id*.3dmap	V	mesh version of geometry using grid geometry *id*
	outputName.cyl*id*.3dmap	V	mesh version of geometry using cylindrical geometry *id*
	outputName.dist*id*.chart	P	distribution input and sampling test for distribution *id*
Mesh Source Saver			
	filename.msm	V	mesh representation of a single source or total source
	filename.id.msm	V	mesh representation of multiple sources
	filename.sampling.msm	V	biased representation of a single source or total source
	*filename.*sampling.*id*.msm	V	biased representation of multiple sources
Tally Files			
	outputName.pd*id*.txt		detailed results for point detector tally *id*
	outputName.pd*id*.chart	P	batch convergence data for point detector tally *id*
	outputName.rt*id*.txt		detailed results for region tally *id*
	outputName.rt*id*.chart	P	batch convergence data for region tally *id*
	outputName.mt*id*.3dmap	V	mesh tally for meshTally *id*
	outputName.mt*id*.resp*xx*.3dmap	V	mesh tally of response by group for meshTally *id*, response *xx*
	outputName.mt*id*.flux.txt		detailed results for the group-wise flux of meshTally *id*
	outputName.mt*id*.tfluxtxt		detailed results for total flux of meshTally *id*
	outputName.mt*id*.resp*xx*.txt		detailed results for response *xx* of meshTally *id*

continues on next page

Table 8.2.18 – continued from previous page

File Type	Filename	Viewer	Description
V - can be displayed with the Mesh File Viewer capabilities Fulcrum. P - can be displayed with the 2D plotting capabilities of Fulcrum.			

8.2.4.4 Mesh source saver files

If the Mesh Source Saver was used, one mesh source file will be created for each defined source. For a single source, the filename will be whatever was listed with the "filename=" keyword or "source.msm" if nothing was given. For multiple sources, the filenames will include the source identification number, such as "source.*id*.msm". If there were multiple sources and the "makeTotal" keyword was used, the total will be stored in "source.msm". Note that if these files are desired, they must be manually copied back from the SCALE temporary area.

If the keyword "meshBiasFile=" was used, then for every mesh source generated above there will also be a biased mesh source that is used in sampling. These files have the names in the form of "source.sampling.*id*.msm".

8.2.4.5 Statistical checks on point detector and region tallies

With each region tally and point detector tally, detailed statistical information is provided in separate files, "*outputName*.rt*id*.txt" or "*outputName*.pd*id*.txt", just after the group-by-group values for the fluxes and responses. For the total fluxes and any responses of each tally, two tables are given.

First, the values of the tally and several statistical quantities are listed as a function of batch number. With each batch, the statistical tests are listed by number with one of the following: "X" for passing, "-" for failing or a blank if the test could not be performed yet. The second table lists the details of the final statistical checks for the last batch completed. This table lists the value for each of the six tests as well as what the goal is for that test.

As an example, consider a 1 Ci point source of Watt spectrum neutrons inside an r=20 cm sphere of polyethylene. Two tallies are used to find the neutron dose rate (rem/hr) 35 cm from the center of the sphere — a region tally (region between two concentric spheres with radii of 34 and 36 cm) and a point detector 35 cm away from the center of the sphere. Twenty batches of 1000 particles each were used in this example.

The first depiction of the region tally dose rate response is shown in Example 8.2.2. Since four of the tests involve curve fits to the table values over the last half of the simulation, the table is split showing each half of the simulation separately. The second depiction of the region tally, showing all of the details for the last batch, is shown in Example 8.2.3. These two tables show that this tally passed all the statistical tests for the entire second half of the simulation. This information, combined with the fact that this is a simple tally for a well-posed problem, indicates that this tally is well converged.

Example 8.2.2: Tally values for a well-converged tally.

```
Tally Values as the Simulation Progressed

          average      standard    relat      rel       FOM    stats check
 batch     value      deviation    uncert     VOV     (/min)   1 2 3 4 5 6
------  -----------  -----------  -------  --------  --------  -----------
     1  2.06269E+01  2.45012E+00  0.11878  1.45E-02  9.96E+02      -   X
     2  1.87353E+01  1.65492E+00  0.08833  8.16E-03  8.76E+02      -   X
     3  1.82317E+01  1.33026E+00  0.07296  5.63E-03  8.45E+02      -   X
     4  1.80905E+01  1.15401E+00  0.06379  4.27E-03  8.28E+02      -   X
```

(continues on next page)

(continued from previous page)

```
         5  1.82780E+01  1.03792E+00  0.05679  3.40E-03  8.36E+02      -   X
         6  1.75679E+01  9.28759E-01  0.05287  2.97E-03  8.00E+02      -   X
         7  1.71843E+01  8.46577E-01  0.04926  2.60E-03  7.89E+02      X   X
         8  1.74775E+01  7.99275E-01  0.04573  2.23E-03  8.03E+02      X   X
         9  1.73520E+01  7.52502E-01  0.04337  2.00E-03  7.94E+02      X   X
        10  1.76020E+01  7.17178E-01  0.04074  1.76E-03  8.11E+02  X X X X X X

        11  1.74628E+01  6.80834E-01  0.03899  1.62E-03  8.03E+02  X X X X X X
        12  1.75161E+01  6.52464E-01  0.03725  1.48E-03  8.08E+02  X X X X X X
        13  1.72561E+01  6.21777E-01  0.03603  1.39E-03  7.96E+02  X X X X X X
        14  1.73196E+01  6.00329E-01  0.03466  1.29E-03  8.00E+02  X X X X X X
        15  1.72025E+01  5.77747E-01  0.03359  1.21E-03  7.97E+02  X X X X X X
        16  1.72019E+01  5.58956E-01  0.03249  1.14E-03  7.98E+02  X X X X X X
        17  1.72998E+01  5.43721E-01  0.03143  1.06E-03  8.02E+02  X X X X X X
        18  1.73534E+01  5.29351E-01  0.03050  9.99E-04  8.06E+02  X X X X X X
        19  1.72752E+01  5.14264E-01  0.02977  9.51E-04  8.01E+02  X X X X X X
        20  1.74155E+01  5.03061E-01  0.02889  8.95E-04  8.08E+02  X X X X X X
    ------  -----------  -----------  -------  --------  --------  -----------
```

Example 8.2.3: Final check for a well-converged tally.

```
  Final Statistical Check (fits are over the last half of the simulation)

quantity               check                      goal    actual  pass
---------------------- -------------------------  -------  --------  ----
1 mean                 rel slope of linear fit  =  0.00   -0.0118  yes
2 standard deviation   exponent of power fit    = -0.50   -0.5092  yes
3 relative uncertainty final value              <  0.05    0.0289  yes
4 relative VOV         exponent of power fit    = -1.00   -0.9833  yes
5 relative VOV         final value              <  0.10    0.0009  yes
6 figure-of-merit (FOM) rel slope of linear fit  =  0.00    0.0075  yes
---------------------- -------------------------  -------  --------  ----
```

Since the point detector is close compared to the size of the sphere, it should converge slower than the region tally. Contributions coming from different parts of the sphere have large differences in attenuation which will cause large fluctuations in the weights arriving at the point detector. The two statistical tables are shown in Example 8.2.4 and Example 8.2.5. This tally is not yet converged enough to pass most of the statistical tests. With thirty times the simulation time, this point detector tally will pass all six tests.

Fig. 8.2.19 shows the behavior of both the region tally (well-converged) and the point detector tally (not-yet-converged) for this example problem as a function of the simulation run time (shown as the twenty batches of particles). The batch values are shown as blue points and the fits for the last half of the simulation are shown as solid black lines. The average value for a tally should be constant, so test 1 looks at the slope of a linear fit over the tally average over the last half of the simulation. The uncertainty of the tally should decrease with the square root of the total number of particles ($\frac{1}{\sqrt{N}}$), so test 3 computes the slope of an exponential fit which should be close to -0.5 (green dotted line). The variance-of-the-variance (VOV) should decrease with $\frac{1}{N}$, so test 4 computes the slope of an exponential fit which should be close to -1.0. The tally figure-of-merit (FOM) should be constant. Test 6 computes the slope of the FOM values which should be zero.

Example 8.2.4: Tally values for a not-yet-converged tally.

```
  Tally Values as the Simulation Progressed

          average      standard    relat      rel       FOM    stats check
batch      value      deviation    uncert     VOV     (/min)   1 2 3 4 5 6
------  -----------  -----------  -------  --------  --------  -----------
```

(continues on next page)

(continued from previous page)

```
         1  1.45258E+01  3.00915E+00  0.20716  1.60E-01  3.27E+02      -   -
         2  1.83822E+01  4.47657E+00  0.24353  5.27E-01  1.15E+02      -   -
         3  1.84192E+01  3.44084E+00  0.18681  3.20E-01  1.29E+02      -   -
         4  1.87897E+01  2.80685E+00  0.14938  2.33E-01  1.51E+02      -   -
         5  1.78133E+01  2.31117E+00  0.12974  2.09E-01  1.60E+02      -   -
         6  1.75190E+01  2.02252E+00  0.11545  1.74E-01  1.68E+02      -   -
         7  1.79523E+01  2.18108E+00  0.12149  1.78E-01  1.30E+02      -   -
         8  1.75624E+01  1.95272E+00  0.11119  1.63E-01  1.36E+02      -   -
         9  1.67648E+01  1.75017E+00  0.10440  1.58E-01  1.37E+02      -   -
        10  1.70234E+01  1.66732E+00  0.09794  1.30E-01  1.40E+02  X X - - - -

        11  1.70104E+01  1.55192E+00  0.09123  1.18E-01  1.47E+02  - - - X - -
        12  1.71598E+01  1.48432E+00  0.08650  1.03E-01  1.50E+02  X - - X - -
        13  1.74796E+01  1.42808E+00  0.08170  8.85E-02  1.55E+02  X - - - X -
        14  1.68710E+01  1.33186E+00  0.07894  8.70E-02  1.54E+02  X - - - X -
        15  1.65807E+01  1.25707E+00  0.07582  8.34E-02  1.56E+02  X - - - X -
        16  1.65731E+01  1.20836E+00  0.07291  7.61E-02  1.59E+02  X - - X X -
        17  1.62390E+01  1.14550E+00  0.07054  7.40E-02  1.59E+02  X - - X X -
        18  1.70002E+01  1.17106E+00  0.06889  5.75E-02  1.58E+02  X - - X X -
        19  1.70863E+01  1.14126E+00  0.06679  5.20E-02  1.59E+02  X - - - X -
        20  1.68215E+01  1.09329E+00  0.06499  5.04E-02  1.60E+02  X - - - X -
------  -----------  -----------  -------  --------  --------  -----------
```

Example 8.2.5: Final check for a not-yet-converged tally.

```
  Final Statistical Check (fits are over the last half of the simulation)

quantity               check                      goal    actual  pass
---------------------- ------------------------  -------  --------  ----
1 mean                 rel slope of linear fit  =  0.00   -0.0468  yes
2 standard deviation   exponent of power fit    = -0.50   -0.6006  no
3 relative uncertainty final value              <  0.05    0.0650  no
4 relative VOV         exponent of power fit    = -1.00   -1.3823  no
5 relative VOV         final value              <  0.10    0.0504  yes
6 figure-of-merit (FOM) rel slope of linear fit  =  0.00    0.1667  no
---------------------- ------------------------  -------  --------  ----
```

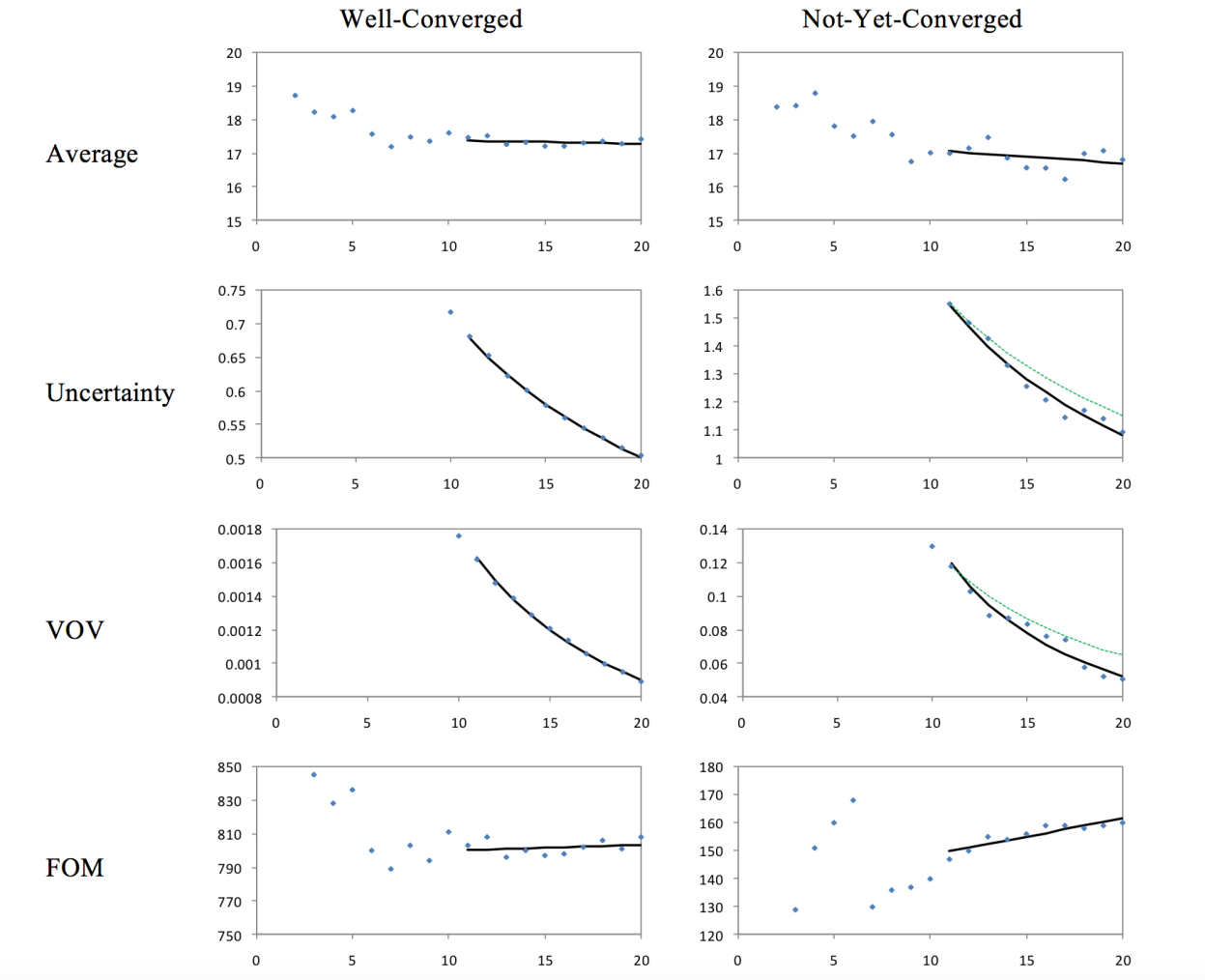


Fig. 8.2.19: Behavior of two tallies as a function of number of particle batches.

Note that the slope of the VOV is found by fitting an exponential curve through the calculated VOV values and is very sensitive to outliers. Users need to apply their own judgment to whether or not the VOV test implemented in Monaco is too strict (failing when the tally seems converged).

Mesh tallies produce similar tables - detailed statistical information is provided in separate files, "*outputName*.mt*id*.flux.txt" and "*outputName*.mt*id*.resp*xx*.txt", for the total flux and responses. First, the statistical values related to the mean relative variance are listed as a function of batch number. With each batch, the statistical tests are listed by number with one of the following: "X" for passing, "-" for failing or a blank if the test could not be performed yet. The second table lists the details of the final statistical checks for the last batch completed. This table lists the value for each of the four tests as well as what the goal is for that test.

8.2.5 EXAMPLE PROBLEMS

8.2.5.1 Neutron transmission through an iron sphere

In the early 1990s, several experiments were performed in order to benchmark ENDF/B-VI cross-section data for iron [MONACO-SWA93] This example problem will use Monaco to simulate one of these experiments: transmission of ^{252}Cf neutrons through a sphere of iron. The Monaco calculations will be compared to two sets of measurements, one by the Czechoslovakian National Research Institute (NRI) and the other by the Skoda Company.

The Monaco model for this sample problem (samplesinputmonaco.ironSphere.inp) will be quite simple-just a point source and a spherical shell of iron. Three different tallies are used which should all give the same result: a region tally (for a thin spherical shell at radius=100cm); a point detector tally at (x,y,z)=(100,0,0); and a coarse mesh tally, with one cell enclosing the point (x,y,z)=(100,0,0).

Input

First, the cross sections need to be computed. Here, csas-mg is used and the "activities" material is included to make sure the flux-to-dose conversion factors are added to the working library.

```
=csas-mg
Materials for Leakage spectrum of Cf-252 through an iron sphere
v7-27n19g
read composition
    wtptSphere 3 8.0 7   26054 5.767   26056 91.618  26057 2.187
                         26058 0.298    6000 0.07     15000 0.03
                         16000 0.03
        1.0 293.0 end
    activities 99 1.0 293.0 end
end composition
end
```

The Monaco input file starts with module name ("monaco") and a title.

```
=monaco
Leakage spectrum of Cf-252 through an iron sphere
```

The mixing table information can be found in the output of the above csas-mg run.

```
read crossSections
    ampxFileUnit=4
    mixture 3
        element     26054 5.150920E-03
        element     26056 7.891181E-02
        element     26057 1.850588E-03
        element     26058 2.478141E-04
        element      6000 2.807736E-04
        element     15031 4.666202E-05
        element     16032 4.283109E-05
        element     16033 3.380690E-07
        element     16034 1.897694E-06
        element     16036 9.015172E-09
    end mixture
end crossSections
```

The SGGP geometry consists of several nested spheres. The regions (in order) are a void for the ^{252}Cf source, the iron shield, a void out to the detector, a thin shell about 100 cm, and then a void to the problem boundary. The only volume that needs to be supplied is the fourth region, since that is where the flux tally will be.

```
read geometry
    global unit 1
        sphere   10   1.25
        sphere   11  25.0
        sphere   12  99.5
        sphere   13  100.5
        sphere   99  200.0
        media  0 1   10       vol=8.181230869
        media  3 1   11 -10   vol=65441.66572
        media  0 1   12 -11   vol=4060822.14
        media  0 1   13 -12   vol=125664.7533
        media  0 1   99 -13   vol=29258384.9
    boundary 99
end geometry
```

For the different tallies, responses, locations, and grid geometry objects need to be defined. For the source, one distribution needs to be defined.

```
read definitions
    response 1
        title="ANSI standard (1977) neutron flux-to-dose-rate factors  (rem/h)/(neutrons/cm2/s)"
        specialDose=9029
    end response
    location 1
        title="true detector location"
        position 100.0 0.0 0.0
    end location
    gridGeometry 1
        title="simple grid"
        xplanes -25 -15 -5 5 15 25 35 55 75 95 105 end
        yLinear 5 -25 25
        zLinear 5 -25 25
    end gridGeometry
    distribution 1
        title="Cf-252 neutrons, Watt spectrum a=1.025 MeV and  b=2.926/MeV"
        special="wattSpectrum"
        parameters 1.025 2.926 end
    end distribution
end definitions
```

The ^{252}Cf source can be modeled using the Watt spectrum with a=1.025 MeV and b=2.926/MeV, which was used for distribution 1. The strength is set so that the total flux at the detector (at r=100 cm) without the shield would be 1 n/cm^2/s. So, strength = 4π = 125664 n/s.

```
read sources
    src 1
        title="Cf-252 neutrons, Watt Spectra using a=1.025 MeV and b=2.926/MeV"
        neutron  strength=125663.70614
        sphere 0.1
        eDistributionID=1
    end src
end sources
```

Three tallies will be defined: a region tally over the fourth region of unit 1 (since this is a symmetric problem); a mesh tally using a coarse mesh over the entire problem; and a point detector tally at the true detector location.

```
read tallies
    regionTally 4
        title="example region tally"
        neutron
        unit=1 region=4
```

(continues on next page)

(continued from previous page)

```
        responseID=1
    end regionTally



    meshTally 1
        title="example mesh tally"
        neutron
        gridGeometryID=1
        responseID=1
    end meshTally
    pointDetector 2
        title="example point detector"
        neutron
        locationID=1
        responseID=1
    end pointDetector
end tallies
```

Monte Carlo parameters include the starting random number seed, the number of particles per batch, and the number of batches to simulate. Since we are only interested in simulating neutrons, use the keywords "neutrons" and "noPhotons" to transport neutrons and not photons. To prevent the production of fission neutrons and secondary gamma rays from neutron interactions, the keywords "fissionMult=0" and "secondaryMult=0" are used. In order to prevent a long run time, a maximum allowable run time using the "maxMinutes=" keyword could also be used.

```
read parameters
    randomSeed=8655745262010033
    perBatch=262000  batches=10
    neutrons  noPhotons
    fissionMult=0  secondaryMult=0
end parameters
```

Without any biasing parameters, Monaco will set the target weights of every energy group in every region to 1 and the values for weight lower bounds (Russian roulette) will be 1/3. Since "noPhotons" was listed in the parameters block, photons should not be generated. So, no biasing block is required.

If desired, a biasing block could be added to change the size of the weight windows (default is 5, giving lower weights of 1/3 and upper weights of 5/3). By using the default target weights of 1.0 and defining the windowRatio as 199, the lower weights are all set to 0.01 and the upper weights for splitting are 1.99.

```
read biasing
    windowRatio=199.0
end biasing
```

or the target weights and lower weights could be listed for every group, every region or for every group/region. For all regions having the same targets and lower bounds, the following could be used:

```
read biasing
    targetWeights  27r1.00 19r0.0 end
    lowerWeights   27r0.01 19r0.0 end
end biasing
```

If this last case were used, the "noPhotons" keyword would not be required since the target weights for all photon groups were explicitly set to 0.

Finally, the Monaco input file is ended with

```
end data
end
```

The complete input file `monaco.ironSphere.inp` is located in the SCALE `samples\input` directory.

Output

When Monaco finishes, there are several output files produced. First, the main output file lists summaries for each of the region tallies and point detector tallies. With each mesh tally, a *.3dmap file is produced which can be viewed with the Mesh File Viewer capabilities of Fulcrum. Region tallies and point detectors produce text files listing group-by-group results for fluxes and any optional responses. Both of these tallies also produce *.chart files which contain information about how the simulation progresses with each batch. These can be viewed with the Interactive Plotter capabilities of Fulcrum.

From the main output file, the final tally results summary is shown below:

```
Final Tally Results Summary
 ============================

Neutron Point Detector 2.  example point detector
                        average      standard     relat      FOM    stat checks
    tally/quantity       value       deviation    uncert   (/min)   1 2 3 4 5 6
    ------------------  -----------  -----------  -------  --------  -----------
    uncollided flux     5.28059E-03  3.15856E-06  0.00060
    total flux          9.76163E-01  1.06422E-03  0.00109  8.39E+04  X X X X X X
    response 1          7.55800E-05  8.63873E-08  0.00114  7.64E+04  X X X X X X
    ------------------  -----------  -----------  -------  --------  -----------

 Neutron Region Tally 4.  example region tally
                        average      standard     relat      FOM    stat checks
    tally/quantity       value       deviation    uncert   (/min)   1 2 3 4 5 6
    ------------------  -----------  -----------  -------  --------  -----------
    total flux (tl)     9.75719E-01  5.54774E-05  0.00006  3.09E+07  X X X X X X
    total flux (cd)     0.00000E+00
    response 1          7.55800E-05  8.63873E-08  0.00114  7.64E+04  X X X X X X
    ------------------  -----------  -----------  -------  --------  -----------
```

The fluxes reported are the total fluxes. Note the close agreement between the point detector flux and the region tally track-length estimate. The region tally did not produce a collision-density estimate of flux since the region was a void. Also notice that the neutron dose (response 1) from both tallies match well. The tally details that are saved in the *.pd*.txt or *.rt*.txt files are shown in Table 8.2.19. Group values for the mesh tally were obtained through the Mesh File Viewer.

For the mesh tally, the total neutron flux and neutron dose can be visualized using the Mesh File Viewer, as shown in Fig. 8.2.20 and Fig. 8.2.21. Using the mouse in the viewer, the flux and the dose rate for the cell that contains the detector location (100,0,0) are found to be 1.004 ± 2.3% n/cm^2/s and 7.82 × 10^{-5} ± 2.4% rem/hr, respectively, matching the other tallies well, given the higher uncertainties for the mesh tally.

A comparison of the results of all three Monaco tallies to the two experimental measurements is shown in Fig. 8.2.22 through Fig. 8.2.26 for different cross section libraries. Note that the point detector results and the region tally results are the same for most of the energy range shown. The line representing the mesh tally result becomes broken in some plots because no neutrons of certain energy groups crossed this particular mesh cell. These four plots are each for 10 minutes of computation using a 2 GHz Linux processor.

Using the library with finer groups, the Monaco results show more of the structure seen in the experiments over the energy range from 0.01 MeV to 1.0 MeV, as shown in Fig. 8.2.23 and Fig. 8.2.24.

Table 8.2.19: Group-by-group details of the three flux tallies (ENDF/B-VII.0 27n19g)

Energy	Mesh Tally		**Point detector**		Region Tally	
(MeV)	**value**	**rel-unc**	**value**	**rel-unc**	**value**	**rel-unc**
2.000E+07	9.59E-04	0.6325	1.30E-03	0.0344	1.35E-03	0.0168
6.376E+06	1.26E-02	0.1472	8.19E-03	0.0117	8.40E-03	0.0067
3.012E+06	2.03E-02	0.1413	2.21E-02	0.0060	2.21E-02	0.0041
1.827E+06	2.88E-02	0.0919	2.41E-02	0.0047	2.42E-02	0.0039
1.423E+06	8.94E-02	0.0770	9.05E-02	0.0026	9.06E-02	0.0020
9.072E+05	3.40E-01	0.0466	3.26E-01	0.0018	3.25E-01	0.0009
4.076E+05	4.19E-01	0.0320	4.15E-01	0.0017	4.15E-01	0.0007
1.111E+05	7.25E-02	0.0790	6.65E-02	0.0046	6.64E-02	0.0023
1.503E+04	1.27E-02	0.1751	1.21E-02	0.0118	1.21E-02	0.0055
3.035E+03	4.05E-03	0.3065	4.80E-03	0.0179	4.75E-03	0.0083
5.830E+02	1.51E-03	0.4935	2.58E-03	0.0237	2.59E-03	0.0109
1.013E+02	6.10E-04	0.6476	1.26E-03	0.0320	1.20E-03	0.0151
2.902E+01	4.75E-04	0.6724	7.23E-04	0.0367	7.16E-04	0.0182
1.068E+01	2.82E-04	0.9487	5.19E-04	0.0402	5.03E-04	0.0210
3.059E+00			1.60E-04	0.0597	1.40E-04	0.0403
1.855E+00			7.91E-05	0.0824	8.31E-05	0.0534
1.300E+00			3.15E-05	0.1059	2.96E-05	0.0901
1.125E+00			2.24E-05	0.1108	1.97E-05	0.1086
1.000E+00			3.21E-05	0.1299	2.45E-05	0.0979
8.000E-01			4.20E-05	0.1127	3.63E-05	0.0816
4.140E-01			1.02E-05	0.1720	1.03E-05	0.1583
3.250E-01			1.13E-05	0.2433	7.62E-06	0.1701
2.250E-01			8.26E-06	0.2554	7.42E-06	0.1746
1.000E-01			1.00E-06	0.4480	6.73E-07	0.6451
5.000E-02			6.61E-08	0.4590		
3.000E-02			6.59E-10	0.4491		
1.000E-02			6.90E-13	0.3458		
1.000E-05						

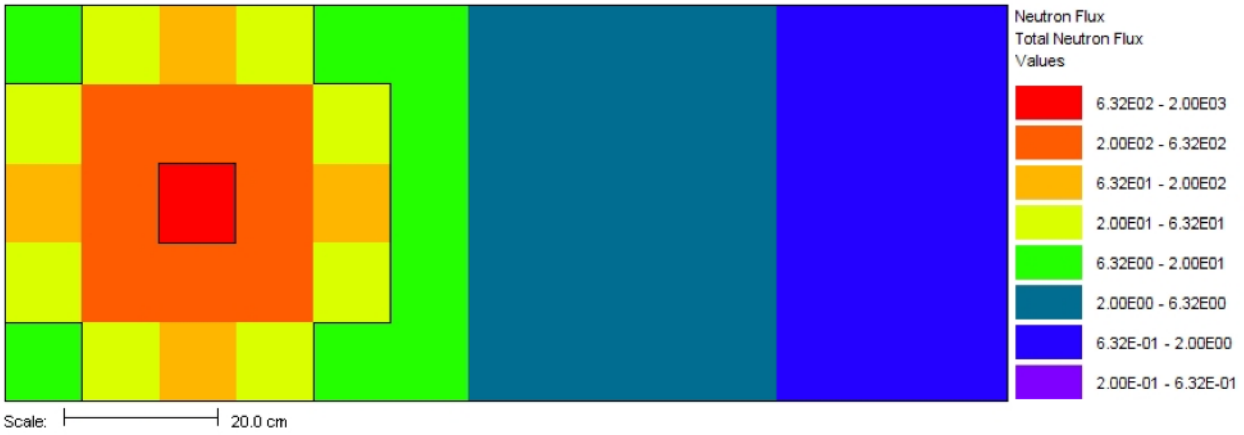


Fig. 8.2.20: Total neutron flux in n/cm2/s.

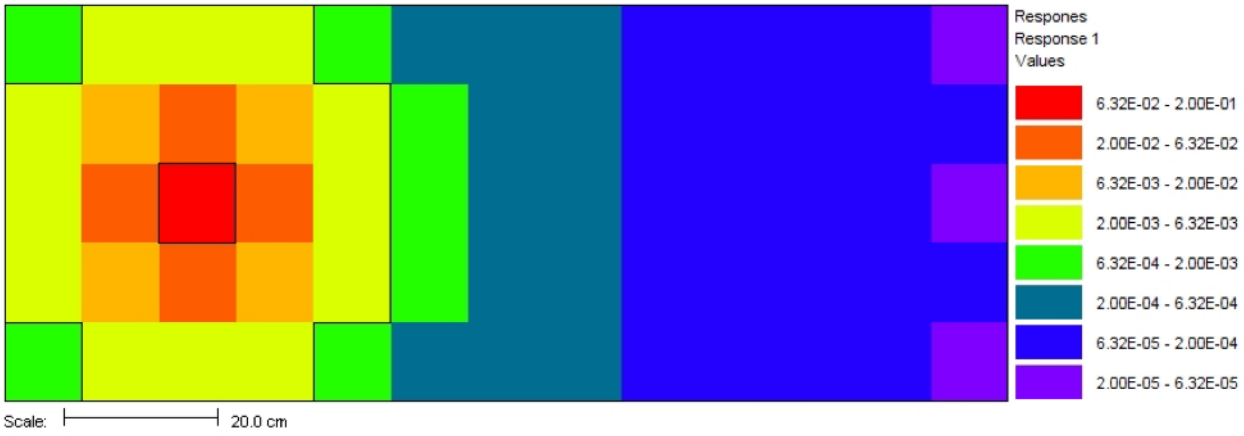


Fig. 8.2.21: Neutron dose rate in rem/hr.

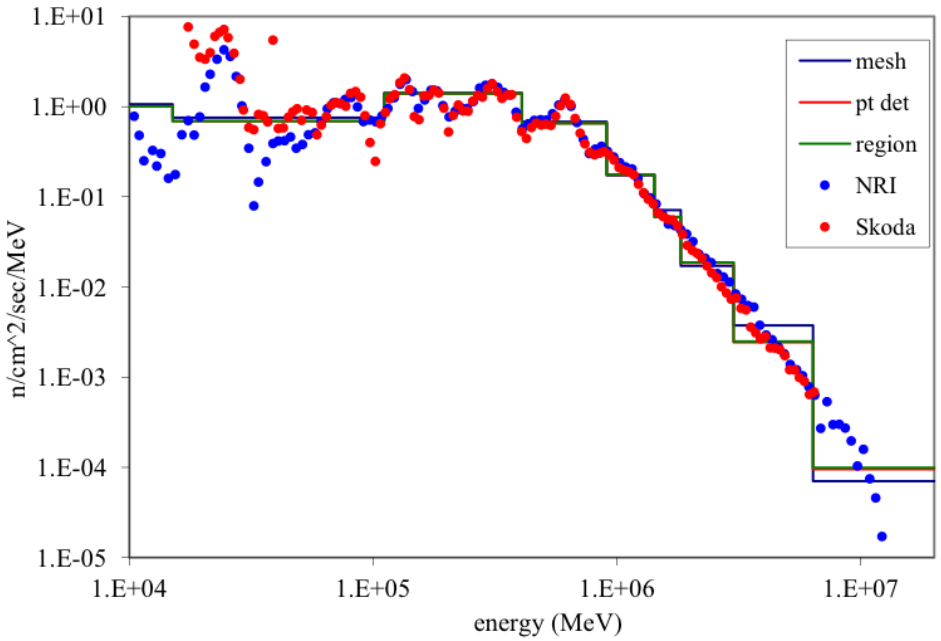


Fig. 8.2.22: Comparison of Monaco results using the ENDF/B-VII.0 27n/19g library and the measured values.

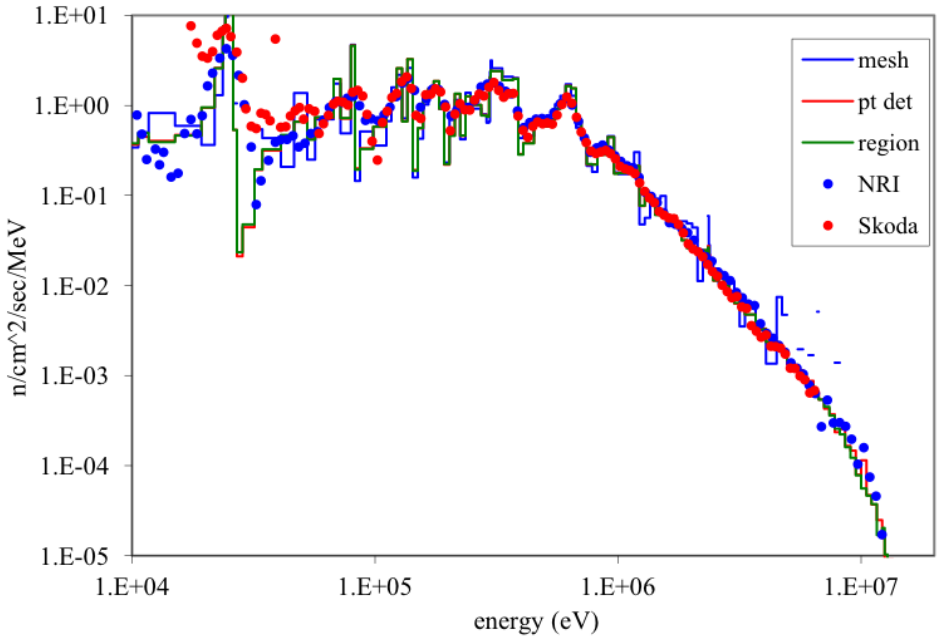


Fig. 8.2.23: Comparison of Monaco results using the new ENDF/B-VII.0 200n/47g library and the measured values.

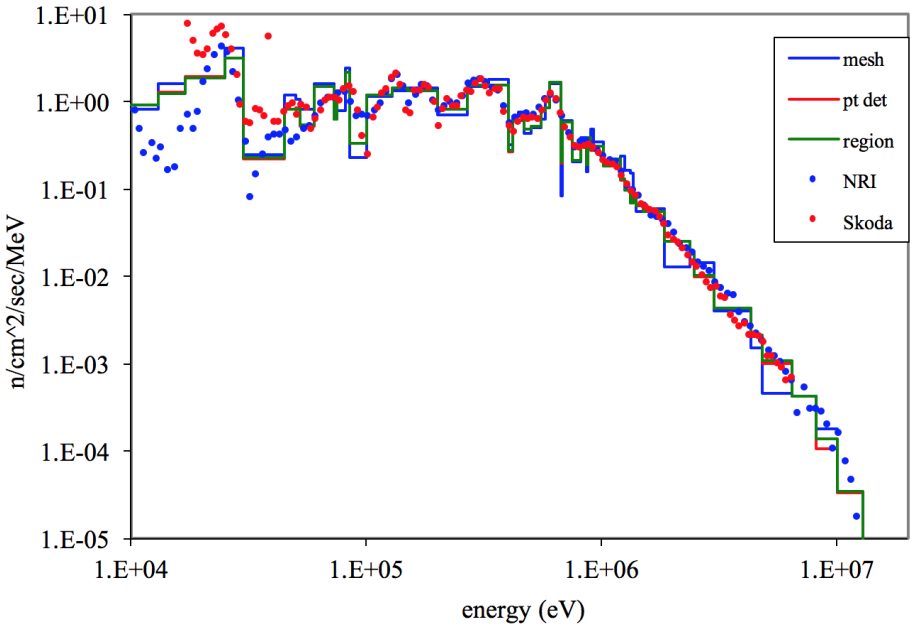


Fig. 8.2.24: Comparison of Monaco results using the ENDF/B-VII 238n library and the measured values.

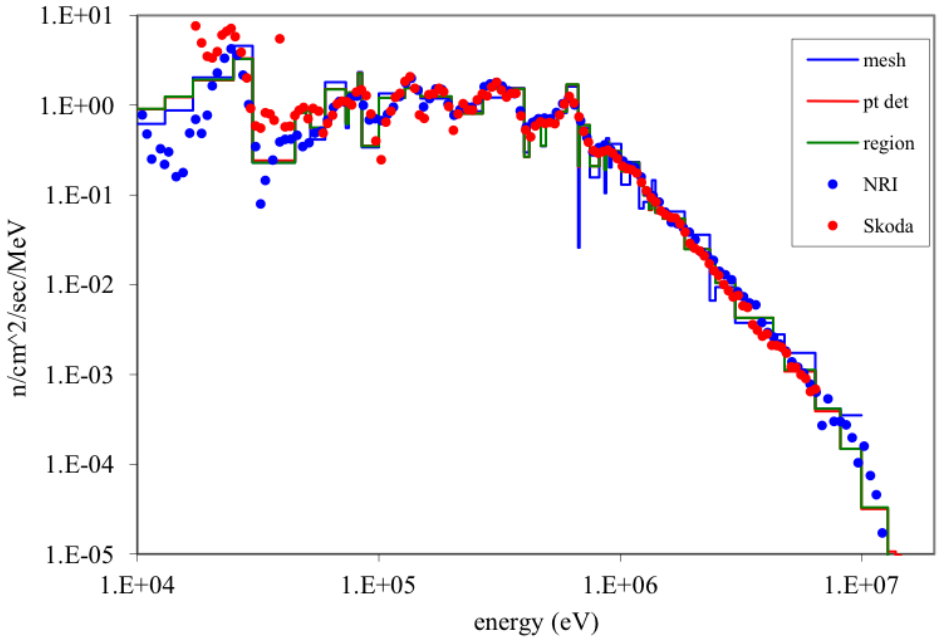


Fig. 8.2.25: Comparison of Monaco results using the ENDF/B-VII 252n library and the measured values.

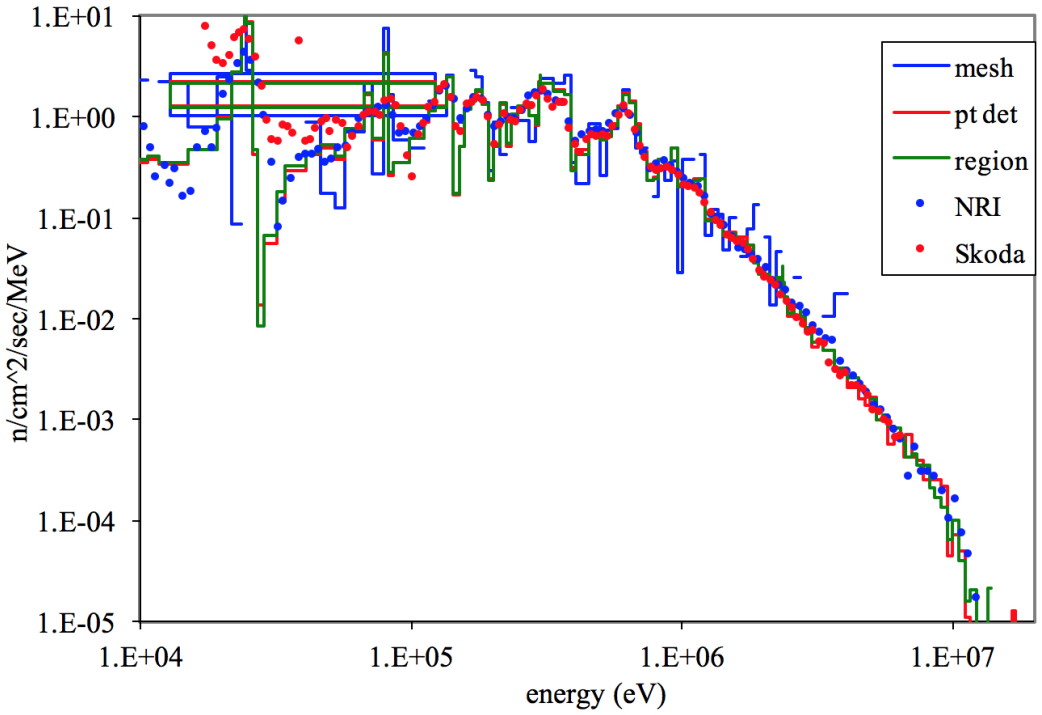


Fig. 8.2.26: Comparison of Monaco results using the ENDF/B-VII CE n/p library (binned using the 200n group structure) and the measured values.

8.2.5.2 Neutrons through a heavy water sphere

Similar to the first example problem, ^{252}Cf neutrons were measured outside of a sphere filled with heavy water [MONACO-JTN+97] Two measurements were made: one without the iron/polyethylene shield and one with the shield. These two measurements were subtracted to account for scatter from the floor (which is about a 5% effect for energies above 10 keV). A great amount of detail is given for the materials and geometry of the source holder, insertion tube, and detectors in Ref. 6. For this sample problem, just the basics will be modeled in two inputs: `monaco.d2oSphereA.inp` and `monaco.d2oSphereB.inp`, both located in the SCALE `samples\input` directory.

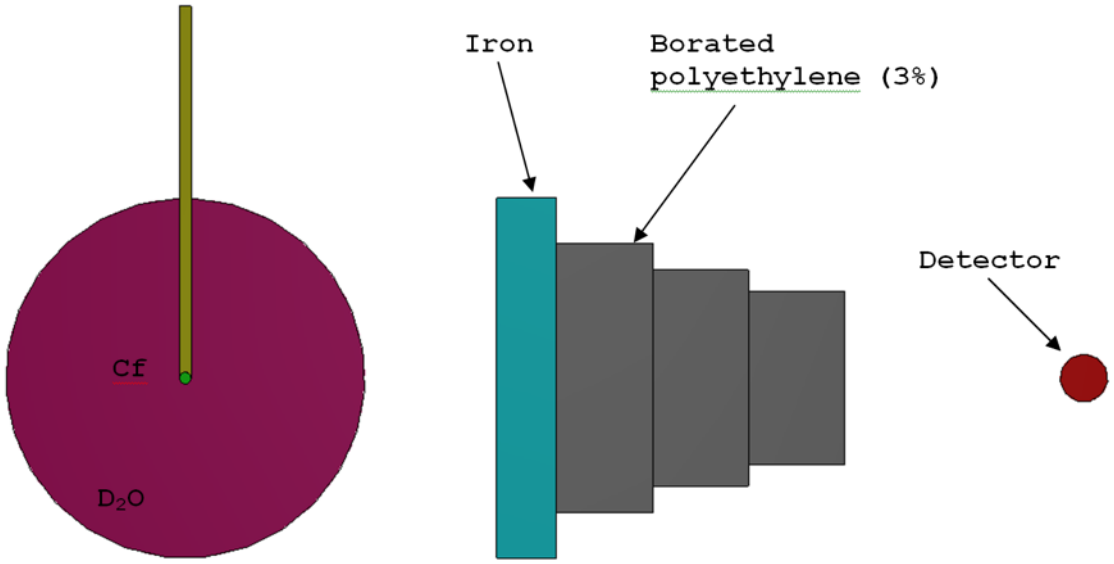


Fig. 8.2.27: Experimental setup.

Input

First, the cross sections for four materials need to be computed. Here, csas-mg is used:

```
=csas-mg
Leakage spectrum of Cf-252 through a heavy water sphere
v7-27n19g
read composition
    d2o          1 0.99286 293.0 end
    h2o          1 0.00714 293.0 end
    polyethylene 2 0.882   293.0 end
    boron        2 0.118   293.0 end
    iron         3 1.0     293.0 end
    orconcrete   4 1.0     293.0 end
end composition
end
```

The Monaco input file starts with module name (“monaco”) and a title.

```
=monaco
Leakage spectrum of Cf-252 through a heavy water sphere, shield in place
```

The mixing table information can be found in the output of the above csas-mg run. The newer ENDF/B-VII libraries contain isotopic data for iron, instead of just the elemental data in the older ENDF/B-V libraries.

```
read crossSections
    ampxFileUnit=4

    mixture 1
        element    1002     6.60172E-02
        element    8016     3.32469E-02
        element    1001     4.76617E-04
    end mixture

    mixture 2
        element 9001001     6.96775E-02
        element    6000     3.48387E-02
```

(continues on next page)

(continued from previous page)

```
        element    5010     3.10004E-03
        element    5011     1.24780E-02
    end mixture

    mixture 3
        element   26054     4.95419E-03
        element   26056     7.77702E-02
        element   26057     1.79605E-03
        element   26058     2.39022E-04
    end mixture
```

```
    mixture 4
        element   26054     1.12815E-05
        element   26056     1.77096E-04
        element   26057     4.08991E-06
        element   26058     5.44292E-07
        element    1001     8.50077E-03
        element    6000     2.01991E-02
        element    8016     3.55123E-02
        element   11023     1.63230E-05
        element   12024     1.46935E-03
        element   12025     1.86017E-04
        element   12026     2.04805E-04
        element   13027     5.55811E-04
        element   14028     1.56780E-03
        element   14029     7.96453E-05
        element   14030     5.25642E-05
        element   19039     3.75869E-05
        element   19040     4.71559E-09
        element   19041     2.71255E-06
        element   20040     1.07616E-02
        element   20042     7.18248E-05
        element   20043     1.49866E-05
        element   20044     2.31571E-04
        element   20046     4.44048E-07
        element   20048     2.07592E-05
    end mixture
end crossSections
```

The SGGP geometry consists of two nested spheres for the source and heavy water sphere. Four cylindrical shields made of either borated polyethylene or iron are placed between the sphere and the detector position (75,0,0). The experiment sat 2 m above the floor of an experimental hall that measured $10 \times 13 \times 25$ m. Here, the origin corresponds to the source at the center of the heavy water sphere.

```
read geometry
    global unit 1
        sphere    10   0.5
        sphere    11  15.0
        sphere    21   2.0  origin x=75.0
        xcylinder 31  15.0  26.0 31.0
        xcylinder 32  11.25 31.0 39.0
        xcylinder 33   9.00 39.0 47.0
        xcylinder 34   7.25 47.0 55.0
        cuboid    41  650.0 -650  500 -500   2300 -200
        cuboid    42  750.0 -750  600 -600   2400 -300
        media  0 1   10
        media  1 1   11 -10
        media  0 1   21          vol=33.510322
        media  3 1   31
        media  2 1   32
        media  2 1   33
        media  2 1   34
        media  0 1   41 -11 -21  -31 -32 -33 -34
```

(continues on next page)

(continued from previous page)

```
        media  4 1   42 -41
    boundary 42
end geometry
```

A second input file (`samples\input\monaco.d2oSphereA.inp`) was created for the geometry without the four xcylinder shields in place.

For this example, a point detector will be used to calculate the flux at the detector location and the source will require a Watt distribution.

```
read definitions
    location 1
        title="true detector location"
        position 75.0 0.0 0.0
    end location
    distribution 1
        title="Cf-252 neutrons, Watt spectrum a=1.025 MeV and  b=2.926/MeV"
        special="wattSpectrum"
        parameters 1.025 2.926 end
    end distribution
end definitions
```

The strength is set so that the total flux at the detector (at r=75 cm) without any shield would be 1 n/cm^2/s. So, strength = 4π $(75)^2$ = 70686 n/s.

```
read sources
    src 1
        title="Cf-252 neutrons, Watt fission spectrum, using a=1.025 and b=2.926"
        neutrons  strength=70685.834704
        sphere 0.1
        eDistributionID=1
    end src
end sources
```

Two tallies will be defined: a region tally over the third region of unit 1 and a point detector tally at the true detector location.

```
read tallies
    regionTally 3
        title="example region tally"
        neutron
        unit=1 region=3
    end regionTally



    pointDetector 2
        title="example point detector"
        neutron
        locationID=1
    end pointDetector
end tallies
```

Monte Carlo parameters include the starting random number seed, the number of particles per batch, and the number of batches to simulate. Since there are no photon tallies, the keyword “secondaryMult=0” is used so that photons are not produced nor transported.

```
read parameters
    randomSeed=8655745262010035
```

(continues on next page)

(continued from previous page)

```
    perBatch=18700  batches=90
    neutrons  noPhotons
    fissionMult=0  secondaryMult=0
end parameters
```

Finally, the Monaco input file is ended with

```
end data
end
```

Output

For the case without the xcylinder shields, the tally summaries from the main output file are reported as:

```
Neutron Point Detector 2.  example point detector
                         average      standard     relat      FOM    stat checks
    tally/quantity         value      deviation    uncert   (/min)   1 2 3 4 5 6
    -------------------  -----------  -----------  -------  --------  -----------
    uncollided flux     4.20334E-02  6.81217E-05  0.00162
    total flux          1.13349E+00  3.28455E-03  0.00290  1.18E+04  X X X X X X
    -------------------  -----------  -----------  -------  --------  -----------

 Neutron Region Tally 3.  example region tally
                         average      standard     relat      FOM    stat checks
    tally/quantity         value      deviation    uncert   (/min)   1 2 3 4 5 6
    -------------------  -----------  -----------  -------  --------  -----------
    total flux (tl)     1.00584E+00  1.47136E-01  0.14628  4.64E+00  - - - X X -
    total flux (cd)     0.00000E+00
    -------------------  -----------  -----------  -------  --------  -----------
```

Note that in this sample calculation (10 minutes), the region tally did not pass all of the statistical checks and the relative uncertainty for the region tally is very large due to the region's small size. This shows how useful a point detector tally can be in estimating fluxes for har-to-reach locations.

For the case with the xcylinder shields in place (450 minutes), the final results were:

```
Neutron Point Detector 2
                        average      standard     relat      FOM    stat checks
    tally/quantity         value      deviation    uncert   (/min)   1 2 3 4 5 6
    -------------------  -----------  -----------  -------  --------  -----------
    uncollided flux     8.37999E-05  1.44384E-07  0.00172
    total flux          1.26869E-01  1.10733E-04  0.00087  4.80E+03  X X X X X X
    -------------------  -----------  -----------  -------  --------  -----------

 Neutron Region Tally 3.  example region tally
                         average      standard     relat      FOM    stat checks
    tally/quantity         value      deviation    uncert   (/min)   1 2 3 4 5 6
    -------------------  -----------  -----------  -------  --------  -----------
    total flux (tl)     1.40236E-01  1.22375E-02  0.08726  4.80E-01  - - - X X -
    total flux (cd)     0.00000E+00
    -------------------  -----------  -----------  -------  --------  -----------
```

Here again, the region tally performed very poorly, showing that very few neutrons actually crossed the tally region. The four shields make this problem very slow to converge because rare particles can arrive at the detector after paths that see different amounts of shield.

The group-wise results list in the point detector detail files from each case can be subtracted and compared to the measurements and calculations listed in [MONACO-JTN+97]. The computed and measured neutron spectra seen by the detector are shown in Fig. 8.2.28 through Fig. 8.2.32 using different cross-section libraries. For each figure, the computational times for case "A", without the shields and case "B", with the shields,

were 10 minutes and 90 minutes, each on a 2 GHz Linux processor. For the CE case, 100 and 900 minutes were used to reduce the statistical uncertainties. Except for the 27/19 multigroup case shown above, none of the "B" cases passed all of the statistical tests.

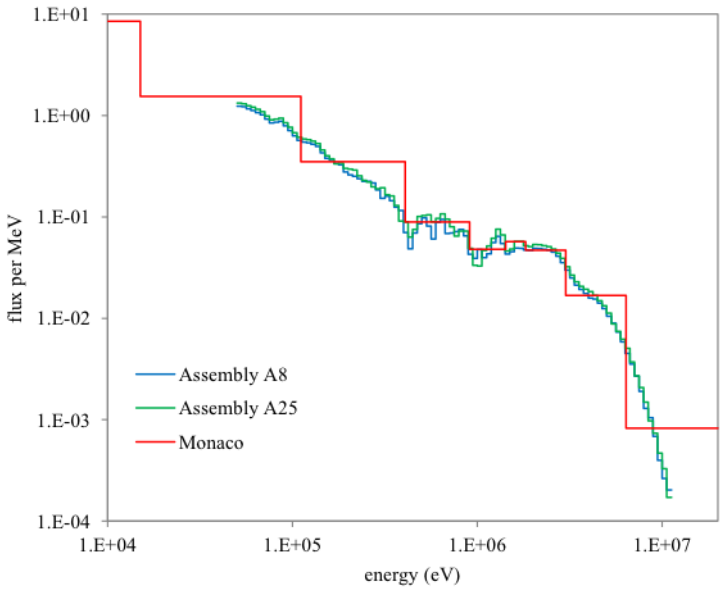


Fig. 8.2.28: Comparison of Monaco calculated results using the ENDF/B-VII.0 27n/19g library to the measured values.

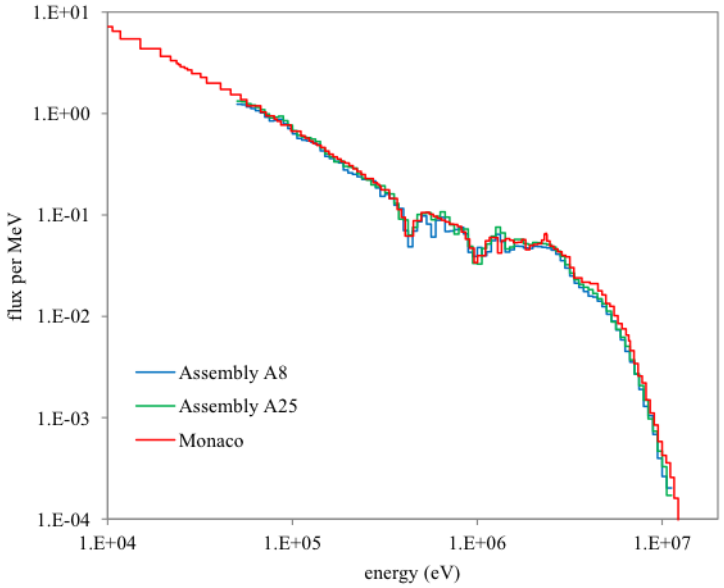


Fig. 8.2.29: Comparison of Monaco calculated results using the ENDF/B-VII.0 200n/47g library to the measured values.

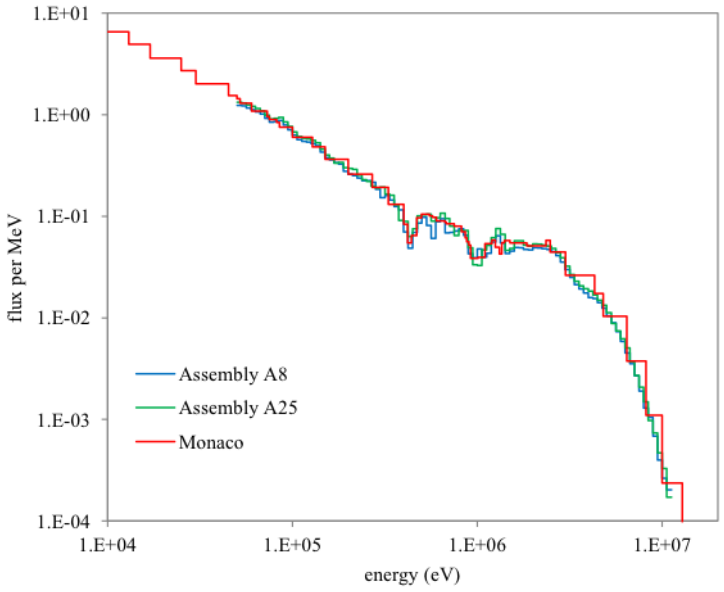


Fig. 8.2.30: Comparison of Monaco calculated results using the ENDF/B-V 238n library to the measured values.

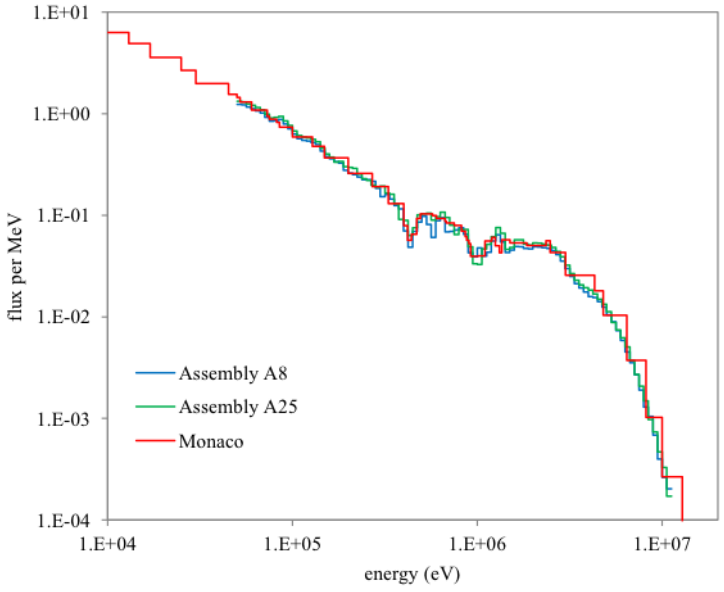


Fig. 8.2.31: Comparison of Monaco calculated results using the ENDF/B-V 252n library to the measured values.

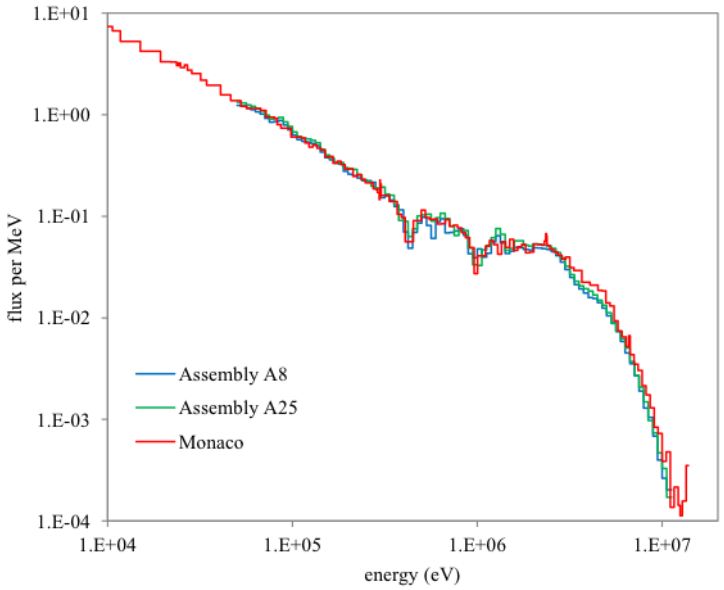


Fig. 8.2.32: Comparison of Monaco calculated results using the ENDF/B-VII CE n/p library (binned using the 200n group structure) to the measured values.

8.2.5.3 Activation rate from a neutron howitzer

A *neutron howitzer* is a common laboratory-sized neutron source. A few curies of an alpha emitter is mixed with an isotope that has a large (α,n) cross section. This source is then stored in a large tank of moderating material (water, paraffin, etc.) with an access port. Materials can be placed into the access port for irradiation. Activation studies are a typical use of a neutron howitzer. Note that the large moderator tank also provides shielding for the users. Typical source strengths are such that users would probably not wish to spend large amounts of time in front of an open access port.

A typical size and shape for a neutron howitzer is an upright cylinder, outer radius of 30 cm by 70 cm in height. The outer wall, top, and bottom are made of 2 cm thick Plexiglas. The access port is a cylinder of radius 2 cm extending 5 cm away from the tank center through the side wall. A Plexiglas rod can be removed/inserted into the port for sample loading. The tank is filled with ordinary water. Fig. 8.2.33 shows a cutaway view of a very simplified model of the tank with the Plexiglas rod inserted into the access port, to prevent neutrons from streaming into the room. The source is located in the center of the tank and is small enough to be considered a point source. A small foil ($1 \times 1 \times 0.001$ cm) gold foil ^{197}Au is included as an activation sample.

The goal is to use Monaco to calculate the activation rate density, R, in units of /cm ^{3}/s, of the gold foil

$$R = \int \Sigma(E)\,\phi(E)\,dE \tag{8.2.23}$$

so that the activity, A, can be calculated as a function of time in the howitzer, T, and time outside the howitzer, t. Note that the half-life of ^{198}Au of 2.7 days (decay constant is λ=2.97 $\times$ 10^{-6} /s).

$$A = RV\left(1 - e^{-\lambda T}\right)e^{-\lambda t} \tag{8.2.24}$$

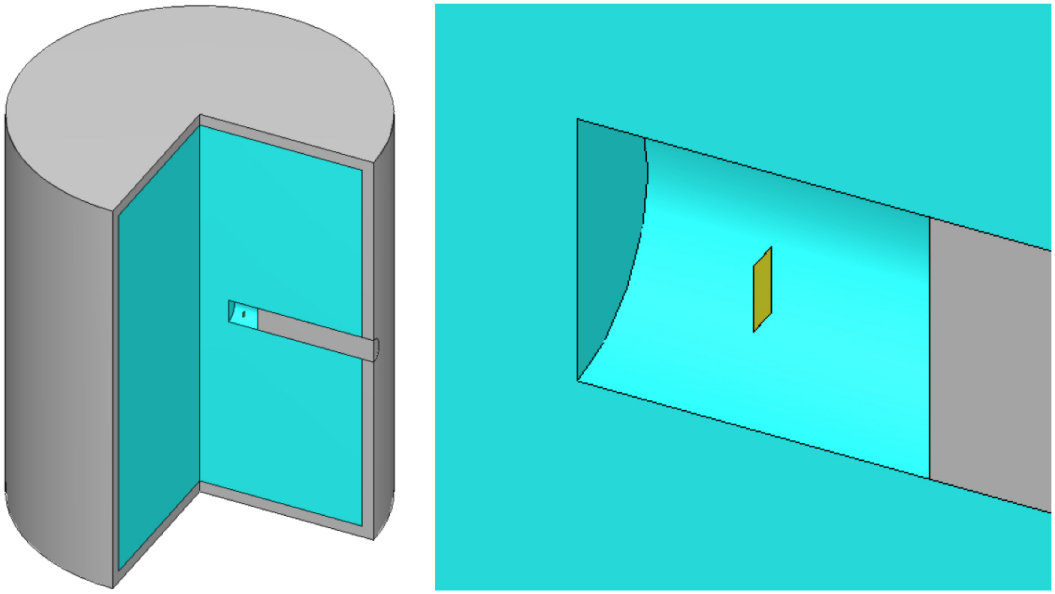


Fig. 8.2.33: Idealized geometry for a neutron howitzer.

Input file

The following input file represents the simple model of a neutron howitzer used for a gold foil activation experiment. The file `monaco.howitzer.inp` is located in the SCALE `samples\input` directory. CSAS-MG is used to generate the cross sections.

```
'===============================================================================
' Generate the Cross Sections
'===============================================================================
=csas-mg
Materials for Monaco/MAVRIC Training - Exercise Problem 1.
v7-27n19g
read composition
    h2o          1 1.0 293.0 end
    plexiglass   2 1.0 293.0 end
    Au           3 1.0 293.0 end
end composition
end


'===============================================================================
' Monaco functional module and title
'===============================================================================
=monaco
Monaco/MAVRIC Training - Exercise Problem 1.  Neutron Howitzer

'-------------------------------------------------------------------------------
' Cross Section Information
'-------------------------------------------------------------------------------
read crossSections
    ampxFileUnit=4
    mixture 1
        element   1001 6.675146E-02
        element   8016 3.337573E-02
    end mixture
```

(continues on next page)

(continued from previous page)

```
    mixture 2
        element   1001 5.678730E-02
        element   6000 3.549206E-02
        element   8016 1.419682E-02
    end mixture
    mixture 3
        element  79197 5.772470E-02
    end mixture
end crossSections

'------------------------------------------------------------------------------
' Geometry Block - SCALE standard geometry package (SGGP)
'------------------------------------------------------------------------------
read geometry
    global unit  1
        cylinder  1  30    35 -35
        cylinder  2  28    33 -33
        xcylinder 3   2    30  10
        xcylinder 4   2    10   5
        cuboid   11   7.501 7.500  0.5 -0.5  0.5 -0.5

        media  1 1   2 -3 -4
        media  2 1   1 -2 -3
        media  2 1   3
        media  0 1   4 -11

        com="gold foil"
        media  3 1   11  vol=0.001
    boundary 4
end geometry


'------------------------------------------------------------------------------
' Definitions Block
'------------------------------------------------------------------------------
read definitions
    response 1
        title="gold macro"
        mat=3 ZAID=79197 MT=102 macro
        makeChart
    end response

    distribution 1
        title="Pu-Be Source Energy Spectra - from Knoll (Anderson & Neff)"
        abscissa 11.0E+06 10.5E+06 10.0E+06  9.5E+06  9.0E+06
                  8.5E+06  8.0E+06  7.5E+06  7.0E+06  6.5E+06
                  6.0E+06  5.5E+06  5.0E+06  4.5E+06  4.0E+06
                  3.5E+06  3.0E+06  2.5E+06  2.0E+06  1.5E+06
                  1.0E+06  0.5E+06  0.0E+06  end
        truePDF 2.74935E-03 7.94257E-03 1.38995E-02 8.70628E-03 1.40522E-02
                3.37559E-02 4.13930E-02 3.54361E-02 4.12403E-02 2.93264E-02
                3.89491E-02 5.97220E-02 6.76646E-02 6.56789E-02 6.41515E-02
                8.11059E-02 6.00275E-02 5.59035E-02 4.78082E-02 5.78891E-02
                8.20223E-02 9.05758E-02 end
        runSampleTest
    end distribution
end definitions

'------------------------------------------------------------------------------
' Sources Block
'------------------------------------------------------------------------------
read sources
    src 1
        neutrons  strength=1.377E+07
        cuboid  0 0  0 0  0 0
```

(continues on next page)

(continued from previous page)

```
        eDistributionID=1
    end src
end sources

'-------------------------------------------------------------------------------
' Tallies Block
'-------------------------------------------------------------------------------
read tallies
    regionTally 1
        neutron
        unit=1 region=5
        responseID=1
    end regionTally
end tallies

'-------------------------------------------------------------------------------
' Parameters Block
'-------------------------------------------------------------------------------
read parameters
    randomSeed= 8650005740006085
    perBatch=47000 batches=10
    neutrons  noPhotons
    fissionMult=0  secondaryMult=0
end parameters

end data
end
```

Output

The main text output file results are reported as:

```
Neutron Region Tally 1.
                          average      standard     relat      FOM    stat checks
    tally/quantity         value       deviation    uncert   (/min)   1 2 3 4 5 6
    -------------------  -----------  -----------  -------  --------  -----------
    total flux (tl)     8.59999E+04  2.64821E+03  0.03079  9.70E+01  X - X - X -
    total flux (cd)     1.23288E+05  8.52974E+04  0.69186  1.92E-01  - - - - - -
    response 1          2.66771E+05  1.12525E+04  0.04218  5.17E+01  - - X X X -
    -------------------  -----------  -----------  -------  --------  -----------
```

In addition to the main output file, the diagnostics from the definitions block are shown in Fig. 8.2.34 and Fig. 8.2.35 The final value for the activation rate density R is 2.65×10^5 /cm^3/sec. Note how the track length tally performs better than the collision density estimate for this thin foil region. But also note that after ten minutes, not all of the statistical checks are converged. Running this problem for longer times (even up to 10 hours) does not make all of the tests pass. Due to the very thin width of the of the tally region, most track-lengths across are very small but occasionally a very long track-length is recorded. This rare but high-value score tends to upset the convergence metrics. A tally region over a larger, more cuboid-like volume would make for a better behaved tally.

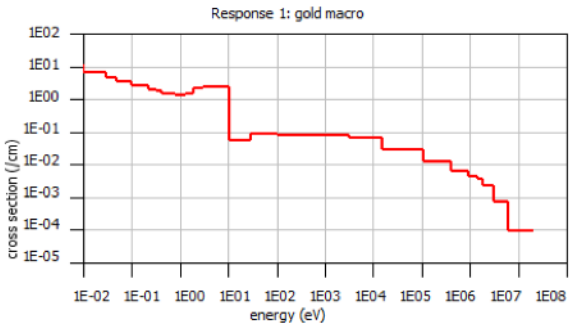


Fig. 8.2.34: Response 1.

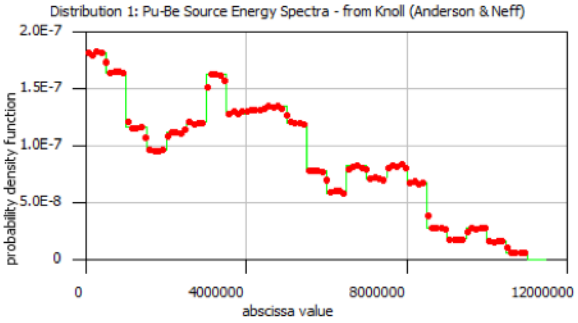


Fig. 8.2.35: Distribution 1.

8.2.5.4 Graphite shielding measurements

K. Ueki of the Nuclear Technology Division, Ship Research Division in Japan performed many simple studies on a variety of shielding materials layered in different combinations. He and his colleagues used both neutron (^{252}Cf) and photon (^{60}Co) sources to investigate the shielding effectiveness of steel, graphite, and many hydrogen-containing materials as single shields or in combinations.

One such series of measurements was for pure graphite [MONACO-UOA92] A ^{252}Cf neutron source was placed in the center of a 50 cm cube of paraffin which had a 45° cone cut-out. A neutron meter was placed 110 cm from the source. Sheets of material, in 5 × 80 ×.

Table 8.2.20: Ueki's experimental results

Thickness (cm)	Dose equivalent attenuation
2	0.828810
5	0.721721
10	0.526054
15	0.364949
20	0.253182
25	0.170514
30	0.112591
35	0.074181

Input file

The following is a listing of the file monaco.graphite.inp located in the SCALE samplesinput directory. This represents a simple model of Ueki's experiment for the 20 cm graphite slab.

```
'===============================================================================
' Generate the Cross Sections
'===============================================================================
=csas-mg
Materials for Monaco/MAVRIC Training - Exercise Problem 2.
v7-200n47g
read composition
    para(h2o)    1         1.0 293.0 end
    carbon       2 den=1.7 1.0 300.0 end
    activities  99         1.0 293.0 end
end composition
end


'===============================================================================
' Monaco functional module and title
'===============================================================================
=monaco
Monaco/MAVRIC Training - Exercise Problem 2.  Graphite Shielding Measurements

'-------------------------------------------------------------------------------
' Cross Section Information
'-------------------------------------------------------------------------------
read crossSections
    ampxFileUnit=4
    mixture 1
        element  1001 7.991204E-02
        element  6000 3.841925E-02
    end mixture
    mixture 2
        element  6000 8.523484E-02
    end mixtureend crossSections

'-------------------------------------------------------------------------------
' Geometry Block - SCALE standard geometry package (SGGP)
'-------------------------------------------------------------------------------
read geometry
    global unit 1
        cuboid  1   25.0 -25.0  25.0 -25.0   25.0 -25.0
        cone    2  10.35948  25.01   0.0 0.0   rotate  a1=-90 a2=-90 a3=0
        cuboid  3   90.0  70.0  40.0 -40.0   40.0 -40.0
        cuboid 99  120.0 -30.0  50.0 -50.0   50.0 -50.0
        media 1 1  1 -2
        media 0 1  2
        media 2 1  3
        media 0 1  99 -1  -2 -3
    boundary 99
end geometry

'-------------------------------------------------------------------------------
' Geometry Plots
'-------------------------------------------------------------------------------
read plot
    scr=yes

    ttl="slice through plane of z=0, whole geometry"
    pic=mixture
    xul=-30.0 yul=50.0 zul=0.0
    xlr=120.0 ylr=-50.0 zlr=0.0
    uax=1.0 vax=0.0 wax=0.0
    udn=0.0 vdn=-1.0 wdn=0.0
```

(continues on next page)

(continued from previous page)

```
    nax=640
    end

    ttl="slice through plane of y=0, through just the paraffin block"
    pic=mixture
    xul=-30.0 yul=0.0 zul=50.0
    xlr=30.0  ylr=0.0 zlr=-50.0
    uax=1.0 vax=0.0 wax=0.0
    udn=0.0 vdn=0.0 wdn=-1.0
    nax=640
    end

    ttl="slice through plane of x=0, through just the paraffin block"
    pic=mixture
    xul=0.0 yul=-50.0 zul=50.0
    xlr=0.0 ylr=50.0 zlr=-50.0
    uax=0.0 vax=1.0 wax=0.0
    udn=0.0 vdn=0.0 wdn=-1.0
    nax=640
    end
end plot

'-------------------------------------------------------------------------------
' Definitions Block
'-------------------------------------------------------------------------------
read definitions
    location 1
        position 110 0 0
    end location
    response 5
        title="ANSI standard (1977) neutron flux-to-dose-rate factors"
        specialDose=9029
    end response
    distribution 1
        title="Cf-252 neutrons, Watt spectrum a=1.025 MeV and b=2.926/MeV"
        special="wattSpectrum"
        parameters 1.025 2.926 end
    end distribution
end definitions

'-------------------------------------------------------------------------------
' Sources Block
'-------------------------------------------------------------------------------
read sources
    src 1
        title="Cf-252 neutrons"
        neutrons  strength=4.05E+07
        cuboid  0.01 0.01  0 0  0 0
        eDistributionID=1
    end src
end sources

'-------------------------------------------------------------------------------
' Tallies Block
'-------------------------------------------------------------------------------
read tallies
    pointDetector 1
        title="center of detector"
        neutron
        locationID=1
        responseID=5
    end pointDetector
end tallies

'-------------------------------------------------------------------------------
```

(continues on next page)

(continued from previous page)

```
' Parameters Block
'--------------------------------------------------------------------------------
read parameters
    randomSeed=00003ecd7b4e3e8b
    perBatch=9500 batches=20
    neutrons  noPhotons
    fissionMult=0  secondaryMult=0
end parameters

end data
end
```

Output

For the 20 cm case, Monaco reports the following results:

```
Neutron Point Detector 1.  center of detector
                       average      standard     relat      FOM    stat checks
    tally/quantity       value      deviation    uncert   (/min)   1 2 3 4 5 6
    ------------------  ----------  ----------  -------  --------  ----------
    uncollided flux     1.06273E+01  2.56481E-02  0.00241
    total flux          2.43530E+02  4.58843E+00  0.01884  1.39E+02  X X X X X X
    response 5          1.30398E-02  2.58938E-04  0.01986  1.26E+02  X X X X X X
    ------------------  ----------  ----------  -------  --------  ----------
```

The plots generated by the "read plots" block are shown in Fig. 8.2.36

To compute the attenuation that Ueki measured, this dose rate can be divided by the dose rate from another Monaco calculation using a slab thickness of 0 cm-the unattenuated case. Combining the results of nine Monaco runs (eight different thicknesses and the unattenuated case), which were allowed to run long enough to achieve 1% uncertainty, gives the results shown in Table 8.2.21. Note how the runtime of the Monaco calculation increases with increasing shield thickness.

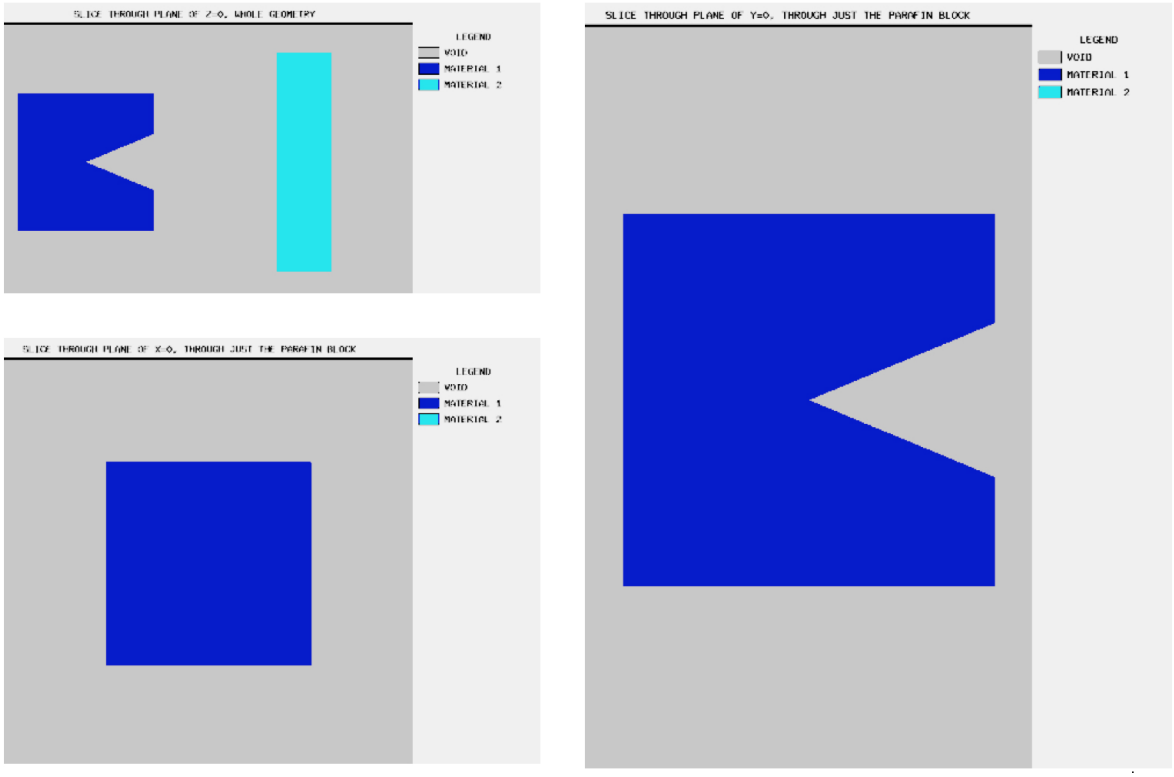


Fig. 8.2.36: Images (*.png files) generated with the “read plot” block.

Table 8.2.21: Measured and calculated dose attenuation rate for graphite

Thickness	Ueki		Monaco		C/E
(cm)	et al.	minutes	value	rel unc	
2	0.8288	10	0.8679	0.0072	1.05
5	0.7217	20	0.7383	0.0086	1.02
10	0.5261	40	0.5387	0.0086	1.02
15	0.3649	60	0.3837	0.0089	1.05
20	0.2532	20	0.2593	0.0199	1.02
25	0.1705	120	0.1774	0.0095	1.04
30	0.1126	180	0.1206	0.0093	1.07
35	0.0742	300	0.0806	0.0088	1.09

Adding a mesh tally

```
read definitions
    ...
    gridGeometry 7
        title="large meshes in paraffin, 5 cm mesh for shield thicknesses"
        xLinear 5 -25 25
        xLinear 12 30 90
        xplanes 100 110 120 -30 end
        yplanes -50 -40 40 50 end
```

(continues on next page)

(continued from previous page)

```
        yLinear 7 -35 35
        zplanes -50 -40 40 50 end
        zLinear 7 -35 35
    end gridGeometry
end definitions

read tallies
    ...

    meshTally 1
        title="example mesh tally"
        neutron
        gridGeometryID=7
        responseID=5
        noGroupFluxes
    end meshTally
end tallies
```

and viewing the resulting *.3dmap file with the Mesh File Viewer can produce the image shown in Fig. 8.2.37.

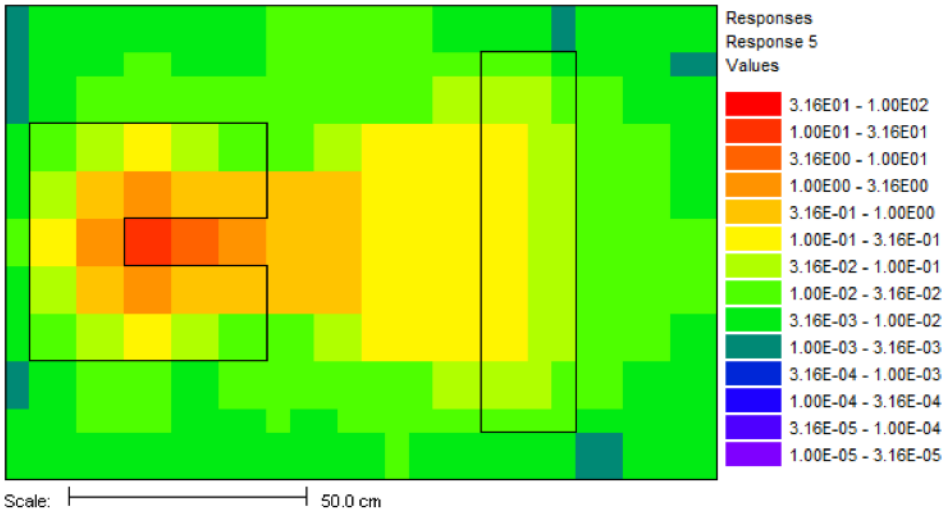


Fig. 8.2.37: Mesh tally showing neutron dose rate (rem/hr)

8.2.5.5 Simple shielding demonstration with line spectra

Sources containing line data are difficult to represent in multigroup. Comparing real measurements of line data to multigroup calculations is also difficult. This sample problem is not the simulation of a real measurement but is just a simple demonstration of the differences between continuous-energy simulation and the multigroup approach.

Consider an isotropic cobalt-60 source on one side of a slab of tungsten (5 cm thick) and the goal is to compute the photon flux and dose on the other side of the slab. Since the continuous-energy library does not have a built-in structure that could be used by tallies, the user needs to supply one (otherwise tally information is recorded in one bin covering the entire range of energy). The user can construct an energy boundaries object in the definitions block that is uniformly spaced, logarithmically spaced, based on one of the multigroup libraries or any combination of the above.

Input file

The sample problem `monaco.wSlab.inp`, is located in the `samples\input` directory. The materials input consist just of tungsten, in its natural isotopic abundances.

```
read crossSections
    ceLibrary="ce_v7.1_endf.xml"
    mixture 1
        element  74182 1.531781E-02
        element  74183 8.202875E-03
        element  74184 1.806470E-02
        element  74186 1.671717E-02
    end mixture
end crossSections
```

The geometry is a simple slab of tungsten, with a small air region on one side for a region tally.

```
read geometry
    global unit 1
        cuboid 10   2.5  -2.5  25.0 -25.0
                             25.0 -25.0
        cuboid 20   6.0   4.0   1.0  -1.0
                              1.0  -1.0
        cuboid 30  10.0 -10.0  25.0 -25.0
                             25.0 -25.0

        media 1 1  10
        media 0 1  20          vol=8.0
        media 0 1  30 -10 -20
    boundary 30
end geometry
```

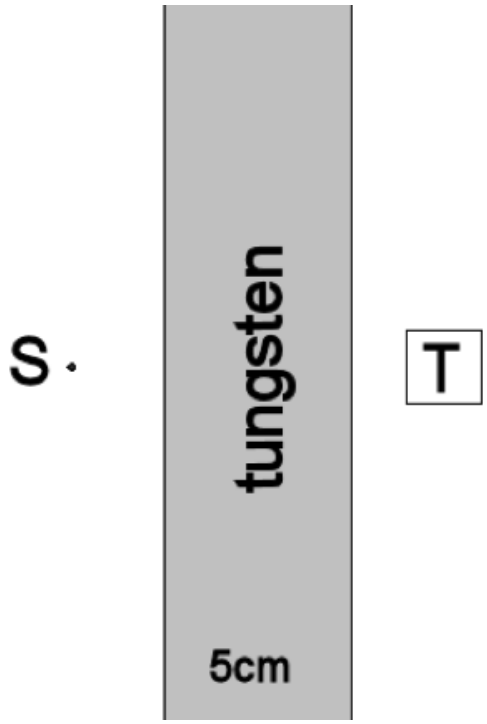


Fig. 8.2.38: Simple slab geometry with a source (S) on the left and a tally region (T) on the right.

In the definitions block, a location (for a point detector), the photon dose response function, the Co-60

line spectra and an energy boundaries structure are all defined. The energyBounds defined here has a base structure of 30 bins that are 50 keV wide, with three bins that are 2 keV wide at the cobalt line energies and the 511 keV annihilation gamma energy.

```
read definitions
    location 1
        position 5.0 0.0 0.0
    end location

    response 5
        title="ANSI standard (1977) photon flux-to-dose-rate factors"
        doseData=9504
    end response

    distribution 1
        title="cobalt-60 gammas/decay"
        discrete   347140   826100 1173228 1332492  2158570    2505692 end
        truepdf  0.000075 0.000076 0.9985 0.999826 0.000012 0.00000002 end
    end distribution


    energyBounds 1
        linear 30 0.00e6 1.50e6
        bounds  0.510e+6 0.512e+6   1.172e6 1.174e6   1.331e6 1.333e6 end
    end energyBounds
end definitions
```

The source is a simple point source 5 cm to the left of the slab. Because the distribution of cobalt gamma rays was entered as gammas per decay, the keyword *useNormConst' will set the source strength to be the total of the energy distribution - about 2 photons/decay. The `multiplier' keyword is used to multiply that strength by 37 :math:*times` 10^9 decays/sec to get 1 Ci.

```
read sources
    src 1
        title="one Ci of cobalt-60"
        useNormConst
        multiplier=37e9
        sphere 0.0   origin x=-5.0 y=0.0 z=0.0
        photons
        eDistributionID=1
    end src
end sources
```

Both a region tally and a point detector tally are defined. Parameters are set for a 10 minute calculation.

```
read tallies
    pointDetector 15
        photon
        locationID=1
        responseID=5
        energyBoundsID=1
    end pointDetector
    regionTally 24
        photon
        unit=1 region=2
        responseID=5
        energyBoundsID=1
    end regionTally
end tallies

read parameters
    randomSeed=00003ecd7b4e3e8b
    perBatch=582000 batches=90
```

(continues on next page)

(continued from previous page)

```
    noNeutrons  photons
    fissionMult=0  secondaryMult=0
end parameters
```

Output

Results for the dose rate from this continuous-energy calculation are shown in Fig. 8.2.39, along with results from two multigroup calculations and an MCNP calculation. To simulate the same type of physics used in the continuous-energy SCALE, the MCNP calculation used a photon cutoff energy of 0.01 MeV, EMCPF=100 MeV (detailed physics), IDES=1 (no electrons/no bremsstrahlung), NOCOH=0 (coherent scattering occurs), ISPN=0 (no photonuclear collisions), and NODOP=1 (no Doppler energy broadening).

The results from the SCALE multigroup calculations differ by 20%, due to how the source and the dose response function are represented. The 19-group structure represents the 1.33 MeV line with the 1.33-1.66 MeV, so it is expected that the computed dose will be high. With the 47-group structure, the 1.17 MeV line is represented by the 1-1.2 MeV group (too low — lower dose will result) and the 1.33 MeV line is represented by the 1.33-1.44 MeV group (too high — higher dose will result). The multigroup results are well-converged — just not as accurate as desired. The continuous-energy results should be more accurate.

	Time	Point Detector		Region Tally	
	(min)	rem/hr	rel unc	rem/hr	rel unc
MCNP	97.8	3.1088	0.0039	3.1376	0.0051
SCALE CE	89.9	3.1886	0.0027	3.0949	0.0136
SCALE 47	89.9	2.9284	0.0012	3.0144	0.0077
SCALE 19	90.0	3.6705	0.0010	3.7414	0.0065

Fig. 8.2.39: Dose rates for the cobalt-60/tungsten problem using different energy treatments.

Fig. 8.2.40 and Fig. 8.2.41 show the results of the point detector and region tally flux as a function of energy, for this problem as well as two multigroup calculations and an MCNP calculation. Tables M and N list the flux values of the cobalt lines and the annihilation gamma for the two tally types.

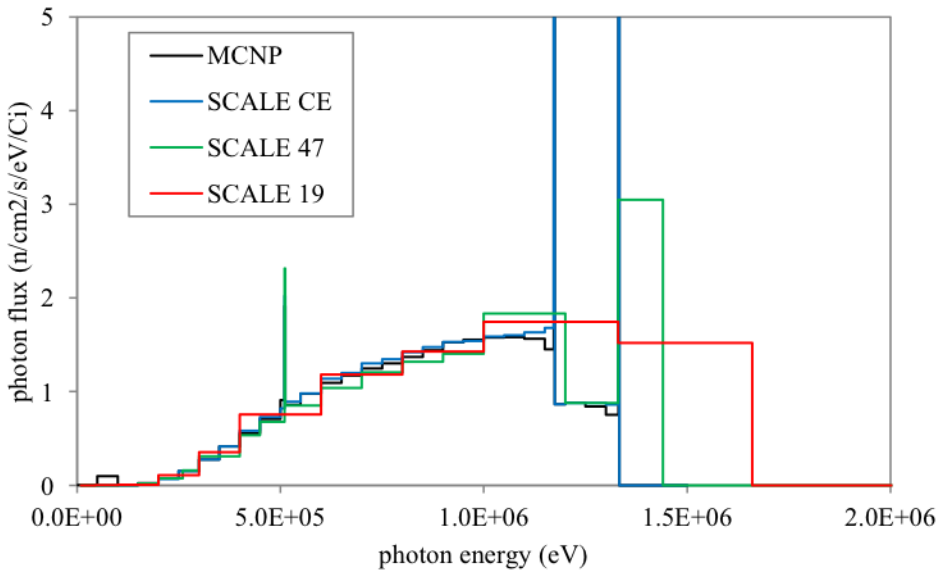


Fig. 8.2.40: Point detector fluxes for the cobalt-60/tungsten problem using different energy treatments.

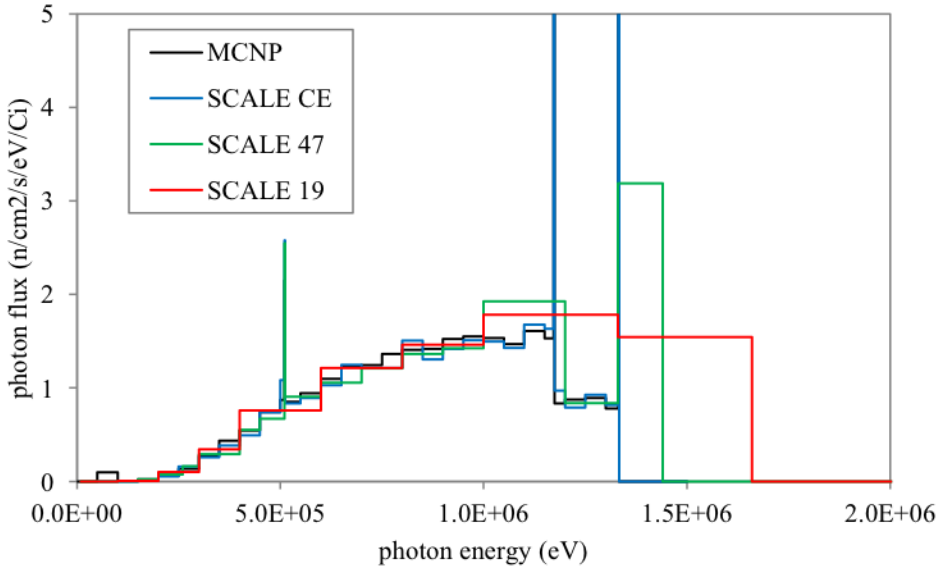


Fig. 8.2.41: Region tally fluxes for the cobalt-60/tungsten problem using different energy treatments.

	511000 eV		1173230 eV		1332490 eV	
	/cm^2/s	rel unc	/cm^2/s	rel unc	/cm^2/s	rel unc
MCNP	1.91E+00	0.0320	9.53E+01	0.0125	1.41E+02	0.0093
SCALE CE	2.02E+00	0.0382	9.94E+01	0.0058	1.42E+02	0.0041
SCALE 47	2.31E+00	0.0149				
SCALE 19						

Fig. 8.2.42: Region tally fluxes for the cobalt-60/tungsten problem using different energy treatments.

	511000 eV		1173230 eV		1332490 eV	
	/cm^2/s	rel unc	/cm^2/s	rel unc	/cm^2/s	rel unc
MCNP	2.15E+00	0.0945	9.60E+01	0.0145	1.44E+02	0.0118
SCALE CE	2.58E+00	0.2231	1.01E+02	0.0402	1.38E+02	0.0345
SCALE 47	2.55E+00	0.1200				
SCALE 19						

Fig. 8.2.43: Region tally flux values for the lines.

9. DETERMINISTIC TRANSPORT

Introduction by S. M. Bowman

SCALE deterministic transport capabilities enable criticality safety, depletion, sensitivity, and uncertainty analysis, as well as hybrid approaches to Monte Carlo analysis. SCALE provides a one-dimensional (1D) transport solver for eigenvalue neutronics and fixed source neutron-gamma analysis with XSDRN, two-dimensional (2D) eigenvalue neutronics with NEWT, and a three-dimensional (3D) transport solver for hybrid acceleration of Monte Carlo fixed source and eigenvalue calculations with Denovo. Generally, the use of these transport solvers in SCALE is best accessed through the capability specific sequences: CSAS for criticality safety, TRITON for 1D and 2D depletion, TSUNAMI-1D for sensitivity and uncertainty analysis, and MAVRIC for 3D fixed source hybrid Monte Carlo analysis.

XSDRN

XSDRN is a multigroup discrete-ordinates code that solves the 1D Boltzmann equation in slab, cylindrical, or spherical coordinates. Alternatively, the user can select P1 diffusion theory, infinite medium theory, or Bn theory. A variety of calculational types is available, including fixed source, eigenvalue, or search calculations. In SCALE, XSDRN is used for several purposes: eigenvalue (k_{eff}) determination; cross section collapsing; and computation of fundamental-mode or generalized adjoint functions for sensitivity analysis.

NEWT

NEWT (New ESC-based Weighting Transport code) is a multigroup discrete-ordinates radiation transport computer code with flexible meshing capabilities that allow 2D neutron transport calculations using complex geometric models. The differencing scheme employed by NEWT-the Extended Step Characteristic approach-allows a computational mesh based on arbitrary polygons. Such a mesh can be used to closely approximate curved or irregular surfaces to provide the capability to model problems that were formerly difficult or impractical to model directly with discrete-ordinates methods. Automated grid generation capabilities provide a simplified user input specification in which elementary bodies can be defined and placed within a problem domain. NEWT can be used for eigenvalue, critical-buckling correction, and source calculations, and it can be used to prepare collapsed weighted cross sections in AMPX working library format.

Like other SCALE modules, NEWT can be run as a standalone module or as part of a SCALE sequence. NEWT has been incorporated into SCALE TRITON control module sequences. TRITON can be used simply to prepare cross sections for a NEWT transport calculation and then automatically execute NEWT. TRITON also provides the capability to perform 2D depletion calculations in which the transport capabilities of NEWT are combined with multiple ORIGEN depletion calculations to perform 2D depletion of complex geometries. In the TRITON depletion sequence, NEWT can also be used to generate lattice-physics parameters and cross sections for use in subsequent nodal core simulator calculations.

DENOVO

Denovo [DTESSC10], is a parallel 3D discrete-ordinates code available in SCALE as part of two control module sequences for different applications, as described below. Because Denovo can only be run in SCALE via the Monaco with Automated Variance Reduction using Importance Calculations (MAVRIC) or Denovo Eigenvalue Calculation (DEVC), it is not documented separately in the section entitled "Deterministic Transport" in this manual.

The MAVRIC hybrid Monte Carlo radiation shielding sequence employs the Consistent Adjoint Driven Importance Sampling (CADIS) and Forward-Weighted CADIS (FW-CADIS) methodologies. Denovo is used

to generate adjoint (and, for FW-CADIS, forward) scalar fluxes for the CADIS methods in MAVRIC. This adjoint flux information is then used by MAVRIC to construct a space- and energy-dependent importance map (i.e., weight windows) to be used for biasing during Monte Carlo particle transport and as a mesh-based biased source distribution. For use in MAVRIC/CADIS, it is highly desirable that the S_N code be fast, positive, and robust. The phase-space shape of the forward and adjoint fluxes, as opposed to a highly accurate solution, is the most important quality for Monte Carlo weight-window generation. Accordingly, Denovo provides a step-characteristics spatial differencing option that produces positive scalar fluxes as long as the source (volume plus in-scatter) is positive. Denovo uses an orthogonal, nonuniform mesh that is ideal for CADIS applications because of the speed and robustness of calculations on this mesh type. Denovo can be run stand-alone in MAVRIC to perform fixed source calculations using the *PARM=forward* (for forward Denovo) or *PARM=adjoint* (for adjoint Denovo). See the MAVRIC chapter for details.

The other sequence that uses Denovo is the DEVC sequence. DEVC generates a reasonably accurate starting source through a Denovo eigenvalue calculation. Denovo can be run stand-alone in DEVC for calculating criticality eigenvalue problems. This sequence reads an input file very similar to a CSAS6 input file that contains an extra block of input for describing the Denovo mesh grid and calculational parameters.

9.1 XSDRNPM: A ONE-DIMENSIONAL DISCRETE-ORDINATES CODE FOR TRANSPORT ANALYSIS

L. M. Petrie, N. M. Greene, M. L. Williams

ABSTRACT

XSDRNPM is a discrete-ordinates code that solves the one-dimensional Boltzmann equation in slab, cylindrical, or spherical coordinates. Alternatively, the user can select P_1 diffusion theory, infinite medium theory, or B_n theory. A variety of calculational types is available, including fixed source, eigenvalue, or "search" calculations. In SCALE, XSDRNPM is used for several purposes: eigenvalue (k-effective) determination, cross-section collapsing, shielding analysis, computation of fundamental-mode or generalized adjoint functions for sensitivity analysis, and for producing bias factors for use in Monte Carlo shielding calculations.

ACKNOWLEDGMENTS

W. W. Engle has been very generous with the use of his notes on S_n theory and on discussing details of various procedures in XSDRNPM which were lifted directly from his ANISN program.

The authors also wish to thank R. H. Odegaarden (the former technical monitor at the U.S. Nuclear Regulatory Commission) who supplied the necessary incentives for completing this report.

9.1.1 INTRODUCTION

XSDRNPM is a one-dimensional (1-D) discrete-ordinates transport code and is the latest in a series of codes in the XSDRN [XSDGCJ69] family. As such, it contains several unique characteristics, as will be detailed in this report, though a large portion of the theoretical bases and intended uses of the program are the same for all versions.

9.1.1.1 Functions performed

The function of XSDRNPM is twofold: (1) perform a 1-D discrete-ordinates calculation in slab, cylindrical, or spherical geometry (optionally, a 1-D diffusion theory or infinite medium |Bn| calculation can be made), and (2) use the fluxes determined from its spectral calculation to collapse input cross sections and write these into one of several formats.

A great deal of flexibility is allowed in describing a problem for XSDRNPM. The number of spatial intervals, the number of energy groups, the number of nuclides, the quadrature order, the order of fits to the angular variation in basic cross sections are all arbitrary and are limited only by computer and monetary resources.

The flux calculation can be performed according to several options, including fixed source calculations, k-calculations, and dimension search calculations.

A variety of weighting options are allowed, including zone, cell, or a special "vein" weighting option which is described herein.

Background on XSDRNPM

Development of the XSDRN1 program started in the mid-1960s. The goal was to develop a program that would combine features from the GAM-II, [XSDJD63] ANISN, [XSDEJ67] and THERMOS [XSDHon61] programs in a more unified and general way than would be possible if one simply elected to use these codes individually.

The salient features to be retained from the programs were as follows:

GAM-II
: The Nordheim Integral Treatment was desired for resonance self-shielding; the generality of including cross sections for an arbitrary number of processes, along with the provisions for truncating zero or impossible transfers in the scattering matrices, was also a requirement.

ANISN
: One-dimensional discrete-ordinates or diffusion theory or infinite-medium theory was to be available to generate a spectrum for cross-section collapsing.

THERMOS
: The ability to perform detailed 1-D spectral calculations, including upscatter effects, was required for the thermal region.

The whole code was required to be dynamically dimensioned to allow calculations for arbitrary group structures, spatial structures, angular quadratures, etc.

The XSDRN program that embodied these features was released in 1969.

In the early 1970s, the Defense Nuclear Agency (DNA) initiated support for the AMPX system, which was to be a total cross-section generation system capable of performing all tasks necessary to take basic neutron and gamma-ray cross-section data and process these data into the proper form needed for weapons effects calculations. Since XSDRN already encompassed many of the features needed, it was selected as a basis for modules in the new system. In this case, experience gained in the original construction of XSDRN served to suggest that a more modular approach would have been better with independent tasks being done in separate, smaller, easier-to-manage programs. Therefore, the code was split into NITAWL-II (for resonance self-shielding and some basic cross-section data manipulation) and XSDRNPM (for spectral calculations and cross-section collapsing). In retrospect, if the AMPX development were initiated today, XSDRN would have been split even further, into perhaps as many as six or seven programs.

The XSDRNPM module differs from XSDRN in several respects:

- It will perform coupled neutron-gamma calculations.
- It allows any mixture to be represented to an arbitrary order of anisotropic representation, whereas XSDRN only allowed through order 3.

- It will perform an adjoint calculation, whereas the option was never provided in XSDRN. In 2010, a generalized adjoint solution was also added.
- It is considerably more efficient in the manner in which data storage is used and, hence, will run much larger problems in less core storage.
- It employs improved thermal flux scaling techniques for better problem convergence.
- Input specifications have been reordered, and more defaults have been provided to make the use of this module easier.
- It will calculate S_n constants for any order for any of the three 1-D geometries available.
- Mixture-dependent fission spectra are calculated and used in XSDRNPM, which takes into account all fissionable nuclides in a problem.

AMPX was released in 1976, about the same time as the U.S. Nuclear Regulatory Commission (NRC) support for the SCALE system was initiated. Although separate versions of XSDRNPM were initially maintained for AMPX and SCALE, in recent years the same version is used for the two systems.

9.1.1.2 Applications in SCALE

XSDRNPM is used in several places in SCALE. In SAS1, XSDOSE uses fluxes from a 1-D shielding calculation to determine a dose rate. Within the CSAS5 and CSAS6 control modules, XSDRNPM is used in the sequences to perform eigenvalue calculations and cell weighting of cross sections. TSUNAMI-1D uses XSDRN to compute forward and adjoint fluxes (fundamental-mode and generalized adjoint) for sensitivity and uncertainty analysis.

Notes on the use of various spectral calculational options

As noted earlier, four options are available in XSDRNPM for calculating fluxes, k-effectives, etc.:

1. S_n theory,
2. diffusion theory,
3. infinite medium theory, and
4. B_n theory.

However, XSDRNPM is primarily an S_n code. The latter three options are provided for reasons of completeness and are not nearly as optimized as they would be in other codes for which these are the primary spectral calculation options.

Without a very detailed calculational study, it is perhaps impossible to be able to quantify the degree of adequacy or inadequacy of any of these methods for performing a particular problem. However, some general comments can be made which may provide some guidance with their selection.

First, S_n theory is the most correct of the options and will solve a larger class of problems. It is the most complicated and time-consuming of the four, but it still runs very fast for most cases. There are problems for which it (or some alternative method based on a solution of the Boltzmann equation) is the only one of the four methods which is adequate. Many shielding applications fall in this class. In deep-penetration problems, anisotropic effects can dominate, thus requiring an accurate treatment of the anisotropy of both flux and cross section. It is well known that diffusion theory is not very accurate when used to calculate systems involving regions of very dissimilar cross-section values, such as is the case when control rods are interspersed in a reactor core. Because of the anisotropy involved in gamma-ray problems, S_n theory should be used.

Diffusion theory, on the other hand, is certainly the most successful of the four methods in terms of the amount of use it has for designing reactors, etc. In cases involving reasonably large, homogeneous regions, it is generally adequate, such as is the case for a large class of "reactor" applications. For most problems, the diffusion theory option should run appreciably faster than S_n theory, since it has essentially one equation to solve, versus number-of-angles equations for S_n theory. This equation also can be explicitly solved using a matrix inversion procedure, whereas the S_n theory requires a more time-consuming iterative procedure. However, in many cases with large numbers of groups (200 to 300), the greater fraction of the calculational time can be spent calculating the scattering source terms, which tends to lessen the impact of time spent on a more correct theory. (This same observation can also be made of the infinite medium and B_n method.)

The infinite medium option is the fastest of the four methods and can be used safely to perform calculations for large homogeneous regions, wherein the spectrum may be needed to collapse cross sections. This option only determines the first moment of the flux, and is, therefore, quite suspect for many applications, such as calculating diffusion coefficients.

The B_n option shares many of the same restrictions as the infinite-medium method; however, this treatment does (as its name implies) use a buckling approximation to account for leakage from the large homogeneous region, thereby giving higher order flux moments that can be used, for example, to determine diffusion coefficients.

9.1.1.3 Selection of output cross-section library formats

XSDRNPM will, on option, collapse cross sections and write the collapsed sets into four different formats:

1. ANISN3 BCD Library,
2. ANISN3 Binary Library,
3. CCCC [XSDCar74] ISOTXS Library, or
4. AMPX [XSDWWCD15] Working Library.

The choice of the output cross-section format is determined by the computer code that will use the data. XSDRNPM always produces an AMPX working library when cross sections are collapsed, and all other formats are produced by reformatting data from this library. Therefore, for archival purposes, if a collapsed library is to be saved, the working format is the best choice, because it is the most general of those provided. AMPX working libraries are used by all multi-group transport codes currently in SCALE, including DENOVO (3D orthogonal mesh discrete ordinates code), NEWT (2D arbitrary mesh discrete ordinates code), and KENO and MONOCO (multigroup Monte Carlo codes. Stand-alone modules exist for converting AMPX working libraries to the other formats.

ANISN formats are used by older ORNL transport codes such as ANISN (a 1-D discrete-ordinates code), by DORT/TORT [XSDRSCEJ79] [two-dimensional (2-D) and three dimensional (3D) discrete-ordinates codes], and by MORSE [XSDEme75] (a multigroup Monte Carlo code). The formats are quite comprehensive and can handle coupled neutron-gamma calculations, arbitrary orders of anisotropy, upscattering, etc. The major shortcoming of the format is its lack of internal documentation as to its structure (e.g., no provisions exist for specifying where a particular kind of cross section is located in the library or even if it is included). ANISN libraries can be produced in a free-form card-image BCD format or in a binary form.

The CCCC (Committee on Computer Code Coordination) ISOTXS file is a format for neutron cross sections that is one of several "standard interfaces" developed to facilitate the exchange of data between different computer codes. It is a self-defined format, which has provisions for identifying cross sections in the library. Scattering matrices can be supplied for elastic, inelastic, and (n,2n) scattering.

9.1.2 THEORY AND PROCEDURES

This section describes the models and procedures which are employed in XSDRNPM.

9.1.2.1 One-dimensional discrete-ordinates theory

The time-independent Boltzmann transport equation can be written:

$$\vec{\Omega} \bullet \nabla\psi(\vec{r}, E, \vec{\Omega}) + \sum_i (\vec{r}, E)\psi(E, \vec{r}, \vec{\Omega}) = S(\vec{r}, E, \vec{\Omega}) \tag{9.1.1}$$

This expression is a balance condition that states simply that losses due to leakage (first term) and collisions (second term) must equal the source of neutrons, at some point in space r energy E, and in direction Ω per unit volume and energy and solid angle. Other terms in the expression are $\sum_t(r, E)$ the total macroscopic cross section of the medium, which is typically assumed isotropic, and the flux, $\psi(r, E, \Omega)$.

The source term $S(r, E, \Omega)$ has three components:

1. a scattering source, $S(r, E, \Omega)$,
2. a fission source, $F(r, E, \Omega)$, and
3. a fixed source, $Q(r, E, \Omega)$.

The scattering source is given by:

$$S(r, E, \Omega) = \int_0^{4\pi} d\Omega' \int_0^{\infty} dE' \sum_s (r, E' \rightarrow E, \Omega' \rightarrow \Omega)\, \psi(r, E', \Omega') \Bigg). \tag{9.1.2}$$

The fission source term, typically, is written

$$F(r, E, \Omega) = \frac{1}{4\pi k}\chi(r, E) \int_0^{4\pi} d\Omega' \int_0^{\infty} dE' \nu(r, E')\, \Sigma_f(r, E')\, \psi(r, E', \Omega'), \tag{9.1.3}$$

where $\sum_s (r, E' \rightarrow E, \Omega' \rightarrow \Omega)$ is the macroscopic scattering cross section per unit energy for scattering from energy E' to E, $\chi(r, E)$ is the fraction of the fission neutrons per unit energy produced at r and E, $\upsilon(r, E)$ is the average number of neutrons produced per fission, $\sum_f(r, E)$ is the macroscopic fission cross section and k is the "effective multiplication constant." Note that, as in the case of the total cross-section value, χ,Σf, and υ are assumed to be isotropic. XSDRN computes a weighted-averaged fission spectrum based on the fissionable materials at r.

Three common coordinate systems are shown in Fig. 9.1.1. XSDRNPM is a 1-D code, which means that in the case of the slab, it is calculating at points along one axis where the system is assumed to extend to infinity along the other two axes. If we assume a calculation along the x-axis, this says that there is no leakage in the y or z directions, and our directions by angles referenced to the x-axis. In the case of the cylinder, the length (z-axis) is infinite and the calculation is for points (shells) located at distance r from the central axis. For the sphere, the calculation is of shells located at radius, r, from the center of the spherical system.

Fig. 9.1.2 illustrates the 1-D coordinate systems for slabs, cylinders, and spheres. Note that the directions are cones in the case of the slab and sphere, whereas in the case of the cylinder, the same simple symmetries do not hold (a cone around the radius does not strike the next cylindrical shell at the same distance from a point on a radius) and the directions must be specifically described. Symmetries in the 1-D cylinder, however, allow one to only describe directions for one quadrant of the direction sphere about a point as will be noted in Sect. 9.1.2.2.

The 1-D geometries allow considerable simplification to be made to Eq. Eq. (9.1.1), especially in the leakage term $\bar{\Omega} \bullet \nabla\psi$. It is traditional to calculate the angular flux as a function of angles expressed in direction-cosine units; i.e., $\mu = \cos\phi$ and $\eta = \cos\xi$. This requires $\psi(x, E, \mu)$ for slabs, $\psi(x, E, \mu, \eta)$ for cylinders and $\psi(r, E, \mu)$ for spheres. Table 9.1.1 gives leakage terms expressed in conservation form for the three geometries.

Table 9.1.1: One-dimensional leakage terms.

Geometry	$\vec{\Omega} \bullet \nabla\psi$
Slab	$\mu\frac{\partial\psi}{\partial x}$
Cylinder	$\frac{\mu}{r}\frac{\partial(r\psi)}{\partial r} - \frac{1}{r}\frac{\partial(\eta\psi)}{\partial\phi}$
Sphere	$\frac{\mu}{r^2}\frac{\partial(r^2\psi)}{\partial r} + \frac{1}{r}\frac{\partial[(1-\mu^2)\psi]}{\partial\mu}$

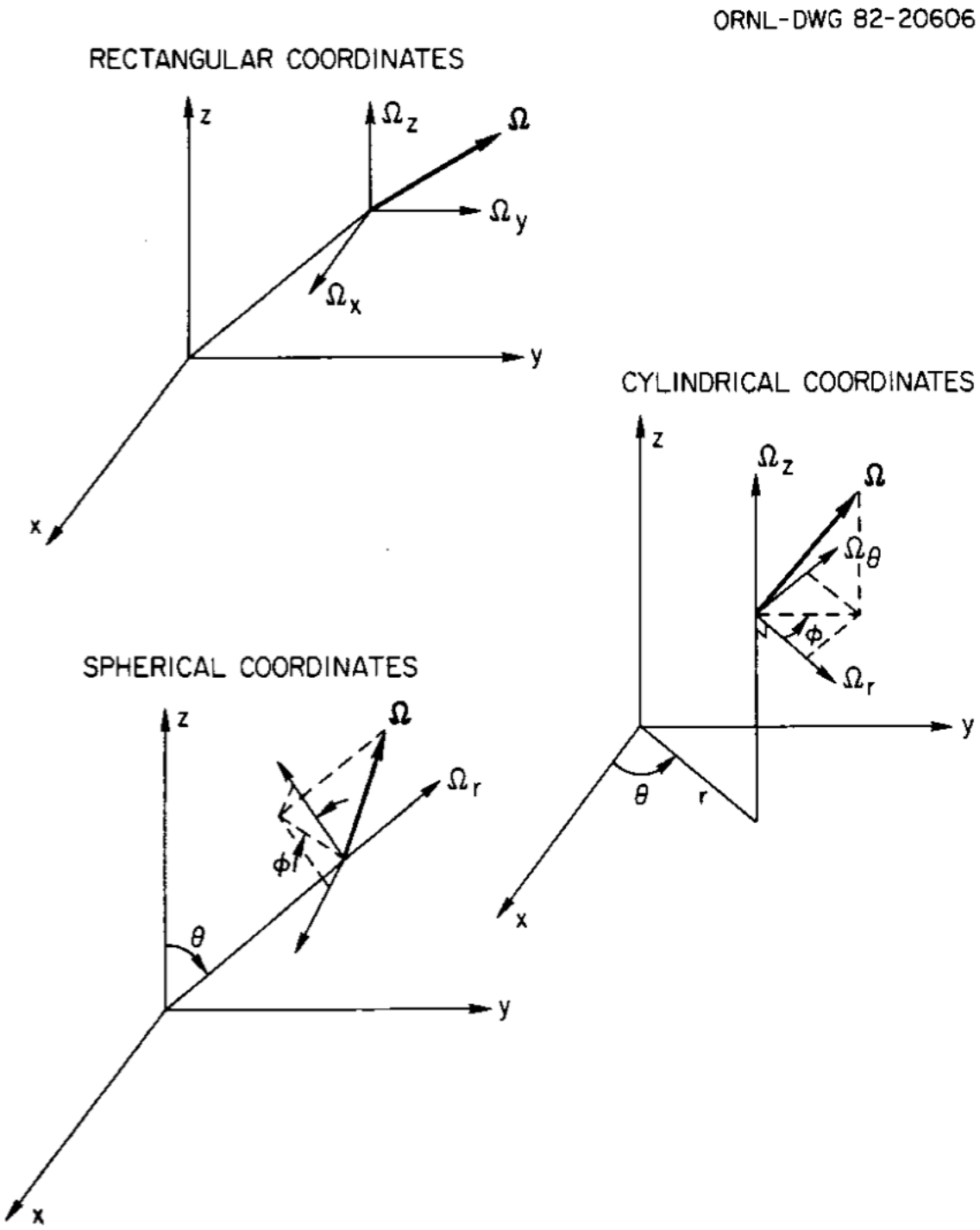


Fig. 9.1.1: Three common coordinate systems.

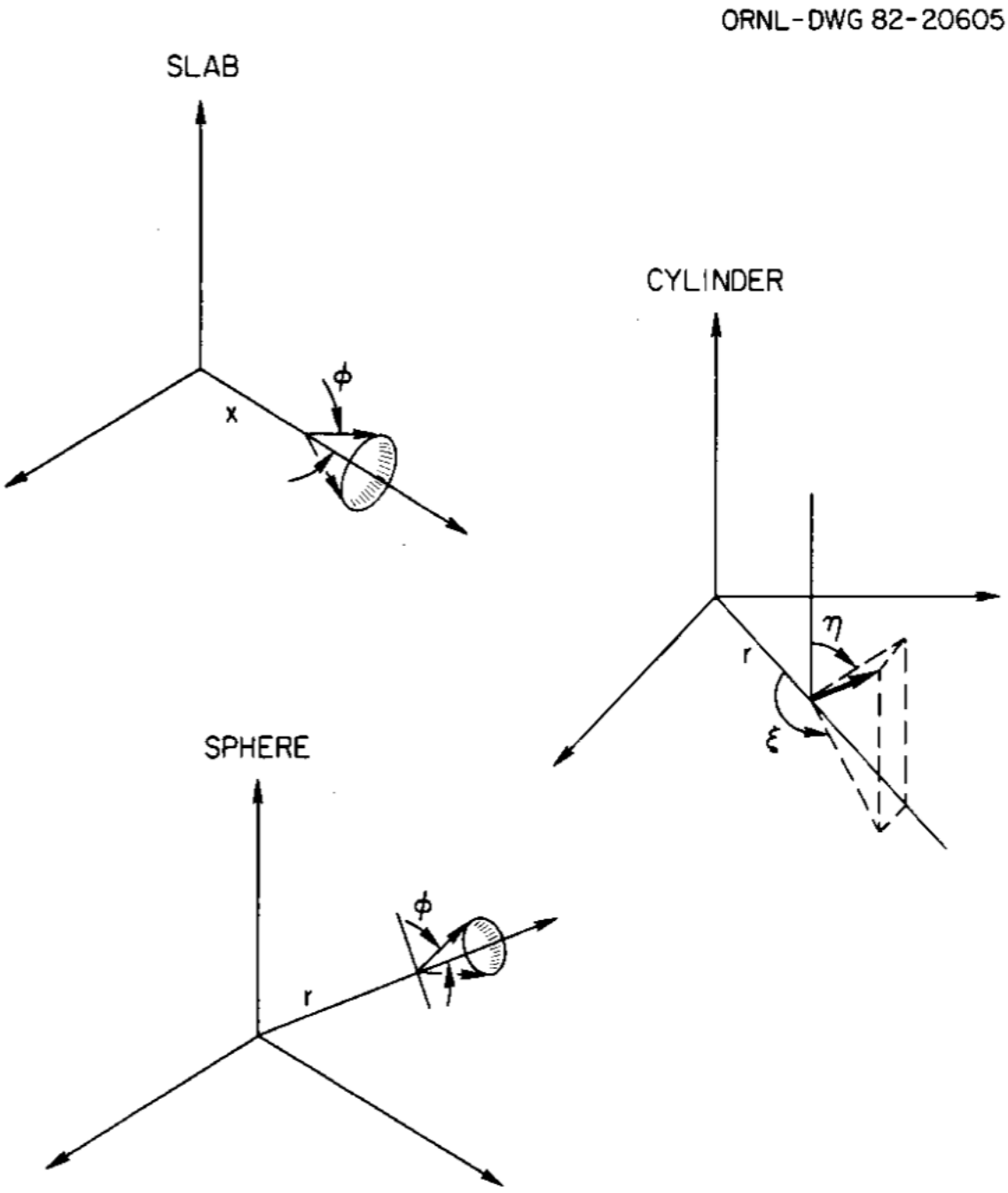


Fig. 9.1.2: Three 1-D coordinate systems.

9.1.2.2 Multigroup one-dimensional Boltzmann equation

In multigroup schemes, the continuous-energy (CE) balance equations are converted to multigroup form by first selecting an energy structure and then writing a multigroup equivalent of the point equation which requires multigroup constants that tend to preserve the reaction rates that would arise from integrating the CE equations by group. First we define the following multigroup values for g,

$$\psi_g(x,\mu) = \int_g dE\psi(x,E,\mu) \tag{9.1.4}$$

and

$$\psi_g(x) = \int_{-1}^{1} d\mu\psi_g(x,\mu) \tag{9.1.5}$$

and

$$\Sigma_{tg}(x) = \frac{\int_g dE\Sigma_{tg}(x,E)W(x,E)}{\int_g dEW(x,E)}, \tag{9.1.6}$$

where *W(x.E)* is the weighting function used to compute the multigroup cross sections at a particular location. To rigorously conserve reaction rates, the weight function should be angle-dependent, but this causes the multigroup cross section to vary with direction; therefore the usual approach is to represent the weight function by an approximation to the scalar flux spectrum. In energy ranges where the CE cross sections have fine-structure due to resonances, the multigroup data must be self-shielded prior to the multigroup transport calculations.

The following multi-group form of 1-D equation can be derived for the slab case:

$$\mu\frac{\partial\psi_g(x,\mu)}{\partial x} + \sum_{tg}(x)\psi_g(x,\mu) = S_g(x,\mu) + F_g(x,\mu) + Q_g(x,\mu). \tag{9.1.7}$$

The equations for the cylinder and sphere are essentially the same, in this notation, except for the differences in the leakage terms from Table 9.1.1.

In Eq. Eq. (9.1.7) , S_g, F_g, and Q_g are the scattering, fission, and fixed sources, respectively. The scattering term is discussed in Sect. 9.1.2.3. The multigroup form of the fission source is

$$F_g(x,\mu) = \frac{\chi_g}{2\pi k}\sum_{g'}\overline{\nu\sum_{fg'}}(x)\psi_{g'}(x), \tag{9.1.8}$$

where χ_g is the fraction of the fission neutrons that are produced in group g, and $\overline{\nu\Sigma_{fg'}}$ is the average of the product of ν, the average number of neutrons produced per fission and Σ_f, the fission cross section.

9.1.2.3 Scattering source term

In discrete-ordinates theory, one typically calculates the Legendre moments of the flux, $\psi_{g,l}$, defined for slab and spherical geometries by

$$\psi_{g,l} = \frac{1}{2}\int_{-1}^{1} d\mu\psi_g(\mu)P_1(\mu). \tag{9.1.9}$$

Cylindrical geometry has a similar expression containing spherical harmonic functions rather than Legendre polynomials, shown in the next section.

The group-to-group scattering coefficients are, themselves, fit with Legendre polynomials, such that

$$\sigma\left(g' \rightarrow g, \mu\right) = \sum_{l=0}^{ISCT} \frac{2l+1}{2} \sigma_l\left(g' \rightarrow g\right) P_l(\mu). \tag{9.1.10}$$

In this example, we have a fit of order ISCT.

Note: AMPX cross-section libraries contain the $2l + 1$ factor in the $\sigma_t\left(g' \rightarrow g\right)$ matrix.

Slab and Spherical Geometries

Because of the symmetries in 1-D slabs and spheres, only one angle is needed to describe a "direction." In the case of the slab, the angle is taken with reference to the x-axis, while for the sphere; it is with reference to a radius vector between the point and the center of the sphere. This means that the flux can be expanded in ordinary Legendre polynomials, such that

$$\begin{array}{l} \psi(r, E, \mu) = \sum_{l=0}^{\infty} \psi_l(r, E) P_l(\mu) \\ \psi_l(r, E) = \int_{-1}^{1} \frac{d\mu}{2} P_l(\mu) \psi(r, E, \mu) \end{array} . \tag{9.1.11}$$

When Eq. Eq. (9.1.11) and Eq. Eq. (9.1.10) are introduced into Eq. Eq. (9.1.2), the following expression is derived for the scattering source:

$$S\left(r, E, \mu\right) = 2\pi P_l(\mu) \int_0^{\infty} dE' \int_{-1}^{1} d\mu' \sum_{l=0}^{ISCT} \frac{2l+1}{2} \sum_{s_l} \left(r, E' \rightarrow E\right) P_l\left(\mu'\right) \psi_l\left(r, E'\right) \tag{9.1.12}$$

where *ISCT* is the order of fit to the fluxes and cross sections.

Cylindrical Geometry

The situation is more complicated in the case of the 1-D cylinder where the flux (and cross section) must be given as a function of two angles. Consider Fig. 9.1.3.

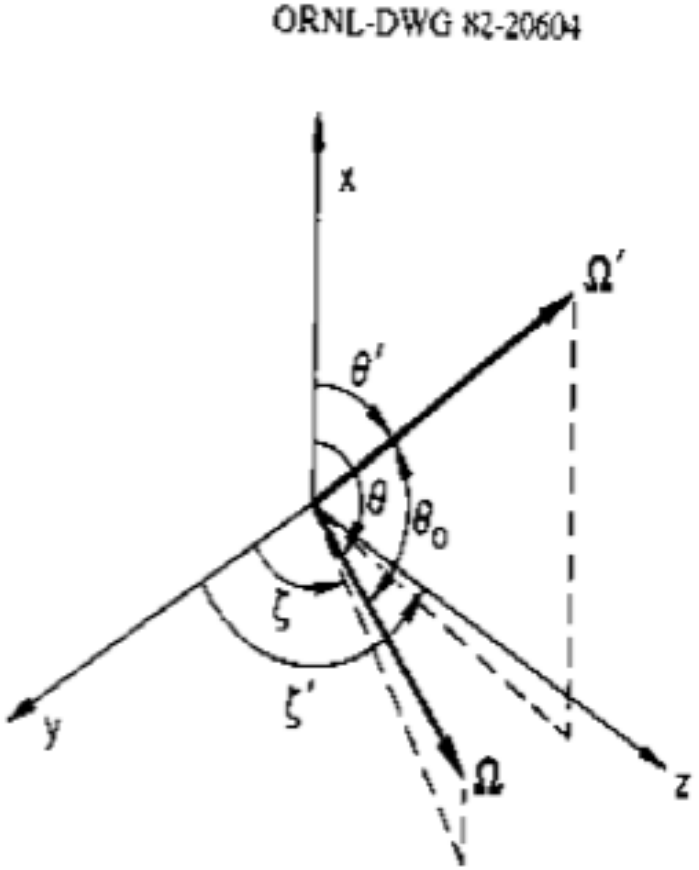


Fig. 9.1.3: One-dimensional cylindrical scattering coordinates.

The addition theorem for associated Legendre polynomials can be used to transform from scattering angle coordinates to the real coordinates required in the cylindrical case:

$$P_l(\mu_0) = \sum_{n=-1}^{1} \frac{(l-n)!}{(l+n)!} P_l^n(\mu) P_l^n(\mu') e^{in(\zeta-\zeta')}, \tag{9.1.13}$$

where $\mu_0 = \cos\theta_0$; $\mu = \cos\theta$ and $\mu' = \cos\theta'$.

If we note that

$$\begin{aligned} \sigma_s(r, E' \to E, \Omega' \to \Omega) &= \sigma_s(r, E' \to E), P_l(\Omega' \bullet \Omega) \\ &= \sigma_s(r, E' \to E), P_l(\mu_0) \end{aligned} \tag{9.1.14}$$

Eq. Eq. (9.1.13) can be introduced into Eq. Eq. (9.1.2) to yield

$$S(r,E,\mu) = \int_0^\infty dE' \int_{-1}^{1} d\mu' \int_0^{2x} d\zeta \psi(r,E',\mu',\zeta') \sum_{l=0}^{ISCT} \frac{2l+1}{2} \sigma_{s_l}(r,E' \rightarrow E) \times \sum_{n=-1}^{l} \frac{(l-n)!}{(l+n)!} P_l^n(\mu) P_l^n(\mu') e^{in(\zeta-\zeta')} \quad . \tag{9.1.15}$$

Now it is convenient to recall that

$$\cos x = \frac{e^{+ix} + e^{-ix}}{2}, \tag{9.1.16}$$

which can be introduced into Eq. Eq. (9.1.15) and rearranged to give

$$\begin{aligned} S(r,E,\mu) = & \sum_{l=0}^{ISCT} \frac{2l+1}{2} \int_0^\infty dE' \sigma_{s_l}(r,E' \rightarrow E) \left[P_l(\mu) \int_{-1}^{1} d\mu' \int_0^{2\pi} d\zeta \psi(r,E',\mu',\zeta') P_l(\mu') \right] \\ & + \sum_{n=1}^{l} 2 \frac{(l-n)!}{(l+n)!} P_l^n(\mu) \left[\int_{-1}^{1} d\mu' \int_0^{2\pi} d\zeta \psi(r,E',\mu',\zeta') P_l^n(\mu') \cos\left[n(\zeta-\zeta')\right] \right] \end{aligned} \quad . \tag{9.1.17}$$

We now define moments of the flux, ψ_l by

$$\phi_l(r,E) = \int_{-1}^{1} d\mu' \int_0^{2\pi} d\zeta \psi(r,E,\mu',\zeta') P_1(\mu') \tag{9.1.18}$$

It is also convenient to make use of the trigonometric relationship

$$\cos\left[n(\zeta-\zeta')\right] = \cos n\zeta \cos n\zeta' + \sin n\zeta \sin n\zeta', \tag{9.1.19}$$

and

$$\psi_l^n(r,E) = \sqrt{2 \frac{(l-n)!}{(l+n)!} \int_{-1}^{1} d\mu'} \int_0^{2\pi} d\zeta \psi(r,E,\mu',\zeta') P_l^n(\mu') \sin n\zeta' \tag{9.1.20}$$

With a 1-D cylinder, the flux is symmetric in ζ; therefore, it is an even function, and the terms involving sin n ζ will vanish. This fact yields the following expression for Eq. Eq. (9.1.17) :

$$\begin{aligned} S(r,E,\mu) = & \sum_{l=0}^{ISCT} \frac{2l+1}{2} \int_0^\infty dE' \sigma_{s_l}(r,E' \rightarrow E) \\ & P_l(\mu) \psi_l(r,E') + \sum_{n=l}^{l} \sqrt{2 \frac{(l-n)!}{(l+n)!}} P_n^l(\mu) \cos n\zeta \psi_l^n(r,E) \Big] \end{aligned} \tag{9.1.21}$$

We observe further that for an even function in ζ, the odd l and odd (l-n) moments must all vanish, such that the following moments are nonzero for various orders of scattering:

ISCT	Nonzero flux moments
0	ψ_0
1	ψ_0, ψ_1^1
2	$\psi_0, \psi_1^1, \psi_2, \psi_2^2$
3	$\psi_0, \psi_1^1, \psi_2, \psi_2^2, \psi_3^1, \psi_3^3$
4	$\psi_0, \psi_1^1, \psi_2, \psi_2^2, \psi_3^1, \psi_3^3, \psi_4, \psi_4^2, \psi_4^4$

In general, [ISCT(ISCT + 4)/4] + 1 flux moments are required.

9.1.2.4 Discrete-ordinates difference equations

In formulating the S_n equations, several symbols are defined which relate to a flux in an energy group g, in a spatial interval i, and in an angle m.

Typically, the flux is quoted as an integral of the flux in an energy group g, whose upper and lower bounds are E_g^U and E_g^L respectively.

$$\psi_g = \int_{E_g^L}^{E_g^U} dE\psi(E). \tag{9.1.22}$$

A mechanical quadrature is taken in space, typically IM intervals with IM + 1 boundaries. Likewise, an angular quadrature is picked compatible with the particular 1-D geometry, typically MM angles with associated directional coordinates and integration weights.

The different equations are formulated in a manner which involves calculating so-called angular fluxes, $\psi_{g,i,m}$ at each of the spatial interval boundaries, and also cell-centered fluxes, $\psi_{g,i+1/2,m}$ at the centers of the spatial intervals. The centered fluxes are related to the angular boundary fluxes by "weighted diamond difference" assumptions as will be described below.

Units on angular fluxes are per unit solid angle w_m and per unit area. Units on the centered fluxes are track length per unit volume of the interval. In both cases the fluxes are integrated in energy over the group g.

The areas and volumes for the three geometries are listed in Table 9.1.2

Table 9.1.2: One-dimensional areas and volumes.

Geometry	Area	Volume
Slab	1.0	$x_{i+1} - x_i$
Cylinder	$2\pi r_i$	$\pi\left(r_{i+1}^2 - r_i^2\right)$
Sphere	$4\pi r^2$	$4/3\pi\left(r_{i+1}^3 - r_i^3\right)$

Discrete-ordinates equation for a slab

Consider a spatial cell bounded by (x_i, x_{i+1}) and write the loss term for flow through the cell in direction μ_m. The net flow in the x-direction out the right side is the product of the angular flux times the area times the solid angle times the cosine of the angle:

$$w_m\mu_m A_{i+1}\psi_{g,i+1,m}. \tag{9.1.23}$$

The net loss from the cell is the difference between the flow over both boundaries:

$$w_m\mu_m\left(A_{i+1}\psi_{g,i+1,m} - A_i\psi_{g,i,m}\right). \tag{9.1.24}$$

The loss in the spatial cell due to collisions is given by the product of the centered angular flux (in per unit volume units) times the total macroscopic cross section times the solid angle times the volume:

$$w_m\sigma_{g,i+1/2}V_i\psi_{g,i+1/2}, m. \tag{9.1.25}$$

The sources in direction μ_m are given by the product of the solid angle times the interval volume times the volume-averaged source (sum of fixed, fission, and scattering) in the direction m:

$$w_m V_i S_{g,i+1/2}, m. \tag{9.1.26}$$

The slab equation is obtained by using Eqs. Eq. (9.1.24), Eq. (9.1.25), and Eq. (9.1.26) and substituting proper values for area and volume:

$$w_m\mu_m\left(\psi_{g,i+l,m}-\psi_{g,i,m}\right)+w_m\sigma_{g,i+1/2}\psi_{g,i+1/2,m}\left(x_{i+l}-x_i\right)=w_mS_{g,i+1/2'm}\left(x_{i+l}-x_i\right). \tag{9.1.27}$$

In an MM angle quadrature set, there are MM of these equations and they are coupled through the assumption on how the cell-centered flux relates to the boundary angular fluxes, the sources, and the boundary conditions, as will be discussed later.

Discrete-ordinates equations for sphere and cylinder

The development of the equations for these geometries is analogous to that for the slab except that the leakage terms are more complicated. Consider Fig. 9.1.4.

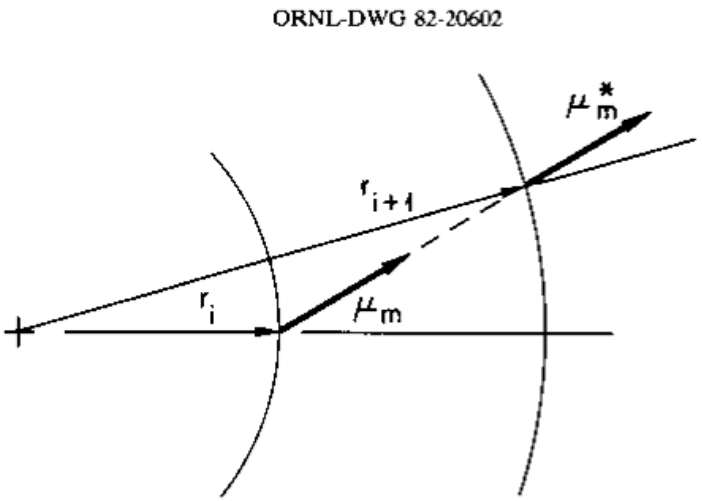


Fig. 9.1.4: Angular redistribution in spherical geometry.

Recall that the directions are taken with reference to the radius vector for a sphere. A particle traveling in direction μ_m at r_i will intersect the radius vector to the next point r_{i+1} at a different angle μ_m^*. The same effect also exists for the cylinder, though in this case the direction coordinates are more complicated. Because of the effect, a loss term is included for the "angular redistribution." It is defined in a manner analogous to Eq. Eq. (9.1.24) as

$$\alpha_{i+1/2},m+1/2\quad\psi_{g,i+1/2,m+1/2}-\alpha_{i+1/2,m-1/2}\psi_{g,i+1/2,m-1/2} \tag{9.1.28}$$

where the α coefficients are to be defined in such a manner as to preserve particle balance. In this case one speaks of $m+1$ and $m-1/2$ as the corresponding angles to μ_m on the I + lth and ith boundaries, respectively. (See Fig. 9.1.5) Here we are interested in an angle μ_m at the center of interval i which redistributes to $\mu_{m-\frac{1}{2}}$ at boundary i and to $\mu_{m+\frac{1}{2}}$ at boundary I + 1.

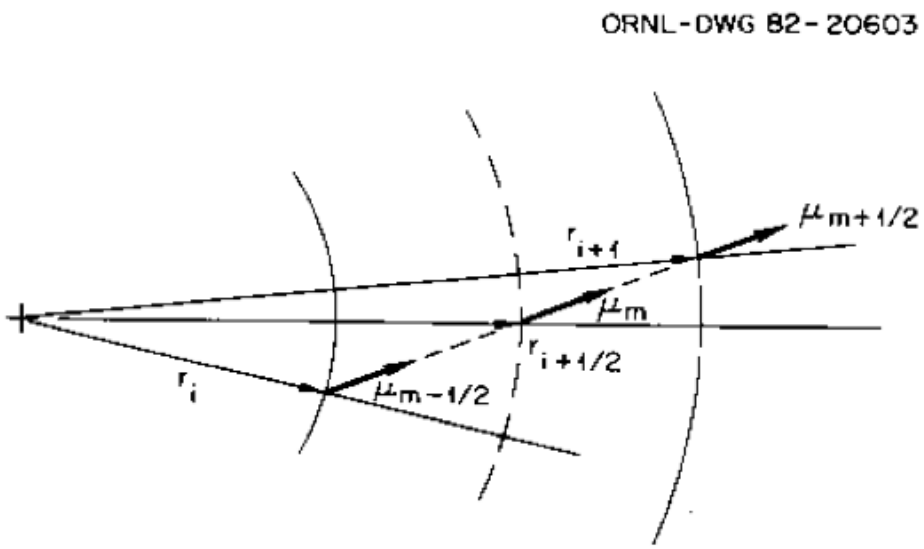


Fig. 9.1.5: Angular redistribution.

Obviously, it is necessary that the net effect of all redistributing be zero, in order to maintain particle balance. This condition is met if

$$\sum_{m=1}^{MM} \alpha_{m-1/2}\psi_{m-1/2} + \alpha_{m+1/2}\psi_{m+1/2} = \alpha_{1/2}\psi_{1/2} + \alpha_{MM+1/2}\psi_{MM+1/2} = 0, \tag{9.1.29}$$

where we have dropped the group and interval indexes.

In order to develop an expression for determining the α's consider an infinite medium with a constant isotropic flux. In this case, there is no leakage and the transport equation reduces to

$$\sum_t \phi = S. \tag{9.1.30}$$

This condition requires that

$$\mu_m w_m \left(A_{i+1}\psi_{g,i+1,m} - A_i\psi_{g,i,m}\right) + \alpha_{i+1/2}, m + 1/2 \quad \psi_{g,i+1/2}, m + 1/2 - \alpha_{i+1/2,m-1/2}\psi_{g,i+1/2,m-1/2} = 0, \tag{9.1.31}$$

which when we note that all the ψ terms in the infinite medium case are equal becomes

$$\mu_m w_m \left(A_{i+1} - A_i\right) = -\alpha_{m+1/2} + \alpha_{m-1/2}, \tag{9.1.32}$$

which is a recursion relationship for α.

From Eq. Eq. (9.1.29) we see that the conservation requirement can be met if

$$\alpha_{1/2} = \alpha_{MM+1/2} = 0 \tag{9.1.33}$$

for any values of flux, and is, therefore, used to evaluate the α's along with Eq. eq:*eq9-1-27* or eq:*eq9-1-28*. (Note that had we included the redistribution term in the slab equation, Eq. eq:*eq9-1-28* would have given zeroes for the terms, which is as one would expect for this geometry.)

The final discrete-ordinates expression for spheres and cylinders is then derived by summing expressions Eqs. eq:*eq9-1-20*, eq:*eq9-1-24*, eq:*eq9-1-21* and setting it equal to expression Eq. eq:*eq9-1-22*.

S_n quadratures for slabs

XSDRNPM will automatically calculate quadrature sets for each of the 1-D geometries, or a user can, if he wants, input a quadrature.

In the case of the 1-D slab, the quadrature is a double Gauss-Legendre set based on recommendations from [XSDCL65].

The ordering of the directions for a slab is shown in Fig. 9.1.6.

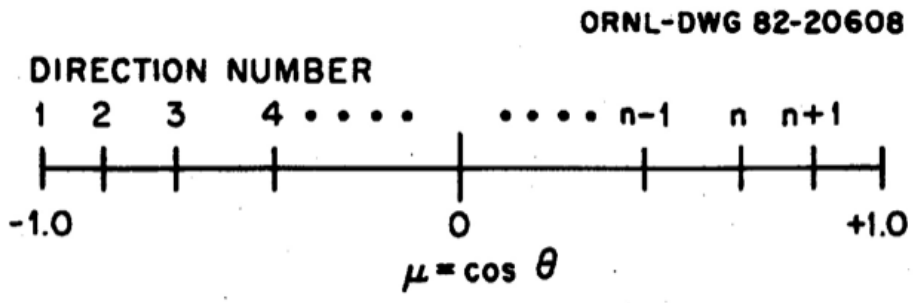


Fig. 9.1.6: Ordering of S_n directions for slabs and spheres.

Note that in referring to the quadratures for any of the geometries, we do not attempt to define an explicit area on a unit sphere, but rather speak of characteristic directions with associated weights. In the case of the slab, it is convenient to think of "directions" which are shaped like cones, because of the azimuthal symmetry around the x-axis.

In an nth order quadrature, there are n +1 angles with the first angle being taken at $\mu = -1.0$. This first angle is not required for the slab, but is needed for the curvilinear geometries because of the angular redistribution terms, as will be noted later. It is included in the slab case for reasons of uniformity of programming, etc.

Several requirements are made regarding the angles and weights in the quadrature set.

The arguments relating to angular redistribution can be expected to show that

$$\sum_{m=1}^{MM} \mu_m w_m = 0.0. \tag{9.1.34}$$

This situation is ensured if the weight of the $\mu = -1.0$ direction is zero and the other directions and weights are symmetric about $\mu = 0$. (The $\mu = 0$ direction is never included in the quadrature set because of its singularity.)

Further, it is required that

$$\sum_{m=1}^{MM} w_m = 1.0. \tag{9.1.35}$$

Due to the above normalization of the quadrature weights, the discrete ordinates angular flux is not "per steradian" but rather "per direction-weight". The calculated angular flux can be converted to steradians by dividing by 4π.

S_n *quadratures for spheres*

The quadratures generated for spheres are Gauss-Legendre coefficients as recommended by [XSDEme75].

The ordering and symmetry requirements for spheres are the same as for slabs.

In the case of the sphere, the initial ($\mu = -1.0$) direction is required, because the difference equations involve three unknown values for each direction, $\mu_m : \psi_m$ and the fluxes at the two "redistributed" angles $\psi_{m-\frac{1}{2}}$ and $\psi_{m+\frac{1}{2}}$. It is obvious that an angle along the radius will not involve the redistribution; hence, the expression for this direction involves only ψ ($\mu = -1.0$) as unknowns. Angle 2 proceeds by assuming $\psi_{2-\frac{1}{2}}$ is given by ψ_1 and also uses a weighted diamond difference model to relate ψ_m, $\psi_{m-\frac{1}{2}}$, $\psi_{m+\frac{1}{2}}$, as will be described below. Subsequent angles will then have values for $\psi_{m-\frac{1}{2}}$ calculated by the previous angle equations.

S_n *quadratures for cylinders*

The quadrature sets for cylinders are more complicated (see Fig. 9.1.2) because the directions must be specified with two angles, ξ and η where $\alpha = \sin\ \eta\ \cos\ \xi$ and $\beta = \cos\eta$.

In this case, practice is to use n/2 levels of directions for an n*th* order set. The levels correspond to fixed values of *eta*. The number of angles by level starts with three in level 1, five in level 2, seven in level 3, etc. (Note that since cylindrical geometry is curvilinear, each level will start with a $\eta = \pi$ direction that has zero weight for reasons analogous to those given for the spherical case. Fig. 9.1.7 shows the ordering of the directions for an S_6 quadrature set. Angles 1, 4, and 9 are the starting directions (zero weight) for the levels.

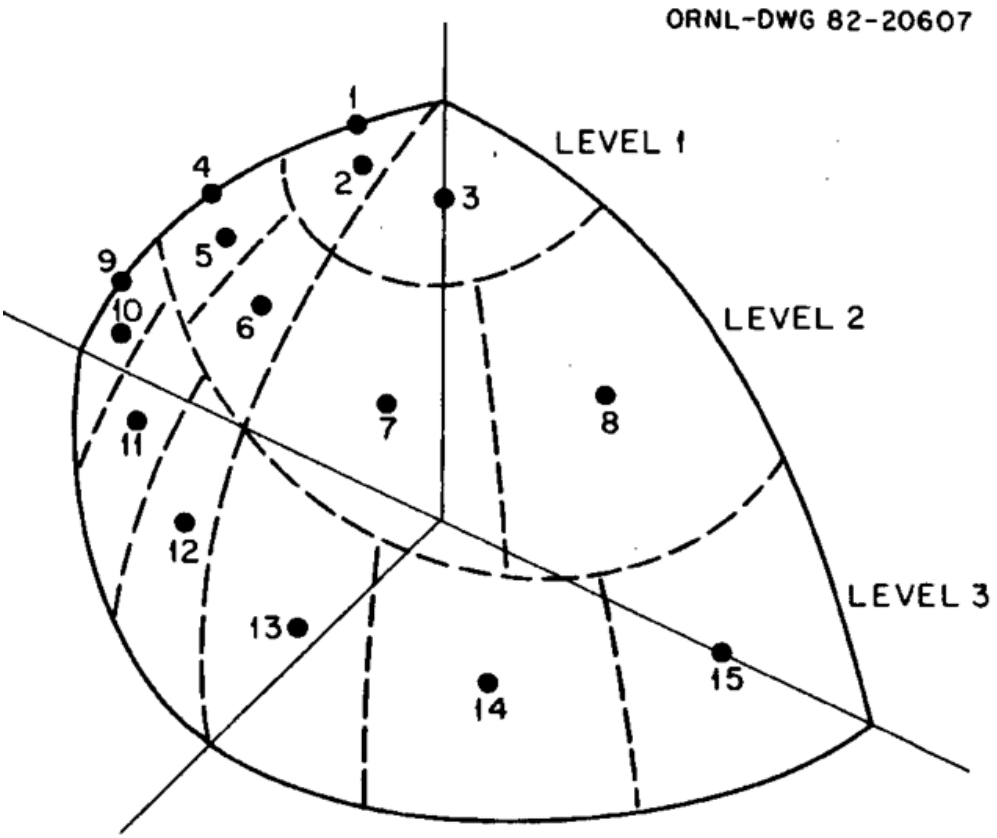


Fig. 9.1.7: Ordering of the directions for an S_6 cylindrical set.

In general, an n*th* order quadrature will contain n(n + 4)/4 angles. The cosines, μ, and the weights are stored in two arrays internally in the code; and, since the weights for the 1*st*, 4*th*, and 9*th* angles are zero, the cosines for the corresponding levels are placed in these locations in the arrays.

The cylindrical sets are based on Gauss-Tschebyscheff schemes as recommended by [XSDEme75] with Gaussian quadratures in β and Tschebyscheff quadratures in α.

9.1.2.5 Weighted-difference formulation for discrete-ordinates equations

In order to solve the discrete-ordinates equations, an assumption is required concerning the relationship of the various flux terms: $\psi_{i,m}, \psi_{i+1,m}, \psi_{i+1/2,m}, \psi_{i,m-1/2}, \psi_{i+1,m-1/2}$.

The solution of the equations involves three major loops: an outer loop over energy groups, a loop over angles, and a loop over the spatial mesh. The spatial loop is made either from the origin to the outside boundary or from the outside to the origin, depending on whether the angle is directed outward or inward, respectively.

Two models are widely used for expressing flux relationships: (1) the step model and (2) the diamond-difference linear model.

The "step model" is a histogram model whereby one sets the centered flux value to the appropriate boundary value, depending on which way the mesh sweep is going. If, for example, the sweep is to the right in space, then

$$\psi_{i+1/2,m} = \psi_{i,m} \tag{9.1.36}$$

or if to the left,

$$\psi_{i+1/2}, m = \psi_{i+1,m}. \tag{9.1.37}$$

Likewise, in angle:

$$\psi_{\mathrm{i}+1/2}, m = \psi_{i+1,m-1/2}. \tag{9.1.38}$$

The step model involves a very crude approximation, but has the marked advantage of helping to ensure positivity of flux values as long as scattering sources are positive.

In the "diamond-difference" model, the centered fluxes are assumed linear with the edge values:

$$\begin{aligned} \psi_{i+1/2} &= 0.5\,(\psi_i + \psi_{i+1}) \\ \psi_m &= 0.5\,(\psi_{m-1/2} + \psi_{m+1/2}) \end{aligned} \tag{9.1.39}$$

Unfortunately, though the linear model is clearly a better model than the step model, care must be taken by selecting a fine spatial mesh, or the linear extrapolation can lead to negative flux values. In some cases, the situation is so severe that it is impractical to take enough mesh points to eliminate the problems. Because of these difficulties XSDRNPM uses a different approach, as described below.

The weighted diamond difference model [XSDREJ77] was developed in an attempt to take advantage of the "correctness" of the linear model, while retaining the positive flux advantages of the step model.

A solution in some S_n codes is to use the linear model in all cases where positive fluxes are obtained and to revert to step model otherwise. Unfortunately, this method leads to artificial distortions in the fluxes.

Note that if one writes

$$\psi_{i+1/2} = a\psi_i + (1 - a)\psi_{i+1} \tag{9.1.40}$$

$$\psi_m = b\psi_{m-1/2} + (1-b)\psi_{m+1/2} \tag{9.1.41}$$

that the same expression can be used to express linear or step model (e.g., $a = b = \frac{1}{2}$ is equivalent to linear, while $a = b = 1.0$ can be used for the step model).

In the weighted model, the intention is to use the linear model when fluxes are positive but to select values for a and b in the range

$$1/2 \le a \text{ or } b \le 1.0 \tag{9.1.42}$$

that ensure positivity, if the source is positive.

At this point, it is convenient to rewrite the discrete-ordinates expression in a simplified notation, without the obvious subscripts on energy group, angle, etc.

$$w\mu\left(A_{i+1}\psi_{i+1} - A_i\psi_i\right) + \alpha_{m+1/2}\psi_{m+1/2} - \alpha_{m-1/2}\psi_{m-1/2} + w\sigma V\psi = wVS. \tag{9.1.43}$$

Combining Eqs. Eq. (9.1.43) and Eq. (9.1.40) or Eq. (9.1.41) yields the following expressions for ψ_{1+1} and $\psi_{m+1/2}$:

$$\psi_{i+1} = \frac{SV + C_2\psi_{m-1/2} + \left[\mu A_i - (1-a)D_1\right]\psi_i}{aD} \tag{9.1.44}$$

$$\psi_{m+1/2} = \frac{SV + C_1\psi_i + \left[\frac{\alpha_{m-1/2}}{w} - (1-b)D_2\right]\psi_{m-1/2}}{bD}, \tag{9.1.45}$$

where

$$C_1 = \mu\left[A_i + A_{i+1}\left(\frac{1}{\alpha} - 1\right)\right] \tag{9.1.46}$$

$$C_2 = \frac{\alpha_{m-1/2}}{w} + \frac{\alpha_{m+1/2}}{w}\left(\frac{1}{b} - 1\right) \tag{9.1.47}$$

$$D = \Sigma V_i + \frac{\mu A_{i+1}}{a} + \frac{\alpha_{m+1/2}}{wb} \tag{9.1.48}$$

$$D_1 = D - \frac{\mu A_{i+1}}{a} \tag{9.1.49}$$

$$D_2 = D - \frac{\alpha_{m+1/2}}{wb}. \tag{9.1.50}$$

In determining a and b, the "theta-weighted" model uses arbitrary multipliers θ_s on SV and θ_n on the C_2 $\psi_{m-\frac{1}{2}}$ or C_1 ψ_i terms in Eqs. Eq. (9.1.44) and Eq. (9.1.45). (In [XSDTREJ80], a thorough discussion is given on the history of using different choices of θ_s and θ_n and the advantages and disadvantages of each method.) In XSDRNPM, a value of 0.9 is used for θ_s and θ_n following the practice in the DOT-IV code.7

For ψ_{i+1} in Eq. Eq. (9.1.44) to be positive, the numerator should be positive, thereby requiring

$$\left[\mu A_i - (1-a)D_1\right]\psi_i < SV + C_2\psi_{m-1/2}, \tag{9.1.51}$$

which in the θ-weighted case becomes

$$\left[\mu A_i - (1-a)D_1\right]\psi_i < SV\theta_s + C_2\psi_{m-1/2}\theta_n. \tag{9.1.52}$$

A similar expression can be written for b using Eq. Eq. (9.1.45).

For reasons of accuracy, it is desirable to use $a = b = \frac{1}{2}$. Therefore, when a or b is determined to be less than $xfrac12$ it is automatically set to $xfrac12$.

9.1.2.6 Boundary conditions

XSDRNPM allows a boundary condition to be specified for each of the two "outside" boundaries of its 1-D geometries. The options are the following:

1. Vacuum boundary – all angular fluxes that are directed inward at the boundary are set to zero (e.g., at the left-hand boundary of slab, $\psi(\mu > 0) = 0$, etc.).

2. Reflected boundary – the incoming angular flux at a boundary is set equal to the outgoing angular flux in the reflected direction (e.g., at the left-hand boundary of a slab),

$$\psi_{in}(\mu) = \psi_{\text{out}}(-\mu) \tag{9.1.53}$$

3. Periodic boundary – the incoming angular flux at a boundary is set equal to the outgoing angular flux in the same angle at the opposite boundary.

4. White boundary – the angular fluxes of all incoming angles on a boundary are set equal to a constant value such that the net flow across the boundary is zero, that is,

$$\psi_{in} = \frac{\sum_m^{out} w_m \mu_m \psi_m}{\sum_m^{in} w_m |\mu_m|} \tag{9.1.54}$$

This boundary condition is generally used as an outer-boundary condition for cell calculation of cylinders and spheres that occur in lattice geometries.

5. Albedo boundary – this option is for the white boundary condition except that a user-supplied group-dependent albedo multiplies the incoming angular fluxes. This option is rarely used, as it is difficult to relate to most practical situations.

9.1.2.7 Fixed sources

Two types of inhomogeneous or fixed sources can be specified in XSDRNPM.

In the first case, an isotropic group-dependent volumetric source can be specified for any or all spatial intervals in a system.

In the second case, an angle- and group-dependent boundary source can be specified for any or all boundaries between spatial intervals in a system, excepting the left-most boundary. In this case, one specifies not a source but a flux condition on the boundary. If one uses the "track length" definition for flux, it is easy to show that the flux condition is related to a source condition by

$$\psi_m^s = \frac{S_m}{\mu_m}. \tag{9.1.55}$$

(This equation says that an isotropic source on a boundary would be input as a constant divided by the cosine of the direction.)

In conventional fixed-source calculations, the total fixed source in the system can be normalized to an input parameter, XNF. In the volumetric source case, the source values will be normalized such that

$$XNF = \sum_{g=1}^{IGM} \sum_{i=1}^{IM} Q_{g,i} V_i, \tag{9.1.56}$$

and in the boundary source case,

$$XNF = \sum_{g=1}^{IGM} \sum_{i=1}^{IM} A_{i+1} \sum_{m=1}^{MM} \mu_m \psi_m^s(g,i) w_m. \tag{9.1.57}$$

In the case where both volumetric and boundary sources are specified, the two sums are normalized to XNF.

The fixed source for a generalized adjoint calculation corresponds to a particular response ratio of interest. The generalized adjoint equation only has a solution for responses that are ratios of linear functionals of the flux, and in this case the source will contain both positive and negative components. These types of sources are described in more detail in ref:*9-1-2-15-1* and in the SAMS chapter, in *Generalized Perturbation Theory*.

9.1.2.8 Dimension search calculations

XSDRNPM has three options for searching for dimensions such that the system will produce a specified effective multiplication factor, k_{eff}. The options are selected by a parameter IEVT in the 1$ array and are as follows:

1. zone width search (IEVT = 4),
2. outer radius search (IEVT = 5),
3. buckling search (IEVT = 6).

By default, the search is made to produce a k_{eff} value of unity. For k_{eff}'s other than unity, IPVT (3$ array) is set to unity and the desired k_{eff} is input as PV (5* array).

Other input parameters which apply specifically to all search calculations are in the 5* array and are EV, the starting eigenvalue guess, EVM, the eigenvalue modifier, EQL, the eigenvalue convergence, and XNPM, the new parameter modifier. These parameters are discussed in more detail below.

Zone-width search (IEVT = 4)

With this option, one can vary the width of any or all zones in a case. Note that it is also possible to change zone widths at different rates.

This option requires the inputting of a zone width modifier array (41*) which is used to specify the relative movements of the zones according to the following expression:

$$\Delta Z_j^f = \Delta Z_j^i \left(1 + EV^* ZM_j\right) \tag{9.1.58}$$

where $\Delta Z_j^i, \Delta Z_j^f$ are the initial and final widths of zone j, respectively, ZM_{j} is the zone width modifier for the zone (as input in the 41* array), and EV is the final "eigenvalue" for the problem. Note that a zero value for ZM will specify a fixed zone width. Negative values for ZM are allowed.

Outer radius search (IEVT = 5)

With this option, all zones are scaled uniformly in order to make the system attain the specified k_{eff}. The final zone widths are found by multiplying the initial values by the final "eigenvalue:"

$$Z_j^f = EV\left(Z_j^i / Z_p^i\right). \tag{9.1.59}$$

Buckling search (IEVT = 6)

This option is used to search for "transverse" dimensions that will yield a specified k_{eff} for a system. This means that the search is for the height for a 1-D cylinder or the y- and/or z-dimensions in a 1-D slab.

For this option, the final dimensions are given by

DY = $DY_0 \times$ EV,

and

DZ = $DZ_0 \times$ EV,

where DY_0 , DZ_0 are the initial dimensions input in the 5* array.

Search calculation strategy

All the "dimension searches" use the same simple strategy. The calculations start by using the input eigenvalue (EV from the 5* array) to determine initial dimensions for the system. These dimensions allow the code to calculate a k_{eff} . The eigenvalue modifier (EVM in the 5* array) is then used to change the dimensions as follows:

IOPT = 4 (Zone width search)

$$\Delta Z_j^f = \Delta Z_j^i \left[1 + (EVM + EV)^* ZM_j\right] \tag{9.1.60}$$

IOPT = 5 (Outer radius search)

$$\Delta Z_j^f = (EVM + EV)\left(\Delta Z_j^i\right) \tag{9.1.61}$$

IOPT = 6 (Buckling search)

DY = DY_0 (EV + EVM)

DZ = DZ_0 (EV + EVM).

The new dimensions are then used in a new calculation which determines a second k_{eff} value.

XSDRNPM searches for a unity value of k_{eff} by default; however, when IPVT = 1 (3$ array), a nonunity value can be specified in PV (5* array) and the search will be made on this value.

Once the two k_{eff}'s are known, which are based on eigenvalues of EV and EV + EVM, respectively, a linear fit is used to project to the next value for EV. This yields an expression of the form

$$EV_{\text{next}} = EVM\frac{(PV - k_1)}{(k_2 - k_1)} + EV, \tag{9.1.62}$$

where k_1 and k_2 are the first and second value of *keff*, respectively. After this iteration, the procedure is to fit a quadratic to the three most recent k_{eff} values in order to obtain an estimate for the next EV.

The procedure continues until a relative convergence of EQL (5* array) or better is obtained on EV.

To prevent oscillations in the search, extrapolations are limited by XNPM, the new parameter modifier from the 5* array.

9.1.2.9 Alpa Search

It is possible to make some of the searches described in Sect. 9.1.2.8 in a more "direct" fashion than the strategy described in Sect. 9.1.2.8.4. XSDRNPM has two such options: (1) the alpha search and (2) a direct buckling search. These are described below.

Alpha search

The time-dependent form of the Boltzmann equation is identical with Eq. Eq. (9.1.1), except for the inclusion of a time-gradient term on the left-hand side:

$$\frac{1}{\mathrm{v}}\frac{\partial\psi(r,E,\Omega,t)}{\partial t}. \tag{9.1.63}$$

All other flux terms in the expression also would include the time (t) argument.

In some analyses it is reasonable to assume that the time variation of the flux is exponential, that is,

$$\psi(r,E,\Omega,t)=\psi(r,E,\Omega)e^{at}. \tag{9.1.64}$$

When this variation is introduced into the expanded form of Eq. Eq. (9.1.1), the exponential terms all cancel leaving a leading term:

$$\frac{\alpha}{\mathrm{v}}\psi(r,E,\Omega) \tag{9.1.65}$$

which is in the same form as the $\Sigma_t\psi$ term.

If one considers integrating over energy, angle, and space, the following expression can be derived:

$$P-A-L-\alpha V=0 \tag{9.1.66}$$

where

where

P ≡ production in the system,

A ≡ absorptions in the system,

L ≡ leakage from the system,

V ≡ $\int_0^\infty dE\int_0^{4\pi}d\vec{\Omega}\int_{system}d\vec{r}\,\frac{\psi(\vec{r},E,\vec{\Omega})}{\mathrm{v}}$

Since all terms other than α can be determined from a calculation, it is possible to determine α directly, thereby avoiding a scheme like that used for dimension searches. In the balance expression, the fission component of the production term is adjusted for the case of a non-unity k_{eff} value (IPVT = 1 in the 3$ array).

An α-search has several practical applications. If, for example, a subcritical assembly is pulsed by a source, the time-dependence of the flux is expected to die off exponentially. Another way to interpret the α-search is as that amount of 1/v absorber which could be added or taken away from a system in order to achieve criticality. This number could be of interest when certain control materials are used, such as $^{10}B_5$, which is a "1/v" material.

Direct-buckling search

A "direct"-buckling search can be made using a procedure analogous to that described in Sect. 9.1.2.9.1. Recall that the buckling is introduced in order to represent a transverse leakage through the use of a $DB^2\,\psi$ term. This suggests that the foregoing balance expression be written:

$$P - A - L - \alpha DB^2 X = 0, \tag{9.1.67}$$

where

$$X \equiv \int_0^{\infty} dE \int_0^{4\pi} d\Omega \int_{\text{system}} dr\psi(r, E, \Omega). \tag{9.1.68}$$

In this case, the diffusion coefficients, D_{g}, are determined from

$$D_g = \frac{1}{3\Sigma_{tr_g}}, \tag{9.1.69}$$

where

$$\Sigma_{tr_g} = \Sigma_{t_g} - \Sigma_{1_g}, \tag{9.1.70}$$

and Σ_1 is the within-group term from the P_1 scattering matrixes:

$$\Sigma_{1_{\text{g}}} = \sum_t (\,\text{g} \rightarrow \text{g}')\,. \tag{9.1.71}$$

The original B^2 value is determined as specified in Sect. 9.1.2.12, and the α is the square of the search parameter, that one multiplies by the original B^2 value in order to determine the final buckling and, hence, the dimensions of the system.

9.1.2.10 Iteration and convergence tests

Two parameters are used to specify the required levels of convergence on an XSDRNPM calculation. These are EPS and PTC, both given in the 5* array. The flux calculations proceed through a series of iterations until either convergence is achieved or the specified iteration limit is exceeded.

The basic iteration strategy in XSDRNPM is now described. The discrete-ordinates difference equation is solved for the first angle and the first energy group. This sweep generally is made from the last interval boundary to the center of the system, and it uses the flux guess supplied as part of the input along with the boundary conditions. The second angle is then calculated, etc., until all angles in the quadrature are treated. At the end of this sweep, new scalar fluxes for the midpoints of all intervals have been determined. The angular sweep continues until either the point scalar fluxes are converged to within PTC or until the code makes IIM inner iterations. An exception to this "inner iteration" pattern occurs on the first outer (defined below) iteration whenever a fission density guess is used, instead of the flux guess. In this case, the program uses 1-D diffusion theory to determine a scalar flux value for all intervals and the angular sweeps are not made until the second outer iteration. After the first group is completed, the calculation goes to the second group and repeats the above procedure. This continues until all groups have been treated.

The pass through all groups, angles, and intervals is called an outer iteration. Most of the convergence checks on the outer iteration have to do with reaction rates involving all energy groups and are made against the EPS parameter mentioned above. For a coupled neutron-gamma problem, outer iterations are only performed for

the neutron groups until convergence is achieved, then the final converged pass is made over all groups. In discussing these checks, it is convenient to define several terms:

Q ≡ total fixed source in the system

F ≡ total fission source in the system

D ≡ total outscatter rate in the system

$D \equiv \sum_{i}^{intervals} \sum_{g}^{groups} \sum_{g' \neq g}^{groups} \psi_{i,g} \sigma_{g \to g'} V_i$

$\psi_{i,g}$ ≡ scalar flux in intervals i and group g

$\sigma_{g \to g'}$ ≡ macroscopic scattering cross section from group g to group

V_i ≡ volume of interval i

k ≡ outer iteration number

IGM ≡ total number of energy groups

$\lambda_k \equiv \frac{Q+F_k}{Q+F_{k-1}}$

$G_k \equiv \frac{D_k}{Q+F_k}$

$\lambda'_k \equiv \frac{G_{k-1}}{G_k}$

$U_k \equiv$ total upscatter rate $= \sum_i \sum_g \sum_{g'<g} \psi_{i,g} \sigma_{g \to g'} V_i$

$\lambda''_k \equiv U_K / U_{k-1}, U_{k-1} \neq 0_j = 1, U_{k-1} = 0$

An inner iteration in XSDRNPM consists of sweeping one time through the entire spatial mesh for all the S_n angles for one energy group. When the fluxes for a particular group are being calculated, inner iterations (j) will continue until (a) the number of inner iterations for this outer iteration exceeds IIM (the inner iteration maximum) or (b) until

(1)

$$\max_i \left| \frac{\psi_{i,g}^{j} - \psi_{i,g}^{j-1}}{\psi_{i,g}^{j}} \right| \leq PTC \tag{9.1.72}$$

At the end of an outer iteration, the following checks are made:

(2)

$$|1.0 - \lambda_k| \leq EPS \tag{9.1.73}$$

(3)

$$R \left|1.0 - \lambda'_k\right| \leq EPS \tag{9.1.74}$$

(4)

$$R \left|1.0 - \lambda''_k\right| \leq EPS \tag{9.1.75}$$

R is a convergence relaxation factor and is set internally to 0.5 in XSDRNPM. If all convergence criteria are met, if ICM (the outer iteration maximum) is reached, or if ITMX (the maximum execution time) is exceeded, the problem will be terminated with full output; otherwise, another outer iteration will be started.

9.1.2.11 Group banding (scaling rebalance)

As described above, the normal mode of operation in XSDRNPM is to do inner iterations on a group until it converges, then go to the next group. For groups where there is no upscatter, the scattering source to a group depends only on higher energy groups for which the fluxes have already been calculated. A fixed source problem with no fission and no upscattering can, therefore, be converged in one outer iteration. Since fission sources and upscattering sources are calculated with fluxes from the previous outer iteration, multiple outer iterations must be done to converge problems involving these kinds of sources. For problems involving many fine thermal groups (groups with both upscatter and downscatter), a special convergence problem arises. Because the groups are fine, within-group scattering is small and the flux calculation is dominated by scattering sources from other groups. This situation leads to a very slow reduction in scattering source errors from one outer iteration to the next. XSDRNPM has a special "group banding" option for treating this problem. It involves collecting several groups together into a band and doing one inner for each group in the band while collecting particle balance information. This balance information is then used to solve for one set of flux rebalance factors to apply to each group in the band. Because the band is much wider than an individual group, the scattering that remains within the band is a much larger fraction of the total scattering source for the band. This condition leads to considerably faster convergence from one outer iteration to the next. The group banding option in XSDRNPM is triggered by the seventh entry in the 2$ array. The absolute value of this entry indicates the number of bands to be used. If the number is negative, these bands are only for the thermal groups. Normally there is no need to band together groups other than the thermal groups. An entry of -1 indicates that all the thermal groups will be treated as one band. This mode is one that is used successfully for many problems, but occasionally will cause a problem to not converge. For these problems using two or three bands for the thermal groups has been successful.

The code generates a default banding structure, but this structure can be overridden by inputting a 52$ array.

9.1.2.12 Buckling correction

XSDRNPM allows "buckling" corrections to be made for the transverse (non-calculated) dimensions in its 1-D slab and cylindrical geometries. Three input parameters-DY, DZ, and BF (5* array)-may be involved.

In the case of the 1-D slab, the height DY and the width DZ can be input. The buckling correction uses an expression based on asymptotic diffusion theory to account for leakage in the transverse direction and is treated analogous to an absorption cross section, that is,

Transverse Leakage = $DB^2\psi$

where B is the geometric buckling and is given by

$$B^2 = \left(\frac{\pi}{Y}\right)^2 + \left(\frac{\pi}{Z}\right)^2 \tag{9.1.76}$$

and Y and Z are the height and width of the slab, respectively, and include extrapolation distances.

Recall that the "extrapolation distance" is defined as the linear extrapolation distance such that if one extrapolated to a zero flux value at this distance from the boundary, the interior flux shape in the body would be correctly represented. The distance can be shown to occur at $0.71\lambda_{tr}$, where λ_{tr} is the transport mean free path given by $1/\Sigma_{tr}$. Note that for a slab, there are two extrapolation distances to include (one on either side) for the height and width, such that

$$Y = DY + 1.42\lambda_{tr}, \tag{9.1.77}$$

and

$$\mathrm{Z} = \mathrm{DZ} + 1.42\lambda_{tr}. \tag{9.1.78}$$

The 1.42 factor (= 2 *X* 0.71) is input in the BF parameter of the 5* array.

In calculating λ_{tr}, a transport cross section, Σ_{tr}, is determined from

$$\Sigma_{tr} = \Sigma_t - \Sigma_{s1} \tag{9.1.79}$$

which varies as a function of energy group and zone. The Σ_{s1} term is the within-group term from the P_1 scattering matrix.

In the case of the 1-D cylinder, the procedure is the same as for the slab except that the buckling is determined from

$$B^2 = \left(\frac{\pi}{Y}\right)^2, \tag{9.1.80}$$

since only one transverse dimension is needed.

The diffusion coefficient in the leakage term is determined from

$$D = \frac{1}{3\Sigma_{tr}} \tag{9.1.81}$$

Note that when comparing with codes or treatments using a fixed value of buckling for every group, a user can force this situation in XSDRNPM by inputting a zero value for BF and DZ and setting DY to determine the required buckling value.

9.1.2.13 Void streaming correction

In real slab and cylindrical geometries, void regions offer streaming paths that are nonexistent in the 1-D cases with quadratures that do not include a vertical angle. A correction for this effect has been suggested by Olsen [XSDOls65], who uses an adjustment to the absorption cross section to account for the transverse leakage.

If one considers a slab of height H, the void streaming correction is introduced through an adjustment to the total cross section and is given by

$$\frac{\sqrt{1-\mu_m^2}}{H/2}, \tag{9.1.82}$$

where μ_m is the cosine of the direction.

In the case of a cylinder of height H, the adjustment is

$$\frac{\mu_m}{H/2}. \tag{9.1.83}$$

These streaming corrections are very approximate and do not properly account for the fact that the streaming is enhanced near the ends of a void channel; however, they are probably better than the alternative, which is to make no correction at all.

9.1.2.14 Cross-section weighting

XSDRNPM weights cross sections according to the following four options:

1. "Cell" weighting,
2. "Zone" weighting,

3. "Region" or "vein" weighting, and
4. "Inner cell" weighting.

In all cases the "averaged" cross sections are defined in a manner that conserves reaction rates, that is,

$$\bar{\sigma}_G \int_{\text{space}} dr N_D(r) \int_G dE\psi(E,r) = \int_{\text{space}} dr N(r) \int_G dE\sigma(E,r)\psi(E,r), \tag{9.1.84}$$

where

$\bar{\sigma}_G \equiv$ average cross section in group G,

$N_D(r) \equiv$ number density used in the definition for the weighting option selected,

$\psi(E,r) \equiv$ weighting spectrum,

$N(r) \equiv$ real number density as a function of spatial position,

$\sigma(E,r) \equiv$ cross section in unreduced form.

If we convert to multigroup notation and use W for the weighting spectrum (instead of ψ), Eq. Eq. (9.1.84) becomes

$$\bar{\sigma}_G \sum_{\substack{\text{applicable}\\ \text{spatial}\\ \text{regions}\\ j}} N_D^j \sum_{g\varepsilon G} W_g^j = \sum_{\substack{\text{applicable}\\ \text{spatial}\\ \text{regions}\\ j}} N^j \sum_{g\varepsilon G} \sigma_g^j W_g^j , \tag{9.1.85}$$

$$W_g^j \equiv \int_j dr \int_g dE\psi(E,r) \tag{9.1.86}$$

"Cell Weighting"

Cell weighting is consistent with homogenizing the cross sections in a heterogeneous cell. This is the recommended option to prepare cross sections for a real reactor calculation that will be made with a 2- or 3-D model of the reactor. Most of these codes have no provisions for explicitly representing individual fuel pins which are interspersed in a moderator region.

Cell-weighted cross sections are defined in a manner that attempts to preserve the reaction rates which occur in a representative cell from the reactor. In Eq. Eq. (9.1.85) the weighting involves the following substitution:

$$N_D^j \equiv \bar{N} = \frac{\sum_j^{cell} V^j N^j}{\sum_j^{cell} V^j} \tag{9.1.87}$$

where

$V_{\text{ j}} \equiv$ volume of zone j.

"Zone" weighting

Zone weighting is the simplest of the three XSDRNPM weighting options. Each zone produces a unique set of cross sections which preserves reaction rates for the zone. In Eq. Eq. (9.1.85), the spatial sum is over the zone considered, and $N_{\text{ j}}\, N_D^j$ and are unity.

Zone weighting is used very frequently, especially for problems whose collapsed cross sections are to be used in a problem whose geometrical and material layout is similar to that in the weighting problem.

"Region" weighting

"Region-" or "vein-" weighted cross sections are weighted "where-the-nuclide-is." In most problems, there are nuclides of secondary importance which do not need a separate "zone-weighted" set for every region in which the nuclide occurs. Examples are the components of stainless steel. Stainless steel is encountered in a variety of locations and flux environments, but generally one set of cross sections for iron, chromium, manganese, nickel, etc., will suffice for most reactor calculations.

In Eq. Eq. (9.1.85), the spatial sum is over all zones which contain the nuclide of interest with

$$N_D^j = N^j \tag{9.1.88}$$

"Inner-cell" weighting

For inner-cell weighting, cell weighting is performed over specified innermost regions in the problem. Nuclides outside these regions are not weighted.

This option is generally employed as follows: A "cell" is described in exactly the same manner as for cell weighting (Sect. 9.1.2.14.1) except that in this case it is surrounded by a homogeneous representation for the remainder of the core and by blankets, reflectors, etc. The flux calculation is made over this complete system, which should have a more realistic treatment of the leakage across the outer boundary of the interior cell. The cell weighting is subsequently made only over the interior cell.

Multigroup weighting equations

1. Cell weighting

$$\bar{\sigma}_G \equiv \frac{\sum_j^{IZM} N^j \sum_{g\varepsilon G} \sigma_g^j W_g^j}{\bar{N} \sum_j^{IZM} \sum_{g\varepsilon G} W_g^j} \tag{9.1.89}$$

where

$$\bar{N} \equiv \frac{\sum_j^{IZM} V^j N^j}{\sum_j^{IZM} V^j} \tag{9.1.90}$$

$$W_g^j \equiv \psi_g^j = \int_j \psi_g(r) dr. \tag{9.1.91}$$

2. Zone weighting

$$\overline{\sigma_G^j} \equiv \frac{\sum_{g\varepsilon G} \sigma_g^j W_g^j}{\sum_{g\varepsilon G} W_G^j}. \tag{9.1.92}$$

3. Region weighting

$$\bar{\sigma}_G \equiv \frac{\sum_j N^j \sum_{g\varepsilon G} \sigma_g^j W_g^j}{\sum_j N^j \sum_{g\varepsilon G} W_g^j}. \tag{9.1.93}$$

Transfer matrices

Collapsing transfer matrices is not quite so simple as collapsing cross sections with a single value per group. A group-to-group term in the broad group sense conserves the scattering rate from one group to the other, that is,

$$\bar{N}^* \bar{\sigma} (G \to G') \psi_G \equiv \int_{\text{space}} dr N(r) \int_g dE \psi(E, r) \int_{g'} dE' \sigma (E \to E') \tag{9.1.94}$$

where the asterisk (*) denotes that the number density on the left side of the equation is consistent with the weighting desired. Therefore, the multigroup forms of the weighting equations for components of the transfer matrices are as follows:

1. Cell weighting

$$\bar{\sigma}_{G \to G'} \equiv \frac{\sum_j^{IZM} N^j \sum_{g \varepsilon G} W_g^j \sum_{g' \varepsilon G'} \sigma^j (g \to g')}{\bar{N} \sum_j^{IZM} \sum_{g \varepsilon G} W_g^j} \tag{9.1.95}$$

2. Zone weighting

$$\bar{\sigma}_{G \to G'} \equiv \frac{\sum_{g \varepsilon G} W_g^j \sum_{g' \varepsilon G'} \sigma^j (g \to g')}{\sum_{g \varepsilon G} W_g^j}. \tag{9.1.96}$$

3. Region weighting

$$\bar{\sigma}_{G \to G'} \equiv \frac{\sum_j N^j \sum_{g \varepsilon G} W_g^j \sum_{g' \varepsilon G'} \sigma^j (g \to g')}{\sum_j N^j \sum_{g \varepsilon G} W_g^j}. \tag{9.1.97}$$

Theoretically, the higher-than-zero order $\sigma_l (g \to g')$ should be weighted over ψ_l. Since these functions are generally positive-negative, ψ_l weighting does not always work in practice, and XSDRNPM weights the $\sigma_l (g \to g') , \quad > 0$, by the scalar flux, which is positive. This procedure gives usable values for most cases.

Weighting of $\bar{\nu}$

In weighting parameters such as $\bar{\nu}$, the average number of neutrons produced per fission, one is interested in preserving the fission source; therefore, the weighting is over $\sigma_f \psi$ instead of just ψ. The weighting procedure in XSDRNPM is to calculate $\left(\overline{\nu\sigma_f}\right)_G$ and $(\sigma_f)G$ using the appropriate choice from Eqs. Eq. (9.1.95), Eq. (9.1.96), or Eq. (9.1.97). Then

$$\bar{\nu}_G = \frac{(\overline{\nu\sigma_F})_G}{\left(\sigma_f\right)_G}. \tag{9.1.98}$$

Transport cross sections

Transport cross sections are not as directly related to the physical properties of a material as much as other group-averaged values. Instead of a reaction rate, these numbers must attempt to preserve a "flux gradient," which not only depends on the cross sections of the material, but is also very strongly influenced by the geometry and the other nuclides in the vicinity of a material.

Two options are provided in XSDRNPM to generate transport cross sections-options based on the "consistent" and "inconsistent" methods for solving the *Pl* transport equations. These approximations are referred to as the "outscatter" and "inscatter" approximations because of the nature of the equations used.

Outscatter approximation (inconsistent method)

In the outscatter approximation, the assumption is made that

$$\sigma_{tr}^{g} = \sigma_{t}^{g} - \bar{\mu}^{g}\sigma_{s}^{g}. \tag{9.1.99}$$

When one notes that

$$\bar{\mu}^{g} \equiv \frac{\sigma_{1}^{g}}{3\sigma_{0}^{g}} \tag{9.1.100}$$

and that

$$\sigma_{1}^{g} = \sum_{g'} \sigma_{1}\left(g \rightarrow g'\right), \tag{9.1.101}$$

where the $\sigma_{l}\left(g \rightarrow g'\right)$ terms are the P_1 coefficients of the scattering matrix, the origin of the term "outscatter" to designate the approximation is evident.

Inscatter approximation (consistent method)

In the "consistent" solution of the P_1 point transport equations, it can be shown that

$$\sigma_{tr}(E) = \sigma_{t}(E) - \frac{1}{3J(E)} \int_{0}^{\infty} dE' \sigma_{1}\left(E' \rightarrow E\right) J\left(E'\right), \tag{9.1.102}$$

where $J\left(E'\right)$ is the current.

If one multiplies the equation by J(E), integrates over group g, and converts to group-averaged form by dividing by $\int_{g} J(E)dE$ the following expression is derived:

$$\sigma_{tr}^{g} = \sigma_{t}^{g} - \frac{1}{3J_{g}} \sum_{g'} \sigma_{1}\left(g' \rightarrow g\right) J_{g'}. \tag{9.1.103}$$

This is the "inscatter" approximation. It is consistent because the transport values are explicitly derived from the P_0 and P_1 equations. As a general rule, the transport values from this treatment are "better" than those from the "inconsistent" treatment. However, in some cases (notably hydrogen at lower energies), negative numbers may be calculated which are unusable and the more approximate approach must be used.

Weighting function for transport cross section

Unfortunately, the matter of choosing a current to use in the "transport" weighting is not simple. In real problems, currents are positive-negative as a function of energy and space. When cross sections are averaged over positive-negative functions, the "law-of-the-mean" no longer holds and the average value can be anything. This unbounded nature leads to real problems in diffusion calculations.

Approximations that inherently guarantee positive currents are generally used in other codes that circumvent the positive-negative problem. For example, in *Bn* theory the current is given by

$$j \sim B\psi \tag{9.1.104}$$

where B and ψ are both positive.

In XSDRNPM, more direct routes that ensure positivity are taken (e.g., one might set $\mathrm{W_g} \equiv \left|\mathrm{W_g}\right|$). This is crudely supported by the following argument:

Consider a 1-D cylindrical calculation. In two dimensions, the current is a vector combination, that is,

$$J = J_r + J_z. \tag{9.1.105}$$

In XSDRNPM, the z direction is treated by using a buckling approximation, that is,

$$J_z = B\psi. \tag{9.1.106}$$

In the weighting calculation, we want to weight over the magnitude of the current. In XSDRNPM, the z current is imaginary, since we are not calculating a z-direction:

$$J = J_r + iB\psi. \tag{9.1.107}$$

The magnitude of a complex quantity is

$$J = \frac{(J_r + iB\psi)(J_r - iB\psi)}{\sqrt{(\text{Value}_r)^2 + B^2\psi^2}}, \tag{9.1.108}$$

which is always positive.

In a discrete-ordinates calculation, the current is easily obtained since it is the first flux moment.

XSDRNPM has the following options for calculating the current:

1.
$$J_g = \sqrt{\left(\psi_1^g\right)^2 + \left(DB\psi_g\right)^2} \tag{9.1.109}$$

2.
$$J_g = \left|\psi_1^g\right| \tag{9.1.110}$$

3.
$$J_g = DB^2\psi_g + \int_0^1 d\mu\mu\psi\left(g, r_{\text{outside}}, \mu\right) \tag{9.1.111}$$

4.
$$J_g = \frac{\psi_0^g}{\Sigma_t^g} \tag{9.1.112}$$

5.
$$J_g = DB\psi_g \tag{9.1.113}$$

The first option is the recommended option; option 2 treats only the current in the primary direction; option 3 will always be positive and is a weighting over the total leakage from the system. Option 4 is sometimes referred to as a "bootstrap" approximation; option 5 is equivalent to that used in codes that employ B_n theory.

Once the currents are determined, the transport values are determined as set forth in the equations discussed above. For example, consider cell weighting and the "inscatter" approximation,

$$\sigma_{tr}^G = \frac{\sum_j N^j \sum_{g\varepsilon G}\left\{J_g\sigma_t^g - \frac{1}{3}\sum_{g'}\sigma_1\left(g' \to g\right)J_g'\right\}}{\bar{N}\sum_j^{IZM}\sum_{g\varepsilon G} J_g} \tag{9.1.114}$$

For cell weighting and the "outscatter" approximation,

$$\sigma_{tr}^G = \frac{\sum_j N^j \sum_{g\varepsilon G} J_g\left\{\sigma_t^g - \frac{1}{3}\sum_{g'}\sigma_1\left(g' \to g\right)\right\}}{\bar{N}\sum_j^{IZM}\sum_{g\varepsilon G} J_g} \tag{9.1.115}$$

9.1.2.15 Adjoint calculations

XSDRNPM will, upon option, solve the adjoint forms of the 1-D transport equation.

Several special procedures apply for the adjoint calculation:

1. The iteration pattern discussed in Sect. 9.1.2.10 is reversed in energy. The scheme starts with the last (lowest energy) group and proceeds to the first group.
2. The angular quadrature is treated as if it has the reverse directions associated with the angle (e.g., many quadratures start with $\mu_1 = -1.0$). In the adjoint case, this direction is for $\mu_1 = +1.0$.
3. All edits of input fluxes and collapsed cross sections are given in their normal ordering, as opposed to many codes which require their reversal.

Adjoint calculations have many uses and advantages. As opposed to the forward calculation which yields particle density values, the adjoint fluxes are more abstract and can be thought of as particle importance.

Consider, for example, the problem of determining the response of a detector to particles as a function of their energy and direction. Assume the detector is a cylindrical fission chamber that utilizes a foil of ^{235}U. The most obvious way to attack this problem is to mock up the detector and make a series of runs that contain sources of identical strength in different angles and energy groups. If an S_8 (24 angles) quadrature were used with 50 energy groups, the 12×50 or 600 independent calculations could be used to completely determine the responses. (Here we have taken note that half of the angles will point away from a detector and, hence, produce no response.) The adjoint calculation produces all 600 responses in one run that is no more difficult and time consuming than the typical forward case. In the adjoint case, the detector response (i.e., the fission cross section of ^{235}U would be specified as a source in the foil region and the adjoint fluxes given as a function of energy and angle would be interpreted as the source of neutrons necessary to produce a response of the magnitude to which one required the response to be normalized.

A second important use of adjoint calculations is to establish good biasing factors for Monte Carlo codes. Two recent reports [XSDHof82, XSDHT82] discuss the time and accuracy advantages of this approach for shielding and criticality applications and give some real examples as to how to make the calculations.

Perturbation theory uses adjoint and forward fluxes in combination in a manner that determines changes in responses that would arise from changing parameters used in a calculation. One [XSDWRHB79] interesting application is to determine the sensitivity of a calculation to changes in one or more cross-section value changes.

Generalized adjoint calculations

Generalized adjoint solutions are needed for generalized perturbation theory (GPT) applications such as sensitivity and uncertainty analysis. The generalized adjoint solution differs from both a conventional external source case and a fundamental mode eigenvalue calculation: It has the transport operator for an adjoint eigenvalue equation, but contains a fixed source term as well. The eigenvalue transport operator is singular, which forces certain restrictions on the allowable sources. The generalized adjoint source term is associated with a particular response ratio of interest in a critical system, such as

$$R = \frac{\sum_{g=1}^{IGM} \sum_{i=1}^{IM} H_N(g,i)\psi_{i,g}V_i}{\sum_{g=1}^{IGM} \sum_{i=1}^{IM} H_D(g,i)\psi_{i,g}V_i} \tag{9.1.116}$$

where H_N and H_D are response functions defining the response of interest and $\psi_{i,g}$ is the scalar flux from a prior forward eigenvalue solution of the same problem. The generalized adjoint source for this response is

defined as

$$Q^*(g,i) \equiv \frac{1}{R}\frac{\partial R}{\partial \psi_{i,g}} = \frac{H_N(g,i)}{\sum_{g=1}^{IGM}\sum_{i=1}^{IM} H_N(g,i)\psi_{i,g}V_i} - \frac{H_D(g,i)}{\sum_{g=1}^{IGM}\sum_{i=1}^{IM} H_D(g,i)\psi_{i,g}V_i} \tag{9.1.117}$$

The above source expression is computed automatically whenever XSDRN is executed in the TSUNAMI-1D sequence, but must it be computed and input by the user if XSDRN is run standalone for a generalized adjoint case.

In order to obtain a unique solution and avoid numerical problems, the generalized adjoint solution is "normalized" to contain no fundamental harmonic of the adjoint eigenvalue calculation. This is done by sweeping out the adjoint fundamental mode "contamination" from the fission source after each outer iteration, as described in the SAMS chapter, in *Generalized Perturbation Theory*. This operation requires both forward and adjoint eigenvalue solutions from prior XSDRN calculations. External files containing the fundamental mode forward and adjoint fluxes are input to the generalized adjoint calculation.

Unlike conventional fixed source and eigenvalue calculations, the generalized adjoint flux has both negative and positive components. This causes some XSDRN acceleration features such as space-dependent rebalance and group-banding to not function properly; and thus these are turned off internally. Typically the outer iterations for generalized adjoint solution converge much slower than an eigenvalue calculation. More background on GPT and generalized adjoint properties can be found in [XSDWil86].

9.1.2.16 Coupled neutron-photon calculations

In XSDRNPM, it is possible to do a neutron or a photon calculation, depending only on whether the input libraries are for neutrons or gamma rays. It is also possible to do a "coupled neutron-photon" calculation which automatically determines the gamma-ray sources arising from neutron induced interactions in its photon calculation. This calculation, of course, requires an input cross-section library containing three classes of data:

1. neutron cross sections, including neutron-to-neutron transfer matrices,
2. photon production cross sections (i.e., neutron-to-gamma transfer matrices), and
3. gamma-ray cross sections, including gamma-ray-to-gamma-ray transfer matrices.

At present there are no provisions for treating neutrons produced from gamma interactions other than having the user introduce these sources by hand in a sort of iterative procedure, though this reaction is certainly not unknown (cf., deuterium, beryllium-9, and carbon-13). There are several cases where the (γ, n) interaction can be important. If, for example, one looks at neutrons in a water-moderated pool reactor or in a water spent fuel storage tank at large distances from the fuel, the dominant source is from the neutrons produced by the deuterium in the water.

Normally the neutron-photon calculation requires no more input than a single particle run, except in the case where extraneous neutron and/or gamma-ray sources need to be specified. Most output edits will be split into a neutron and a gamma-ray part and will be labeled as such

9.1.2.17 Diffusion theory option

XSDRNPM can make a 1-D diffusion theory calculation in user-specified energy groups (enter 1's for the appropriate groups of the 46$ array). In this case, the P_1 diffusion equations [XSDAld63] are solved:

$$A_{I+1}\psi_{1,I+1} - A_I\psi_{1,I} + \sigma_0\left(\psi_{0,I} + \psi_{0,I}\right) = S_0^* \tag{9.1.118}$$

$$\bar{A}_I\left(\psi_{0,I+1} - \psi_{0,I}\right) + \sigma_1\left(\psi_{1,I+1} + \psi_{1,I}\right) = S_1^*, \tag{9.1.119}$$

where

$$\psi_1 \equiv P_1 \text{ current} \tag{9.1.120}$$

$$\sigma_0 = \left[\Sigma_t - \Sigma_0(g \to g)\right]\frac{V_I}{2.0} \tag{9.1.121}$$

$$\sigma_1 = \left[3.0\Sigma_t - \Sigma_1(g \to g)\right]\frac{V_I}{2.0} \tag{9.1.122}$$

$$S_0^* = P_0 \text{ sources less the within-group term} \tag{9.1.123}$$

$$S_1^* = P_1 \text{ sources less the within-group term} \tag{9.1.124}$$

$$\bar{A}_I = \frac{A_I + A_{I+1}}{2} \tag{9.1.125}$$

$$V_I = \text{ volume of Ith interval.} \tag{9.1.126}$$

Solving Eq. Eq. (9.1.118) for $\psi_{0,I+1}$ and substituting into Eq. Eq. (9.1.119), one can write

$$\psi_{1,I+1} = \frac{\bar{A}_I S_0^* - 2\sigma_0\bar{A}_I\psi_{0,I} - \sigma_0 S_1^* + \psi_{1,I}\left(\sigma_0\sigma_1 + \bar{A}_I A_I\right)}{\bar{A}_I A_{I+1} - \sigma_0\sigma_1}. \tag{9.1.127}$$

Solving Eq. Eq. (9.1.119) for $\psi_{1,I}$ and substituting into Eq. Eq. (9.1.118) , one can write

$$\psi_{0,I+1} = \frac{A_{I+1}S_1^* - 2\sigma_1\bar{A}_I\psi_{1,I} - \sigma_1 S_0^* + \psi_{0,I}\left(\sigma_0\sigma_1 + \bar{A}_I A_{I+1}\right)}{\bar{A}_I A_{I+1} - \sigma_0\sigma_1}. \tag{9.1.128}$$

If one assumes

$$\psi_{1,I+1} = P_{I+1}\psi_{0,I+1} - q_{I+1} \tag{9.1.129}$$

$$\psi_{1,I} = P_I\psi_{0,I} - q_I \tag{9.1.130}$$

and plugs Eqs. Eq. (9.1.127) and Eq. (9.1.128) into , solving for $\psi_{1,I}$ yields:

$$\begin{array}{l}\psi_{1,I} = \frac{P_{I+1}\left(\sigma_0\sigma_1 + \bar{A}_I A_{I+1}\right) + 2\sigma_0\bar{A}_I}{\sigma_0\sigma_1 + \bar{A}_I A_I + P_{I+1}2\sigma_1\bar{A}_I}\psi_{0,I} \\ -\frac{S_0^*\left(\bar{A}_I + \sigma_1 P_{I+1}\right) - S_1^*(\sigma_0 + A_{I+1}P_{I+1}) + q_{I+1}\left(\bar{A}_I A_{I+1} - \sigma_0\sigma_1\right)}{\sigma_0\sigma_1 + \bar{A}_I A_I + P_{I+1}2\sigma_1\bar{A}_I}\end{array} \tag{9.1.131}$$

which by inspection and comparison with Eq. Eq. (9.1.130) gives expressions for P_{I} and q_{I}.

Equations Eq. (9.1.129) and Eq. (9.1.130) can be substituted into Eq. Eq. (9.1.118) and solved for $\psi_{0,I+1}$:

$$\psi_{0,I+1} = \frac{\psi_{0,I}\left(A_I P_I - \sigma_0\right) + A_{I+1}q_{I+1} - A_I q_I + S_0^*}{A_{I+1}P_{I+1} + \sigma_0}, \tag{9.1.132}$$

which is the expression used in XSDRNPM. The procedure solves for arrays of P_{I} and q_{I} which are plugged back into the above expression to yield the fluxes.

9.1.2.18 Infinite-medium theory option

It is possible to force the flux calculation in XSDRNPM to use an infinite medium option for any or all energy groups by entering 2's in the appropriate positions in the 46$ array. When a multiregion calculation is requested, the program will first determine spatially averaged cross sections to use in the infinite-medium expression and then place the infinite-medium flux in all spatial regions for use in any subsequent calculations, such as cross-section weighting. All higher flux moments are set to zero.

The balance expression is

$$\left[\Sigma_t^g - \Sigma(g \to g)\right]\psi_g = \frac{1}{k}F_g + S_g, \tag{9.1.133}$$

where F_g is the fission source in group g, S_g is the sum of any fixed source and inscattering source, and $\sum_t^g$ and $\Sigma(g \to g)$ are homogenized total and group-to-group scattering cross sections.

9.1.2.19 B_N theory option

XSDRNPM can make a B_n calculation in user-specified energy groups (enter 3's for the appropriate groups of the 46$ array). As in the infinite-medium option discussed in Sect. 9.1.2.18, cross sections in a multiregion system are not homogenized.

The B_n equations [XSDHTS76] can be written

$$\begin{aligned}\frac{l+1}{2l+1}iB\psi_{l+1} + \frac{l}{2l+1}iB\psi_{l-1} + \Sigma_t\psi_l = S(u)\delta_l^0 \\ + \int du'\Sigma_s^l(u' \to u)\,\psi_l(u') \quad l = 0, 1, \quad , N-1\end{aligned} \tag{9.1.134}$$

$$\frac{N}{2N+1}iB\psi_{N-1} + \gamma\Sigma_t\psi_N = \int du'\Sigma_g^N(u' \to u)\,\psi_N(u') \tag{9.1.135}$$

$$\psi_{-1} = 0 \tag{9.1.136}$$

$$\gamma = 1 + \frac{N+1}{2N+1}\frac{iB}{\Sigma_T}\frac{Q_{N+1}(-\Sigma_t/iB)}{Q_N(-\Sigma_t/iB)} \tag{9.1.137}$$

where δ_l^0 is the Kronecker delta function and Q_N is a Legendre polynomial of the second kind. In mutigroup form, the above expressions become:

$$\begin{aligned}\frac{l+1}{2l+1}iB\psi_{l+1}^g + \frac{l}{2l+1}iB\psi_{l-1}^g + \Sigma_t^g\psi_l^g = S_g\delta_l^0 \\ + \sum_{g'}\Sigma_l(g' \to g)\,\psi_l^g \quad l = 0, 1, \quad N-1\end{aligned} \tag{9.1.138}$$

$$\frac{N}{2N+1}iB\psi_{N-1}^g + \gamma\Sigma_t^g\psi_N^g = \sum_{g'}\Sigma_N(g' \to g)\,\psi_N^g \tag{9.1.139}$$

$$\psi_{-1}^g = 0. \tag{9.1.140}$$

In Eqs. Eq. (9.1.134) and Eq. (9.1.138), the S term includes fission, fixed, and scattering source components.

9.1.3 XSDRNPM INPUT DATA

The input data to XSDRNPM consist of a title card and up to five data blocks, depending on the particular problem. All data in these blocks are entered using the FIDO formats discussed in the chapter on FIDO.

In the description that follows, the quantity in square brackets is the number of items in an array. The quantity in braces is the condition which requires the array to be input. If no condition is specified, an array must be input. Default parameters that are used if an array is not input are shown in parentheses if nonzero.

****

Title Card - Format (20A4)

This is the title card for the problem. It will be used to label the problem output.

Data Block 1

This block contains information to set up various array dimensions and most calculational and editing options. Various convergence criteria and special constants can be input.

0$$ Logical Assignments [17]

1. LPUN – Logical number for punched card output (7).
2. LRSF – Random-access scratch for fluxes (10).
3. LAWL – Input AMPX working library (4).
4. LANC – ANISN binary or CCCC ISOTXS library (20).
5. LOWL – Output weighted library (3).
6. LANG – Angular flux scratch file (16).
7. LSFF – Scalar flux output file (17).
8. LSF2 – Sequential scratch space (18).
9. LSF3 – Sequential scratch space (19).
10. LRSM – Random-access scratch for macroscopic cross sections (8).
11. LRSX – Random-access scratch for macroscopic cross sections (9).
12. LACF – Activities output file (75).
13. LBTF – Balance table output file (76).
14. LIDF – Input dump file (73).
15. LSEN – Sensitivity output file (6).
16. LEXT – Not used (0).
17. LISF – Scalar flux input guess file (0).

1$ General Problem Description [15]

1. IGE - problem geometry (1)

0 - homogeneous (This causes a BN calculation to be made for all zones – Sect. 9.1.2.19)

1 - slab

2 - cylinder

3 - sphere

2. IZM – number of separate material regions or zones. (1)
3. IM – number of spatial intervals in the problem. (1)
4. IBL – the boundary condition at the left-hand boundary of the system. (1)

   0 - vacuum boundary

   1 - reflected boundary

   2 - periodic boundary

   3 - white/albedo boundary

Boundary conditions are discussed in Sect. 9.1.2.6.

5. IBR - the boundary condition at the right-hand boundary of the system. (1)

   0 - vacuum boundary

   1 - reflected boundary

   2 - periodic boundary

   3 - white/albedo boundary

6. MXX – the number of compositions used in the problem mock-up.
7. MS – the number of entries in the mixing table which specifies the makeup of the MXX compositions.
8. ISN – the order of angular quadrature to be used. If ISN > 0, XSDRNPM will calculate an angular quadrature for the appropriate geometry. If ISN < 0, the calculation is bypassed, and the user must supply a set in the 42# and 43# arrays.
9. ISCT – the order of scattering. Flux moments will be calculated through this order.
10. IEVT – the type of calculation. (1)

    0 - fixed source

    1 - k calculation

    2 - α calculation (flux is assumed to have an $e^{-\alpha t}$ time variation)

    3 - inoperable in present version

    4 - zone width search

    5 - outer radius search

    6 - buckling search

    7 - direct buckling search

11. IIM – the inner iteration maximum used in an Sn calculation. (10)
12. ICM – the outer iteration maximum. (10)

    After ICM outer iterations, the problem will be forced into the termination phase and the program will continue as if full convergence was attained. A message to this effect is printed.
13. ICLC – theory option. (0)

    0 - use S_n theory always

    N - use alternative theory (diffusion, infinite medium, or Bn) for N outer iterations, after which revert back to Sn theory.

    -N - always use alternative theory
14. ITH – forward/adjoint selector. (0)

    0 - solve the forward Boltzmann equation.

    1 - solve the adjoint Boltzmann equation.
15. IFLU – Generalized adjoint calculation flag. (0)

    0 - standard calculation

    1 - Generalized adjoint calculation. Requires input forward and adjoint fundamental mode fluxes on units 31 and 32, respectively

2$ Editing and Special Options [10]

1. IPRT – fine-group mixture cross-section edits. (-1)

   -2 - no edits

   -1 - edit 1-D cross sections

   0-N - edit through PN cross sections. 1-D edits are made, also.
2. ID1 – flux editing options. (0)

ID1	Print Angular fluxes	Print Scalar fluxes	Punch*a* Scalar fluxes
-1	No	No	No
0	No	Yes	No
1	Yes	Yes	No
2	No	Yes	Yes
3	Yes	Yes	Yes
a The fluxes will be punched in a format suitable for restarting an XSDRNPM calculation.			

3. IPBT – balance table edits. (0)

   -1 - none

   0 - make fine-group balance tables

   1 - make fine- and broad-group balance tables

4. ISX – broad-group flux edit as a function of interval. (0) (0/1 = no/yes)
5. ISEN – outer iteration acceleration. Input a zero. (0)
6. IBLN – control number of outer iteration groups. (0)
7. NBANDS – number of flux rebalance bands. (0) < 0, then this is the number of bands in the thermal range.
8. IFSN – If > 0 means no fission source if IEVT=0. (0)
9. ISQ3 – sequence number of file opened on unit LOWL. (1)
10. IDM4 – not used. (0)

The structure of the "activity" and the "balance table" files are described in Appendix A.

3$ Various Options [12]

1. IFG – cross-section weighting. (0)

   0 - none required

   1 - collapse cross sections
2. IQM – volumetric sources. (0)

   0 - none

   N - N volumetric source spectra will be input in the 31* array
3. IPM – boundary sources. (0)

   0 - none

   N - N boundary source spectra will be input in the 32* array
4. IFN – starting guess. (0)

   0 - flux guess (33# array)

   1 - fission density guess (34# array)
5. ITMX – maximum time allowed for the flux calculation in minutes. A value of zero specifies that the calculation should not be terminated because of time; otherwise the problem will be forced into the termination phase when ITMX is exceeded. (0) Bear in mind that this is an internal timing check and has no connection with operator or system terminations due to excessive times.
6. IDAT1 – external data storage. (0)

   0 - keep all arrays in core if possible

   1 - store mixture cross sections externally on a direct access device

   2 - store cross sections and fixed sources externally on direct access devices
7. IPN – diffusion coefficient option for transverse leakage corrections. (3)

0 - determine a transport cross section for each zone using P0 and P1 cross sections and, hence, a diffusion coefficient from 1/3 Σ_{tr}.

1 - spatially average the diffusion coefficients determined as for the above option and use it for all zones.

2 - spatially average the transport cross sections for all zones and determine a diffusion coefficient to be used in all zones by taking one over three times this value.

3 - flux weight the transport cross sections for all zones and determine a diffusion coefficient to be used in all zones by taking one over three times this value.

Normally, the first option (IPN = 0) is adequate; however, in cases involving regions of low concentration (near void) and, hence, very low transport cross sections, the very large diffusion coefficients lead to nonphysical behavior. In this case, the IPN = 3 option has been demonstrated to operate the best.

8. IDFM – density factors. (0)

   0 - none

   1 - read in density factors in the 38* array

9. IAZ – activity calculation trigger. (0)

   0 - none

   N - calculate the reaction rates by material zone for N different processes specified in the 49$ and 50$ arrays

10. IAI – spatially dependent activity rates. (0)

    0 - none

    1 - calculate reaction rates in each interval for IAZ processes

11. IFCT – thermal upscatter scaling. (0)

    0 - none

    1 - use upscatter scaling for accelerated problem convergence

12. IPVT – parametric eigenvalue search. (0)

    0 - none

    1 - a search calculation will be made for an eigenvalue equal to PV

    2 - an α loss term with α = PV will be added to the transport equation.

The term α will depend on the IEVT option selected

4$ Cross-Section Weighting Options [XSDEme75] {IFG = 1}

Note: Currently XSDRNPM does not support outputting weighted cross section libraries in a format other than AMPX. The following items are still allowed to be entered into XSDRNPM input files for legacy inputs, but anything not relating to AMPX library format will have no effect.

---

1. ICON – type of weighting. (See Sect. 9.1.2.14)

-N - inner cell (with N zones in the cell). Cell weighting is performed

over the N innermost regions in the problem. Nuclides outside these regions are not weighted.

-1 - cell

0 - zone

1 - region or vein

3. ITP – collapsed output format desired.

   0-19 – cross sections are written only in the AMPX weighted library formats on logical 3. A weighted library is always written when IFG= 1. 20-29 – (deprecated feature). 30-39 – (deprecated feature). 40-49 – (deprecated feature).

   The various values of ITP (modulo 10) are used to select the different transport cross-section weighting options mentioned earlier. The options are:

   ITP = 0 => $\sqrt{\left(\psi_1^g + \left(DG\psi_g\right)\right)^2}$

   ITP = 1 => absolute value of current

   ITP = 2 => $DB^2\psi_g$+ outside leakage

   ITP = 3 => $\psi / \sum_t^g$

   ITP = 4 => $DB\psi_g$

   ITP = other values are reserved for future development and should not be used.

4. IPP – weighted cross-section edit option -1.

   -2 – none

   -1 – edit 1-D data

   0-N – edit through *PN* cross-section arrays. 1-D edits are given.

5. IHTF – (deprecated feature)
6. NDSF – (deprecated feature)
7. NUSF – (deprecated feature)
8. IAP – (deprecated feature)
9. (deprecated feature)

5* Convergence Criteria and Assorted Constants [12]

1. EPS – overall problem convergence. (10:sup:-*4*)
2. PTC – scalar flux convergence. (10:sup:-*5*)
3. XNF – normalization factor. (1.0)

   When IEVT = 0, the fixed sources are normalized to XNF.

   For IEVT > 0, the fission source is normalized to XNF.

   When XNF = 0.0, no normalization is made.

   XNF should only be specified as 0 for a fixed source problem (IEVT = 0).

4. EV – starting eigenvalue guess for search calculations.

5. EVM – eigenvalue modifier used in a search calculation. The following is a tabulation of recommended values for EV and EVM.

IEVT	Calculation type	EV	EVM
0	Fixed source	0	0
1	k-calculation	0	0
2	Direct α-search	0	0
3	–	–	–
4	Zone width search	0	-0.1
5	Outer radius search	Starting outer radius	-0.1*EV
6	Buckling search	1.0	-0.1
7	Direct buckling search	0.0	0.0

6. BF – buckling factor (1.420892). This parameter is two times the multiplier on the "extrapolation" distance used to determine where a linearly extrapolated line from the asymptotic flux shape would go to zero (e.g., for slabs, the extrapolation distance is $0.71\lambda_{tr}$ and, hence, BF 1.42).

7. DY – first transverse dimension in centimeters used in a buckling correction to calculate leakage normal to the principal calculation direction (i.e., the height of a slab or a cylinder).

8. DZ – second transverse dimension in centimeters used for a buckling correction (i.e., the width of a slab).

9. VSC – void streaming correction. This is the height of a void streaming path in a cylinder or slab in centimeters. See Sect. 9.1.2.13.

10. PV – parametric eigenvalue or value for α used when IPVT > 0. When IPVT = 1 and IEVT > 1, this is the value of k-effective on which the search calculation is to be made (0.0)

11. EQL – eigenvalue convergence for a search. (10^{-3})

12. XNPM – new parameter modifier used in search calculations. (0.75)

T Terminate data block.

Data Block 2

This block contains information on the composition of the materials used in the calculation. Also included is an array to select special cross sections for ANISN and an array to identify cross sections written on the CCCC ISOTXS library.

Note: Currently XSDRNPM does not support outputting weighted cross section libraries in any format other than AMPX. XSDRNPM will still read the deprecated arrays, but they will have no effect.

10$ (deprecated array)

12$ (deprecated array)

11$ Composition Numbers in Mixing Table [MS]

If the input multigroup library was previously self-shielded by running the XSProc module, the composition numbers are the mix numbers given in the READ COMP block of XSProc. If the input library has not been previously self-shielded, enter all zeros.

Note: this is a new array added in SCALE 6.2 which MUST be present when using a microscopic library produced by the XSProc module. It may be omitted if using a macroscopic library from XSProc

---

13$ Local Mixture Numbers in Mixing Table [MS]

The values range from 1 to MXX, and are used only in XSDRN for referencing mixtures in the mixing table.

14$ Isotope Identifiers in Mixing Table [MS]

A set of data with this identification must be on unit LAWL, the XSDRNPM working library, though the code will not make checks to ensure this is the case.

15* Isotope Concentrations in Mixing Table [MS]

16$ (deprecated array)

18U (deprecated array)

T Terminate Data Block

Data Block 3 {IEVT = 0}

This block is used to specify fixed sources. ..

30$ Source Spectrum Number by Interval [IM]

31* Volumetric Source Spectra [IQM*IGM]

32* Surface Source Spectra [IPM*IGM*MM]

Each of the IQM or IPM spectra is specified in the 31* or 32* array and are stacked one after the other in that array. If both volumetric and surface sources are used in the same problem, the surface source number is multiplied by (IQM + 1) when entered in the 30$ array.

A volumetric spectrum will consist of IGM (number of energy groups) entries which are the relative integrated values of the source in each group.

A surface source is always assumed to be on the right-hand side of a spatial interval. It is input as was the volumetric source, except that each group contains entries for the MM angles in the S_n quadrature chosen for the problem. Note that a surface source is an **integrated** value and is actually a flux condition in the Sn|equations.

In the 30$ array, a zero entry specifies that no source is in an interval.

Data Block 4

This data block contains starting guesses for fluxes and fission densities. If fluxes are read from an external device (LISF>0), this data block is omitted. Both arrays in this block are double-precision arrays, which will require the use of the "#" array designator; otherwise the number of entries read into the arrays will be incorrect or may contain nonsensical values for the starting guess.

33# Flux Guess [IM*IGM] {IFN=0}

A guess for the scalar flux is specified in the order: ((FLUX(I,J),I=1,IM),J=1,IGM), where IM is the number of spatial intervals and IGM is the total number of energy groups. For fixed-source problems, without better information, use zeroes. For eigenvalue problems, a nonzero flux guess must be used. The fluxes punched by using the ID1 parameter in the 2$ array can be used here in restart calculations.

34# Fission Density Guess [IM] {IFN=1}

This is a guess at the number of fission neutrons produced in an interval. When IFN = 1, XSDRNPM uses diffusion theory for the first outer iteration, after which it reverts to the normal mode.

T Terminate this data block.

Data Block 5

This block contains the remaining data needed for an XSDRNPM calculation.

35* Interval Boundaries [IM+1] (cm)

This array describes the spatial quadrature into which the problem model is divided. The boundaries are nonnegative for curvilinear geometries and in increasing order. Usually they will start with a zero value, though this is not necessary. The origin for slab geometry may be negative.

36$ Zone Number for Each Spatial Interval [IM]

Spatial zones should be contiguous.

38* Density Factors by Interval [IM] {IDFM>0} (1.0)

These factors are used to effect a density variation in a mixture as a function of spatial interval. Zero for a density factor affords a convenient way for modeling a void region.

39$ Mixture Numbers by Zone [IZM]

The mixture that is in a zone is specified here.

40$ Order of Scattering by Zone [IZM] (ISCT)

This is the order, *l*, of the *Sn Pl *calculation which is desired in a zone. This number should be no larger than ISCT.

41* Radius Modifiers by Zone [IZM] {IEVT=4}

These parameters specify the relative movement of the width of a zone in a zone width search. A zero indicates that a zone's width is fixed. (See Sect. 9.1.2.8)

42# Weights of the Angles in the Discrete-Ordinates Quadrature MM = ISN + 1 for slabs and spheres, ISN*(ISN + 4)/4 for a cylinder.

Input this set if you wish to override those provided by XSDRNPM. See Sect. 9.1.2.4.3, Sect. 9.1.2.4.4, or Sect. 9.1.2.4.5.

43# Cosines of the Angles in the Discrete-Ordinates Quadrature [MM]

Input this set if you wish to override those provided by XSDRNPM. See Sect. 9.1.2.4.3, Sect. 9.1.2.4.4, or Sect. 9.1.2.4.5.

46$ Calculational Option by Group [IGM] {ICLC>0}

0 - perform discrete-ordinates calculation for this group.

1 - perform a diffusion calculation for this group for ICLC outer iterations; use discrete-ordinates theory after this.

2 - perform a homogeneous calculation for this group for ICLC outer iterations; then revert back to discrete-ordinates theory.

3 - perform a homogeneous calculation using *Bn* theory for this group.

47* Right-Boundary Albedos by Group [IGM] {IBR=3} (1.0)

A right-boundary albedo is specified for each fine group. The return current is distributed isotropically in angle.

48* Left-Boundary Albedos by Group [IGM] {IBL=3} (1.0)

As for the 47* array but for the left boundary. Note that if IBR or IBL is 3 and the corresponding 47* or 48* array is omitted, XSDRNPM fills the array with 1.0's effecting a boundary with zero net current and with isotropic neutron return.

49$ Material Number for Activities [IAZ] {IAZ>0}

50$ Process Number for Activities [IAZ] {IAZ>0}

The 49$ and 50$ arrays provide a means of obtaining the activity (reaction rate) for any process for which cross sections are available in the XSDRNPM calculation. A representative activity table entry is shown below:

ACTIVITY TABLE ENTRY	
49$	50$
M	N

This entry specifies that the activity N for material M be calculated for all parts of the system which contain that material.

If N is < 0, a density of 1.0 is used to calculate activities instead of densities in the mixing table. Allowable process identifiers are given in Appendix M4.B.

If M is < 0, the activities by interval will be multiplied by a 1-D area as follows:

1.0 for a slab (IGE=1)

$2 \times r$ for a cylinder (IGE=2)

$4 \times r^2$ for a sphere (IGE=3)

51$ Broad-Group Numbers [IGM] {IFG>0}

This array contains the broad-group numbers into which the fine groups are collapsed in a flux-weighting calculation. For example, if the first five fine groups are to be collapsed to the first broad group, the first five entries in the 51$ array are 1, etc. A zero value can be used to ignore (or truncate) a group.

52$ Lower Band Group Numbers [NBANDS]

Group numbers giving the last group in a flux rebalance band. Overrides the default set supplied by XSDRNPM.

9.1.3.1 Abbreviated XSDRNPM input description

Title Card - (18A4)	
DATA BLOCK 1	
-1$ - Storage Assignment	1$ - General Description [15]
1. Maximum Length of Storage Array	1. IGE - geometry
0$- Logical Assignments [17]	2. IZM - number of zones
1. Punched Cards (7)	3. IM - number of intervals
2. Direct Access - Fluxes (10)	4. IBL - left boundary condition
3. Working Library (4)	5. IBR - right boundary condition
4. ANISN or ISOTXS (20)	6. MXX - number of mixtures
5. Weighted Library (3)	7. MS - mixing table length
6. Angular Fluxes (16)	8. ISN - angular quadrature
7. Scalar Fluxes (17)	9. ISCT - order of scattering
8. Scratch (18)	10. IEVT - problem type
9. Scratch (19)	11. IIM - inner iteration maximum
10. Direct Access (8)	12. ICM - outer iteration maximum
11. Direct Access (9)	13. ICLC - optional theory
12. Activities (75)	14. ITH - forward or adjoint
13. Balances (76)	15. IFLU - GPT calculation flag
14. Input Dump (73)	
15. Sensitivity Data (0)	
16. Not Used (0)	
17. Flux Guess (0)	

DATA BLOCK 1 (continued)	
2$ - Editing and Control Options [10]	3$ - Other Options [12]
1. Fine-Group-Mixture Edit	1. IFG - Weighting Option
2. Fine-Group-Flux Edit	2. IQM - Volumetric Sources
3. Balance Table Edit	3. IPM - Boundary Sources
4. Broad-Group-Flux Edit	4. IFN - Starting Guess
5. Not Used	5. ITMX - Time Shut-off
6. Outers-Group-Limit-Option	6. IDAT1 - Storage Scheme
7. Number of Bands	7. IPN - Diff. Coeff. Option
8. Suppress Fixed-Source Fission	8. IDFM - Density Factors
9. Not Used	9. IAZ - Activities by Zone
10. Not Used	10. IAI - Activities by Interval
	11. IFCT - Thermal Scaling
	12. IPVT - Search on k ≠ 1
4$ - Weighting Options [9]	5* - Floating Point Values [12]
1. Type of weighting	1. EPS - overall convergence
2. Number of broad groups	2. PTC - Point flux convergence
3. Output format	3. XNF - normalization
4. Edit option	4. EV - starting guess for search
5. σ_T position or number	5. EVM - modifier for search
6. σ_{gg} position	6. BF - buckling factor
7. Table length	7. DY - height
8. ANISN edit option	8. DZ - width
9. Extra cross sections	9. VSC - void streaming height
	10. PV - k for search
	11. EQL - search convergence
	12. XNPM - search modifier
	T Terminate Data Block 1

Title Card - (18A4)
DATA BLOCK 2
10$ CCCC Transport Cross Section Selector [IHTF] 11$ Composition numbers 12$ Additional Processes to be put on ANISN Library [MSCM] 13$ Mixture Numbers [MS] 14$ Isotope Identifiers [MS] 15* Isotope Concentrations [MS] 16$ CCCC Identifiers from Working Library [IHTF] 18U or 18# CCCC Identifiers on ISOTXS [IHTF] T Terminate the second Data Block.
DATA BLOCK 3
{Required only when IQM or IPM is nonzero.} 30$ Spectrum Number by Interval (IQM>0) or Right-Hand Interval Boundary (IPM>0) [IM] 31* Volumetric Sources [IQM*IGM] 32* Boundary Sources [IPM*IGM*MM] T Terminate the third Data Block.
DATA BLOCK 4
{When fluxes are read from an external device-IFN>3-this block is omitted.} 33# Flux Guess [IM*IGM] {IFN=0} 34# Fission Density Guess [IM] {IFN=1} T Terminate the fourth Data Block.

Title Card - (18A4)
DATA BLOCK 5
35* Interval Boundaries [IM + 1] 36$ Zone Numbers by Interval [IM] 38* Density Factors [IM] 39$ Mixture Number by Zone [IZM] 40$ Order of Scattering by Zone [IZM] 41* Radius Modifier by Zone [IZM] {IEVT=4} 42# Discrete-Ordinates Cosines [MM] 43# Discrete-Ordinates Weights [MM] 46$ Alternative Theory Selection [IGM] {ICLC>0} 47* Right-Boundary Albedos [IGM] 48* Left-Boundary Albedos [IGM] 49$ Activity Material or Nuclide Numbers [IAZ] 50$ Activity Process Numbers [IAZ] (See Appendix B) 51$ Broad-Group Numbers [IGM] {IFG>0} 52$ Lower-Band-Group Numbers [NBANDS] T Terminate the fifth Data Block.

9.1.4 XSDRNPM INPUT/OUTPUT ASSIGNMENTS

The following logical units can be required in an XSDRNPM calculation.

Default Logical Unit	Purpose
3	Weighted Library (Produced by XSDRNPM)
4	Working Library (Input)
5	Card Input
6	Standard Output
7	Punch Fluxes or ANISN Libraries
8	Scratch Direct-Access Device for External Cross-Section Storage
9	Scratch Direct-Access Device for Mixing and Weighting Operations
10	Scratch Direct-Access Device for External Flux Moment Storage
16	Angular Fluxes
17	Scalar Fluxes
18	Scratch Device
19	Scratch Device
20	ANISN Binary Libraries or CCCC ISOTXS Interface
31	Fundamental mode forward angular flux input unit for generalized adjoint calculation (iflu>0)
32	Fundamental mode adjoint angular flux input unit for generalized adjoint calculation (iflu>0)
73	Dump of Input and Derived Data
75	Activities
76	Balance Tables

9.1.5 XSDRN SAMPLE PROBLEM

In this section, the input and output for a sample case involving a bare, homogeneous 16-cm sphere of a 93% enriched *UO2-F2* solution is presented. The input AMPX working format cross-section library will be read from logical unit 4.

The input working library is a temporary 238-group library created by CSAS-mg containing the following nuclides:

Nuclide	Identifier[a]
$^{235}U_{92}$	92235
$^{238}U_{92}$	92238
^{1}H	1001[b]
$^{16}O_{8}$	8016
$^{19}F_{9}$	9019
a These are the identifiers of the sets of data on the library created for the problem. *b* Water-bound kernel.	

An S_{16} quadrature is selected with 32 spatial intervals. Activities are requested for $^{235}U_{92}$ absorption and fission and $^{238}U_{92}$ absorption.

```
=xsdrn
93% uo2f2 solution sphere
1$$ 3 1 32 1 0 1 5 16 1 1 30 20 0 0 0
2$$ a7 -1 e
3$$ 1 a9 3 1 e
4$$ 0 4 0 -1 5 e
5** 1.-7 1.-8 e  1t
13$$        1        1        1        1        1
14$$     1001     8016     9019    92235    92238
15**  6.548-2  3.342-2  6.809-4  3.169-4  2.355-5
16$$    1001    8016    9019   92235   92238
18## 6hh-1    6ho-16   6hf-19   6hu-235  6hu-238    2t
33## f1  4t
35** 31i0.0 16.0
36$$ f1
49$$ 92235 92235 92238
50$$    18    27    27
51$$ 74r1 74r2 45r3 45r4   5t
end
```

9.1.6 OUTPUT CROSS SECTIONS

One of the most common uses of XSDRNPM is to collapse cross sections and write them onto a file for input into another computer code. At present, two options are allowed:

1. Output library in AMPX Working Library format. This library is always written when cross sections are collapsed.
2. Output library in ANISN binary or BCD *format* [XSDJD63]. The binary library is written on logical 20 by default; the BCD library is produced on logical 7. The identifiers on this library range from 1 to the total number of blocks required to accommodate the data.

9.1.7 ERROR MESSAGES

During the course of a problem, XSDRNPM makes many checks to determine if input data are in the required form. If inconsistencies are spotted, a message is printed, and the problem may be terminated. Some of these messages are listed below along with a brief description of their possible cause.

DATA N N arrays have been input with incorrect length. See the messages produced as arrays are read to determine specific arrays.

SN-1 N The Nth entry in the S_n quadrature directions is zero. (43# array)

SN-2 0 The S_n weights do not sum to 1.0. (42# array)

SN-3 0 The sum of the products of S_n weights and directions is not 0.0, that is, the directions are not symmetric. (42# and 43# arrays)

FIXS 0 Fixed source calculation requested (IEVT=0) and total fixed sources are zero.

Q-HI N A volumetric source spectrum numbered N has been requested where N is greater than IQM.

B-HI N A boundary source spectrum numbered N has been requested where N is greater than IPM.

FISS 0 IEVT≥ 1 and the total fission source is zero.

8101 N The N^{th} radius is negative.

8102 N The N^{th} radius is less than the $(N-1)^{th}$ radius.

8103 N Zone N dimensions have become negative in a zone width search.

MIX N A request has been made to use the N^{th} component from the mixing table, but this nuclide has not been requested from a library.

Several messages may be encountered during an XSDRNPM run which indicate problems with either the code or the setup:

ROOT L The polynomials from which the default angles are derived are incorrect.

BAND N The number of bands specified is greater than the number of groups.

WAT1 N The number of sets of weighted cross sections is incorrect.

WAT2 N The number of sets of weighted cross sections is incorrect.

```
*********************************************************INFACE*************************************
*** WARNING    YOU REQUESTED nn SETS OF CROSS SECTIONS, BUT ONLY mm SETS WERE FOUND *************
*********************************************************MIXEM**************************************
MAGIC WORD ERROR DETECTED IN MIXEM, MW= xx
*********************************************************SPOUT**************************************
NO PROGRAMMING PROVIDED FOR ITP= nn
*********************************************************FIDAS**************************************
****** ERROR nn ENTRIES REQUIRED IN xx? ARRAY
DATA EDIT CONTINUES
*********************************************************FIDAS**************************************
****** FILL OPTION IGNORED IN xx? ARRAY
*********************************************************FIDAS**************************************
****** WARNING ADDRESS aa IS BEYOND LIMITS OF xx? ARRAY
*********************************************************STORXS*************************************
MAGIC WORD ERROR IN STORXS - GROUP gg     MIXTURE mm    L  ln
MAGIC WORD mw     IGI  ig    MXI mx    MNI mn      LLL  11
*********************************************************STORXS*************************************
ERROR #1 IN STORXS.$
*********************************************************STORXS*************************************
ERROR #2 IN STORXS.$
```

For the cryptic messages above (e.g., the last two), contact the code developers as to their possible cause.

9.1.8 APPENDICES

9.1.8.1 Special XSDRNPM Files

Three special files that can be optionally produced by XSDRNPM are described in this appendix. (See Sect. 9.1.5 and the discussion of the logical units in the 0$ array.) The files will be created with file names of the form **ftNNfXXX.EXT** where **NN** is the 2 digit logical unit number (from the 0$$ array), **XXX** is a 3 digit number which is incremented by one starting with one to make the name unique, and **EXT** is an extension identifying which type of file it is (**acf** for activity file, **btf** for balance table file, and **idf** for input and derived data file).

Activity file

The data on the activity file depends on what input options are specified. The data is in ASCII sets, which consist of a label record followed by the record(s) of the activity. There will be at most **IAZ** sets ordered as the 49$ and 50$ arrays. The first sets of data will be the activities by interval (if the input parameter **IAI** was specified). A set will be formatted as below.

activity by interval for nuclide *nnnnnnnn* **reaction type** *rrrrrrrr*

Activity(first interval)

.

.

.

.

Activity(last interval)

The preceding set will be repeated **IAZ** times. Then sets giving the activities by zone will be given. They will be formatted as below.

activity by zone for nuclide *nnnnnnnn* **reaction** *rrrrrrrr*
Activity(first zone)

.

.

.

.

Activity(last zone)

Balance table file

The contents of the balance table are defined in Table 9.1.4 and Table 9.1.5. The structure of the "balance table file" written to **LBTF** is:

Record 1 **KEFF, SP**

KEFF - $k_{effective}$ for problem

SP - search parameter for case

Record 2 - Sets of ASCII data consisting of a label record followed by data records.

Record last

A set of data is as follows (igp is the total number of groups plus one):

fine(few) group summary for zone *zzzzz* **set type**

Set type data(group 1)

Set type data(group 2)

.

.

.

.

Set type data(group igp)

The data for a set type will be written for each zone of the problem, plus a system summary if there is more than one zone. After one set type is finished, the next set type will be written. The order of the set types is as follows:

fixed source

fission source

absorption rate

total leakage

fission rate

flux

<n,2n> rate

buckling loss

right current

left current

right leakage

left leakage

The fine group summary data will be written if **LBTF** is > 0. After the fine group data is finished, the few group summary data will follow if a weighting calculation is specified with a broad group collapse.

Input and derived data file

The contents of the input and derived data file (specified by **LIDF**) is as follows:

Record 1 – **title** (80 characters)

Record 2 – **1$ $ array** (label)

Record 3,4 – **data from 1$ array**

Record 5 – **2$ $ array** (label)

Record 6 – **data from 2$ array**

Record 7 – **3$ $ array** (label)

Record 8,9 – **data from 3$ array**

Record 10 – **4$ $ array** (label)

Record 11 – **data from 4$ array**

Record 12 – **5** array** (label)

Record 13,14 – **data from the 5* array**

Record 15 – **cross section parameters** (label)

Record 16 – **total groups, neutron groups, gamma groups, first thermal group**

Record 17 – **nuclides on library** (label)

Records 17a – **list of nuclides on the cross section library**

Record 18 – **mixture numbers** (label)

Records 18a – **data from the 13$ array**

Record 19 – **component numbers** (label)

Records 19a – **data from the 14$ array**

Record 20 – **densities** (label)

Records 20a – **data from the 15* array**

Record 21 – **cccc identifiers** (label)

Records 21a – **data from the 16$ array**

Record 22 – **neutron energy group boundaries** (label)

Records 22a — **list of the energy boundaries for the neutron groups**

Record 23 – **neutron lethargy group boundaries** (label)

Records 23a – **list of the lethargy boundaries for the neutron groups**

Record 24 – **neutron weighted velocities** (label)

Record 24a – **list of the neutron average velocities**

Record 25 – **gamma energy group boundaries** (label)

Record 25a – **list of the energy boundaries for the gamma groups**

Record 26 – **gamma lethargy group boundaries** (label)

Records 26a – **list of the lethargy boundaries for the gamma groups**

Record 27 – **gamma weighted velocities** (label)

Records 27a – **list of the gamma velocities**

Record 28 – **broad group numbers** (label)

Records 28a – **list of the broad group numbers by fine group - 51$ array**

Record 29 – **group band** (label)

Records 29a – **group band numbers by fine group**

Record 30 – **calculation type** (label)

Records 30a – **calculation type by fine group**

Record 31 – **right albedo** (label)

Records 31a – **list of the right boundary albedos by group - 47* array**

Record 32 – **left albedo** (label)

Records 32a – **list of the left boundary albedos by group - 48* array**

Record 34 – **mixture by zone** (label)

Records 34a – **data from the 39$ array**

Record 35 – **order of scattering by zone** (label)

Records 35a – **data from the 40$ array**

Record 36 – **activity materials** (label)

Records 36a – **data from the 49$ array**

Record 37 – **activity reaction types** (label)

Records 37a – **data from the 50$ array**

Record 38 – **quadrature weights** (label)

Records 38a – **data from the 43* array**

Record 39 – **quadrature cosines** (label)

Records 39a – **data from the 42* array**

Record 40 – **weights x cosines** (label)

Records 40a – **product of quadrature weights times quadrature**

Record 41 – **reflected directions** (label)

Records 41a – **reflected direction index array**

Record 42 – **pl scattering constants** (label)

Records 42a – **constants for converting from discrete angles to Legendre moments**

Record 43 – **interval boundaries** (label)

Records 43a – **data from the 35* array**

Record 44 – **interval midpoints** (label)

Records 44a – **array containing the midpoints of each interval**

Record 45 – **zone by interval** (label)

Records 45a – **data from the 36$ array**

Record 46 – **interval boundary areas** (label)

Records 46a – **area of each interval boundary**

Record 47 – **interval volumes** (label)

Records 47a – **volume of each interval**

Record 48 – **interval density factors** (label)

Records 48a – **data from the 38* array**

Record 49 – **zone width modifiers** (label)

Records 49a – **data from the 41* array**

Record 50 – **source spectrum by interval** (label)

Records 50a – **data from the 30$ array**

Table 9.1.4: Balance table definitions.

FS = Fission Source$_{\text{grp,zone}} = 1/\lambda\Sigma_{\text{i}\subset\text{zone}} \left[X_{i,\text{grp}} \Sigma_{\text{grp}'} \left(\nu\Sigma_{\text{fgrp}',i} \varphi_{\text{grp}',\text{i}} \right) V_i \right]$
XS = Fixed Source$_{\text{grp,zone}} = \Sigma_{1\subset\text{zone}} \left[Q_{\text{grp},i} V_i + A_i \Sigma_{\mu m>0} BS_{i,\text{ grp},m} \mu_m w_m - A_{i+1} \Sigma_{\mu m<0} BS_{i,\text{ grp},m} \mu_m w_m \right]$
IS = Inscatter$_{\text{grp,zone}} = \Sigma_{\text{i}\subset\text{zone}} \sum_{j\neq\text{grp}} \left[\Sigma_{j\rightarrow\text{grp},i} \varphi_{j,i} V_i \right]$
SS = Selfscatter$_{\text{grp,zone}} = \Sigma_{\text{i}\subset\text{zone}} \left[\Sigma_{\text{grp}\rightarrow\text{grp}} \varphi_{\text{grp},\text{i}} \text{V}_\text{i} \right]$
OS = Outscatter$_{\text{grp,zone}} = \Sigma_{\text{i}\subset\text{zone}} \sum_{j\neq\text{grp}} \left[\sum_{\text{grp}\rightarrow j} \varphi_{\text{grp},i} V_i \right]$
AB = Absorption$_{\text{grp,zone}} = \Sigma_{\text{i}\subset\text{zone}} \left[\Sigma_{\text{abs grp},i} \varphi_{\text{grp},\text{i}} \text{V}_\text{i} \right]$
LK = Leakage$_{\text{grp,zone}} = \left[A_{\text{zr}} \Sigma_m \left(\psi_{m,\text{ zr}} \mu_m w_m \right) - A_{z1} \Sigma_m \left(\psi_{m,\text{ z1}} \mu_m w_m \right) \right]$
RF = Right Boundary Flux$_{\text{grp,zone}} = \Sigma_\text{m} \left(\psi_{\text{m,zr,grp}} \text{W}_\text{m} \right)$
LF = Left Boundary Flux$_{\text{grp,zone}} = \Sigma_\text{m} \left(\psi_{\text{m,zr,grp}} \text{W}_\text{m} \right)$
RL = Right Leakage$_{\text{grp,zone}} = \text{A}_{\text{zr}} \Sigma_\text{m} \left(\psi_{\text{m,zr,grp}} \mu_\text{m} \text{W}_\text{m} \right)$
LL = Left Leakage$_{\text{grp,zone}} = A_{zl} \Sigma_m \left(\psi_{m,zl,grp} \mu_m W_m \right)$
NN = n,2n Rate$_{\text{grp,zone}} = \Sigma_{i\subset\text{ zone}} \Sigma_{p\geq 2} \left[p/2 \Sigma_{n,pn} \varphi_{grp,i} V_i \right]$
FR = Fission Rate$_{\text{grp,zone}} = \sum_{i\subset\text{ zone}} \left[\Sigma_{fgrp,i} \varphi_{grp,i} V_i \right]$
DB = DB^2 Flux$_{\text{grp,zone}} = \sum_{i\subset zone} \left[D_{grp,i} B^2_{grp,i} \varphi_{grp,i} V_i \right]$
TF = Total Flux$_{\text{grp,zone}} = \sum_{i\subset zone} \left[\varphi_{grp,i} V_i \right]$
BAL = {FS+XS+IS+NN+max(LL,0)-min(RL,0)} / {OS+AB+max(RL,0)-min(LL,0)}

Table 9.1.5: Balance table definition symbols.

$\sum_{\text{i}\subset\text{zone}}$ is the sum over all intervals i in the zone
$\sum_{\text{grp}}$ is the sum over all groups grp
$\sum_{j\neq grp}$ is the sum over all groups j not equal to group grp
$\sum_\text{m}$ is the sum over the quadrature
$\sum_{\text{p}\geq 2}$ is the sum over all processes $\sum_{\text{n,pn}}$
λ = the eigenvalue
χ = the fission spectrum
V = the average number of neutrons produced in a fission
$\sum_\text{f}$ = the fission cross section
φ = the scalar flux
V = the volume of a mesh interval
Q = the volumetric external source in a mesh interval
A = the area of a boundary of a mesh interval
BS = the angular flux boundary source on an interval boundary
μ_m = the mth discrete angle of the quadrature
W_m = the mth weight of the quadrature
$\sum_{j\rightarrow grp}$ = the scattering cross section for scattering from group j to group grp
$\sum_{\text{grp}\rightarrow\text{j}}$ = the scattering cross section for scattering from group grp to group j
$\sum_{\text{grp}\rightarrow\text{grp}}$ = the scattering cross section for within-group scattering (i.e., from group grp to the same group grp)
$\sum_{\text{abs}}$ = the absorption cross section
ψ = the angular flux
A_{zr} = the area of the right-hand boundary of the zone
A_{zl} = the area of the left-hand boundary of the zone

continues or

Table 9.1.5 – continued from previous page

$\sum_{n,pn}$ = the cross section for producing p neutrons, p=2,3,. . . ,p an integer
D = the diffusion coefficient (used in providing a buckling correction for 2 and 3 dimensions)
B^2 = the buckling for the second and third dimensions (includes an extrapolation distance)
max(LL,0) means that a positive Left Leakage is a source into the zone
min(RL,0) means that a negative Right Leakage is a source into the zone. It is included with a minus sign to make it a pos
max(RL,0) means that a positive Right Leakage is a loss from the zone
min(LL,0) means that a negative Left Leakage is a loss from the zone. It is included with a minus sign to make it a positiv

9.1.8.2 XSDRNPM Mixed Cross Sections

When **IPRT** (2$$ array) is set > -1, XSDRNPM prints the mixed reaction rate cross sections that are used in its calculations. The order of the cross sections for each group is given below in Table 9.1.6. The diffusion coefficient is used in computing buckling corrections, and in some of the options for computing the current used in weighting the transport cross section. The upscatter cross section is used to checking upscatter convergence. The <n,2n> cross section is used in computing the balance for the balance tables. It is actually a weighted sum of all the multiple neutron exit reactions other than fission. These are all treated in XSDRNPM as scattering reactions. Chi is the fission spectrum, and is used to distribute the fission source in energy space. The fission cross section is used to compute the fission rate reported in the balance tables. The absorption cross section is used to compute the absorptions in the balance tables, and to compute the absorption term in the eigenvalue. Nu*fission cross section is used to generate the source term for all except a fixed source calculation. The total cross section is used to determine the neutron transport.

Table 9.1.6: Order of mixed reaction cross sections

1. Diffusion coefficient (for use in buckling corrections)
2. Upscatter cross section
3. <n,2n> cross section
4. Chi (fission spectrun)
5. Fission cross section
6. Absorption cross section
7. Nu*Fission cross section
8. Total cross section

9.2 NEWT: A NEW TRANSPORT ALGORITHM FOR TWO-DIMENSIONAL DISCRETE-ORDINATES ANALYSIS IN NON-ORTHOGONAL GEOMETRIES

M. A. Jessee, M. D. DeHart

ABSTRACT

NEWT (**N**ew **E**SC-based **W**eighting **T**ransport code) is a multigroup discrete-ordinates radiation transport computer code with flexible meshing capabilities that allow two-dimensional (2-D) neutron transport calculations using complex geometric models. The differencing scheme employed by NEWT, the Extended Step Characteristic approach, allows a computational mesh based on arbitrary polygons. Such a mesh can be used to closely approximate curved or irregular surfaces to provide the capability to model problems that were formerly difficult or impractical to model directly with discrete-ordinates methods. Automated grid generation capabilities provide a simplified user input specification in which elementary bodies can be defined and placed within a problem domain. NEWT can be used for eigenvalue, critical-buckling correction, and source calculations and it can be used to prepare collapsed weighted cross sections in AMPX working library format.

Like other SCALE modules, NEWT can be run as a standalone module or as part of a SCALE sequence. NEWT has been incorporated into the SCALE TRITON control module sequences. TRITON can be used simply to prepare cross sections for a NEWT transport calculation and then automatically execute NEWT. TRITON also provides the capability to perform 2-D depletion calculations, in which the transport capabilities of NEWT are combined with multiple ORIGEN depletion calculations to perform 2-D depletion of complex geometries. In the TRITON depletion sequence, NEWT can also be used to generate lattice-physics parameters and cross sections for use in subsequent nodal core simulator calculations.

ACKNOWLEDGMENTS

The author expresses gratitude to B. T. Rearden and S. M. Bowman for their supervision of the SCALE project and review of the manuscript. The author acknowledges R. Y. Lee of the U.S. Nuclear Regulatory Commission (NRC) and A. P. Ulses (formerly NRC) for their support of the development of NEWT. Appreciation is extended to G. Ilas, B. L. Broadhead, Deokjung Lee (formerly ORNL), and B. J. Ade for their review of this or previous versions of the manuscript. The efforts of Z. Zhong (Argonne National Laboratory), A. P. Ulses (formerly NRC), K. S. Kim, C. F. Weber, G. Ilas, and K. T. Clarno (Oak Ridge National Laboratory) in methods development and testing of the code have been invaluable in the continued evolution and improvement of the code.

9.2.1 INTRODUCTION

NEWT (**N**ew **E**SC-based **W**eighting **T**ransport code) is a two-dimensional (2-D) discrete-ordinates transport code developed based on the Extended Step Characteristic (ESC) approach [NEWTDeh92] for spatial discretization on an arbitrary mesh structure. This discretization scheme makes NEWT an extremely powerful and versatile tool for deterministic calculations in real-world non-orthogonal problem domains. The NEWT computer code evolved from the earlier proof-of-principle CENTAUR code [NEWTDeh92] and has been developed to run within SCALE. Thus, NEWT uses AMPX-formatted cross sections processed by other SCALE modules. If cross sections are properly prepared, NEWT can be run in stand-alone mode. NEWT can also be used within the TRITON control module for transport analysis, depletion analysis, and sensitivity and uncertainty analysis.

9.2.1.1 How to use this manual

This users' manual is intended to assist both the novice and the expert in the application of NEWT for transport analysis. As such, the document is divided into subsections, each with a specific purpose. Not all sections will be of value to all users. It is not intended that the user of this manual read through the manual from start to end. Rather, the manual is designed to serve as a reference, with each section meeting different needs. This introductory section has been written to provide a general overview of the background, nature, functionality, and applications of NEWT; it should prove of interest to users at all levels. Sect. 9.2.2 provides detail on the theory of NEWT in terms of derivations, equations, and relationships used in the NEWT solution. This information will be of interest to those with a background in transport methods desiring a comprehensive understanding of the NEWT solution scheme. However, this information may provide too much detail or simply not be relevant for the beginning user or someone desiring to improve or expand an existing model. These users will find Sect. 9.2.3 to be of more value, where input data requirements and formats are described in detail, along with examples of each data type. This information is supplemented by Sect. 9.2.4, in which complete sample inputs with descriptions of the features of each model are provided. Sect. 9.2.5 describes the components of an output listing obtained from a successful NEWT calculation.

9.2.1.2 Background

The radiation transport equation, a linearized derivative of the Boltzmann equation, provides an exact description of a neutral-particle radiation field in terms of the position, direction of travel, and energy of every particle in the field. Both stochastic (Monte Carlo simulation) and deterministic (direct numerical solution) forms of the transport equation have been developed and are used extensively in nuclear applications. Each approach has its strengths and weaknesses. Stochastic approaches are extremely effective for problems with complex geometries where the calculations of integral quantities, such as radiation dose and neutron multiplication factors, are desired. However, calculations to obtain accurate differential information, such as the neutron flux as a function of space and energy, can be difficult and inefficient at best and prone to inaccuracies (even if the integral quantity is correct). Deterministic techniques, such as integral transport, collision probability, diffusion theory, and discrete-ordinates methods, are better suited for problems where differential quantities, such as the neutron flux as a function of energy or space, are desired. However, integral transport, collision probability, and diffusion approximations are based on simplifying assumptions, which can limit their applicability. The discrete-ordinates approach is a more rigorous approximation to the transport equation but is typically very limited in its flexibility to describe complex geometric systems.

Discrete-ordinates approaches are derived from the integro-differential form of the Boltzmann transport equation, where space, time, and energy dependencies are normally treated by the use of a finite-difference grid, while angular behavior is treated by considering a number of discrete directions in space. The angular solution is coupled to a scalar spatial solution via some form of numerical integration. Because of the direct angular treatment of the discrete-ordinates approach, angularly dependent distribution functions can be computed; thus, this approach is the preferred method of solution in many specific applications where angular anisotropy is important. However, as indicated earlier, it is often limited in applicability because of the geometric constraints of the orthogonal grid system associated with the finite-difference numerical approximation.

9.2.1.3 Discrete-ordinates solution on an arbitrary grid

The ESC approach was developed to obtain a discrete-ordinates solution in complicated geometries to handle the needs of irregular configurations. Deterministic solutions to the transport equation generally calculate a solution in terms of the particle flux; the flux is the product of particle density and speed and is a useful quantity in the determination of reaction rates that characterize nuclear systems. General 2-D *xy* discrete-ordinates methods perform calculations that provide four side-averaged fluxes and a cell-averaged flux for each cell in a rectangular problem grid; iteration is performed to obtain a converged distribution. This approach is usually termed the diamond-difference approach. Using the ESC approach, a more flexible and completely arbitrary problem grid may be defined in terms of completely arbitrary polygons. Side-averaged fluxes for each polygon in the problem domain are computed and are used to calculate a cell-averaged flux. This process is repeated for each cell in the problem domain, and as with the traditional approach, iteration is performed for convergence. This geometric flexibility is a significant enhancement to existing technology, as it provides the capability to model problems that are currently difficult or impractical to model directly.

9.2.1.4 Functions performed

NEWT provides multiple capabilities that can potentially be used in a wide variety of application areas. These include 2-D eigenvalue calculations, forward and adjoint flux solutions, multigroup flux spectrum calculations, and cross section collapse calculations. NEWT provides significant functionality to support lattice-physics calculations, including assembly cross section homogenization and collapse, calculation of assembly discontinuity factors (for internal and reflected assemblies), diffusion coefficients, pin powers, and group form factors. Used as part of the TRITON depletion sequence, NEWT provides spatial fluxes, weighted multigroup cross sections, and power distributions used for multi-material depletion calculations and coupled depletion and branch calculations needed for lattice-physics analysis.

9.2.2 THEORY AND PROCEDURES

This section provides the theoretical basis for the ESC discretization technique, the NEWT solution algorithm, and cross section processing procedures used by NEWT. Although this information is not necessary to be able to use NEWT for transport calculations, it provides a deeper understanding of the basic operations performed within NEWT.

9.2.2.1 Boltzmann transport equation

The neutron transport equation may be presented in various forms, and simplifications are often applied to tailor the equation to the requirements of a specific application. In nuclear engineering applications, the transport equation is often written in terms of the angular neutron flux as the dependent variable. The angular neutron flux is defined as the product of the angular neutron density and the neutron velocity. The time-independent form of the linear transport equation is then expressed as [NEWTDH76]

$$\Omega \cdot \nabla\psi(\mathbf{r}, \Omega, E) + \sigma_t(\mathbf{r}, E)\psi(\mathbf{r}, \Omega, E) = Q(\mathbf{r}, \Omega, E), \tag{9.2.1}$$

where

$\psi(\mathbf{r}, \Omega, E) \equiv$ angular flux at position per unit volume, in direction Ω per unit solid angle and at energy E per unit energy;

$\sigma_t(\mathbf{r}, E) \equiv$ total macroscopic cross section at position $\mathbf{r}$ and energy E; and

Q ≡ source at position $\mathbf{r}$ per unit volume, in direction Ω per unit solid angle and at energy E per unit energy.

The source Q is generally composed of three terms:

1. a scattering source,

$$S(\mathbf{r}, \Omega, E) = \int_{4\pi} d\Omega' \int_0^\infty dE' \sigma_s(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E)\, \psi(\mathbf{r}, \Omega', E'), \tag{9.2.2}$$

where

$\sigma_s(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E) \equiv$ macroscopic scattering cross section at position $\mathbf{r}$ from initial energy E' and direction Ω' to final energy E and direction Ω,

2. a fission source,

$$F(\mathbf{r}, \Omega, E) = \chi(\mathbf{r}, E) \int_0^\infty dE' \nu(\mathbf{r}, E')\, \sigma_f(\mathbf{r}, E')\, \psi(\mathbf{r}, \Omega, E'), \tag{9.2.3}$$

where

$\sigma_f(\mathbf{r}, E')$ ≡ macroscopic fission cross section at position **r** and energy E' (assumed to be isotropic),

$\nu(\mathbf{r}, E')$ ≡ number of neutrons released per fission event at position **r** and energy E',

$\chi(\mathbf{r}, E)$ ≡ fraction of neutrons that are born at **r** and at energy E, and

3. an external or fixed source, S(**r** ,E).

In general, the transport equation can be difficult to apply and can be solved analytically only for highly idealized cases. Hence, simplifications and numerical approximations are often necessary to apply the equation in engineering applications. Traditional discrete-ordinates methods are based on a finite-difference approximation to solve the flux streaming (leakage) term. Such differencing schemes are intimately tied to the coordinate system in which the differencing equations are developed, and it becomes difficult to represent non-orthogonal volumes within that coordinate system. For example, it is not possible to exactly represent a cylinder in a 2-D Cartesian coordinate system; one must approximate the cylinder with a number of rectangular cells. A close approximation can require a large number of computational cells. However, the ESC approach for discretizing computational cells allows the use of non-orthogonal computational cells composed of arbitrary polygons. Using this method, practically any shape can be represented within a Cartesian grid to a very close approximation. The ESC approach is discussed in the following sections.

9.2.2.2 The step characteristic approximation

Efficient application of discrete-ordinates methods is difficult when dealing with complicated non-orthogonal geometries because of the nature of finite difference approximations for spatial derivatives. An alternative to the discrete representation of the spatial variable is achieved in the method of characteristics, in which the transport equation is solved analytically along characteristic directions within a computational cell. The angular flux is solved along the *s*-axis, where this axis is oriented along the characteristic direction Ω. Since only the angular flux in direction Ω is of concern, then the streaming term can be rewritten as

$$\Omega \cdot \nabla\psi(\mathbf{r}, \Omega, E) = \frac{d\psi(s, E)}{ds}. \tag{9.2.4}$$

Hence Eq. (9.2.1) can be written in the characteristic form (omitting E for clarity) as

$$\frac{d\psi(s)}{ds} + \sigma_t(s)\psi(\mathrm{s}) = Q(s), \tag{9.2.5}$$

which has a solution of the form [NEWTHil76]

$$\psi(s) = \psi_0 e^{-\sigma_t s} + e^{-\sigma_t s} \int_0^s Q e^{\sigma_t s'} ds', \tag{9.2.6}$$

where s is the distance along the characteristic direction Ω, and ψ_0 is the known angular flux at s=0. The value for ψ_0 is given from boundary conditions for known cell sides, and angular fluxes on unknown sides are computed using Eq. Eq. (9.2.6). Methods for the determination of an appropriate value for ψ_0 and for evaluation of the integral term vary in different solution techniques.[4-9] [NEWTAL81, NEWTALMJW79, NEWTHil76, NEWTLA81, NEWTLat68, NEWTLat69, NEWTLM84]. One of the simplest schemes employing the Method of Characteristics is the Step Characteristic (SC) method developed by Lathrop [NEWTAL81]. In this approach, the source Q and macroscopic total cross section σt are assumed to be constant within a computational cell and the angular flux is assumed constant on the cell boundaries of incoming direction. Integration of Eq. Eq. (9.2.6) can be performed to obtain

$$\psi(s) = \psi_0 e^{-\sigma_t s} + \frac{Q}{\sigma_t}\left(1 - e^{-\sigma_t s}\right). \tag{9.2.7}$$

Fig. 9.2.1 shows a sample computational cell in which the SC method can be applied. For a given characteristic direction Ω, the angular flux on any unknown side may be expressed in terms of a suitable average of fluxes from known sides, which contribute to the unknown side. For the characteristic direction Ω shown in Fig. 9.2.1, the unknown "top" flux ψ_T may be computed as a linearly weighted average of contributions from known sides ψ_B and ψ_L. The fluxes on each of the two known sides are taken to be constant along the length of each side, representing the average angular flux in direction Ω and must be specified from external boundary conditions or from a completed calculation in an adjacent cell.

The set of characteristic directions is chosen from a quadrature set, so that the resulting angular fluxes may be numerically integrated to obtain a scalar flux. Knowing the lengths of the sides of a rectangular cell (ΔDelta`y) and the direction cosines of Ω in the *x*-*y* plane (μ and η), a function for the length *s* can easily be determined. The solution for from Eq. Eq. (9.2.7) can then be integrated along the length of each unknown side to determine the average angular flux of the unknown side. Once the angular flux is known on all four sides, a neutron balance on the cell can be used to determine the cell's average angular flux.

Although the SC method described above is based on rectangular cells, the derivation of Eq. Eq. (9.2.7) makes no assumptions about the shape of the cell. It merely requires knowledge of the relationship between cell edges along the direction of the characteristic. Hence, the method is not restricted to any particular geometry. Because it is an extension of the SC approach into generalized cells, the method developed here for generalized geometries is referred to as the Extended Step Characteristic (ESC) method.

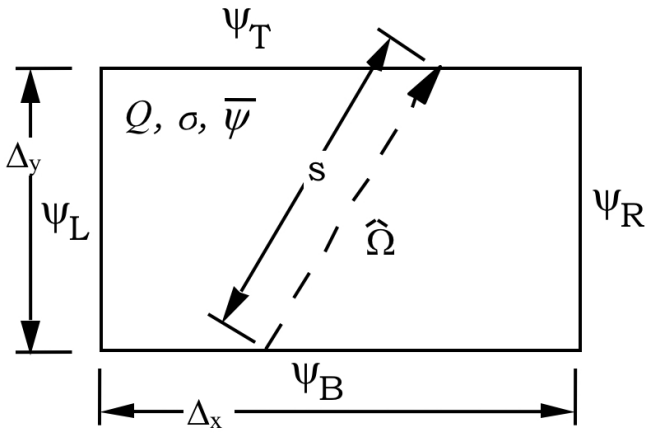


Fig. 9.2.1: Typical rectangular cell used in the step characteristic approach.

9.2.2.3 The Extended Step Characteristic approach

The theory of the ESC approach is developed and explained in detail in [NEWTDeh92]. However, the work has evolved significantly from that time, most notably in the elimination of a requirement for non-reentrant polygons (convex). The following subsections describe the primary equations applied in the ESC approach as currently applied in NEWT.

Cell properties and geometries

The two primary assumptions of the ESC method are that (1) within each computational cell all properties (i.e., σt and Q) are uniform and (2) cell boundaries are defined by straight lines. The restriction of a computational cell to boundaries consisting of a set of straight lines results in computational cells that are limited to polygons. However, as will be seen later, no restrictions are placed on the shape of the polygon or on the number of sides in the polygon. However, the size of the polygon will be limited. In practical applications, properties are unlikely to remain constant over significant volumes. Thus this approach, like many other differencing schemes, is a poor approximation when cell volumes become too large. Although σt is a material property and may remain spatially constant, the source term Q, which depends on the neutron flux, will vary with position. However, since the solution would become exact in an infinitesimally small cell, it is expected that the approximation will be reasonable for computational cells in which the change in the flux (and therefore the source) is small over the domain of the cell.

As a result of this geometric configuration, each side of a cell can have one of three possible attributes relative to particle flow in a given characteristic direction, as illustrated in Fig. 9.2.2: (1) flow can enter the cell when crossing a side (as shown by sides E and F in the figure); (2) flow can exit the cell when crossing a side (sides B and C); or (3) in a special case, flow may be parallel to the orientation of a given side (sides A and D). Expressed mathematically, these relationships become

$$\text{Category 1} : \Omega_k \cdot \hat{n}_i < 0 \tag{9.2.8}$$

$$\text{Category 2} : \Omega_k \cdot \hat{n}_i > 0 \tag{9.2.9}$$

$$\text{Category 1} : \Omega_k \cdot \hat{n}_i = 0 \tag{9.2.10}$$

where $\hat{n}_i$ is a unit vector in the cell-outward direction normal to side i, and Ω_k is the k^{th} discrete element of a set of characteristic directions. A category 1 side will be termed an "incoming" side with respect to the direction Ω_k, and a category 2 side will be referred to as an "outgoing" side. For simplicity, the definition of Eq. Eq. (9.2.10) will be included as a special case of Eq. Eq. (9.2.8) for an incoming side. Thus, Eq. Eq. (9.2.8) can be rewritten as

$$\text{Side } i \text{ is incoming with respect to } \Omega_k : \Omega_k \cdot \hat{n}_i \leq 0 \tag{9.2.11}$$

$$\text{Side } i \text{ is outgoing with respect to } \Omega_k : \Omega_k \cdot \hat{n}_i > 0 \tag{9.2.12}$$

To solve for fluxes (flow) on outgoing sides of a cell, one must know fluxes on all incoming sides. Each incoming side of each cell will be given from a boundary condition or will be the outgoing side of an adjacent cell.

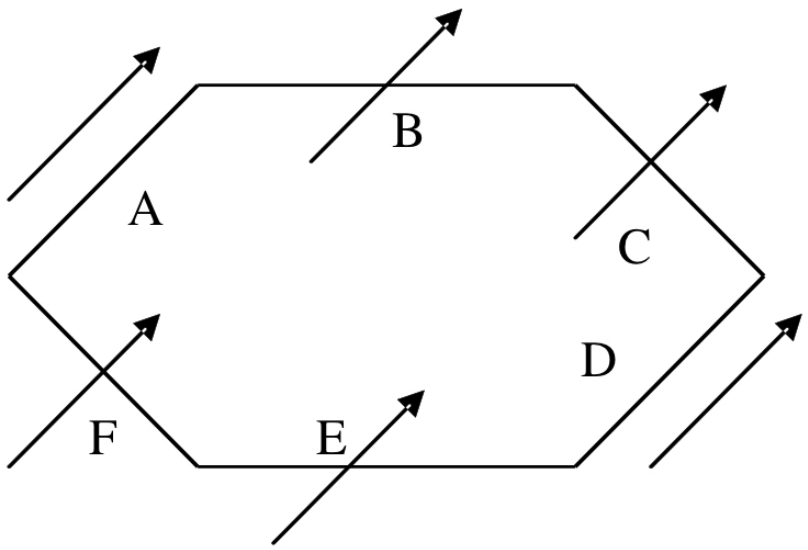


Fig. 9.2.2: Orientation of the sides of a cell with respect to a given direction vector.

Relationships between cells

In the ESC method, the shape of the computational cell and the form of the neutron balance differ from that used in traditional discrete-ordinates methods. Nevertheless, the relationships between cells are treated essentially as they would be in traditional approaches. The entire problem domain is mapped in terms of a set of finite cells. Each side of each cell is adjacent to either an external boundary condition or another cell. For each discrete direction, cells are swept in a predetermined order beginning at a known boundary (from a specified external boundary condition) moving in the given direction. The precise order of sweep is such that as the solution for one cell is obtained, the cell provides sufficient boundary conditions for the solution of an adjacent cell. Hence, cells sharing a given side share the value of the angular flux on that side. Knowledge of the flux on all incoming sides of a cell is sufficient to solve for all outgoing sides. Once the angular flux has been determined for all sides of the cell for the given direction, it is possible to use a neutron balance to compute the average value of the angular flux within the cell.

The sweeping of cells continues for a given direction until all cell fluxes have been calculated. The procedure is then repeated for the next direction until all directions have been computed. At this point, the cell average angular fluxes are known for each cell for each direction used. Numerical quadrature can then be used to determine the average scalar flux in each cell in the problem domain. The scalar fluxes are used to determine fission and scattering reaction rates in each cell and to update the value of the cell average source, Q. The process is repeated, and the iteration continues until all scalar fluxes converge to within a specified tolerance.

This approach can be performed assuming a single energy group or any number of discretized energy groups. The multigroup approach used in the ESC method is the standard approach used in most multigroup methods and is independent of the shape of each computational cell. Hence, the details of the multigroup formalism will be omitted from this discussion.

The set of characteristic directions

The characteristic solution to the transport equation gives only the angular flux in the direction of the characteristic direction vector Ω_k. To compute interaction rates within a cell, one must compute scalar fluxes. In computing the scalar flux from the set of angular fluxes, it is convenient to choose the set of characteristic directions from an appropriate quadrature set. Then the set of computed angular fluxes can be combined with appropriate directional weights and summed to obtain a scalar flux solution within a cell. Therefore, it is most appropriate to choose characteristic directions from an established set of base points and weights. Such quadrature sets that have been developed and used in numerous earlier discrete- ordinates approaches are used in NEWT. No restriction is placed on the nature or order of the quadrature set, as long as it is sufficient to adequately represent the scalar flux from computed angular fluxes.

Angular flux at a cell boundary

As in the development of the SC method, as well as most finite-difference methods, the ESC approach does not explicitly determine the flux distribution as a function of position along the sides of a computational cell. Instead, the angular flux on each cell side is represented in terms of the average angular flux along the length of the side. This is sufficient to determine the net leakage across each cell side, which is needed in order to maintain a cell balance. An average value of the flux for an incoming side must be specified from a boundary condition or from the prior solution of an adjacent cell. The average flux along a given outgoing side can be computed by integrating the flux along the side and dividing by the length of the side. However, the form of the distribution of the angular flux on the side must be known to perform this integration. This distribution can be determined from the properties of the cell and from the average flux on each of the known incoming sides.

Because the characteristic solution [Eq. Eq. (9.2.6)] allows calculation of the angular flux at any point s in a single cell given an initial condition, the exact value of the flux can be computed at any point on any outgoing side if the flux along each incoming side is known. As an initial condition, it is assumed that the angular flux in some characteristic direction is known at some starting point, $s = 0$ [i.e., $\psi(0) = \psi_0$], on an incoming side. To determine the flux at some point on an outgoing side, one need know only the distance s measured along a characteristic direction to the appropriate incoming side. This method can then be expanded to determine a functional form of the flux for every point on the outgoing side, which can be integrated to produce the average outgoing flux on the side.

To develop a mathematical relationship between two arbitrary sides of a cell, one should first consider two arbitrary coplanar line segments in space whose endpoints each lie on a pair of parallel lines laid in the direction Ω_k, as shown in Fig. 9.2.3. Points B_1 and B_2 can be considered to be the "projections" of A_1 and A_2, respectively, relative to Ω_k. Because s is the distance between a point on segment A and its projection on segment B, it can be seen that s varies linearly in moving from the "beginning" to the "end" of the pair of segments.

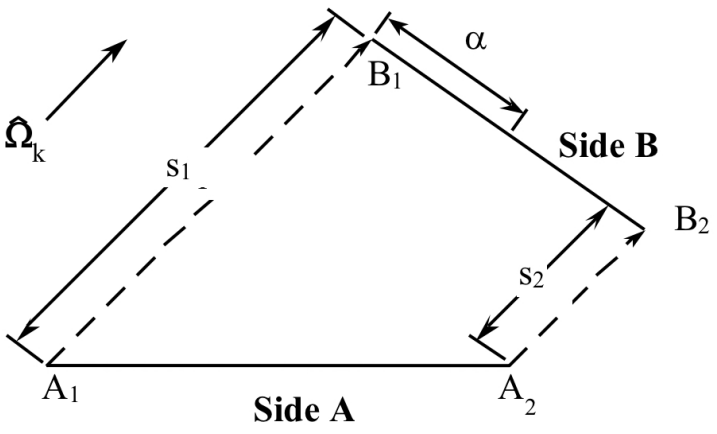


Fig. 9.2.3: Line endpoints for computation of average fluxes.

If α is the distance along segment B measured from endpoint B_1 and B has a total length L, then the distance s between A and B along direction Ω_k can be written as a linear function in terms of the position α:

$$s(\alpha) = s_1 + \left(\frac{s_2 - s_1}{L}\right)\alpha, \tag{9.2.13}$$

where s_1 and s_2 are related to the distances along the characteristic direction between A_1, B_1 and A_2, B_2, respectively. (It is important to note that the length s is the same as the distance between the endpoints only when the characteristic vector lies in the plane of the computational cell. This is not necessarily the case, depending on the choice of quadrature directions. This situation is discussed in more detail later.)

If $\psi(\alpha)$ is the angular flux on side B at a distance α from B_1, then $\bar{\psi}_B$, the average value of ψ on B, is given by

$$\bar{\psi}_B = \frac{\int_0^L \psi(s(\alpha))d\alpha}{\int_0^L d\alpha}. \tag{9.2.14}$$

Equation Eq. (9.2.6), the solution to the characteristic equation in the step approximation, can be rewritten in terms of the average known angular flux on side A

$$\psi_B(s) = (\bar{\psi}_A - Q/\sigma_t)\, e^{-\sigma_t s} + Q/\sigma_t. \tag{9.2.15}$$

Inserting Eqs. Eq. (9.2.13) and Eq. (9.2.15) into Eq. Eq. (9.2.14) and simplifying yields

$$\bar{\psi}_B = \frac{1}{L}\int_0^L \left[(\bar{\psi}_A - Q/\sigma_t)\exp\left(-\sigma_t\left(s_1 + \left(\frac{s_2 - s_1}{L}\right)\alpha\right)\right) + Q/\sigma_t\right] d\alpha. \tag{9.2.16}$$

For the special case in which A and B are parallel, $s_1 = s_2$ and the second term in the exponential drops out. Equation Eq. (9.2.16) can easily be integrated to obtain

$$\bar{\psi}_B = (\bar{\psi}_A - Q/\sigma_t)\, e^{-\sigma_t s_1} + Q/\sigma_t. \tag{9.2.17}$$

In the more general case, $s_1 \neq s_2$, the result is slightly more complicated:

$$\bar{\psi}_B = \frac{(\bar{\psi}_A - Q/\sigma_t)}{\sigma_t\,(s_2 - s_1)}\left[e^{-\sigma_t s_1} - e^{-\sigma_t s_2}\right] + Q/\sigma_t. \tag{9.2.18}$$

Equations Eq. (9.2.17) and Eq. (9.2.18) can also be written in a simplified form:

$$\bar{\psi}_B = \beta_{AB}\bar{\psi}_A + (1 - \beta_{AB})\, Q/\sigma_t \tag{9.2.19}$$

where

$$\beta_{AB} = \begin{cases} \frac{e^{-\sigma_t s_1} - e^{-\sigma_t s_2}}{\sigma_t (s_2 - s_1)} & s_1 \neq s_2 \\ e^{-\sigma_t s_1} & s_1 = s_2 \end{cases} \tag{9.2.20}$$

Thus far, this development has considered only the special case where contributions to side B are the result only of the cell internal source and a single incoming side (i.e., side A). For an arbitrarily shaped cell and discrete direction Ω_k, it is likely that the outgoing side would receive contributions from two or more incoming sides, as illustrated in Fig. 9.2.4, for a cell with three incoming sides (X, Y, and Z) contributing to the flux on a single outgoing side (B). In such a situation, the outgoing side can be subdivided into multiple components. Side B of Fig. 9.2.4 can be represented by three components, B_X, B_Y, and B_Z, representing contributions from line segments X, Y, and Z, respectively. The average angular flux $\bar{\psi}$ can be computed for each component of side B using Eq. Eq. (9.2.19); then $\bar{\psi}_B$, the average flux for the entire length of B, can be calculated by the length-weighted average of each component. In general, for a given side B composed of n components, the average flux of the side is given by

$$\bar{\psi}_B = \sum_{i=1}^{n} \frac{\bar{\psi}_i i}{L_B} \tag{9.2.21}$$

where

ℓ_i is the length of the projection of the ith side onto B, and

$\bar{\psi}_i$ is the average flux computed for segment B_i due to the flux on side i

Using Eqs. Eq. (9.2.19) and Eq. (9.2.21), one can compute the average flux on each of the outgoing sides for a given cell, once the angular flux on each incoming side is known. At this point, only distances s_1 and s_2 and the lengths ℓ_i and L need be determined to estimate fluxes in an iterative process. These can be computed from the geometry of the cell and the direction Ω_k.

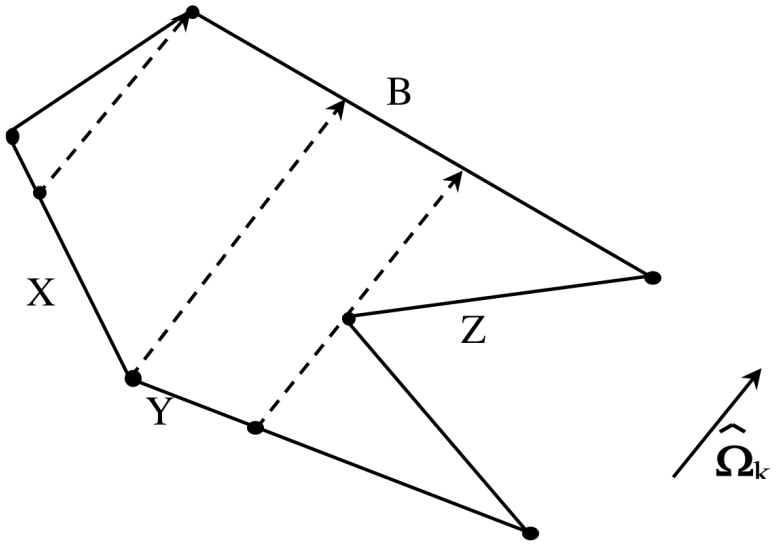


Fig. 9.2.4: Contributions of multiple incoming sides to an outgoing side.

Mapping a characteristic vector into the two-dimensional problem domain

Even in a 2-D x-y system in which the scalar flux is constant with respect to the z axis, the angular flux has components in the z direction. Thus, to obtain the scalar flux at a point on the x-y plane, one must integrate over the unit sphere in all 4π directions of Ω. Recall that the choices of characteristic directions for this model were selected to be the same as the set of directions composing a conventional quadrature set. Quadrature sets specified in the literature [NEWTCL65, NEWTCar70, NEWTLee62] and used in other discrete-ordinates codes [NEWTEJ67, NEWTLB73] are based on a unit sphere and are usually specified in terms of μ_k and η_k, the respective x and y components of Ω_k, where is one of a set of discrete directions composing the quadrature set. Because Ω_k is a unit vector, ξ_k, the z component of the direction, is implicit: $\xi_k = \sqrt{1-\mu_k^2-\eta_k^2}$. However, because of the 2D nature of the problem, the z component is never explicitly used. It is therefore sufficient to evaluate the angular flux at a finite number of points in 4π of Ω-space in terms of just the μ_k and η_k components of the discrete directions Ω_k. One must recognize, however, that the length of the path traveled by particles moving in a direction out of the x-y plane is always longer than the x-y projection of the path, by a factor of $\left(\mu^2+\eta^2\right)^{-1/2}$. Thus, for any path length s' measured in the x-y plane for a given direction Ω_k, the true path length traveled is s, where

$$s = \frac{s'}{\sqrt{\mu^2+\eta^2}}. \tag{9.2.22}$$

This is illustrated in Fig. 9.2.5.

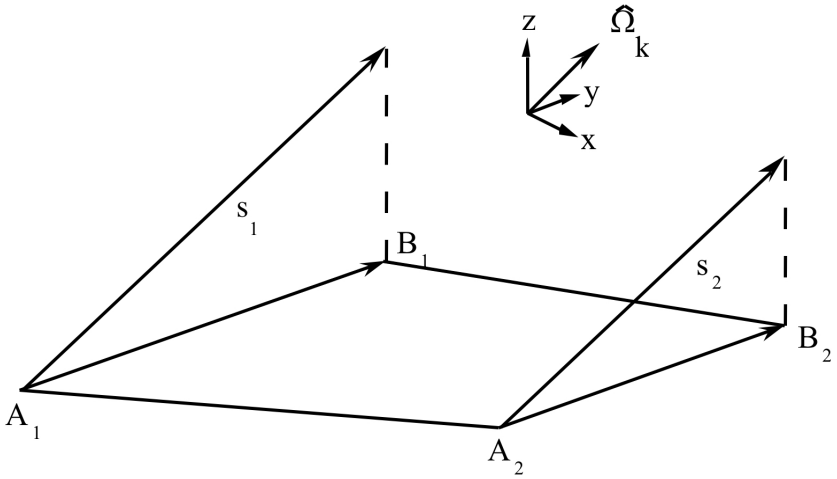


Fig. 9.2.5: Relationship between s_1 and s_2 and their projections in the x-y plane.

Neutron balance within a computational cell

Once angular fluxes have been computed for all sides of a cell, it is necessary to compute the cell-averaged angular flux. To enforce conservation, a balance condition is applied to the cell. This provides the equation necessary to determine the average flux in the cell. The neutron balance for an arbitrary cell in steady state may be expressed as

$$\begin{bmatrix} \text{net number of} \\ \text{neutrons moving in} \\ \text{direction } \hat{\Omega} \text{ escaping} \\ \text{from the cell} \end{bmatrix} + \begin{bmatrix} \text{number of neutrons} \\ \text{removed from the cell} \\ \text{or from direction } \hat{\Omega} \\ \text{by interactions} \end{bmatrix} = \begin{bmatrix} \text{number of} \\ \text{neutrons produced} \\ \text{in the cell moving} \\ \text{in direction } \hat{\Omega} \end{bmatrix} \tag{9.2.23}$$

or, expressed mathematically,

$$\oint_s n \cdot \hat{\Omega}_k \psi dS + \sigma_t \bar{\psi} V = QV, \tag{9.2.24}$$

where n is the outward normal direction at each side of the cell and V is the 2-D volume of the cell. Note that in this context, S represents the surface area or perimeter of the cell. Hence, for a cell with m sides, each of the sides having a constant angular flux $\bar{\psi}_i$ and an outward normal direction n_i,

$$\bar{\psi}_{cell} = \frac{Q}{\sigma_t} - \frac{1}{\sigma_t V} \sum_{i=1}^{m} \bar{\psi}_i \int_{S_i} \mathrm{n}_i \cdot \hat{\Omega}_k dS_i. \tag{9.2.25}$$

Because each cell is restricted to be a polygon, each side in the cell will be a straight line and $\mathrm{n}_i \cdot \hat{\Omega}_k$ will be constant along the length of the side. Equation Eq. (9.2.25) can then be simplified to obtain

$$\bar{\psi}_{cell} = \frac{Q}{\sigma_t} - \frac{1}{\sigma_t V} \sum_{i=1}^{m} \bar{\psi}_i \left(\mathrm{n}_i \cdot \hat{\Omega}_k \, \mathrm{L}_i \right), \tag{9.2.26}$$

where $\mathrm{L_i}$ is the length of the *i*th side and the term in parentheses represents a leakage coefficient for the side.

9.2.2.4 Coarse-mesh finite-difference acceleration

Beyond cell discretization and solution described above for the ESC approach, the NEWT iterative approach is similar to that used in other discrete-ordinates methods. Inner iterations are used to solve spatial fluxes in each energy group to generate updated source terms; outer iterations use these source terms to converge all energy groups. This source-iteration approach can be somewhat slow to converge, especially when significant scattering is present. Hence, it is desirable to apply some form of acceleration to the iterative solution used by NEWT. To this end, a coarse-mesh finite-difference acceleration (CMFD) approach has been added to NEWT. The CMFD formulation uses a simplified representation of a complex problem, in which selected rectangular regions are derived from the global NEWT Cartesian grid and homogenized. The CMFD formulation utilizes coupling correction factors for each homogenized cell to dynamically homogenize the constituent ESC-based polygonal cells during the iterative solution process such that the heterogeneous transport solution can be preserved. Dynamic-group collapse is also possible with a two-level CMFD formulation in which alternating multigroup and two-group calculations are performed. By extending the concept of the equivalence theory to energy and angle, it is possible to apply a consistent lower-order formulation in the form of a homogenized pin-cell, few-group, diffusion-like finite-difference scheme. This simplified lower-order formulation is much less expensive to solve, and its solution can be used to accelerate the original higher-order transport solution in NEWT, resulting in much faster convergence of the fission and scattering source distributions. This work is described in detail in [NEWTZDX+08] and in previous versions of the NEWT manual.

Although the original implementation of the CMFD acceleration method is extremely efficient and actively maintained, its use is limited to rectangular-domain configurations (e.g., square-pitched fuel lattices). An alternative CMFD acceleration method has been developed to support triangular- and hexagonal-domain configurations (e.g., triangular-pitched fuel lattices such as the VVER or prismatic graphite models). The new CMFD acceleration method does not require the coarse-mesh cells to be rectangles but rather arbitrary polygons. However in the current implementation, the "unstructured" coarse-mesh cells are still constructed from the global NEWT Cartesian grid. Therefore, for a hexagonal configuration, interior coarse-mesh cells will be rectangular shape whereas cells near the boundary will be triangular or trapezoidal shapes.

The new unstructured CMFD iterative solution scheme is essentially identical to the original solution scheme; the two methods differ only in how the lower-order system is solved. Additionally the two-group acceleration is not employed in the unstructured CMFD method. Input options for both CMFD methods are described in Sect. 9.2.3.2.

9.2.2.5 Assembly discontinuity factors

In nodal multi-assembly or core calculations, lattice transport solutions are used to generate few-group homogenized cross sections. These cross sections are in general obtained from single-assembly transport calculations with zero-current boundary conditions. Generation of few-group homogenized cross sections for nodal calculations typically includes the generation of discontinuity factors (i.e., additional parameters used to preserve both reaction rates and the interface currents in the homogenization process). The discontinuity of the flux at an assembly interface that can arise by the use of homogenized cross sections is illustrated in Fig. 9.2.6. The so-called "homogeneous" flux, computed in the nodal calculation, is discontinuous at the assembly interface, as opposed to the exact "heterogeneous" flux, computed in the transport calculation, which is continuous at the assembly interface. The interface condition employed in nodal calculations between two assemblies (nodes) i and i+1 is given as

$$\phi^{+}_{i,\,\text{homogeneous}} \cdot F^{+}_{i} = \phi^{-}_{i+1,\,\text{homogeneous}} \cdot F^{-}_{i+1}, \tag{9.2.27}$$

where F^{+}_{i} and F^{-}_{i+1} are assembly discontinuity factors (ADFs) on each side of the interface between assemblies i and i+1.

The ADF on the assembly interface is defined as the ratio of the heterogeneous flux $\phi_{\text{heterogeneous}}$ at that assembly interface to the homogeneous flux evaluated at the interface, denoted $\phi^{+}_{i,\,\text{homogeneous}}$ (or $\phi^{-}_{i+1,\,\text{homogeneous}}$):

$$F^{+}_{i} = \frac{\phi_{\text{heterogeneous}}}{\phi^{+}_{i,\,\text{homogeneous}}}, F^{-}_{i+1} = \frac{\phi_{\text{heterogeneous}}}{\phi^{-}_{i+1,\,\text{homogeneous}}}. \tag{9.2.28}$$

Fluxes, and therefore ADFs, vary with energy; therefore, few-group homogenized cross sections are always accompanied by corresponding few-group ADFs.

In a single-assembly calculation with zero-current boundary conditions, the heterogeneous flux at each boundary is easily calculated as the surface-averaged scalar flux on the boundary, whereas the homogenous flux at each boundary is simply the assembly-averaged flux. Hence, for each energy group, the ADF is calculated for each boundary as the ratio of the average flux on that boundary to the average flux across the assembly.

In other configurations, such as a multi-assembly calculation or an assembly located on the edge of a core next to the core baffle and reflector, the ADF calculation requires more effort. For reflector situations, NEWT applies a simple one-dimensional (1-D) multigroup diffusion approximation to determine the ADF at the assembly boundary. In this approximation, it is assumed that the reflector is infinite and that the scalar flux goes to zero at infinity. The reflector ADF can be determined analytically using this boundary condition along with the known surface-averaged current and scalar flux evaluated at the assembly/reflector interface.

The reflector ADFs computed by NEWT may potentially be different from the ADFs calculated using the diffusion approximations employed by the nodal code. Moreover, ADFs computed for multi-assembly or hexagonal-domain configurations will depend on the nodal method employed. For these reasons, NEWT supports the option to edit surface-averaged scalar flux and current values along user-defined line segments so that appropriate ADFs can be computed directly by the nodal code. The input options for the single-assembly ADF, reflector ADF, and arbitrary line-segment edit are discussed in Sect. 9.2.3.11.

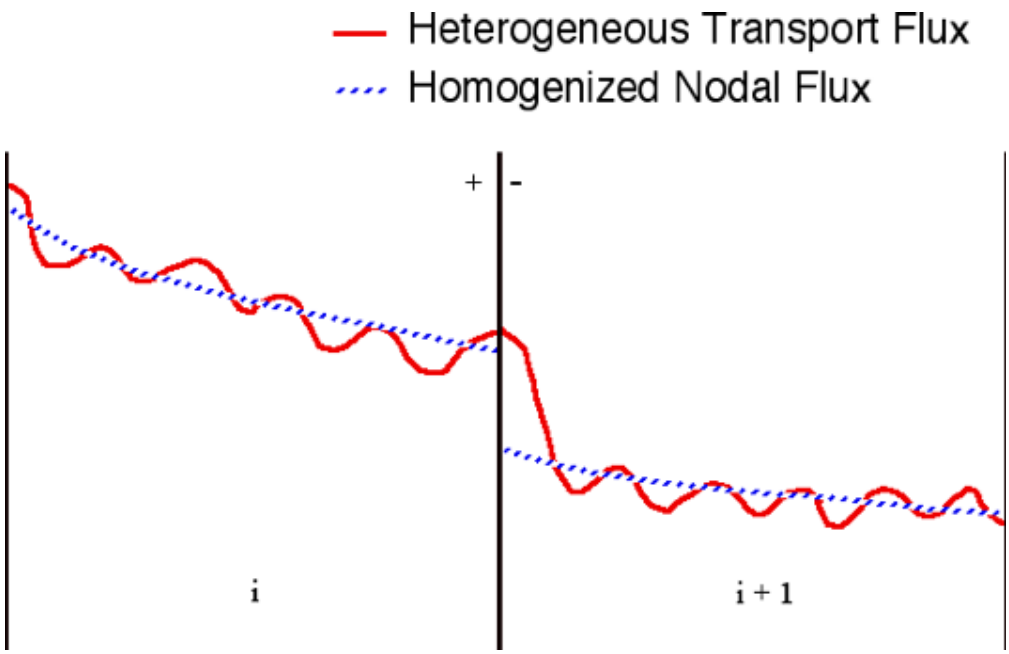


Fig. 9.2.6: Heterogeneous vs homogeneous fluxes in a multi-assembly solution.

9.2.3 INPUT FORMATS

NEWT input is free form and keyword based, similar in form to the input for many other modules in the SCALE code package. Input may start with a title card record, but this line may be omitted if desired; remaining data are supplied in data blocks. The order of the data blocks is arbitrary (with two exceptions), and many blocks are optional. Only one instance of a data block is allowed.

9.2.3.1 Overview of newt data blocks

The NEWT input deck data blocks are defined by keyword delimiters in the following form:

```
read keyword        [data]      end keyword
```

Read routines are terminated by the "end *keyword*" label, and any intervening carriage returns or line feeds are ignored. Thus, data can also be entered in this format:

```
read keyword
[data]
[data]
end keyword
```

Within each block, specific control or specification parameters are input. Each block contains a fixed set of input parameters (also defined by keyword).

As with other keyword-driven modules within SCALE, lines beginning with a single quote (') in the first column are treated as comments and ignored.

The keyword name and general contents of each data block are as follows:

Block type	Recognized keywords	Description
Problem control parameters Sect. 9.2.3.2	parameter, parameters, param, parm, para	General problem parameters-r
Material properties Sect. 9.2.3.3	material, materials, matl	Assigns characteristics (e.g., F
Broad group collapse Sect. 9.2.3.5	collapse, coll	Defines broad group energy ra
Simple-body geometry Sect. 9.2.3.6	geometry, geom	Defines basic grid structure ar
Boundary conditions Sect. 9.2.3.7	bounds, bnds	Defines boundary conditions t
Array specifications Sect. 9.2.3.9	array	Defines composition of all arr
Homogenization instructions Sect. 9.2.3.10	homog, hmog, homo	Defines mixtures to be flux we
Assembly discontinuity factors Sect. 9.2.3.11	adf	Assigns type and location of p
Flux plane Sect. 9.2.3.12	flux	Allows definition of an x- or y
Mixing table Sect. 9.2.3.13	mixtable, mixt	Mixing table specification (op
Source definition Sect. 9.2.3.4	src, source	Defines particle source strengt

Each of the following subsections describes the parameters associated with a specific data block, lists default values (if available), and describes meaning of the parameter and its effect on a NEWT calculation.

9.2.3.2 Parameter block

Parameter Block keyword = param, parm, para, parameter, or parameters

The Parameter block contains problem control parameters and must come immediately after the title card if one is used. Valid parameter specifications are described below. For each keyword, allowable values are listed in parentheses, and the default (if any) is listed in brackets. Input that can take an arbitrary integer value is indicated by an *IN*; similarly, any parameter that can take an arbitrary real/floating point value is indicated by *RN* as the allowable value. However, note that SCALE read routines do allow input of integers for real numbers, and vice versa; the number will be converted accordingly. The order of the parameters within the block is arbitrary, and may be skipped if a default value is desired for that parameter. Control parameters are set in the order in which they are input; this means that the same parameter may be listed multiple times, but only the final value is used.

Convergence and acceleration parameters

epseigen=(*RN*) - Convergence criterion for k_{eff}. [0.0001]

epsinner=(*RN*) - Spatial convergence criterion for inner iterations. [0.0001]

epsouter=(*RN*) - Spatial convergence criterion for outer iterations. [0.0001]

epsthrm=(*RN*) - Spatial convergence criterion for thermal-upscattering iterations, if enabled. [same value as **epsouter**]

epsilon=(*RN*) - Simultaneously sets all (spatial and eigenvalue) convergence criteria to the same value. [uses individual defaults]

converg=(*cell/mix*) - Sets the region within which convergence testing is applied. Use of *cell* will force converged scalar fluxes in every computation cell, while *mix* will relax convergence such that averaged scalar fluxes within a mixture are converged. The latter is useful for mixtures in which fluxes become very small-large reflectors or near a vacuum BC. [cell]

therm=(*yes/no*) - Enables/disables thermal-upscattering iterations. [yes]

inners=(*IN*) - Maximum number of inner iterations in an energy group. [5]

therms=(*IN*) - Maximum number of thermal-upscattering iterations, if enabled. [2]

outers=(*IN*) - Maximum number of outer iterations. NEWT will stop with an error code if more than *outers* outer iterations are required for convergence. [250]

inrcvrg=(*yes/no*) - If inrcvrg=yes, NEWT will continue outer iterations until all convergence criteria are met. If inrcvrg=no, NEWT will stop whenever outer iteration and k_{eff} convergence criterion are met, regardless of the convergence of inner or thermal-upscattering iterations. [no]

cmfd=(*no/rect/yes/part*) - CMFD acceleration option. If cmfd=no, CMFD acceleration is not employed. If cmfd=rect, the CMFD method is employed. The original NEWT CMFD method can be applied only to rectangular-domain configurations. If cmfd=yes, the unstructured CMFD method is employed. The new unstructured CMFD method can be applied to rectangular-, triangular-, and hexagonal- domain configurations. If cmfd=part, an alternative version of the unstructured CMFD method is employed and uses a "partial-current" acceleration scheme. Alternatively, users can use cmfd=0/1/2/3 for no, rect, yes, and part, respectively. [no]

cmfd2g=(*yes/no*) - Enables/disables the second-level two-group CMFD accelerator within the CMFD solver. This parameter has an effect only when cmfd=rect is set. [yes]

accel=(*yes/no*) - Enables/disables source (k_{eff}) acceleration. This parameter is automatically disabled if unstructured CMFD is employed (cmfd=yes or cmfd=part). [yes]

xcmfd=(*IN),* **ycmfd**=(*IN),* **xycmfd**=(*IN)* - These inputs specify the number of fine-mesh cells in the global NEWT grid per coarse-mesh cell. These options are used only when CMFD acceleration is enabled. The parameter *xcmfd* specifies the number fine-mesh cells per coarse-mesh cell in the x-direction. Likewise, *ycmfd* specifies the number of fine-mesh cells per coarse-mesh cell in the y-direction. The parameter *xycmfd* simultaneously sets *xcmfd* and *ycmfd* to the same value. In a special case for rectangular-domain configurations in which the entire domain is completely filled by a square-type array (see Sect. 9.2.3.9), *xycmfd=0* sets the coarse mesh based on the size of the array elements.

Important: User Guidance: Default convergence parameters are recommended for general analysis. Larger convergence criteria are useful for debugging if shorter run time is desired over solution accuracy. Smaller convergence criteria are recommended for generating reference solutions or benchmark calculations. CMFD acceleration should be applied whenever possible. The CMFD method with second-level 2-group acceleration should be applied for rectangular-domain configurations [e.g., light water reactor (LWR) assembly models (**cmfd=rect**), by default **cmfd2g=yes**]. The unstructured CMFD method should be applied for triangular- or hexagonal-domain configurations (**cmfd=yes**). If NEWT detects an unstable CMFD condition, a warning message is printed and NEWT continues with CMFD disabled. NEWT may also provide a terminating error message if improper selection of the coarse mesh is detected. Internal investigation has shown that the coarse mesh should be approximately the same size as the unit cell used in the model. For LWR assembly models, a fine mesh of 4 x 4 is recommended for the square-pitched unit cell, implying that **xycmfd** should be 4 only if the global unit has a mesh. If individual meshes are used in each unit definition, then the global unit coarse-mesh cells should be sized based on the unit cell size and, therefore, **xycmfd=1** should be used. The values of **xcmfd** and **ycmfd** do not have to be a common factor of the number of fine-mesh cells in a given direction (NEWT will make the last coarse-mesh cell smaller than the other coarse-mesh cells), but it is highly recommended.

Users can gauge solution convergence by the outer iteration edit as it is printed to the terminal window (echo=yes, see below). One can terminate a calculation prematurely (via the Control-C option on most platforms) if convergence or iteration parameters need to be modified.

In adjoint mode, CMFD acceleration is not currently supported and NEWT automatically disables its use if cmfd=yes, =rect, or =part. In adjoint mode with defined fixed source [i.e., generalized perturbation theory (GPT) analysis], it is observed that tighter convergence and iteration parameters are needed to properly remove fundamental mode contamination. (For more details, see SAMS chapter: Generalized Perturbation Theory.) To facilitate the CMFD options and larger convergence criteria for the forward calculations as well as smaller convergence criteria for GPT adjoint calculations, the following parameters are also available.

---

gptepsinner=(RN) - Spatial convergence criterion for inner iterations in GPT analysis. [0.0001] **gptepsouter**=(RN) - Spatial convergence criterion for outer iterations in GPT analysis. [0.001] **gptepsthrm**=(RN) - Spatial convergence criterion for thermal-upscattering iterations, if enabled, in GPT analysis. [same value as **gptepsouter**] **gptsepsilon**=(RN) - Simultaneously sets all spatial convergence criteria to the same value in GPT analysis. [uses individual defaults] **gpttherm**=(yes/no) - Enables/disables thermal-upscattering iterations in GPT analysis. [yes] **gptinners**=(IN) - Maximum number of inner iterations in an energy group in GPT analysis. [500] **gpttherms**=(IN) - Maximum number of thermal-upscattering iterations, if enabled, in GPT analysis. [10] **gptouters**=(IN) - Maximum number of outer iterations in GPT analysis. NEWT will stop with an error code if more than outers outer iterations are required for convergence. [2000]

Important: User Guidance: Default values for GPT convergence may change with future releases, as more experience is gained and user feedback is received. If the GPT calculation is not converging because of fundamental mode contamination, it is recommended that convergence criteria be decreased and/or inner and thermal-upscattering iteration limits be increased. If the solution convergence is slow, **gptinners** can potentially be decreased. Again, it is highly recommended that **echo=yes** be used to monitor speed of convergence.

Output editing

drawit=(*yes/no*) - Create a PostScript file showing the grid structure determined from input. Two files are created-the first showing the grid structure and the second showing the material placement. (Features and use of this simple graphics capability are described further in Sect. 9.2.5.12.1) [no]

echo=(*yes/no*) - During the iteration phase of execution, output is generated at the beginning of each outer iteration. This same information can be printed to SCALE message file (.msg) during iteration by setting echo=yes. [no]

prtbalnc=*(yes/no)* - Flag indicating whether or not balance tables for fine-group mixtures should be printed. [no]

prtbroad=(*yes/no/1d*) - Flag indicating whether or not broad group cross sections should be printed in problem output. The *1d* option indicates that 2-D scattering tables are not to be printed. This flag has no effect if collapse=no is specified. [no]

prthmmix=(*yes/no*) - Flag indicating whether or not homogenized mixture macroscopic cross sections should be printed in problem output. Homogenized cross sections are printed only if Homogenization Block is provided (Sect. 9.2.3.10). [yes]

prtflux=*(yes/no)* - Create a PostScript plot file showing flux distribution for each energy group in problem. If an energy collapse is performed, a second plot file is generated for the fluxes of the collapsed group structures. [no]

prtmxsec=(*yes/no/1d*) - Flag indicating whether or not mixture macroscopic cross sections should be printed in problem output. The *1d* option indicates that 2-D scattering tables are not to be printed. [no]

prtmxtab=(*yes/no*) - Flag indicating whether or not the input mixing table should be printed in problem output. [no]

prtxsec=(*yes/no/1d*) - Flag indicating whether or not input microscopic cross sections should be printed in problem output. The *1d* option indicates that 2-D scattering tables are not to be printed. [no]

timed=(yes/no) - Turns on printing of iteration timing and CPU use data. [no]

det=(*IN*) - Specifies the mixture used to represent a local power range monitor (LPRM) and/or Traversing In-core Probe (TIP) detector located within a fuel lattice. The mixture must also be included in a homogenization block in order to obtain detector cross sections. [has no default]

Important: With the exception of*__prthmmix__*, all output edit options are disabled unless requested by the user. The output edits are disabled by default to minimize the size of the output. The__drawit__option is recommended to generate PostScript plots of the model grid structure and material placement. As previously mentioned, the__echo__and__timed__options are recommended to monitor solution convergence. If the timed option is enabled, each line in the outer iteration edit will be longer than 80 characters. Therefore, it is recommended that Windows users should increase the Command Window size from 80 characters to 132* characters.

Angular quadrature

sn=(*2/4/6/8/10/12/14/16*) - Order of Sn level symmetric quadrature set. [6]

nazim=(*IN*) - Number of equally spaced azimuthal directions in a product quadrature set. Used in tandem with *npolar* keyword (both must be specified). Total number of angles in the product quadrature set is the product of *nazim* and *npolar*. [No default. If not specified, level symmetric quadrature default is used.]

npolar=(*IN*) - Number of polar angles in a product quadrature set (determined using a Gauss-Legendre polynomial). Used in tandem with *nazim* keyword (both must be specified). Total number of angles in the product quadrature set is the product of *nazim* and *npolar*. [No default. If not specified, level symmetric quadrature default is used.]

dgauss=(yes/no) - Enables/disables use of double Gauss-Legendre product quadrature set. If disabled, single Gauss-Legendre product quadrature sets are used. [no]

Important: If both level symmetric quadrature sets and product quadrature sets are requested, the level symmetric quadrature set is to be used. Level symmetric quadrature sets are recommended for general analysis. If reflective boundary conditions are desired for hexagonal-domain configurations, product quadrature sets must be used and nazim must be a multiple of 3. If reflective boundary conditions are desired for triangular-domain configurations, product quadrature sets must be used and*__nazim__*must be an odd number.*

Control options

adjoint=(*yes/no*) - This keyword specifies either a forward (adjoint=no) or adjoint (adjoint=yes) calculation. [no]

forward=(*yes/no*) - This keyword specifies either a forward (forward=yes) or adjoint (forward=no) calculation. If adjoint and forward are both specified, NEWT uses the last specification. [yes]

gpt=(*yes/no*) - This keyword specifies whether this is a GPT adjoint calculation. The *gpt* keyword is active only for adjoint calculations. [no]

Important: The TRITON control module automatically sets the values for forward, adjoint, and gpt keywords; therefore, they can typically be omitted from the Parameter Block. Default values are recommended unless running stand-alone NEWT adjoint calculations.

---

run=(*yes/no*) - A run=no calculation will perform all setup calculations normally performed before beginning iterations and then will stop. It is useful for debugging input and obtaining plots of the input geometry. Run=yes will perform a complete calculation. [yes]

premix=(*yes/no*) - This flag indicates whether the cross section library contains microscopic (premix=no) or macroscopic (premix=yes) cross sections. In essence, it creates a mixing table with a mixture fraction of 1.0 for each mixture on the library. Other mixing tables are ignored. The premixed cross section option is active only for stand-alone NEWT calculations. [no]

kguess=(*RN*) - Initial guess at eigenvalue for an eigenvalue calculation. This parameter may be entered but is not used if a source calculation is performed or a restart file is used to determine the initial guess. [1.0]

restart=(*yes/no*) - If restart=yes is specified, NEWT will open file *restart_newt* and read scalar fluxes and fission rates, enabling a restart from the point at which a previous calculation ended. The file *restart_newt* is always written by NEWT at the end of every successful calculation. The code assumes that all geometry is unchanged from the previous calculation but does allow restart with a different angular quadrature set and P_n scattering coefficients. A low-order solution can be used to accelerate a higher-order solution by restarting using the converged flux of the lower-order solution. [no]

savrest=(*yes/no*) - Determines whether or not a geometry restart file *worf* is written at the end of a calculation. If written, it will overwrite any existing geometry restart file. [yes]

Important: The default values of savrest and kguess are recommended. The TRITON control module automates generation and reuse of the geometry restart file, as well as the initial guess of the eigenvalue. Keywords run, premix, and restart can generally be omitted unless the following conditions are applicable:

- TRITON T-NEWT sequence calculation or stand-alone NEWT calculation with user-supplied restart file, restart=yes.
- Stand-alone NEWT calculation with user-supplied premixed cross section file, premix=yes.
- Interested only in performing setup calculations to debug input and generate geometry plots, run=no, and/or PARM=CHECK in the TRITON sequence input.

---

solntype=(keff/b1/src) - Specifies solution mode type: keff is eigenvalue, b1 is eigenvalue mode followed by a buckling correction, and src is fixed source (no eigenvalue calculation). Fixed source calculations require

additional data for the source specification (see Materials and Source data blocks in Sect. 9.2.3.3 and Sect. 9.2.3.4). [keff]

collapse=(*yes/no*) - If collapse=yes is specified, a flux-weighted collapse is performed by material number; cross sections for each nuclide in each material in the problem are collapsed to a specified (or default) group structure based on the average flux in that material. If collapse=yes, NEWT will look for the *collapse* parameter block; if not found, NEWT will generate cross sections based on the original group structure. If a Homogenization block is present, then collapse is always set to yes. [no]

saveangflx=(*yes/no*) - Option to save angular flux solution. Because the angular flux can require significant file storage. [no]

Important:

Keyword threads should be omitted in favor of the SCALE command line -I option. Keywords solntype, collapse, and saveangflx should be omitted unless the following conditions are applicable.

- For homogenized few-group cross section generation for nodal calculations, solntype **should be b1**. This option will perform a critical spectrum calculation, which will be folded into cross section homogenization calculation. The critical spectrum is also folded into the generation of ADFs and reaction rates for depletion calculations.
- Generation of a new collapsed cross section library, collapse=yes.

Geometry processing options

combine=(*yes/no*) - Automatic grid generation can result in very small grid cells in some locations. Setting parameter combine to *yes* performs automatic combination of smaller grid cells into adjacent neighbor of same material, if possible. Combine is automatically set to *no* if CMFD is enabled; this setting cannot be overridden. [no]

clearint=(*yes/no*) - Grid generation option that removes the global NEWT grid if a local unit grid is supplied. (For meshing options, see the *boundary* keyword in the Geometry block description in Sect. 9.2.3.6) By default, clearint is set to yes, which means the global grid is removed if local grids are provided. If CMFD acceleration is enabled, clearint is set to no, which means both the global grid and optional local grids are used. [yes]

grid_tol=(*RN*) - Tolerance used in determining if polygon vertices are numerically identical during NEWT grid generation. [0.000001]

cell_tol=(*RN*) - Tolerance used in determining if polygon vertices are numerically identical during NEWT cell generation. [0.000001]

line_tol=(*RN*) - Tolerance used in determining if polygon vertices are numerically identical during NEWT line generation. [1.0e-10]

Important: The default values for all geometry-processing keywords are recommended and can be omitted. For problems with very fine mesh, tighter grid and cell tolerances should be applied. For problems that terminate with a ray-tracing error (i.e., tracer error), tighter grid and cell tolerances should be applied.

Critical spectrum options

useb1=(*yes/no*) -Turns on/off the use of the B1 approximation to determine the critical spectrum. If useb1 is set to no, the P1 approximation is used. [yes]

b2=(*RN*) - Material buckling factor, in units of $1/cm^2$. [0.0]

height=(*RN*) - Height (transverse dimension) in centimeters. Used in a geometric buckling correction to calculate leakage normal to the plane of the input 2-D model. Keywords **dz=** and **deltaz=** are equivalent. When set to zero (default), no buckling correction is performed. [0.0]

bf=(*RN*) - Twice the extrapolation distance multiplier used to determine the geometric buckling correction. [1.420892]

Important: If critical spectrum corrections are to be applied, the default values listed above are recommended along with **solntype=b1**. In this option, NEWT will search for the material buckling value such that the homogenized infinite-medium system is critical. NEWT currently uses the B1 approximation as the default. If the P1 approximation is preferred, useb1should be set to no. The infinite-medium B1 (or P1) buckling search is performed in the energy group structure as the original model.

Alternatively, the user can supply the material buckling value using the b2 keyword, and specifying the B1 (default) or P1 approximation (**useb1=no**). In this case, **solntype** should be set to **keff**.

Alternatively, if the user knows the transverse dimension, a geometry buckling factor can be applied, derived from the user-defined **height** and extrapolation distance term **bf** as the following:

$B_g^2 = \left(\frac{\pi}{H+z/\sigma_{tr}}\right)^2$

In this formula, H is keyword height, z is keyword bf, and σ_{tr} is the collapsed, homogenized macroscopic transport cross section.

File unit options

Important: It is highly recommended that the file unit options below be omitted or that default values be used. Alternate file unit values are acceptable for stand-alone NEWT calculations, but changing their values may adversely impact other SCALE modules if NEWT is invoked through a SCALE sequence.

---

hmoglib=(*IN, 0<IN<100*) - This input value specifies the unit number to which a collapsed and homogenized cross section library is written if homogenization instructions are provided (ft*IN*f001). [13]

mixtab=(*IN, 0<IN<100*) - NEWT is able to use a mixing table prepared by SCALE (which may be generated using the T-XSEC sequence). The value of *IN* defines the filename that NEWT will try to locate to read mixing data (i.e., mixtab=92 will cause NEWT to seek the file named ft92f001). This is the default filename produced by the T-XSEC sequence. Alternatively, a mixing table may be specified in NEWT input in the *read mixtable* block; if such a mixing table is supplied, the value of *mixtab* is ignored. [92]

wtdlib=(*IN, 0<IN<100*) - This input value specifies the unit number to which a collapsed cross section library is written if *collapse=yes* is specified (ft*IN*f001). *IN* must be positive and less than 100. [30]

xnlib=(*IN, 0<IN<100*) - This number indicates the filename containing cross sections prepared in a problem-dependent AMPX working library format. The input xnlib=*IN* will cause NEWT to open file ft*IN*f001. This is the only method for providing cross sections as input for NEWT [NEWTHil76].

Examples of input for the parameter block are given below. Note that the two inputs are functionally identical. In the first example, parameters are specified, while in the second example, the input is structured differently and takes advantage of default values.

```
read parm
  solnmode=keff adjoint=no run=yes prtflux=no prtbroad=yes
  mixtab=92 xnlib=4 wtdlib=30 collapse=yes accel=yes sn=6
  outers=100 epsinner=1.0e-4 epsouter=1.0e-4 epseigen=1.0e-5
  kguess=1.34 restart=no prtxsec=yes prtmxsec=yes prtmxtab=yes
end parm

read parm collapse=yes outers=100 epseigen=1.0e-5 kguess=1.34
prtxsec=yes prtmxsec=yes prtbroad=yes end parm
```

9.2.3.3 Material Block

Material block keyword = matl, material, materials

The Material block is always required. Material data must be specified for each mixture used in the calculation. The general format of the Material block is as follows:

```
READ materials
      mix=M pn=N srcid=I com='embedded comment' end
END materials
```

where

M = mixture ID;

N = P_n order for scattering in mixture M (by default, N is 1);

I = Source ID number (the source description for each source ID number is given in the Source block).

Up to 80 characters of text may be entered after *com*=, delimited by single quotes (‘) or double quotes (‘’). A mixture specification is required for each mixture used in the NEWT calculation. The order of the keywords in each specification is unimportant, and only the mix= keyword is required; however, each mixture specification **must** be terminated by the **end** keyword.

A sample Material block is provided below for three different mixtures. Each mixture is specified in a different manner to illustrate different input formats. In this example, P3 scattering is applied in mixture 3, and water and P1 are applied in the other mixtures. The pn= keyword is omitted for mixture 1. The com= keyword is omitted for mixture 2.

```
READ materials
  mix=3  pn=3  com='water' end
  mix=1  com='3.0 enriched fuel' end
  mix=2  pn=1  end
END materials
```

Consider this same set of mixtures but with a fixed source identified by source ID 100 in mixture 1. This specification could be written as follows:

```
READ materials
  mix=3  pn=3  com='water' end
  com='3.0 enriched fuel' mix=1  pn=1 srcid=100 end
  pn=1  mix=2 end
END materials
```

9.2.3.4 Source block

Source block keyword=source, src

The Source block contains source strength specifications associated with a given source ID. The source is assigned to a mixture via the srcid= keyword in the Material block (Sect. 9.2.3.3). Data are input using a keyword-based format:

```
READ source
      id=I typ=T com='embedded comment' src=X end
END source
```

where

I = Source ID number,

T = Source type,

X = List of source strength values, according to type T.

Up to 80 characters of text may be entered after *com*=, delimited by single quotes (‘). The comment string is optional-the remaining parameters are required. Currently, only two source types are supported; the definition of X depends on the source type.

Source type 0 (*typ=0*): A single value of X is supplied-this source strength is placed in all energy groups.

Source type 1 (*typ=1*): *G* values of X are supplied, one value for each energy group. FIDO-type repeat command is supported.

An example of a source specification for two different sources is the following.

```
READ source
      id=1 typ=1 com='44-g fuel source' src=0.44 0.32 0.25 0.01 40r0.0 end
      id=5 typ=0 src=0.001 end
END source
```

The Material block is used to associate a given source definition with a given mixture. The same source may be placed in multiple mixtures. For generalized adjoint calculations-which require a fixed source derived for the generalized response of interest (see *Generalized Perturbation Theory* in the SAMS chapter)-the TRITON control sequence automatically prepares the NEWT Source block.

9.2.3.5 Collapse block

Collapse block keyword = coll, collapse

The Collapse block contains the broad (collapsed) group assignment for each energy group in the original input group structure. Broad group assignments must be contiguous. A FIDO-type repeat factor is allowed. For example, given that a calculation is performed using a 44-energy-group library, in which it is desired to collapse the first 9 groups into a single group, the second 17 groups into a second broad group, and the remaining 18 groups into a third group, either of the following could be used.

```
read collapse
1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
end collapse

read coll 9r1 17r2 18r3 end coll
```

If a collapsing operation is requested, then upon the completion of the transport iteration, NEWT performs a collapsing operation on all cross sections for all mixtures in the problem. Cross sections are flux weighted using the average flux in the mixture in which each nuclide resides and saved in an AMPX working-format library at the unit specified by the *wtdlib*= parameter (default=30). Collapsed, or "broad group," cross sections may also be printed by setting the parameter *prtbroad=yes* (default=no). Note that the energy boundaries of the collapsed cross section are always a subset of the boundaries of the parent library. Cross sections may not be collapsed to arbitrary energy boundaries.

9.2.3.6 Geometry block

Geometry block keyword = geom, geometry

The Geometry block is always required. This data block contains geometric descriptions for all bodies included in the model. NEWT geometry input is performed based on the SCALE Generalized Geometry Package (SGGP) paradigm employed in the KENO-VI Monte Carlo code within SCALE. Those familiar with SGGP as applied in KENO-VI will find the new format very familiar; however, they will quickly realize that the NEWT geometry package contrasts most sharply with the 3-D implementation in KENO-VI because NEWT is a 2-D code. Hence, third dimension (z-axis) specifications are omitted, along with other inherently 3-D bodies supported by KENO-VI. Two other more subtle differences are seen: (1) users must specify the underlying grid structure associated with each *unit*, and (2) curved surfaces (e.g., cylinders) are approximated as N-sided polygons, with user control. Details on these differences are described in the following subsections and illustrated in examples.

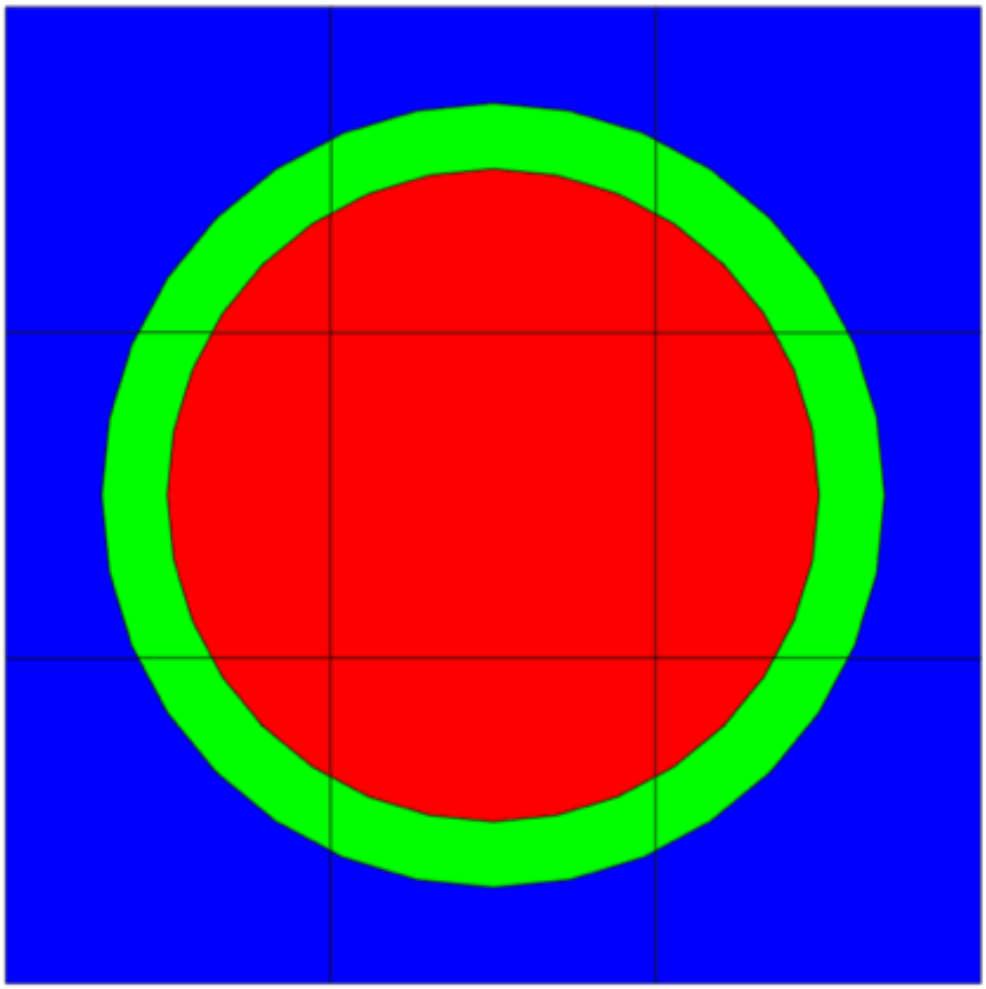
The SGGP approach for model development is combinatorial in nature. Hence, intersections are allowed, and the user is given enormous flexibility to specify, translate, rotate, and combine bodies to create complex configurations. However, the novice user must first focus on the basics of model development, as outlined in this subsection. Sample inputs are provided in Sect. 9.2.4 to demonstrate the development of more complicated models.

Geometric arrangements in NEWT are based on a fundamental building block called a unit. Different units can be arranged in an array. Fig. 9.2.7 illustrates a simple unit and an array of such units. Arrays of units can be contained inside larger units, and in principle, any level of nesting can be achieved. Within a unit, various shapes can be specified, each representing some geometrically distinct medium. In every geometry specification, a single global unit, which forms the outer boundary for the entire problem, must be specified.

Note that in the models pictured in Fig. 9.2.7, bodies are laid within a Cartesian grid. This is a hallmark of any NEWT model-the body specifications combined with an underlying grid structure are used to define a computational grid in which the NEWT ESC solution algorithm is applied. Fig. 9.2.8 illustrates the grid structure associated with the array example above. The model consists of a set of arbitrary polygons used to spatially discretize the bodies of interest. The underlying Cartesian mesh may be specified for any unit; a Cartesian mesh **must** be specified for the global unit. The mesh for the global unit is the primary mesh for the entire problem and is often referred to as the base grid, whereas the mesh for constituent units within the global unit constitutes localized refinement and may be referred to as the local, or unit, grid.

The NEWT geometry block consists of specifications for a set of basic building blocks known as units. A *unit* is defined as a collection of shapes, one of which must be defined as the unit boundary. A complete unit specification consists of a header and three distinct components:

1. **Bodies:** shapes, holes, or array placements that define the bodies within the unit;
2. **Media** specifications that define the material content (composition) of the various shapes; and
3. **Boundary** definition that defines the extent of the unit and its associated grid structure.



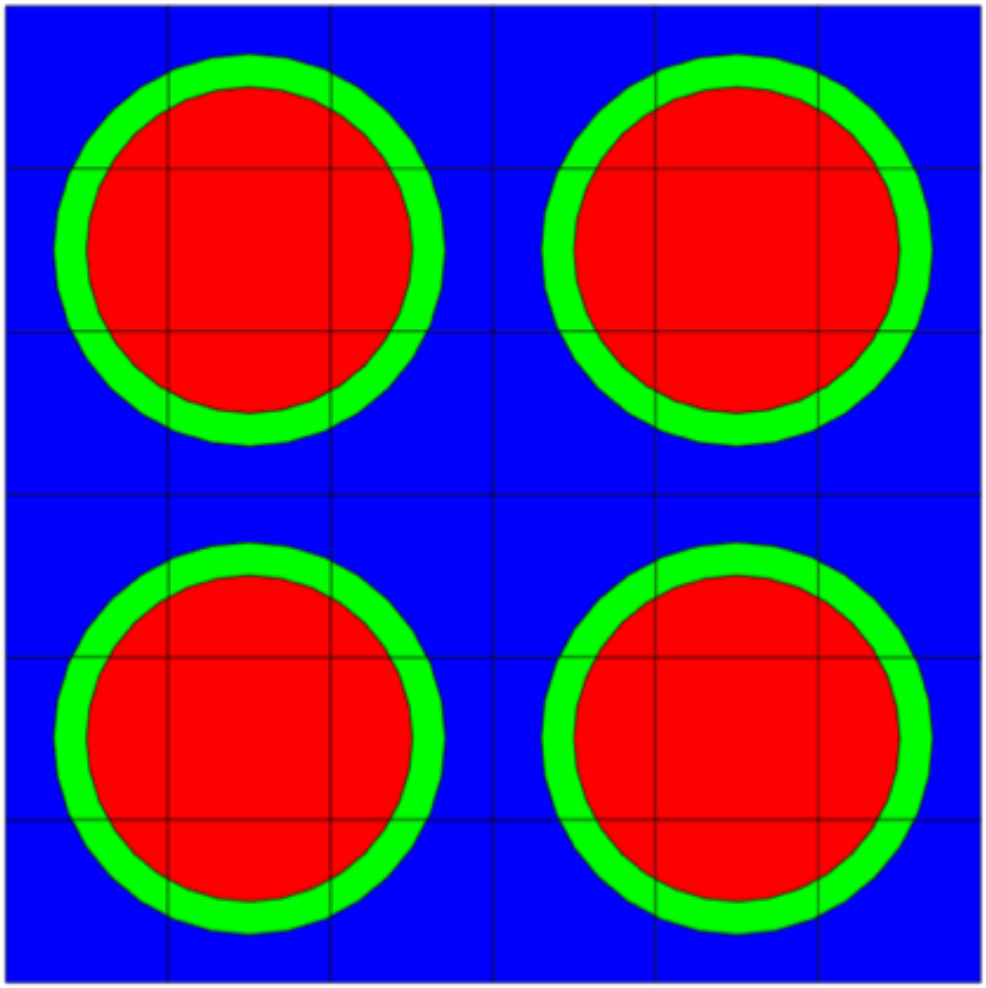


Fig. 9.2.7: A simple unit (top) and an array of units (bottom).

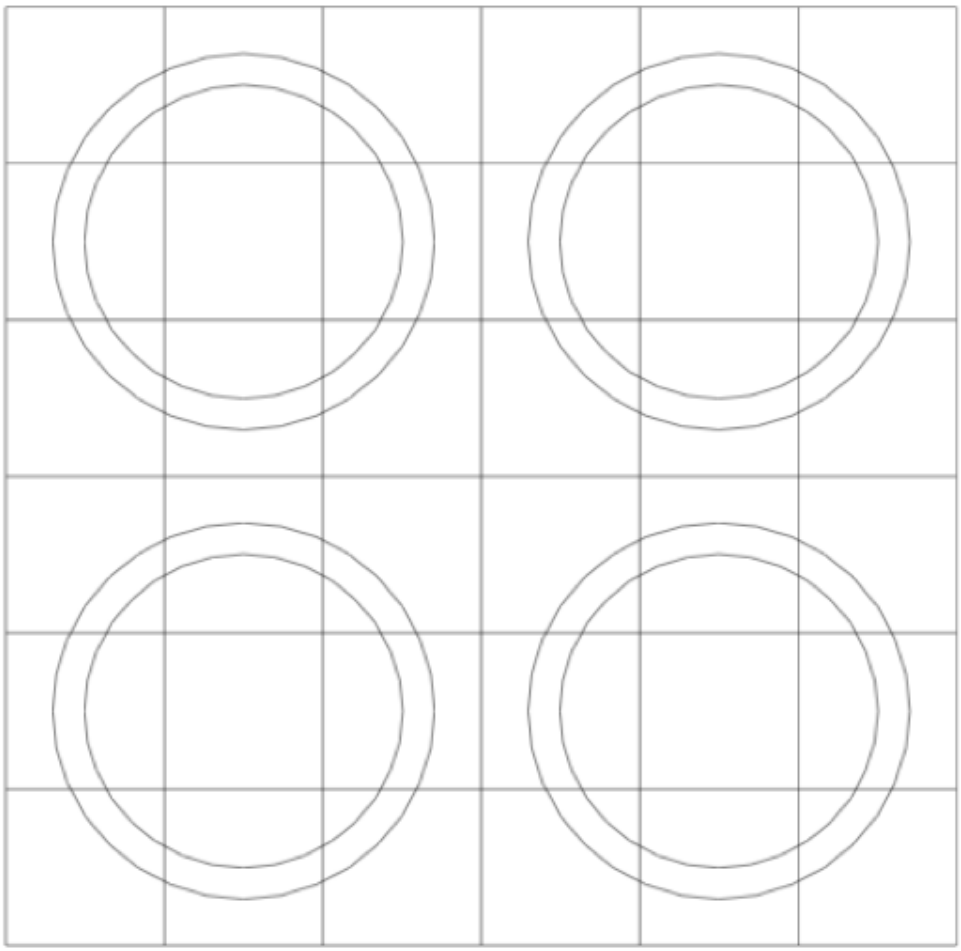


Fig. 9.2.8: Computational grid structure in a NEWT model.

Every unit begins with a header consisting of the keyword *unit* followed by a unique integer label (*unit_id)* that serves to identify the unit:

```
unit unit_id
```

The header is followed by a complete unit description consisting of the three components described above; each of these components of the unit specification is described in the following subsections. In every NEWT model, one unit must be defined as the *global unit*. This unit defines the global coordinate system for the entire problem, and all other units (if any) must fit within the global unit. Specification of the global unit is accomplished simply with the format:

```
global unit unit_id
```

The global unit may occur anywhere in the list of units. If only one unit is defined in an input, it must be identified as the global unit.

As indicated earlier, the geometry block consists of a list of one or more units. Each unit is terminated by the beginning of another unit or by the end of the geometry block. Conceptually, a geometry block will have the following structure:

```
read geom
global unit 1
(unit specifications)
unit 2
(unit specifications)
unit 3
(unit specifications)
...
unit 10
(unit specifications)
end geom
```

The unit numbers are arbitrary and can occur in any order, although they must be unique; they serve simply as labels.

The remainder of this section describes the various components of units.

Bodies

Every unit contains a set of body specifications in terms of (1) basic *shapes* that are placed directly within a unit; (2) one or more arrays, each of which is defined elsewhere and placed within a unit with an *array placement* operator; and (3) holes. Units **must** contain at least one *shape* specification, which is used to define the spatial boundaries of the unit. Additional shape specifications may be used as needed. Holes and/or array placements are optional; there is no theoretical limit on the number of each that may be used within a unit.

Shapes

Shapes are simple predefined bodies. NEWT currently supports six shapes:

1. cylinder,
2. cuboid,
3. hexprism,
4. rhexprism (rotated hexprism),
5. wedge, and
6. polygon.

The names of these shapes are generally associated with 3-D bodies but are used in NEWT to be consistent with KENO-VI nomenclature. In NEWT, a cylinder is equivalent to a circle, a cuboid is equivalent to a rectangle, a hexprism is equivalent to a hexagon, and a wedge is equivalent to a triangle.

Because the SGGP is combinatorial in nature, intersection and overlap of shapes is permitted. For this reason, no specific mixture is associated with each shape. Combinatorial logic allows a fraction of a shape to be filled with one mixture, while the remainder or another fraction thereof may be assigned a different mixture. This is discussed further in the section **Media Specifications** in the description of *media* assignment (Sect. 9.2.3.6.17).

Each shape is specified by name, an associated body identification (*body_id*) number, and dimensioning data. The *body_id* number is arbitrary but must be unique within each unit. Specific formats for each shape are provided below.

Cylinder

Cylinder

The cylinder specification has the following format:

```
cylinder  body_id  radius  [modifier_list]
```

where radius is the radius of the circle. The circle will be centered at (0,0). The modifier list is an optional set of operations that may be performed on each shape. One of the modifiers allowed is the *origin* modifier, which lets one translate the origin of a shape to a different location. Modifier commands are described later in this section.

Cuboid

The cuboid specification has the following format:

```
cuboid  body_id  xmax, xmin, ymax, ymin  [modifier_list]
```

where (x_{min}, y_{min}) and (x_{max}, y_{max}) represent the lower-left and upper-right vertices of a rectangle on a Cartesian coordinate system. Note that the cuboid is explicitly placed by its coordinates; no translation is required (or allowed).

Hexprism and rhexprism

Both hexprisms are specified in a manner identical to that of a cylinder:

```
hexprism body_id  radius  [modifier_list]
rhexprism  body_id  radius  [modifier_list]
```

where *radius* is the inner/minor radius of the hexagon. A standard hexagon (*hexprism)* is oriented with vertices at the top and bottom, as illustrated in Fig. 9.2.9. A rotated hexagon (*rhexprism*) is oriented with vertices on the left and right sides, as illustrated in Fig. 9.2.10. Both types of hexprisms, like cylinders, are by default placed with their origins at (0,0). However, like cylinders, they can also be translated in space via the *origin* translation command.

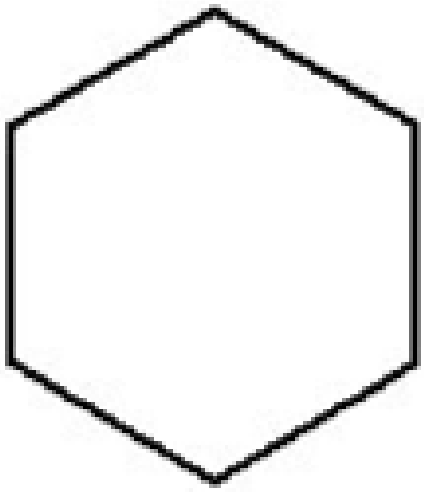


Fig. 9.2.9: Orientation of a standard hexprism.

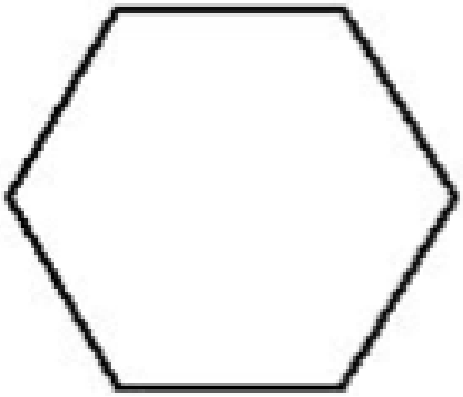


Fig. 9.2.10: Orientation of a rotated hexprism.

Wedge

A wedge, or triangle, specification has the following format:

```
wedge  body_id  xbase xpt ypt [modifier_list]
```

where the vertices of the shape are defined as (0,0), (x_{base},0), and (x_{pt},y_{pt}). Thus, one side always lies on the x-axis. The modifiers *origin* and *rotate* may be used to position and orient the triangle in the problem domain. Fig. 9.2.11 illustrates placement of a wedge using these parameters.

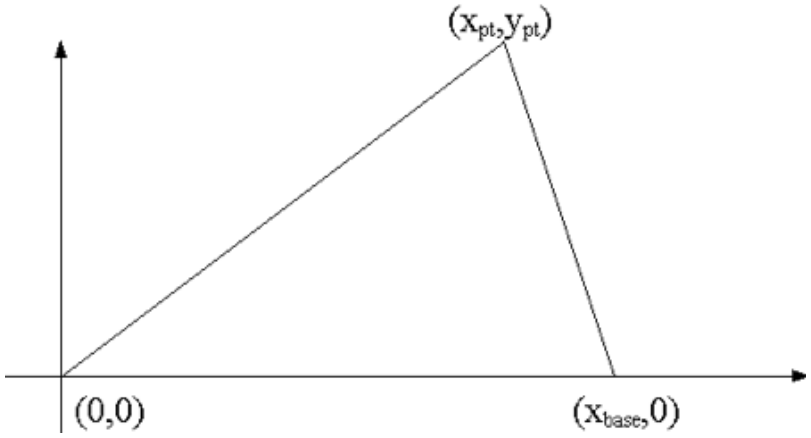


Fig. 9.2.11: Initial positioning of the wedge body.

Polygon

The polygon specification has the following format:

```
polygon  body_id  x0, y0, x1, y1, ..., xN, yN, x0, y0
```

where (x_i, y_i) are the polygon vertices (the first and last pair in this list refer to the same vertex). Note that the polygon is explicitly placed by its coordinates; no translation is required (or allowed).

Example of shape specifications

Use of shapes within a unit can be illustrated with a simple example. Consider a unit, arbitrarily labeled with *unit_id*=10, containing a cuboid and two cylinders. Each shape is given a unique (but arbitrary) *body_id*.

```
unit 10
cuboid   11   3.0   -5.0   1.0   -2.0
cylinder 12   0.8
cylinder 13   0.6
```

The cuboid is explicitly placed by its coordinates; the two cylinders will by default be placed at (0,0). Use of the *origin* command to relocate cylinders is introduced below. Fig. 9.2.12 illustrates the body placement that occurs for the given example.

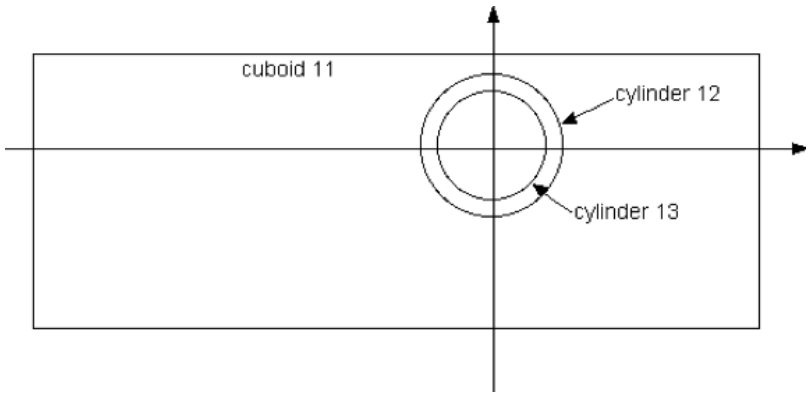


Fig. 9.2.12: Body placement for two cylinders and a cuboid.

Shape modifier commands

Modifier commands are provided as a means to perform specific functions relative to the input shape. The five available modifier commands and the shapes to which they may be applied are listed here:

origin - translation of center bodies (cylinders, hexprisms, rhexprisms);

rotate - rotation of bodies with respect to the transverse axis (all);

chord - removal of a portion of a body with x-plane and y-plane cuts (cylinders, hexprisms, rhexprisms);

com - addition of a comment to a specific shape (all);

sides - number of sides used to approximate a circle (cylinder, default is 12).

The format for each of these commands follows.

ORIGIN

The *origin* modifier is used to translate the origin of a cylinder or hexprism from the default origin of (0,0) to some other location. It has the format

```
origin x=xnew y=ynew
```

where (x_{new}, y_{new}) is the new center of the shape. The modifier *origin* may not be applied to cuboids, as the location of a cuboid is explicitly set by its shape specification. If not specified, each ordinate is set to zero, such that

```
origin x=5
```

is equivalent to

```
origin x=5 y=0 .
```

For example, consider a cuboid whose lower-left and upper-right corners are located at (0,0) and (1,1), respectively. If one places a cylinder with radius 0.3 in the center of this box [i.e., centered at (0.5, 0.5)], this would be specified as follows:

```
unit 1
cuboid 10 1.0 0.0 1.0 0.0
cylinder 20 0.3 origin x=0.5 y=0.5
```

This configuration is illustrated in Fig. 9.2.13.

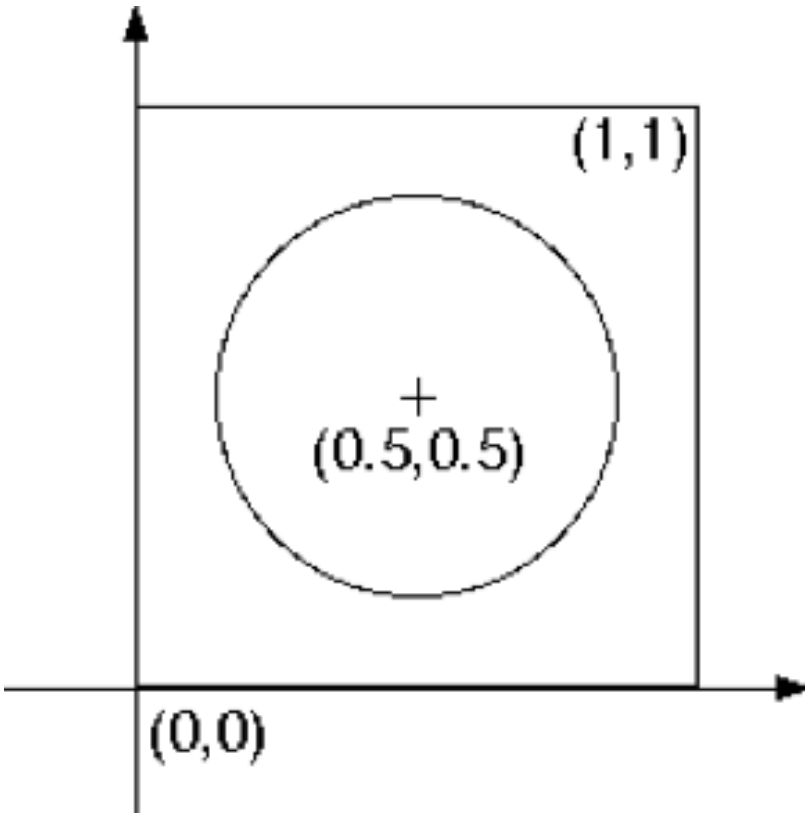


Fig. 9.2.13: Relocation (translation) of a cylinder using origin.

Rotate

The *rotate* body modifier is used to rotate a body around its geometric center. It can be applied to any body type, but rotation of a cylinder has no real meaning. The format of the *rotate* modifier is as follows:

```
rotate a1=A
```

where *A* is the angle of rotation, in degrees, in a counterclockwise direction. All bodies lie in the (x,y) plane, with rotation around the z-axis, but with respect to the centroid of the body. Rotation always occurs **before** translation (via *origin*), irrespective of the order of *rotate* and *origin* commands in the modifier list for a body. KENO-VI allows rotation about the x and y axes as well, through a2= and a3=; however, only rotation about the z axis is permitted in NEWT.

As an example, consider a 1 by 1 cm cuboid centered at (0,0), place a smaller 0.5 by 0.5 cm cuboid inside it, and rotate it 30 degrees clockwise.

```
unit 1
cuboid 10 0.5 -0.5  0.5 -0.5
cuboid 10 0.25 -0.25  0.25 -0.25 rotate a1=-30
```

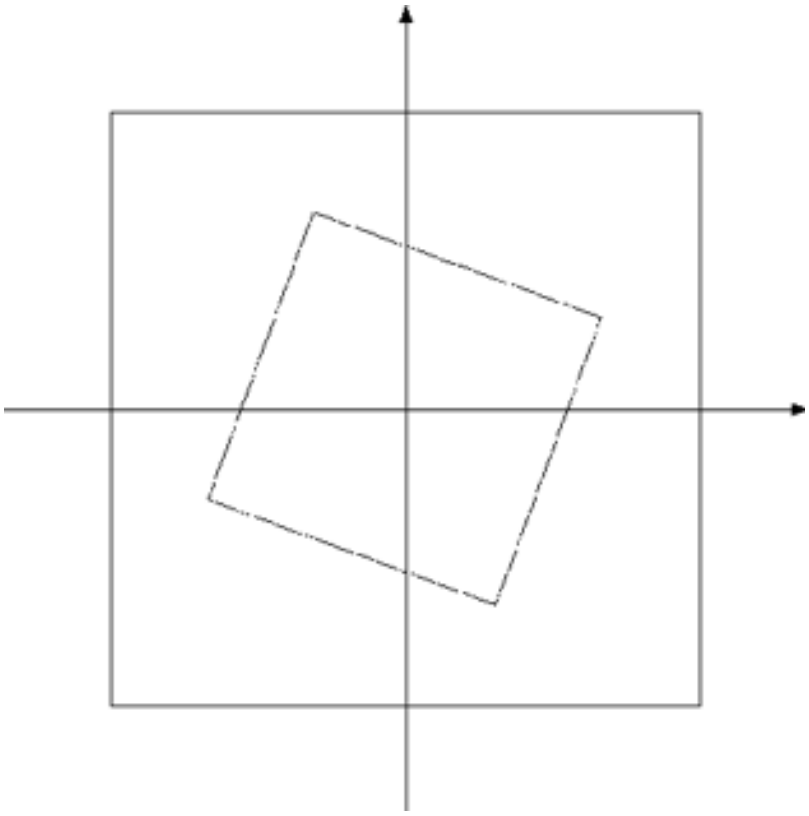


Fig. 9.2.14: Rotation of a cuboid.

Fig. 9.2.14 illustrates the configuration generated using this specification. Note that clockwise rotation was performed by specifying a negative angle. In this case, the centroid of the inner cuboid coincides with (0,0); however, the same geometric rotation would have occurred if the bodies had not been centered at (0,0).

Chord

The *chord* modifier is used to remove a portion of a body. It provides for horizontal and/or vertical cuts on a body, with the portion of the body on a specified side of that body discarded. Chords may be applied to cylinders or hexprisms but may not be applied to cuboids. (Such "cuts" may be explicitly defined in the body definition.) The format of the specification combines the selection of the plane (horizontal or vertical cut), location of the plane, and the portion of the body to be retained, all in one terse modifier.

The four possible chord specifications for a body are as follows:

```
chord +x=xplane
chord +y=yplane
chord -x=xplane
chord -y=yplane
```

where *xplane* and *yplane* are the ordinates on the x and y axes, respectively. The sign on x and y indicates the portion of the body to be retained after the cut. A plus (+) sign indicates that the portion of the body in the positive (increasing x or y) direction should be kept, and a minus (-) sign indicates that the portion of the body in the negative direction of the cut plane (decreasing x or y) direction is retained. Chords are applied **after** any translation (*origin*) or rotation (*rotate*) modifier operations. Multiple chords may be specified for a single body to obtain multiple cuts. The keyword *chord* **must** precede each specification.

The use of chords is best illustrated by example. Fig. 9.2.15 through Fig. 9.2.18 show unit body descriptions with various chord specifications.

```
unit 10
cuboid 1 0.5 -0.5 0.5 -0.5
cylinder 2 0.25 chord +x=0.0
```

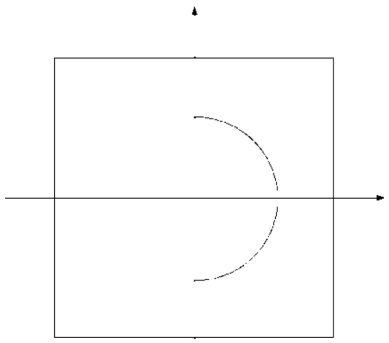


Fig. 9.2.15: Example of chord +x behavior.

```
unit 10
cuboid 1 0.5 -0.5 0.5 -0.5
cylinder 2 0.25 chord -y=0.0
```

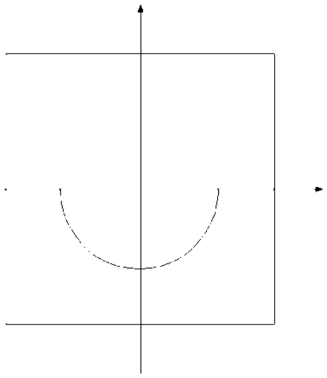


Fig. 9.2.16: Example of chord -y behavior.

```
unit 10
cuboid 1 0.5 -0.5 0.5 -0.5
cylinder 2 0.25
  origin x=0.5  y=-0.5
  chord -x=0.5 chord +y=-0.5
```

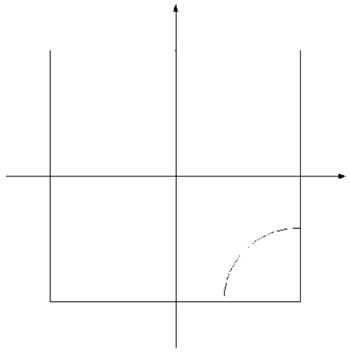


Fig. 9.2.17: Use of two chords to create a quarter-cylinder body.

```
unit 10
cuboid 1   0.5 -0.5 0.5 -0.5
cylinder 2 0.5
  chord -x=0.25 chord +x=-0.25
  chord -y=0.25 chord +y=-0.25
```

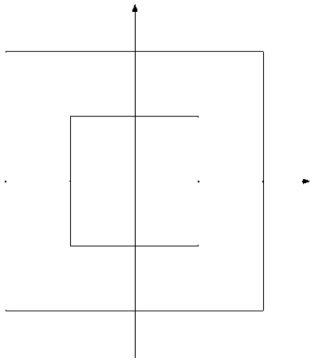


Fig. 9.2.18: Use of four chords on a cylinder to create a square body.

In Fig. 9.2.15, a chord is placed at x=0 and the positive portion relative to the chord (x >0) is retained because +x mode is specified. Since the cylinder is centered at (0, 0), this chord cuts the cylinder in half and retains the right half of the cylinder. The unit in Fig. 9.2.16 uses the same cylinder but with a chord cutting at the plane located at y=0. The bottom half (y < 0) is kept because -y is specified.

Fig. 9.2.17 is somewhat more complicated but represents perhaps the most common use of chords in lattice models. In this case, it is desired to create a one-quarter cylinder located in the bottom right quadrant of a cuboid. A 1 by 1 cm square cuboid is centered at (0, 0), and a cylinder is placed at (0.5, -0.5), which is the lower right-hand corner of the cuboid. Since we are interested only in the portion of the cylinder within the cuboid, we choose to keep the top (+x) and left (-y) portions of the cylinder. This requires two separate chord modifiers. (Each chord specifies only one cutting plane.) Additionally, because the cylinder was relocated to a new origin, the chords are specified such that the cuts go through the new origin.

Note that there is no requirement that a chord cut through the origin of a body. Fig. 9.2.18 illustrates the use of four chords to set four cutting planes. A 0.5 cm cylinder is specified centered within the unit cuboid. All four of the four permitted cutting planes are specified. We have effectively created a cuboid by retaining the portion of the cylinder above (+y) the xz plane located at y= -0.25, below (-y) the plane at y=+0.25, to the right (+x) of the yz plane at x= -0.25, and to the left (-x) of the plane located at x=+0.25. There is, of course, a much more direct means to create a cuboid-this example is provided only for illustrative purposes.

For guidance on how to cut a cylinder at an oblique angle, refer to Sect. 9.2.3.6.15.

Com

The *com* modifier is a means to label specific bodies. It is provided primarily for consistency with KENO-VI. At this time, NEWT simply reads and then ignores *com* data. It can, however, be used as a means to help annotate an input listing. The format for the *com* modifier is as follows:

```
com="comment string"
```

where *"comment string"* is any text description of up to 132 characters, delimited by single (‘) or double quotes (“). For example, the input description of Fig. 9.2.18 is shown here with comments added via *com* modifiers.

```
unit 10
cuboid 1   0.5 -0.5 0.5 -0.5 com="unit cuboid centered at (0,0)"
cylinder 2 0.5 com="cylinder with four chords"
  chord -x=0.25 chord +x=-0.25
  chord -y=0.25 chord +y=-0.25
```

Sides

The *sides* modifier applies only to cylinders and is unique to NEWT (i.e., it is not used in KENO-VI). Because NEWT’s solution grid is based on arbitrary polygons, all cells must be straight sided. Hence, the curved surfaces of a cylinder are approximated as an N-sided regular polygon. By default, N=12. The *sides* operator allows the user to override the default. The format is very simple:

```
sides=N
```

where N is the number of sides desired for the full cylinder. In general, a 12-sided polygon provides an adequate approximation of a cylinder. Use of additional sides will create a cylinder that has a smoother appearance and increase the computational effort required to solve the cells associated with the cylinder.

Fig. 9.2.19 shows a model built with three nested cylinders inside a unit cuboid. Cylinder 10 is the innermost cylinder, with no *sides* modifier; hence, it uses the default 12-sided approximation. The second cylinder is specified with sides=16; the refinement in this approximation is seen in the figure. Finally, cylinder 30 is specified with 40 sides-this is visually a very close approximation to a cylinder.

```
unit 10
cuboid 1   0.5 -0.5 0.5 -0.5
cylinder 10 0.2
cylinder 20 0.3 sides=16
cylinder 30 0.4 sides=40
```

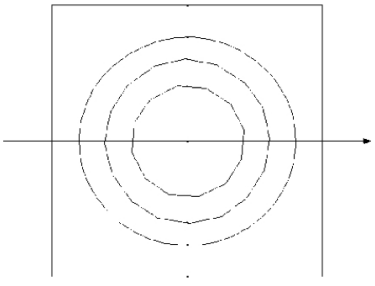


Fig. 9.2.19: Use of the sides modifier for cylinders.

Holes

The next level of complexity within a *unit* is provided through the use of a *hole* specification. The *hole* specification is simply a means by which one unit may be placed within another unit. In some instances, a well-defined set of structures, assembled as a *unit*, needs to be placed within a larger unit. NEWT provides two methods to do this-holes and arrays. Arrays are used to place a unit (or a number of similar units) in a regular repeating pattern within an enclosing unit. A *hole*, on the other hand, is a means to place a single unit. This is often used when units being placed do not have a regular repeating pattern.

The format for a *hole* specification within a unit is as follows:

```
hole unit_id [modifier_list]
```

where *unit_id* is the identification number for the unit that is being placed within the current unit. (A unit cannot be placed within itself.) Unlike the shapes described earlier, holes do not have a distinct identification number of their own-they are simply a mechanism to place a unit defined elsewhere.

By default, the *hole* operator places the origin of the new unit at the origin (0,0) of the current unit. The *origin* modifier may also be used with a *hole* specification to position the placed unit at a location other than (0,0) of the current unit. However, placement of the body is **always** relative to the origin of the original unit, which can be defined in a number of different ways.

Holes are also associated with a particular shape. Hole specifications must immediately follow the shape into which they are being placed. Holes redefine the boundaries of a shape by figuratively cutting holes in that shape into which units are placed. When mixtures are defined for a given shape (through media specifications, described below), the mixture is placed throughout the region, except in the space excluded by the hole placements.

The *rotate* modifier can also be applied to a hole, as can the *com* modifier. However, *chord* specifications cannot be used to remove a portion of a hole. To construct a cylinder that is cut at an oblique angle, users should construct a cylinder that is cut by a chord and then use the *hole* operator combined with the *origin* and *rotate* modifiers to place and rotate the unit to the desired position and orientation. This can be particularly useful in hexagonal or triangular geometries.

As an example, consider a unit, unit_id=1, consisting of two concentric cylinders, and a second unit, consisting of two concentric cuboids. Descriptions for these two units are given below. Note that these are incomplete unit specifications; other components of the unit specification have not yet been introduced. However, for the purposes of this example, incomplete unit specification will suffice.

```
unit 1
cylinder  12   0.8
cylinder  13   0.6

unit 2
cuboid 12 0.8 -0.8 0.8 -0.8
cuboid 13 0.6 -0.6 0.6 -0.6
```

Now suppose that we wished to place two of the unit 1 cells and one of the unit 2 cells inside unit 3, with unit 2 rotated by 45°. We can define a cuboid as unit 3 and place the units 1 and 2 inside the cuboid using *hole* specifications:

```
unit 3
cuboid 10 4.5 0.0 4.5 0.0
hole 1 origin x=1.3 y=1.3
hole 1 origin x=1.3 y=3.3
hole 2 origin x=3.1 y=2.3 rotate a1=45
```

In this example, a square cuboid is defined such that its lower-left corner is situated at (0,0). Three hole operations are used: the first to place unit 1 at (1.3, 1.3), the second to place another instance of unit 1 2 cm above the first, at (1.3, 3.3). Lastly, unit 2 is placed inside unit 3 at (3.1, 2.3) and then rotated 45°. Fig. 9.2.20 illustrates how such a unit would appear.

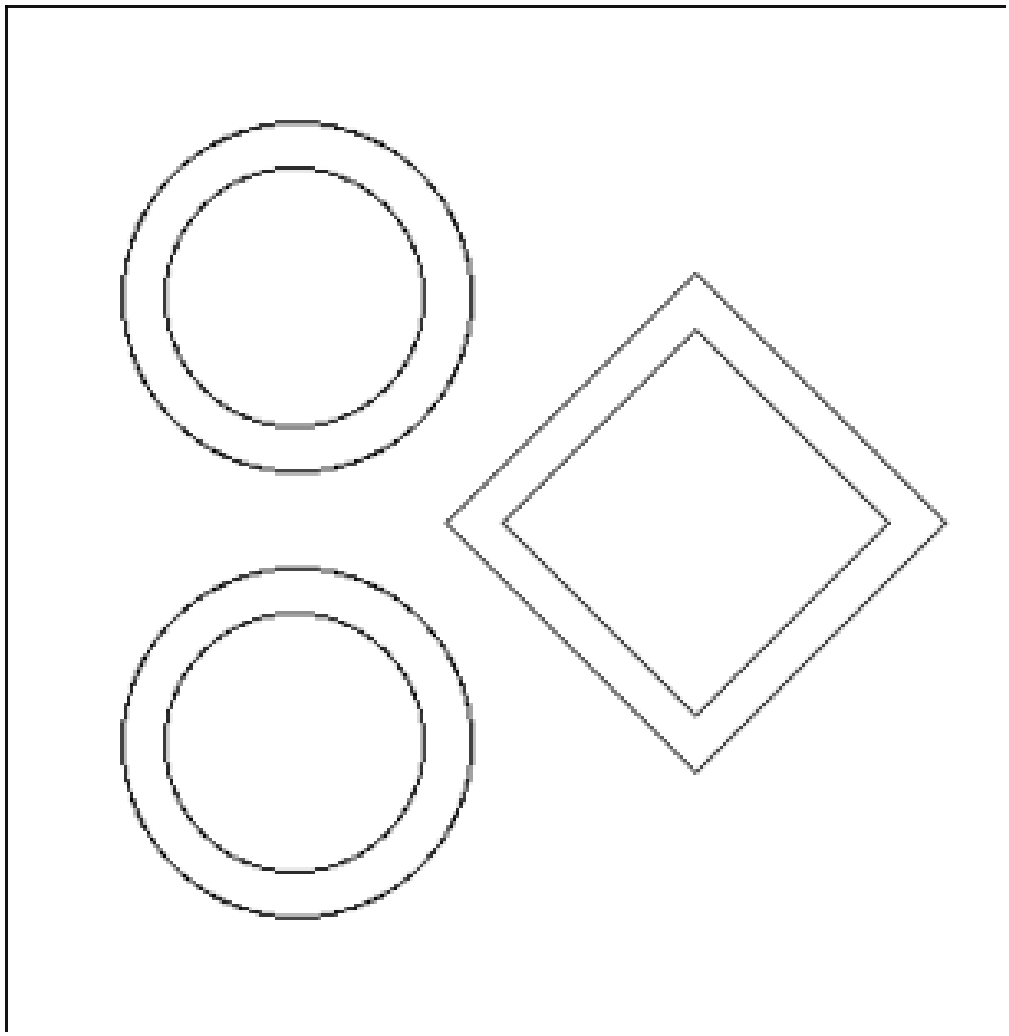


Fig. 9.2.20: Unit placement within a unit using holes.

Array placement

As indicated in the previous section, arrays are a method for arranging one or more units within another unit. Arrays specifications are typically used when units are placed in a repeating pattern. While the *hole* specification is used to place different units within a given unit, the array placement specification is used to place an *array* within a unit.

All arrays are specified (declaration of size, type, and fill) in the array data block, as described in Sect. 9.2.3.9. The array placement operator is used to locate an array within a unit. The format for the array placement operator is as follows:

```
array arrayid body_id place i j xij yij
```

where *arrayid* is the identification number assigned to the array in the array data block and *body_id* is the identification number of the *shape* into which the array is placed. The remainder of the array placement operator is used to fix the position of the array within the body, identified by *body_id*. A general discussion of this concept follows, after which the actual placement of the array is described.

Arrays are defined by two dimensioning parameters-the number of rows and the number of columns. Each element of an array is filled by a unit; each unit has its own local coordinate system. In other words, one unit may have the origin (0,0) in its local coordinate system defined as the lower-left corner while another unit may have its origin defined at its geometric center. The array itself has no coordinate system; it is simply a list of relative positions of units, defined by their row/column position. The *place* directive of the array placement operator is used to locate the array within the body into which it is being placed.

In the *place* directive, *i* represents the column (counting from left to right) and *j* represents the row (counting from bottom to top) of a specific element of the array. The coordinate system of that specific unit is used to set the position of the entire array. The coordinates *xij* and *yij* represent the location in the current unit where the array is to be placed. Placement occurs by situating the origin of the unit in column i, row j at coordinate (x:sub:*ij*,y_{ij}). Placement of arrays within a unit is best understood through examples. Consider three (partial) unit specifications, as defined and illustrated in Fig. 9.2.21.

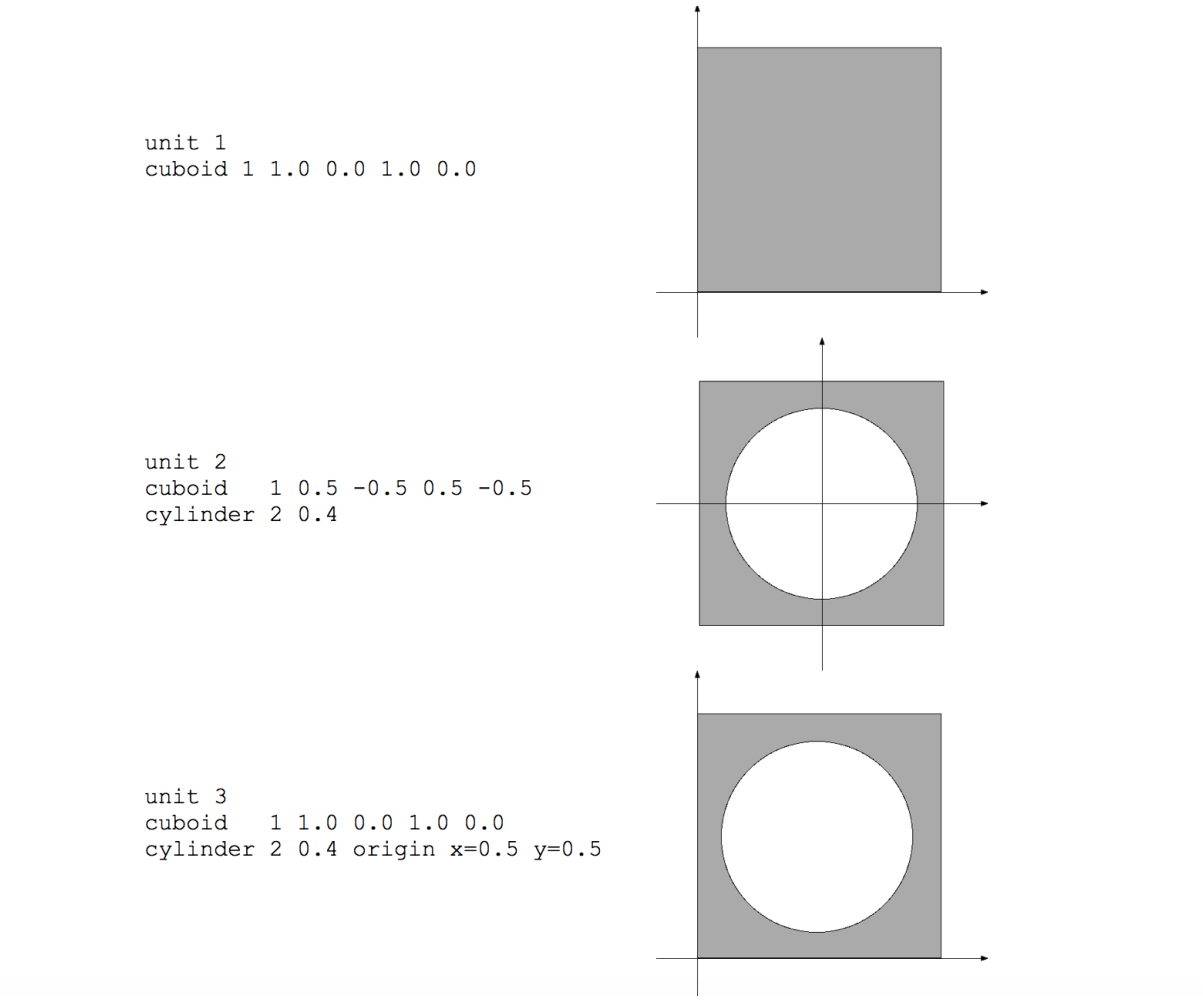


Fig. 9.2.21: Example of units to be placed in an array.

Unit 1 is a simple 1 by 1 cm cuboid with its origin located at its bottom-left corner; unit 2 is a similar-sized cuboid but with its origin located at its geometric center and with a cylinder centered in it; and unit 3 is identical to unit 1 but with a cylinder centered in it. Units 2 and 3 are identical in structure but use a different local coordinate system.

Now assume an array has been defined in the array data block and assigned *arrayid*=50. The relative positions of the units are shown in Fig. 9.2.22; unit 3 is located in row 1, column 1.

We desire to place this array within a unit 4, a 2 by 2 cm cuboid with its lower-left corner located at (0,0). Because there are four different array positions, this array has four possible placements:

```
unit 4
cuboid 10 2.0 0.0 2.0 0.0
array 50 10 place 1 1 0.0 0.0

unit 4
cuboid 10 2.0 0.0 2.0 0.0
array 50 10 place 2 1 1.5 0.5

unit 4
cuboid 10 2.0 0.0 2.0 0.0
array 50 10 place 1 2 0.5 1.5

unit 4
cuboid 10 2.0 0.0 2.0 0.0
array 50 10 place 2 2 1.0 1.0
```

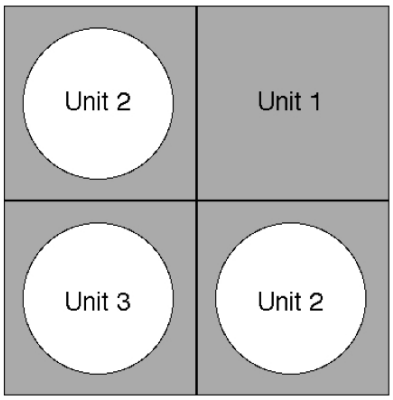


Fig. 9.2.22: Layout of units in array 50.

For the first example, the unit located in row 1, column 1 (i.e., unit 3) is placed such that its origin (its lower-left corner) is located at (0,0), which is the origin of unit 4. For the second example, the unit located in column 2, row 1, is placed such that its local origin, which is in the center of the unit, is located at x=1.5, y=0.5 in the coordinate system of unit 4.

Media specifications

A unit is only partially specified by its constituent bodies. At this point, no composition has been associated with the various regions of the problem nor has the outer boundary of the unit been defined. This section provides information on the use of *media* specifications to define the contents of each shape that has been defined.

Each shape statement defines a basic shape, with optional modifiers, which represents a spatial region within the unit. Assignment of compositions to regions is performed via *media* specifications.

As discussed earlier in the introduction to shapes, input processing in the SGGP is combinatorial. This permits intersection of shapes, and different compositions (or media) may be assigned to different portions of intersecting bodies.

The format of a media specification is as follows:

```
media  materialid  bias_placeholder  reg_def_vector
```

where *materialid* is the composition number being placed in this entry, *bias_placeholder* is a simple placeholder that is required but not used, and *reg_def_vector* is the region definition vector used to define the shape or shapes to which the mixture is assigned.

The *bias_placeholder* is used to be as consistent as possible with KENO-VI input. KENO-VI allows the user to assign biases within the media assignments to improve the Monte Carlo solution performance. Biases have no meaning in NEWT, so the field has no meaning. In KENO-VI, if no special biasing is desired, a value of 1 is assigned. If it is desired to move models between NEWT and KENO-VI format, a placeholder value of 1 is recommended. However, the value itself has no meaning within NEWT; it is simply read and ignored. (This may change in a future release.)

The region definition vector is used to describe the location of the composition within the current unit. This is done by providing a list of shapes for which the media is either "inside" or "outside." The sense of the media with respect to a shape is specified by listing the shape number with a negative sign if "outside" and with a positive (or no) sign when the media is placed "inside" the shape.

Consider a simple cylinder inside a cuboid. Assume composition 1 is to be placed inside the cylinder and composition 2 outside the cylinder but inside the cuboid. The shape and media specifications could have the following format:

```
unit 1
cylinder 10 0.5
cuboid 20 1.0 -1.0 1.0 -1.0
media 1 1 10
media 2 1 20 -10
```

In the above example, the first media record places mixture 1 inside all of shape 10 (the cylinder). The second media entry places mixture 2 in all regions that are outside shape 10 but inside shape 20. Note also that a bias placeholder value of 1 is used in each media statement.

It is necessary to specify media for all regions of the unit. If any regions remain unassigned, NEWT will stop with an error message. If the second record had been omitted, regions outside the cylinder would be unspecified and the code would stop. Note also that if the second media statement had read only

```
media 2 1 20
```

then composition 2 would have been placed inside all of cuboid 20, including inside the cylinder 10. The fact that the contents of 10 have already been specified is ignored. The above statement directs the code to put mixture 2 everywhere inside the boundaries of the cuboid.

Each region definition vector combines all specifications with a logical AND. In other words, the second media record in "`media 2 1 20`" places composition 2 in all regions that are simultaneously outside shape 10 **and** inside shape 20. Separate media specifications are required to place a composition in two independent shapes. The following represents a cuboid with two nonintersecting cylinders.

```
unit 1
cylinder 10 0.4 origin x=0.5
cylinder 20 0.4 origin x=1.5
cuboid 30 2.0 0.0 1.0 -1.0
media 1 1 10
media 1 1 20
media 2 1 30 -10 -20
```

A media statement is necessary to place composition 1 inside shape 10; a similar statement is necessary to place composition 1 inside shape 20. Finally, all space inside cuboid 30 but outside both 10 and 20 is filled with composition 2. If one attempted to fill both 10 and 20 with composition 1 in a single media record, for example,

```
media 1 1 10 20
```

then an error would occur. The code would attempt to place composition 1 in all space that is simultaneously within shape 10 **and** within shape 20-and no such space exists.

A more common example is the configuration of a fuel pin (composition 1), gas gap (composition 2), clad (composition 3), and moderator (composition 4) in a lattice. Consider a pin in a hexagonal lattice:

```
unit 1
cylinder 10 0.4
cylinder 20 0.41
cylinder 30 0.5
hexprism 40 0.8
media 1 1 10
media 2 1 20 -10
media 3 1 30 -20
media 4 1 40 -30
```

In this example, for the hexagonal moderator region outside the clad, it is sufficient to specify that mixture 4 is inside 40 **and** outside 30. Although it is true that the moderator is also outside shapes 10 and 20, it is not necessary to specify this. Logically, since 10 and 20 are inside 30, then everything outside 30 must be outside 10 and outside 20. The use of a hexprism in this example is irrelevant. If the outer body had been a cuboid, the result would have been the same.

As a final example, consider a unit with intersecting bodies. It becomes possible to assign a unique composition to each intersection of shapes (Fig. 9.2.23).

```
unit 1
cylinder 10 9.0 origin x=4.5
cylinder 20 9.0 origin x=-4.5 y=-4.5
cylinder 30 9.0 origin x=-4.5 y=4.5
cuboid 40 15.0 -15.0 15.0 -15.0
media 1 0 10 -20 -30
media 2 0 20 -10 -30
media 3 0 30 -10 -20
media 4 0 10 20 30
media 5 0 10 20 -30
media 6 0 20 30 -10
media 7 0 10 30 -20
media 8 0 40 -10 -20 -30
```

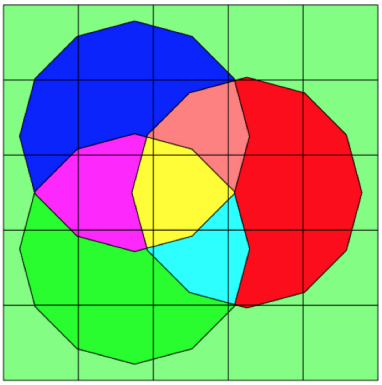


Fig. 9.2.23: Media assignments in overlapping regions.

In this model, cylinder 10 is on the right, 20 is the lower left, and 30 is the upper left. Media 1, placed inside cylinder 10 but outside 20 and 30, is represented by the partially filled right-hand side of the right cylinder. The central region of the unit is filled with composition 4 and represents all areas that are simultaneously within shapes 10, 20, and 30. The outermost region is everything that is inside 40, but outside 10, 20, and 30.

Media statements apply only to shapes, and only to those shapes within the unit. Holes and array specifications are used to define placement of one or more units in which media have already been specified in the corresponding unit definitions. Like shape statements, media statements may occur in any order. However, if one region is erroneously assigned two different compositions in two different media statements, the code will allow this and will proceed with the calculation. The last specification for a shape will always take precedence. Thus, it is important that newly developed models be visually inspected using mixture plots (files named "*.newtmatl.ps") created using *drawit=yes* in the parameter block.

Unit boundary

The final section of a unit description is the *boundary* specification. This input record serves two purposes: to specify the shape that defines the outer bounds of the unit, and hence the shape of the unit, and (optionally) to specify the underlying grid associated with the unit. The format of the boundary specification is as follows:

```
boundary  body_id  [x-discretization y-discretization]
```

where *body_id* is the identification number for the body that is to serve as the unit boundary. The *x-discretization* and *y-discretization* terms are integers (≥2) that specify the number of rectangular cells to be placed in the unit in the x-direction and y-direction, respectively. A grid specification is **required** for the global unit but is optional for other units. If a grid is specified for a grid other than the global unit, that grid replaces the base grid. (An exception to this principle is discussed later.)

In general, grid refinement should be such that cell sizes are on the order of or smaller than a mean free path for a neutron. Grid spacing can be easily varied in order to converge on the parameter of interest. Global factors, such as a system eigenvalue, can tolerate a relatively coarse grid. However, if fluxes are known to vary rapidly in space, then a more refined grid may be necessary. NEWT does provide the ability to locally refine a grid structure so that detail can be modeled where needed, without having to pay the computational penalty of refining the grid everywhere. NEWT does place one limit on grid refinement: every shape, hole, or array placed within a unit must be intersected by at least one gridline. The grid may be locally defined or part of the global grid, but it must intersect each body at least once. Thus, if small geometric shapes are modeled, a detailed grid structure is generally necessary.

Examples of boundary specifications follow, as parts of partial unit specifications. Media descriptions are omitted for simplicity. Accompanying figures illustrate the grid structure(s) associated with each specification. Fig. 9.2.24 shows a single (global) unit with a 2 by 2 base grid. Cuboid 10 serves as the boundary for the unit. This represents the minimum grid structure that can be specified for a unit. Fig. 9.2.25 shows a more complex configuration in which a unit defined with a 5 by 5 grid is placed in the center of a larger enclosing unit, specified to have a 3 by 3 grid. Note that because the first unit has its own (local) grid, the underlying grid structure is removed in favor of the local grid structure. The grid is applied to the boundary shape of the unit, which is cuboid 10.

Fig. 9.2.26 shows a similar structure; however, the cuboid was removed from unit 1 and the outer hexprism was defined as the unit boundary. Note that the grid structure applied to the nonrectangular body is the same as the one that would be assigned for a cuboid with the same minima and maxima in x and y directions. Fig. 9.2.27 illustrates the grid structure that would be applied to the same model as was used in the previous figure but with CMFD acceleration enabled. Because CMFD is normally applied to a coarse mesh defined by the base global grid (unless xycmfd=0), the global grid is always retained when CMFD acceleration is used. Finally, Fig. 9.2.28 illustrates the use of a base grid only. In this case, no grid structure is assigned for unit 1; the bodies are inlaid but are adapted to the base global grid structure.

```
global unit 1
cylinder 30 0.5
cylinder 20 0.6
cuboid 10 0.75 -0.75 0.75 -0.75
…
boundary 10 2 2
```

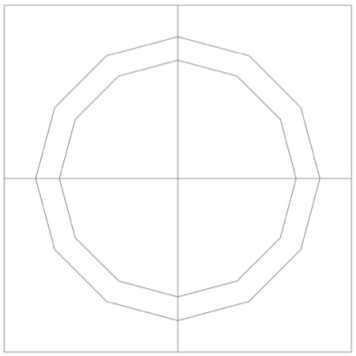


Fig. 9.2.24: Unit with 2 by 2 grid in a simple pin-cell model.

```
unit 1
hexprism 30 0.5
hexprism 20 0.6
…
boundary 20 5 5
global unit 10
cuboid 10 1.0 -1.0 1.0 -1.0
hole 1
…
boundary 10 3 3
```

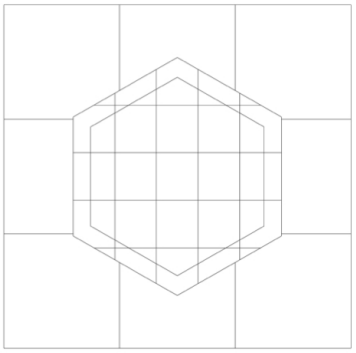


Fig. 9.2.25: Unit with 5 by 5 grid inset into unit with 3 by 3 grid.

```
unit 1
hexprism 30 0.5
hexprism 20 0.6
…
boundary 20 5 5
global unit 10
cuboid 10 1.0 -1.0 1.0 -1.0
hole 1
…
boundary 10 3 3
```

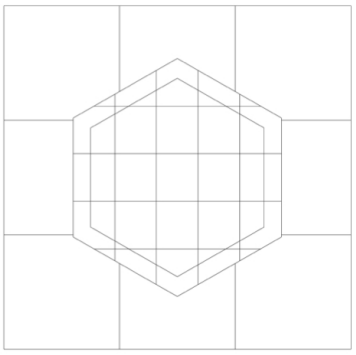


Fig. 9.2.26: Effect of boundary grid specification on noncuboidal unit placed as a hole within a larger unit.

```
unit 1
hexprism 30 0.5
hexprism 20 0.6
…
boundary 20 5 5
global unit 10
cuboid 10 1.0 -1.0 1.0 -1.0
hole 1
…
boundary 10 3 3
```

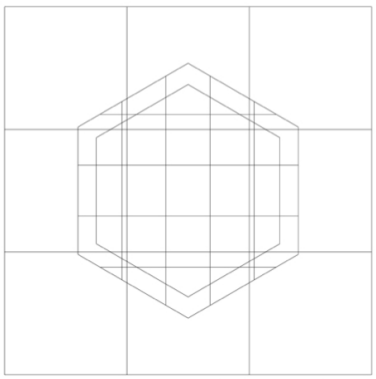


Fig. 9.2.27: Effect of coarse-mesh finite-difference acceleration on grid structure.

```
unit 1
hexprism 30 0.5
hexprism 20 0.6
…
boundary 20
global unit 10
cuboid 10 1.0 -1.0 1.0 -1.0
hole 1
…
boundary 10 3 3
```

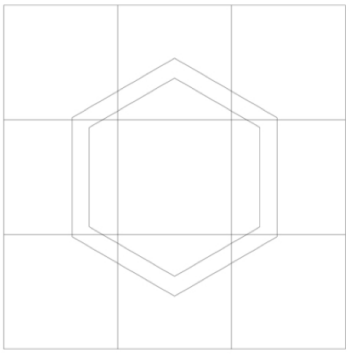


Fig. 9.2.28: Use of base grid without localized grid specification.

The following section provides examples of complete geometry specifications for various models, including bodies, media, and boundary specifications. Each will include one or more uses or boundary specifications for units.

Geometry block examples

The following three subsections present geometry block examples to show how various models may be assembled. Each listing is described briefly and is followed by a figure showing the NEWT grid structure generated for each set of geometry instructions.

Simple pin cell

The following geometry block (Fig. 9.2.29) shows the specifications necessary to define a single pin-cell. The model is reduced to a 1/4 cell to take advantage of symmetry. Mixture 1 is fuel, mixture 2 is fill gas, mixture 3 is clad, and mixture 4 is moderator. Features of this model include the use of chords to obtain 1/4 cylinders and the specification of 20 sides for each cylinder (five sides for a 1/4 cylinder).

```
read geom
global unit 100
cuboid 10 0.63 0.0 0.63 0.0
cylinder 20 0.410
   chord +x=0.0 chord +y=0.0 sides=20
cylinder 30 0.417
   chord +x=0.0 chord +y=0.0 sides=20
cylinder 40 0.475
   chord +x=0.0 chord +y=0.0 sides=20
media 1  1  20
media 2  1  30 -20
media 3  1  40 -30
media 4  1  10 -40
boundary 10  4  4
end geom
```

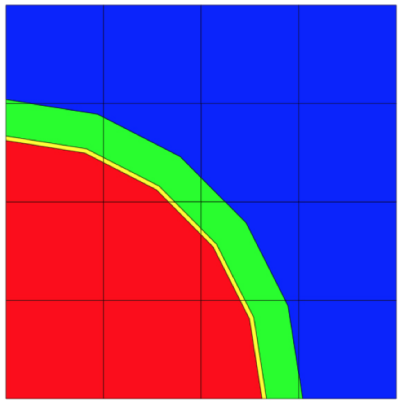


Fig. 9.2.29: Geometry model for infinite-lattice pin cell with fuel, gap, clad, and moderator.

Hexagonal assembly

The geometry block below (Fig. 9.2.30) is used to describe a hexagonal fuel assembly within a hexagonal shroud. Each cylinder is placed individually, followed by a series of media statements that fill each cylinder. The hexagonal moderator area is surrounded by a hexagonal shroud of cladding material. Note that NEWT allows only cuboid and hexprisms as outer boundaries for the global unit. This model could also have been assembled with a unit definition for a set of cylinders, which could then be placed in the global unit using holes or by defining a single pin cell and placing it using an array.

```
read geom
global unit 10
hexagon 15 1.6281 origin y=2.44 x=2.1131
hexagon 14 1.5848 origin y=2.44 x=2.1131
cylinder 21 0.37750 origin y=1.83 x=1.05655
cylinder 51 0.4550 origin y=1.83 x=1.05655
cylinder 22 0.37750 origin y=3.05000 x=1.05655
cylinder 52 0.4550 origin y=3.05000 x=1.05655
cylinder 23 0.37750 origin y=1.22000 x=2.11310
cylinder 53 0.4550 origin y=1.22000 x=2.11310
cylinder 24 0.37750 origin y=2.44000 x=2.11310
cylinder 54 0.4550 origin y=2.44000 x=2.11310
cylinder 25 0.37750 origin y=3.66000 x=2.11310
cylinder 55 0.4550 origin y=3.66000 x=2.11310
cylinder 26 0.37750 origin y=1.83 x=3.16965
cylinder 56 0.4550 origin y=1.83 x=3.16965
cylinder 27 0.37750 origin y=3.05000 x=3.16965
cylinder 57 0.4550 origin y=3.05000 x=3.16965
media 2 0 21
media 2 0 22
media 2 0 23
media 2 0 24
media 2 0 25
media 2 0 26
media 2 0 27
media 5 0 51 -21
media 5 0 52 -22
media 5 0 53 -23
media 5 0 54 -24
media 5 0 55 -25
media 5 0 56 -26
media 5 0 57 -27
media 5 0 15 -14
media 6 0 14 -51 -52 -53 -54 -55 -56 -57
boundary 15 15 15
end geom
```

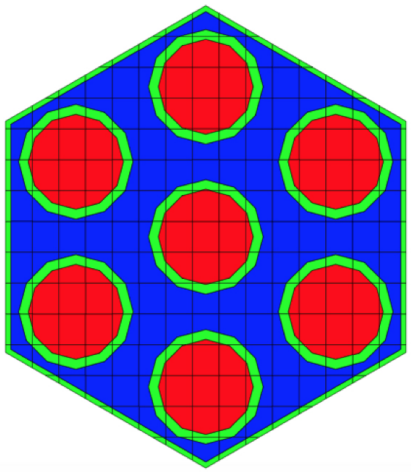


Fig. 9.2.30: Geometry model of hexagonal fuel assembly.

Advanced CANDU reactor ACR-700 assembly

This example of a geometry block (Fig. 9.2.31) is included to illustrate the complexity of design that is possible through the use of simple bodies, units, and holes. The ACR-700 design cannot be modeled using an array because pins are not placed in a repeating lattice pattern. Features of this example include use of holes; use of noncuboidal units placed in holes; and localized pin-cell grid refinement by (1) decreased mesh size in fuel elements (three outer rings) and (2) increased radial discretization (central pin).

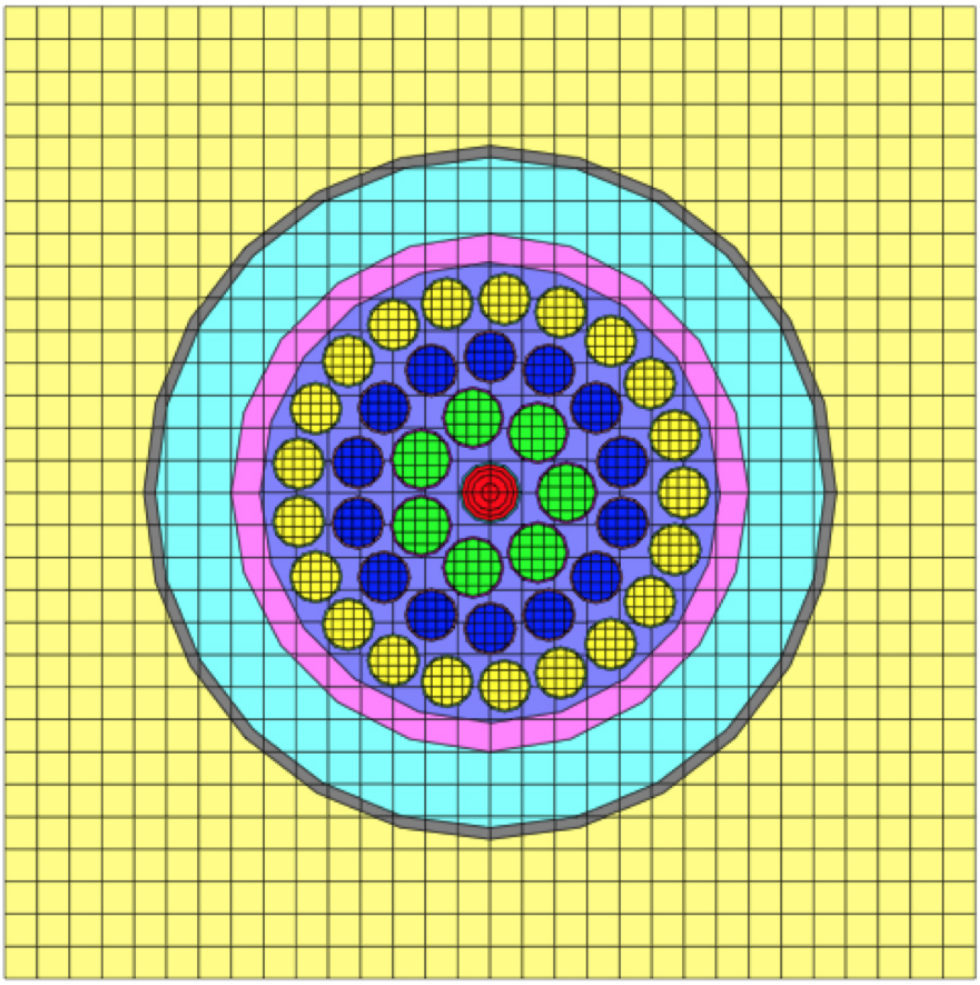


Fig. 9.2.31: ACR-700 fuel assembly model.

Example 9.2.1: ACR-700 fuel assembly geometry.

```
read geom

'central pin
unit 1
cylinder 101 0.200
cylinder 102 0.400
cylinder 103 0.529
cylinder 104 0.629
cylinder 151 0.675
media 11 104
media 11 1 151 -104
boundary 151 2 2

'first ring pin
unit 2
cylinder   201  0.629
cylinder   251  0.675
media      2  1  201
media     21  1  251 -201
```

(continues on next page)

(continued from previous page)

```
boundary 251 5 5

'second ring pin
unit 3
cylinder   301  0.533
cylinder   351  0.575
media      3  1  301
media     31  1  351 -301
boundary 351 5 5

'third ring pin
unit 4
cylinder   401  0.533
cylinder   451  0.575
media     4   1  401
media     41  1  451 -401
boundary  451 5 5

global unit 10
cuboid   100  11 -11 11 -11
cylinder   500  5.1689 sides=20

hole 1
hole 2  origin x=-1.56318 y= 0.75279
hole 2  origin x=-0.38607 y= 1.69150
hole 2  origin x= 1.08176 y= 1.35648
hole 2  origin x= 1.73500
hole 2  origin x= 1.08176 y=-1.35648
hole 2  origin x=-0.38607 y=-1.69150
hole 2  origin x=-1.56318 y=-0.75279
hole 3 origin x=-2.99790 y= 0.68425
hole 3 origin x=-2.40413 y= 1.91723
hole 3 origin x=-1.33419 y= 2.77048
hole 3 origin            y= 3.07500
hole 3 origin x= 1.33419 y= 2.77048
hole 3 origin x= 2.40413 y= 1.91723
hole 3 origin x= 2.99790 y= 0.68425
hole 3 origin x= 2.99790 y=-0.68425
hole 3 origin x= 2.40413 y=-1.91723
hole 3 origin x= 1.33419 y=-2.77048
hole 3 origin x= 0.00000 y=-3.07500
hole 3 origin x=-1.33419 y=-2.77048
hole 3 origin x=-2.40413 y=-1.91723
hole 3 origin x=-2.99790 y=-0.68425
hole 4 origin x=-4.33602 y= 0.65355
hole 4 origin x=-3.95075 y= 1.90258
hole 4 origin x=-3.21443 y= 2.98256
hole 4 origin x=-2.19250 y= 3.79752
hole 4 origin x=-0.97575 y= 4.27506
hole 4 origin x= 0.32769 y= 4.37274
hole 4 origin x= 1.60202 y= 4.08188
hole 4 origin x= 2.73400 y= 3.42833
hole 4 origin x= 3.62306 y= 2.47016
hole 4 origin x= 4.19019 y= 1.29250
hole 4 origin x= 4.38500
hole 4 origin x= 4.19019 y=-1.29250
hole 4 origin x= 3.62306 y=-2.47016
hole 4 origin x= 2.73400 y=-3.42833
hole 4 origin x= 1.60202 y=-4.08188
hole 4 origin x= 0.32769 y=-4.37274
hole 4 origin x=-0.97575 y=-4.27506
hole 4 origin x=-2.19250 y=-3.79752
hole 4 origin x=-3.21443 y=-2.98256
hole 4 origin x=-3.95075 y=-1.90258
```

(continues on next page)

(continued from previous page)

```
hole 4 origin x=-4.33602 y=-0.65355
cylinder   501  5.8169 sides=20
cylinder   502  7.55   sides=24
cylinder   503  7.8    sides=24
media     13  1  500
media      5  1 501 -500
media      6  1 502 -501
media      7  1 503 -502
media      8  1 100 -503
boundary  100 50 50

end geom
```

Summary of geometry specifications

This section is provided as a quick reference for geometry statements. Details and examples of the usage of each geometry specification are provided in previous subsections of this manual. For each definition, the lists of permitted modifiers are listed.

Unit definition statements

```
[global] unit   unit_id
```

The *unit* statement is used to initiate the definition of each unit used. *unit_id* is an integer identification label for the unit and must be unique. One (and only one) global unit is required in each geometry model. Modifiers: none.

```
cuboid  body_id  xmax xmin ymax ymin [modifier_list]
```

The *cuboid* statement is used to define a rectangular shape. *body_id* is the integer identification label for the cuboid and must be unique within the unit it is used. The rectangle is defined such that the coordinates (x_{min}, y_{min}) and (x_{max}, y_{max}) represent the lower-left and upper-right vertices of the cuboid. Modifiers: *rotate, com*.

```
cylinder body_id  radius [modifier_list]
```

The *cylinder* statement defines a circle with integer label *body_id* and radius *radius*, placed with its center at the origin (0,0) of the unit. Modifiers: *origin, rotate, chord, sides, com*.

```
hexprism  body_id  radius [modifier_list]
```

The *hexprism* statement defines a standard hexagon with integer label *body_id* and inscribed radius *radius*, placed with its center at the origin (0,0) of the unit. A standard hexagon has vertices located on north (top) and south (bottom) regions of the shape. Modifiers: *origin, rotate, chord, sides, com.*

```
rhexprism  body_id  radius [modifier_list]
```

The *rhexprism* statement defines a rotated hexagon with integer label *body_id* and inscribed radius *radius*, placed with its center at the origin (0,0) of the unit. A rotated hexagon has vertices located on east (right) and west (left) regions of the shape. Modifiers: *origin, rotate, chord, sides, com.*

```
wedge  body_id  xbase xpt ypt [modifier_list]
```

The *wedge* statement defines a triangle with integer label *body_id* placed with a vertex at (0,0), a vertex at (x_{base},0), and a vertex at (x_{pt},y_{pt}). Modifiers: *origin, rotate, com.*

```
array arrayid body_id place i j xij yij  [modifier_list]
```

The *array* placement statement specifies the placement of an array with identification number *arrayid* (defined in the array data block), within shape *body_id*. If the *place* statement is used, the array element located in row *i* (counted from the bottom) and column *j* (counted from the left) is placed such that its origin is located at spatial coordinate (*xij*, *yij*) of the unit in which it is placed. Modifiers: *com*.

```
hole unit_id [modifier_list]
```

The *hole* statement is used to place a different unit identified by label *unit_id* within the current unit. The origin of the unit being placed will be located at the origin of the current unit but can be repositioned using the *origin* modifier. Modifiers: *origin, rotate, com*.

```
media  materialid  bias_placeholder  reg_def_vector
```

The *media* statement assigns material properties associated with mixture index *materialid* to a shape region defined within a unit. The *bias_placeholder* term is not currently used but is retained for consistency with KENO-VI; typically it is assigned a value of 1. *reg_def_vector* is the region definition vector and assigns the mixture placement relative to shapes within the unit. If the index is positive, the shape region is included in the material assignment; if negative, it is excluded. Modifiers: none.

```
boundary  body_id  [x-discretization y-discretization]
```

The *boundary* statement is used to define the outer boundary of the unit, corresponding to the outer bounds of the shape *body_id* within the unit. This shape must exist and must contain all other bodies associated with the unit. The *x-discretization* and *y-discretization* terms are integers (≥2) that specify the number of cells to be placed in the unit in the x-direction and y-direction, respectively. A grid specification is required for the global unit but is optional for other units. Modifiers: none.

Geometry modifiers

```
origin x=xnew y=ynew
```

Used to relocate the origin of cylinders, hexprisms, rhexprisms, and holes to new co-ordinate (x_{new}, y_{new}). The default (if not specified) is to place the origin of the body at (0, 0).

```
rotate  a1=A
```

Causes a body to be rotated by an angle of *A* degrees (counterclockwise) around its origin. It can be applied to holes and all basic shapes; the default is 0 degrees.

```
chord  +/- x=xplane
chord  +/- y=yplane
```

Chords are used to truncate a shape at the line x=*chord* (or y=*chord*). Multiple chord commands are allowed, but only one line (either in x- or y-direction) is specified for each. If the negative keyword `-x` (or `-y`) appears, then the part of the shape to the left of (below) the chord cut is retained. Similarly, if the positive keyword (`-x` or `-y`) is used, then the portion of the shape to the right of or above the chord is retained. Chords may be applied to cylinders, hexprisms, and rhexprisms only.

```
sides=N
```

For use with the cylinder statement, the *sides* modifier specifies the number of sides on the regular polygon used to approximate the cylinder. Its default is N = 12. The radius of the polygon is adjusted such that the area of the polygon matches the area of the cylinder it is approximating. The *sides* modifier is unique to NEWT and is not used by KENO-VI.

9.2.3.7 Boundary conditions

The *geometry data* block is generally followed by the *bounds* data block, in which boundary conditions for the sides of the bounding shape in the global array are specified. NEWT supports the use of cuboid, hexprism, and wedge shapes to define outer boundaries. This results in the need to specify up to 6 surface boundaries, on up to 8 spatial orientations. In other words, a cuboid will have boundaries on +x, -x, +y, and -y faces. A regular hexprism will have boundaries on +x and -x faces; it will also have boundaries on the four sloped sides of the hex. In order to identify the sense of sides for specification of boundary conditions, NEWT applies an eight-point compass nomenclature. The four permitted rectangular boundary surfaces are identified as +x, -x, +y, and –y, corresponding to east (E), west (W), north (N), and south (S) faces, as illustrated in Fig. 9.2.32 Sloped (non-rectangular) surfaces are identified as +x+y, +x-y, -x-y, and -x+y, for northeast (NE), southeast (SE), southwest (SW), and northwest (NW) surfaces. No assumptions are made on the slope of the various non-rectangular surfaces; for the bodies available within NEWT, it is not possible to have more than one surface in each octant.

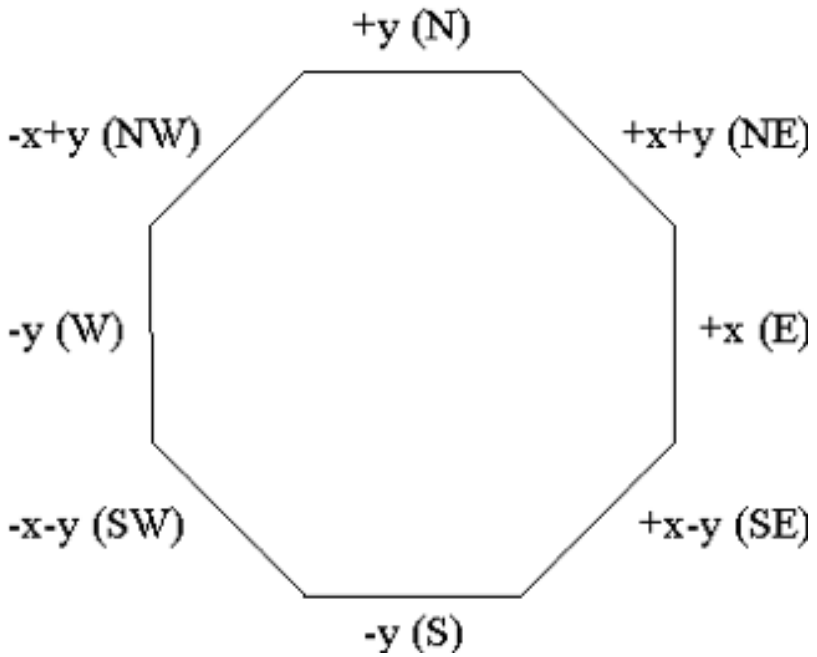


Fig. 9.2.32: Two-dimensional boundary condition surface orientations.

Currently, full specification of boundary conditions is permitted only when the boundary body for the global unit is a cuboid. Only white and vacuum boundary conditions are permitted on non-rectangular surfaces.

Boundary conditions therefore may only be specified for the ± x and ± y faces of a boundary cuboid. Four boundary conditions are currently supported:

1. reflective (default),

2. white,
3. vacuum, and
4. periodic.

Albedo boundary conditions are not yet supported but will be available in a future release. Additional information on the meaning of each boundary condition type is provided in the following section.

Boundary conditions descriptions

Boundary conditions are mathematical approximations used to describe the behavior of neutrons when they cross a problem boundary. Typically, transport methods provide for reflective (or mirror), white, vacuum, or periodic boundary conditions. The following subsections describe and illustrate these four types of boundary conditions.

Reflective boundary condition

For the reflective boundary condition, the incoming angular flux is set equal to the outgoing angular flux in the direction corresponding to mirror or specular reflection. As shown in Fig. 9.2.33, a given quantity of neutrons leaving a boundary (dotted line) in a particular direction will be returned (solid line of same color) to the system with the same quantity but at a mirrored angle to the initial leakage direction.

Note: In the following figures, a dashed arrow indicates neutrons leaving the system while a solid arrow represents those returning to the system. The length of the arrow is proportional to the number of neutrons; therefore, longer arrows represent more neutrons than do shorter arrows.

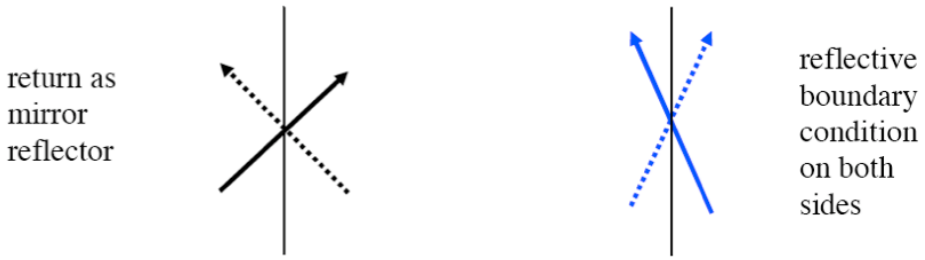


Fig. 9.2.33: Reflective boundary condition.

White boundary condition

For the white boundary condition, the incoming angular fluxes are each set equal to a single value chosen such that the net flow across the boundary is zero. The white boundary provides isotropic return (solid lines) at a boundary (Fig. 9.2.34).

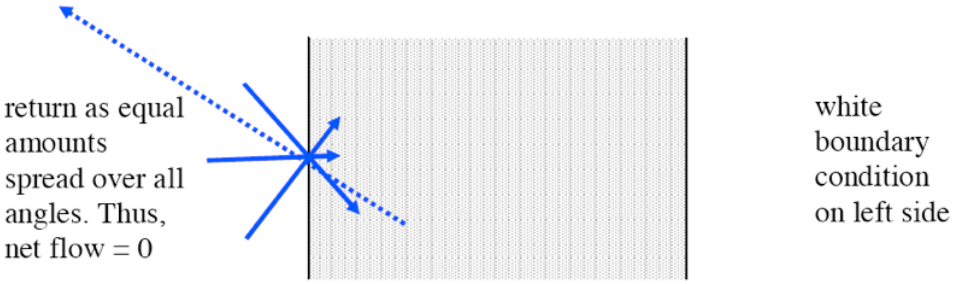


Fig. 9.2.34: White boundary condition.

Vacuum boundary condition

A vacuum boundary condition means that no neutrons will reenter the boundary. Thus, any neutron exiting the system through a boundary with a vacuum boundary condition is permanently lost to the system. This condition is illustrated in Fig. 9.2.35.

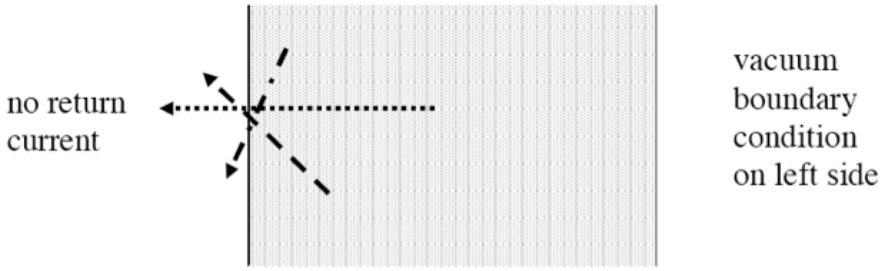


Fig. 9.2.35: Vacuum boundary condition.

Periodic boundary condition

For the periodic boundary condition, the incoming angular flux on a boundary is set equal to the outgoing angular flux on the opposite boundary. Fig. 9.2.36 shows the leakage leaving each boundary (dotted lines) being returned at the same quantity and angle on the opposite boundary (solid line of same color). When the periodic boundary condition is used, it must be applied to both opposing boundaries.

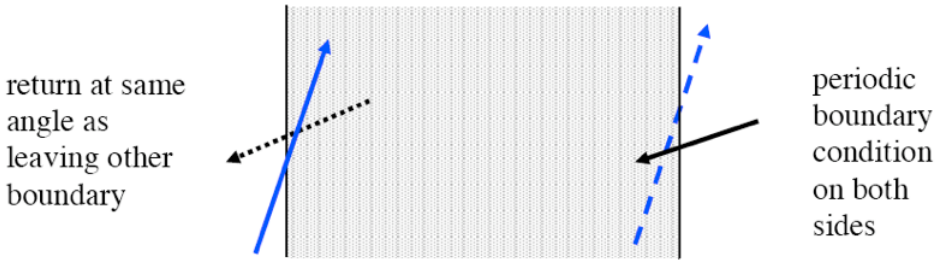


Fig. 9.2.36: Periodic boundary condition.

Boundary condition specification

The standard format for boundary condition specifications is as follows:

```
read bounds
-x=west_BC  +x=east_BC  -y=south_BC  +y=north_BC
+x+y=northeast_BC   +x-y=southeast_BC
-x+y=northwest_BC   -x-y=southwest_BC
end bounds
```

where *west_BC, east_BC, south_BC, north_BC, northeast_BC, southeast_BC, northwest_BC, southwest_BC* are each one of the eight possible boundary condition options. The name of the boundary condition requires only the number of leading characters required to make the name unique. For this set, the first letter is sufficient; that is, +x=v, +x=vac +x=vacu, and +x=vacuum are all equivalent and specify a vacuum (zero return) boundary condition on the right side of the global cuboid.

In keeping with KENO-VI, multiple shortcuts exist to simplify the specification. For example, a single boundary condition can be assigned to all four sides simultaneously with the *all*= specifier:

```
read bounds
all=refl
end bounds
```

All KENO-VI boundary face keywords that do not reference the z-dimension are allowed and are listed in Table 9.2.2.

Table 9.2.2: Boundary condition specifiers accepted by NEWT.

Keyword	**Boundary edge**
+x, +xb	East (right)
-x, -xb	West (left)
+y, +yb	North (top)
-y, -yb	South (bottom)
all, xyf, yxf	All boundaries
+xy, +yx, +fc	East (right) + north (top)
-xy, -yx, -fc	West (left) + south (bottom)
xfc	West (left) + east (right)
yfc	North (top) + south (bottom)
+x+y	Northeast (top right)
-x+y	Southeast (bottom right)
-x-y	Southwest (bottom left)
-x+y	Northwest (top left)

9.2.3.8 General cross section weighting

NEWT performs cross section weighting by mixture and optionally by homogenization zone (Sect. 9.2.3.10). Weighting is always used in conjunction with an energy collapse. Cross section weighting is performed over a spatial and energy domain; the resulting average (weighted) cross sections will preserve all reaction rates in the collapsed cross section set; that is,

$$\sigma^i_G = \frac{\int_r N^i(r)dr \int_G \sigma^i(E,r)W(E,r)dE}{\int_r N^i(r)dr \int_G W(E,r)dE}, \tag{9.2.29}$$

where

$\sigma^i_G \equiv$ average (weighted) cross section in energy group G for nuclide I,

$N^i(r) \equiv$ number density of nuclide i in region r,

$W(E,r) \equiv$ the weighting function within region r,

$\sigma^i(E,r) \equiv$ the cross section from the input library for nuclide i in region r.

Within NEWT, each collapsing region is the spatial region or regions in which a given mixture is placed. Hence, for most of the cross section types, an average cross section for the mixture associated with the spatial domain r (which may include one or more defined regions occupied by that mixture) is calculated by weighting the original problem-specific cross section data for that mixture using as a weighting function the neutron spectrum calculated within spatial domain r.

In practice, the integration of Eq. (9.2.29) is performed as a simple summation over all cells j within region r:

$$\sigma_{i,G} = \frac{N^i_r \sum_{j\in r} \sum_{g\in G} \sigma^i_{g,j} W_{g,j}}{N^i_r \sum_{j\in r} \sum_{g\in G} W_{g,j}}. \tag{9.2.30}$$

Because any region r is defined as the sum of all spatial regions containing a given mixture, is constant everywhere within r.

All multigroup cross sections and related data present on the AMPX library are weighted using appropriate weighting functions. For most basic cross sections, the multigroup flux obtained from the transport solution is the appropriate weighting function and $W_{g,r}$ in Eq. (9.2.30) becomes $\phi_{g,r}$. However, special cross sections and data need special treatment, as described in the following sections.

Scattering cross section transfer matrix weighting

In weighting scattering cross sections, the form of the weighting is slightly more complex:

$$\sigma^i_{s,G\to G'} = \frac{\int_r N^i(r)dr \int_G W(E,r)dE \int_{G'} \sigma^i(E\to E',r)\,dE'}{\int_r N^i(r)dr \int_G W(E,r)dE}, \tag{9.2.31}$$

or, in multigroup format,

$$\sigma^i_{s,G\to G'} = \frac{N^i_r \sum_{j\in r} \sum_{g\in G} W_{g,r} \sum_{g'\in G'} \sigma^i(g\to g')}{N^i_r \sum_{j\in r} \sum_{g\in G} W_{g,r}}. \tag{9.2.32}$$

In general, the scalar flux, $\phi_{\mathrm{g,r}}$ is the appropriate weighting function for scattering cross sections:

$$\sigma^i_G = \frac{N^i_r \sum_{j\in r} \sum_{g\in G} \sigma^i_{g,r}\phi_{g,r}}{N^i_r \sum_{j\in r} \sum_{g\in G} \phi_{g,r}}. \tag{9.2.33}$$

This is an approximation for the higher order moments ($l > 0$) of the scattering cross sections, which should be more properly weighted using the lth moment of the flux instead of the 0th moment (scalar) flux as used in Eq. Eq. (9.2.33). However, because flux moments are generally both positive and negative, NEWT performs higher-order-moment scattering cross section weighting using the positive scalar flux.

Weighting of the collapsed fission spectrum, χ

Weighting is not required for collapsing a fission spectrum vector; the format for a collapsed fission spectrum (χ_G) is a very straightforward summation of the fission spectra in energy groups g spanning the energy domain of the collapsed energy group G:

$$\chi_G = \sum_{g \in G} \chi_g. \tag{9.2.34}$$

Weighting of the number of neutrons per fission

Accurate weighting of ν in an energy and space domain requires weighting by the fission rate; that is,

$$W(E, r) = \sigma^i_f(E, r)\phi(E, r). \tag{9.2.35}$$

Hence, for ν, Eq. Eq. (9.2.33) has the form

$$\nu^i_G = \frac{N^i_r \sum_{j \in r} \sum_{g \in G} \nu^i_g \sigma^i_{f,g,r} \phi_{g,r}}{N^i_r \sum_{j \in r} \sum_{g \in G} \sigma^i_{f,g,r} \phi_{g,r}}. \tag{9.2.36}$$

Weighting of (n,2n), (n,3n), and (n,4n) cross sections

In creating AMPX weighted libraries with NEWT, all data on the original library are collapsed and written to a collapsed working-format library, for each reaction type and for each nuclide. Therefore, each of the (n,*X*n) libraries is collapsed independently using Eq. Eq. (9.2.30) and is written on the weighted library.

However, during NEWT transport calculations, NEWT computes and stores a single effective (n,2n) reaction rate, determined as the weighted sum of the individual reactions:

$$\sigma^{\text{effective}}_{n,2n} = \sigma_{n,2n} + 2\sigma_{n,3n} + 3\sigma_{n,4n}. \tag{9.2.37}$$

The (n,2n) reaction rates reported in NEWT output are those computed for the effective cross section. The effective (n,2n) cross section is not saved to the weighted library.

Calculation and weighting of transport cross sections

Transport cross sections are processed in a different manner from other cross sections. The transport cross section does not represent a purely measurable quantity. Introduced within the P_1 (diffusion) approximation to the neutron transport equation, it attempts to preserve a flux gradient in addition to reaction rate information. Hence, the magnitude of a microscopic transport cross section is affected by both the physics properties of the nuclide in question and the geometrical attributes of the spatial domain where the nuclide resides and the other nuclides present in the same vicinity.

Consistent with XSDRNPM, NEWT provides two options to generate a microscopic transport cross section-based on the "consistent" and "inconsistent" methods for solving the P_1 transport equations. These approximations are referred to as the "outscatter" and "inscatter" approximations because of the nature of the equations used.

Outscatter approximation (inconsistent method)

In the outscatter approximation, the following assumption is made for the transport cross section in group g:

$$\sigma_{tr}^{g} = \sigma_{t}^{g} - \bar{\mu}^{g}\sigma_{s}^{g}, \tag{9.2.38}$$

where σ_{t}^{g} and σ_{s^g} are the total and scattering cross section in group g,

$$\bar{\mu}^{g} = \frac{\sigma_{s,1}^{g}}{3\sigma_{s,0}^{g}} \tag{9.2.39}$$

and

$$\sigma_{s,N}^{g} = \sum_{g'} \sigma_{s,N}\left(g \rightarrow g'\right). \tag{9.2.40}$$

Note that the $\sigma_{s,N}\left(g \rightarrow g'\right)$ terms are the P_N coefficients of the scattering matrix, hence the origin of the term "outscatter."

Inscatter approximation (consistent method)

In the "consistent" P_1 approximation of the transport equation, the transport cross section is defined as

$$\sigma_{tr}(E) = \sigma_{t}(E) - \frac{1}{3J(E)} \int_{0}^{\infty} \sigma_{s,1}\left(E' \rightarrow E\right) J\left(E'\right) dE' \tag{9.2.41}$$

where J is the neutron current and $\sigma_{s,1}$ is the first moment (P_1 coefficient) of the scattering transfer matrix.

If one multiplies Eq. Eq. (9.2.41) by J(E), integrates over group g, and converts to a group-averaged form by dividing by $\int_g J(E)dE$, the following expression is derived:

$$\sigma_{tr}^{g} = \sigma_{t}^{g} - \frac{1}{3J_g} \sum_{g'} \sigma_{s,1}\left(g' \rightarrow g\right) J_{g'}. \tag{9.2.42}$$

This is the "inscatter" approximation. It is consistent because the transport values are explicitly derived from the P_0 and P_1 equations.

Weighting function for transport cross section

Internal investigations have shown that transport cross sections computed using the "outscatter" approximation are more robust in subsequent nodal core calculations as compared with transport cross sections computed using the "inscatter approximation." NEWT computes transport cross sections using the outscatter approximation and collapses the cross section with the scalar flux.

9.2.3.9 Array definition

Any arrays specified in unit definitions within the *read geom* block are defined in terms of form and content in the *read array* data block. The block has the form shown below:

```
read array
ara=arrayid  nux=nx  nuy=ny  typ=aratype  [pinpow=yes/no]
fill i1 i2 i3 i4 ... iN end fill
...
end array
```

where *arrayid* is a unique integer identifier for the array, *nx* is the number of array elements moving left to right (i.e., columns), and *ny* is the number of array elements moving from bottom to top (i.e., rows). The type of array is indicated by *aratype* (e.g., square, hexagonal). The optional parameter *pinpow* may be specified as either *yes* or *no* (default is no) and is used to enable/disable pin power edits for units within the array. The *fill…end fill* specifier set is used to input the elements of the array. A total of *N *entries are required, where* *N = nx*ny.

Each of these portions of the array definition statement is described in more detail below.

Array types

NEWT supports arrays of cuboids, hexprisms, and rotated hexprism elements. Four different array types may be selected. Table 9.2.3 lists the supported array types and corresponding element types. The array type given in the first column lists the keyword associated with each type, as used in the typ= specifier; in some cases, multiple keywords are associated with a specific array type. The element type in the second column provides the boundary shape that can be used in the given array type. For example, a cuboidal (square) array may only be filled with cuboidal units.

Table 9.2.3: NEWT array types with corresponding element types.

Array type	**Element type**
Cuboidal, square	Cuboid
Hexagonal, triangular	Hexprism
Shexagonal	Hexprism
Rhexagonal	Rhexprism

All arrays are filled in a 2-D *i, j* pattern, with *i* varying from 1 to *nux* and *j* varying from 1 to *nuy*. All *nux*nuy* elements of each array must be filled. Fig. 9.2.37 illustrates the layout of a conceptual 4 by 4 cuboidal array, showing the row/column index for each element of the array. Fig. 9.2.38 shows the row/column designation for a 4 by 4 hexagonal array. Because of the shape of a hexprism, the array itself is staggered. However, the row/column numbering is simple to understand.

The stacked hexagon (shexagon) layout, as shown in Fig. 9.2.39, was developed to simply allow an alternate placement algorithm for hexprisms. Any model that can be specified with *typ=hexagonal* can also be specified with *type=shexagonal*; the choice of which form to use is simply one of user preference. It is important to note that beginning with row 3, units will be placed in a physical location different from that of the hexagonal layout when the shexagonal layout is used.

Finally, the rotated hexprism (rhexprism) array is pictured in Fig. 9.2.40. This array is intended to facilitate placement of rhexprisms. The numbering of cells is somewhat irregular because of the staggered rows, but it is easy to follow if one is aware of the fill pattern as illustrated in the figure. Note that the layout of a rhexagonal array can be replicated exactly using a hexagonal or rhexagonal array, placed in a hole, and rotated 90°. Thus, the type of hexprism-based array used can always be tailored to the preferences of the user and all can be used to create the same model.

It is often the case, especially for hexagonal-type arrays, that one does not need to fill all array positions. While the array fill procedure does require that all positions be filled, NEWT provides a mechanism to fill a position with a null unit, effectively skipping that position. This is discussed further under Fill Operations.

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2	3,2	4,2
1,1	2,1	3,1	4,1

Fig. 9.2.37: Layout of a 4 by 4 cuboidal array.

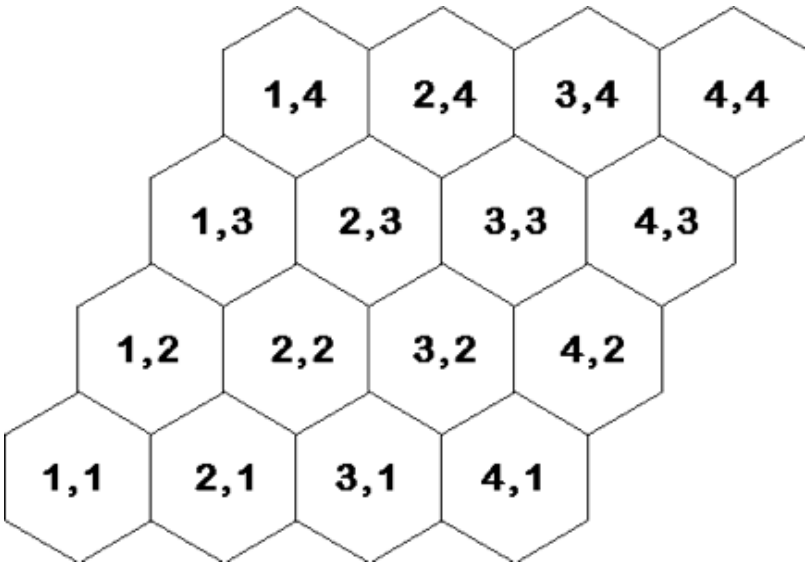


Fig. 9.2.38: Layout of a 4 by 4 hexagonal array.

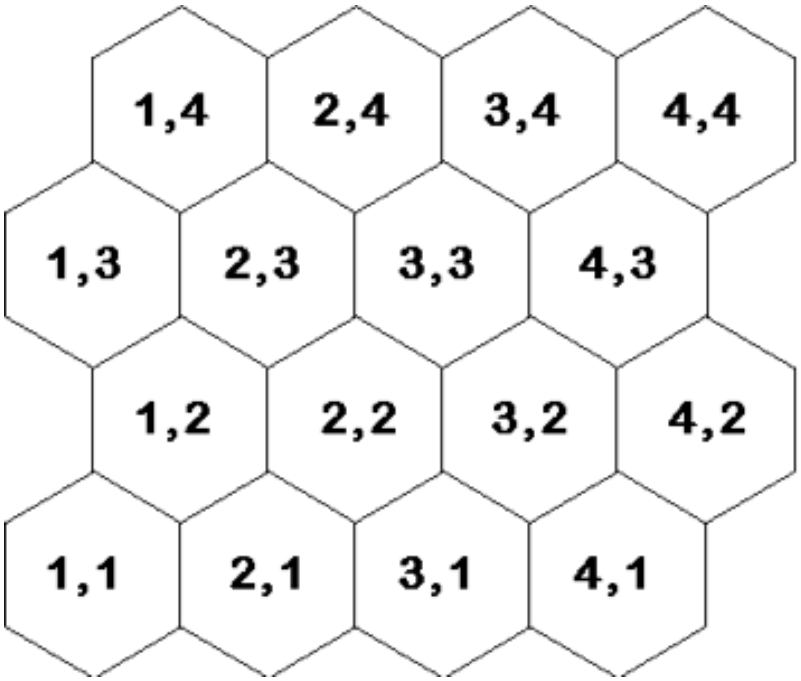


Fig. 9.2.39: Layout of a 4 by 4 stacked hexagonal (shexagonal) array.

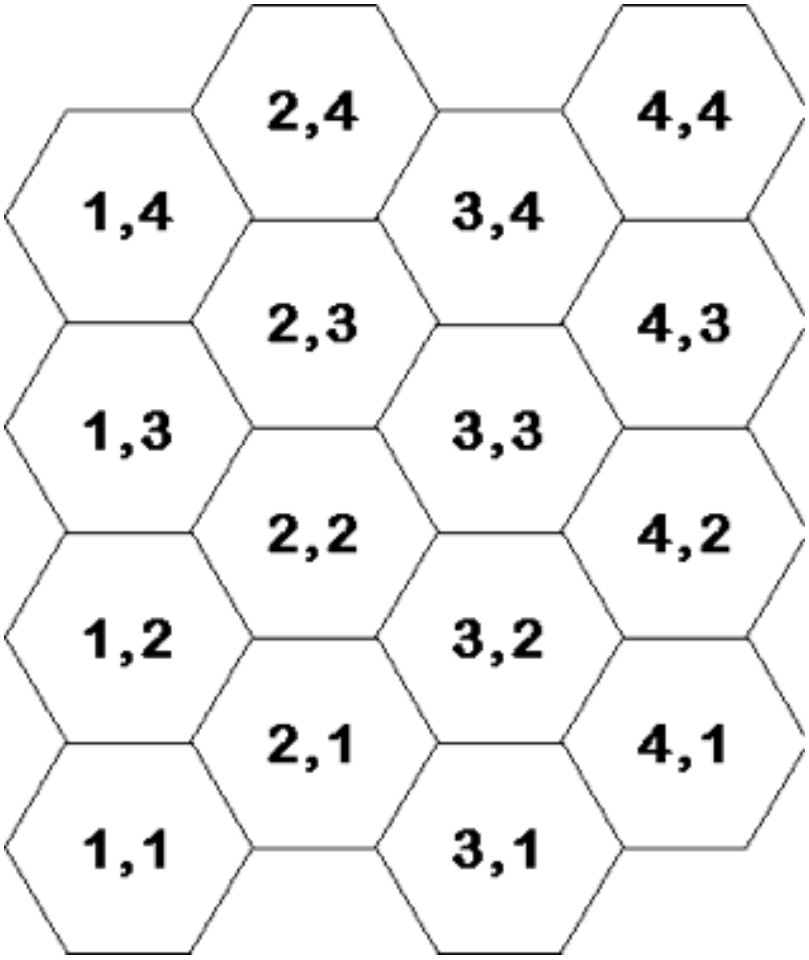


Fig. 9.2.40: Layout of a 4 by 4 rotated hexagonal (rhexagonal) array.

Although all elements of cuboidal arrays **must** be cuboids, they need not be the same size. Elements of each row must have the same height but may have varying widths. Similarly, elements of each column must be of a single common width but may vary in height. Less flexibility is available in hex-based arrays, because of their very nature. Hexagonal and stacked hexagonal arrays may contain only hexprisms, and all must be of the same outer size (although unit contents may vary as needed). Rotated hexagonal arrays likewise are limited to rhexprisms with a single boundary size.

Pin-power edits

In lattice-physics calculations, it is often necessary to obtain a pin-power edit showing the power produced in each fuel pin cell. NEWT uses the array functionality to define pin cells. When *pinpow=yes* is specified, an extra edit is produced that gives the normalized pin power in each pin cell. A pin cell is defined as any element within the array that contains a fissionable nuclide. Pin powers are normalized such that the average of all fuel-bearing array elements is 1.0. Array elements such as burnable poison rods or water holes, which produce no *fission* power, are not included in the power normalization process. The *pinpow* functionality is not available for hexagonal, shexagonal, or rhexagonal lattices.

Output provides an edit of each assembly for which *pinpow=yes* is specified. In addition, a final edit is provided for the entire system, normalized to all fuel cells in all arrays for which *pinpow=yes* is specified.

Pin-power edits are shown in the description of output in Sect. 9.2.5.

Fill operations

The final section of an array specification is the *fill* list. Delimited by *fill* and *end fill* keywords, NEWT expects a list of N=nux*nuy unit numbers, specifying the unit to be placed at each array position. Arrays are filled left to right, starting at the bottom left-hand corner and moving up a row after all columns in the current row are filled. In other words, any of the 4×4 arrays in the figures above would be filled in the following order: (1,1), (2,1), (3,1), (4,1), (1,2), (2,2), (3,2), (4,2), (1,3), (2,3), (3,3), (4,3), (1,4), (2,4), (3,4), (4,4).

The list of elements used to fill an array consists of unit numbers. Each unit used in the fill list must be defined in the geometry block and must be of the shape and size required for the array type and position. However, NEWT provides the ability to fill an array location with a null unit, which in essence skips the current array location. This is accomplished simply by entering unit number "0" (a "null" unit) at the location to be skipped.

Examples of array definitions

Consider a simple 3×3 square array with array ID 10, with unit 1 to be placed in the center of the array, surrounded by unit 2 cells. Such a specification would take the following form:

```
ara=10 nux=3 nuy=3 type=cuboidal fill 2 2 2 2 1 2 2 2 2 end fill
```

This may also be written spanning several lines to help visualize the layout:

```
ara=10 nux=3 nuy=3 type=cuboidal fill
2 2 2
2 1 2
2 2 2 end fill
```

Note that the array is being filled from the bottom so that the actual unit layout is inverted relative to the fill description. A fill specification that places unit 1 at the top center position in this array would be input as follows, where the second-to-last entry in the list is placed in the horizontal center of the top row.

```
ara=10 nux=3 nuy=3 type=cuboidal fill
2 2 2
2 1 2
2 1 2 end fill
```

Arrays may be nested within arrays. Each array must be placed in a unit, but a unit containing an array may be placed within another array. The following example demonstrates the use of nested arrays, along with the use of a null unit.

In this example, we define two units containing cells, a unit containing a smaller array and a global unit containing a larger array:

```
unit 1
'0.5/0.6 cm radius pin
cylinder 30 0.5 sides=20
cylinder 20 0.6 sides=20
cuboid 10 4p0.75
media 1 1 30
media 2 1 20 -30
media 3 1 10 -20
boundary 10 3 3

      unit 2
'0.25/0.3 cm radius pin
cylinder 30 0.25 sides=8
```

(continues on next page)

(continued from previous page)

```
cylinder 20 0.3 sides=8
cuboid 10 4p0.375
media 1 1 30
media 2 1 20 -30
media 3 1 10 -20
boundary 10 3 3

'small array
unit 3
cuboid 10 1.5 0 1.5 0
array 1 10 place 1 1 0.375 0.375
media 3 1 10
boundary 10 5 5       'large array
global unit 4
cuboid 10 3.0 0.0 3.0 0.0
array 2 10 place 1 1 0.75 0.75
media 3 1 10
boundary 10 5 5
```

Now, in a *read array* block, we define array 1, a 2 by 2 array filled with unit 2 cells, and array 2, filled with two unit 1 cells, one unit 3 cell (containing array 1), and one null unit:

```
read array
ara=1 nux=2 nuy=2 typ=cuboidal
fill 2 2
     2 2 end fill
'
ara=2 nux=2 nuy=2 typ=cuboidal
fill 1 3
     0 1 end fill
end array
```

When assembled, the model would appear as shown in Fig. 9.2.41. Note that local grids override the global grid in each array location but that the global grid is seen where the null unit is placed.

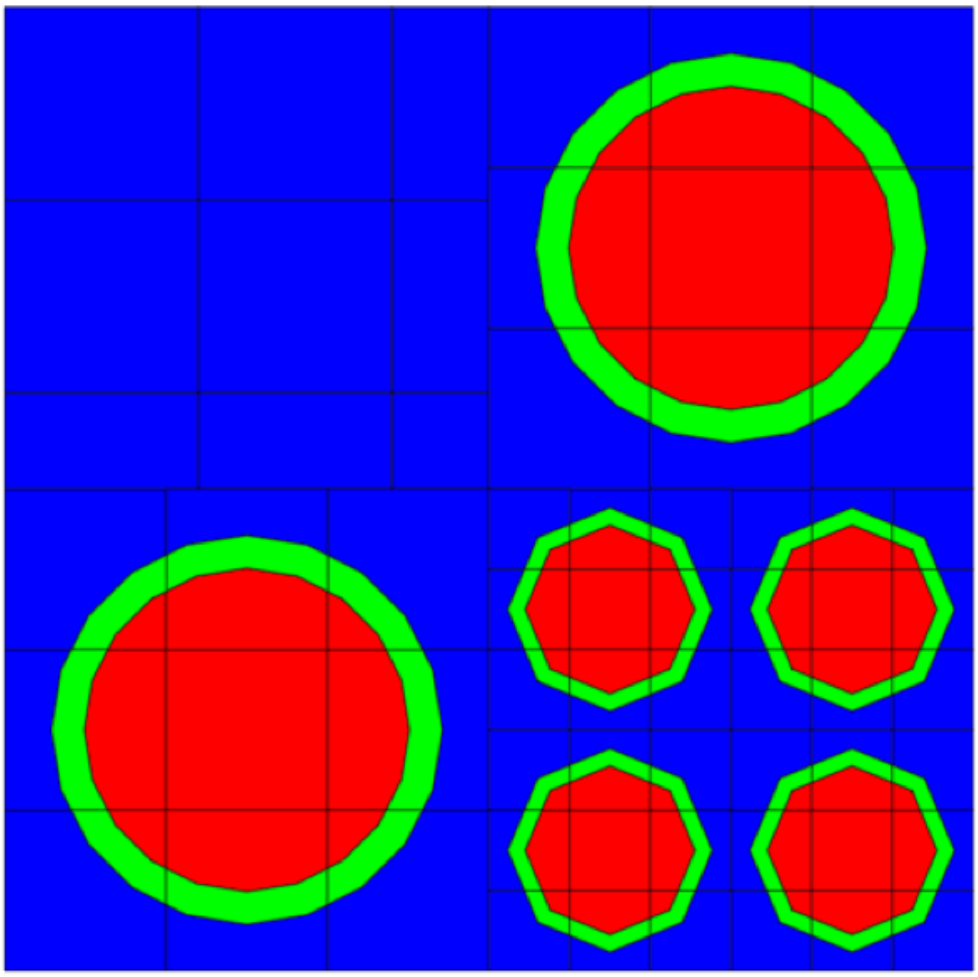


Fig. 9.2.41: Example of nested arrays and a null unit specification.

9.2.3.10 Homogenization block

Homogenization block keyword = homog, hmog, homo

As discussed earlier, NEWT can be used to collapse cross sections to a reduced broad-group format. The cross sections produced from this operation are written as microscopic cross sections for each nuclide in each mixture. NEWT also provides the ability to produce macroscopic weighted cross sections homogenized over one or more mixtures. Homogenized cross sections are created using the collapsing energy structure defined in the collapse data block or the original library's group structure if no collapsing instructions are provided. Flux-weighted collapsed cross sections are combined with number densities and added such that reaction rates in homogenized materials are conserved.

Within the homogenization block, multiple homogenization records are permitted, and the same mixtures may be repeated in different records. Each record provides a recipe defining the mixtures to be homogenized. Homogenization records have the following form:

```
Homogenized     Mixture          List of        end
Mixture ID     Description    Mixtures
```

The homogenized mixture ID is the "nuclide" number under which the mixture is saved on the homogenized cross section library. The value is arbitrary and serves only as a means to identify the cross section set on the library, although each ID must be unique. The mixture description is an alphanumeric label of up to 12 characters that is associated with the mixture; this label provides a little more descriptive ability than the ID itself. The label may not contain blanks. Finally, the label is followed by the list of mixtures to be homogenized, terminated by the end keyword. The list may contain up to 1000 unique mixtures.

A sample homogenization block is shown below. In this illustration, two homogenized mixtures are created. This first consists of five different fuel mixtures (201–205); this could be used to obtain the average fuel cross section for an assembly containing five different fuel types. The homogenized cross sections will be written to the homogenized library as nuclide 500, with label "fuel." The second instruction homogenizes mixtures 201, 210, and 220 from the original problem; this could be used, for example, to homogenize the fuel, clad, and moderator of a fuel pin cell. This cross section set would be written on the same library as nuclide 501, with the label "fuel_cell201."

```
read hmog
500   fuel  201 202 203 204 205  end
501   fuel_cell201   201 210 220 end
end hmog
```

Homogenized (macroscopic) cross sections are saved in an AMPX working-format library at the unit specified by the *hmoglib*= parameter (default=13) [ft13f001].

Instead of calculating the sometimes complicated decay chains to determine cumulative yields, nodal codes often apply simplified models for xenon and samarium. The homogenization block therefore also enables the calculation of fitted yields for ^{135}I, ^{135}Xe and ^{149}Pm for use in nodal codes. Like the Polaris sequence, NEWT relies for this calculation on a set of pre-calculated fitted yields. Up to SCALE 6.2, NEWT and Polaris used different sets of pre-calculated fitted yields. Polaris used yields that were determined based on a set of LWR Polaris calculations with subsequent yield fitting.

To improve consistency between the sequences, since SCALE 6.3, NEWT by default uses the same fitted yields as Polaris. Yield resource `lwr_yields.h5` in SCALE's data directory is used by both sequences. For backwards compatibility, the yields used by NEWT up to SCALE 6.2 are available in resource `legacy_yields.h5`.

Users can switch the yield resource or use their own resource by linking the respective resource into the temporary working directory.

Example 1: Use NEWT legacy yield data

```
read shell
  cp ${DATA}/legacy_yields.h5 ${TMPDIR}/lwr_yields.h5
end shell
```

Example 2: Use user-defined yield data

```
read shell
  cp /path/to/user-resource.h5 ${TMPDIR}/lwr_yields.h5
end shell
```

9.2.3.11 Assembly discontinuity factors

Assembly discontinuity factor (adf) block keyword = adf

Because discontinuity factors have meaning only with respect to homogenized cross sections, ADFs are calculated only if homogenized cross sections are also specified via the Homogenization data block (see Sect. 9.2.3.10). Calculation of ADFs is specified in the *read adf* data block. The three supported formats of this data block are as follows. For a single-assembly model, the following format is used:

```
read adf
1 homg_assm_id n=Y1 s=Y2 e=X1 w=X2
end adf
```

For a reflected assembly model, the following format is used:

```
read adf
2 homg_assm_id homog_refl_id w=Xi
end adf
```

```
read adf
3 homg_assm_id line1 line2 line3 ...
end adf
```

where

`hmog_assm_id` is the identifying label assigned to the homogenized assembly,

`hmog_refl_id` is the label assigned to the homogenized reflector region,

`Y1` is the y-ordinate of the north boundary of the assembly (typically this is y_{max} for the global unit),

`Y2` is the y-ordinate of the south boundary of the assembly (typically this is y_{min} for the global unit),

`X1` is the x-ordinate of the east boundary of the assembly (typically this is x_{max} for the global unit),

`X2` is the x-ordinate of the east boundary of the assembly (typically this is x_{max} for the global unit),

`Xi` is the x-ordinate for the fuel/reflector interface,

`linei` is a series of two ordered pairs (`X1`,``Y2``), (`X1`,``Y2``)that define a line segment in the NEWT grid.

In a single-assembly calculation, only a single homogenized mixture is specified. Leading index 1 indicates that ADFs for the fuel assembly are being calculated; ADFs may be calculated on any or all faces of a rectangular assembly. In a reflected assembly, the leading index is 2, followed by the homogenized mixture ID for the fuel assembly first, then the homogenized mixture ID for the reflector region. An ADF may be requested only for one location, at the fuel/reflector interface. In any configuration, ADFs may be requested for a set of arbitrary line segments defined in the NEWT geometry. In this case, the leading index is 3, followed by the homogenized mixture ID, followed by up to 12 line segments, which are defined by their beginning and ending points. ADFs along these lines are defined as the surface-averaged flux divided by the average flux defined for the associated homogenized mixture. Surface-averaged currents are also edited for each arbitrary line segment; both full and partial currents in both the x- and y- directions are provided. The

net current is also provided in the few-group cross section database file *xfile016*. The orientation of the net current across the line segment is further discussed in Appendix A of TRITON chapter.

Although any homogenized set of mixtures can be specified for each homogenized region, the ADF will have physical meaning only if the homogenized set represents all mixtures in the assembly. Similarly, if a reflector calculation is performed, the *hmog_refl_id* should represent the set of homogenized reflector mixtures. The average collapsed flux in the homogenized mixtures is used to calculate the homogeneous flux for a single-assembly ADF. In a reflector calculation, the homogenized cross sections for the reflector are used to solve the multigroup diffusion approximation (Sect. 9.2.2.5).

Examples illustrate the use of the ADF input specification. Consider a 17×17 pressurized-water-reactor (PWR) design. Because of symmetry, it can be modeled a $1/4$ assembly; therefore, we choose to model the upper right quadrant, as shown in Fig. 9.2.42. Mixtures 1, 2, 3, and 4 represent the fuel, clad, moderator, and guide tube materials, respectively. The west and south sides of the model are the assembly midplanes, so ADFs calculated on these boundaries would have no physical meaning. (These are not real assembly boundaries.) However, because of the symmetry of the assembly, fluxes would be identical on all boundaries. Therefore, selection of either the north or east boundary will yield a valid ADF for all boundaries. We choose to request an ADF calculation for the east (right) side of the assembly.

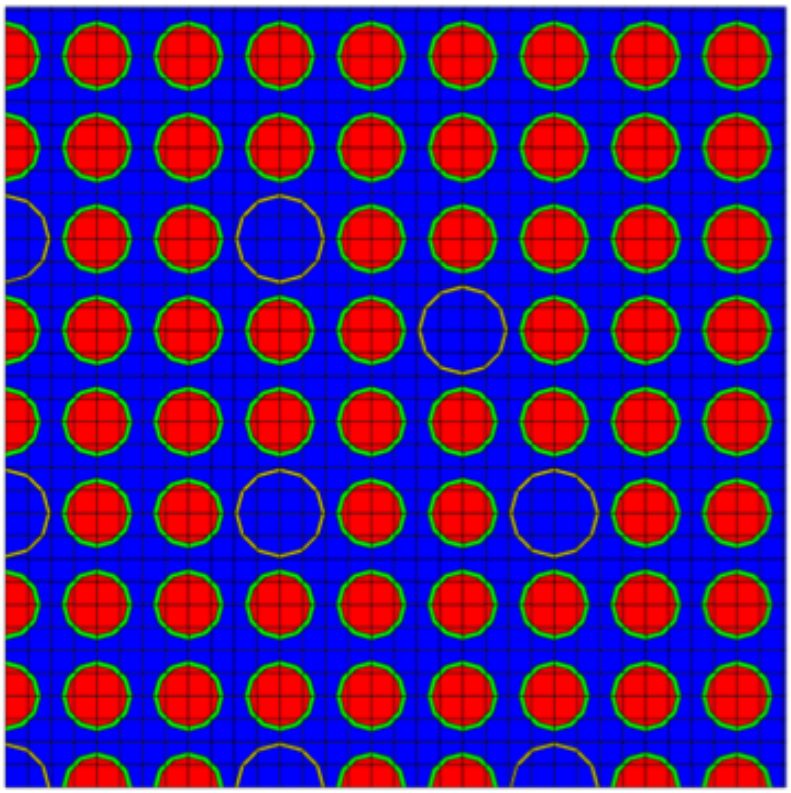


Fig. 9.2.42: Upper-right quadrant of an unreflected PWR assembly.

Assuming we are collapsing 44 energy groups to 2 energy groups, we would specify the following *collapse*, *homog*, and *adf* blocks to obtain 2-group ADFs representative of this assembly:

```
read collapse
22r1 22r2
end collapse
read hmog
500 assm   1 2 3 4 end
end hmog
read adf
1 500 e=10.752
end adf
```

Again, recall that for a single assembly, the ADF in each energy group is simply the average flux on the specified boundary divided by the average flux for the entire assembly, which in this case is the flux in homogenized mixture 500.

ADF can also be calculated using the arbitrary line-segment ADF type. Using this new ADF type, the ADF input would be the following:

```
read collapse
22r1 22r2
end collapse
read hmog
500 assm   1 2 3 4 end
end hmog
read adf
3 500 10.752 0.0 10.752 10.752
end adf
```

In this example, the east-side ADF will be calculated along the line segment starting at (10.752,0) and ending at (10.752,10.752). The values of the line segments depend on a coordinate system of the global unit.

For a reflected model, consider the same type of assembly but representing an assembly placed on the core periphery. It is bounded on one side by a 2 cm stainless steel baffle and 10 cm of water, beyond which is treated as vacuum. Because the reflector calculation is a 1-D solution, it is not possible to directly model a fuel assembly with two reflector boundaries. Typically the assumption is made that the same ADF may be applied in any assembly/reflector boundary and that a 1-D reflector model is all that is necessary. This model is pictured in Fig. 9.2.43. Notice that different mixtures are used in the reflector model. Fluxes used in homogenized mixtures and for generating homogenized cross sections are spatially averaged; thus, it is important to separate the moderator in the reflector from that in the fuel assembly such that average fluxes in each region properly characterize that region. For example, the flux in the reflector will be significantly different (far more thermal) from the flux within the assembly.

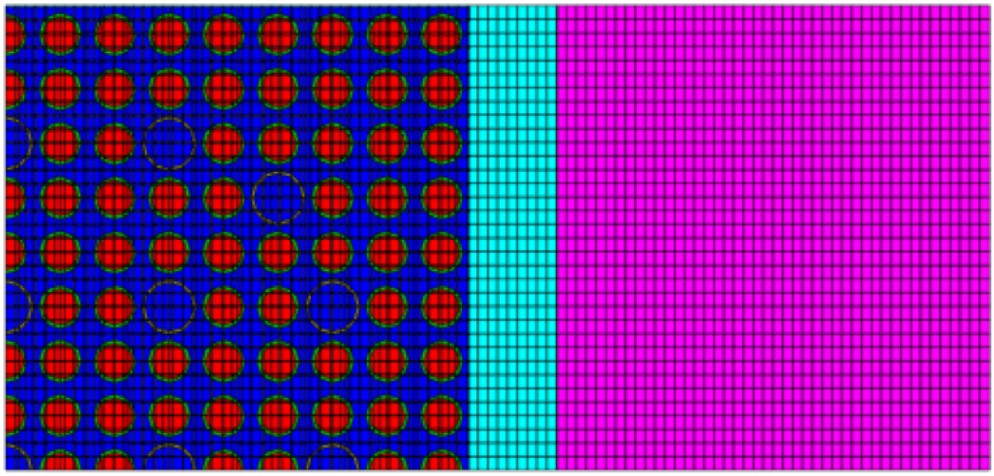


Fig. 9.2.43: Upper-right quadrant of a PWR assembly with baffle and reflector.

Assuming again a collapse from 44 energy groups to 2, the following *collapse*, *homog*, and *adf* blocks would be specified to obtain 2-group ADFs representative of the assembly and the reflector; 2-group cross sections for each homogenized region would also be prepared from the collapse and homogenization instructions:

```
read collapse
22r1 22r2
end collapse
read hmog
500 assm   1 2 3 4 end
501 reflector 5 6 end
end hmog
read adf
2 500 501 w=10.752
end adf
```

9.2.3.12 Flux planes

Fluxplane block keyword = fluxplane, fluxplan, flux, fluxplanes

The fluxplane block is a special output edit that lets one obtain the average scalar flux and currents along any line segment or any continuous set of collinear line segments. One must simply specify the start and end points of a line segment for which a linearly averaged flux is desired. This line segment must correspond to one or more line segments in the model's grid structure, which requires some knowledge of where grid lines exist in the model.

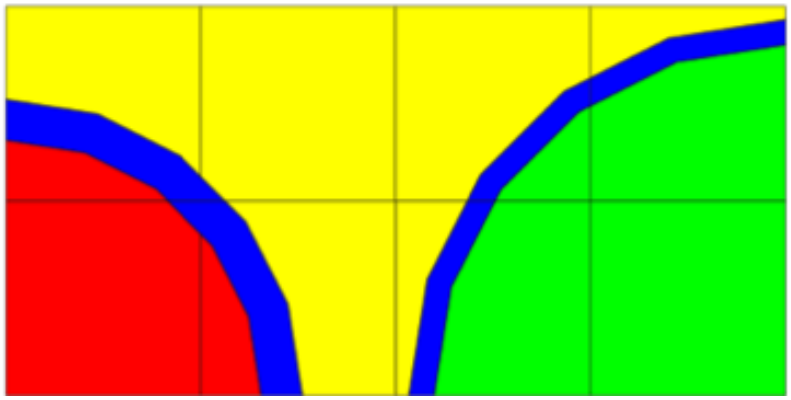
The format of a flux plane specification is as follows:

```
read fluxplane
text_label homg_assm_id xstart ystart xend yend
...
end fluxplane
```

where *text_label* is an alphanumeric description used to label the selected plane in the output, *homg_assm_id* is the identifier in the *homog* block, and *(xstart, ystart)* and *(xend, yend)* are the start and end points, respectively, for the line segment for which an average flux is desired. The *text_label* string must not contain white space and may be up to 16 characters in length.

As an example, we consider a simple model consisting of two dissimilar pin cells (1/4 cells):

```
global unit 1
cuboid 1 1.26 0.0 0.63 0.0
cylinder 2 .4750 chord +x=0 chord +y=0 sides=20
cylinder 3 .4095 chord +x=0 chord +y=0 sides=20
cylinder 4 .6030 chord +y=0 chord -x=1.26 origin x=1.26 sides=20
cylinder 5 .5630 chord +y=0 chord -x=1.26 origin x=1.26 sides=20
media 1 1 3
media 2 1 2 -3
media 10 1 5
media 2 1 4 -5
media 3 1 1 -2 -4
boundary 1 4 2
```



We know a line segment (actually, two) exists at $x = 0$, $x = 0.63$, and $x = 1.26$, between $y = 0$ and $y = 0.63$. Thus, a legitimate set of flux plane specifications would be the following:

```
read fluxplane
cell_interface 0.63 0.0 0.63 0.63
midplane_cell1 0.0 0.0 0.0 0.63
midplane_cell2 1.26 0.0 1.26 0.63
end fluxplane
```

This will provide a summary of fluxes and currents at each line segment in fine-group structure, and if a collapse is performed, in broad-group structure. Results from a calculation with a two-group collapse appear as follows:

```
Broad Group Fluxes:
 Group    cell_interface    midplane_cell1    midplane_cell2
    1      5.906586D+01      5.903113D+01      5.930420D+01
    2      7.645792D+00      7.641424D+00      6.705809D+00

Broad Group Currents (x):
 Group    cell_interface    midplane_cell1    midplane_cell2
    1     -3.481505D-01      0.000000D+00      0.000000D+00
    2      3.004759D-01      0.000000D+00      0.000000D+00

Broad Group Currents (y):
 Group    cell_interface    midplane_cell1    midplane_cell2
    1      1.081450D-01      1.652161D-01      1.120160D-01
    2     -1.031163D-01     -1.647396D-01     -9.993214D-02
```

Output also includes +x, -x, +y, and -y components of currents.

9.2.3.13 Mixing table block

Mixing table block keyword = mixt, mixtable

Generally, NEWT calculations are performed using a cross section library and mixing table prepared in advance by other SCALE modules. However, NEWT allows the user the ability to manually specify the isotopic composition of each mixture used in a NEWT model. If a mixing table block is read, any existing mixing table data file is ignored. Therefore, all mixtures specified in the material block must be mixed in the mixing table block.

The format of the mixing table is simple and straightforward. For each nuclide used, three parameters must be supplied: (1) *mixid*, the mixture ID number into which the nuclide is to be mixed; (2) *nuclideid*, the SCALE ID number for the nuclide (which must exist on the cross section library being referenced); and (3) *concentration*, the number density (atoms/b-cm) of the nuclide in this mixture. The same nuclide may appear in multiple mixtures or more than once in a single mixture if desired. Macroscopic cross sections are determined for each mixture by the following formula:

$$\Sigma^R = \sum_i \sigma_i^R N_i \tag{9.2.43}$$

where

Σ^R is the mixed macroscopic cross section for reaction R in the mixture,

N_i is the number density of nuclide i,

σ_i^R is the microscopic cross section for reaction R in nuclide i.

The form of the mixing table block is as follows:

```
read mixt
mixid1 nuclideid1 concentration1
mixid2 nuclideid2 concentration2
mixid3 nuclideid3 concentration3
...
mixidN nuclideidN concentrationN
end mixt
```

This concludes this list of input blocks available within NEWT. The following section provides a list of sample inputs used to represent a variety of configurations and use of codes. These examples are intended to provide a broader illustration of the use of NEWT in a range of potential applications.

9.2.4 EXAMPLES OF INPUTS

This section provides annotated sample input listings for three different model types, showing the use of a number of different options and approaches in model development for a variety of applications. These samples use the TRITON T-XSEC sequence to prepare cross sections for stand-alone NEWT calculations. In general, this is more easily accomplished as a TRITON T-NEWT calculation in which cross section processing and a NEWT transport solution are combined into a single calculation. However, the user is directed to the TRITON user's manual (Chapter T1.4 of the SCALE manual) for details on T-XSEC and T-NEWT calculations. The examples are intended simply to demonstrate the use of the NEWT code. The T-XSEC data are included to allow a user to observe the mixture definitions used in the NEWT input in its calculation. These problems are also available as sample problems in the SCALE distribution.

9.2.4.1 Sample 1

Sample 1 illustrates the use of a series of three consecutive stand-alone NEWT calculations. Annotated input for this problem is given in Fig. 9.2.44. The calculation begins with SCALE standard composition specifications used to prepare a problem-specific weighted cross section library and mixing table for use by NEWT. In this case the T-XSEC sequence of the TRITON control module is used. This input is described in the TRITON chapter and is not described further here.

The first NEWT case uses no parameter block; thus, all default parameters are applied. The default is an eigenvalue calculation, with cross sections read from ft04f001 (xnlib=4) and collapsed cross sections written to ft30f001 (wtdlib=30). The 238-group cross section library is collapsed to a 44-group library using mixture-weighted fluxes. The model calculates the eigenvalue for a simple 1/4 pin cell. The center of the pin is placed at the origin, the lower-left corner of the global unit boundary, inlaid into a 2 by 2 base grid. The grid structure is illustrated in Fig. 9.2.45.

The second case performs the same calculation using the collapsed cross section library created by the first case. Parameter *restart*=no is set to prevent the code from attempting a restart from the existing library. Because the first case saved 238-group fluxes and the second case uses 44 energy groups from the collapsed set, a restart is not possible.

The third NEWT case is a calculation identical to the second case, although the input is different. In this case, the flux restart file from the previous calculation is used as a first guess for fluxes. This is permitted since both cases used the same cross section library and therefore have the same energy boundaries. For this case, the "read geom" data block is omitted, telling NEWT to use the geometry restart file from the previous case. This allows a rapid restart, since no geometric data need to be recomputed. Because no other parameters are changed, this case will converge after one outer iteration to the same eigenvalue as in the first case.

=t-xsec	T-XSEC sequence of TRITON
2-D 1/4 pin-cell model with MOX fuel	TRITON title
V7-238	238-group ENDF/B-VII library
read comp 'Fuel u-234 1 0 2.5952E-7 900 end pu-238 1 0 4.6610E-5 900 end pu-241 1 0 1.7491E-4 900 end pu-242 1 0 1.3201E-4 900 end o-16 1 0 4.6586E-2 900 end pu-240 1 0 4.8255E-4 900 end pu-239 1 0 1.0156E-3 900 end u-235 1 0 5.4287E-5 900 end u-238 1 0 2.1387E-2 900 end	Mixed oxide fuel isotopic specification
' zirc zr-90 2 0 3.8657E-2 620 end fe 2 0 1.3345E-4 620 end cr 2 0 6.8254E-5 620 end	Zircaloy clad
' h2o h-1 3 0 4.8414E-2 575 end o-16 3 0 2.4213E-2 575 end b-10 3 0 4.7896E-6 575 end b-11 3 0 1.9424E-5 575 end end comp	Borated water moderator
read celldata latticecell squarepitch pitch=1.3127 3 fueld=0.8200 1 cladd=0.9500 2 end end celldata end	Square-pitch pin-cell lattice specification
=newt	First stand-alone NEWT calculation
Infinite lattice PWR pin cell	NEWT title
read materials mix=3 pn=2 end mix=2 pn=1 end mix=1 pn=1 end end materials	No parameter block specified; defaults are used. Mixtures 1, 2, and 3 on cross section library used in this problem. Moderator has P-2 scattering; other mixtures use P-1 scattering.
read collapse 7r1 2 3 2r4 5 6 7 8 8 8r9 14r10 6r11 10r12 13 7r14 11r15 12r16 30r17 16r18 2r19 6r20 3r21 6r22 14r23 27r24 10r25 5r26 27 28 29 2r30 31 32 33 2r34 2r35 3r36 2r37 38 39 40 41 42 3r43 9r44 end collapse	238-group cross sections are to be collapsed to this 44-group structure.
read geom global unit 10 cylinder 11 0.41 chord +x=0.0 chord +y=0.0 sides=24 cylinder 12 0.475 chord +x=0.0 chord +y=0.0 sides=24 cuboid 13 0.65635 0.0 0.65636 0.0 media 1 1 11 media 2 1 12 -11 media 3 1 13 -12 boundary 13 2 2 end geom	Geometry specification. Two concentric (fuel/clad) cylinders centered at 0,0, trimmed to 1/4 (upper-right) quadrant using chords. Global unit boundary is defined by the outer boundary of rectangular pin-cell moderator (1/2 of pitch). Bodies are laid onto a 2 by 2 grid. Media 1 (fuel) is inside cylinder 11. Media 2 (clad) is inside cylinder 12 and outside cylinder 11. Media 3 (moderator) is placed inside cuboid 13 but outside cylinder 12.
read bounds all=refl end bounds	Reflective BCs used on all boundaries
end	End of this NEWT case
=newt Infinite lattice PWR pin cell using collapsed x-s	Second NEWT case NEWT title card
read parm xnlib=30 collapse=no restart=no end parm	Parameter block specifies to use unit 30 (file ft30f001) cross sections. These are the collapsed cross sections from first case. No collapsing is to be done, and no restart occurs.
read materials mix=3 pn=2 end mix=2 pn=1 end mix=1 pn=1 end end materials	Same mixtures as before
read geom global unit 10 cylinder 11 0.41 chord +x=0.0 chord +y=0.0 sides=24 cylinder 12 0.475 chord +x=0.0 chord +y=0.0 sides=24 cuboid 13 0.65635 0.0 0.65636 0.0 media 1 1 11 media 2 1 12 -11 media 3 1 13 -12 boundary 13 2 2 end geom read bounds all=refl end bounds end	Same geometry as before
=newt Infinite lattice PWR pin cell with both flux and read parm xnlib=30 collapse=no restart=yes end parm read materials mix=3 pn=2 end mix=2 pn=1 end mix=1 pn=1 end end materials end	Third NEWT case Restart using fluxes from previous case Same mixtures No geometry is specified; geometry from previous case is reused.

Fig. 9.2.44: Sample 1 input listing (annotated).

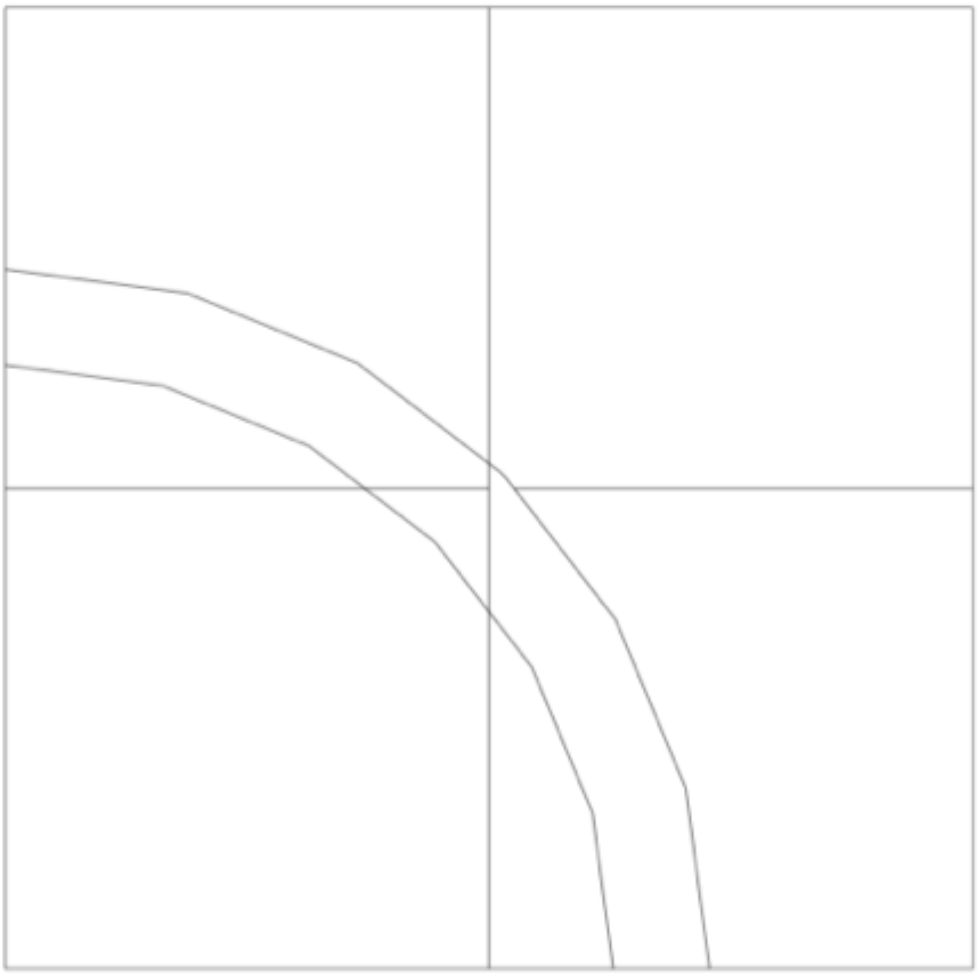


Fig. 9.2.45: Grid structure for 1/4 pin cell of Sample 1.

9.2.4.2 Sample 2

Sample 2 (shown in Fig. 9.2.46 and Fig. 9.2.47) illustrates the use of multiple bodies within a single unit. It highlights the use of media definitions to include and exclude regions when various bodies are used. Although an array can be used to place bodies, this example illustrates a method suitable for use in developing a model for a configuration with an irregular non-array-type structure. This sample problem also highlights the use of partial-current unstructured-mesh CMFD acceleration, which reduces the number of outer iterations from 35 to 21 and the CPU run time by ~25%.

```
=t-xsec
Transport calculation for an inf. lattice triangular pitch fuel
v7-238


read comp
u-234 2 0  2.6436E-7 1000  end
pu-238 2 0 3.6128E-5 1000  end
pu-241 2 0 1.3557E-4 1000  end
pu-242 2 0 1.0233E-4 1000  end
o-16  2 0  4.6553E-2 1000  end
pu-240 2 0 3.7403E-4 1000  end
pu-239 2 0 7.8717E-4 1000  end
u-235 2 0  5.5300E-5 1000  end
u-238 2 0  2.1786E-2 1000  end


' zirc
zr-90 5 0 3.8657E-2 620 end
fe 5 0 1.3345E-4 620 end
cr 5 0 6.8254E-5 620 end


' h2o
h-1 6 0 4.8414E-2 575  end
o-16 6 0 2.4213E-2 575  end
b-10 6 0 4.7896E-6 575  end
b-11 6 0 1.9424E-5 575  end
end comp


read celldata
latticecell triangpitch pitch=1.2200 6 fueld=0.755 2
  cladd=0.91 5  end
end celldata
end


=newt
Transport calculation for an inf. lattice triangular pitch fuel


read parm nazim=3 npolar=2 cmfd=3 xycmfd=4 timed=yes echo=yes
  epsilon=1e-5 inners=3 therms=1 end parm


read materials
  mix=2  pn=1  com="fuel" end
  mix=5  pn=1  com="cladding" end
  mix=6  pn=2  com="water" end
end materials
```

T-XSEC sequence of TRITON
238-group ENDF/B-VII library

Mixed-oxide fuel specification, mixture number 2

Zircaloy clad, mixture number 5

Borated water moderator, mixture number 6

Cell specification for a triangular lattice

End of T-XSEC case

Beginning of NEWT calculation

Parameter block uses product quadrature set: two polar angles with three azimuthal angles, partial-current CMFD acceleration with four fine-mesh cells per coarse-mesh cell. The echo option is turned on, as well as other convergence control options.

List of all problem materials—P_1 scattering is specified in fuel and clad; P_2 scattering, in moderator

Fig. 9.2.46: Sample 2 input listing (annotated).

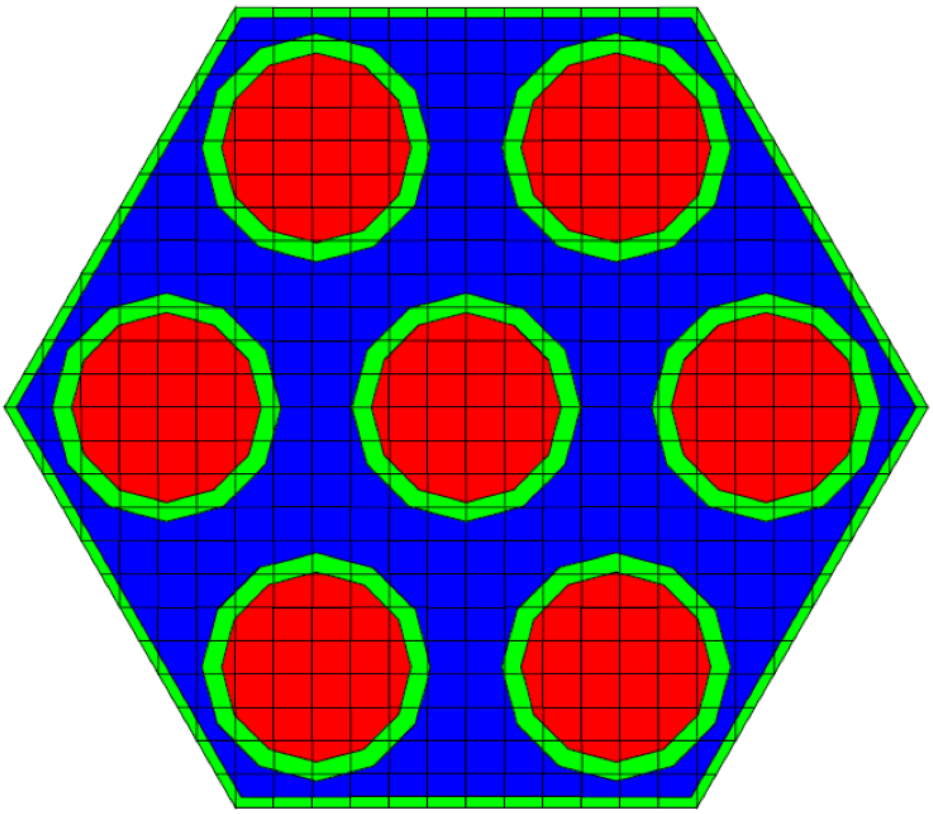


Fig. 9.2.47: Mixture placement and grid structure for model described in Sample 2.

Sample 3

Sample 3 demonstrates the development of a VVER-440 hexagonal fuel assembly. Annotated input for this problem is given in Fig. 9.2.48. The output plot for this model is shown in Fig. 9.2.49. The key attributes of this model are as follows:

1. the use of hexagonal (hexprism) units in a stacked hexagonal array,
2. the use of null units as placeholders in the array,
3. a full model within a rhexagonal outer boundary,
4. the use of white boundary conditions,
5. **the use of the new partial-current-based unstructured CMFD** acceleration for hexagonal-domain configurations, and
6. new type-3 ADF inputs.

Using CMFD acceleration, the number of outer iterations needed for convergence decreased from 21 to 8 with a run-time speedup of ~2.58.

t-xsec sequence of TRITON
title card
238-group ENDF/B-VII library

Define aliases: four fuels, four clads, and four moderators will be defined by the three aliases

Read compositions

Four different fuel mixtures are defined. Because each mixture contains different specifications (different enrichments, Gd in one rod), these cannot be defined by using an alias

Use aliases to define four identical clads with different mixture IDs

Use aliases to define four identical moderators with different mixture IDs

Aliases are used to define a single cell description that will be expanded to four different cell descriptions
End of t-sec case

Begin NEWT calculation
Title card followed by parameter block. CMFD option 3 used in the parameter block, along with four fine meshes per course mesh.

Material block specified four fuel mixtures, a clad mixture, and outer (can) clad mix—all with P-1 scattering—and a single moderator with P-2 scattering

Begin geometry description. Geometry block consists of a list of units

Unit 2 contains an annular fuel pin. Model consists of mixture 23 (moderator), fuel mixture 2, more moderator, and clad—all within a hexprism filled with moderator

Unit 3 is identical to unit 2 but places mixture 3 in the fuel region

Unit 5 is identical to unit 2 but places mixture 5 in the fuel region

Unit 6 is identical to unit 2 but places mixture 6 in the fuel region

Unit 9 is a control rod guide tube, consisting of moderator within the tube, which is, in turn, set into a hexagonal region of moderator

Unit 100 is the global unit. Three nested hexprisms are specified: filled with moderator, zirc, and more moderator. An array is placed with element 7,7 centered in the unit. The outer boundary is a hexprism. A 28 by 28 grid is specified for the unit. Therefore, a 7 by 7 coarse-mesh grid is used

All units have been specified. This completes the geometry description

A single array is used in the model and is specified here. A stacked hexagonal array format is used for a 13 by 13 array. The guide tube is place in the center (unit 9); units 2, 3, 5, and 6 are placed in appropriate positions. Null units are placed in corner regions where no array elements are needed

White BCs are specified for the horizontal outer boundaries. White BCs are always used on nonhorizontal and nonvertical boundaries

Homogenization block is used to define one new mixture ID 500—composed of all the mixtures used in the model. The string identifier for the homogenized material is "fuel"

Associated with the new homogenized mixture 500 are type 3 assembly discontinuity factors. In this case, surfaced-averaged flux and currents are provided for six line segments that correspond to the outer boundary of the assembly

A one-group collapse is provided in the Collapse block

End of this NEWT case

Fig. 9.2.48: Sample 3 input listing (annotated).

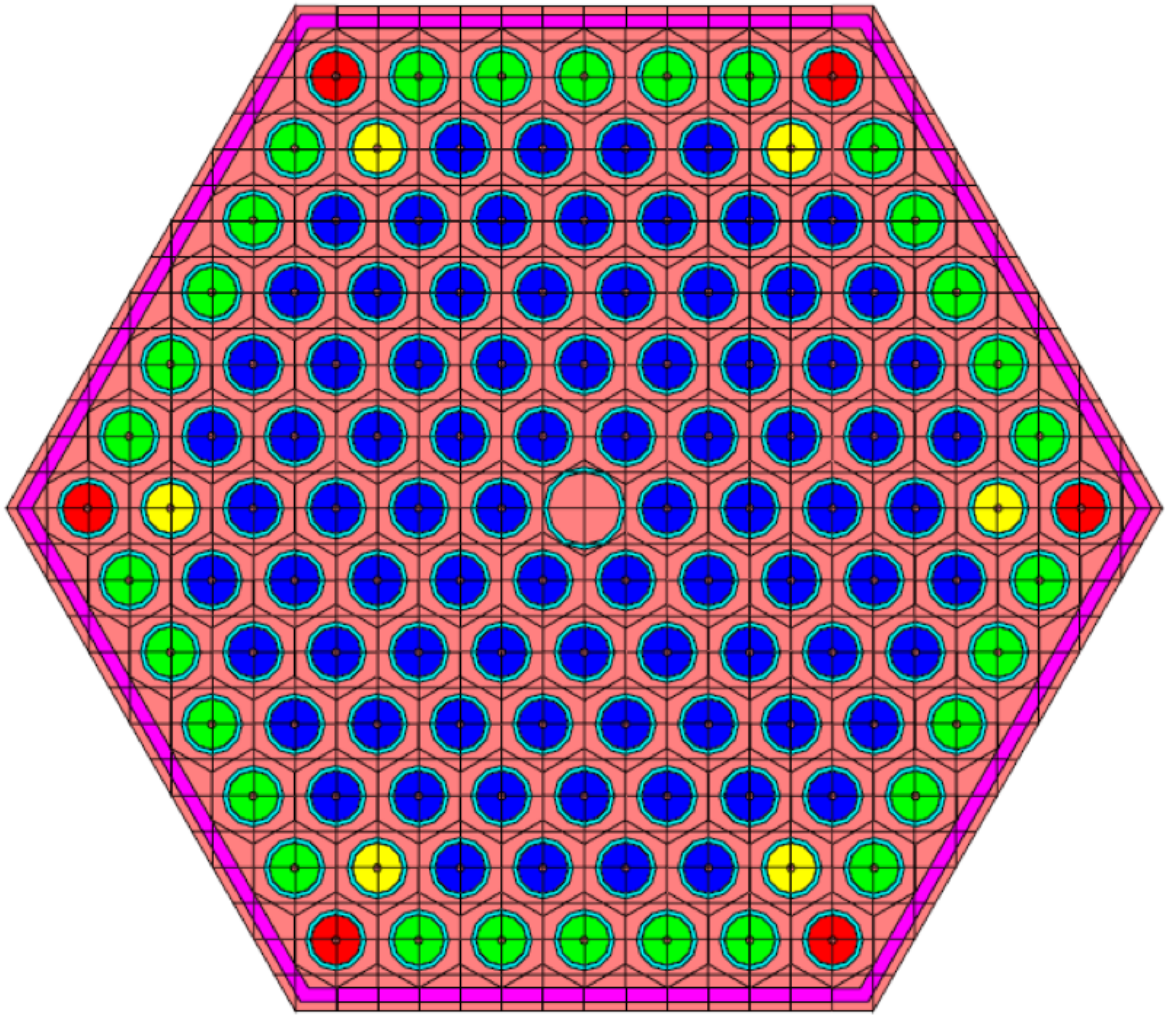


Fig. 9.2.49: Grid structure and material placement for VVER-440 model.

9.2.4.3 Sample 4

Sample 4 demonstrates the use of NEWT in modeling a larger, more complex configuration. Annotated input for this problem is given in Fig. 9.2.50. The calculation begins with the use of SCALE standard composition specifications to prepare a problem-specific weighted cross section library and mixing table for use by NEWT. In this case the T-XSEC sequence of the TRITON control module is used.

This NEWT case is used to calculate the eigenvalue of an infinite lattice of fuel assemblies. Symmetry at the assembly center is used to reduce a 15 by 15 assembly lattice to a smaller one-quarter assembly. The grid structure is illustrated in Fig. 9.2.51. A similar illustration showing media placements by color is given in Fig. 9.2.52.

This input illustrates several features of NEWT modeling capabilities. Some important features of this model are as follows.

- In this sample problem, S-6 quadrature, P-1 scattering (P-2 in the moderator), spatial convergence criteria of 0.005, and an eigenvalue convergence criteria of 0.001 are used. These are an order of a magnitude larger than the values typically used for LWR lattice calculations.

- Two sets of UO_2 cross sections are prepared in the T-XSEC calculation. These cross sections are identical with the exception of the mixture number. Since NEWT reports fluxes, reaction rates, etc., by mixture, the placement of a unique mixture at a specific location in a model allows one to determine, for example, the reaction rates at that model location. Mixture 7, placed in unit 9 in this model, occurs in only one pin location in the model. Mixture 1, placed in all other fuel rod locations, will yield reaction rates close to the average of those for all fuel in the assembly. If the flux or reaction rate was needed in each unique fuel location, a unique mixture would be needed for each location.
- The use of chords for cutting cylinders allows inclusion of one-half and one-quarter fuel cells in the quarter-assembly model. Because the fuel assembly has an odd number of rods in each dimension, use of symmetry at the assembly midplanes requires the rods to be bisected.
- In this model, local grid spacing was selected such common grid spacings occur in all cells. However, this is not a requirement. For example, a much more refined local grid could have been specified for unit 9. There is no requirement that grid lines match between different elements of an array.
- Unstructured coarse-mesh finite-difference acceleration (cmfd=2 or cmfd=yes) was employed to accelerate the convergence of the solution. For this case, 14 outer iterations were required for full spatial convergence as compared with 30 outer iterations when CMFD is disabled. The CMFD-accelerated case ran 2.5 times faster than its unaccelerated counterpart. In this sample problem, xycmfd=2 was used to define the coarse-mesh grid to have two fine-mesh cells per coarse-mesh cell.
- Two-group homogenized cross sections were generated along with single-assembly (i.e., type 1) ADFs derived from the Collapse block, ADF block, and the Homogenization block. In addition, a B1 critical spectrum search is computed after the transport calculation, which is folded into the transport solution to generated homogenized constants.

```
=t-xsec
H. B Robinson assy 05, rod N-9
v7-238
read comp
'Fuel, density homog. from 9.944 g/cc to 9.550 g/cc
```

T-XSEC sequence of TRITON
238-group ENDF/B-VII library

```
uo2 1 den=9.550 1 743  92234 0.023 92235 2.561 92236
  0.013 92238 97.403 end
'
```

UO_2 fuel specification

```
'Cladding
wtptzirc 2 6.44 4 40000 97.91 26000 0.5 50118 0.64
  50120 0.95 1 595 end
'
```

Zircaloy clad

```
'borated water, 652.5 ppm
h2o 3 den=0.7544 1 559 end
boron 3 den=0.7544 652.5e-6 559 end
'
```

Borated water moderator

```
wtptbpr 4 2.081 6 8016 53.58 11000 2.82 13027 1.758
              14000 37.63 19000 0.33  5000 3.882
              1 559 end
```

Borosilicate glass burnable poison rod

```
' air
wtptair 5 0.00129 1 7000 100  1 559.0 end
'
```

Air (low-density nitrogen)

```
' SS304, same temp as moderator
ss304 6 1 559.0 end
'
```

Stainless steel

```
' fuel sample N-9 cell materials
uo2 7 den=9.550 1 743  92234 0.023 92235 2.561
  92236 0.013 92238 97.403 end

wtptzirc 8 6.44 4 40000 97.91 26000 0.5 50118 0.64 50120 0.95 1 595 end
h2o 9 den=0.7544 1 559 end
boron 9 den=0.7544 652.5e-6 559 end
end comp
```

Second set of UO_2, zirc, and borated moderator specifications (same properties, different mixture numbers)

```
'********************
'* Cell data          *
'********************
read celldata
   latticecell squarepitch pitch=1.43 3 fueld=0.9484 1
   cladd=1.0719 2 end
   latticecell squarepitch pitch=1.43 9 fueld=0.9484 7
   cladd=1.0719 8 end
end celldata
end
```

Lattice specification for the two fuel pin cells

```
=newt
H. B Robinson assy 05, rod N-9
```

Stand-alone NEWT calculation
Title card

```
read parm
  prtflux=no prtmxsec=no prtbroad=no prtmxtab=no
  drawit=yes collapse=yes inners=2 outers=200
  epsilon=5e-3 epseigen=1e-3 sn=6
  cmfd=2 xycmfd=2 echo=yes solntype=b1 timed=yes
end parm
```

Parameter block specification as described above

```
read materials
  mix=6  pn=1  com="SS-304 - BRP clad" end
  mix=5  pn=1  com="air in BPRs" end
  mix=4  pn=1  com="borosilicate glass" end
  mix=3  pn=2  com="water" end
  mix=2  pn=1  com="cladding" end
  mix=1  pn=1  com="2.561 wt % enriched fuel " end
  mix=7  pn=1  com="rod N-9" end
end materials
```

Mixtures 1–7 are used from the cross section library and mixing table produced in the preceding T-XSEC calculation. P-1 scattering used in all mixtures except moderator, which uses P-2

```
read geom
global unit 10
cuboid 13 10.725 0.0 10.725 0.0
array 101 13  place 1 1 -0.715 -0.715
media 3 1 13
boundary 13 30 30
```

Global unit (10) consists of a cuboid (13) filled with mixture 3. An array (101) is placed such that the origin of the element in row 1, column 1, is located at (–0.715, -0.715). The contents of the array will be defined in the *read array* data block below.

```
unit 1
cuboid 13 1.43 0.0 1.43 0.0
cylinder 12 0.53595 origin x=0.715 y=0.715
cylinder 11 0.4742 origin x=0.715 y=0.715
media 3 1 13 -12
media 2 1 12 -11
media 1 1 11
boundary 13 4 4
```

Unit 1 is a nominal fuel pin cell, placed in a 1.43 by 1.43 cm cuboid with a 4 by 4 local grid.

```
'2: BPR location
unit 2
cuboid 13 1.43 0.0 1.43 0.0
cylinder 14 0.28385 origin x=0.715 y=0.715
cylinder 15 0.30035 origin x=0.715 y=0.715
cylinder 16 0.50865 origin x=0.715 y=0.715
cylinder 17 0.55755 origin x=0.715 y=0.715
media 3 1 13 -17
media 6 1 17 -16
media 4 1 16 -15
media 6 1 15 -14
media 5 1 14
boundary 13 4 4
```

Unit 2 is an annular burnable poison rod, placed in a 1.43 by 1.43 cm cuboid with a 4 by 4 local grid. Radially, the rod consists of air, steel, poison material, and steel

```
'3: full guide tube cell
unit 3
cuboid 13 1.43 0.0 1.43 0.0
cylinder 12 0.6934 origin x=0.715 y=0.715
cylinder 11 0.6502 origin x=0.715 y=0.715
media 3 1 13 -12
media 2 1 12 -11
media 3 1 11
boundary 13 4 4
```

Unit 3 is a guide tube consisting of a cylinder of moderator enclosed in a cylinder of Zircaloy, centered in a 1.43 by 1.43 cm cuboid with a 4 by 4 local grid.

```
'4: right half of a fuel rod cell
unit 4
cuboid 13 1.43 0.715 1.43 0.0
cylinder 12 0.53595 origin x=0.715 y=0.715 chord +x=0.715
cylinder 11 0.4742 origin x=0.715 y=0.715 chord +x=0.715
media 3 1 13 -12
media 2 1 12 -11
media 1 1 11
boundary 13 2 4
```

Unit 4 is the right half of a nominal fuel cell, to be placed on the left boundary of an array. The cylinder is cut with a chord and placed on the left side of a 0.715 by 1.43 cm cuboid with a 2 by 4 local grid.

```
'5: top half of a fuel rod cell
unit 5
cuboid 13 1.43 0.0 1.43 0.715
cylinder 12 0.53595 origin x=0.715 y=0.715 chord +y=0.715
cylinder 11 0.4742 origin x=0.715 y=0.715 chord +y=0.715
media 3 1 13 -12
media 2 1 12 -11
media 1 1 11
boundary 13 4 2
```

Unit 5 is the top half of a nominal fuel cell, to be placed on the bottom boundary of an array. The cylinder is cut with a chord and placed on the bottom side of a 1.43 by 0.715 cm cuboid with a 4 by 2 local grid.

```
'6: right half of a guide tube cell
unit 6
cuboid 13 1.43 0.715 1.43 0.0
cylinder 12 0.6934 origin x=0.715 y=0.715 chord +x=0.715
cylinder 11 0.6502 origin x=0.715 y=0.715 chord +x=0.715
media 3 1 13 -12
media 2 1 12 -11
media 3 1 11
boundary 13 2 4
```

Similar in construct to unit 4, unit 6 is the right half of a guide tube cell.

```
'7: top half of a guide tube cell
unit 7
cuboid 13 1.43 0.0 1.43 0.715
cylinder 12 0.6934 origin x=0.715 y=0.715 chord +y=0.715
cylinder 11 0.6502 origin x=0.715 y=0.715 chord +y=0.715
media 3 1 13 -12
media 2 1 12 -11
media 3 1 11
boundary 13 4 2
```

Similar in construct to unit 5, unit 7 is the top half of a guide tube cell

```
'8: top-right quarter of a guide tube cell
unit 8
cuboid 13 1.43 0.715 1.43 0.715
cylinder 12 0.6934 origin x=0.715 y=0.715 chord +x=0.715
  chord +y=0.715
cylinder 11 0.6502 origin x=0.715 y=0.715 chord +x=0.715
  chord +y=0.715
media 3 1 13 -12
media 2 1 12 -11
media 3 1 11
boundary 13 2 2
```

Unit 8 represents the 1/4 cell that will be placed in the lower-left corner of the array. It is a guide tube cell but reduced to 1/4 cylinders using chords. It is located within a 0.715 by 0.715 cm cuboid with a 2 by 2 local grid.

```
'9: fuel rod N-9 cell
unit 9
cuboid 13 1.43 0.0 1.43 0.0
cylinder 12 0.53595 origin x=0.715 y=0.715
cylinder 11 0.4742 origin x=0.715 y=0.715
media 3 1 13 -12
media 2 1 12 -11
media 7 1 11
boundary 13 4 4
end geom
```

Unit 9 is identical to unit 1 in structure but with a different fuel mixture specified. Local parameters for this pin will be available in output.

This completes the geometry specification.

```
read array
ara=101 nux=8 nuy=8 typ=cuboidal fill
8 5 5 5 7 5 5 5
4 1 1 1 1 1 1 1
4 1 1 1 1 2 1 1
4 1 1 3 1 1 1 1
6 1 1 1 1 1 1 1
4 9 2 1 1 2 1 1
4 1 1 1 1 1 1 1
4 1 1 1 1 1 1 1 end fill
```

Unit placement is specified for the single array used in this configuration. An 8 by 8 cuboidal (square) array is filled—starting from the bottom, moving left to right, and then bottom to top.

```
read bounds
all=refl
end bounds
```

Reflective boundary conditions are applied to all faces.

```
read collapse
199r1 39r2
end collapse
```

Two-group collapse specification

```
read homog
500 all 1 2 3 4 5 6 7 end
end homog
```

Homogenize all mixtures to form new mixture 500

```
read adf
1 500 n=10.725 e=10.725
end adfend
```

Type 1 ADFs generated on the north and east faces

This completes the input for this case.

Fig. 9.2.50: Sample 4 input listing (annotated).

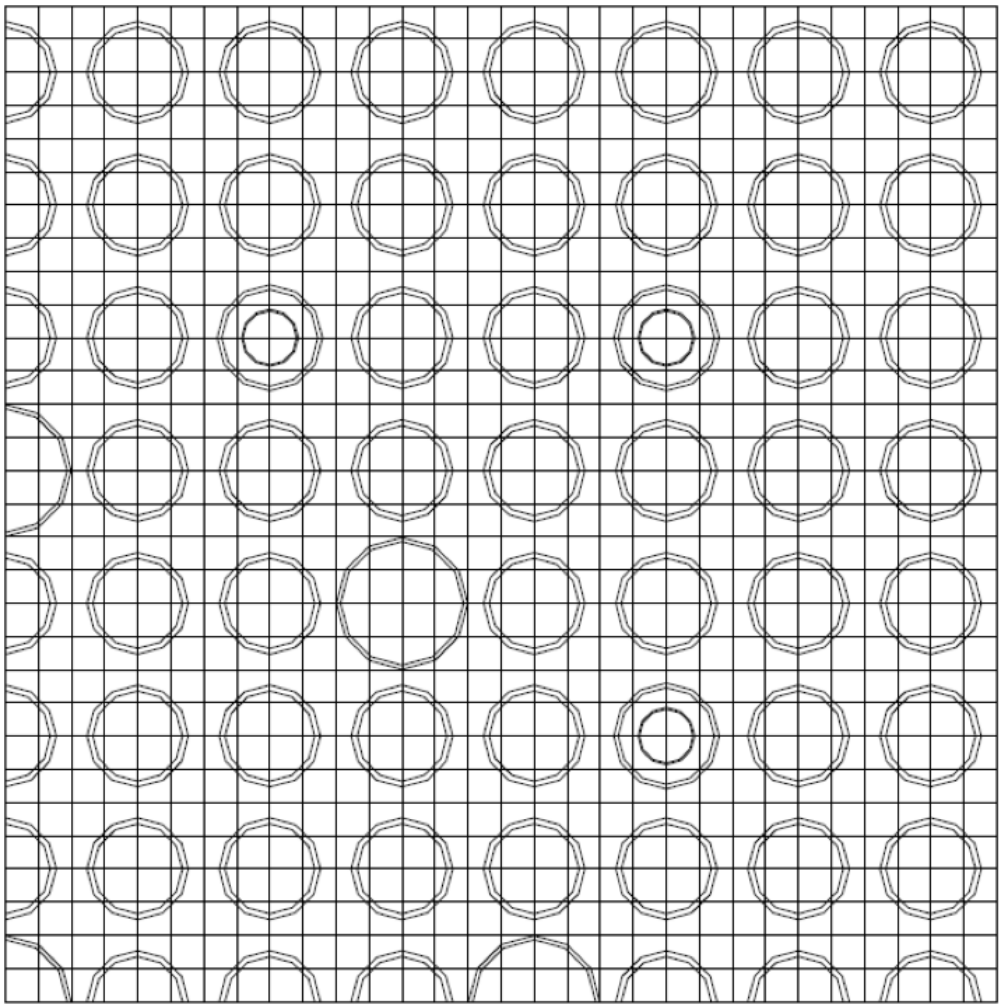


Fig. 9.2.51: Grid structure for one-quarter assembly of Sample 4.

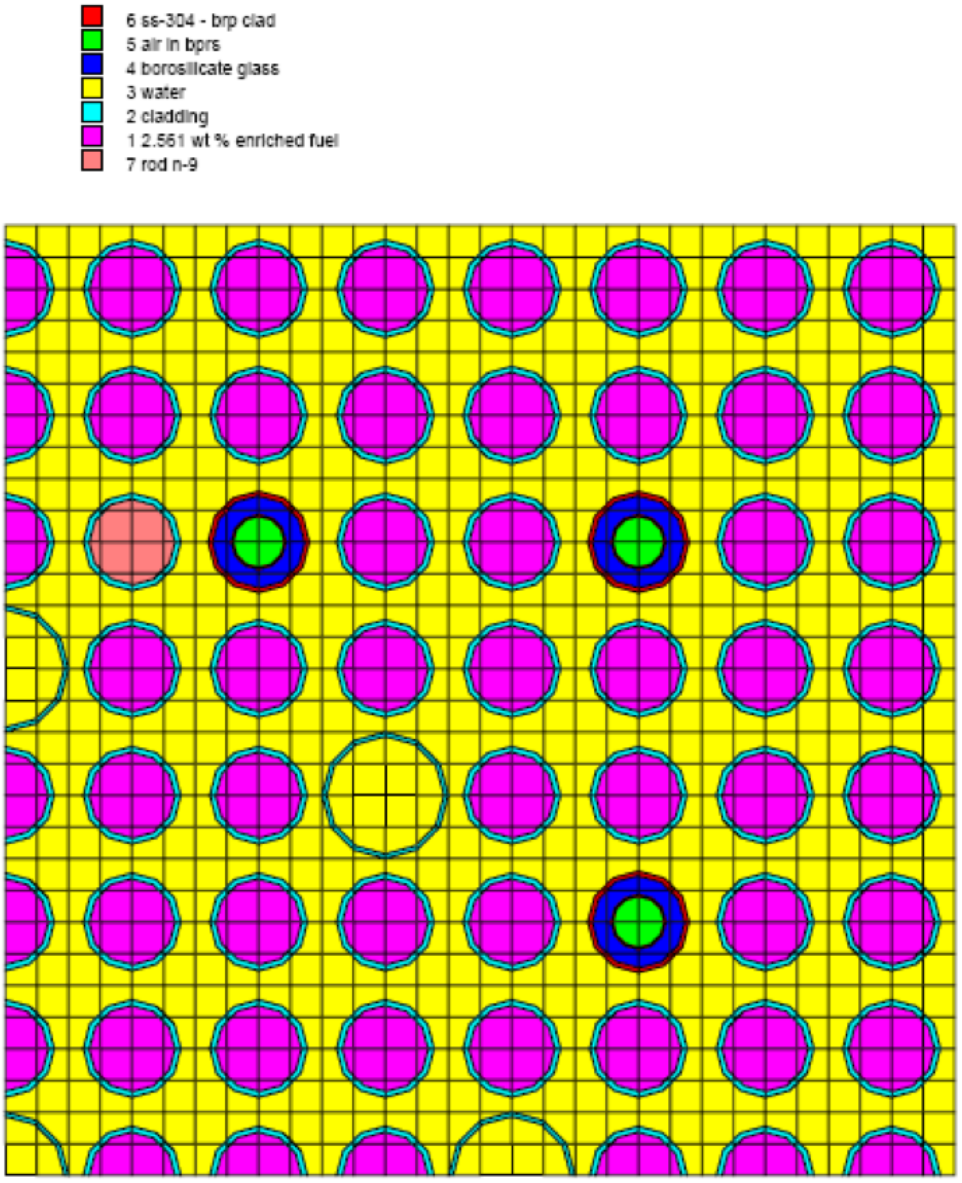


Fig. 9.2.52: Mixture placement for quarter-assembly model of Sample 4.

9.2.4.4 Sample 5

Sample 5 (Fig. 9.2.53) illustrates a calculation for a fuel assembly with a large water boundary and a vacuum boundary condition. The calculation begins with the use of SCALE standard composition specifications to prepare a problem-specific weighted cross section library and mixing table for use by NEWT.

In this model, seven UO_2 pins are adjacent to eight MOX pins, which, in turn, are adjacent to a large reflector region. The outer boundary of the reflector is vacuum. Reflection on the top and bottom boundaries makes the problem infinite in the y direction. The grid structure for this problem is illustrated in Fig. 9.2.54. This problem illustrates the use of the original CMFD acceleration scheme in NEWT (cmfd=1 or cmfd=rect). Because of the large degree of scattering within the reflector region, the problem can be relatively slow to converge. Without CMFD acceleration, 40 outer iterations are required for spatial convergence as compared

with 12 when CMFD is enabled. A total run-time speedup of ~1.4 is achieved with the CMFD acceleration scheme.

In addition to the application of CMFD, Sample 5 also illustrates the use of NEWT's reflector ADF capability. Reflector ADFs are computed along the fuel/reflector interface.

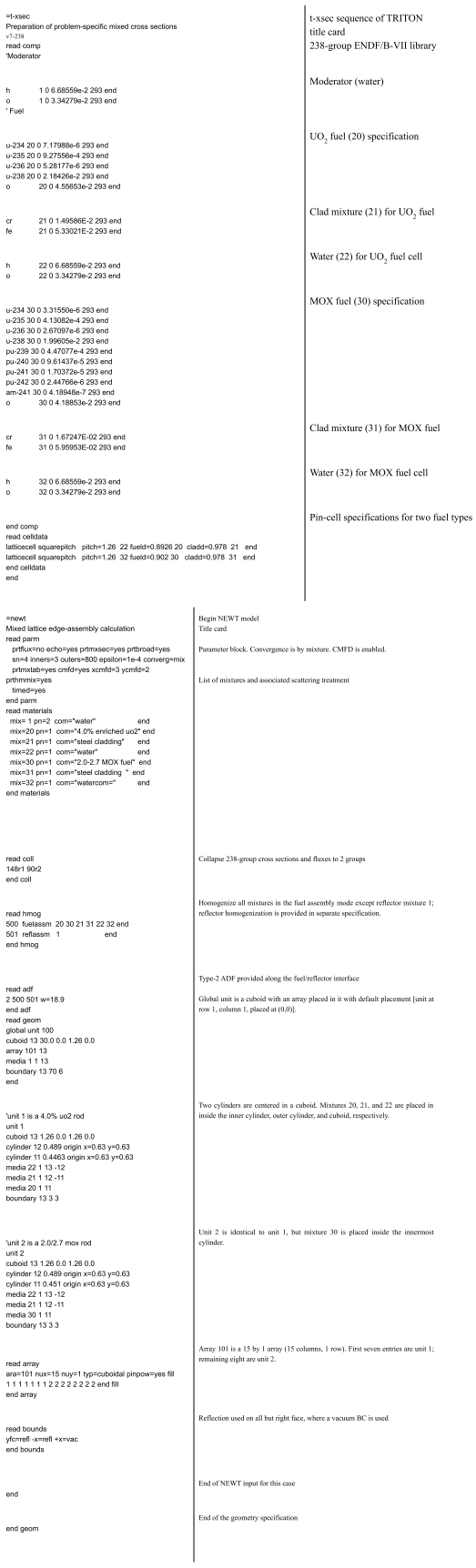


Fig. 9.2.53: Sample 5 input listing (annotated).

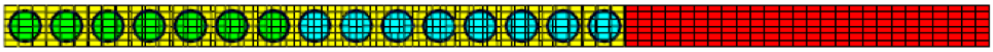


Fig. 9.2.54: Grid structure for 15-pin row of Sample 5.

9.2.5 DESCRIPTION OF OUTPUT

This section contains a brief description and explanation of NEWT output. Portions of the output will not be printed for every problem. Some output is optional, depending on user input specifications and is so noted in the description. As with any SCALE module, output begins with an input echo, module execution records with times and completion codes, and the program verification information banner page. These outputs are common to all SCALE modules and are not described here.

9.2.5.1 NEWT banner

Following the SCALE program verification information, the first section unique to NEWT output is the NEWT banner, which appears as shown in Fig. 9.2.55. The bottom of the banner gives the title of the case as given in input. The NEWT banner is printed only if the command line option -p is used to run SCALE.

```
*******************************************************************************************************
*                                                                                                     *
*               NN          NN  EEEEEEEEEEEE   WW          WW  TTTTTTTTTTT                       *
*               NNN         NN  EEEEEEEEEEEE   WW          WW  TTTTTTTTTTT                       *
*               NNNN        NN  EE             WW          WW       TT                           *
*               NN NN       NN  EE             WW          WW       TT                           *
*               NN   NN     NN  EE             WW          WW       TT                           *
*               NN    NN    NN  EEEEEEEEE      WW     W    WW       TT                           *
*               NN     NN   NN  EEEEEEEEE      WW    WWW   WW       TT                           *
*               NN      NN  NN  EE             WW   WW WW  WW       TT                           *
*               NN       NN NN  EE             WW WW    WW WW       TT                           *
*               NN        NNNN  EE             WWWW      WWWW       TT                           *
*               NN         NNN  EEEEEEEEEEEE   WWW        WWW       TT                           *
*               NN          NN  EEEEEEEEEEEE   WW          WW       TT                           *
*                                                                                                   *
*                               NEW Transport algorithm                                             *
*                               --- -                                                               *
*                                                                                                   *
*                         .                    ..............                                       *
*                         $            ...ee+**"""""~~~~"""""""**meo..                             *
*                        .$E      ..ee***"~~~~~~~~~~~~~~~~~~~~~~~~~~"**e.                         *
*                     .o*@?\$ee*#""~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"**e.                      *
*          ..e*TTTTTTTTTl@N$$$Uuzeeeeeeemmmmmmmmmmmeeeiuuu~~~~~~~~~~~~~~~~~~~~"*bo.                 *
*       .od*T!!!!!!!!!!*****B$?!!!!!!!!!!!!!!!!!!!!!!!!!!!!TT%mmuuuuuuuuuuuuuuuuuuuJP*e.          *
*      d#?!!!!!!!!!!!!!!!$@@$*QX!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!TTTTII$$$r  *
*     4E!!!!!!!!!!!!!!!!!#NWW$$MUWW@$$$$$$$$$$$$$$$WX!!!!!!UUWWP*****************"""""7$P"      *
*     "*WX!!!!!!!!!!UW*R**@$$$$$$$$$$$$$$$$$$$$$$$$$$XUU!UWP#~~~~~~~~~~~~~~~~~~~~uer*""          *
*       "*NWUUUUUUU@#*****4u"R$$$$$$$$$$$$$$$$$$$$P"IIIuUP"#*meuuuu/~~uuuuuee****"               *
*                        .oee#uP"""*********""""  d"z*#T8"       ^""""""""""""                     *
*                     o**Iue*"                   "*u""*@c.                                       *
*                   .$Ne$""                        "#*me@$                                        *
*                                                                                                   *
*******************************************************************************************************
Copyright (c) 1994-2008 Oak Ridge National Laboratory

-------------------------------------------------------------------------------------------------------------------------------

Pin-cell infinite lattice model

-------------------------------------------------------------------------------------------------------------------------------
```

Fig. 9.2.55: NEWT copyright banner page and case title.

9.2.5.2 Input summary

The next several pages of output provide a summary of input parameters. As described in Sect. 9.2.5, default parameters are used when no user specification is supplied. The input summary lists all parameters and states used in the calculation, whether user supplied or default. The following subsections describe the various blocks of output information provided in the input summary.

Control options

The Control Options block lists global control parameters that determine the type of analysis being performed. A sample Control Options page is shown in Fig. 9.2.56. Parameters are self-explanatory. More information is available in the description of the keywords in Sect. 9.2.5.2.

```
1          **********************
           **********************
           **  Control Options  **
           **********************
           **********************

 Run problem to completion/test input only         run=               yes
 Weighted cross section collapse/do not collapse   collapse=          yes
 Source acceleration on/off                        accel=             yes
 Restart from earlier case/begin new case          restart=           no
 CMFD acceleration (0/1/2/3 = no/std/ucmfd/upcmfd)  cmfd=             0
 Number of fine mesh base grid/coarse grid in X    xcmfd=                1
 Number of fine mesh base grid/coarse grid in Y    ycmfd=             1
 CMFD Multigroup solution only (no 2g)             cmfd2g=            no
 Create restart file upon completion               savrest=           yes
 Calculation type:                                     eigenvalue
 Forward/adjoint solution                          forward=           yes
```

Fig. 9.2.56: Control Options page.

Output options

The Output Options block (Fig. 9.2.57) lists selections made for output. Portions of the output listing will be printed only if the appropriate printing option was selected.

```
1          **********************
           **********************
          **  Output Options  **
           **********************
           **********************

 Print micro. cross sections for problem nuclides   prtxsec=          yes
 Print macro. cross sections for mixtures          prtmxsec=          yes
 Print balance table                               prtbalnc=          no
 Print mixing table                                prtmxtab=          yes
 Print postscript graphics file for fluxes         prtflux=           yes
 Print postscript grid/mixture geometry graphics   drawit=           yes
 Print broad-group fluxes                          prtbroad=          no
 Echo convergence tracking data to std. output     echo=              yes
```

Fig. 9.2.57: Output Options page.

Input/output unit assignments

The Input/Output (I/O) Unit Assignments block (Fig. 9.2.58) simply lists the unit numbers selected for reading or writing various data files, as appropriate for the calculation.

```
1          ***************************
           ***************************
           ** I/O Unit Assignments  **
           ***************************
           ***************************

 Unit no. for mixing table                        mixtab=             92
 Unit no. of working library                      xnlib=              4
 Unit no. for weighted broad-group library        wtdlib=             30
 Unit no. for homogenized macroscopic x-sect. lib.  hmoglib=          13
```

Fig. 9.2.58: Input/Output Unit Assignments page.

Convergence control parameters

The Convergence Control block (Fig. 9.2.59) summarizes all parameters used to control spatial, angular, and eigenvalue convergence for the iterative phases of the solution process.

```
Quadrature order                                  sn=                 6
 Maximum no. of inner itrs./outer/energy group    inners=             5
 Maximum no. of outer energy groups               outers=           250
 Performing thermal group subouter iterations     therm=               yes
 Maximum no. of thermal group subouter itrs./outer  suboutrs=         2
 Convergence criterion on inner iterations        epsinner= 1.00E-04
 Convergence criterion on subouter iterations     epsthrm=  1.00E-04
 Convergence criterion on outer iterations        epsouter= 1.00E-04
 Convergence criterion on eigenvalue              epseigen= 1.00E-04
 First guess for eigenvalue (k-eff)               kguess=             1.000
```

Fig. 9.2.59: Convergence Control Parameters page.

Pin-power edit requests

If pin-power edits are requested for one or more arrays, a listing is provided of the arrays for which this request was made (Fig. 9.2.60).

```
1          ********************************
           ********************************
           **  Pin Power Specifications  **
           ********************************
           ********************************

  Pin power edits requested for units of following array(s):
           Array no.   101 (   8,   8), type = cuboidal
```

Fig. 9.2.60: Pin-power edit request summary.

Geometry specifications

The Geometry Specifications block (Fig. 9.2.61) lists parameters associated with the geometric model specified by the user. The first section lists the characteristics of the global unit. This is followed by a listing of the four boundary conditions. Finally, the last section in this block lists all bodies specified for the model. The appearance and contents of this section of input depend on the nature of the input model.

```
1         ****************************
          ****************************
          ** Geometry Specifications **
          ****************************
          ****************************

          geomtype=   1
 Left side x-coordinate                          xmin=  0.00000E+00
 Right side x-coordinate                         xmax=  7.36600E-01
 Bottom side y-coordinate                        ymin=  0.00000E+00
 Top side y-coordinate                           ymax=  7.36600E-01
 Number of subdivisions in x direction           nx=              4
 Number of subdivisions in y direction           ny=              4

 Boundary Conditions (0/1/2/3=vacuum/reflective/white/periodic):
 Left (x=xmin)  minxbc=       1         Right (x=xmax) maxxbc=        1
 Bottom (y=ymin)minybc=       1         Top (y=ymax)   maxybc=        1

  Body Characteristics.  Rotation angle in degrees counterclockwise.
  Size parameters. cyl: A=radius, B=no. of sides. cub: A=width, B=height. hex: A(B)=major(minor) radius.
  Body  Input  Zone Subdiv    --Center--          Size Parameters   Rotation   Chord
  Type  Unit   Number  nx  ny           x         y         A         B         Angle     Information
  ----  -----  -----  --- ---   ------   ------   -----     ------   --------   -----------
 cub      10        1         4   4    0.368     0.368     0.737     0.737     0.000
 cyl      10        2         0   0     0.737     0.737     0.559   12.000      0.000     -x=  0.737          -y=  0.737
 cyl      10        3         0   0     0.737     0.737     0.478   12.000      0.000     -x=  0.737          -y=  0.737
```

Fig. 9.2.61: Geometry Specifications page.

Homogenization region specifications

The Homogenization Region Specifications block (Fig. 9.2.62) summarizes all sets of homogenized cross sections requested in user input.

```
1           ****************************************
            ****************************************
            ** Homogenization Region Specifications **
            ****************************************
            ****************************************

assm        contains the following mixtures:
            1         2         3         4         5         6         7
```

Fig. 9.2.62: Homogenization Region Specifications page.

Material specifications

The Material Specification block (Fig. 9.2.63) lists the NEWT material number, counting in the order read in; the SCALE mixture number; and the P_n order assigned for that mixture.

```
1           ****************************
            ****************************
            ** Material Specifications **
            ****************************
            ****************************

 Mat'l No.   SCALE ID   Pn order   External source ID   Description (if any)
           1         6          1                    0        ss-304 - brp clad
           2         5          1                    0        air in bprs
           3         4          1                    0        borosilicate glass
           4         3          2                    0        water
           5         2          1                    0        cladding
           6         1          1                    0        2.561 wt % enriched fuel
           7         7          1                    0        rod n-9
```

Fig. 9.2.63: Material Specifications page.

Derived parameters

The Derived Parameters block (Fig. 9.2.64) lists values not specifically input but derived from other sources of input. Some of this information comes from the cross section library, some from the model geometry, and some from the S_n and P_n values specified.

```
1          ************************
           ************************
           ** Derived Parameters **
            ************************
           ************************

 Number of energy groups                          numen=            44
 Number of broad groups in collapsed set          nbroad=           2
 Number of neutron groups                         igm=              44
 Number of gamma groups                           ipm=              0
 First thermal energy group                       iftg=             23
 Mixing table length                              nmix=             34
 Number of mixtures created                       nummat=           7
 Number of computational cells                    numcells=         2274
 Number of computational surfaces (sides)         numlines=         5586
 Average number of sides/cell                     avnosides=        4.83
 Minimum number of sides/cell                     mnnosides=        4
 Maximum number of sides/cell                     mxnosides=        8
 Number of angles in quadrature set               ndir=             24
 Maximum order of Legendre scattering               maxpn=          2
 Number of flux moments                           nmom              6
```

Fig. 9.2.64: Derived Parameters page.

Energy group structure listing

The Energy Group Structures block (Fig. 9.2.65) lists the energy and lethargy boundaries found in the cross section library. If a broad-group collapse was requested, the boundaries of the broad-group library that will be produced are also identified. This example shows the structure of the SCALE 44GROUPNDF5 library and 2-group fast/thermal collapse structure. The final entry (group 45, broad group 3) indicates the lower bound of the previous energy group.

```
1       ****************************
        ****************************
        ** Energy Group Structures **
        ****************************
        ****************************

 Energy   Energy           Lethargy   Broad              Energy   Energy     Lethargy
Broad

 Group  Boundaries   Boundaries  Group          Group  Boundaries   Boundaries  Group
  No.   (eV)                No.                No.       (eV)                No.
    1   2.00000E+07 -6.93147E-01       1                 23   3.00000E+00  1.50195E+01 2
    2   8.18730E+06  2.00001E-01                  24   1.77000E+00  1.55471E+01
    3   6.43400E+06  4.40989E-01                  25   1.00000E+00  1.61181E+01
    4   4.80000E+06  7.33969E-01                  26   6.25000E-01  1.65881E+01
    5   3.00000E+06  1.20397E+00                  27   4.00000E-01  1.70344E+01
    6   2.47900E+06  1.39473E+00                  28   3.75000E-01  1.70989E+01
    7   2.35400E+06  1.44647E+00                  29   3.50000E-01  1.71679E+01
    8   1.85000E+06  1.68740E+00                  30   3.25000E-01  1.72420E+01
    9   1.40000E+06  1.96611E+00                  31   2.75000E-01  1.74091E+01
   10   9.00000E+05  2.40795E+00                  32   2.50000E-01  1.75044E+01
   11   4.00000E+05  3.21888E+00                  33   2.25000E-01  1.76098E+01
   12   1.00000E+05  4.60517E+00                  34   2.00000E-01  1.77275E+01
   13   2.50000E+04  5.99146E+00                  35   1.50000E-01  1.80152E+01
   14   1.70000E+04  6.37713E+00                  36   1.00000E-01  1.84207E+01
   15   3.00000E+03  8.11173E+00                  37   7.00000E-02  1.87774E+01
   16   5.50000E+02  9.80818E+00                  38   5.00000E-02  1.91138E+01
   17   1.00000E+02  1.15129E+01                  39   4.00000E-02  1.93370E+01
   18   3.00000E+01  1.27169E+01                  40   3.00000E-02  1.96247E+01
   19   1.00000E+01  1.38155E+01                  41   2.53000E-02  1.97950E+01
   20   8.09999E+00  1.40262E+01                  42   1.00000E-02  2.07233E+01
   21   6.00000E+00  1.43263E+01                  43   7.50000E-03  2.10109E+01
   22   4.75000E+00  1.45600E+01                  44   3.00000E-03  2.19272E+01
                                                   45   1.00000E-05  2.76310E+01 3
```

Fig. 9.2.65: Energy Group Structure Listing page.

Quadrature parameters

The Quadrature Parameters block (Fig. 9.2.66) lists the first-quadrant angles and weights used for the specified order of quadrature. The same angles and weights are applied in the other three quadrants; however, the signs of the angles vary with the quadrant. Also listed are the P_n moments associated with the maximum P_n scattering order requested in all materials. Of course, only a subset of these moments applies to the lower-order P_n assignments.

```
1         ***************************
          ***************************
          ** Quadrature Parameters **
          ***************************
          ***************************

Quadrature angles and weights for first quadrant:
(  3 angles/quadrant for S- 4 quadrature.)

Angle No.           Mu        Eta       Weight
          1         3.01638780D-01               3.01638780D-01
          8.33333325D-02
          2         9.04449050D-01               3.01638780D-01
          8.33333325D-02
          3         3.01638780D-01               9.04449050D-01
          8.33333325D-02

Scattering constants for first quadrant
(10 moments for p-3 scattering)
Angle     ------------------------------------          Moment           No.
-------------------------------------
 No.      0         1         2         3         4         5               6
          7
   1      1.00000E+0     3.01639E-1     3.01639E-1   -3.63521E-1     1.57592E-1
-6.29637E-1 -3.83846E-1 -1.00683E-1
   2      1.00000E+0     9.04449E-1     3.01639E-1    7.27042E-1     4.72533E-1
0.00000E+0  4.92988E-1  5.70796E-1
   3      1.00000E+0     3.01639E-1     9.04449E-1   -3.63521E-1     4.72533E-1
6.29637E-1 -3.83846E-1 -3.01892E-1

Angle     - Moment No. -
 No.      8         9
   1   -4.24681E-1 -5.63520E-1
   2      0.00000E+0 -4.33942E-2
   3      4.24681E-1  3.89741E-1
```

Fig. 9.2.66: Quadrature Parameters page.

Mixture volumes listing

The Mixture Volumes block (Fig. 9.2.67) provides a summary of the volume and volume fraction of each mixture in the problem, together with the total volume. This block can be used as a simple check of the input model by ensuring that the calculated volumes of mixtures used for a given problem match the expected volumes or volume fractions.

```
1              *************************
       *************************
               **** Mixture Volumes *****
               *************************
               *************************

 Mixture       Volume              Volume
   ID          (cc/unit length)    Fraction
     3         1.01436E+00         5.88655E-01
     2         1.80720E-01         1.04876E-01
     1         5.28102E-01         3.06469E-01
Total          1.72318E+00         1.00000E+00
```

Fig. 9.2.67: Mixture Volumes page.

Mixing table listing

The Mixing Table block summarizes the input mixing table, whether user supplied or read from a SCALE-generated file. Number densities are in units of atoms per barn-centimeter. Although optional, the mixing table is printed by default. This default setting can be disabled by specifying *prtmxtab*=no in the Parameter block. A sample mixing table is shown in Fig. 9.2.68.

```
1      **********************
       **********************
       **   Mixing Table   **
       **********************
       **********************

Entry    Mixture    Isotope    Number Density
   1   1           1092234    2.59520E-07
   2   1           1094238    4.66100E-05
   3   1           1094241    1.74910E-04
   4   1           1094242    1.32010E-04
   5       1       1008016    4.65860E-02
   6   3           3008016    2.42130E-02
   7   1           1094240    4.82550E-04
   8   1           1094239    1.01560E-03
   9   1           1092235    5.42870E-05
  10   1           1092238    2.13870E-02
  11   2            2040090   3.86570E-02
  12   2           2026000    1.33450E-04
  13   2           2024000    6.82540E-05
  14   3           3001001    4.84140E-02
  15   3           3005010    4.78960E-06
  16   3           3005011    1.94240E-05
```

Fig. 9.2.68: Mixing Table Listing page.

Nuclide cross sections

The Nuclide Cross Section block is optional and is printed only when *prtxsec*=yes is specified in the Parameter Block. The volume of output generated is quite extensive, especially when a very fine group library is used and/or a large number of nuclides are included in the mixing table. The nuclide data are taken directly from the working library used for the calculation. A sample showing a partial listing for a single nuclide is shown in Fig. 9.2.69.

Following the block header, nuclide data are listed for all nuclides. For each record, the same format is used. Nuclide data begin with a listing of nuclide header information. This is followed by a listing of the 1-D cross sections that are important in NEWT calculations. The sample below shows a partial listing of the 1-D cross sections. Following the 1-D cross section listing is the scattering matrix for the nuclide. This abbreviated listing shows a portion of the P_0 matrix for this nuclide; however, in a full listing, all higher-order elements are printed as well.

As was indicated in the input description, specification of *prtxsec*=1d can be used to obtain header and 1D cross section data only, skipping the printing of scattering matrices.

```
1         ************************************************...
          ************************************************
          ** Nuclide cross sections from working library **
          ************************************************
          ************************************************

=============================================================================================================================

Nuclide description: 5b 100las1 evaldec76 g.hale l.stewart mod1 12/11/92 free gas
Nuclide ID No.:  3005010
Nuclide symbol: b-10
Length of P-0 total scattering matrix:  834
Order of expansion of total scattering matrix:  5
Max length of any P-l array in total transfer matrix:       834
Number of 1-D Neutron processes: 25
Number of 1-D Gamma-ray processes:  0
A - neutron equiv. mass number:   9.927
ZA - 1000*Z + A:    5010.
Energy per fission (Joules): 0.0000E+00
Energy per capture (Joules): 0.0000E+00
ENDF MAT for fast neutron data: ****
ENDF MAT for thermal neutron data:      0
ENDF MAT for gamma data:      0
ENDF MAT for gamma production data:     0
Number of records in this set:   8

          ***** 1-D Microscopic Neutron Cross Sections *****

 Energy   Total               Absorption          Fission             n-2n      Nu - Average        Chi - Fission       Group Flux
 Group   Cross Section   Cross Section   Cross Section   Cross Section  Neutrons/Fission   Distribution   Weight Integral
*************************************************************************************************************************

          1         1.46503E+00         2.58537E-01         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00
7.91211E-01
          2         1.54555E+00         3.06249E-01         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00
9.75374E-01
          3         1.52524E+00         4.13432E-01         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00
2.97757E+00
          4         1.77619E+00         3.36016E-01         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00
1.05424E+01
          5         2.27853E+00         3.58337E-01         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00
6.06130E+00
          6         1.99554E+00         3.20463E-01         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00
1.75427E+00
          7         2.03572E+00         4.59178E-01         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00
8.46044E+00

          …

   42     6.52952E+03         6.52692E+03         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00     1.25445E-01
   43     8.38683E+03         8.38388E+03         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00         1.53596E-01
   44     1.46697E+04         1.46654E+04         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00         3.35559E-02

 p0 matrix from working file id # 3005010 (5b 100las1 evaldec76 g.hale l.stewart mod1 12/11/92 free gas                )

From   to group   1 to group   2        to group   3        to group   4        to group   5        to group   6        to group   7        to   group
8
Group
  8       7.44537E-01
  7       6.51747E-01         7.65186E-01
  6       2.56093E-01         8.62693E-01         5.33248E-01
  5       7.77653E-01         1.95839E-01         7.09427E-01         2.18085E-01
  4       8.88258E-01         3.03588E-01         6.52317E-02         1.47352E-01         1.34825E-02
  3    5.76023E-01  4.52666E-01         1.89246E-02         4.46762E-03         1.64959E-02         1.02690E-02
  2       6.16145E-01         4.14460E-01         7.71776E-02         1.47449E-02         3.68320E-03         1.73554E-02         2.09442E-02
  1       6.36997E-01         2.74273E-01         8.64998E-02         6.51673E-02         2.89232E-02         7.61749E-03         3.47580E-02
          2.96962E-02

From   to group   9 to group  10        to group  11        to group  12        to group  13        to group  14        to group  15        to   group
16
Group
 16       1.81129E+00
 15           1.79943E+00     2.21000E-01
 14       1.83148E+00         2.12550E-01

  …

From   to group  41 to group  42        to group  43        to group  44
Group
 44       1.45883E+00         3.94789E-01         8.82942E-01         5.52638E-01
 43       1.08725E+00         2.84488E-01     5.38606E-01   1.84116E-01
 42       1.04868E+00         2.52445E-01         3.30365E-01         9.54113E-02

…

 28       9.23008E-06
 27       3.47458E-06
 26       2.96893E-08

1.35621E+00         -1.35202E-02        3.88089E-02         -1.22485E-01        2.04772E-03
```

Fig. 9.2.69: Partial listing of Nuclide Cross section data pages.

Mixture cross sections

The Mixture Cross Section block provides mixed macroscopic cross sections for each mixture provided in the input mixing table. The block is also optional and is printed only when *prtmxsec*=yes is specified in the Parameter Block. Although the volume of output generated is not as extensive as that of the nuclide listings, the mixture cross section print can be voluminous, especially when a very fine group library is used and/or a large number of mixtures are included in the mixing table. A sample showing a partial listing for a single mixture is shown in Fig. 9.2.70.

Following the block header, information is provided for each mixture using the same format. Mixture data begin with a listing of general mixture information, including the mixing table for that mixture. This is followed by a listing of 1-D cross sections important in NEWT calculations. The sample below shows a partial listing of the 1-D mixed macroscopic cross sections. Following the 1-D cross section listing is the scattering matrix for the nuclide for all moments requested for the mixture. This abbreviated listing shows a portion of the P_0 matrix for this nuclide; however, in a full listing, all higher-order elements are printed as well if greater than P_0 scattering was requested.

As was indicated in the input description, specification of *prtmxsec*=1d can be used to skip the printing of scattering matrices.

```
1         ****************************
          ****************************
          ** Mixture cross sections **
          ****************************
          ****************************

 Mixture number   1 of   3
 Order of expansion of total scattering matrix: 3
 Mixture components:

 Entry   Isotope   Number Density   Symbol   Description
    1      1092234   0.25952E-06         u-234     92U 234 BNL HEDL + EVALJUL78 DIVADEENAM MANN      MOD3 01/10/91
    2      1094238   0.46610E-04         pu-238    94PU238 HEDL AI + EVALAPR78 MANN SCHENTER A       MOD3 01/18/91
    3      1094241   0.17491E-03         pu-241   AMPX MASTER FILE FOR ENDF MAT 1381  ***  PU-241 ***
    4      1094242   0.13201E-03         pu-242    94PU242 HEDL SRL + EVALOCT78 MANN BENJAMIN M      MOD2 01/21/91
    5      1008016   0.46586E-01         o-16      8O 16 from version 6 evaluation
    6      1094240   0.48255E-03         pu-240    94pu240 ornl evalapr77 l.w. westonmod3 12/12/88
    7      1094239   0.10156E-02         pu-239    94pu239 lanl jun83 e.arthur p.you mod2 02/28/89
    8      1092235   0.54287E-04         u-235     92u 235 bnl evalapr77 m.r.bhat          mod3 02/28/89
    9      1092238   0.21387E-01         u-238     92U 238 ANL+ EVALJUN77 E.PENNINGTON A.            MOD3 02/13/92

          ***** Mixed 1-D Macroscopic Neutron Cross Sections *****
  Energy   Total               Absorption          Fission             n-2n      Nu*Fission          Chi - Fission       Diffusion
  Group    Cross Section   Cross Section   Cross Section   Cross Section   Cross Section   Distribution        Coefficient
 ********************************************************************************************************************************

          1         7.97199E-02         5.09597E-03         0.00000E+00         8.59977E-09         0.00000E+00         0.00000E+00         4.93263E+00
          2         8.71285E-02         3.92736E-03         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00         4.37968E+00
          3         1.06414E-01         1.94917E-03         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00         3.77133E+00
          …

     43     4.87030E+00         7.52840E-02         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00         7.29428E-02
     44     7.40122E+00         1.31687E-01         0.00000E+00         0.00000E+00         0.00000E+00         0.00000E+00         4.67696E-02

 Mixed macroscopic p0 for mixture   1

From   to group   1   to group   2        to group   3        to group   4        to group   5        to group   6        to group   7        to group   8
Group
  8         4.83014E-02
  7         3.09818E-02         4.95922E-02
  6         5.13383E-03         3.96657E-02         2.35971E-02
  5         2.20307E-02         1.10327E-02         3.10043E-02         1.95595E-02
  4         5.90644E-02         2.64571E-02         3.88119E-03         1.35671E-02         1.21026E-02
  3         2.67193E-02         3.63730E-02         7.18501E-03         1.72385E-03         6.95049E-03         6.20564E-03
  2         1.49526E-02         2.33428E-02         1.54516E-02         4.47217E-03         1.07300E-03         4.33270E-03         4.12880E-03
  1         1.70253E-02         1.54926E-02         8.79257E-03         1.04831E-02         3.61363E-03         9.76821E-04         4.52526E-03         4.43305E-03
From   to group   9   to group  10        to group  11        to group  12        to group  13        to group  14        to group  15        to group  16
Group
 16         6.07201E-01
 15         5.98374E-01         3.91600E-01
 14         5.83338E-01         3.69956E-01         6.67880E-02
 13         2.11470E-01         6.27675E-01         1.04688E-01         1.92283E-02
 12         4.24856E-01         1.35256E-01         2.24154E-01         3.92269E-02         7.20492E-03
 11         2.96026E-01         1.87710E-01         1.94162E-02         3.39783E-02         5.94620E-03         1.09216E-03
 10         1.74082E-01         1.49390E-01         3.32181E-02         3.54327E-03         6.20072E-03         1.08513E-03         1.99309E-04
  9         1.07007E-01         1.19184E-01         5.20151E-02         1.30038E-02         1.38707E-03         2.42737E-03         4.24790E-04         7.80226E-05
  8         7.10411E-02         5.00580E-02         3.00348E-02         7.50869E-03         8.00925E-04         1.40162E-03         2.45285E-04         4.50521E-05
  7         3.32477E-02         3.32455E-02         1.99473E-02         4.98682E-03         5.31925E-04         9.30874E-04         1.62903E-04         2.99209E-05
  6         2.61944E-02         2.61943E-02         1.57166E-02         3.92915E-03         4.19110E-04         7.33442E-04         1.28353E-04         2.35750E-05
  5         2.17299E-02         2.17300E-02         1.30380E-02         3.25948E-03         3.47678E-04         6.08436E-04         1.06476E-04         1.95569E-05
  4         1.34473E-02         1.34473E-02         8.06828E-03         2.01707E-03         2.15154E-04         3.76519E-04         6.58910E-05         1.21024E-05
  3         6.89527E-03         6.89546E-03     4.13729E-03   1.03431E-03         1.10325E-04         1.93069E-04         3.37873E-05         6.20605E-06
  2         5.43708E-03         5.84780E-03         3.23595E-03         7.00922E-04         7.19486E-05         1.25249E-04         2.19640E-05         4.13445E-06
  1         3.99075E-03         3.15636E-03         1.61363E-03         3.90313E-04         4.14581E-05         7.25221E-05         1.27056E-05         2.34441E-06
```

Fig. 9.2.70: Partial listing of Mixture Cross section data pages.

9.2.5.3 Iteration history

The next portion of NEWT output lists the iteration convergence history for the iterative solution (Fig. 9.2.71). This information can be used to track and understand the performance of the outer loop of the iterative solution. The first column provides the outer iteration count. The second column lists the system eigenvalue after each outer iteration. The third column lists the change in the eigenvalue from the last outer iteration; this is one of the parameters tested for convergence. The fourth column, "Max Flux Delta," gives the maximum change in cell flux for all cells and all energy groups; this is also used as a convergence test. The next column lists the cell number and energy group corresponding to the maximum flux change in this iteration. The next two columns list the same flux information for mixtures with fissionable nuclides. This can be used to track spatial convergence in fuel when convergence is slowed by significant scattering outside the fuel. Finally, the last column provides information on the convergence of inners in each outer iteration. Inner iterations do not need to converge within early outer iterations, but final convergence will not be achieved until all inners are converged. The maximum number of inner iterations per energy group is set by the *inners*= parameter in the

parameter input block. After convergence is achieved, the table is terminated by printing the final version of k_{eff}.

If the parameter keyword *timed=* is set to *yes*, four additional columns are introduced that give timing information on the solution process, listing real ("wall clock") time, elapsed CPU time since beginning the iteration process, elapsed CPU time per outer iteration, and an estimate of the fractional CPU usage during each outer. Fig. 9.2.72 illustrates the form of output produced when *timed=yes* is input. Additionally, a supplementary edit follows the iteration edit when *timed=yes*, giving information on average time per transport sweep (outer iteration) within different components of the solution. This edit is especially useful when coarse-mesh finite-difference acceleration is used, to assess the overhead of the CMFD accelerator.

```
================================================================================
  Outer iteration sweep begins.

 Outer  Eigen-  Eigenvalue Max Flux  Max Flux         Max Fuel  Max Fuel  Inners
 It. #   value          Delta     Delta    Location(r,g)   Delta    Location(r,g)
   Cnvrged
 - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   1  1.0000   0.000E+00 3.212E+04 (       5,238)   3.212E+04 (          5,238)
            F
   2  0.8532   1.721E-01 2.333E+01 (       5,237)   2.333E+01 (          5,237)
            F
   3  1.2598   3.227E-01 1.354E+00 (       5,230)      1.354E+00  (        5,230)
            F
   4  1.2860   2.041E-02 3.842E-01 (       5,226)   3.842E-01 (          5,226)
            F
   5  1.2401   3.702E-02 1.867E-01 (   10,  1)   1.442E-01 (   5,224)    F
   6  1.1992   3.408E-02 1.264E-01 (   10,  1)   8.883E-02 (   6,  1)    F
   7  1.1755   2.019E-02 8.365E-02 (   10,  1)   5.741E-02 (   6,  1)    F
   8  1.1634   1.038E-02 5.479E-02 (   10,  1)   3.727E-02 (   6,  1)    F
   9  1.1576   5.020E-03 3.573E-02 (   10,  1)   2.427E-02 (   6,  1)    F
  10  1.1548   2.459E-03 2.325E-02 (   10,  1)   1.579E-02 (   6,  1)    F
  11  1.1532   1.400E-03 1.509E-02 (   10,  1)   1.023E-02 (   6,  1)    F
  12  1.1522   8.313E-04 9.775E-03 (   10,  1)   6.617E-03 (   6,  1)    F
  13  1.1516   5.150E-04 6.331E-03 (       1,   1)     4.285E-03  (      6,   1)
            F
  14  1.1510   5.393E-04 4.076E-03 (       1,  1)   2.734E-03 (          6,   1)
            F
  15  1.1507   2.764E-04 2.634E-03 (       1,  1)   1.774E-03 (          6,   1)
            F
  16  1.1506   7.381E-05 1.723E-03 (       1,  1)   1.177E-03 (          6,   1)
            F
  17  1.1505   4.181E-05 7.749E-04 (       1,  1)   5.322E-04 (          6,   1)
            T
  18  1.1505   1.044E-05 9.475E-05 (       1,  1)   7.133E-05 (          7,  93)
            T
  19  1.1505   1.326E-05 9.121E-05 (       1,  1)   7.114E-05 (          5,  89)
            T
 k-eff =  1.150538E+0
```

Fig. 9.2.71: Nominal iteration history output.

```
================================================================================================================================
   Outer iteration sweep begins 12/ 04/ 2003 at 10:39:33

 Outer   Eigenvalue Eigenvalue Max Flux    Max Flux         Max Fuel    Max Fuel           Wall    Elapsed   Iteration  CPU    Inners
Iter. #            Delta      Delta    Location(r,g)    Delta    Location(r,g) Clock    CPU Time    CPU Time Usage Converged
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
         1        1.0000   0.000E+00 2.947E+02 (  677, 44)   2.863E+01 ( 1709, 44) 10:40:26   53.0 s        53.0 s 100.0%       F
         2        1.0664   6.224E-02 3.202E+00 ( 1890, 43)   2.000E+00 ( 1712, 43) 10:41:14  100.0 s        47.0 s  98.0%       F
         3        1.1495   7.235E-02 5.127E-01 ( 1903, 42)   4.685E-01 ( 1713, 42) 10:42:01  146.3 s        46.2 s  98.4%       F
         4        1.1474   1.890E-03 1.311E-01 ( 1902, 40)   1.217E-01 ( 1713, 40) 10:42:36  180.9 s        34.6 s  98.9%       T
         5        1.1364   9.617E-03 5.794E-02 ( 1902, 40)   5.497E-02 ( 1713, 40) 10:43:08  212.7 s        31.8 s  99.3%       F
         6        1.1105   2.335E-02 3.356E-02 ( 1902, 38)   3.247E-02 ( 1713, 38) 10:43:49  252.8 s        40.1 s  97.8%       F
         7        1.0931   1.589E-02 2.155E-02 ( 1905, 38)   2.113E-02 ( 1713, 38) 10:44:29  291.3 s        38.5 s  96.3%       F
         8        1.0798   1.232E-02 1.492E-02 (  580, 40)   1.477E-02 ( 1732, 40) 10:45:30  329.0 s        37.7 s  61.8%       F
         9        1.0704   8.833E-03 1.067E-02 ( 1747, 40)   1.067E-02 ( 1736, 40) 10:46:43  364.7 s        35.8 s  49.0%       F
         10       1.0639   6.064E-03 7.814E-03 ( 1918, 40)   7.814E-03 ( 1918, 40) 10:47:52  397.9 s        33.1 s  48.0%       F
         11       1.0596   4.095E-03 6.072E-03 ( 1941, 40)   6.072E-03 ( 1941, 40) 10:48:45  423.4 s        25.5 s  48.2%       F
         12       1.0559   3.467E-03 4.854E-03 (   19, 41)   4.780E-03 ( 1942, 40) 10:49:36  447.1 s        23.8 s  46.6%       T
         13       1.0533   2.517E-03 3.947E-03 (   59, 40)   3.759E-03 ( 1946, 38) 10:50:20  467.7 s        20.5 s  46.7%       T
         14       1.0513   1.897E-03 6.219E-03 (   23, 35)   2.935E-03 ( 2306, 40) 10:51:00  484.9 s        17.2 s  43.1%       T
         15       1.0500   1.245E-03 5.003E-03 (   26, 36)   2.790E-03 ( 2307, 35) 10:51:33  500.1 s        15.2 s  46.1%       T
         16       1.0491   8.609E-04 3.562E-03 (   23, 35)   2.138E-03 ( 2307, 36) 10:52:03  514.4 s        14.3 s  47.6%       T
         17       1.0485   5.663E-04 2.971E-03 (   59,  9)   1.703E-03 ( 2307, 37) 10:52:32  527.8 s        13.4 s  46.0%       T
         18       1.0480   4.330E-04 2.221E-03 (   59,  9)   1.254E-03 ( 2307, 39) 10:53:03  540.6 s        12.8 s  41.4%       T
         19       1.0477   2.999E-04 1.893E-03 (   59,  9)   9.254E-04 ( 2306, 38) 10:53:33  551.9 s        11.4 s  37.8%       T
         20       1.0475   1.993E-04 9.251E-04 (          6, 19)   7.073E-04 (  966, 21) 10:53:54  561.7 s  9.8 s  46.6%       T
         21       1.0474   7.757E-05 2.143E-03 (   59,  4)   6.659E-04 (  966, 19) 10:54:12  570.0 s        8.3 s  46.2%       T
         22       1.0474   4.905E-05 2.106E-03 (   59,  4)   5.033E-04 (  966, 19) 10:54:31  578.8 s        8.8 s  46.2%       T
         23       1.0474   2.550E-05 2.148E-03 (   59,  4)   4.338E-04 (  970, 18) 10:54:49  587.1 s        8.3 s  46.3%       T
         24       1.0473   4.792E-06 1.960E-03 (   59,  4)   4.077E-04 (  970, 25) 10:55:07  595.6 s        8.5 s  47.2%       T
         25       1.0474   1.150E-05 1.946E-03 (   59,  4)   3.783E-04 (  970, 25) 10:55:23  603.6 s        7.9 s  49.6%       T
         26     1.0474   2.488E-05 8.748E-04 (   59,  8)   3.488E-04 (  970, 26) 10:55:40  611.5 s          7.9 s  46.3%       T
 k-eff =  1.047386E+0
Time per Iteration (sec):

Transport Sweeping:  1500.353
Inscatter Source  : 32.077
Coarse-Mesh Accel :           7.373
Else (Convergence):           1.782

Total Time          : 12332.681
```

Fig. 9.2.72: Timed iteration history output.

9.2.5.4 Four-factor formula

Following the iteration history listing, NEWT output provides edit listing the four traditional components of the four-factor formula. This is followed by an alternate three-group formulation that separates out resonance and fast escape probabilities (Fig. 9.2.73).

```
Four-Factor Estimate of k-infinity.  Fast/Thermal boundary:   0.6250 eV
          Fiss. neutrons/thermal abs. (eta):                 1.774719
          Thermal utilization (f):                           0.713935
          Resonance Escape probability (p):                  0.646659
          Fast-fission factor (epsilon):                     1.268752
                                                            --------------
          Infinite neutron multiplication                    1.039538


   Alternate 3-Group Formulation Terms.
   Fast/Res. boundary:   0.8210 MeV.  Res./Therm. boundary:   0.6250eV
          Fiss. neutrons/thermal abs. (eta):                 1.774719
          Thermal utilization (f):                           0.713935
          Resonance Escape probability (p):                  0.676460
          Fast Escape probability (pf):                      0.955945
          Fast-fission factor (epsilon):                     1.268752
```

Fig. 9.2.73: Four-factor formula with alternate three-group formulation.

9.2.5.5 Fine-group balance tables

Following the iteration history and flux convergence, a fine-group balance table is provided for each mixture used in the calculation. Fine group refers to the group structure of the library used for the calculation. Broad-group data, discussed later, refer to a group structure collapsed from the original fine-group structure. After tables for all mixtures are printed, a last table provides a fine-group summary for the entire problem (i.e., the volume-weighted average for all mixtures). Balance tables are printed by default but may be disabled by setting *prtbalnc=no* in the Parameter block.

Fig. 9.2.74 shows a clipped excerpt from the fine-group summary of an output listing. Similar tables are produced for each mixture in the problem for all energy groups in the problem. The header lists the NEWT mixture number; the mixture ID (i.e., the SCALE mixture number); and the mixture description, if provided in the original input specification. The header also gives the number of computational cells in which the mixture was present and the volume of the mixture in the problem.

For each mixture, two tables are printed. The first table provides a balance of all sources and loss terms: the fixed source, the fission source, in-scatter, out-scatter, absorption, leakage, n-2n production, and the net balance of all terms for each energy group. The final row lists the mixture total for all groups. The fixed source lists the user-supplied source for fixed-source problems. This field is disabled (set to zero) for eigenvalue calculations. The fission source is the number of neutrons born into each energy group in the mixture. In this example, the mixture is water, which is not fissile; hence, no fission source is present. In-scatter represents the number of neutrons scattered into each group from all other groups; conversely, out-scatter is the loss from each energy group by scattering. Absorption is the number of neutrons absorbed in reactions that do not emit a neutron (e.g., n-γ). Leakage is the net loss of neutrons from the mixture to another mixture or a nonreflective boundary, and n-2n is the effective n-2n production rate calculated from a weighted sum of all n-*x*n reactions. The balance table is the ratio of production to loss in each energy group.

The second fine-group balance table, also shown in Fig. 9.2.74, lists other reactions rates of interest. The first two columns after the group number list in-scatter broken into its upscatter and downscatter components. The subsequent two columns provide a similar breakdown for out-scatter from the energy group. Self-scatter is the amount of within-group scattering occurring within each energy group. The fission rate is the number of (n-fission) reactions occurring in each group. The next column provides the transverse leakage (i.e., the product of the flux and the DB^2 term). This column will provide only nonzero values when a nonzero buckling height is specified in input. The final column lists the total (scalar) flux for each energy group.

```
Fine group summary for mix   1, id no.   3 {  water (background material)},
 comprised of   12 cells with a total volume of  1.01436E+00 cm^2

  Group  Fixed Src   Fission Src  In-scatter   Out-scatter  Absorption   Leakage          n-2n      Balance
         1  0.00000E+00  0.00000E+00  0.00000E+00  2.73845E-03  2.42285E-04 -2.98064E-03  4.08871E-10  9.99970E-01
         2  0.00000E+00  0.00000E+00  7.36587E-04  8.38269E-03  4.82391E-04 -8.12839E-03  0.00000E+00  9.99989E-01
         3  0.00000E+00  0.00000E+00  3.28519E-03  2.83897E-02  7.11766E-04 -2.58158E-02  0.00000E+00  9.99984E-01
         4  0.00000E+00  0.00000E+00  1.56784E-02  1.09274E-01  9.61233E-04 -9.45590E-02  0.00000E+00  1.00002E+00
         5  0.00000E+00  0.00000E+00  3.42126E-02  9.38023E-02  2.62964E-06 -5.95934E-02  0.00000E+00  1.00001E+00
         6  0.00000E+00  0.00000E+00  1.37882E-02  3.16448E-02  7.46346E-07 -1.78574E-02  0.00000E+00  1.00000E+00
         7  0.00000E+00  0.00000E+00  5.20554E-02  1.31633E-01  3.56610E-06 -7.95816E-02  0.00000E+00  1.00000E+00

         …

        41  0.00000E+00  0.00000E+00  9.67567E-02  7.62245E-02  1.68405E-03  1.88723E-02  0.00000E+00  9.99750E-01
        42  0.00000E+00  0.00000E+00  1.68407E-02  1.44931E-02  2.35638E-04  2.11355E-03  0.00000E+00  9.99906E-01
        43  0.00000E+00  0.00000E+00  2.03504E-02  1.74257E-02  3.47624E-04  2.57891E-03  0.00000E+00  9.99910E-01
        44  0.00000E+00  0.00000E+00  6.50424E-03  5.81542E-03  1.24190E-04  5.64893E-04  0.00000E+00  9.99959E-01
Total  0.00000E+00  0.00000E+00  8.42221E+00  8.42221E+00  2.71040E-02 -2.88142E-02  4.08871E-10  1.00020E+00

         In-scatter   In-scatter   Out-scatter  Out-scatter  Fission          Total
 Group from above   from below   to above        to below  Self-scatter Rate            DB^2 * Flux  Flux
         1  0.00000E+00  0.00000E+00  0.00000E+00  2.73845E-03  8.09455E-04  0.00000E+00  0.00000E+00  4.75444E-02
         2  7.36587E-04  0.00000E+00  0.00000E+00  8.38269E-03  1.83660E-03  0.00000E+00  0.00000E+00  1.22828E-01
         3  3.28519E-03  0.00000E+00  0.00000E+00  2.83897E-02  9.75693E-03  0.00000E+00  0.00000E+00  3.65164E-01
         4  1.56784E-02  0.00000E+00  0.00000E+00  1.09274E-01  6.89111E-02  0.00000E+00  0.00000E+00  1.16671E+00
         5  3.42126E-02  0.00000E+00  0.00000E+00  9.38023E-02  1.68778E-02  0.00000E+00  0.00000E+00  7.66106E-01
         6  1.37882E-02  0.00000E+00  0.00000E+00  3.16448E-02  1.18925E-03  0.00000E+00  0.00000E+00  2.31649E-01
         7  5.20554E-02  0.00000E+00  0.00000E+00  1.31633E-01  2.85827E-02  0.00000E+00  0.00000E+00  9.22565E-01

         …

        41  8.26269E-02  1.41299E-02  6.16059E-02  1.46187E-02  5.51777E-02  0.00000E+00  0.00000E+00  4.02868E-02
        42  1.40104E-02  2.83027E-03  1.16481E-02  2.84497E-03  1.88120E-03  0.00000E+00  0.00000E+00  4.02048E-03
        43  1.88663E-02  1.48409E-03  1.59333E-02  1.49242E-03  4.71525E-03  0.00000E+00  0.00000E+00  4.61749E-03
        44  6.50424E-03  0.00000E+00  5.81542E-03  0.00000E+00  1.04022E-03  0.00000E+00  0.00000E+00  9.43064E-04
Total  7.96845E+00  4.53757E-01  4.53757E-01  7.96845E+00  6.85400E+00  0.00000E+00  0.00000E+00  2.23894E+01
```

Fig. 9.2.74: Partial mixture fine-group balance table output.

9.2.5.6 Planar fluxes and currents

If planar fluxes are requested, an edit is printed to provide fluxes and currents on each line segment specified, identified by label (Fig. 9.2.75). Fine-group fluxes are listed for each energy group, followed by x and y net currents and partial currents (+x, -x, +y, and -y). Fluxes and currents are printed for each group in the input group structure. The example below shows only partial listings of each for simplicity. If a broad-group collapse is requested, the fine-group output is followed by the set of fluxes and currents for each broad energy group.

Note that discontinuity factors make internal use of planar fluxes to determine the flux and current on each boundary. Hence, planar flux edits will be present any time an ADF calculation is performed.

```
Average Scalar Fluxes on Specified Planes
Fine Group Fluxes:
 Group  fuel west side      fuel south side
       1         5.686166D-04        5.686166D-04
       2         1.616545D-03        1.616545D-03
       3         5.102822D-03        5.102822D-03
       4         1.701169D-02        1.701169D-02
…
   42   3.386267D-04        3.386267D-04
   43   3.658349D-04        3.658349D-04
   44   6.313846D-05        6.313846D-05
Fine Group Currents (x):
 Group  fuel west side      fuel south side
       1         0.000000D+00        0.000000D+00
       2         0.000000D+00        0.000000D+00
       3         0.000000D+00        0.000000D+00
       4         0.000000D+00        0.000000D+00
…
   42   0.000000D+00        0.000000D+00
   43   0.000000D+00        0.000000D+00
   44   0.000000D+00        0.000000D+00
Fine Group Currents (y):
 Group  fuel west side      fuel south side
       1         0.000000D+00        0.000000D+00
       2         0.000000D+00        0.000000D+00
       3         0.000000D+00        0.000000D+00
       4         0.000000D+00        0.000000D+00
…
   42   0.000000D+00        0.000000D+00
   43   0.000000D+00        0.000000D+00
   44   0.000000D+00        0.000000D+00
Fine Group Currents (+x):
 Group  fuel west side      fuel south side
       1         1.416865D-04        1.443455D-04
       2           4.029062D-04      4.100419D-04
       3         1.272173D-03        1.293594D-03
       4         4.242596D-03        4.308137D-03
…
   42   0.000000D+00        0.000000D+00
   43   0.000000D+00        0.000000D+00
   44   0.000000D+00        0.000000D+00
Fine Group Currents (+y):
 Group  fuel west side      fuel south side
       1         1.443455D-04        1.416865D-04
       2         4.100419D-04        4.029062D-04
       3         1.293594D-03        1.272173D-03
       4         4.308137D-03        4.242596D-03
…
   42   0.000000D+00      0.000000D+00
   43   0.000000D+00        0.000000D+00
   44   0.000000D+00        0.000000D+00
```

Fig. 9.2.75: Example of planar flux and current output (continued below).

```
Fine Group Currents (-x):
 Group  fuel west side      fuel south side
       1         1.416865D-04        1.420273D-04
       2         4.029062D-04        4.039803D-04
       3         1.272173D-03        1.276183D-03
       4         4.242596D-03        4.259287D-03
…
   42   0.000000D+00        0.000000D+00
   43   0.000000D+00        0.000000D+00
   44   0.000000D+00        0.000000D+00
Fine Group Currents (-y):
 Group  fuel west side      fuel south side
       1         1.420273D-04        1.416865D-04
       2         4.039803D-04        4.029061D-04
       3         1.276183D-03        1.272173D-03
       4         4.259287D-03        4.242596D-03
…
   42   0.000000D+00        0.000000D+00
   43   0.000000D+00        0.000000D+00
   44   0.000000D+00        0.000000D+00

Broad Group Fluxes:
 Group  fuel west side      fuel south side
       1         3.125488D-01        3.125488D-01
       2         8.729971D-02        8.729971D-02

Broad Group Currents (x):
 Group  fuel west side      fuel south side
       1         0.000000D+00        0.000000D+00
       2         0.000000D+00        2.217642D-04

Broad Group Currents (y):
 Group  fuel west side      fuel south side
       1         0.000000D+00        0.000000D+00
       2         2.217642D-04        0.000000D+00

Broad Group Currents (+x):
 Group  fuel west side      fuel south side
       1         7.824686D-02        7.861045D-02
       2         2.175537D-02        2.209011D-02

Broad Group Currents (+y):
 Group  fuel west side      fuel south side
       1         7.861045D-02        7.824686D-02
       2         2.209011D-02        2.175537D-02

Broad Group Currents (-x):
 Group  fuel west side      fuel south side
       1         7.824686D-02        7.832338D-02
       2         2.175537D-02        2.146369D-02

Broad Group Currents (-y):
 Group  fuel west side      fuel south side
       1         7.832338D-02        7.824686D-02
       2         2.146369D-02        2.175537D-02
```

9.2.5.7 Pin-power edits

The next section of the NEWT output listing is the pin-power edit (Fig. 9.2.76). This information is printed only if *pinpow=yes* is set in one or more arrays. Two maps are provided. The first is the power of each fuel location relative to all other fuel pins in all other arrays for which *pinpow=yes*. This is useful if multiple fuel assemblies are present or if more than one array is used to describe a fuel assembly. The second power map shows the pin edit normalized to the set of pins within the single array. Both maps are identical in a relative sense; different normalization factors are applied. If *pinpow=yes* is specified for only one array, then the two edits will have the same normalization factor and will be identical. The location of each pin is identified by the (x,y) coordinate of the center of each element of the array, in centimeters. Please note that the pin-power option is available only for square (cuboidal) arrays.

Following the two maps is a one-line edit identifying the location and magnitude of the maximum pin power.

```
 ****************************************************************
   **                          Pin Power Edits                   **
 ****************************************************************


        Array   101
   ---------------

--      Power relative to all pins in entire domain     --

y/x->     0.358  1.430  2.860  4.290  5.720  7.150  8.580 10.010
|
V

10.010   1.0034 0.9936 0.9769 0.9759 0.9718 0.9667 0.9789 0.9893

 8.580   1.0058 0.9781 0.9227 0.9491 0.9427 0.9076 0.9507 0.9789

 7.150   1.0483 0.9675 0.0000 0.9173 0.9023 0.0000 0.9076 0.9667

 5.720   0.0000 1.0679 0.9874 1.0246 0.9704 0.9023 0.9427 0.9718

 4.290   1.1248 1.0908 1.0966 0.0000 1.0246 0.9173 0.9491 0.9759

 2.860   1.1096 1.0975 1.0922 1.0966 0.9874 0.0000 0.9227 0.9769

 1.430   1.1555 1.1181 1.0975 1.0908 1.0679 0.9675 0.9781 0.9936

 0.358   0.0000 1.1555 1.1096 1.1248 0.0000 1.0483 1.0058 1.0034


--      Power relative to pins in this array            --

y/x->   0.358  1.430  2.860  4.290  5.720  7.150  8.580 10.010
|
V

10.010   1.0034 0.9936 0.9769 0.9759 0.9718 0.9667 0.9789 0.9893

 8.580   1.0058 0.9781 0.9227 0.9491 0.9427 0.9076 0.9507 0.9789

 7.150   1.0483 0.9675 0.0000 0.9173 0.9023 0.0000 0.9076 0.9667

 5.720   0.0000 1.0679 0.9874 1.0246 0.9704 0.9023 0.9427 0.9718

 4.290   1.1248 1.0908 1.0966 0.0000 1.0246 0.9173 0.9491 0.9759

 2.860   1.1096 1.0975 1.0922 1.0966 0.9874 0.0000 0.9227 0.9769

 1.430   1.1555 1.1181 1.0975 1.0908 1.0679 0.9675 0.9781 0.9936

 0.358   0.0000 1.1555 1.1096 1.1248 0.0000 1.0483 1.0058 1.0034


  Peak pin power of 1.1555 occurs in array  101 in element (  2,  1), centered at x= 1.4300, y= 0.3575
```

Fig. 9.2.76: Example of pin-power edit.

9.2.5.8 Broad-group collapse

Broad-group summary data

The next section of the NEWT output listing is the broad-group summary listing (Fig. 9.2.77). This is printed only if a broad-group collapse is performed. This section lists broad-group data calculated based on the collapsing scheme applied. First, the energy group structure is printed, followed by cell-averaged fluxes in each mixture, for all collapsed groups. This is followed by flux disadvantage factors for each mixture and each broad group.

Note that when NEWT is used as the transport solver within TRITON depletion calculations, a three-group collapse is always done automatically. If a second user-specified collapse is requested, broad-group summary data will be provided for both collapsing structures.

```
Collapsing cross sections to specified  3-group format
 ******************************************************

 Creating library entitled: Cell-weighted cross sections

 Broad Group Parameters

 Group         Upper          Upper          Mid            Mid     Velocity
Fission
  No.  Energy (eV)   Lethargy       Energy (eV)   Lethargy       (cm/s)
Spectrum
       1      2.0000E+07   -6.9315E-01     0.0000E+00     8.5740E-01     0.0000E+00
7.4581E-01
       2      9.0000E+05     2.4079E+00     0.0000E+00     9.7212E+00     0.0000E+00
2.5419E-01
       3      4.0000E-01     1.7034E+01     0.0000E+00     2.2333E+01     0.0000E+00
7.8147E-10
       4      3.0000E-03   2.1927E+01

 Cell Averaged Fluxes

 Mixture  Group  1    Group  2       Group  3

       3      6.19535E+00   1.50375E+01   8.39555E-01
    2  6.24594E+00   1.49470E+01   6.91304E-01
       1      6.52660E+00   1.48732E+01   4.64954E-01
 Total   6.30217E+00   1.49777E+01   7.09204E-01

 Flux Disadvantage Factors (material/total flux)

 Mixture  Group  1    Group  2       Group  3

       1      9.83049E-01   1.00400E+00   1.18380E+00
       2      9.91077E-01   9.97954E-01   9.74761E-01
       3      1.03561E+00   9.93023E-01   6.55600E-01
 Total   1.00000E+00   1.00000E+00   1.00000E+00

 Material Volumes (cc/unit length)

Mixture  Volume       Vol. Fraction
   3   1.01436E+00  5.88655E-01
   2   1.80720E-01  1.04876E-01
   1   5.28102E-01  3.06469E-01
```

Fig. 9.2.77: Broad-group summary output.

Broad-group cross section data

The next section of data in a NEWT output listing is the broad-group cross section output (Fig. 9.2.78). This is printed only if a broad-group collapse is performed and if *prtbroad=yes* is specified in Parameter block input. This block lists the collapsed cross section data for key reactions for each nuclide used in the calculation. This is a summary form of the data that are written to the collapsed cross section library. It does not list all reactions. Such data may be read directly from the working-format library by other SCALE utilities if needed. The listing below shows the data printed for a single nuclide. Data are written in the same format for all nuclides used in the analysis.

```
***** Broad group parameters by nuclide *****


Nuclide ID:   1092238 (92U 238 ANL+ EVALJUN77 E.PENNINGTON A.         MOD3 02/13/92)
  Group             Flux               Total     Absorption          Fission            Transport          Transport           Nu        Chi
   No.                        Cross section  Cross section  Cross section  Cross section  Cross section
                                                                  (Outscatter)   (Inscatter)
         1         5.67670E+01         8.74022E+00         3.02771E-01         1.89115E-01         6.53840E+00         6.53562E+00         2.81103E+00         9.98336E-01
         2         2.89544E+01         1.51675E+01         2.08034E+00         1.20259E-04         1.51213E+01         1.51260E+01         2.41388E+00         1.66474E-03
         3         2.47419E+00         9.69117E+00         1.29810E+00         2.48680E-06         9.66742E+00         9.67663E+00         2.41360E+00         2.09884E-10

 p0 matrix from working file id #   1092238 (92U 238 ANL+ EVALJUN77 E.PENNINGTON A.                MOD3 02/13/92)

From   to group   1 to group   2        to group   3
Group
  3                6.28117E-02         8.33018E+00
  2                1.30795E+01         7.72704E-03
  1       8.43377E+00         1.37608E-02         4.63052E-09

 p1 matrix from working file id #   1092238 (92U 238 ANL+ EVALJUN77 E.PENNINGTON A.                MOD3 02/13/92)

From   to group   1 to group   2        to group   3
Group
  3               -3.30806E-02         1.04314E-01
  2                1.43688E-01        -5.18547E-03
  1       6.63884E+00        -8.31698E-03        -7.46863E-09

 p2 matrix from working file id #   1092238 (92U 238 ANL+ EVALJUN77 E.PENNINGTON A.                MOD3 02/13/92)

From   to group   1 to group   2        to group   3
Group
  3               -1.11434E-02         1.12753E-02
  2                1.40316E-03        -9.59964E-04
  1       6.40138E+00        -3.73698E-04         4.08463E-09
```

Fig. 9.2.78: Partial broad-group cross section listing.

9.2.5.9 Critical spectrum edit

When a critical buckling correction is requested (e.g., *solntype=b1* is set in the NEWT parameter block or user-defined material buckling or transverse height), the critical spectrum is computed using either the B1 approximation or the P1 approximation (Fig. 9.2.79). The output lists the buckling in $1/cm^2$, the method (B1 or P1), and the computed critical spectrum as a function of energy. Note that the spectrum is normalized "per unit lethargy" to be equal to 1.0. In addition to the critical spectrum, the critical adjoint spectrum and the zero-buckling spectra (forward and adjoint) are also edited.

```
Critical Spectra Calculated with the B1 Method
 Critical Buckling is  0.73987E-03
 Cross sections were weighted using the B1 critical flux.

  Group   Upper   Critical   Inf.Med.   Critical   Inf.Med.
   No   Energy(eV)   Flux              Flux      Adj.Flux   Adj.Flux

       1  2.0000E+07  7.611E-10  7.785E-10  1.993E-03  2.185E-03
       2  1.7333E+07  3.436E-06  3.482E-06  2.637E-03  2.864E-03
       3  1.5683E+07  9.885E-06  9.975E-06  3.241E-03  3.502E-03
       4  1.4550E+07  1.861E-05  1.878E-05  4.621E-03  4.990E-03
       5  1.3840E+07  3.440E-05  3.468E-05  2.915E-03  3.144E-03
       6  1.2840E+07  1.535E-04  1.546E-04  8.079E-04  8.676E-04
       7  1.0000E+07  5.751E-04  5.767E-04  9.573E-04  1.023E-03
       8  8.1873E+06  1.811E-03  1.819E-03  7.315E-04  7.812E-04
       9  6.4340E+06  4.493E-03  4.495E-03  5.372E-04  5.712E-04
      10  4.8000E+06  6.706E-03  6.683E-03  1.421E-03  1.499E-03
      11  4.3040E+06  9.693E-03  9.620E-03  4.275E-04  4.474E-04
      12  3.0000E+06  1.460E-02  1.451E-02  7.967E-04  8.313E-04
    13  2.4790E+06  1.645E-02  1.640E-02  2.920E-03  3.042E-03
      14  2.3540E+06  1.426E-02  1.415E-02  6.226E-04  6.447E-04
      15  1.8500E+06  1.475E-02  1.463E-02  7.019E-04  7.230E-04
      16  1.5000E+06  1.539E-02  1.527E-02  2.095E-03  2.154E-03
      17  1.4000E+06  1.491E-02  1.479E-02  4.486E-03  4.607E-03
      18  1.3560E+06  1.267E-02  1.255E-02  4.892E-03  5.019E-03
      19  1.3170E+06  1.392E-02  1.380E-02  2.730E-03  2.797E-03
      20  1.2500E+06  1.557E-02  1.544E-02  3.487E-03  3.573E-03
      21  1.2000E+06  1.433E-02  1.421E-02  1.637E-03  1.676E-03
      22  1.1000E+06  1.151E-02  1.139E-02  1.671E-03  1.706E-03
      23  1.0100E+06  1.164E-02  1.152E-02  1.528E-03  1.560E-03
      24  9.2000E+05  1.330E-02  1.317E-02  6.485E-03  6.624E-03
...

   236  7.5000E-04  1.531E-06  1.547E-06  4.707E-04  4.673E-04
   237  5.0000E-04  2.933E-07  2.963E-07  1.182E-04  1.173E-04
   238  1.0000E-04  6.975E-09  7.046E-09  8.223E-05  8.164E-05
```

Fig. 9.2.79: Partial collapsing spectra listing for a case with critical buckling correction.

9.2.5.10 Assembly discontinuity factors

When calculation of assembly discontinuity factors (ADFs) is requested, a broad-group edit is provided for each face for which an ADF was selected (Fig. 9.2.80). Up to four ADFs may be printed for the fuel region. If the model contains a reflector region in addition to the required fuel region, then ADFs are printed for a single face, typically the fuel/moderator interface. Discontinuity factors for user-input surfaces may also be edited.

```
1Fuel Assembly Discontinuity Factors for W Face
 ********************************************
 Energy Group   Discontinuity Factor
          1                   1.00843
          2                    1.08890


1Fuel Assembly Discontinuity Factors for S Face
 ********************************************
 Energy Group   Discontinuity Factor
          1                   1.01903
          2                   0.90881
```

Fig. 9.2.80: Output of assembly discontinuity factors.

9.2.5.11 Groupwise form factors

Whenever homogenization is performed and pin-power edits are requested, NEWT will automatically calculate groupwise form factors (GFFs). GFFs are used in pin-power reconstruction calculations for homogenized assemblies used in nodal diffusion methods (Fig. 9.2.81).

```
          ******************************************************************
          **                  Groupwise Form Factor Edits                 **
          ******************************************************************

Homogenization region:  500 type6a

          Array     1
  ________________

--        GFF by pin position in this array.          --

Grp Pin->           (  1,  1)   (  2,  1)   (  3,  1)   (  4,  1)   (  5,  1)   (  6,  1)   (  7,  1)   (  8,  1)   (  1,  2)   (  2,  2)
  |
  V

  1                0.9978     1.0118     1.0179     1.0224     1.0236     1.0139     1.0125     1.0000     1.0132     1.0404
  2                1.1399     1.0013     0.9267     0.8883     0.8759     0.8869     0.9225     1.0000     1.0009     0.8614
Grp Pin->           (  3,  2)   (  4,  2)   (  5,  2)   (  6,  2)   (  7,  2)   (  8,  2)   (  1,  3)   (  2,  3)   (  3,  3)   (  4,  3)
  |
  V

  1                1.0333     1.0429     1.0414     1.0354     1.0286     1.0175     1.0188     1.0332     1.0436     1.0418
  2                0.7917     0.7593     0.7498     0.7548     0.7844     0.8609      0.9265    0.7920     0.7305     0.7087
Grp Pin->           (  5,  3)   (  6,  3)   (  7,  3)   (  8,  3)   (  1,  4)   (  2,  4)   (  3,  4)   (  4,  4)   (  5,  4)   (  6,  4)
  |
  V

  1                1.0433     1.0321     1.0320     1.0195     1.0233     1.0425     1.0419     1.0524     0.0000     1.0394
  2                0.7114     0.6960     0.7163     0.7867     0.8884     0.7597     0.7088     0.7182     0.0000     0.6811
Grp Pin->           (  7,  4)   (  8,  4)   (  1,  5)   (  2,  5)   (  3,  5)   (  4,  5)   (  5,  5)   (  6,  5)   (  7,  5)   (  8,  5)
  |
  V

  1                1.0365     1.0234     1.0246     1.0416     1.0433     0.0000     1.0473     1.0344     1.0359     1.0193
  2                0.6821     0.7459      0.8761    0.7503     0.7116     0.0000     0.6895     0.6523     0.6645     0.7293
Grp Pin->           (  1,  6)   (  2,  6)   (  3,  6)   (  4,  6)   (  5,  6)   (  6,  6)   (  7,  6)   (  8,  6)   (  1,  7)   (  2,  7)
  |
  V

  1                 1.0149    1.0352     1.0323     1.0394     1.0345     1.0330     1.0221     1.0154     1.0137     1.0288
  2                0.8870     0.7553     0.6963     0.6812     0.6524     0.6450     0.6660     0.7340     0.9227     0.7850
Grp Pin->           (  3,  7)   (  4,  7)   (  5,  7)   (  6,  7)   (  7,  7)   (  8,  7)   (  1,  8)   (  2,  8)   (  3,  8)   (  4,  8)
  |
  V

  1                1.0322     1.0369     1.0359     1.0223     1.0268     1.0078     0.9998     1.0160     1.0183     1.0223
  2                0.7166     0.6822     0.6646     0.6660     0.6919     0.7605     1.0009     0.8623     0.7878     0.7467
Grp Pin->           (  5,  8)   (  6,  8)   (  7,  8)   (  8,  8)
  |
  V

  1                1.0182     1.0144      1.0064    1.0007
  2                0.7301     0.7349     0.7614     0.8243
```

Fig. 9.2.81: Partial collapsing spectra listing for a case with no critical buckling correction.

Homogenized cross sections

When homogenization is performed and parameter *prthmmix=yes* is set, the final output section of a NEWT calculation is the homogenized cross section edit, as shown in Fig. 9.2.82. This information is generally passed to nodal analysis codes and hence is presented in a slightly different format from other cross sections. Output includes a region-averaged k-infinity value, transport-corrected cross section, and two interpretations of absorption. The first is the directly collapsed absorption cross section, while the second (Total-Scatter) is a more consistent definition of absorption as applied in nodal calculations. The difference between the two definitions is the effective (n-2n) cross section. Both cross sections exclude contributions from ^{135}Xe and ^{149}Sm; microscopic cross sections and number densities for these two nuclides are printed explicitly elsewhere in the table. Nu*fission is the product of the fission cross section and the number of neutrons produced per fission, while Kappa*fission is the product of the fission cross section and the energy release per fission (J). Inverse velocity is the inverse (1/x) of the group neutron speed.

The table also lists the two-group isotropic scattering matrix and the prompt fission fraction distribution. Finally, NEWT lists approximate six-group decay constants (lambdas) and group fractions (betas) for each

group.

```
1Flux weighted cross-sections for homogenized regions
 ******************************************************

 Cell No.  1: fuel

 k-infinity for region 500 using flux weighted cross-sections =       0.90151870

    Group     Total          Transport      Absorption     Total-Scatter  Nu*Fission     Kappa*Fission  Fission        Flux
-------------------------------------------------------------------------------------------------------------------------------
      1        4.21626E-01    3.01356E-01    3.31363E-02    3.30118E-02    2.89905E-02    3.59028E-13    1.14985E-02    1.79670E+01
      2        7.74202E-01    7.18995E-01    3.73077E-01    3.73077E-01    7.11011E-01    9.07058E-12    2.91793E-01    2.12458E+00

    Group     Inv. Velocity  Absorption     Total-Scatter  Xe-135         Sm-149         Xe-135         Sm-149
                             (less Xe/Sm)   (less Xe/Sm)   Micro. Absorp. Micro. Absorp  Atom Density   Atom Density
-------------------------------------------------------------------------------------------------------------------------------
      1        5.25355E-08    3.31363E-02    3.30118E-02    0.00000E+00    0.00000E+00    0.00000E+00    0.00000E+00
      2        2.26246E-06    3.73077E-01    3.73077E-01    0.00000E+00    0.00000E+00    0.00000E+00    0.00000E+00

    P0 Scattering Matrix
-------------------------------------------------------------------------------------------
    To:               1             2
    From:  1        3.87843E-01    7.71384E-04
           2        2.10833E-03    3.99016E-01

...

    Key fission product yields:
         I-135      6.28810E-02
         Xe-135     2.56610E-03
         Pm-149     6.81187E-05


     Prompt fission fraction (chi)
     Group, g      Chi(g)
-------------------------------------
        1           1.00000E+00
        2           2.11326E-09

     Delayed Neutron Spectrum (adjoint weighted)
    Group          Decay Constant Fraction
                   (Lambda, 1/s)  (Beta)
-------------------------------------------------
      1             1.24943E-02    2.12473E-04
      2             3.08079E-02    1.41816E-03
      3             1.14647E-01    1.30178E-03
      4             3.09698E-01    2.70481E-03
      5             1.23542E+00    8.96161E-04
      6             3.30386E+00    3.01302E-04
              Total Beta           0.68347E-02

     Delayed Neutron Spectrum (fission weighted)
    Group          Decay Constant Fraction
                   (Lambda, 1/s)  (Beta)
-------------------------------------------------
      1             1.24943E-02    2.18547E-04
      2             3.08078E-02    1.45869E-03
      3             1.14646E-01    1.33897E-03
      4             3.09696E-01    2.78207E-03
      5             1.23541E+00    9.21727E-04
      6             3.30380E+00    3.09897E-04
              Total Beta           0.70299E-02
```

Fig. 9.2.82: Homogenized cross section edit for nodal diffusion applications.

9.2.5.12 End-of-calculation banner

NEWT output listings are terminated with an end-of-calculation banner (shown in Fig. 9.2.83) upon successful completion of a calculation. If this banner is not present, then the calculation ended abnormally, and the output listing must be reviewed to determine the cause of the error. In general, the final lines of an output file describe the error condition that caused the calculation to stop.

```
*****************************************************************************************************************************
         NEWT execution completed with zero errors
*****************************************************************************************************************************
```

Fig. 9.2.83: End-of-calculation banner listing.

Postscript graphics files

Two user-selectable options within NEWT provide the ability to generate PostScript-based graphics files for visualization of both input specifications and output results. By specification of *drawit=yes* in the NEWT parameter block, NEWT will generate two PostScript-based plot files: `newtgrid.ps` and `newtmatl.ps`. The former, a grayscale plot of the line segments generated by NEWT based on the input specification, will be generated if all body placement input is valid. If input contains errors such that the code stops before grid generation routines are completed, no `newtgrid.ps` output is created.

The `newtmatl.ps` plot illustrates the same grid structure but with material placement indicated by color. At this time, no user control is provided for color assignment or plot control. This plot also requires complete grid generation; additionally, it requires completion of all media placement routines before the plot will be produced.

Figures used throughout this manual were generated from newtgrid and newtmatl PostScript plot files. Files `newtgrid.ps` and `newtmatl.ps` are automatically copied back from SCALE's temporary directory to the original location of the input case, with the names *casename*.newtgrid.ps and *casename*.newtmatl.ps.

When *prtflux=yes* is input, NEWT will generate a set of flux plots showing relative neutron number densities in each energy group. A plot file will be generated with the name fluxplot_*N*g.ps, where *N* is the number of energy groups in the problem. If an energy collapse is performed, an additional file named fluxplot_*M*g.ps is created, where M is the number of energy groups in the collapsed set. Fig. 9.2.84 is an example of a flux plot output for the fast group of a two-group flux collapse.

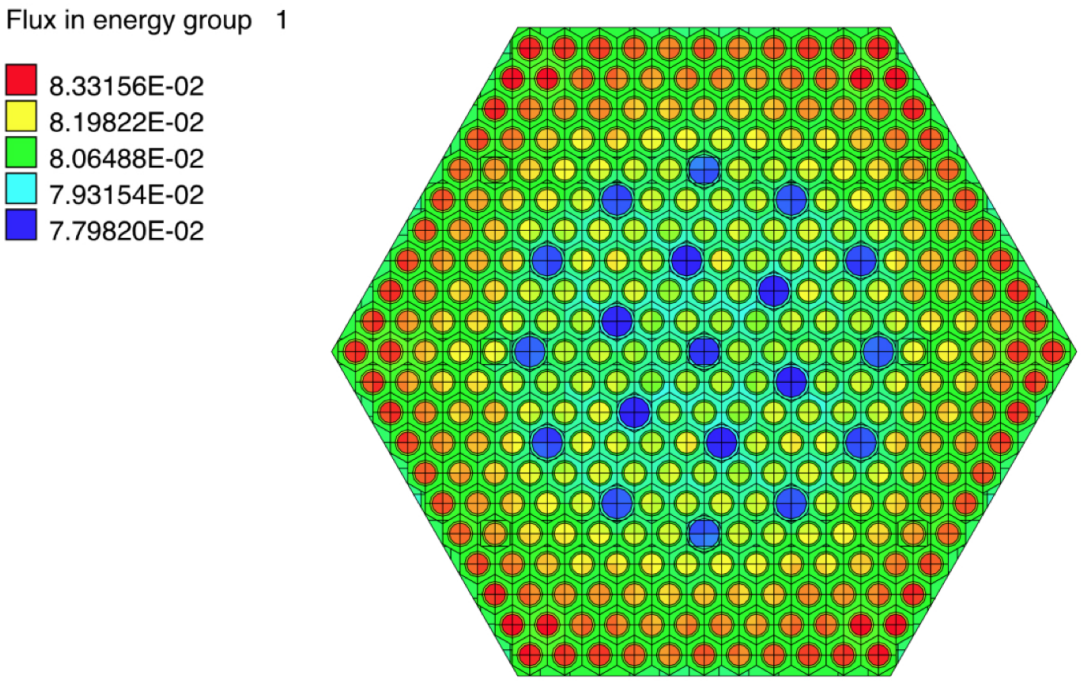


Fig. 9.2.84: Example of a flux plot image created with prtflux=yes.

9.2.5.13 Media zone edits

NEWT automatically determines "zones" representing spatially independent regions of the same media. For example, in a fuel pin cell, the fuel, clad, and moderator are all considered separate zones. In an array of such pin cells, each unique location is a unique zone. Zone numbers and the geometric location of each zone are listed in the *Geometry Specification*" in Sect. 9.2.5.2.6.

Upon completion of a calculation, NEWT provides an output edit of each zone by number, giving the mixture number, average flux, fission power, and volume, as shown in Fig. 9.2.85.

Zone data

Zone No.	Mix No.	Relative Flux	Relative Fission Power	Zone Volume
1	31	3.33874E-01	0.00000E+00	2.19695E-01
2	11	3.33263E-01	1.00000E+00	1.31704E-01
3	21	3.32862E-01	0.00000E+00	4.55018E-02

Fig. 9.2.85: Media zone output edit.

10. SCALE NUCLEAR DATA LIBRARIES

Introduction by A. Holcomb, D. Wiarda and M. L. Williams

Chapter 10 describes the SCALE cross section data libraries for use with deterministic and Monte Carlo radiation transport modules. All cross section libraries were processed from ENDF/B-VII.1 or -VIII.0 evaluated data files using the AMPX code system [SCALE-DATA-WWCD15]. SCALE includes continuous-energy libraries, as well as multigroup libraries with several group structures. Libraries are available for neutron, gamma, and coupled neutron-gamma transport calculations. The fine and broad multigroup libraries provided for reactor physics and criticality safety applications in SCALE 6.3 include intermediate resonance parameters (lambdas) and improved Bondarenko data for self-shielding calculations using the Bondarenko method, or the traditional CENTRM-based procedures in SCALE can be used for self-shielding. Section 10.1 in this chapter describes the available cross section libraries.

Fine and broad group covariance libraries containing cross section uncertainties and correlations are also distributed with SCALE for sensitivity/uncertainty analysis with the Sampler and TSUNAMI modules. The covariance libraries include a comprehensive collection of data for all nuclides included in the SCALE cross section libraries. New 252-group and 56-group covariances based on ENDF/B-VII.1 and other data sources are available, along with the older 44-group covariance library distributed with earlier releases of SCALE. The Covariance Libraries chapter describes the contents of the SCALE 6.3 covariance libraries and explains how they were processed.

Additional libraries used for transmutation calculations with ORIGEN are described in the ORIGEN Data Resources section of the ORIGEN chapter. These libraries include fission product yields, decay data, decay gamma spectra, etc., as well as supplemental cross section data not available in ENDF/B.

10.1 SCALE CROSS SECTION LIBRARIES

A. Holcomb, D. Wiarda, C. Celik, K. S. Kim, M. L. Williams, M. E. Dunn, B. T. Rearden

ABSTRACT

The cross section data libraries available in the SCALE code system are briefly described in this section. All libraries were processed from ENDF/B-VII.1 and -VIII.0 evaluated data files using the AMPX code system. Continuous-energy libraries as well as several multigroup libraries for a variety of applications are included in SCALE. Several fine-group and broad-group structures are available so that a user may select the nuclear data library based on considerations of application, accuracy, and execution time.

ACKNOWLEDGMENTS

We would like to acknowledge all those who participated in developing and implementing the SCALE cross section libraries for their assistance in preparing previous versions of this document. Special thanks go to S. Goluoglu, N. M. Greene, and L. M. Petrie for their contributions to SCALE data processing efforts, and to C. R. Daily and W. J. Marshall for their extensive validation testing of the libraries described here.

10.1.1 INTRODUCTION

SCALE includes both multigroup (MG) and pointwise continuous energy (CE) nuclear data libraries, which were processed using the AMPX code system [XSLibWWCD15]. Libraries are available for neutron and for coupled neutron-photon transport calculations. The CE libraries are used for Monte Carlo calculations with CE-KENO (criticality) and CE-Monaco (shielding), and are also used by the pointwise (PW) discrete ordinates code CENTRM to obtain PW flux spectra for computing self-shielded MG cross sections. The MG libraries are used in the MG-KENO and MG-Monaco Monte Carlo codes, and in the deterministic transport codes XSDRNPM, NEWT, and DENOVO. All cross section libraries in SCALE 6.3 and later versions are based strictly on ENDF/B-VII.1 [XSLibCHO+11] and ENDF/B-VIII.0 [XSLibDAB18], while earlier SCALE releases included libraries processed from ENDF/B-VI.8, V, and IV. The ENDF/B-VII.1 CE and MG libraries include the 423 nuclides shown in Table 10.1.1, and the ENDF/B-VIII.0 CE and MG libraries contain data for 556 nuclides (Note that while ENDF/B-VIII.0 contains data for neutron-as-a-target, this data is not useful for SCALE calculations and is thus the processed data is not released with SCALE.). ENDF does not have evaluated data for several isotopes in the SCALE Standard Composition Data; therefore the libraries do not include cross sections for these nuclides, which are listed in Table 10.1.2. If one of these isotopes is explicitly requested or if the natural element containing the isotope is requested in the SCALE input, a warning message is written saying that the nuclide is being omitted from the calculations. The most commonly encountered example of this is O-18 which has an abundance of approximately 0.2% in elemental oxygen (Note that O-18 data are available in ENDF/B-VIII.0, but not ENDF/B-VII.1.).

The ENDF/B-VII.1 libraries include 21 thermal-scattering moderators for which bound-scattering kernels [e.g., S(α,β)] are provided as given in Table 10.1.3. This table also lists the temperatures available in MG and CE libraries for the materials with bound kernels. The ENDF/B-VIII.0 libraries include 34 thermal-scattering moderators for which bound-scattering kernels are provided; of particular interest are updated thermal scatter laws for HinH2O and multiple graphite compositions. The thermal scattering kernels for all other materials are based on the free-gas kernel. On the MG libraries, these nuclides have 2D scattering matrices processed from the free-gas kernel evaluated at the temperatures of 293 K, 565 K, 600 K, 900 K, 1200 K, 2000 K, and 2400 K. SCALE 6.3 includes routines that automatically interpolate the CE cross section data, as well as thermal scattering kernels to any arbitrary temperature for Monte Carlo calculations [XSLibHCML16]. Analogous procedures are available for MG data. The CE libraries do not include free gas kernel data because the transport codes internally evaluate the free gas scattering information at the specified temperature.

For each reaction and nuclide, CE data were processed from ENDF/B in the manner described in Sect. 10.1.2.5. The CE data were further processed into several generic MG libraries focused on different applications. CE data are stored as individual files for each nuclide, and an associated cross section directory file contains the names of the individual files. Furthermore, the MG libraries each contain data for all nuclides. AMPX master libraries are very general and contain essentially all reaction data available in ENDF/B, as well as several specialized cross sections used in SCALE. The master libraries also include Bondarenko shielding factors tabulated as a function of background cross section and temperature for all nuclides and groups, and the master libraries may contain intermediate resonance parameters for self-shielding. A much improved methodology in AMPX was used to process more accurate shielding factors for the most important resonance absorbers.[1]

The CE and MG libraries available in SCALE 6.3 are given in Table 10.1.4, which also lists the main source of data for each library. The desired MG or CE library for a computation is typically selected by specifying the appropriate mnemonic name from Table 10.1.4. The CE library mnemonic corresponds to the actual name of the cross section directory file, and the MG mnemonic is an alias for the file containing the actual master library, shown in the third column of Table 10.1.4. SCALE control modules recognize the mnemonics in the first column, or the standard MG library name in the third columns of the table. Users also may supply

their own MG library by specifying the file name as the library name in the control module input (only lower case alphanumeric characters are allowed in the filename, and the filename must be eight characters or less in length). The library must be an AMPX master library and must be located in the SCALE data directory or the temporary working directory where the problem is run.

The CE data libraries do not require additional processing prior to their use in transport computations. However the MG libraries distributed with the SCALE code system contain problem-*independent* cross sections processed with generic weighting functions appropriate for a specified type of application (e.g., criticality safety, shielding, etc.). These data must be further processed into problem-*dependent* libraries by performing resonance self-shielding and other modifications. This is done by the XSProc module during the execution of a SCALE control sequence prior to executing a transport solver.

Table 10.1.1: Nuclides in CE and MG ENDF/B-VII.0 and -VII.1 libraries.

SCALE ID	**Name**	**Gamma production data**[a]	**Full range Bondarenko factors**[b]	**Gamma interaction evaluation**[c]	**Notes**
1001	h	yes	yes	h	
1001001	h-liquid_ch4	yes	yes	h	
2001001	h-solid_ch4	yes	yes	h	
4001001	h-cryo_ortho	yes	yes	h	
5001001	h-cryo_para	yes	yes	h	
6001001	h-benzene	yes	yes	h	
7001001	h-zrh2	yes	yes	h	
8001001	hfreegas	yes	yes	h	
9001001	h-poly	yes	yes	h	
1002	d	yes	yes	h	
4001002	d-cryo_ortho	yes	yes	h	
5001002	d-cryo_para	yes	yes	h	
8001002	dfreegas	yes	yes	h	
1003	h-3		yes	h	
2003	he-3		yes	he	
2004	he-4		yes	he	
3006	li-6	yes	yes	li	
3007	li-7	yes	yes	li	
4007	be-7		yes	be	
4009	be-9	yes	yes	be	
3004009	bebound	yes	yes	be	
5004009	be-beo	yes	yes	be	
5010	b-10	yes	yes	b	
5011	b-11	yes	yes	b	
6000	c	yes	yes	c	
3006000	graphite	yes	yes	c	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
5006000	h-benzene	yes	yes	c	
7014	n-14	yes	yes	n	
7015	n-15	yes	yes	n	
8016	o-16	yes	yes	o	
5008016	o-beo	yes	yes	o	
8017	o-17		yes	o	
9019	f-19	yes	yes	f	
11022	na-22		yes	na	
11023	na-23	yes	yes	na	
12024	mg-24	yes	yes	mg	
12025	mg-25	yes	yes	mg	
12026	mg-26	yes	yes	mg	
13027	al-27	yes	yes	al	
1013027	albound	yes	yes	al	
14028	si-28	yes	yes	si	
14029	si-29	yes	yes	si	
14030	si-30	yes	yes	si	
1014028	sibound	yes	yes	si	ENDF/B-VII.1 only
1014029	sibound	yes	yes	si	ENDF/B-VII.1 only
1014030	sibound	yes	yes	si	ENDF/B-VII.1 only
15031	p-31	yes	yes	p	
16032	s-32	yes	yes	s	
16033	s-33	yes	yes	s	
16034	s-34	yes	yes	s	
16036	s-36	yes	yes	s	
17035	cl-35	yes	yes	cl	
17037	cl-37	yes	yes	cl	
18036	ar-36		yes	ar	
18038	ar-38		yes	ar	
18040	ar-40		yes	ar	
19039	k-39	yes	yes	k	
19040	k-40	yes	yes	k	
19041	k-41	yes	yes	k	
20040	ca-40	yes	yes	ca	
20042	ca-42	yes	yes	ca	
20043	ca-43	yes	yes	ca	
20044	ca-44	yes	yes	ca	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
20046	ca-46	yes	yes	ca	
20048	ca-48	yes	yes	ca	
21045	sc-45	yes	yes	sc	
22046	ti-46	yes	yes	ti	
22047	ti-47	yes	yes	ti	
22048	ti-48	yes	yes	ti	
22049	ti-49	yes	yes	ti	
22050	ti-50	yes	yes	ti	
23000	v	yes		v	ENDF/B-VII.0 only
23050	v-50	yes	yes	v	ENDF/B-VII.1 only
23051	v-51	yes	yes	v	ENDF/B-VII.1 only
24050	cr-50	yes	yes	cr	
24052	cr-52	yes	yes	cr	
24053	cr-53	yes	yes	cr	
24054	cr-54	yes	yes	cr	
25055	mn-55	yes	yes	mn	
26054	fe-54	yes	yes	fe	
26056	fe-56	yes	yes	fe	
26057	fe-57	yes	yes	fe	
26058	fe-58	yes	yes	fe	
1026000	febound	yes	yes	fe	
27058	co-58	ENDF/B-VII.1	yes	co	
1027058	co-58m		yes	co	
27059	co-59	yes	yes	co	
28058	ni-58	yes	yes	ni	
28059	ni-59	yes	yes	ni	
28060	ni-60	yes	yes	ni	
28061	ni-61	yes	yes	ni	
28062	ni-62	yes	yes	ni	
28064	ni-64	yes	yes	ni	
29063	cu-63	yes	yes	cu	
29065	cu-65	yes	yes	cu	
30000	zn			zn	ENDF/B-VII.0 only
30064	zn-64	yes	yes	zn	ENDF/B-VII.1 only

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
30065	zn-65	yes	yes	zn	ENDF/B-VII.1only
30066	zn-66	yes	yes	zn	ENDF/B-VII.1 only
30067	zn-67	yes	yes	zn	ENDF/B-VII.1 only
30068	zn-68	yes	yes	zn	ENDF/B-VII.1 only
30070	zn-70	yes	yes	zn	ENDF/B-VII.1 only
31069	ga-69		yes	ga	
31071	ga-71		yes	ga	
32070	ge-70	yes	yes	ge	
32072	ge-72	yes	yes	ge	
32073	ge-73	yes	yes	ge	
32074	ge-74	yes	yes	ge	
32076	ge-76	yes	yes	ge	
33074	as-74	yes	yes	as	
33075	as-75	yes	yes	as	
34074	se-74		yes	se	
34076	se-76		yes	se	
34077	se-77		yes	se	
34078	se-78		yes	se	
34079	se-79		yes	se	
34080	se-80		yes	se	
34082	se-82		yes	se	
35079	br-79		yes	br	
35081	br-81		yes	br	
36078	kr-78	ENDF/B-VII.1	yes	kr	
36080	kr-80		yes	kr	
36082	kr-82		yes	kr	
36083	kr-83		yes	kr	
36084	kr-84		yes	kr	
36085	kr-85	yes	yes	kr	
36086	kr-86		yes	kr	
37085	rb-85		yes	rb	
37086	rb-86	yes	yes	rb	
37087	rb-87		yes	rb	
38084	sr-84	yes	yes	sr	
38086	sr-86		yes	sr	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
38087	sr-87		yes	sr	
38088	sr-88		yes	sr	
38089	sr-89		yes	sr	
38090	sr-90		yes	sr	
39089	y-89	yes	yes	y	
39090	y-90	yes	yes	y	
39091	y-91		yes	y	
40090	zr-90	yes	yes	zr	
1040090	zr90-zr5h8	yes	yes	zr	
40091	zr-91	yes	yes	zr	
1040091	zr91-zr5h8	yes	yes	zr	
40092	zr-92	yes	yes	zr	
1040092	zr92-zr5h8	yes	yes	zr	
40093	zr-93	ENDF/B-VII.1	yes	zr	
1040093	zr93-zr5h8	ENDF/B-VII.1	yes	zr	
40094	zr-94	yes	yes	zr	
1040094	zr94-zr5h8	yes	yes	zr	
40095	zr-95	ENDF/B-VII.1	yes	zr	
1040095	zr95-zr5h8	ENDF/B-VII.1	yes	zr	
40096	zr-96	yes	yes	zr	
1040096	zr96-zr5h8	yes	yes	zr	
41093	nb-93	yes	yes	nb	
41094	nb-94		yes	nb	
41095	nb-95		yes	nb	
42092	mo-92	yes	yes	mo	
42094	mo-94	yes	yes	mo	
42095	mo-95	yes	yes	mo	
42096	mo-96	yes	yes	mo	
42097	mo-97	yes	yes	mo	
42098	mo-98	yes	yes	mo	
42099	mo-99		yes	mo	
42100	mo-100		yes	mo	
43099	tc-99	yes	yes	tc	
44096	ru-96		yes	ru	
44098	ru-98		yes	ru	
44099	ru-99		yes	ru	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
44100	ru-100		yes	ru	
44101	ru-101	yes	yes	ru	
44102	ru-102		yes	ru	
44103	ru-103		yes	ru	
44104	ru-104		yes	ru	
44105	ru-105		yes	ru	
44106	ru-106		yes	ru	
45103	rh-103	yes	yes	rh	
45105	rh-105		yes	rh	
46102	pd-102	yes	yes	pd	
46104	pd-104	yes	yes	pd	
46105	pd-105	yes	yes	pd	
46106	pd-106	yes	yes	pd	
46107	pd-107		yes	pd	
46108	pd-108	yes	yes	pd	
46110	pd-110	yes	yes	pd	
47107	ag-107	yes	yes	ag	
47109	ag-109	yes	yes	ag	
1047110	ag-110m		yes	ag	
47111	ag-111	yes	yes	ag	
48106	cd-106	yes	yes	cd	
48108	cd-108		yes	cd	
48110	cd-110		yes	cd	
48111	cd-111	yes	yes	cd	
48112	cd-112		yes	cd	
48113	cd-113		yes	cd	
48114	cd-114		yes	cd	
1048115	cd-115m	yes	yes	cd	
48116	cd-116		yes	cd	
49113	in-113		yes	in	
49115	in-115		yes	in	
50112	sn-112		yes	sn	
50113	sn-113	yes	yes	sn	
50114	sn-114		yes	sn	
50115	sn-115		yes	sn	
50116	sn-116		yes	sn	
50117	sn-117		yes	sn	
50118	sn-118		yes	sn	
50119	sn-119		yes	sn	
50120	sn-120		yes	sn	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
50122	sn-122		yes	sn	
50123	sn-123		yes	sn	
50124	sn-124		yes	sn	
50125	sn-125	yes	yes	sn	
50126	sn-126		yes	sn	
51121	sb-121		yes	sb	
51123	sb-123		yes	sb	
51124	sb-124		yes	sb	
51125	sb-125		yes	sb	
51126	sb-126	yes	yes	sb	
52120	te-120		yes	te	
52122	te-122		yes	te	
52123	te-123		yes	te	
52124	te-124		yes	te	
52125	te-125		yes	te	
52126	te-126		yes	te	
1052127	te-127m		yes	te	
52128	te-128		yes	te	
1052129	te-129m		yes	te	
52130	te-130		yes	te	
52132	te-132	yes	yes	te	
53127	i-127	yes	yes	i	
53129	i-129		yes	i	
53130	i-130	yes	yes	i	
53131	i-131		yes	i	
53135	i-135		yes	i	
54123	xe-123	ENDF/B-VII.1	yes	xe	
54124	xe-124	ENDF/B-VII.1	yes	xe	
54126	xe-126		yes	xe	
54128	xe-128		yes	xe	
54129	xe-129		yes	xe	
54130	xe-130		yes	xe	
54131	xe-131	yes	yes	xe	
54132	xe-132		yes	xe	
54133	xe-133		yes	xe	
54134	xe-134		yes	xe	
54135	xe-135		yes	xe	
54136	xe-136		yes	xe	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
55133	cs-133	yes	yes	cs	
55134	cs-134		yes	cs	
55135	cs-135		yes	cs	
55136	cs-136		yes	cs	
55137	cs-137		yes	cs	
56130	ba-130		yes	ba	
56132	ba-132		yes	ba	
56133	ba-133	yes	yes	ba	
56134	ba-134		yes	ba	
56135	ba-135		yes	ba	
56136	ba-136		yes	ba	
56137	ba-137		yes	ba	
56138	ba-138		yes	ba	
56140	ba-140		yes	ba	
57138	la-138		yes	la	
57139	la-139		yes	la	
57140	la-140	yes	yes	la	
58136	ce-136	yes	yes	ce	
58138	ce-138	yes	yes	ce	
58139	ce-139	yes	yes	ce	
58140	ce-140		yes	ce	
58141	ce-141		yes	ce	
58142	ce-142		yes	ce	
58143	ce-143	yes	yes	ce	
58144	ce-144		yes	ce	
59141	pr-141	yes	yes	pr	
59142	pr-142	yes	yes	pr	
59143	pr-143		yes	pr	
60142	nd-142	yes	yes	nd	
60143	nd-143	yes	yes	nd	
60144	nd-144	yes	yes	nd	
60145	nd-145	yes	yes	nd	
60146	nd-146	yes	yes	nd	
60147	nd-147	yes	yes	nd	
60148	nd-148	yes	yes	nd	
60150	nd-150	yes	yes	nd	
61147	pm-147		yes	pm	
61148	pm-148		yes	pm	
1061148	pm-148m		yes	pm	
61149	pm-149		yes	pm	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
61151	pm-151	yes	yes	pm	
62144	sm-144	yes	yes	sm	
62147	sm-147	yes	yes	sm	
62148	sm-148	yes	yes	sm	
62149	sm-149	yes	yes	sm	
62150	sm-150	yes	yes	sm	
62151	sm-151	yes	yes	sm	
62152	sm-152	yes	yes	sm	
62153	sm-153	yes	yes	sm	
62154	sm-154	yes	yes	sm	
63151	eu-151		yes	eu	
63152	eu-152		yes	eu	
63153	eu-153	yes	yes	eu	
63154	eu-154		yes	eu	
63155	eu-155		yes	eu	
63156	eu-156		yes	eu	
63157	eu-157	yes	yes	eu	
64152	gd-152	yes	yes	gd	
64153	gd-153	yes	yes	gd	
64154	gd-154	yes	yes	gd	
64155	gd-155	yes	yes	gd	
64156	gd-156	yes	yes	gd	
64157	gd-157	yes	yes	gd	
64158	gd-158	yes	yes	gd	
64160	gd-160	yes	yes	gd	
65159	tb-159		yes	tb	
65160	tb-160	yes	yes	tb	
66156	dy-156	yes	yes	dy	
66158	dy-158	yes	yes	dy	
66160	dy-160	yes	yes	dy	
66161	dy-161	yes	yes	dy	
66162	dy-162	yes	yes	dy	
66163	dy-163	yes	yes	dy	
66164	dy-164	yes	yes	dy	
67165	ho-165	yes	yes	ho	
1067166	ho-166m	yes	yes	ho	
68162	er-162	yes	yes	er	
68164	er-164	yes	yes	er	
68166	er-166	yes	yes	er	
68167	er-167	yes	yes	er	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
68168	er-168	yes	yes	er	
68170	er-170	yes	yes	er	
69168	tm-168	yes	yes	tm	ENDF/B-VII.1 only
69169	tm-169	yes	yes	tm	ENDF/B-VII.1 only
69170	tm-170	yes	yes	tm	ENDF/B-VII.1 only
71175	lu-175		yes	lu	
71176	lu-176		yes	lu	
72174	hf-174	ENDF/B-VII.1	yes	hf	
72176	hf-176	ENDF/B-VII.1	yes	hf	
72177	hf-177	ENDF/B-VII.1	yes	hf	
72178	hf-178	ENDF/B-VII.1	yes	hf	
72179	hf-179	ENDF/B-VII.1	yes	hf	
72180	hf-180	ENDF/B-VII.1	yes	hf	
73180	ta-180	yes	yes	ta	ENDF/B-VII.1 only
73181	ta-181	yes	yes	ta	
73182	ta-182		yes	ta	
74180	w-180	yes	yes	w	ENDF/B-VII.1 only
74182	w-182	yes	yes	w	
74183	w-183	yes	yes	w	
74184	w-184	yes	yes	w	
74186	w-186	yes	yes	w	
75185	re-185	ENDF/B-VII.1	yes	re	
75187	re-187	ENDF/B-VII.1	yes	re	
77191	ir-191	yes	yes	ir	
77193	ir-193	yes	yes	ir	
79197	au-197	yes	yes	au	
80196	hg-196	yes	yes	hg	
80198	hg-198	yes	yes	hg	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
80199	hg-199	yes	yes	hg	
80200	hg-200	yes	yes	hg	
80201	hg-201	yes	yes	hg	
80202	hg-202	yes	yes	hg	
80204	hg-204	yes	yes	hg	
81203	tl-203	yes	yes	tl	ENDF/B-VII.1 only
81205	tl-205	yes	yes	tl	ENDF/B-VII.1 only
82204	pb-204	yes	yes	pb	
82206	pb-206	yes	yes	pb	
82207	pb-207	yes	yes	pb	
82208	pb-208	yes	yes	pb	
83209	bi-209	yes	yes	bi	
88223	ra-223		yes	ra	
88224	ra-224		yes	ra	
88225	ra-225		yes	ra	
88226	ra-226		yes	ra	
89225	ac-225	ENDF/B-VII.1	yes	ac	
89226	ac-226	ENDF/B-VII.1	yes	ac	
89227	ac-227	ENDF/B-VII.1	yes	ac	
90227	th-227	ENDF/B-VII.1	yes	th	
90228	th-228	ENDF/B-VII.1	yes	th	
90229	th-229	ENDF/B-VII.1	yes	th	
90230	th-230	ENDF/B-VII.1	yes	th	
90231	th-231	yes	yes	th	ENDF/B-VII.1 only
90232	th-232	yes	yes	th	
90233	th-233	ENDF/B-VII.1	yes	th	
90234	th-234	ENDF/B-VII.1	yes	th	
91229	pa-229	yes	yes	pa	ENDF/B-VII.1 only

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
91230	pa-230	yes	yes	pa	ENDF/B-VII.1 only
91231	pa-231	yes	yes	pa	
91232	pa-232	ENDF/B-VII.1	yes	pa	
91233	pa-233	yes	yes	pa	
92230	u-230	yes	yes	u	ENDF/B-VII.1 only
92231	u-231	yes	yes	u	ENDF/B-VII.1 only
92232	u-232	yes	yes	u	
92233	u-233	yes	yes	u	
92234	u-234	yes	yes	u	ENDF/B-VII.1 only
92235	u-235	yes	yes	u	
92236	u-236	yes	yes	u	
92237	u-237	yes	yes	u	
92238	u-238	yes	yes	u	
92239	u-239	yes	yes	u	
92240	u-240	yes	yes	u	
92241	u-241	yes	yes	u	
93234	np-234	yes	yes	np	
93235	np-235	ENDF/B-VII.1	yes	np	
93236	np-236	ENDF/B-VII.1	yes	np	
93237	np-237	yes	yes	np	
93238	np-238	ENDF/B-VII.1	yes	np	
93239	np-239	ENDF/B-VII.1	yes	np	
94236	pu-236	ENDF/B-VII.1	yes	pu	
94237	pu-237	ENDF/B-VII.1	yes	pu	
94238	pu-238		yes	pu	
94239	pu-239	yes	yes	pu	
94240	pu-240	yes	yes	pu	
94241	pu-241	yes	yes	pu	
94242	pu-242	yes	yes	pu	
94243	pu-243	yes	yes	pu	

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
94244	pu-244	ENDF/B-VII.1	yes	pu	
94246	pu-246	ENDF/B-VII.1	yes	pu	
95240	am-240	yes	yes	am	ENDF/B-VII.1 only
95241	am-241	yes	yes	am	
95242	am-242		yes	am	
1095242	am-242m		yes	am	
95243	am-243	yes	yes	am	
95244	am-244		yes	am	
1095244	am-244m		yes	am	
96240	cm-240	yes	yes	cm	ENDF/B-VII.1 only
96241	cm-241	ENDF/B-VII.1	yes	cm	
96242	cm-242	yes	yes	cm	
96243	cm-243	ENDF/B-VII.1	yes	cm	
96244	cm-244	ENDF/B-VII.1	yes	cm	
96245	cm-245	ENDF/B-VII.1	yes	cm	
96246	cm-246	ENDF/B-VII.1	yes	cm	
96247	cm-247	ENDF/B-VII.1	yes	cm	
96248	cm-248	yes	yes	cm	
96249	cm-249	ENDF/B-VII.1	yes	cm	
96250	cm-250	ENDF/B-VII.1	yes	cm	
97245	bk-245	yes	yes	bk	ENDF/B-VII.1 only
97246	bk-246	yes	yes	bk	ENDF/B-VII.1 only
97247	bk-247	yes	yes	bk	ENDF/B-VII.1 only
97248	bk-248	yes	yes	bk	ENDF/B-VII.1 only

continues on next page

Table 10.1.1 – continued from previous page

SCALE ID	Name	Gamma production data[a]	Full range Bondarenko factors[b]	Gamma interaction evaluation[c]	Notes
97249	bk-249	ENDF/B-VII.1	yes	bk	
97250	bk-250	ENDF/B-VII.1	yes	bk	
98246	cf-246	yes	yes	cf	ENDF/B-VII.1 only
98248	cf-248	yes	yes	cf	ENDF/B-VII.1 only
98249	cf-249	ENDF/B-VII.1	yes	cf	
98250	cf-250	yes	yes	cf	
98251	cf-251	yes	yes	cf	
98252	cf-252	yes	yes	cf	
98253	cf-253	ENDF/B-VII.1	yes	cf	
98254	cf-254	ENDF/B-VII.1	yes	cf	
99251	es-251	yes	yes	es	ENDF/B-VII.1 only
99252	es-252	yes	yes	es	ENDF/B-VII.1 only
99253	es-253	ENDF/B-VII.1	yes	es	
99254	es-254	ENDF/B-VII.1	yes	es	
1099254	es-254m	yes	yes	es	ENDF/B-VII.1 only
99255	es-255	ENDF/B-VII.1	yes	es	
100255	fm-255	yes	yes	fm	

[a] Yield data are only available in coupled MG libraries and in the CE libraries.

[b] Narrow and/or intermediate resonance factors are only available on MG libraries.

[c] Incident gamma cross sections are only available on coupled MG libraries. A separate incident gamma CE library is available

Table 10.1.2: Isotopes with no ENDF/B-VII.0 or -VII.1 nuclear data.

Element	SCALE standard composition ID	Missing Isotopes	ZA numbers	% Abundance
oxygen	8000	18	8018	0.20
neon	10000	21, 22	10021, 10022	0.27, 9.25
ytterbium	70000	All(1)	(1)	
osmium	76000	All(2)	(2)	
platinum	78000	All(3)	(3)	
tantalum	73000	180m	1073180	0.01

(1) no data for any of the 7 naturally-occurring ytterbium isotopes

(2) no data for any of the 7 naturally-occurring osmium isotopes

(3) no data for any of the 6 naturally-occurring platinum isotopes

Table 10.1.3: Temperatures at which thermal moderator data are available[a].

ID	**Name**	**Temperatures**
1001	h-1	293.6 350.0 400.0 450.0 500.0 550.0 600.0 650.0 800.0
1001001	h-liquid_ch4	100.0
2001001	h-solid_ch4	22.0
4001001	h-cryo_ortho	20.0
5001001	h-cryo_para	20.0
6001001	h-benzene	296.0 350.0 400.0 450.0 500.0 600.0 800.0 1000.0
7001001	h-zrh2	296.0 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0
9001001	h-poly	296.0 350.0
1002	h-2	293.6 350.0 400.0 450.0 500.0 550.0 600.0 650.0
4001002	d-cryo_ortho	19.0
5001002	d-cryo_para	19.0
3004009	bebound	296.0 400.0 500.0 600.0 700.0 800.0001 1000.0 1200.0
5004009	be-beo	293.6 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0
3006000	c-graphite	296.0 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0 1600.0 2000.0
5006000	h-benzene	296.0 350.0 400.0 450.0 500.0 600.0 800.0 1000.0
5008016	o-beo	293.6 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0
1013027	albound	20.0 80.0 293.6 400.0 600.0 800.0
1014028[a]	sibound	293.6 350.0 400.0 500.0 800.0 1000.0 1200.0
1014029[a]	sibound	293.6 350.0 400.0 500.0 800.0 1000.0 1200.0
1014030[a]	sibound	293.6 350.0 400.0 500.0 800.0 1000.0 1200.0
1026000	febound	20.0 80.0 293.6 400.0 600.0 800.0
1040090	zr90-zr5h8	296.0 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0
1040091	zr90-zr5h8	296.0 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0
1040092	zr90-zr5h8	296.0 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0
1040093	zr90-zr5h8	296.0 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0
1040094	zr90-zr5h8	296.0 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0
1040095	zr90-zr5h8	296.0 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0

continues on next page

Table 10.1.3 – continued from previous page

1040096	zr90-zr5h8	296.0 400.0 500.0 600.0 700.0 800.0 1000.0 1200.0

[a)] only available in ENDF/B-VII.1

Table 10.1.4: Standard SCALE cross section libraries.

Mnemonic names	**Primary data source/format**	**Last field of cross section library filename**
v7.1-252 v7.1-252n	ENDF/B-VII.1 252-group neutron	xn252v7.1
v8.0-252	ENDF/B-VIII.0 252-group neutron library	xn252v8.0
v7-56 v7-56n v7.1-56n	ENDF/B-VII.1 56-group neutron library	xn56v7.1
test-8grp	TEST LIBRARY 8-group ENDF/B-VII.1 neutron library	test8g_v7.1
v7.1-200n47g	ENDF/B-VII.1 200 neutron/47 gamma library	xn200g47v7.1
v8.0-200n47g	ENDF/B-VIII.0 200 neutron/47 gamma library	xn200g47v8.0
v7.1-28n19g	ENDF/B-VII.1 28 neutron/19 gamma library	xn28g19v7.1
v8.0-28n19g	ENDF/B-VIII.0 28 neutron/19 gamma library	xn28g19v8.0
ce_v7.1_endf	ENDF/B-VII.1 Continuous-energy neutron and gamma library	
ce_v8_endf ce_v8.0	ENDF/B-VIII.0 Continuous-energy neutron and gamma library	
xn302 fine_fast_e7.1	ENDF/B-VII.1 302-group neutron library	
fine_fast_e8.0	ENDF/B-VIII.0 302-group neutron library	
v7.1-1597 vfine_e7.1	ENDF/B-VII.1 1597-group neutron library	
v8.0-1597 vfine_e8.0	ENDF/B-VIII.0 1597-group neutron library	

Additional convenience mnemonics are also available to always alias to the most recent nuclear data libraries for the intended purpose. The mnemonics shown in Table 10.1.2 will allow the use of the same input files with this and future versions of SCALE, but will always access the most recent nuclear data libraries and group structures.

Table 10.1.5: SCALE convenience mnemonics.

Mnemonic name	**Aliased library**
broad_lwr	xn56v7.1
fine_therm	xn252v7.1
ce	ce_v7.1_endf
ce.xml	ce_v7.1_endf.xml
test_n	test8g_v7.1

10.1.2 DESCRIPTION OF THE SCALE CROSS SECTION LIBRARIES

10.1.2.1 The 238-group and 252-group ENDF/B-VII libraries (V7-238, v7-252)

SCALE includes a fine group structure for criticality safety and reactor physics applications: a 252-group library structure based on either ENDF/B-VII.1 or ENDF/B-VIII.0 is available for either criticality safety or reactor physics. Table 10.1.8, shows the group structure for the 252 fine-group libraries. The 252-group structure was developed to adequately capture spectral and temperature effects important for reactor systems and was processed with newer, improved procedures.

The SCALE control sequences for criticality safety and reactor physics applications normally perform self-shielding of the fine-group libraries using the BONAMI module for the unresolved resonance range; and the CENTRM/PMC modules for the resolved resonance/thermal range. However the 252-group libraries include Bondarenko self-shielding factors for the entire energy range, which provides the option of using the Bondarenko method to self-shield both the resolved and unresolved resonance ranges, as an alternative to the more rigorous (and computationally intensive) CENTRM/PMC approach. As discussed in the following section, one objective of the ENDF/B-VII.1 and ENDF/B-VIII.0 252-group and 56-group libraries was to provide a more accurate Bondarenko treatment for the resolved resonance range.

Differences in the 238-group and 252-group libraries

The standard weighting function described in Table 10.1.6 is typically used to create MG data for all materials in a library, and Bondarenko shielding factors can be computed using the narrow resonance (NR) approximation for the flux spectrum: $\sigma_t + \sigma_0$)*C(E) where σ_0 is the background cross section, and C(E) is the standard weight function. Bondarenko factors are tabulated at temperatures of at 293 K, 565 K, 600 K, 900 K, 1200 K, 2000 K, and 2400 K in the 252-group library.

Table 10.1.6: Standard weighting function for processing MG data.

Energy Range	Standard Weight Function
10^{-5} eV - 0.1 eV	Maxwellian, with peak at 0.025 eV
0.1 eV - 80 keV	1/E
80 keV - 10 MeV	Watt Fission spectrum at temperature of 1.273 MeV
10 MeV -20 MeV	1/E

Several enhancements were made in the MG processing procedures used to produce the 252-group library so that it would be more applicable to reactor physics as well as criticality safety applications. Some of the improvements in the 252-group library compared to the SCALE-6.2 238-group library are given below

(a) The base weighting function for processing MG data of actinide materials (Z>89) was computed by the PW transport code CENTRM for a PWR lattice at 300 K. This approach provides more representative weighted 2D scattering matrices for most cases of interest. The standard weighting function is still used for materials with Z<90.

(b) The thermal energy range which includes up-scattering reactions was extended to 5 eV, compared to 3 eV in the SCALE 6.2 238-group library

(c) Temperature-dependent thermal-scattering matrices for water-bound H, O-16, and actinide materials were processed with temperature-dependent thermal flux spectra obtained from CENTRM calculations for a PWR pincell. Actinide and O-16 MG thermal scattering kernels were weighted with the fuel zone flux at temperatures of 293 K, 600 K, 900 K, 1200 K, and 2400 K, and the water scatter kernels were weighted with the moderator flux at 293 K, 500 K, 600 K, 650 K, 900 K, and 1200 K. In the SCALE 6.2 238-group library,

thermal scattering matrices at all temperatures were weighted with a temperature-independent Maxwellian spectrum.

(d) Group-dependent IR parameters ("lambdas") were calculated for all materials and are included in the 252-group libraries. This allows the Bondarenko self-shielding method in SCALE to use the IR approximation for the 252-group libraries, while the SCALE 6.2 238-group library was limited to the NR approximation.

(e) A number of improvements were made in processing of Bondarenko self-shielding data.

- The number of temperatures for the Bondarenko factors was increased. Shielding factors are tabulated at temperatures 292 K, 600 K, 900 K, 1200 K, and 2400 K for the 252-group libraries.
- In addition to the Bondarenko factors normally included for capture, fission, elastic, and total cross sections, self-shielding factors are also included for the multigroup elastic within-group cross section to address the impact of resonance reactions on the scattering distribution.
- In the unresolved resonance range, self-shielding factors were calculated using probability tables.
- Bondarenko factors for nuclides with atomic masses Z>39 were calculated with CENTRM PW flux spectra rather than the analytical NR approximation. Two types of CENTRM models were used. Heterogeneous models of water-moderated lattices spanning the range of expected self-shielding were used to calculate shielding factors for ^{235}U, ^{238}U, ^{239}Pu, ^{240}Pu, ^{241}Pu, ^{90}Zr, and ^{96}Zr. The CENTRM transport calculations were performed using the method of characteristics method for 2D unit cell models. Homogeneous models were used to compute shielding factors for the remaining nuclides with Z>39. These CENTRM calculations were performed for homogeneous media containing the absorber material mixed with hydrogen, and the hydrogen concentration was varied to obtain the desired set of background cross sections. Table 10.1.7 summarizes the processing options for the 252-group library.

Table 10.1.7: Standard weighting functions for processing 252-group library data library.

Nuclide	**Base Weight Function**	**Bondarenko Factor**
U-238, -235; Pu-239, -240, -241	PWR spectrum	CENTRM heterogeneous calculations
Other actinides Z>90	PWR spectrum	CENTRM homogeneous calculations
39<Z<90	standard	CENTRM homogeneous calculations
Z<40	standard	NR analytical spectrum

Table 10.1.8: 252 Multigroup energy structure (5 eV thermal boundary).

Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)
1	2.000E+07	43	1.490E+05	85	1.877E+02
2	1.733E+07	44	1.283E+05	86	1.800E+02
3	1.568E+07	45	1.000E+05	87	1.700E+02
4	1.455E+07	46	8.500E+04	88	1.430E+02
5	1.384E+07	47	8.200E+04	89	1.220E+02
6	1.284E+07	48	7.500E+04	90	1.190E+02
7	1.000E+07	49	7.300E+04	91	1.175E+02
8	8.187E+06	50	6.000E+04	92	1.160E+02
9	6.434E+06	51	5.200E+04	93	1.130E+02

continues on next page

Table 10.1.8 – continued from previous page

Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)
10	4.800E+06	52	5.000E+04	94	1.080E+02
11	4.304E+06	53	4.500E+04	95	1.050E+02
12	3.000E+06	54	3.000E+04	96	1.012E+02
13	2.479E+06	55	2.000E+04	97	9.700E+01
14	2.354E+06	56	1.700E+04	98	9.000E+01
15	1.850E+06	57	1.300E+04	99	8.170E+01
16	1.500E+06	58	9.500E+03	100	8.000E+01
17	1.400E+06	59	8.030E+03	101	7.600E+01
18	1.356E+06	60	5.700E+03	102	7.200E+01
19	1.317E+06	61	3.900E+03	103	6.750E+01
20	1.250E+06	62	3.740E+03	104	6.500E+01
21	1.200E+06	63	3.000E+03	105	6.300E+01
22	1.100E+06	64	2.500E+03	106	6.100E+01
23	1.010E+06	65	2.250E+03	107	5.800E+01
24	9.200E+05	66	2.200E+03	108	5.340E+01
25	9.000E+05	67	1.800E+03	109	5.060E+01
26	8.750E+05	68	1.550E+03	110	4.830E+01
27	8.611E+05	69	1.500E+03	111	4.520E+01
28	8.200E+05	70	1.150E+03	112	4.400E+01
29	7.500E+05	71	9.500E+02	113	4.240E+01
30	6.790E+05	72	6.830E+02	114	4.100E+01
31	6.700E+05	73	6.700E+02	115	3.960E+01
32	6.000E+05	74	5.500E+02	116	3.910E+01
33	5.730E+05	75	3.050E+02	117	3.800E+01
34	5.500E+05	76	2.850E+02	118	3.763E+01
35	4.920E+05	77	2.400E+02	119	3.727E+01
36	4.700E+05	78	2.200E+02	120	3.713E+01
37	4.400E+05	79	2.095E+02	121	3.700E+01
38	4.200E+05	80	2.074E+02	122	3.600E+01
39	4.000E+05	81	2.020E+02	123	3.550E+01
40	3.300E+05	82	1.930E+02	124	3.500E+01
41	2.700E+05	83	1.915E+02	125	3.375E+01
42	2.000E+05	84	1.885E+02	126	3.325E+01
127	3.175E+01	169	2.470E+00	211	7.500E-01
128	3.125E+01	170	2.380E+00	212	7.000E-01
129	3.000E+01	171	2.300E+00	213	6.500E-01
130	2.750E+01	172	2.210E+00	214	6.250E-01
131	2.500E+01	173	2.120E+00	215	6.000E-01
132	2.250E+01	174	2.000E+00	216	5.500E-01
133	2.175E+01	175	1.940E+00	217	5.000E-01
134	2.120E+01	176	1.860E+00	218	4.500E-01
135	2.050E+01	177	1.770E+00	219	4.000E-01
136	2.000E+01	178	1.680E+00	220	3.750E-01

continues on next page

Table 10.1.8 – continued from previous page

Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)
137	1.940E+01	179	1.590E+00	221	3.500E-01
138	1.850E+01	180	1.500E+00	222	3.250E-01
139	1.700E+01	181	1.450E+00	223	3.000E-01
140	1.600E+01	182	1.400E+00	224	2.750E-01
141	1.440E+01	183	1.350E+00	225	2.500E-01
142	1.290E+01	184	1.300E+00	226	2.250E-01
143	1.190E+01	185	1.250E+00	227	2.000E-01
144	1.150E+01	186	1.225E+00	228	1.750E-01
145	1.000E+01	187	1.200E+00	229	1.500E-01
146	9.100E+00	188	1.175E+00	230	1.250E-01
147	8.100E+00	189	1.150E+00	231	1.000E-01
148	7.150E+00	190	1.140E+00	232	9.000E-02
149	7.000E+00	191	1.130E+00	233	8.000E-02
150	6.875E+00	192	1.120E+00	234	7.000E-02
151	6.750E+00	193	1.110E+00	235	6.000E-02
152	6.500E+00	194	1.100E+00	236	5.000E-02
153	6.250E+00	195	1.090E+00	237	4.000E-02
154	6.000E+00	196	1.080E+00	238	3.000E-02
155	5.400E+00	197	1.070E+00	239	2.530E-02
156	5.000E+00	198	1.060E+00	240	1.000E-02
157	4.700E+00	199	1.050E+00	241	7.500E-03
158	4.100E+00	200	1.040E+00	242	5.000E-03
159	3.730E+00	201	1.030E+00	243	4.000E-03
160	3.500E+00	202	1.020E+00	244	3.000E-03
161	3.200E+00	203	1.010E+00	245	2.500E-03
162	3.100E+00	204	1.000E+00	246	2.000E-03
163	3.000E+00	205	9.750E-01	247	1.500E-03
164	2.970E+00	206	9.500E-01	248	1.200E-03
165	2.870E+00	207	9.250E-01	249	1.000E-03
166	2.770E+00	208	9.000E-01	250	7.500E-04
167	2.670E+00	209	8.500E-01	251	5.000E-04
168	2.570E+00	210	8.000E-01	252	1.000E-04
					1.000E-05

10.1.2.2 The 56-group library

An ENDF/B-VII.1 and an ENDF/B-VIII.0 broad group library with 56 energy groups are available mainly for light water reactor physics calculations. The group structure is shown in Table 10.1.9. This library was processed using the same PW flux spectra used to generate the 252-group libraries (i.e., for a PWR fuel lattice). This library includes the same materials and properties as the 252-group library, and the data were computed in the same manner, except for several specially weighted nuclides which have heterogeneous Bondarenko shielding factors (in addition to those given in Table 10.1.4) computed for specific LWR components. These are summarized in Table 10.1.10.

Table 10.1.9: 56-Group energy structure (5 eV thermal boundary).

Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)
1	2.0000E+07	24	1.0500E+02	47	2.0000E-01
2	6.4340E+06	25	1.0120E+02	48	1.5000E-01
3	4.3040E+06	26	6.7500E+01	49	1.0000E-01
4	3.0000E+06	27	6.5000E+01	50	8.0000E-02
5	1.8500E+06	28	3.7130E+01	51	6.0000E-02
6	1.5000E+06	29	3.6000E+01	52	5.0000E-02
7	1.2000E+06	30	2.1750E+01	53	4.0000E-02
8	8.6110E+05	31	2.1200E+01	54	2.5300E-02
9	7.5000E+05	32	2.0500E+01	55	1.0000E-02
10	6.0000E+05	33	7.0000E+00	56	4.0000E-03
11	4.7000E+05	34	6.8750E+00		1.0000E-05
12	3.3000E+05	35	6.5000E+00		
13	2.7000E+05	36	6.2500E+00		
14	2.0000E+05	37	5.0000E+00		
15	5.0000E+04	38	1.1300E+00		
16	2.0000E+04	39	1.0800E+00		
17	1.7000E+04	40	1.0100E+00		
18	3.7400E+03	41	6.2500E-01		
19	2.2500E+03	42	4.5000E-01		
20	1.9150E+02	43	3.7500E-01		
21	1.8770E+02	44	3.5000E-01		
22	1.1750E+02	45	3.2500E-01		
23	1.1600E+02	46	2.5000E-01		

Table 10.1.10: Speciality nuclides with special shielding factors(*)
in 56-group library.

Nuclide	**ID**	**Component configuration used to compute Bondarenko factors**
Zr-91	10040091	standard library weighting
Zr-96	10040096	standard library weighting
Zr-91	40091	LWR lattice cladding, with U238 resonance interference
Zr-96	40096	LWR lattice cladding, with U238 resonance interference
Ag-107	47107	PWR Ag-In-Cd control rod
Ag-109	47109	PWR Ag-In-Cd control rod
In-113	49113	PWR Ag-In-Cd control rod
In-115	49115	PWR Ag-In-Cd control rod
Cd-113	48113	PWR Ag-In-Cd control rod

10.1.2.3 The test-8grp library for code testing

The library named test-8grp is used for code testing and verification of reproducibility. It was collapsed from the fine-group v7-252 library, using the standard weight functions in Table 10.1.6. This library has all the nuclides and same types of nuclear data as in the v7-252 library; but the eight energy-group structure provides capability to test codes and input in shorter times than with the standard production libraries. Table 10.1.11 gives the eight group structure, which has four thermal groups below 3 eV, and four fast groups.

The library can also be used with the CENTRM/PMC resonance shielding methodology.

Important: THIS LIBRARY SHOULD NOT BE USED FOR REAL APPLICATIONS.

Table 10.1.11: 8-Group energy structure

Group	Upper energy (eV)
1	2.000E+07
2	8.200E+05
3	2.000E+04
4	1.050E+02
5	5.000E+00
6	6.250E-01
7	1.500E-01
8	4.000E-01
	1.000E-05

10.1.2.4 The 200N-47G (V7-200N47G) library for shielding

A coupled fine-group neutron-gamma library based on ENDF/B-VII.1 is available for radiation transport calculations with SCALE shielding modules. The 200 neutron and 47 gamma energy group structures are provided in Table 10.1.12 and Table 10.1.13, respectively. The neutron group structure is identical to the 199-group VITAMIN-B6 [XSLibWDP+09] structure except that an additional group has been added to extend the top energy boundary to 20 MeV. The MG neutron data were generated using the standard weighting function described in Table 10.1.6, and the MG photon data were weighted with a flat spectrum with roll-offs. Full-range Bondarenko factors are provided for all nuclides, and the default self-shielding method for this library is to use BONAMI for all energy groups, enabling faster neutron resonance self-shielding calculations. The Bondarenko shielding factors for all nuclides are computed with the NR approximation. If the Bondarenko approach is not appropriate, self-shielding calculations can be done with the CENTRM module. The 200n-47g libraries have dose factor and response function data shown in Table 10.1.14 which are consistent with previous SCALE shielding libraries.

The fine-group coupled libraries were validated by performing radiation transport calculations with the SCALE shielding sequence MAVRIC for several shielding benchmark calculations [XSLibWDP+09]. The calculated results for transmission/attenuation values and spectral results matched experimental measurements well. Overall, the results obtained with using the 200n-47g coupled library demonstrate the effectiveness of the SCALE methods and data for shielding applications.

Table 10.1.12: Energy boundaries for the 200 neutron group structure.

Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)
1	2.0000E+07	42	2.7253E+06	83	3.3373E+05	124	2.3579E+04	165	5.0435E+00
2	1.9640E+07	43	2.5924E+06	84	3.0197E+05	125	2.1875E+04	166	3.9279E+00
3	1.7332E+07	44	2.4660E+06	85	2.9849E+05	126	1.9305E+04	167	3.0590E+00
4	1.6905E+07	45	2.3852E+06	86	2.9721E+05	127	1.5034E+04	168	2.3824E+00
5	1.6487E+07	46	2.3653E+06	87	2.9452E+05	128	1.1709E+04	169	1.8554E+00
6	1.5683E+07	47	2.3457E+06	88	2.8725E+05	129	1.0595E+04	170	1.4450E+00
7	1.4918E+07	48	2.3069E+06	89	2.7324E+05	130	9.1188E+03	171	1.3000E+00
8	1.4550E+07	49	2.2313E+06	90	2.4724E+05	131	7.1017E+03	172	1.1253E+00
9	1.4191E+07	50	2.1225E+06	91	2.3518E+05	132	5.5308E+03	173	1.0800E+00
10	1.3840E+07	51	2.0190E+06	92	2.2371E+05	133	4.3074E+03	174	1.0400E+00
11	1.3499E+07	52	1.9205E+06	93	2.1280E+05	134	3.7074E+03	175	1.0000E+00
12	1.2840E+07	53	1.8268E+06	94	2.0242E+05	135	3.3546E+03	176	8.7643E-01
13	1.2523E+07	54	1.7377E+06	95	1.9255E+05	136	3.0354E+03	177	8.0000E-01
14	1.2214E+07	55	1.6530E+06	96	1.8316E+05	137	2.7465E+03	178	6.8256E-01
15	1.1618E+07	56	1.5724E+06	97	1.7422E+05	138	2.6126E+03	179	6.2506E-01
16	1.1052E+07	57	1.4957E+06	98	1.6573E+05	139	2.4852E+03	180	5.3158E-01
17	1.0513E+07	58	1.4227E+06	99	1.5764E+05	140	2.2487E+03	181	5.0000E-01
18	1.0000E+07	59	1.3534E+06	100	1.4996E+05	141	2.0347E+03	182	4.1399E-01
19	9.5123E+06	60	1.2874E+06	101	1.4264E+05	142	1.5846E+03	183	3.6680E-01
20	9.0484E+06	61	1.2246E+06	102	1.3569E+05	143	1.2341E+03	184	3.2500E-01
21	8.6071E+06	62	1.1648E+06	103	1.2907E+05	144	9.6112E+02	185	2.7500E-01
22	8.1873E+06	63	1.1080E+06	104	1.2277E+05	145	7.4852E+02	186	2.2500E-01
23	7.7880E+06	64	1.0026E+06	105	1.1679E+05	146	5.8295E+02	187	1.8400E-01
24	7.4082E+06	65	9.6164E+05	106	1.1109E+05	147	4.5400E+02	188	1.5000E-01
25	7.0469E+06	66	9.0718E+05	107	9.8037E+04	148	3.5357E+02	189	1.2500E-01
26	6.7032E+06	67	8.6294E+05	108	8.6517E+04	149	2.7536E+02	190	1.0000E-01
27	6.5924E+06	68	8.2085E+05	109	8.2503E+04	150	2.1445E+02	191	7.0000E-02
28	6.3763E+06	69	7.8082E+05	110	7.9499E+04	151	1.6702E+02	192	5.0000E-02
29	6.0653E+06	70	7.4274E+05	111	7.1998E+04	152	1.3007E+02	193	4.0000E-02
30	5.7695E+06	71	7.0651E+05	112	6.7379E+04	153	1.0130E+02	194	3.0000E-02
31	5.4881E+06	72	6.7206E+05	113	5.6562E+04	154	7.8893E+01	195	2.1000E-02
32	5.2205E+06	73	6.3928E+05	114	5.2475E+04	155	6.1442E+01	196	1.4500E-02
33	4.9659E+06	74	6.0810E+05	115	4.6309E+04	156	4.7851E+01	197	1.0000E-02
34	4.7237E+06	75	5.7844E+05	116	4.0868E+04	157	3.7266E+01	198	5.0000E-03
35	4.4933E+06	76	5.5023E+05	117	3.4307E+04	158	2.9023E+01	199	2.0000E-03
36	4.0657E+06	77	5.2340E+05	118	3.1828E+04	159	2.2603E+01	200	5.0000E-04
37	3.6788E+06	78	4.9787E+05	119	2.8501E+04	160	1.7604E+01		1.0000E-05
38	3.3287E+06	79	4.5049E+05	120	2.7000E+04	161	1.3710E+01		
39	3.1664E+06	80	4.0762E+05	121	2.6058E+04	162	1.0677E+01		
40	3.0119E+06	81	3.8774E+05	122	2.4788E+04	163	8.3153E+00		
41	2.8651E+06	82	3.6883E+05	123	2.4176E+04	164	6.4760E+00		

Table 10.1.13: Energy boundaries for the 47 gamma group structure.

Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)
1	2.0000E+07	11	5.0000E+06	21	1.8000E+06	31	7.0000E+05	41	1.0000E+05
2	1.4000E+07	12	4.5000E+06	22	1.6600E+06	32	6.0000E+05	42	7.5000E+04
3	1.2000E+07	13	4.0000E+06	23	1.5700E+06	33	5.1200E+05	43	7.0000E+04
4	1.0000E+07	14	3.5000E+06	24	1.5000E+06	34	5.1000E+05	44	6.0000E+04
5	8.0000E+06	15	3.0000E+06	25	1.4400E+06	35	4.5000E+05	45	4.5000E+04
6	7.5000E+06	16	2.7500E+06	26	1.3300E+06	36	4.0000E+05	46	3.0000E+04
7	7.0000E+06	17	2.5000E+06	27	1.2000E+06	37	3.0000E+05	47	2.0000E+04
8	6.5000E+06	18	2.3500E+06	28	1.0000E+06	38	2.6000E+05		1.0000E+04
9	6.0000E+06	19	2.1500E+06	29	9.0000E+05	39	2.0000E+05		
10	5.5000E+06	20	2.0000E+06	30	8.0000E+05	40	1.5000E+05		

Table 10.1.14: Available dose functions on the coupled neutron-gamma libraries.

MAT	MT	Description
999	1	1/v total cross section normalized to 1.0 at 0.0253 eV 1/v absorption cross section normalized to 1.0 at 0.0253 eV capture, same as MT 1 and 27 Radiative capture, same as MT 1 and 27
	27	
	101	
	102	
900	9032	International Commission on Radiation Units and Measurements, Report 44 (ICRU-44), Table B.3 (air) Kerma (Gy/h)/(neutron/cm^2/s)
	9027	Henderson conversion from neutron flux to absorbed dose rate in tissue (rad/h)/(neutrons/cm^2/s)
	9033	ICRU-44, Table B.3 (air) Kerma (rad/h)/(neutron/cm^2/s)
	9034	Ambient dose equivalent (ICRU-57, Table A.42) (Sv/h)/(neutron/cm^2/s)
	9035	Ambient dose equivalent (ICRU-57, Table A.42) (rem/h)/(neutron/cm^2/s)
	9036	Effective dose (ICRU-57, Table A.41) (Sv/h)/(neutron/cm^2/s)
	9037	Effective dose (ICRU-57, Table A.41) (rem/h)/(neutron/cm^2/s)
	9029	American National Standards Institute (ANSI) standard (1977) neutron flux-to-dose rate factors (rem/h)/(neutron/cm^2/s)
	9031	ANSI standard (1991) neutron flux-to-dose rate factors (rem/h)/(neutron/cm^2/s)
	9502	Henderson conversion factors (rad/h)/(photons/cm^2/s)
	9503	Claiborne-Trubey conversion factors (rad/h)/(photons/cm^2/s)
	9504	ANSI standard (1977) gamma flux-to-dose rate factors (rem/h)/(photons/cm^2/s)
	9505	ANSI standard (1991) gamma flux-to-dose rate factors (rem/h)/(photons/cm^2/s)
	9506	ICRU-57 Table A.21 (air) Kerma (Gy/h)/(photons/cm^2/s)

continues on next page

Table 10.1.14 – continued from previous page

MAT	MT	Description
	9507	ICRU-57 Table A.21 (air) Kerma (rad/h)/(photons/cm^2/s)
	9508	Ambient dose equivalent (ICRU-57 Table A.21) (Sv/h)/(photons/cm^2/s)
	9509	Ambient dose equivalent (ICRU-57 Table A.21) (rem/h)/(photons/cm^2/s)
	9510	Effective dose (ICRU-57 Table A.17) (Sv/h)/(photons/cm^2/s)
	9511	Effective dose (ICRU-57 Table A.17) (rem/h)/(photons/cm^2/s)

10.1.2.5 The 28N-19G shielding libraries (V7.1-28N19G)

In addition to the fine-group shielding library, SCALE has a broad-group ENDF/B-VII.1 library. The ENDF/B-VII.1 library has 28 neutron and 19 gamma groups. The change in the neutron group structure (as compared with the SCALE 6.2 27 group structure) was necessary to allow for thermal up-scatter up to 5 eV; the gamma groups are identical in both libraries. The broad-group library was primarily developed to perform adjoint discrete ordinates calculations needed to prepare importance maps and biased source distributions for biasing forward Monte Carlo shielding calculations with the fine-group libraries in the MAVRIC shielding analysis sequence. The constituents of the 278n-19g library and the processing methods are the same as the 200n-47g library. The 28 neutron and 19 gamma group structures are provided in Table 10.1.15 and Table 10.1.16, respectively. The broad-group library has the same dose factor information described in Table 10.1.14 for the fine group library.

Table 10.1.15: Energy boundaries for the 28 neutron group structure.

Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)
1	2.0000E+07	7	4.0762E+05	13	2.9023E+01	19	1.1253E+00	25	1.0000E-01
2	6.3763E+06	8	1.1109E+05	14	1.0677E+01	20	1.0000E+00	26	5.0000E-02
3	3.0119E+06	9	1.5034E+04	15	5.0000E+00	21	8.0000E-01	27	3.0000E-02
4	1.8268E+06	10	3.0354E+03	16	3.0590E+00	22	4.1399E-01	28	1.0000E-02
5	1.4227E+06	11	5.8295E+02	17	1.8554E+00	23	3.2500E-01		1.0000E-05
6	9.0718E+05	12	1.0130E+02	18	1.3000E+00	24	2.2500E-01		

Table 10.1.16: Energy boundaries for the 19 gamma group structure.

Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)	Grp	Energy(eV)
1	2.0000E+07	5	5.0000E+06	9	2.0000E+06	13	8.0000E+05	17	2.0000E+05
2	1.0000E+07	6	4.0000E+06	10	1.6600E+06	14	6.0000E+05	18	1.0000E+05
3	8.0000E+06	7	3.0000E+06	11	1.3300E+06	15	4.0000E+05	19	4.5000E+04
4	6.5000E+06	8	2.5000E+06	12	1.0000E+06	16	3.0000E+05		1.0000E+04

10.1.2.6 The continuous-energy libraries

The ENDF/B-VII.1 and ENDF/B-VIII.0 CE libraries are general-purpose libraries used for both criticality calculations and shielding calculations, although the former do not use the photon data in the libraries. These libraries have CE data for the same nuclides as the MG libraries. CE data for each nuclide are stored in individual files contained in the SCALE data directory. The CE libraries were generated using AMPX to an energy-mesh tolerance of 0.1%; i.e., the data value at any intermediate energy point can be interpolated linearly within an error of 0.1%. Cross section data and kinematic data are provided for all reactions given in the ENDF evaluations and are identical to the reactions available on the MG libraries. Kinematic data are given for a range of incident energies as marginal probability distributions over the exit angles and conditional probability distributions over the exit energies in the laboratory system. Usually, 32 equiprobable exit angle bins are used for the conditional probability distribution, except if the distribution is isotropic or can be described with fewer exit angles. For elastic and discrete inelastic reaction, a larger number of exit angles are used as needed to accurately describe the kinematic in the laboratory system. Gamma production kinematic data are provided if available for nuclides that provide gamma production kinematic data. If gamma production data are present, sections for each discrete photon and the continuum are given along with the yield for each of the section. The incident neutron CE libraries are generated at temperature 293 K, 565 K, 600 K, 900 K, 1200 K, 200 K, and 2400 K. The kinematic data are not temperature dependent except for thermal moderators, which are included at the temperatures provided by the evaluator (see Table 10.1.3 for a list of thermal moderator nuclides and the list of available temperatures). Probability tables for the unresolved resonance range are provided, if the evaluations included unresolved resonance data [XSLibDL04]. Probability tables are available at 293 K, 565 K, 600 K, 900 K, 1200 K, and 2400 K. For neutron transport calculations, the CE cross sections and thermal kernels are automatically interpolated to the specified temperature using the methods described in [XSLibWDP+09].

For use in CE_MONACO, CE libraries for incident gammas are also available, based on ENDF/B-VII.1 ENDF/B-VIII.0 data, respectively. Only temperature independent cross section and kinematic data are available in the incident gamma libraries.

10.1.2.7 Gleaning Data from the Multigroup Libraries

As mentioned in Section 11.1, "AMPX LIBRARY UTILITY MODULES," the AMPX module rade can be used to print useful information, such as the available neutron targets, neutron energy group structure, and gamma energy group structure (when present). Please see the AMPX manual :cite: *XSL-wiarda_ampx-2000_2015* for more detailed instructions explaining the input for the relevant AMPX modules.

10.1.3 APPENDICES

10.1.3.1 MT Reaction Types on SCALE Cross-Section Libraries

ENDF reaction types (MT) are identified by an integer number from 1 through 999. Table 10.1.17 includes a copy of information taken from Ref. 8. Within SCALE, MT numbers are used to identify reactions on various input and output arrays.

MT numbers greater than 999 are additional SCALE-specific identifiers that may appear in SCALE cross-section libraries. These reaction types are not ENDF reaction types but may be added by various preprocessing codes in the SCALE and/or AMPX systems.

These reactions are listed in Table 10.1.18.

The dose conversion factors and their corresponding MT numbers in the SCALE shielding cross-section libraries can be found in Table 10.1.14.

Table 10.1.17: ENDF MT Reaction Types.

MT	Reaction	Description
1	(n,total)	Neutron total cross sections. Sum of MT=2, 4, 5, 11, 16-18, 22-26, 28-37, 41-42, 44-45, 102-117.
2	(z,z_0)	Elastic scattering cross section for incident particles.
3	(z,nonelastic)	Nonelastic neutron cross section. Sum of MT=4, 5, 11, 16-18, 22-26, 28-37, 41-42, 44-45, 102-117.
4	(z,n)	Production of one neutron in the exit channel. Sum of the MT=50-91.
5	(z,anything)	Sum of all reactions not given explicitly in another MT number. This is a partial reaction to be added to obtain MT=1.
6-9		Not allowed in version 6.
10	(z,continuum)	Total continuum reaction; includes all continuum reactions and excludes all discrete reactions.
11	(z,2nd)	Production of two neutrons and a deuteron, plus a residual.
12-15		(Unassigned)
16	(z,2n)	Production of two neutrons and a residual [XS1]. Sum of MT=875-891, if they are present.
17	(z,3n)	
18	(z,fission)	
19	(n,f)	
20	(n,nf)	Second-chance fission [XS2].
21	(n,2nf)	Third-chance fission[XS2].
22	(z,nα)	Production of a neutron and an alpha particle, plus a residual.
23	(n,n3α	Production of a neutron and three alpha particles, plus a residual.
24	(z,2nα)	Production of two neutrons and an alpha particle, plus a residual.
25	(z,3nα)	Production of three neutrons and an alpha particle, plus a residual.
26		Not allowed in version 6.
27	(n,abs)	Absorption; sum of MT=18 and MT=102 through MT=117.
28	(z,np)	Production of a neutron and a proton, plus a residual.
29	(z,n2α)	Production of a neutron and two alpha particles, plus a residual.
30	(z,2n2α)	Production of two neutrons and two alpha particles, plus a residual.
31		Not allowed for version 6.
32	(z,nd)	Production of a neutron and a deuteron, plus a residual.
33	(z,nt)	Production of a neutron and a triton, plus a residual.
34	(z,n^{3}He)	Production of a neutron and a ^{3}He particle, plus a residual.
35	(z,nd2α)	Production of a neutron, a deuteron, and 2 alpha particles, plus a residual.
36	(z,nt2α)	Production of a neutron, a triton, and 2 alpha particles, plus a residual.
37	(z,4n)	Production of 4 neutrons, plus a residual.
38	(n,3nf)	Fourth-chance fission cross section[2].
39		Not allowed for version 6.
40		Not allowed for version 6.
41	(z,2np)	Production of 2 neutrons and a proton, plus a residual.
42	(z,3np)	Production of 3 neutrons and a proton, plus a residual.
43		(Unassigned)
44	(z,n2p)	Production of a neutron and 2 protons, plus a residual.

continues on next page

Table 10.1.17 – continued from previous page

45	(z,npα)	Production of a neutron, a proton, and an alpha particle, plus a residual.
46-49		Not allowed in Version 6.
50	(y,n_0)	Production of a neutron, leaving the residual nucleus in the ground state.
51	(z,n_1)	Production of a neutron, with residual in the 1st excited state.
52	(z,n_2)	Production of a neutron, with residual in the 2nd excited state.
	...	
	...	
90	(z,n_{40})	Production of a neutron, with residual in the 40th excited state.
91	(z,n_c)	Production of a neutron in the continuum not included in the above discrete representation.
92-100		(Unassigned)
101	(n,disap)	Neutron disappearance; equal to sum of MT=102-117.
102	(z,:math:*gamma*)	Radiative capture.
103	(z,p)	Production of a proton, plus a residual. Sum of MT=600-649, if they are present.
104	(z,d)	Production of a deuteron, plus a residual. Sum of MT=650-699, if they are present.
105	(z,t)	Production of a triton, plus a residual. Sum of MT=700-749, if they are present.
106	(z,^{3}He)	Production of a ^{3}He particle plus a residual. Sum of MT=750-799, if they are present.
107	(z,α)	Production of an alpha particle, plus a residual. Sum of MT=800-849, if they are present.
108	(z,2α)	Production of 2 alpha particles, plus a residual.
109	(z,3α)	Production of 3 alpha particles, plus a residual.
110		(Unassigned)
111	(z,2p)	Production of 2 protons, plus a residual.
112	(z,pα)	Production a proton and an alpha particle, plus a residual.
113	(z,t2α)	Production of a triton and 2 alpha particles, plus a residual.
114	(z,d2α)	Production of a deuteron and 2 alpha particles, plus a residual.
115	(z,pd)	Production of proton and a deuteron, plus a residual.
116	(z,pt)	Production of proton and a triton, plus a residual.
117	(z,dα)	Production of deuteron and an alpha particle, plus a residual.
118-119		(Unassigned)
120		Not allowed for version 6.
121-150		(Unassigned)
151	(n,RES)	Resonance parameters that can be used to calculate cross sections at different temperatures in the resolved and unresolved energy regions.
152-200		(Unassigned)
201	(z,Xn)	Total neutron production.
202	(z,X γ)	Total gamma production.
203	(z,Xp)	Total proton production.

continues on next page

Table 10.1.17 – continued from previous page

204	(z,Xd)	Total deuteron production.
205	(z,Xt)	Total triton production.
206	(z,X ^{3}He)	Total ^{3}He production.
207	(z,Xα)	Total alpha particle production.
208	(z,X π^+)	Total π^+ production.
209	(z,X π^0)	Total pi^0 production.
210	(z,X π^-)	Total π^- production.
211	(z,X :math:mu^+`)	Total μ^+ production.
212	(z,X μ^-)	Total μ^- production.
213	(z,Xκ^+)	Total κ^+ production.
214	(z,Xκ^0_{long})	Total κ^0_{long} production.
215	(z,Xκ^0_{short})	Total κ^0_{short} production.
216	(z,Xκ^-)	Total κ^- production.
217	(z,Xp)	Total anti-proton production.
218	(z,Xn)	Total anti-neutron production.
219-250		(Unassigned)
251	(n,...)	$\bar{\mu}_L$, average cosine of the scattering angle (laboratory system) for elastic scattering of neutrons.
252	(n,...)	ξ, average logarithmic energy decrement for elastic scattering of neutrons.
253	(n,...)	γ, average of the square of the logarithmic energy decrement divided by twice the average logarithmic energy decrement, for elastic scattering of neutrons.
254-300		(Unassigned)
301-450	(z,...)	Energy release parameters,$E * \sigma$, for total and partial cross sections; MT= 300 plus the reaction MT number, *e.g.*, MT=302 is the elastic scattering kerma.
451	(z,...)	Heading or title information; given in File 1 only.
452	(z,...)	$\bar{V}_T$, average total (prompt plus delayed) number of neutrons released per fission event.
453		(Unassigned)
454	(z,...)	Independent fission product yield data.
455	(z,...)	$\bar{V}d$, average number of delayed neutrons released per fission event.
456	(z,...)	$\bar{V}p$, average number of prompt neutrons released per fission event.
457	(z,...)	Radioactive decay data.
458	(n,...)	Energy release in fission for incident neutrons.
459	(z,...)	Cumulative fission product yield data.
460-464		(Unassigned)
465-466		Not allowed in version 6.
467-499		(Unassigned)

continues on next page

Table 10.1.17 – continued from previous page

500		Total charged-particle stopping power.
501		Total photon interaction.
502		Photon coherent scattering.
503		(Unassigned)
504		Photon incoherent scattering.
505		Imaginary scattering factor.
506		Real scattering factor.
507-514		(Unassigned)
515		Pair production, electron field.
516		Pair production; sum of MT=515, 517.
517		Pair production, nuclear field.
518		Not allowed in version 6.
519-521		(Unassigned)
522		Photoelectric absorption.
523		Photo-excitation cross section.
524-525		(Unassigned)
526		Electro-atomic scattering.
527		Electro-atomic bremsstrahlung.
528		Electro-atomic excitation cross section.
529-531		(Unassigned)
532		Not allowed in version 6.
533		Atomic relaxation data.
534	K	(1s1/2) subshell photoelectric or electro-atomic cross section.
535	L1	(2s1/2) subshell photoelectric or electro-atomic cross section.
536	L2	(2p1/2) subshell photoelectric or electro-atomic cross section.
537	L3	(2p3/2) subshell photoelectric or electro-atomic cross section.
538	M1	(3s1/2) subshell photoelectric or electro-atomic cross section.
539	M2	(3p1/2) subshell photoelectric or electro-atomic cross section.
540	M3	(3p3/2) subshell photoelectric or electro-atomic cross section.
541	M4	(3d3/2) subshell photoelectric or electro-atomic cross section.
542	M5	(3d5/2) subshell photoelectric or electro-atomic cross section.
543	N1	(4s1/2) subshell photoelectric or electro-atomic cross section.
544	N2	(4p1/2) subshell photoelectric or electro-atomic cross section.
545	N3	(4p3/2) subshell photoelectric or electro-atomic cross section.
546	N4	(4dp3/2) subshell photoelectric or electro-atomic cross section.
547	N5	(4d5/2) subshell photoelectric or electro-atomic cross section.
548	N6	(4f5/2) subshell photoelectric or electro-atomic cross section.
549	N7	(4f7/2) subshell photoelectric or electro-atomic cross section.
550	O1	(5s1/2) subshell photoelectric or electro-atomic cross section.
551	O2	(5p1/2) subshell photoelectric or electro-atomic cross section.

continues on next page

Table 10.1.17 – continued from previous page

552	O3	(5p3/2) subshell photoelectric or electro-atomic cross section.
553	O4	(5d3/2) subshell photoelectric or electro-atomic cross section.
554	O5	(5d5/2) subshell photoelectric or electro-atomic cross section.
555	O6	(5f5/2) subshell photoelectric or electro-atomic cross section.
556	O7	(5f7/2) subshell photoelectric or electro-atomic cross section.
557	O8	(5g7/2) subshell photoelectric or electro-atomic cross section.
558	O9	(5g9/2) subshell photoelectric or electro-atomic cross section.
559	P1	(6s1/2) subshell photoelectric or electro-atomic cross section.
560	P2	(6p1/2) subshell photoelectric or electro-atomic cross section.
561	P3	(6p3/2) subshell photoelectric or electro-atomic cross section.
562	P4	(6d3/2) subshell photoelectric or electro-atomic cross section.
563	P5	(6d5/2) subshell photoelectric or electro-atomic cross section.
564	P6	(6f5/2) subshell photoelectric or electro-atomic cross section.
565	P7	(6f7/2) subshell photoelectric or electro-atomic cross section.
566	P8	(6g7/2) subshell photoelectric or electro-atomic cross section.
567	P9	(6g9/2) subshell photoelectric or electro-atomic cross section.
568	P10	(6h9/2) subshell photoelectric or electro-atomic cross section.
569	P11	(6h11/2) subshell photoelectric or electro-atomic cross section.
570	Q1	(7s1/2) subshell photoelectric or electro-atomic cross section.
571	Q2	(7p1/2) subshell photoelectric or electro-atomic cross section.
572	Q3	(7p3/2) subshell photoelectric or electro-atomic cross section.
573-599		(Unassigned)
600	(z,p_0)	Production of a proton leaving the residual nucleus in the ground state.
601	(z,p_1)	Production of a proton, with residual in the 1st excited state.
602	(z,p_2)	Production of a proton, with residual in the 2nd excited state.
603	(z,p_3)	Production of a proton, with residual in the 3rd excited state.
604	(z,p_4)	Production of a proton, with residual in the 4th excited state.
	...	
	...	
649	(z,p_c)	Production of a proton in the continuum not included in the above discrete representation.
650	(z,d_0)	Production of a deuteron leaving the residual nucleus in the ground state.
651	(z,d_1)	Production of a deuteron, with the residual in the 1st excited state.
652	(z,d_2)	Production of a deuteron, with the residual in the 2nd excited state.
	...	
	...	
699	(z,d_c)	Production of a deuteron in the continuum not included in the above discrete representation.
700	(z,t_0)	Production of a triton leaving the residual nucleus in the ground state.
701	(z,t_1)	Production of a triton, with residual in the 1st excited state.
702	(z,t_2)	Production of a triton, with residual in the 2nd excited state.
	...	
	...	

continues on next page

Table 10.1.17 – continued from previous page

749	(z,t_c)	Production of a triton in the continuum not included in the above discrete representation.
750	(n,:sup`3`He_1)	Production of a ^{3}He particle leaving the residual nucleus in the ground state.
751	(n,:sup`3`He_2)	Production of a ^{3}He, with residual in the 1st excited state.
	...	
	...	
799	(n,:sup`3`He_c)	Production of a ^{3}He in the continuum not included in the above discrete representation.
800	(z,:math:*alpha_0*)	Production of an alpha particle leaving the residual nucleus in the ground state.
801	(z,:math:*alpha_1*)	Production of an alpha particle, with residual in the 1st excited state.
	...	
	...	
849	(z,:math:*alpha_c*)	Production of an alpha particle in the continuum not included in the above discrete representation.
850		(Unassigned)
851-870		Lumped reaction covariances.
871-874		(Unassigned)
875	(z,2n_0)	Production of 2 neutrons with residual in the ground state.
876	(z,2n_1)	Production of 2 neutrons with residual in the 1st excited state.
	...	
891	(z,2n_c)	Production of 2 neutrons in the continuum not included in the above discrete representation.
892-999		(Unassigned)

Table 10.1.18: SCALE-Specific MT Reaction Types

SCALE MT	**Description**
1000	Transport cross section based on the outscatter approximation. See XSDRNPM chapter, section **Outscatter approximation (inconsistent method)**.
1001	Transport cross section based on the inscatter approximation. See XSDRNPM chapter, section **Inscatter approximation (consistent method)**
1007	Thermal scattering matrix
1008	Elastic part of thermal scattering matrix
1018	Fission spectrum
1019	First chance fission spectrum
1020	Second chance fission spectrum
1021	Third chance fission spectrum
1038	Fourth chance fission spectrum
1099	Group integral of the weight function
1111	Flux moment (P_1) weighted total cross section

continues on next page

Table 10.1.18 – continued from previous page

1112	Flux moment (P_2) weighted total cross section
1113	Flux moment (P_3) weighted total cross section
1114	Flux moment (P_4) weighted total cross section
1115	Flux moment (P_5) weighted total cross section
1116	Flux moment (P_6) weighted total cross section
1117	Flux moment (P_7) weighted total cross section
1118	Flux moment (P_8) weighted total cross section
1119	Flux moment (P_9) weighted total cross section
1452	Product of $\bar{V}T$ times the fission cross section
1456	Product of $\bar{V}_p$ times the fission cross section
1455	Product of $\bar{v}_d$ times the fission cross section
1500-1501	Same as 1000, 1001 except for gamma-ray cross sections
1527	Gamma-ray energy absorption coefficient factors
2006	Non-absorption collision probability (CE libraries only)
2016	Probability of emitting two neutrons (CE libraries only)
2017	Probability of emitting three neutrons (CE libraries only)
2018	Fission probability (CE libraries only)
2022	Within-group scattering cross section
2027	Absorption probability (CE libraries only)
4561	$\bar{V}_p$ for first chance fissions
4562	$\bar{V}_p$ for second chance fissions
4563	$\bar{V}_p$ for third chance fissions
4564	$\bar{V}_p$ for fourth chance fissions

10.2 ORIGEN DATA RESOURCES

ORIGEN data resources include nuclear decay data, multigroup neutron reaction cross sections, neutron-induced fission product yields, and decay emission data for photons, neutrons, alpha particles and beta particles. The nuclear decay data are based primarily on ENDF/B-VII.1 evaluations. The multigroup nuclear reaction cross section libraries now include evaluations from the JEFF-3.0/A neutron activation file containing data for 774 target nuclides, more than 12,000 neutron-induced reactions, and more than 20 different reaction types below 20 MeV, provided in various energy group structures. Energy-dependent ENDF/B-VII.0-based fission product yields are available for 30 fissionable actinides. Gamma-ray and x-ray emission data libraries are based on ENDF/B-VII.1. The photon libraries contain discrete photon line energy and intensity data for decay gamma and x-rays emission for 1,132 radionuclides, prompt and delayed continuum spectra for spontaneous fission, (α, n) reactions in oxide fuel, and bremsstrahlung from decay beta (electron and positron) particles slowing down in either a UO_2 fuel or water matrix. Methods and data libraries used to calculate the neutron yields and energy spectra for spontaneous fission, (α, n) reactions, and delayed neutron emission are adopted from the SOURCES4C code. Capabilities to calculate the beta and alpha particle emission source and spectra have also been added.

10.2.1 ACKNOWLEDGEMENTS

Development and testing of ORIGEN data resources, libraries, and methods have been sponsored by many organizations including the US Nuclear Regulatory Commission (NRC), the US Department of Energy (DOE), and nuclear power and research institutions.

10.2.2 VERSION INFORMATION

Following is a description of the data resources available for use with ORIGEN in different SCALE versions. Methodologies and algorithms used in applying the data are described in the ORIGEN chapter.

10.2.2.1 Version 6.3 (2021)

Data lead(s): A. Holcomb and W. Wieselquist

Nuclear data in ORIGEN has had minor updates for SCALE 6.3. The old ENDF/B-VII.0 libraries have been removed. There are new libraries for 302-groups (for fast-spectrum systems) and 1597-groups as an very fine group option for any spectrum. Data for recoverable energy per capture for Gd-155 and Gd-157 has been updated from 5 MeV to 8.5360 and 7.9370 MeV, respectively.

10.2.2.2 Version 6.2 (2016)

Data lead(s): I. C. Gauld, D. Wiarda, M. Pigni, and W. Wieselquist

Nuclear data in ORIGEN are unchanged from SCALE 6.1.3 except for the modification of independent fission yields for thermal fission of ^{235}U and ^{241}Pu and fast fission of ^{238}U to provide greater compatibility between the direct and cumulative fission yields when using the updated decay data from ENDF/B-VII.1. Additionally, ORIGEN no longer has its own independent source of nuclide mass and abundance data and now relies on the SCALE Standard Composition library so that there is consistency in this data across SCALE. D. Mueller and W. Wieselquist are acknowledged for testing of the new yield data. W. Wieselquist and S. Hart are acknowledged for the revision of this chapter.

10.2.2.3 Version 6.1.3 (2011)

Data lead (s): I. C. Gauld and D. Wiarda

SCALE 6.1 represented a complete revision and update of the nuclear data available in ORIGEN. The following is a summary from the SCALE 6.1 manual.

> The ORIGEN data libraries include nuclear decay data, neutron reaction cross sections, neutron induced fission product yields, delayed gamma-ray emission data, and neutron emission data. The nuclear decay data libraries have been updated based on ENDF/B-VII evaluations and expanded to include 903 activation products and structural materials, 174 actinides, and 1149 fission products. The cross section libraries have been revised using evaluations from the JEFF-3.0/A neutron activation file, containing data for 774 target nuclides, more than 12,000 neutron-induced reactions, and more than 20 different reaction types below 20 MeV. The JEFF-3.0/A activation file is processed into several multigroup cross section libraries, from 44 groups to 238 groups, that can be used to determine the neutron reaction transition rates in ORIGEN. Energy-dependent ENDF/B-VII fission product yields are provided for 30 fissionable actinides. Photon yield data libraries have been updated based on the most recent ENSDF nuclear structure evaluations processed using the NuDat program. The photon libraries contain discrete photon line energy and intensity data for decay gamma and x-rays emission for 982 radionuclides, prompt and equilibrium continuum fission product spectra from spontaneous fission, (α, n) reactions in oxide fuel, and bremsstrahlung from decay beta (negatron and positron) particles

slowing down in either UO_2 fuel or water matrix. Methods and data libraries used to calculate the neutron yields and energy spectra for spontaneous fission, (α, n) reactions in any matrix, and delayed neutron emission are adopted from the SOURCES code. The libraries used by ORIGEN can be coupled directly with detailed and problem-dependent physics calculations to obtain self-shielded problem-dependent cross sections based on the most recent evaluations of ENDF/B-VII. In addition, the library formats allow multiple sets of cross section data to be stored on a library to represent the changes in cross sections during irradiation.

10.2.3 INTRODUCTION

ORIGEN data resources include nuclear decay data, multigroup neutron reaction cross sections, neutron-induced fission product yields, and decay emission data for photons, neutrons, alpha particles and beta particles. The available resources are summarized in Table 10.2.1 and described in greater detail in the subsequent sections. The "Unit" column shows the corresponding unit number for use with FIDO input systems (e.g. with COUPLE).

Table 10.2.1: Available resources in ORIGEN

Description	Alias	Unit	Category	Location in SCALE data directory
ENDF/B-VII.1 decay data	decay	27	Decay	origen_data/origen.rev03.decay.data
ENDF/B-VII.0-based fission yield data	yields	17	Yield	origen_data/origen.rev05.yields.data
JEFF-3.0/A - 56g	n56	75	Reaction	origen.rev01.jeff56g
JEFF-3.0/A - 200g	n200	78	Reaction	origen.rev03.jeff200g
JEFF-3.0/A - 252g	n252	74	Reaction	origen.rev01.jeff252g
JEFF-3.0/A - 302g	n302		Reaction	origen.rev00.jeff302g
JEFF-3.0/A - 1597g	n1597		Reaction	origen.rev00.jeff1597g
Energy per fission and capture			Energy	n/a
Master photon (x-ray and gamma) emission data			Emission	origen_data/origen.rev04.mpkkxgam.data
Spontaneous fission and (α, n) reaction gamma rays			Emission	origen_data/origen.rev00.mpsfangm.data
Bremsstrahlung from beta particles slowing down in water			Emission	origen_data/origen.rev00.mpbrh2om.data
Bremsstrahlung from positrons slowing down in water			Emission	origen_data/origen.rev00.mpbrh2op.data
Bremsstrahlung from beta particles slowing down in UO_2			Emission	origen_data/origen.rev00.mpbruo2m.data igen_data/or
Bremsstrahlung from positrons			Emission	origen_data/origen.rev00.mpbruo2p.data

continues on next page

Table 10.2.1 – continued from previous page

Description	Alias	Unit	Category	Location in SCALE data directory
Neutron source emission and alpha decay data			Emission	origen_data/origen.rev01.alphdec.data rigen_data/o
Alpha particle stopping cross section expansion coeffcients			Emission	origen_data/origen.rev00.stcoeff.data
Target (α, n) product excited level branching data			Emission	origen_data/origen.rev00.alphyld.data
Target (α, n) cross section data			Emission	origen_data/origen.rev00.alphaxs.data
Beta source emission data			Emission	origen_data/origen.rev00.ensdf95beta.data

10.2.4 DECAY RESOURCE

The nuclear data stored on the decay resource is based on ENDF/B-VII.1 evaluations [Origen-Data-ResourcesCHO+11], including half-lives, decay modes and branching fractions, and recoverable energy per disintegration. Decay modes include beta (β^-), positron (β^+) and electron capture (EC), isomeric transition (IT), alpha (α), spontaneous fission (SF), delayed neutron (β^- n) emission, neutron emission (n), double beta decay ($\beta^-\beta^-$), and decay by beta and alpha emission ($\beta^-\alpha$). The decay resource also includes radiotoxicity factors based on the radioactivity concentration guides (RCGs) for air and water as defined in Part 10, Title 20, of the Code of Federal Regulations (10CFR20) [Origen-Data-Resources10c]. RCGs specify the maximum permissible concentrations of an isotope in soluble and insoluble forms for both ingestion and inhalation and for occupational and unrestricted exposure. The radiotoxicity is calculated as the dilution volume of a nuclide for cases of direct ingestion or inhalation. The values are defined to be the smaller (i.e., more toxic) of the values for soluble and insoluble forms of the isotope. The maximum permissible RCGs for air and water are the public exposure limits for adult ingestion and inhalation dose coefficients of ICRP Publication 72 [Origen-Data-ResourcesInternationalCoRPICRP77]. External exposure dose coefficients for noble gases were obtained from the Environmental Protection Agency (EPA) Federal Guidance Report 12 [Origen-Data-ResourcesUEPAEPA93]. Recoverable energy includes the delayed energy from all electron-related radiations (e.g., β^-, β^+, Auger electrons), all gamma rays, x-rays, annihilation radiations, and the average energy of all heavy charged particles and delayed neutrons. The average alpha energy includes the energy of the recoil nucleus. A part of the recoverable energy per decay not included in the ENDF/B-VII.1 values is the additional contribution from spontaneous fission. This energy was calculated as the product of the spontaneous fission branching fraction and recoverable energy per fission using a value of 200 MeV per fission and then added to the ENDF/B-VII.1 recoverable Q energy. A value of 12.56 MeV gamma energy per fission was used in computing the fraction of recoverable spontaneous fission energy from gamma rays. External Bremsstrahlung radiation is **not** included in the Q-value since the Bremsstrahlung spectrum depends on electron interactions with the medium that contains the decay nuclide. The energy from capture gamma rays accompanying (α, n) reactions is not included either since it also depends on the medium.

Appendix A describes the decay resource file format. It is important to note that the decay resource not only defines fundamental decay data, but also the complete ORIGEN nuclide set, including the "duplicates" of nuclides across sublibraries. For example, a version of ^{155}Gd is contained in both the light nuclide/activation product and fission product sublibraries. Appendix D includes the full list of the nuclides on the ORIGEN decay library "end7dec" created by COUPLE based on the current decay resource, including duplicates.

Appendix E contains a list of the fundamental decay data only, without duplicates. To consider a different set of nuclides in an ORIGEN calculation, the current process is to alter the decay resource and then regenerate the "end7dec" decay library with COUPLE. By default, all subsequent libraries created from COUPLE using problem-dependent reaction transitions are based on the "end7dec" decay library and will therefore include the modified nuclide set.

10.2.5 NEUTRON REACTION RESOURCE

The neutron cross sections defining the nuclear reaction transmutation rates use a comprehensive collection of nuclear data evaluations compiled from the JEFF-3.0/A neutron activation files [Origen-Data-ResourcesSKFK03]. The JEFF-3.0/A files contain continuous energy neutron data for 774 target nuclei, including ground and metastable excited states, and 12,617 neutron-induced reactions below 20 MeV. The JEFF-3.0/A cross section data are developed directly from the European Activation File (EAF-2003) [Origen-Data-ResourcesFKS02] formatted as standard ENDF-6 format data. JEFF-3.0/A cross sections are stored using File 3, multiplicities on File 10, and isomeric branching to different metastable levels using File 9. The evaluations include many reactions that may be important for modeling fast fission and other high-energy systems. Neutron reactions are available for 23 reaction types, including (n,n'), $(n,2n)$, $(n,3n)$, (n,f), $(n,n'\alpha)$, $(n,2n\alpha)$, $(n,3n\alpha)$, $(n,n'p)$, $(n,n2\alpha)$, $(n,n'd)$, $(n,n't)$, $\left(n,n'^{3}He\right)$, $(n,4n)$, $(n,2np)$, (n,γ), (n,p), (n,d), (n,t), $\left(n,^{3}He\right)$, (n,α), $(n,2\alpha)$, $(n,2p)$, and $(n,p\alpha)$.

The JEFF-3.0/A evaluations also include extensive compilations of energy-dependent branching fractions that define neutron reaction transitions to ground and metastable energy states. Energy-dependent branching is fully implemented in the ORIGEN cross section libraries. Implementation of the JEFF-3.0/A cross sections as ORIGEN multigroup data was accomplished by processing and collapsing the JEFF-3.0/A pointwise cross sections into a standard multigroup AMPX format using ENDF data-processing modules of the AMPX [Origen-Data-ResourcesDG02] cross section processing code system. The collapse is performed using a thermal Maxwellian-1/E-fission-1/E weighting spectrum (see Fig. 10.2.1) to provide infinite dilution multigroup cross sections.

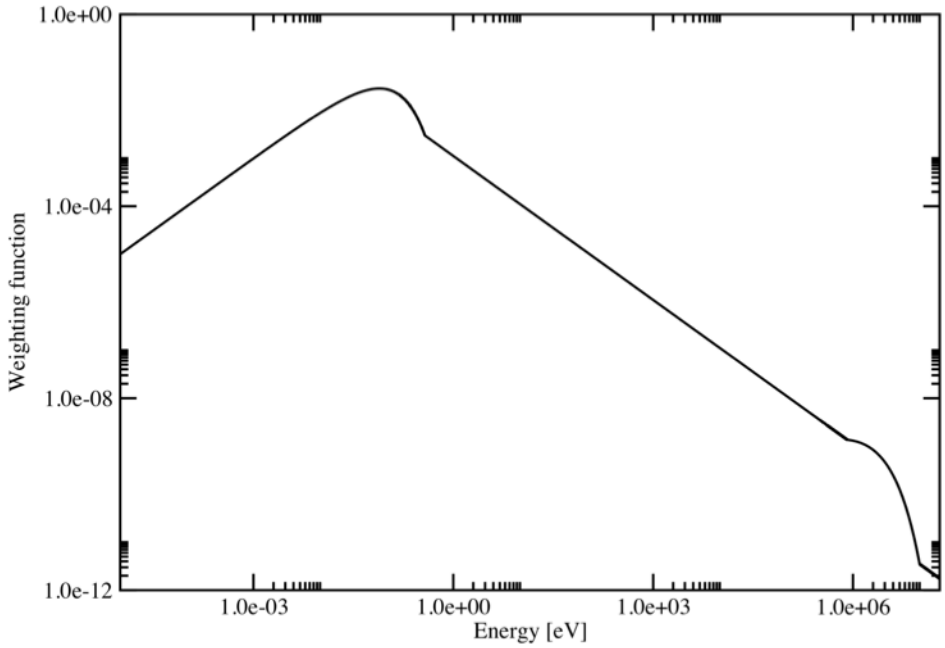


Fig. 10.2.1: Pointwise flux spectrum used to generate collapsed cross section libraries.

Neutron reactions with transitions to multiple states of the daughter product are represented using separate cross sections to the ground and metastable states. A special reaction identifier (MT') is defined for this implementation of metastable transitions as

$$\mathrm{MT}' = \mathrm{MT} * 10000 + 100 * \mathrm{LP} + \mathrm{LT} \tag{10.2.1}$$

where MT is the reaction identifier, LP is the product metastable state, and LT is the target metastable state. Using the ^{187}W(n,3n)^{185}W cross section (MT=17) as an example, the reaction identifier 170000 defines the partial cross section to the ground state of ^{185}W, and 170100 defines the cross section to metastable ^{185m}W.

Cross section data from the JEFF-3.0/A neutron activation file are first converted to point-wise cross section data, are Doppler broadened to 900K, and then they are collapsed to different group structures. The following group strucures are available in SCALE:

- 238-group neutron (thermal applications),
- 252-group neutron (thermal applications),
- 56-group neutron (thermal applications),
- 200-group neutron (fast applications and shielding),
- 47-group neutron (applications using the BUGLE shielding transport library),
- 49-group neutron (collapsed version of 238 groups),
- 44-group neutron (collapsed version of 238 groups), and
- 999-group neutron (multipurpose).

Several minor modifications were made to the JEFF-3.0/A data:

- The ^{239}Np radiative neutron capture cross section was replaced with data from ENDF/B-VII.0. Neutron capture using JEFF-3.0/A cross sections was significantly larger than ENDF/B-VII.0 due to differences in the resonance cross section region. Although experimental resonance parameters are not available for ^{239}Np, comparisons of ^{240}Pu production during irradiation [Origen-Data-ResourcesGum54] obtained using the two evaluations showed that better agreement with the experiment was obtained using the ENDF/B-VII.0 evaluation.
- The $^{241}Am(n,\gamma)$ branching fraction to the ^{242}Am ground and metastable states was replaced by the evaluation from ENDF/B-VII.0 to yield better agreement with the results of destructive radiochemical assay measurements of irradiated fuels. The branching fraction of ^{241}Am to ^{242m}Am for thermal neutron capture changed from 8.2% in JEFF-3.0/A to 10.0% in ENDF/B-VII.0.

The cross section library header record information and a complete list of nuclides in JEFF-3.0/A libraries developed for ORIGEN are provided in Appendix E.

Because JEFF-3.0/A-based libraries are formatted as standard AMPX working libraries, they can be accessed and/or manipulated using standard AMPX utility modules in SCALE. For example, multigroup cross sections may be listed using the PALEALE module. Additionally, the data may be visualized using the Fulcrum user interface. Cross section plots of the 238-group JEFF-3.0/A library are illustrated in Fig. 10.2.2 for (n,γ), (n,α), (n,2n), and (n,3n) cross sections to the ground and metastable states.

Before the cross sections in ORIGEN can be used, they must be collapsed with a user-defined multigroup flux to a one-group cross section and added to the ORIGEN binary library (see the COUPLE input description).

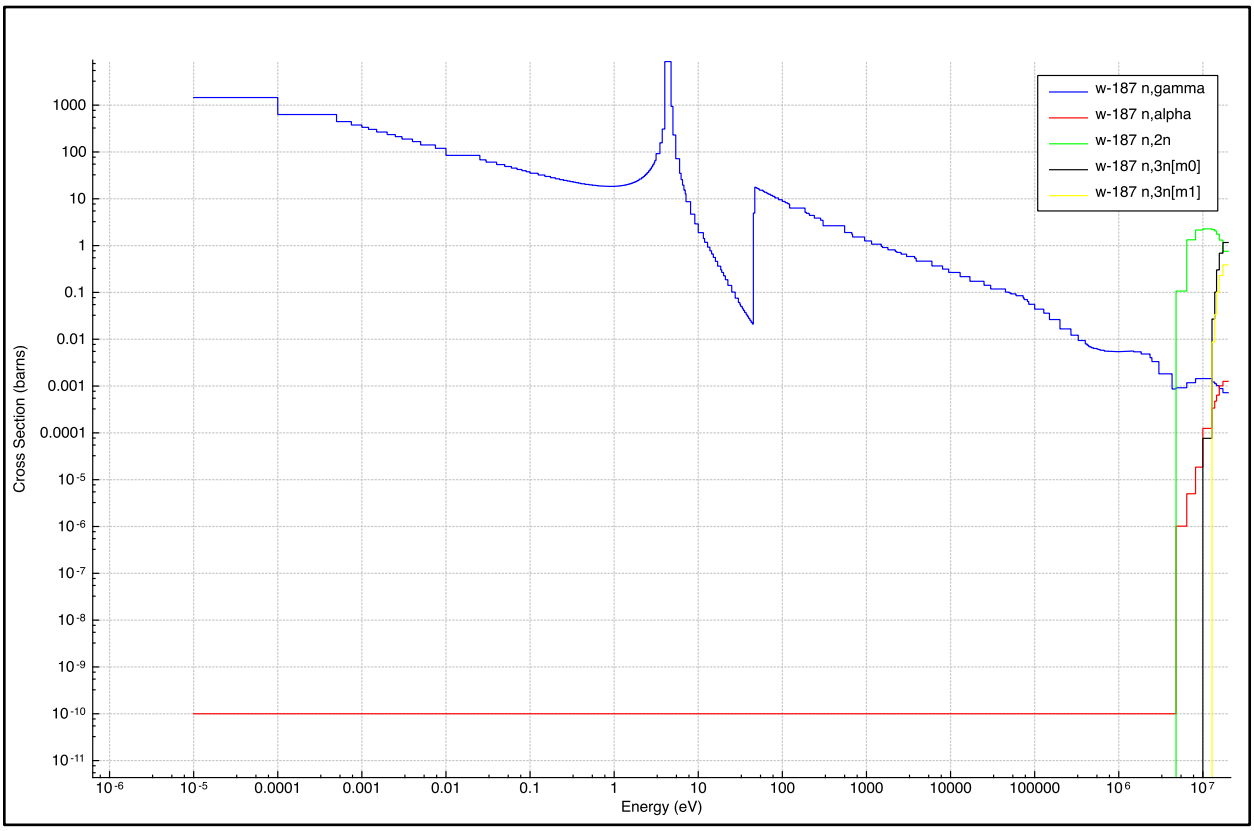


Fig. 10.2.2: 238-group JEFF-3.0/A cross sections for ^{187}W.

10.2.6 FISSION YIELD RESOURCE

The fission-yield resource contains the energy-dependent direct yields of each fission product for 30 fissionable actinides, including ^{227,228,232}Th, ^{231}Pa, $^{232–238}$U, $^{238\text{-}242}$Pu, ^{241,242m,243}Am, ^{237,238}Np, $^{242\text{-}246,248}$Cm, ^{249,252}Cf, and ^{254}Es. Independent (direct) fission product yields are stored as atom percent per fission, and except for ^{235}U(thermal), ^{238}U(fast), and ^{241}Pu(thermal), they are obtained from ENDF/B-VII.0 [Origen-Data-ResourcesCOH+06] File 8 and MT=454.

Revised independent fission product yields for ^{235}U(thermal), ^{238}U(fast), and ^{241}Pu(thermal) were adopted to address inconsistencies between the direct and cumulative fission yields in ENDF/B-VII.0 caused by the use of updated nuclear decay schemes in the decay sublibrary [Origen-Data-ResourcesFWP15, Origen-Data-ResourcesPFG15]. Namely, recent changes in the decay data, particularly the delayed neutron branching fractions, result in calculated fission product concentrations that do not agree with the cumulative fission yields in the ENDF/B-VII.0 library. These issues were particularly evident for the three cited isotopes because their fissioning systems result in a preferential formation of fragments that are sensitive to the changes in the decay data. For example, a study on ^{239}Pu(thermal) showed negligible differences between cumulative yields calculated (using the recent decay data sublibrary) and the cumulative yields in ENDF/B-VII.0. Energy-dependent product yields are available for thermal, fast, and high-energy incident neutron energies. For fast fission, the value of the energy of incident neutron was modified from the value of 500 keV tabulated in ENDF/B-VII.0 to more accurately represent the relationship between the energy distribution of the neutrons causing fission and the and the fission neutron spectrum energy. For this implementation of the yield data, the effective incident neutron energy for fast fission was adjusted from 500 keV to 2.0 MeV to better reflect the average fission energy of most nuclides. The neutron energies for thermal fission (0.0253 eV) and high energy fission (14 MeV) are unchanged.

The fission product yields also include cumulative ternary yields from the JEF-2.2 fission yield library [Origen-Data-ResourcesNuclearEAgency00] for ^{3}H and ^{4}He. The nuclide ^{3}He was also added to the fission product library since it is a decay product of tritium.

Note that inclusion of fission yields for each actinide in an ORIGEN library can be controlled by the user through COUPLE. Actinides not assigned with explicit yields do not produce fission products during fission.

Table 10.2.2: Fissionable isotopes having explicit fission yields

Nuclide	**Neutron-induced fission energies**[Page 2199, 1]		
^{227}Th	Thermal		
^{229}Th	Thermal		
^{232}Th		Fast	High energy
^{231}Pa		Fast	
^{232}U	Thermal		
^{233}U	Thermal		
^{234}U		Fast	High energy
^{235}U	Thermal	Fast	High energy
^{236}U		Fast	High energy
^{237}U		Fast	
^{238}U		Fast	High energy
^{237}Np	Thermal	Fast	High energy
^{238}Np		Fast	
^{238}Pu		Fast	

continues on next page

Table 10.2.2 – continued from previous page

Nuclide	Neutron-induced fission energies[Page 2199, 1]		
^{239}Pu	Thermal	Fast	High energy
^{240}Pu	Thermal	Fast	High energy
^{241}Pu	Thermal	Fast	
^{242}Pu	Thermal	Fast	High energy
^{241}Am	Thermal	Fast	High energy
^{242m}Am	Thermal		
^{243}Am		Fast	
^{242}Cm		Fast	
^{243}Cm	Thermal	Fast	
^{244}Cm		Fast	
^{245}Cm	Thermal		
^{246}Cm		Fast	
^{248}Cm		Fast	
^{249}Cf	Thermal		
^{251}Cf	Thermal		
^{254}Es	Thermal		

10.2.7 ENERGY RESOURCE

The energy resource is a set of data defined internally to ORIGEN to compute the total power during irradiation if the flux is known, or the total flux if the power is known. The data include the energy contributed by fission and capture. The recoverable energy values taken primarily from ENDF/B evaluations are listed in Table 10.2.3 and Table 10.2.4. The recoverable energy for fission and neutron capture for nuclides not listed in the tables are assumed to be 200 MeV and 5.0 MeV, respectively.

Table 10.2.3: Recoverable energy (MeV) values for actinides

Nuclide	Fission	Capture
^{230}Th	190.00	5.010
^{232}Th	189.21	4.786
^{233}Th	190.00	6.080
^{231}Pa	190.00	5.660
^{233}Pa	189.10	5.197
^{232}U	200.00	5.930
^{233}U	191.29	6.841
^{234}U	190.30	5.297
^{235}U	194.02	6.545
^{236}U	192.80	5.124
^{238}U	198.12	4.804
^{237}Np	195.10	5.490
^{239}Np	200.00	4.970
^{238}Pu	197.80	5.550
^{239}Pu	200.05	6.533

continues on next page

[1] Neutron energy causing fission

Table 10.2.3 – continued from previous page

^{240}Pu	199.79	5.241
^{241}Pu	202.22	6.301
^{242}Pu	200.62	5.071
^{243}Pu	200.00	6.020
^{241}Am	202.30	5.529
^{242m}Am	202.29	6.426
^{243}Am	202.10	5.363
^{244}Cm	200.00	6.451
^{245}Cm	200.00	6.110

Table 10.2.4: Recoverable energy (MeV) values for activation and fission products

Nuclide	**Capture**
^{1}H	2.225
^{10}B	2.790
^{16}O	4.143
^{56}Fe	7.600
^{58}Ni	9.020
^{90}Zr	7.203
^{91}Zr	8.635
^{92}Zr	6.758
^{96}Zr	5.571
^{95}Mo	9.154
^{95}Tc	7.710
^{101}Ru	9.216
^{103}Rh	6.999
^{105}Rh	7.094
^{109}Ag	6.825
^{131}Xe	8.936
^{135}Xe	7.880
^{133}Cs	6.704
^{134}Cs	6.550
^{143}Nd	7.817
^{145}Nd	7.565
^{147}Pm	5.900
^{148}Pm	7.266
^{148m}Pm	7.266
^{147}Sm	8.140
^{149}Sm	7.982
^{150}Sm	5.596
^{151}Sm	8.258
^{152}Sm	5.867
^{153}Eu	6.444
^{154}Eu	8.167

continues on next page

Table 10.2.4 – continued from previous page

^{155}Eu	6.490

10.2.8 EMISSION RESOURCES

The two main groups for emission resources are the photon (gamma) resource, which includes beta particle emission data, and the neutron resource, which includes alpha emission data.

10.2.8.1 Gamma Emission

The resources for gamma emission are stored as separate files (see Table 10.2.5) containing the photon data associated with different modes of decay or photon production. The photon data sets include decay gamma and x-ray line-energy data, gamma rays accompanying spontaneous fission, gamma rays accompanying (α, n) reactions in oxide fuels, and Bremsstrahlung spectra from decay electrons/positrons slowing down in UO_2 and water. The photon energy spectra can be generated in any energy group structure for all activation products, actinides, and fission product nuclides with photon yield data.

Table 10.2.5: Photon data files

File name	Description
MPDKXGAM	x-ray and gamma emissions line data
MPSFANGM	spontaneous fission and (α, n) reactions
MPBRH2OM	bremsstrahlung from beta particles slowing down in water
MPBRH2OP	bremsstrahlung from positrons slowing down in water
MPBRUO2M	bremsstrahlung from beta particles slowing down in UO_2
MPBRUO2P	bremsstrahlung from positrons slowing down in UO_2

All photon data sets are constructed with the same format (see Appendix C). The majority of the photon emissions are discrete energy lines. Photon continuum data, used to represent Bremstrahlung and some other gamma-ray emission spectra, are stored at discrete energies and approximately expanded to a continuum, as needed. Gamma and x-ray yields are directly from ENDF/B-VII.1 decay files containing spectral data for decay transitions of 1,132 nuclides. A separate file contains emission spectra for gamma rays accompanying spontaneous fission and for gamma rays accompanying (α, n) reactions in oxide fuels [Origen-Data-ResourcesCHG79]. The spontaneous fission spectra combine prompt and equilibrium fission product gamma-ray components. The prompt spectrum is similar to that of ^{235}U, and the delayed fission product gamma intensity at equilibrium is about 0.75 of that from the prompt fission gamma rays. Based on measured prompt fission gamma spectra from ^{235}U, spontaneous fission spectra are computed from the following approximation:

$$N(E) \cong \begin{cases} 11.5 & 0.1 \leq E \leq 0.6\ MeV \\ 35.4e^{-1.78E} & 0.6 \leq E < 1.5\ MeV \\ 12.6e^{-1.09E} & 1.5 \leq E \leq 10.5\ MeV \\ 0 & \text{otherwise.} \end{cases} \tag{10.2.2}$$

where

N(E) = number of photons per unit energy per fission (photons/MeV per fission) at energy E, where E is the photon energy (MeV).

For medical and industrial spontaneous fission source applications, a more accurate simulation of the source may be desirable. Work has been performed on ^{252}Cf source modeling to explicity represent the fission

product generation from fission and the delayed gamma emission. In this application, the equilibrium spontaneous fission gamma spectrum was replaced with an evaluation of the ^{252}Cf prompt gamma spectrum, and the delayed fission product gamma rays was modeled explicitly in ORIGEN by generating the time-dependent fission products using ^{252}Cf spontaneous fission product yields from ENDF/B-VII.0 [Origen-Data-ResourcesIGW11]. This was performed by adding decay transitions to the ORIGEN library from the actinides to the fission products. The spectrum of gamma rays accompanying (α, n) reactions is based on reaction data for alpha interactions on ^{18}O and from studies for $^{238}PuO_2$ systems. The spectrum is computed from the approximation:

$$N(E) \cong 2.13 \cdot 10^{-8}\ e^{-1.38E} \tag{10.2.3}$$

where

> N(E) = number of photons per unit energy per alpha decay (photons/MeV per disintegration) at energy E (MeV).

The photon yields in this data set are continuum spectra represented by discrete lines with an energy width of 500 keV and range from 250 keV to 10.25 MeV.

Two photon data sets contain bremsstrahlung spectra from decay electrons and positrons slowing down in a UO_2 fuel matrix. The yields are in the form of continuum spectra represented in the data sets as discrete lines using up to 70 quasi-logarithmic spaced energy points over the energy range between 0 and 13.5 MeV. Two libraries contain bremsstrahlung spectra from decay electrons and positrons slowing down in water. Bremsstrahlung spectra were calculated using a computer program developed by Dillman *et al.* [Origen-Data-ResourcesDSF73] using beta spectra derived from **Evaluated Nuclear Structure Data Files (**ENSDF) decay data with a computer program written by Gove and Martin [Origen-Data-ResourcesGM71].

10.2.8.2 Neutron Emission

There are four neutron emission resources used by ORIGEN to calculate the neutron intensities and spectra: (1) neutron decay data, (2) an alpha-particle stopping power, (3) a target (α, n) cross section, and (4) a target (α, n) product level branching. All of the neutron data are stored in a text format with names and descriptions given in Table 10.2.6. The neutron decay data contain basic decay information for decay processes that lead to direct and indirect emission of neutrons, including spontaneous fission branching fractions, alpha decay branching fractions, delayed neutron branching fractions, alpha-particle decay energies, Watt fission spectrum parameters, and delayed neutron spectra. The stopping cross sections, (α, n) target cross sections, and product-level branching data are used in calculating the neutron yield and spectra from (α, n) reactions.

The neutron data were obtained directly from the updated SOURCES-4B code package. The sources of the neutron data are described by Shores [Origen-Data-ResourcesSho00]. An update was made to correct an error in the ^{250}Cf spontaneous fission neutron branching fraction in the neutron source decay data distributed with the SOURCES code. The ^{250}Cf branching fraction was incorrectly assigned the value from ^{252}Cf of $3.092 \cdot 10^{-2}$. A corrected value of $7.700 \cdot 10^{-4}$ from ENDF/B-VII.1 is used.

Table 10.2.6: Neutron source data libraries

File name	Description
ALPHDEC	Neutron source decay data
STCOEFF	Stopping cross section expansion coefficients
ALPHYLD	Target (α, n) product level branching
ALPHAXS	Target (α, n) cross section

The neutron source decay contains spontaneous fission data for the 49 actinides listed in Table 10.2.7. These data include the spontaneous fission branching fraction, the number of neutrons per fission (*nu*), and the watt spectrum parameters for spontaneous fission. The spontaneous fission neutron energy spectrum is approximated using spectral parameters A and B, such that

$$N(E) \cong \mathrm{C}\, e^{-\frac{E}{A}} \sinh \sqrt{\mathrm{BE}} \tag{10.2.4}$$

where E is the neutron energy and C is a normalization constant.

Table 10.2.7: Nuclides with spontaneous fission data and spectral parameters

^{230}Th	^{239}U	^{240}Pu	^{244}Am	^{250}Cm
^{232}Th	^{236}Np	^{241}Pu	^{244m}Am	^{249}Bk
^{231}Pa	^{236m}Np	^{242}Pu	^{240}Cm	^{248}Cf
^{232}U	^{237}Np	^{243}Pu	^{241}Cm	^{250}Cf
^{233}U	^{238}Np	^{244}Pu	^{242}Cm	^{252}Cf
^{234}U	^{239}Np	^{240}Am	^{243}Cm	^{254}Cf
^{235}U	^{236}Pu	^{241}Am	^{244}Cm	^{253}Es
^{236}U	^{237}Pu	^{242}Am	^{245}Cm	^{254m}Es
^{237}U	^{238}Pu	^{242m}Am	^{246}Cm	^{255}Es
^{238}U	^{239}Pu	^{243}Am	^{248}Cm	

Delayed neutron branching fractions and neutron spectra for 105 fission products are listed in Table 10.2.8. The delayed neutron spectra are tabulated in discrete 10 keV bins from 50 keV to about 2 MeV.

Table 10.2.8: Nuclides with delayed neutron emission spectral data

^{79}Zn	^{89}Br	^{97}Y	^{128}In	^{41}I
^{79}Ga	^{90}Br	^{97m}Y	^{129}In	^{42}I
^{80}Ga	^{91}Br	^{98}Y	^{129m}In	^{43}I
^{81}Ga	^{92}Br	^{98m}Y	^{130}In	^{141}Xe
^{82}Ga	^{93}Br	^{99}Y	^{131}In	^{142}Xe
^{83}Ga	^{92}Kr	^{100}Y	^{132}In	^{143}Xe
^{83}Ge	^{93}Kr	^{104}Zr	^{133}Sn	^{144}Xe
^{84}Ge	^{94}Kr	^{105}Zr	^{134}Sn	^{141}Cs
^{85}Ge	^{95}Kr	^{103}Nb	^{135}Sn	^{142}Cs
^{86}Ge	^{92}Rb	^{104}Nb	^{134m}Sb	^{143}Cs
^{84}As	^{93}Rb	^{105}Nb	^{135}Sb	^{144}Cs
^{85}As	^{94}Rb	^{106}Nb	^{136}Sb	^{145}Cs
^{86}As	^{95}Rb	^{109}Mo	^{137}Sb	^{146}Cs
^{87}As	^{96}Rb	^{110}Mo	^{136}Te	^{147}Cs
^{87}Se	^{97}Rb	^{109}Tc	^{137}Te	^{147}Ba
^{88}Se	^{98}Rb	^{110}Tc	^{138}Te	^{148}Ba
^{89}Se	^{99}Rb	^{122}Ag	^{139}Te	^{149}Ba
^{90}Se	^{97}Sr	^{123}Ag	^{137}I	^{150}Ba
^{91}Se	^{98}Sr	^{128}Cd	^{138}I	^{147}La

continues on next page

Table 10.2.8 – continued from previous page

^{87}Br	^{99}Sr	^{127}In	^{139}I	^{149}La
^{88}Br	^{100}Sr	^{127m}In	^{140}I	^{150}La

Neutron yields from alpha-particle interaction are available for 19 (α, n) target nuclides: ^{7}Li, ^{9}Be, ^{10}B, ^{11}B, ^{13}C, ^{14}N, ^{17}O, ^{18}O, ^{19}F, ^{21}Ne, ^{22}Ne, ^{23}Na, ^{25}Mg, ^{26}Mg, ^{27}Al, ^{29}Si, ^{30}Si, ^{31}P, and ^{37}Cl. The neutron decay data contain discrete alpha-particle energies and branching fractions for 89 actinides and 7 fission products listed in Table 10.2.9. The sources of the level branching fraction data and the (α, n) cross section data are listed in Table 10.2.10. The stopping cross sections and (α, n) target cross section and product level branching libraries are used in calculating the neutron yield and spectra from Ziegler [Origen-Data-ResourcesZie77] for all elements with Z < 93, and from Wilson [Origen-Data-ResourcesWPS+83] for all elements geq 93.

Table 10.2.9: Nuclides with α-particle emission data for neutron yield calculations

^{142}Ce	^{216}Po	^{226}Ac	^{237}Np	^{245}Cm
^{144}Nd	^{218}Po	^{227}Ac	^{235}Pu	^{246}Cm
^{146}Sm	^{215}At	^{226}Th	^{236}Pu	^{247}Cm
^{147}Sm	^{217}At	^{227}Th	^{237}Pu	^{248}Cm
^{148}Sm	^{218}At	^{228}Th	^{238}Pu	^{249}Bk
^{149}Sm	^{219}At	^{229}Th	^{239}Pu	^{248}Cf
^{152}Gd	^{217}Rn	^{230}Th	^{240}Pu	^{249}Cf
^{210}Pb	^{218}Rn	^{232}Th	^{241}Pu	^{250}Cf
^{210}Bi	^{219}Rn	^{230}Pa	^{242}Pu	^{251}Cf
^{211}Bi	^{220}Rn	^{231}Pa	^{244}Pu	^{252}Cf
^{212}Bi	^{222}Rn	^{230}U	^{240}Am	^{253}Cf
^{213}Bi	^{221}Fr	^{231}U	^{241}Am	^{254}Cf
^{214}Bi	^{222}Fr	^{232}U	^{242m}Am	^{253}Es
^{210}Po	^{223}Fr	^{233}U	^{243}Am	^{254}Es
^{211}Po	^{222}Ra	^{234}U	^{240}Cm	^{254m}Es
^{212}Po	^{223}Ra	^{235}U	^{241}Cm	^{255}Es
^{213}Po	^{224}Ra	^{236}U	^{242}Cm	^{254}Fm
^{214}Po	^{226}Ra	^{238}U	^{243}Cm	^{255}Fm
^{215}Po	^{225}Ac	^{235}Np	^{244}Cm	^{256}Fm
				^{257}Fm

Table 10.2.10: Target (α, n) cross section and branching level isotopes and sources

Isotope	**ZAID**	**Level branching fraction source data**	**Cross section data**
^{7}Li	30070	GNASH	Gibbons and Macklin [Origen-Data-ResourcesGM59]
^{9}Be	40090	Geiger and Van der Zwain [Origen-Data-ResourcesGZ75]	Geiger and Van der Zwain [Origen-Data-ResourcesGZ75]

continues on next page

Table 10.2.10 – continued from previous page

Isotope	ZAID	Level branching fraction source data	Cross section data
^{10}B	50010	GNASH	Bair *et al.* [Origen-Data-ResourcesBC79]
^{11}B	50110	GNASH	Bair *et al.* [Origen-Data-ResourcesBC79]
^{13}C	60130	GNASH*a*	Bair and Haas [Origen-Data-ResourcesBH73]
^{14}N	70140	N/A	GNASH
^{17}O	80170	Lessor and Schenter [Origen-Data-ResourcesLS71]	Perry and Wilson [Origen-Data-ResourcesPW81]
^{18}O	80180	Lesser and Schenter [Origen-Data-ResourcesLS71]	Perry and Wilson [Origen-Data-ResourcesPW81]
^{19}F	90190	Lesser and Schenter [Origen-Data-ResourcesLS71]	Balakrishnan *et al.* [Origen-Data-ResourcesMBM78]
^{21}Ne	100210	N/A	GNASH
^{22}Ne	100220	N/A	GNASH
^{23}Na	110230	GNASH	GNASH*a*
^{25}Mg	120250	GNASH	GNASH
^{26}Mg	120260	GNASH	GNASH
^{27}Al	130270	GNASH	GNASH*a*
^{29}Si	140290	GNASH	GNASH*a*
^{30}Si	140300	GNASH	GNASH*a*
^{31}P	150310	GNASH	GNASH
^{37}Cl	170370	GNASH	Woosley et. al. [Origen-Data-ResourcesWFHZ75]

10.2.8.3 Beta Emission

Beta emission rates and energy spectra are calculated using an analytic expression for the kinetic energy of the emitted β^- particles [Origen-Data-ResourcesGM71]:

$$N(Z,W) = \frac{g^2}{2\pi^3} F(Z,W)\,\rho W\,(W_0 - W)^2\,S_n(W)\,dW \tag{10.2.5}$$

where

Z = atomic number of the daugher nucleus

g = weak interaction coupling constant

W = kinetic energy of beta particle (in $m_e c^2$ units)

$F(Z,W)$ = Fermi function

W_0 = endpoint beta energy

$\rho = \sqrt{W^2 - 1}$ = electron momentum

$S_n(W)$ = spectral shape factor based on transition type

n = classification of the transition type

Internal conversion electron emission is not considered.

The calculation requires nuclear data on the fraction of the beta transition to each exicited state of the daughter nucleus, the maximum endpoint energy of the transition (W0), and a classification of the beta transition (n) defined by the spin and parity change of the transition which defines the spectral shape factor. The transition classification uses n=0 for allowed and forbidden non-unique transitions, n=1 for first forbidden unique transitions, n=2 for second forbidden unique transitions, and n=3 for third forbidden unique transitions. These data are not stored in the decay data resource but are included in a separate beta decay resource used only for the beta calculation.

The beta decay data are stored in the formatted file origen.rev00.ensdf95beta.data. The data are derived from ENSDF as compiled in 1995. The file includes beta decay information for 715 beta decay nuclides and has 8486 beta transition branches.

10.2.8.4 Alpha Emission

Calculation of the alpha emission intensity and spectrum requires detailed information that is not available on the decay resource. The calculation requires the alpha particle energy and branching fraction for each transition branch. Unlike the beta spectrum, the alpha particles are emitted with discrete energies, and the source spectrum may be generated by straightforward binning into the user-defined group structure. Alpha particle emission data are also used in the (α, n) neutron source calculation. Therefore, the alpha emission spectra are calculated using the same alpha decay library in the neutron emission resource: origen.rev01.alphdec.data.

10.2.9 DECAY RESOURCE FORMAT

The decay resource is a simple text format file that can be processed by COUPLE to create a binary decay-only library that can be used directly by ORIGEN. In general, this is not necessary, as the decay resource distributed with SCALE has already been processed with COUPLE to produce the end7dec ORIGEN decay-only binary library file. Modifying the decay data or the set of nuclides ORIGEN tracks requires modification of the decay resource file. The format is described in Table 10.2.11. Note that as of the SCALE 6.2 release, ORIGEN now uses the SCALE Standard Composition resource for abundance data and the "ABUND" field shown below is ignored by COUPLE when reading the decay resource.

Table 10.2.11: Definitions of data in the decay resource

Data name	Definition
LIB	Nuclide sublib (used by COUPLE)
NUC1	Nuclide identifier
IU	Units for the half life value (see numref:*tab-origen-hl-units*)
HALFL	Value of the half life in IU units
FB1	Beta decay transition leading to a daughter in the metastable state
FP	Positron emission decay fraction or orbital electron capture to the ground state
FP1	Positron emission decay fraction or orbital electron capture to a metastable state
FA	Alpha particle emission decay fraction
FT	Isomeric transition decay fraction
LIB1	Nuclide type in the library
FSF	Spontaneous fission decay fraction
FBN	Delayed neutron decay (beta particle and a neutron) fraction

continues on next page

Table 10.2.11 – continued from previous page

Q	Recoverable energy per decay (MeV)
ABUND	Natural atom isotopic abundance in percent (**no longer used**)
AMPC	Maximum permissible concentration in air
WMPC	Maximum permissible concentration in water
LIB1	Nuclide type in the library (used by COUPLE)
FG	Fraction of recoverable decay energy Q associated with gamma rays
FB	Beta decay transition leading to a daughter in the ground state
FBB	Double beta decay fraction
FN	Neutron decay fraction
FBA	Beta decay plus an alpha particle emission decay fraction

The variable LIB (and LIB1)defines the nuclide sublibrary (1/2/3=activation product/actinide/fission product). Variable LIB1 is included for formatting purposes only.

The nuclide identifier is read in variable NUC1 and is subsequently stored in array NUCL. The nuclide identifier is given by

$$\text{NUCL} = \text{Z} * 10000 + \text{A} * 10 + \text{I} \tag{10.2.6}$$

where Z is the atomic number, A is the atomic mass number, and I is the isomeric state, where $I = 0$ designates a ground state, and $I = 1$ is the first metastable state.

The variable HALFL is the physical half-life in units designated by the variable IU, as shown in Table 10.2.12. The definitions of 11 variables representing the different decay mode branching fractions are given in Table 10.2.11. The decay branching fractions are used in constructing the transition matrix.

Table 10.2.12: Units of half-life indicated by the variable IU

IU	**Units of half-life**
1	seconds
2	minutes
3	hours
4	days
5	years
6	stable
7	10^3 years
8	10^6 years
9	10^9 years

The variable Q is the total amount of recoverable energy (MeV) per disintegration released by radioactive decay used for decay heat calculations. It does not include the energy of neutrinos emitted during beta decay transitions. The variable FG is the fraction of recoverable energy per disintegration that comes from gamma rays and x-rays. The value of Q is obtained directly from ENDF/B-VII.1 as the sum of the average beta, gamma, and alpha decay energy values. The quantity includes the energy from all electron- related radiations such as β^-, β^+, Auger electrons, etc., all gamma rays, x-rays, and annihilation radiations, and the average energy of all heavy charged particles and delayed neutrons. The contribution from alpha decay energy includes the energy of the recoil nucleus. A part of the recoverable energy per decay not included in the ENDF/B-VII.1 values is the additional contribution from spontaneous fission. This energy was calculated

as the product of the spontaneous fission branching fraction and recoverable energy per fission using a value of 200 MeV per fission and added to the ENDF/B-VII.1 recoverable Q energy. A value of 12.56 MeV gamma energy per fission was used in computing the fraction of recoverable spontaneous fission energy from gamma rays.

External bremsstrahlung radiation is **not** included in the values of FG since the bremsstrahlung spectrum depends on electron interactions with the medium that contains the decay nuclide. The energy from capture gamma rays accompanying (α, n) reactions is also not included since it also depends on the medium. The variable ABUND is the atom percent abundance of naturally occurring isotopes.

An example of the decay resource content for selected fission products is presented as Example 10.2.1.

Example 10.2.1: Example of the ENDF/B-VII.1 decay data resource entries for selected fission products.

```
 3 DECAY LIBRARY: fission products (ENDF/B-VII.1)
 3 10030  5 1.2320E+01 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 5.6900E-03 0.0000E+00 6.4100E-09 4.4000E-04
 3          0.0000E+00 1.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 20030  6 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 0.0000E+00 1.3700E-04 1.0000E+00 1.0000E+00
 3          0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 20040  6 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 0.0000E+00 1.0000E+02 1.0000E+00 1.0000E+00
 3          0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 260650 1 8.1000E-01 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 4.7424E+00 0.0000E+00 1.0000E+00 1.0000E+00
 3          3.5861E-01 1.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 270650 1 1.1600E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 2.7723E+00 0.0000E+00 1.0000E+00 1.0000E+00
 3          3.3206E-02 1.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 280650 3 2.5172E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 1.1863E+00 0.0000E+00 4.6300E-09 1.0300E-04
 3          4.7061E-01 1.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 290650 6 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 0.0000E+00 3.0830E+01 1.0000E+00 1.0000E+00
 3          6.0103E-01 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 240660 1 1.0000E-02 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 8.2733E+00 0.0000E+00 1.0000E+00 1.0000E+00
 3          5.0000E-01 1.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 250660 1 6.4000E-02 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 1.0880E-01 8.4471E+00 0.0000E+00 1.0000E+00 1.0000E+00
 3          4.8945E-01 8.9120E-01 0.0000E+00 0.0000E+00 0.0000E+00
 3 260660 1 4.4000E-01 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 4.2271E+00 0.0000E+00 1.0000E+00 1.0000E+00
 3          5.0000E-01 1.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 270660 1 2.0000E-01 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 5.8904E+00 0.0000E+00 1.0000E+00 1.0000E+00
 3          4.1561E-01 1.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 280660 4 2.2750E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 7.3330E-02 0.0000E+00 9.2600E-10 6.1700E-06
 3          0.0000E+00 1.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 290660 2 5.1200E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 1.1645E+00 0.0000E+00 1.0000E+00 1.0000E+00
 3          8.3989E-02 1.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3 300660 6 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 3          0.0000E+00 0.0000E+00 0.0000E+00 2.7900E+01 1.0000E+00 1.0000E+00
 3          9.8809E-01 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
```

10.2.10 FISSION YIELD RESOURCE FORMAT

The independent fission product yields are stored as a formatted text file. The header record for each set of fission product yields includes the fissionable nuclide ID and an unused entry (0.0), followed by the number of incident neutron energies included for this nuclide. The fission yields for each energy are preceeded by a single record containing the incident neutron energy (eV), an unused entry (0.0), an index for the incident energy, the number of data entries per fission product, the total number of entries for each incident energy, and the number of fission products. The fission product yields for each fissionable nuclide and incident neutron energy are then listed as pairs of entries for the fission product nuclide ID and the independent (direct) fission yield as atom percent per fission. An example of the format is shown below in Example 10.2.2 for ^{227}Th.

The number and order of the fission product yields must be the same for all fissionable nuclides and must correspond to the fission products in the nuclear decay data. The fission product yields for each fissionable nuclide, excluding the yields for the terniary fission products ^{3}H, ^{3}He, and ^{4}He, sum to 200.

The fissionable nuclides and the tabulated incident neutron energies for which yields are available are listed in Table 10.2.2.

Example 10.2.2: Fission yield format example showing a portion of ^{227}Th.

```
ENDF/B-VII.0 modified library
9.02270+05 0.00000+00 1 0 0 0
2.53000-02 0.00000+00 1 2 2302 1151
1.00300+04 0.00000+00 2.00300+04 0.00000+00 2.00400+04 0.00000+00
2.60650+05 0.00000+00 2.70650+05 0.00000+00 2.80650+05 0.00000+00
2.90650+05 0.00000+00 2.40660+05 0.00000+00 2.50660+05 6.97001-13
2.60660+05 2.57000-10 2.70660+05 3.23000-10 2.80660+05 2.71000-10
2.90660+05 7.76001-13 3.00660+05 0.00000+00 3.10660+05 0.00000+00
3.20660+05 0.00000+00 2.40670+05 0.00000+00 2.50670+05 1.35000-13
2.60670+05 1.13000-10 2.70670+05 8.12001-10 2.80670+05 1.21000-09
2.90670+05 2.23000-11 3.00670+05 5.02000-14 3.10670+05 0.00000+00
3.20670+05 0.00000+00 2.50680+05 1.01000-14 2.60680+05 6.08001-11
2.70680+05 8.95001-10 2.80680+05 7.02001-09 2.90680+05 7.68001-11
2.90681+05 1.79000-10 3.00680+05 4.04000-12 3.10680+05 0.00000+00
3.20680+05 0.00000+00 2.50690+05 0.00000+00 2.60690+05 1.23000-11
2.70690+05 1.24000-09 2.80690+05 1.83000-08 2.90690+05 3.80000-09
3.00690+05 2.61000-11 3.00691+05 1.11000-10 3.10690+05 3.65000-14
3.20690+05 0.00000+00 3.30690+05 0.00000+00 2.60700+05 1.74000-12
2.70700+05 4.09000-10 2.80700+05 3.57000-08 2.90700+05 3.24000-09
2.90701+05 9.71001-09 3.00700+05 2.91000-09 3.10700+05 1.84000-12
3.20700+05 0.00000+00 2.60710+05 9.67001-14 2.70710+05 1.71000-10
2.80710+05 3.28000-08 2.90710+05 6.54001-08 3.00710+05 5.49000-09
3.00711+05 2.34000-08 3.10710+05 1.30000-10 3.20710+05 3.13000-14
3.20711+05 3.13000-14 3.30710+05 0.00000+00 2.60720+05 0.00000+00
2.70720+05 2.90000-11 2.80720+05 3.94000-08 2.90720+05 1.53000-07
3.00720+05 3.73000-07 3.10720+05 1.85500-09 3.10721+05 1.85500-09
3.20720+05 1.35000-11 3.30720+05 0.00000+00 3.40720+05 0.00000+00
2.70730+05 2.19000-12 2.80730+05 7.05001-09 2.90730+05 1.68000-07
```

10.2.11 GAMMA RESOURCE FORMAT

An example of the photon data entries for the emissions from ^{140}La decay is shown below in Example 10.2.3. The header record for each nuclide contains the nuclide ID, the total number of emission lines in the evaluation, as well as the number of discrete x-ray lines, discrete gamma lines, and number of pseudo lines used to represent continuum data if present in an evaluation used to reconstruct continuous energy emission spectra from the discrete representation. The last entries in the header record include the total gamma energy (MeV), and the character nuclide name. The emission spectrum is listed using pairs of entries for the photon energy (MeV) and photon emission (photons per disintegration).

Example 10.2.3: Gamma resource format example showing ^{140}La decay photon emission.

```
571400 52. 14. 38. 0. 0.2.3083E+00 la 140
4.3847E-03 2.1017E-04 4.8247E-03 1.7789E-03 5.3304E-03 1.5654E-03
6.0946E-03 2.3418E-04 3.4291E-02 5.9015E-03 3.4743E-02 1.0817E-02
3.9196E-02 1.0523E-03 3.9285E-02 2.0389E-03 3.9550E-02 1.2513E-05
3.9570E-02 1.6866E-05 4.0227E-02 2.2404E-04 4.0247E-02 4.3591E-04
4.0340E-02 2.6312E-06 4.0344E-02 3.5410E-06 2.4595E-02 1.4971E-05
6.4135E-02 1.4310E-04 6.8916E-02 7.5366E-04 1.0942E-01 2.1942E-03
1.3112E-01 4.6746E-03 1.7354E-01 1.2688E-03 2.4193E-01 4.1404E-03
2.6654E-01 4.6555E-03 3.0690E-01 2.4804E-04 3.2876E-01 2.0320E-01
3.9752E-01 7.3458E-04 4.3249E-01 2.9002E-02 4.3850E-01 3.9114E-04
4.4550E-01 2.8620E-05 4.8702E-01 4.5506E-01 6.1812E-01 3.7206E-04
7.5164E-01 4.3312E-02 8.1577E-01 2.3278E-01 8.6785E-01 5.5046E-02
9.1955E-01 2.6617E-02 9.2519E-01 6.8974E-02 9.5099E-01 5.1898E-03
9.9290E-01 1.3356E-04 1.0451E+00 2.4804E-04 1.0972E+00 2.2896E-04
1.3035E+00 4.1976E-04 1.4052E+00 5.9148E-04 1.5962E+00 9.5400E-01
1.8773E+00 4.1022E-04 1.9246E+00 1.3356E-04 2.0832E+00 1.1543E-04
2.3479E+00 8.4906E-03 2.4641E+00 1.1448E-04 2.5214E+00 3.4630E-02
2.5473E+00 1.0112E-03 2.8996E+00 6.6780E-04 3.1185E+00 2.4804E-04
3.3204E+00 3.8160E-05
```

10.2.12 ORIGEN "END7DEC" NUCLIDE SET

Table 10.2.13 shows a list of the 2,237 nuclides on the origen.rev04.end7dec ORIGEN binary decay-only library, and because this library provides the basis for all other libraries, effectively the set of nuclides tracked by ORIGEN in any decay or irradiation calculation. The "index" column is the index of that nuclide in the set (internally every ORIGEN isotopics vector has this order), the "sublib" column is the sublibrary (LT=light nuclide, AC=actinide, FP=fission product) in which the nuclide resides, the "nuclide" column is the nuclide identifier, the "mass" column is the mass of the nuclide in grams per mole, the "abundance" column is the natural abundance in atom percent for the nuclide (note only light nuclides have abundances), and the "decay" column is the decay constant. Note that the mass and abundance data are embedded on the library with the values from the current SCALE Standard Composition Library.

Table 10.2.13: Nuclide listing for "end7dec" ORIGEN library

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1	LT	1-H-1	1.0078	1.00E+02	0.00E+00
2	LT	1-H-2	2.0141	1.15E-02	0.00E+00
3	LT	1-H-3	3.0161	0.00E+00	1.78E-09
4	LT	2-He-3	3.0160	1.00E-04	0.00E+00
5	LT	2-He-4	4.0026	1.00E+02	0.00E+00
6	LT	2-He-5	5.0122	0.00E+00	6.93E+02
7	LT	2-He-6	6.0189	0.00E+00	8.59E-01
8	LT	3-Li-6	6.0151	7.59E+00	0.00E+00
9	LT	3-Li-7	7.0160	9.24E+01	0.00E+00
10	LT	3-Li-8	8.0225	0.00E+00	8.27E-01
11	LT	4-Be-7	7.0169	0.00E+00	1.51E-07
12	LT	4-Be-8	8.0053	0.00E+00	6.93E+02
13	LT	4-Be-9	9.0122	1.00E+02	0.00E+00
14	LT	4-Be-10	10.0135	0.00E+00	1.45E-14
15	LT	4-Be-11	11.0217	0.00E+00	5.02E-02

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
16	LT	5-B-10	10.0129	1.99E+01	0.00E+00
17	LT	5-B-11	11.0093	8.01E+01	0.00E+00
18	LT	5-B-12	12.0143	0.00E+00	3.43E+01
19	LT	6-C-12	12.0000	9.89E+01	0.00E+00
20	LT	6-C-13	13.0034	1.07E+00	0.00E+00
21	LT	6-C-14	14.0032	0.00E+00	3.85E-12
22	LT	6-C-15	15.0106	0.00E+00	2.83E-01
23	LT	7-N-13	13.0057	0.00E+00	1.16E-03
24	LT	7-N-14	14.0031	9.96E+01	0.00E+00
25	LT	7-N-15	15.0001	3.64E-01	0.00E+00
26	LT	7-N-16	16.0061	0.00E+00	9.72E-02
27	LT	8-O-16	15.9949	9.98E+01	0.00E+00
28	LT	8-O-17	16.9991	3.80E-02	0.00E+00
29	LT	8-O-18	17.9992	2.05E-01	0.00E+00
30	LT	8-O-19	19.0036	0.00E+00	2.58E-02
31	LT	9-F-19	18.9984	1.00E+02	0.00E+00
32	LT	9-F-20	20.0000	0.00E+00	6.21E-02
33	LT	10-Ne-20	19.9924	9.05E+01	0.00E+00
34	LT	10-Ne-21	20.9939	2.70E-01	0.00E+00
35	LT	10-Ne-22	21.9914	9.25E+00	0.00E+00
36	LT	10-Ne-23	22.9945	0.00E+00	1.86E-02
37	LT	11-Na-22	21.9944	0.00E+00	8.44E-09
38	LT	11-Na-23	22.9898	1.00E+02	0.00E+00
39	LT	11-Na-24	23.9910	0.00E+00	1.28E-05
40	LT	11-Na-24m	23.9910	0.00E+00	3.43E+01
41	LT	11-Na-25	24.9900	0.00E+00	1.17E-02
42	LT	12-Mg-24	23.9850	7.90E+01	0.00E+00
43	LT	12-Mg-25	24.9858	1.00E+01	0.00E+00
44	LT	12-Mg-26	25.9826	1.10E+01	0.00E+00
45	LT	12-Mg-27	26.9843	0.00E+00	1.22E-03
46	LT	12-Mg-28	27.9839	0.00E+00	9.21E-06
47	LT	13-Al-26	25.9869	0.00E+00	3.06E-14
48	LT	13-Al-27	26.9815	1.00E+02	0.00E+00
49	LT	13-Al-28	27.9819	0.00E+00	5.15E-03
50	LT	13-Al-29	28.9804	0.00E+00	1.76E-03
51	LT	13-Al-30	29.9830	0.00E+00	1.91E-01
52	LT	14-Si-28	27.9769	9.22E+01	0.00E+00
53	LT	14-Si-29	28.9765	4.69E+00	0.00E+00
54	LT	14-Si-30	29.9738	3.09E+00	0.00E+00
55	LT	14-Si-31	30.9754	0.00E+00	7.34E-05
56	LT	14-Si-32	31.9741	0.00E+00	1.44E-10
57	LT	15-P-31	30.9738	1.00E+02	0.00E+00
58	LT	15-P-32	31.9739	0.00E+00	5.62E-07

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
59	LT	15-P-33	32.9717	0.00E+00	3.17E-07
60	LT	15-P-34	33.9736	0.00E+00	5.58E-02
61	LT	16-S-32	31.9721	9.50E+01	0.00E+00
62	LT	16-S-33	32.9715	7.50E-01	0.00E+00
63	LT	16-S-34	33.9679	4.25E+00	0.00E+00
64	LT	16-S-35	34.9690	0.00E+00	9.17E-08
65	LT	16-S-36	35.9671	1.00E-02	0.00E+00
66	LT	16-S-37	36.9711	0.00E+00	2.29E-03
67	LT	17-Cl-35	34.9688	7.58E+01	0.00E+00
68	LT	17-Cl-36	35.9683	0.00E+00	7.30E-14
69	LT	17-Cl-37	36.9659	2.42E+01	0.00E+00
70	LT	17-Cl-38	37.9680	0.00E+00	3.10E-04
71	LT	17-Cl-38m	37.9680	0.00E+00	9.69E-01
72	LT	18-Ar-36	35.9675	3.37E-01	0.00E+00
73	LT	18-Ar-37	36.9668	0.00E+00	2.29E-07
74	LT	18-Ar-38	37.9627	6.32E-02	0.00E+00
75	LT	18-Ar-39	38.9643	0.00E+00	8.17E-11
76	LT	18-Ar-40	39.9624	9.96E+01	0.00E+00
77	LT	18-Ar-41	40.9645	0.00E+00	1.05E-04
78	LT	18-Ar-42	41.9631	0.00E+00	6.68E-10
79	LT	19-K-39	38.9637	9.33E+01	0.00E+00
80	LT	19-K-40	39.9640	1.17E-02	1.76E-17
81	LT	19-K-41	40.9618	6.73E+00	0.00E+00
82	LT	19-K-42	41.9624	0.00E+00	1.56E-05
83	LT	19-K-43	42.9607	0.00E+00	8.63E-06
84	LT	19-K-44	43.9616	0.00E+00	5.22E-04
85	LT	20-Ca-40	39.9626	9.69E+01	0.00E+00
86	LT	20-Ca-41	40.9623	0.00E+00	2.15E-13
87	LT	20-Ca-42	41.9586	6.47E-01	0.00E+00
88	LT	20-Ca-43	42.9588	1.35E-01	0.00E+00
89	LT	20-Ca-44	43.9555	2.09E+00	0.00E+00
90	LT	20-Ca-45	44.9562	0.00E+00	4.93E-08
91	LT	20-Ca-46	45.9537	4.00E-03	0.00E+00
92	LT	20-Ca-47	46.9546	0.00E+00	1.77E-06
93	LT	20-Ca-48	47.9525	1.87E-01	9.55E-28
94	LT	20-Ca-49	48.9557	0.00E+00	1.33E-03
95	LT	21-Sc-44	43.9594	0.00E+00	4.85E-05
96	LT	21-Sc-44m	43.9594	0.00E+00	3.29E-06
97	LT	21-Sc-45	44.9559	1.00E+02	0.00E+00
98	LT	21-Sc-45m	44.9559	0.00E+00	2.18E+00
99	LT	21-Sc-46	45.9552	0.00E+00	9.57E-08
100	LT	21-Sc-46m	45.9552	0.00E+00	3.70E-02
101	LT	21-Sc-47	46.9524	0.00E+00	2.40E-06

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
102	LT	21-Sc-48	47.9522	0.00E+00	4.41E-06
103	LT	21-Sc-49	48.9500	0.00E+00	2.02E-04
104	LT	21-Sc-50	49.9522	0.00E+00	6.76E-03
105	LT	22-Ti-44	43.9597	0.00E+00	3.66E-10
106	LT	22-Ti-45	44.9581	0.00E+00	6.25E-05
107	LT	22-Ti-46	45.9526	8.25E+00	0.00E+00
108	LT	22-Ti-47	46.9518	7.44E+00	0.00E+00
109	LT	22-Ti-48	47.9479	7.37E+01	0.00E+00
110	LT	22-Ti-49	48.9479	5.41E+00	0.00E+00
111	LT	22-Ti-50	49.9448	5.18E+00	0.00E+00
112	LT	22-Ti-51	50.9466	0.00E+00	2.01E-03
113	LT	23-V-48	47.9523	0.00E+00	5.02E-07
114	LT	23-V-49	48.9485	0.00E+00	2.43E-08
115	LT	23-V-50	49.9472	2.50E-01	1.57E-25
116	LT	23-V-51	50.9440	9.98E+01	0.00E+00
117	LT	23-V-52	51.9448	0.00E+00	3.09E-03
118	LT	23-V-53	52.9443	0.00E+00	7.49E-03
119	LT	23-V-54	53.9464	0.00E+00	1.39E-02
120	LT	24-Cr-48	47.9540	0.00E+00	8.93E-06
121	LT	24-Cr-49	48.9513	0.00E+00	2.73E-04
122	LT	24-Cr-50	49.9460	4.35E+00	0.00E+00
123	LT	24-Cr-51	50.9448	0.00E+00	2.90E-07
124	LT	24-Cr-52	51.9405	8.38E+01	0.00E+00
125	LT	24-Cr-53	52.9407	9.50E+00	0.00E+00
126	LT	24-Cr-54	53.9389	2.37E+00	0.00E+00
127	LT	24-Cr-55	54.9408	0.00E+00	3.30E-03
128	LT	25-Mn-52	51.9456	0.00E+00	1.43E-06
129	LT	25-Mn-53	52.9413	0.00E+00	5.94E-15
130	LT	25-Mn-54	53.9404	0.00E+00	2.57E-08
131	LT	25-Mn-55	54.9380	1.00E+02	0.00E+00
132	LT	25-Mn-56	55.9389	0.00E+00	7.47E-05
133	LT	25-Mn-57	56.9383	0.00E+00	8.12E-03
134	LT	25-Mn-58	57.9400	0.00E+00	2.31E-01
135	LT	26-Fe-54	53.9396	5.85E+00	0.00E+00
136	LT	26-Fe-55	54.9383	0.00E+00	8.00E-09
137	LT	26-Fe-56	55.9349	9.18E+01	0.00E+00
138	LT	26-Fe-57	56.9354	2.12E+00	0.00E+00
139	LT	26-Fe-58	57.9333	2.82E-01	0.00E+00
140	LT	26-Fe-59	58.9349	0.00E+00	1.80E-07
141	LT	26-Fe-60	59.9341	0.00E+00	1.46E-14
142	LT	27-Co-55	54.9420	0.00E+00	1.10E-05
143	LT	27-Co-56	55.9398	0.00E+00	1.04E-07
144	LT	27-Co-57	56.9363	0.00E+00	2.95E-08

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
145	LT	27-Co-58m	57.9358	0.00E+00	2.12E-05
146	LT	27-Co-58	57.9357	0.00E+00	1.13E-07
147	LT	27-Co-59	58.9332	1.00E+02	0.00E+00
148	LT	27-Co-60	59.9338	0.00E+00	4.17E-09
149	LT	27-Co-60m	59.9338	0.00E+00	1.10E-03
150	LT	27-Co-61	60.9325	0.00E+00	1.17E-04
151	LT	27-Co-62	61.9341	0.00E+00	7.70E-03
152	LT	28-Ni-56	55.9421	0.00E+00	1.32E-06
153	LT	28-Ni-57	56.9398	0.00E+00	5.41E-06
154	LT	28-Ni-58	57.9353	6.81E+01	0.00E+00
155	LT	28-Ni-59	58.9343	0.00E+00	2.89E-13
156	LT	28-Ni-60	59.9308	2.62E+01	0.00E+00
157	LT	28-Ni-61	60.9311	1.14E+00	0.00E+00
158	LT	28-Ni-62	61.9283	3.63E+00	0.00E+00
159	LT	28-Ni-63	62.9297	0.00E+00	2.17E-10
160	LT	28-Ni-64	63.9280	9.26E-01	0.00E+00
161	LT	28-Ni-65	64.9301	0.00E+00	7.65E-05
162	LT	28-Ni-66	65.9291	0.00E+00	3.53E-06
163	LT	29-Cu-62	61.9326	0.00E+00	1.19E-03
164	LT	29-Cu-63	62.9296	6.92E+01	0.00E+00
165	LT	29-Cu-64	63.9298	0.00E+00	1.52E-05
166	LT	29-Cu-65	64.9278	3.09E+01	0.00E+00
167	LT	29-Cu-66	65.9289	0.00E+00	2.26E-03
168	LT	29-Cu-67	66.9277	0.00E+00	3.11E-06
169	LT	30-Zn-63	62.9332	0.00E+00	3.00E-04
170	LT	30-Zn-64	63.9291	4.83E+01	0.00E+00
171	LT	30-Zn-65	64.9292	0.00E+00	3.29E-08
172	LT	30-Zn-66	65.9260	2.80E+01	0.00E+00
173	LT	30-Zn-67	66.9271	4.10E+00	0.00E+00
174	LT	30-Zn-68	67.9248	1.90E+01	0.00E+00
175	LT	30-Zn-69	68.9266	0.00E+00	2.05E-04
176	LT	30-Zn-69m	68.9266	0.00E+00	1.40E-05
177	LT	30-Zn-70	69.9253	6.31E-01	0.00E+00
178	LT	30-Zn-71	70.9277	0.00E+00	4.72E-03
179	LT	30-Zn-71m	70.9277	0.00E+00	4.86E-05
180	LT	30-Zn-72	71.9269	0.00E+00	4.14E-06
181	LT	31-Ga-67	66.9282	0.00E+00	2.46E-06
182	LT	31-Ga-68	67.9280	0.00E+00	1.71E-04
183	LT	31-Ga-69	68.9256	6.01E+01	0.00E+00
184	LT	31-Ga-70	69.9260	0.00E+00	5.46E-04
185	LT	31-Ga-71	70.9247	3.99E+01	0.00E+00
186	LT	31-Ga-72	71.9264	0.00E+00	1.37E-05
187	LT	31-Ga-72m	71.9264	0.00E+00	1.75E+01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
188	LT	32-Ge-68	67.9281	0.00E+00	2.96E-08
189	LT	32-Ge-69	68.9280	0.00E+00	4.93E-06
190	LT	32-Ge-70	69.9242	2.04E+01	0.00E+00
191	LT	32-Ge-71	70.9249	0.00E+00	7.02E-07
192	LT	32-Ge-71m	70.9249	0.00E+00	3.40E+01
193	LT	32-Ge-72	71.9221	2.73E+01	0.00E+00
194	LT	32-Ge-73	72.9235	7.76E+00	0.00E+00
195	LT	32-Ge-73m	72.9235	0.00E+00	1.39E+00
196	LT	32-Ge-74	73.9212	3.67E+01	0.00E+00
197	LT	32-Ge-75	74.9229	0.00E+00	1.40E-04
198	LT	32-Ge-75m	74.9229	0.00E+00	1.45E-02
199	LT	32-Ge-76	75.9214	7.83E+00	0.00E+00
200	LT	32-Ge-77	76.9236	0.00E+00	1.70E-05
201	LT	32-Ge-77m	76.9236	0.00E+00	1.31E-02
202	LT	33-As-71	70.9271	0.00E+00	2.95E-06
203	LT	33-As-72	71.9268	0.00E+00	7.41E-06
204	LT	33-As-73	72.9238	0.00E+00	9.99E-08
205	LT	33-As-74	73.9239	0.00E+00	4.51E-07
206	LT	33-As-75	74.9216	1.00E+02	0.00E+00
207	LT	33-As-75m	74.9216	0.00E+00	3.93E+01
208	LT	33-As-76	75.9224	0.00E+00	7.34E-06
209	LT	33-As-77	76.9206	0.00E+00	4.96E-06
210	LT	34-Se-72	71.9271	0.00E+00	9.55E-07
211	LT	34-Se-73	72.9268	0.00E+00	2.69E-05
212	LT	34-Se-74	73.9225	8.90E-01	0.00E+00
213	LT	34-Se-75	74.9225	0.00E+00	6.70E-08
214	LT	34-Se-76	75.9192	9.37E+00	0.00E+00
215	LT	34-Se-77	76.9199	7.63E+00	0.00E+00
216	LT	34-Se-77m	76.9199	0.00E+00	3.99E-02
217	LT	34-Se-78	77.9173	2.38E+01	0.00E+00
218	LT	34-Se-79	78.9185	0.00E+00	7.45E-14
219	LT	34-Se-79m	78.9185	0.00E+00	2.95E-03
220	LT	34-Se-80	79.9165	4.96E+01	0.00E+00
221	LT	34-Se-81	80.9180	0.00E+00	6.26E-04
222	LT	34-Se-81m	80.9180	0.00E+00	2.02E-04
223	LT	34-Se-82	81.9167	8.73E+00	0.00E+00
224	LT	34-Se-83	82.9191	0.00E+00	5.18E-04
225	LT	34-Se-83m	82.9191	0.00E+00	9.89E-03
226	LT	35-Br-76	75.9245	0.00E+00	1.19E-05
227	LT	35-Br-77	76.9214	0.00E+00	3.38E-06
228	LT	35-Br-77m	76.9214	0.00E+00	2.70E-03
229	LT	35-Br-78	77.9212	0.00E+00	1.79E-03
230	LT	35-Br-79	78.9183	5.07E+01	0.00E+00

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
231	LT	35-Br-80	79.9185	0.00E+00	6.53E-04
232	LT	35-Br-80m	79.9185	0.00E+00	4.36E-05
233	LT	35-Br-81	80.9163	4.93E+01	0.00E+00
234	LT	35-Br-82	81.9168	0.00E+00	5.46E-06
235	LT	35-Br-82m	81.9168	0.00E+00	1.88E-03
236	LT	35-Br-83	82.9152	0.00E+00	8.02E-05
237	LT	36-Kr-76	75.9259	0.00E+00	1.30E-05
238	LT	36-Kr-77	76.9247	0.00E+00	1.55E-04
239	LT	36-Kr-78	77.9204	3.55E-01	0.00E+00
240	LT	36-Kr-79	78.9201	0.00E+00	5.49E-06
241	LT	36-Kr-79m	78.9201	0.00E+00	1.39E-02
242	LT	36-Kr-80	79.9164	2.29E+00	0.00E+00
243	LT	36-Kr-81	80.9166	0.00E+00	9.59E-14
244	LT	36-Kr-81m	80.9166	0.00E+00	5.29E-02
245	LT	36-Kr-82	81.9135	1.16E+01	0.00E+00
246	LT	36-Kr-83	82.9141	1.15E+01	0.00E+00
247	LT	36-Kr-83m	82.9141	0.00E+00	1.05E-04
248	LT	36-Kr-84	83.9115	5.70E+01	0.00E+00
249	LT	36-Kr-85	84.9125	0.00E+00	2.04E-09
250	LT	36-Kr-85m	84.9125	0.00E+00	4.30E-05
251	LT	36-Kr-86	85.9106	1.73E+01	0.00E+00
252	LT	36-Kr-87	86.9134	0.00E+00	1.51E-04
253	LT	36-Kr-88	87.9145	0.00E+00	6.78E-05
254	LT	37-Rb-82	81.9182	0.00E+00	9.19E-03
255	LT	37-Rb-83	82.9151	0.00E+00	9.31E-08
256	LT	37-Rb-84	83.9144	0.00E+00	2.44E-07
257	LT	37-Rb-85	84.9118	7.22E+01	0.00E+00
258	LT	37-Rb-86	85.9112	0.00E+00	4.31E-07
259	LT	37-Rb-86m	85.9112	0.00E+00	1.14E-02
260	LT	37-Rb-87	86.9092	2.78E+01	4.57E-19
261	LT	37-Rb-88	87.9113	0.00E+00	6.50E-04
262	LT	37-Rb-89	88.9123	0.00E+00	7.63E-04
263	LT	38-Sr-82	81.9184	0.00E+00	3.16E-07
264	LT	38-Sr-83	82.9176	0.00E+00	5.94E-06
265	LT	38-Sr-84	83.9134	5.60E-01	0.00E+00
266	LT	38-Sr-85	84.9129	0.00E+00	1.24E-07
267	LT	38-Sr-85m	84.9129	0.00E+00	1.71E-04
268	LT	38-Sr-86	85.9093	9.86E+00	0.00E+00
269	LT	38-Sr-87	86.9089	7.00E+00	0.00E+00
270	LT	38-Sr-87m	86.9089	0.00E+00	6.84E-05
271	LT	38-Sr-88	87.9056	8.26E+01	0.00E+00
272	LT	38-Sr-89	88.9074	0.00E+00	1.59E-07
273	LT	38-Sr-90	89.9077	0.00E+00	7.63E-10

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
274	LT	38-Sr-91	90.9102	0.00E+00	2.00E-05
275	LT	38-Sr-93	92.9140	0.00E+00	1.56E-03
276	LT	39-Y-86	85.9149	0.00E+00	1.31E-05
277	LT	39-Y-87	86.9109	0.00E+00	2.41E-06
278	LT	39-Y-87m	86.9109	0.00E+00	1.44E-05
279	LT	39-Y-88	87.9095	0.00E+00	7.52E-08
280	LT	39-Y-89	88.9059	1.00E+02	0.00E+00
281	LT	39-Y-89m	88.9059	0.00E+00	4.43E-02
282	LT	39-Y-90	89.9072	0.00E+00	3.01E-06
283	LT	39-Y-90m	89.9072	0.00E+00	6.04E-05
284	LT	39-Y-91	90.9073	0.00E+00	1.37E-07
285	LT	39-Y-91m	90.9073	0.00E+00	2.32E-04
286	LT	39-Y-92	91.9090	0.00E+00	5.44E-05
287	LT	39-Y-93	92.9096	0.00E+00	1.89E-05
288	LT	39-Y-93m	92.9096	0.00E+00	8.45E-01
289	LT	39-Y-94	93.9116	0.00E+00	6.18E-04
290	LT	39-Y-96	95.9159	0.00E+00	1.30E-01
291	LT	40-Zr-86	85.9165	0.00E+00	1.17E-05
292	LT	40-Zr-87	86.9148	0.00E+00	1.15E-04
293	LT	40-Zr-88	87.9102	0.00E+00	9.62E-08
294	LT	40-Zr-89	88.9089	0.00E+00	2.46E-06
295	LT	40-Zr-90	89.9047	5.15E+01	0.00E+00
296	LT	40-Zr-90m	89.9047	0.00E+00	8.57E-01
297	LT	40-Zr-91	90.9056	1.12E+01	0.00E+00
298	LT	40-Zr-92	91.9050	1.72E+01	0.00E+00
299	LT	40-Zr-93	92.9065	0.00E+00	1.44E-14
300	LT	40-Zr-94	93.9063	1.74E+01	0.00E+00
301	LT	40-Zr-95	94.9080	0.00E+00	1.25E-07
302	LT	40-Zr-96	95.9083	2.80E+00	1.10E-27
303	LT	40-Zr-97	96.9109	0.00E+00	1.15E-05
304	LT	41-Nb-90	89.9113	0.00E+00	1.32E-05
305	LT	41-Nb-90m	89.9113	0.00E+00	3.69E-02
306	LT	41-Nb-91	90.9070	0.00E+00	3.23E-11
307	LT	41-Nb-91m	90.9070	0.00E+00	1.32E-07
308	LT	41-Nb-92	91.9072	0.00E+00	6.33E-16
309	LT	41-Nb-92m	91.9072	0.00E+00	7.90E-07
310	LT	41-Nb-93	92.9064	1.00E+02	0.00E+00
311	LT	41-Nb-93m	92.9064	0.00E+00	1.36E-09
312	LT	41-Nb-94	93.9073	0.00E+00	1.08E-12
313	LT	41-Nb-94m	93.9073	0.00E+00	1.84E-03
314	LT	41-Nb-95	94.9068	0.00E+00	2.29E-07
315	LT	41-Nb-95m	94.9068	0.00E+00	2.22E-06
316	LT	41-Nb-96	95.9081	0.00E+00	8.25E-06

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
317	LT	41-Nb-97	96.9081	0.00E+00	1.60E-04
318	LT	41-Nb-97m	96.9081	0.00E+00	1.18E-02
319	LT	41-Nb-98	97.9103	0.00E+00	2.42E-01
320	LT	41-Nb-100	99.9142	0.00E+00	4.62E-01
321	LT	42-Mo-92	91.9068	1.48E+01	0.00E+00
322	LT	42-Mo-93m	92.9068	0.00E+00	2.81E-05
323	LT	42-Mo-93	92.9068	0.00E+00	5.49E-12
324	LT	42-Mo-94	93.9051	9.23E+00	0.00E+00
325	LT	42-Mo-95	94.9058	1.59E+01	0.00E+00
326	LT	42-Mo-96	95.9047	1.67E+01	0.00E+00
327	LT	42-Mo-97	96.9060	9.56E+00	0.00E+00
328	LT	42-Mo-98	97.9054	2.42E+01	0.00E+00
329	LT	42-Mo-99	98.9077	0.00E+00	2.92E-06
330	LT	42-Mo-100	99.9075	9.67E+00	3.01E-27
331	LT	42-Mo-101	100.9103	0.00E+00	7.91E-04
332	LT	43-Tc-95	94.9077	0.00E+00	9.63E-06
333	LT	43-Tc-95m	94.9077	0.00E+00	1.32E-07
334	LT	43-Tc-96	95.9079	0.00E+00	1.87E-06
335	LT	43-Tc-97	96.9064	0.00E+00	5.22E-15
336	LT	43-Tc-97m	96.9064	0.00E+00	8.82E-08
337	LT	43-Tc-98	97.9072	0.00E+00	5.23E-15
338	LT	43-Tc-99	98.9062	0.00E+00	1.04E-13
339	LT	43-Tc-99m	98.9062	0.00E+00	3.21E-05
340	LT	43-Tc-100	99.9077	0.00E+00	4.48E-02
341	LT	43-Tc-101	100.9073	0.00E+00	8.14E-04
342	LT	44-Ru-96	95.9076	5.54E+00	0.00E+00
343	LT	44-Ru-97	96.9076	0.00E+00	2.83E-06
344	LT	44-Ru-98	97.9053	1.87E+00	0.00E+00
345	LT	44-Ru-99	98.9059	1.28E+01	0.00E+00
346	LT	44-Ru-100	99.9042	1.26E+01	0.00E+00
347	LT	44-Ru-101	100.9056	1.71E+01	0.00E+00
348	LT	44-Ru-102	101.9044	3.16E+01	0.00E+00
349	LT	44-Ru-103	102.9063	0.00E+00	2.04E-07
350	LT	44-Ru-104	103.9054	1.86E+01	0.00E+00
351	LT	44-Ru-105	104.9078	0.00E+00	4.34E-05
352	LT	44-Ru-106	105.9073	0.00E+00	2.16E-08
353	LT	44-Ru-107	106.9099	0.00E+00	3.08E-03
354	LT	45-Rh-99	98.9081	0.00E+00	4.98E-07
355	LT	45-Rh-99m	98.9081	0.00E+00	4.10E-05
356	LT	45-Rh-100	99.9081	0.00E+00	9.26E-06
357	LT	45-Rh-101	100.9062	0.00E+00	6.66E-09
358	LT	45-Rh-101m	100.9062	0.00E+00	1.85E-06
359	LT	45-Rh-102	101.9068	0.00E+00	3.87E-08

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
360	LT	45-Rh-102m	101.9068	0.00E+00	5.87E-09
361	LT	45-Rh-103	102.9055	1.00E+02	0.00E+00
362	LT	45-Rh-103m	102.9055	0.00E+00	2.06E-04
363	LT	45-Rh-104	103.9067	0.00E+00	1.64E-02
364	LT	45-Rh-104m	103.9067	0.00E+00	2.66E-03
365	LT	45-Rh-105	104.9057	0.00E+00	5.45E-06
366	LT	45-Rh-105m	104.9057	0.00E+00	1.73E-02
367	LT	45-Rh-106	105.9073	0.00E+00	2.31E-02
368	LT	45-Rh-106m	105.9073	0.00E+00	8.82E-05
369	LT	45-Rh-107	106.9068	0.00E+00	5.32E-04
370	LT	46-Pd-100	99.9085	0.00E+00	2.21E-06
371	LT	46-Pd-101	100.9083	0.00E+00	2.27E-05
372	LT	46-Pd-102	101.9056	1.02E+00	0.00E+00
373	LT	46-Pd-103	102.9061	0.00E+00	4.72E-07
374	LT	46-Pd-104	103.9040	1.11E+01	0.00E+00
375	LT	46-Pd-105	104.9051	2.23E+01	0.00E+00
376	LT	46-Pd-106	105.9035	2.73E+01	0.00E+00
377	LT	46-Pd-107	106.9051	0.00E+00	3.38E-15
378	LT	46-Pd-107m	106.9051	0.00E+00	3.25E-02
379	LT	46-Pd-108	107.9039	2.65E+01	0.00E+00
380	LT	46-Pd-109	108.9060	0.00E+00	1.41E-05
381	LT	46-Pd-109m	108.9060	0.00E+00	2.46E-03
382	LT	46-Pd-110	109.9052	1.17E+01	0.00E+00
383	LT	46-Pd-111	110.9077	0.00E+00	4.94E-04
384	LT	46-Pd-111m	110.9077	0.00E+00	3.50E-05
385	LT	46-Pd-112	111.9073	0.00E+00	9.16E-06
386	LT	47-Ag-105	104.9065	0.00E+00	1.94E-07
387	LT	47-Ag-106	105.9067	0.00E+00	4.82E-04
388	LT	47-Ag-106m	105.9067	0.00E+00	9.69E-07
389	LT	47-Ag-107	106.9051	5.18E+01	0.00E+00
390	LT	47-Ag-107m	106.9051	0.00E+00	1.56E-02
391	LT	47-Ag-108	107.9060	0.00E+00	4.85E-03
392	LT	47-Ag-108m	107.9060	0.00E+00	5.01E-11
393	LT	47-Ag-109	108.9047	4.82E+01	0.00E+00
394	LT	47-Ag-109m	108.9047	0.00E+00	1.75E-02
395	LT	47-Ag-110	109.9061	0.00E+00	2.82E-02
396	LT	47-Ag-110m	109.9062	0.00E+00	3.21E-08
397	LT	47-Ag-111	110.9053	0.00E+00	1.08E-06
398	LT	47-Ag-111m	110.9053	0.00E+00	1.07E-02
399	LT	47-Ag-112	111.9070	0.00E+00	6.15E-05
400	LT	48-Cd-106	105.9065	1.25E+00	0.00E+00
401	LT	48-Cd-107	106.9066	0.00E+00	2.96E-05
402	LT	48-Cd-108	107.9042	8.90E-01	0.00E+00

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
403	LT	48-Cd-109	108.9050	0.00E+00	1.74E-08
404	LT	48-Cd-110	109.9030	1.25E+01	0.00E+00
405	LT	48-Cd-111	110.9042	1.28E+01	0.00E+00
406	LT	48-Cd-111m	110.9042	0.00E+00	2.38E-04
407	LT	48-Cd-112	111.9028	2.41E+01	0.00E+00
408	LT	48-Cd-113	112.9044	1.22E+01	2.73E-24
409	LT	48-Cd-113m	112.9044	0.00E+00	1.56E-09
410	LT	48-Cd-114	113.9034	2.87E+01	0.00E+00
411	LT	48-Cd-115	114.9054	0.00E+00	3.60E-06
412	LT	48-Cd-115m	114.9051	0.00E+00	1.80E-07
413	LT	48-Cd-116	115.9048	7.49E+00	7.09E-28
414	LT	48-Cd-117	116.9072	0.00E+00	7.73E-05
415	LT	48-Cd-117m	116.9072	0.00E+00	5.73E-05
416	LT	48-Cd-119	118.9099	0.00E+00	4.29E-03
417	LT	48-Cd-121	120.9130	0.00E+00	5.13E-02
418	LT	49-In-111	110.9051	0.00E+00	2.86E-06
419	LT	49-In-112	111.9055	0.00E+00	7.72E-04
420	LT	49-In-113	112.9041	4.29E+00	0.00E+00
421	LT	49-In-113m	112.9041	0.00E+00	1.16E-04
422	LT	49-In-114	113.9049	0.00E+00	9.64E-03
423	LT	49-In-114m	113.9049	0.00E+00	1.62E-07
424	LT	49-In-115	114.9039	9.57E+01	4.98E-23
425	LT	49-In-115m	114.9039	0.00E+00	4.29E-05
426	LT	49-In-116	115.9053	0.00E+00	4.92E-02
427	LT	49-In-116m	115.9053	0.00E+00	2.13E-04
428	LT	49-In-117	116.9045	0.00E+00	2.67E-04
429	LT	49-In-117m	116.9045	0.00E+00	9.94E-05
430	LT	49-In-118	117.9063	0.00E+00	1.39E-01
431	LT	49-In-119	118.9059	0.00E+00	4.81E-03
432	LT	49-In-119m	118.9059	0.00E+00	6.42E-04
433	LT	49-In-120	119.9080	0.00E+00	2.25E-01
434	LT	49-In-120m	119.9080	0.00E+00	1.50E-02
435	LT	49-In-121	120.9079	0.00E+00	3.00E-02
436	LT	49-In-121m	120.9079	0.00E+00	2.98E-03
437	LT	50-Sn-112	111.9048	9.70E-01	0.00E+00
438	LT	50-Sn-113	112.9052	0.00E+00	6.97E-08
439	LT	50-Sn-113m	112.9052	0.00E+00	5.40E-04
440	LT	50-Sn-114	113.9028	6.60E-01	0.00E+00
441	LT	50-Sn-115	114.9033	3.40E-01	0.00E+00
442	LT	50-Sn-116	115.9017	1.45E+01	0.00E+00
443	LT	50-Sn-117	116.9029	7.68E+00	0.00E+00
444	LT	50-Sn-117m	116.9029	0.00E+00	5.90E-07
445	LT	50-Sn-118	117.9016	2.42E+01	0.00E+00

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
446	LT	50-Sn-119	118.9033	8.59E+00	0.00E+00
447	LT	50-Sn-119m	118.9033	0.00E+00	2.74E-08
448	LT	50-Sn-120	119.9022	3.26E+01	0.00E+00
449	LT	50-Sn-121	120.9042	0.00E+00	7.12E-06
450	LT	50-Sn-121m	120.9042	0.00E+00	5.00E-10
451	LT	50-Sn-122	121.9034	4.63E+00	0.00E+00
452	LT	50-Sn-123	122.9057	0.00E+00	6.21E-08
453	LT	50-Sn-123m	122.9057	0.00E+00	2.88E-04
454	LT	50-Sn-124	123.9053	5.79E+00	0.00E+00
455	LT	50-Sn-125	124.9078	0.00E+00	8.32E-07
456	LT	50-Sn-125m	124.9078	0.00E+00	1.21E-03
457	LT	50-Sn-126	125.9077	0.00E+00	9.55E-14
458	LT	51-Sb-118	117.9055	0.00E+00	3.21E-03
459	LT	51-Sb-119	118.9039	0.00E+00	5.04E-06
460	LT	51-Sb-120	119.9051	0.00E+00	7.27E-04
461	LT	51-Sb-120m	119.9051	0.00E+00	1.39E-06
462	LT	51-Sb-121	120.9038	5.72E+01	0.00E+00
463	LT	51-Sb-122	121.9052	0.00E+00	2.95E-06
464	LT	51-Sb-122m	121.9052	0.00E+00	2.76E-03
465	LT	51-Sb-123	122.9042	4.28E+01	0.00E+00
466	LT	51-Sb-124	123.9059	0.00E+00	1.33E-07
467	LT	51-Sb-124m	123.9059	0.00E+00	7.45E-03
468	LT	51-Sb-125	124.9053	0.00E+00	7.96E-09
469	LT	51-Sb-126	125.9072	0.00E+00	6.50E-07
470	LT	51-Sb-126m	125.9072	0.00E+00	6.03E-04
471	LT	51-Sb-127	126.9069	0.00E+00	2.08E-06
472	LT	52-Te-118	117.9058	0.00E+00	1.34E-06
473	LT	52-Te-119	118.9064	0.00E+00	1.20E-05
474	LT	52-Te-119m	118.9064	0.00E+00	1.71E-06
475	LT	52-Te-120	119.9040	9.00E-02	0.00E+00
476	LT	52-Te-121	120.9049	0.00E+00	4.19E-07
477	LT	52-Te-121m	120.9049	0.00E+00	4.89E-08
478	LT	52-Te-122	121.9030	2.55E+00	0.00E+00
479	LT	52-Te-123	122.9043	8.90E-01	0.00E+00
480	LT	52-Te-123m	122.9043	0.00E+00	6.73E-08
481	LT	52-Te-124	123.9028	4.74E+00	0.00E+00
482	LT	52-Te-125	124.9044	7.07E+00	0.00E+00
483	LT	52-Te-125m	124.9044	0.00E+00	1.40E-07
484	LT	52-Te-126	125.9033	1.88E+01	0.00E+00
485	LT	52-Te-127	126.9052	0.00E+00	2.06E-05
486	LT	52-Te-127m	126.9052	0.00E+00	7.36E-08
487	LT	52-Te-128	127.9045	3.17E+01	2.50E-27
488	LT	52-Te-129	128.9066	0.00E+00	1.66E-04

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
489	LT	52-Te-129m	128.9074	0.00E+00	2.39E-07
490	LT	52-Te-130	129.9062	3.41E+01	0.00E+00
491	LT	52-Te-131	130.9085	0.00E+00	4.62E-04
492	LT	52-Te-131m	130.9085	0.00E+00	5.79E-06
493	LT	52-Te-132	131.9086	0.00E+00	2.50E-06
494	LT	53-I-122	121.9076	0.00E+00	3.18E-03
495	LT	53-I-123	122.9056	0.00E+00	1.46E-05
496	LT	53-I-124	123.9062	0.00E+00	1.92E-06
497	LT	53-I-125	124.9046	0.00E+00	1.35E-07
498	LT	53-I-126	125.9056	0.00E+00	6.20E-07
499	LT	53-I-127	126.9045	1.00E+02	0.00E+00
500	LT	53-I-128	127.9058	0.00E+00	4.62E-04
501	LT	53-I-129	128.9050	0.00E+00	1.40E-15
502	LT	53-I-130	129.9067	0.00E+00	1.56E-05
503	LT	53-I-130m	129.9067	0.00E+00	1.31E-03
504	LT	53-I-131	130.9061	0.00E+00	1.00E-06
505	LT	53-I-132	131.9080	0.00E+00	8.39E-05
506	LT	53-I-133	132.9078	0.00E+00	9.26E-06
507	LT	54-Xe-122	121.9084	0.00E+00	9.58E-06
508	LT	54-Xe-123	122.9085	0.00E+00	9.26E-05
509	LT	54-Xe-124	123.9059	9.52E-02	0.00E+00
510	LT	54-Xe-125	124.9064	0.00E+00	1.14E-05
511	LT	54-Xe-125m	124.9064	0.00E+00	1.22E-02
512	LT	54-Xe-126	125.9043	8.90E-02	0.00E+00
513	LT	54-Xe-127	126.9052	0.00E+00	2.20E-07
514	LT	54-Xe-127m	126.9052	0.00E+00	1.00E-02
515	LT	54-Xe-128	127.9035	1.91E+00	0.00E+00
516	LT	54-Xe-129	128.9048	2.64E+01	0.00E+00
517	LT	54-Xe-129m	128.9048	0.00E+00	9.03E-07
518	LT	54-Xe-130	129.9035	4.07E+00	0.00E+00
519	LT	54-Xe-131	130.9051	2.12E+01	0.00E+00
520	LT	54-Xe-131m	130.9051	0.00E+00	6.78E-07
521	LT	54-Xe-132	131.9041	2.69E+01	0.00E+00
522	LT	54-Xe-133	132.9059	0.00E+00	1.53E-06
523	LT	54-Xe-133m	132.9059	0.00E+00	3.66E-06
524	LT	54-Xe-134	133.9054	1.04E+01	0.00E+00
525	LT	54-Xe-135	134.9072	0.00E+00	2.11E-05
526	LT	54-Xe-135m	134.9072	0.00E+00	7.56E-04
527	LT	54-Xe-136	135.9072	8.86E+00	0.00E+00
528	LT	54-Xe-137	136.9116	0.00E+00	3.03E-03
529	LT	55-Cs-128	127.9078	0.00E+00	3.19E-03
530	LT	55-Cs-129	128.9061	0.00E+00	6.01E-06
531	LT	55-Cs-130	129.9067	0.00E+00	3.95E-04

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
532	LT	55-Cs-131	130.9055	0.00E+00	8.28E-07
533	LT	55-Cs-132	131.9064	0.00E+00	1.24E-06
534	LT	55-Cs-133	132.9055	1.00E+02	0.00E+00
535	LT	55-Cs-134	133.9067	0.00E+00	1.06E-08
536	LT	55-Cs-134m	133.9067	0.00E+00	6.61E-05
537	LT	55-Cs-135	134.9060	0.00E+00	9.55E-15
538	LT	55-Cs-136	135.9073	0.00E+00	6.10E-07
539	LT	55-Cs-137	136.9071	0.00E+00	7.30E-10
540	LT	55-Cs-138	137.9110	0.00E+00	3.46E-04
541	LT	56-Ba-128	127.9083	0.00E+00	3.30E-06
542	LT	56-Ba-129	128.9087	0.00E+00	8.63E-05
543	LT	56-Ba-130	129.9063	1.06E-01	0.00E+00
544	LT	56-Ba-131	130.9069	0.00E+00	6.98E-07
545	LT	56-Ba-131m	130.9069	0.00E+00	7.91E-04
546	LT	56-Ba-132	131.9051	1.01E-01	0.00E+00
547	LT	56-Ba-133	132.9060	0.00E+00	2.09E-09
548	LT	56-Ba-133m	132.9060	0.00E+00	4.95E-06
549	LT	56-Ba-134	133.9045	2.42E+00	0.00E+00
550	LT	56-Ba-135	134.9057	6.59E+00	0.00E+00
551	LT	56-Ba-135m	134.9057	0.00E+00	6.71E-06
552	LT	56-Ba-136	135.9046	7.85E+00	0.00E+00
553	LT	56-Ba-136m	135.9046	0.00E+00	2.25E+00
554	LT	56-Ba-137	136.9058	1.12E+01	0.00E+00
555	LT	56-Ba-137m	136.9058	0.00E+00	4.53E-03
556	LT	56-Ba-138	137.9052	7.17E+01	0.00E+00
557	LT	56-Ba-139	138.9088	0.00E+00	1.39E-04
558	LT	56-Ba-140	139.9106	0.00E+00	6.29E-07
559	LT	56-Ba-141	140.9144	0.00E+00	6.32E-04
560	LT	57-La-134	133.9085	0.00E+00	1.79E-03
561	LT	57-La-135	134.9070	0.00E+00	9.87E-06
562	LT	57-La-136	135.9076	0.00E+00	1.17E-03
563	LT	57-La-137	136.9065	0.00E+00	3.66E-13
564	LT	57-La-138	137.9071	9.00E-02	2.15E-19
565	LT	57-La-139	138.9064	9.99E+01	0.00E+00
566	LT	57-La-140	139.9095	0.00E+00	4.78E-06
567	LT	57-La-141	140.9110	0.00E+00	4.91E-05
568	LT	58-Ce-134	133.9089	0.00E+00	2.54E-06
569	LT	58-Ce-135	134.9091	0.00E+00	1.09E-05
570	LT	58-Ce-136	135.9072	1.85E-01	0.00E+00
571	LT	58-Ce-137	136.9078	0.00E+00	2.14E-05
572	LT	58-Ce-137m	136.9078	0.00E+00	5.60E-06
573	LT	58-Ce-138	137.9060	2.51E-01	0.00E+00
574	LT	58-Ce-139	138.9066	0.00E+00	5.83E-08

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
575	LT	58-Ce-139m	138.9066	0.00E+00	1.26E-02
576	LT	58-Ce-140	139.9054	8.85E+01	0.00E+00
577	LT	58-Ce-141	140.9083	0.00E+00	2.47E-07
578	LT	58-Ce-142	141.9092	1.11E+01	0.00E+00
579	LT	58-Ce-143	142.9124	0.00E+00	5.83E-06
580	LT	58-Ce-144	143.9137	0.00E+00	2.82E-08
581	LT	58-Ce-145	144.9172	0.00E+00	3.84E-03
582	LT	59-Pr-140	139.9091	0.00E+00	3.41E-03
583	LT	59-Pr-141	140.9077	1.00E+02	0.00E+00
584	LT	59-Pr-142	141.9100	0.00E+00	1.01E-05
585	LT	59-Pr-142m	141.9100	0.00E+00	7.91E-04
586	LT	59-Pr-143	142.9108	0.00E+00	5.91E-07
587	LT	59-Pr-144	143.9133	0.00E+00	6.69E-04
588	LT	59-Pr-144m	143.9133	0.00E+00	1.60E-03
589	LT	59-Pr-145	144.9145	0.00E+00	3.22E-05
590	LT	60-Nd-140	139.9095	0.00E+00	2.38E-06
591	LT	60-Nd-141	140.9096	0.00E+00	7.73E-05
592	LT	60-Nd-141m	140.9096	0.00E+00	1.12E-02
593	LT	60-Nd-142	141.9077	2.72E+01	0.00E+00
594	LT	60-Nd-143	142.9098	1.22E+01	0.00E+00
595	LT	60-Nd-144	143.9101	2.38E+01	9.59E-24
596	LT	60-Nd-145	144.9126	8.30E+00	0.00E+00
597	LT	60-Nd-146	145.9131	1.72E+01	0.00E+00
598	LT	60-Nd-147	146.9161	0.00E+00	7.31E-07
599	LT	60-Nd-148	147.9169	5.70E+00	0.00E+00
600	LT	60-Nd-149	148.9202	0.00E+00	1.11E-04
601	LT	60-Nd-150	149.9209	5.60E+00	2.78E-27
602	LT	60-Nd-151	150.9238	0.00E+00	9.29E-04
603	LT	61-Pm-143	142.9109	0.00E+00	3.03E-08
604	LT	61-Pm-144	143.9126	0.00E+00	2.21E-08
605	LT	61-Pm-145	144.9128	0.00E+00	1.24E-09
606	LT	61-Pm-146	145.9147	0.00E+00	3.97E-09
607	LT	61-Pm-147	146.9151	0.00E+00	8.37E-09
608	LT	61-Pm-148	147.9175	0.00E+00	1.49E-06
609	LT	61-Pm-148m	147.9207	0.00E+00	1.94E-07
610	LT	61-Pm-149	148.9183	0.00E+00	3.63E-06
611	LT	61-Pm-150	149.9210	0.00E+00	7.18E-05
612	LT	61-Pm-151	150.9212	0.00E+00	6.78E-06
613	LT	61-Pm-152	151.9235	0.00E+00	2.80E-03
614	LT	62-Sm-144	143.9120	3.07E+00	0.00E+00
615	LT	62-Sm-145	144.9134	0.00E+00	2.36E-08
616	LT	62-Sm-146	145.9130	0.00E+00	2.13E-16
617	LT	62-Sm-147	146.9149	1.50E+01	2.07E-19

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
618	LT	62-Sm-148	147.9148	1.12E+01	3.14E-24
619	LT	62-Sm-149	148.9172	1.38E+01	0.00E+00
620	LT	62-Sm-150	149.9173	7.38E+00	0.00E+00
621	LT	62-Sm-151	150.9199	0.00E+00	2.44E-10
622	LT	62-Sm-152	151.9197	2.68E+01	0.00E+00
623	LT	62-Sm-153	152.9221	0.00E+00	4.14E-06
624	LT	62-Sm-154	153.9222	2.28E+01	0.00E+00
625	LT	62-Sm-155	154.9246	0.00E+00	5.18E-04
626	LT	63-Eu-145	144.9163	0.00E+00	1.35E-06
627	LT	63-Eu-146	145.9172	0.00E+00	1.75E-06
628	LT	63-Eu-147	146.9167	0.00E+00	3.33E-07
629	LT	63-Eu-148	147.9181	0.00E+00	1.47E-07
630	LT	63-Eu-149	148.9179	0.00E+00	8.62E-08
631	LT	63-Eu-150	149.9197	0.00E+00	5.95E-10
632	LT	63-Eu-150m	149.9197	0.00E+00	1.50E-05
633	LT	63-Eu-151	150.9198	4.78E+01	0.00E+00
634	LT	63-Eu-152	151.9217	0.00E+00	1.62E-09
635	LT	63-Eu-152m	151.9217	0.00E+00	2.07E-05
636	LT	63-Eu-153	152.9212	5.22E+01	0.00E+00
637	LT	63-Eu-154	153.9230	0.00E+00	2.55E-09
638	LT	63-Eu-155	154.9229	0.00E+00	4.62E-09
639	LT	63-Eu-156	155.9247	0.00E+00	5.28E-07
640	LT	63-Eu-157	156.9254	0.00E+00	1.27E-05
641	LT	64-Gd-146	145.9183	0.00E+00	1.66E-07
642	LT	64-Gd-147	146.9191	0.00E+00	5.06E-06
643	LT	64-Gd-148	147.9181	0.00E+00	2.94E-10
644	LT	64-Gd-149	148.9193	0.00E+00	8.65E-07
645	LT	64-Gd-150	149.9187	0.00E+00	1.23E-14
646	LT	64-Gd-151	150.9203	0.00E+00	6.47E-08
647	LT	64-Gd-152	151.9198	2.00E-01	2.03E-22
648	LT	64-Gd-153	152.9218	0.00E+00	3.34E-08
649	LT	64-Gd-154	153.9209	2.18E+00	0.00E+00
650	LT	64-Gd-155m	154.9226	0.00E+00	2.17E+01
651	LT	64-Gd-155	154.9226	1.48E+01	0.00E+00
652	LT	64-Gd-156	155.9221	2.05E+01	0.00E+00
653	LT	64-Gd-157	156.9240	1.57E+01	0.00E+00
654	LT	64-Gd-158	157.9241	2.48E+01	0.00E+00
655	LT	64-Gd-159	158.9264	0.00E+00	1.04E-05
656	LT	64-Gd-160	159.9270	2.19E+01	0.00E+00
657	LT	64-Gd-161	160.9297	0.00E+00	3.16E-03
658	LT	64-Gd-162	161.9310	0.00E+00	1.38E-03
659	LT	65-Tb-152	151.9241	0.00E+00	1.10E-05
660	LT	65-Tb-153	152.9234	0.00E+00	3.43E-06

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
661	LT	65-Tb-154	153.9247	0.00E+00	8.96E-06
662	LT	65-Tb-154m	153.9247	0.00E+00	2.05E-05
663	LT	65-Tb-155	154.9235	0.00E+00	1.51E-06
664	LT	65-Tb-156	155.9247	0.00E+00	1.50E-06
665	LT	65-Tb-156m	155.9247	0.00E+00	7.89E-06
666	LT	65-Tb-157	156.9240	0.00E+00	3.09E-10
667	LT	65-Tb-158	157.9254	0.00E+00	1.22E-10
668	LT	65-Tb-159	158.9254	1.00E+02	0.00E+00
669	LT	65-Tb-160	159.9272	0.00E+00	1.11E-07
670	LT	65-Tb-161	160.9276	0.00E+00	1.16E-06
671	LT	65-Tb-162	161.9295	0.00E+00	1.52E-03
672	LT	66-Dy-154	153.9244	0.00E+00	7.32E-15
673	LT	66-Dy-155	154.9258	0.00E+00	1.94E-05
674	LT	66-Dy-156	155.9243	5.60E-02	0.00E+00
675	LT	66-Dy-157	156.9255	0.00E+00	2.37E-05
676	LT	66-Dy-158	157.9244	9.50E-02	0.00E+00
677	LT	66-Dy-159	158.9257	0.00E+00	5.56E-08
678	LT	66-Dy-160	159.9252	2.33E+00	0.00E+00
679	LT	66-Dy-161	160.9269	1.89E+01	0.00E+00
680	LT	66-Dy-162	161.9268	2.55E+01	0.00E+00
681	LT	66-Dy-163	162.9287	2.49E+01	0.00E+00
682	LT	66-Dy-164	163.9292	2.83E+01	0.00E+00
683	LT	66-Dy-165	164.9317	0.00E+00	8.25E-05
684	LT	66-Dy-165m	164.9317	0.00E+00	9.19E-03
685	LT	66-Dy-166	165.9328	0.00E+00	2.36E-06
686	LT	67-Ho-160	159.9287	0.00E+00	4.51E-04
687	LT	67-Ho-160m	159.9287	0.00E+00	3.84E-05
688	LT	67-Ho-161	160.9279	0.00E+00	7.76E-05
689	LT	67-Ho-163	162.9287	0.00E+00	4.81E-12
690	LT	67-Ho-163m	162.9287	0.00E+00	6.36E-01
691	LT	67-Ho-164	163.9302	0.00E+00	3.98E-04
692	LT	67-Ho-164m	163.9302	0.00E+00	3.08E-04
693	LT	67-Ho-165	164.9303	1.00E+02	0.00E+00
694	LT	67-Ho-166	165.9323	0.00E+00	7.18E-06
695	LT	67-Ho-166m	165.9324	0.00E+00	1.83E-11
696	LT	68-Er-160	159.9291	0.00E+00	6.74E-06
697	LT	68-Er-161	160.9300	0.00E+00	6.00E-05
698	LT	68-Er-162	161.9288	1.39E-01	0.00E+00
699	LT	68-Er-163	162.9300	0.00E+00	1.54E-04
700	LT	68-Er-164	163.9292	1.60E+00	0.00E+00
701	LT	68-Er-165	164.9307	0.00E+00	1.86E-05
702	LT	68-Er-166	165.9303	3.35E+01	0.00E+00
703	LT	68-Er-167	166.9321	2.29E+01	0.00E+00

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
704	LT	68-Er-167m	166.9321	0.00E+00	3.05E-01
705	LT	68-Er-168	167.9324	2.70E+01	0.00E+00
706	LT	68-Er-169	168.9346	0.00E+00	8.54E-07
707	LT	68-Er-170	169.9355	1.49E+01	0.00E+00
708	LT	68-Er-171	170.9380	0.00E+00	2.56E-05
709	LT	68-Er-172	171.9394	0.00E+00	3.91E-06
710	LT	69-Tm-165	164.9324	0.00E+00	6.41E-06
711	LT	69-Tm-166	165.9335	0.00E+00	2.50E-05
712	LT	69-Tm-167	166.9328	0.00E+00	8.67E-07
713	LT	69-Tm-168	167.9342	0.00E+00	8.62E-08
714	LT	69-Tm-169	168.9342	1.00E+02	0.00E+00
715	LT	69-Tm-170	169.9358	0.00E+00	6.24E-08
716	LT	69-Tm-171	170.9364	0.00E+00	1.14E-08
717	LT	69-Tm-172	171.9384	0.00E+00	3.03E-06
718	LT	69-Tm-173	172.9396	0.00E+00	2.34E-05
719	LT	70-Yb-166	165.9339	0.00E+00	3.40E-06
720	LT	70-Yb-167	166.9350	0.00E+00	6.60E-04
721	LT	70-Yb-168	167.9339	1.30E-01	0.00E+00
722	LT	70-Yb-169	168.9352	0.00E+00	2.51E-07
723	LT	70-Yb-170	169.9348	3.04E+00	0.00E+00
724	LT	70-Yb-171	170.9363	1.43E+01	0.00E+00
725	LT	70-Yb-172	171.9364	2.18E+01	0.00E+00
726	LT	70-Yb-173	172.9382	1.61E+01	0.00E+00
727	LT	70-Yb-174	173.9389	3.18E+01	0.00E+00
728	LT	70-Yb-175	174.9413	0.00E+00	1.92E-06
729	LT	70-Yb-175m	174.9413	0.00E+00	1.02E+01
730	LT	70-Yb-176	175.9426	1.28E+01	0.00E+00
731	LT	70-Yb-177	176.9453	0.00E+00	1.01E-04
732	LT	71-Lu-169	168.9377	0.00E+00	5.65E-06
733	LT	71-Lu-170	169.9385	0.00E+00	3.99E-06
734	LT	71-Lu-171	170.9379	0.00E+00	9.74E-07
735	LT	71-Lu-172	171.9391	0.00E+00	1.20E-06
736	LT	71-Lu-172m	171.9391	0.00E+00	3.12E-03
737	LT	71-Lu-173	172.9389	0.00E+00	1.60E-08
738	LT	71-Lu-174	173.9403	0.00E+00	6.64E-09
739	LT	71-Lu-174m	173.9403	0.00E+00	5.65E-08
740	LT	71-Lu-175	174.9408	9.74E+01	0.00E+00
741	LT	71-Lu-176	175.9427	2.59E+00	5.84E-19
742	LT	71-Lu-176m	175.9427	0.00E+00	5.30E-05
743	LT	71-Lu-177	176.9438	0.00E+00	1.21E-06
744	LT	71-Lu-177m	176.9438	0.00E+00	5.00E-08
745	LT	72-Hf-170	169.9396	0.00E+00	1.20E-05
746	LT	72-Hf-171	170.9405	0.00E+00	1.59E-05

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
747	LT	72-Hf-172	171.9395	0.00E+00	1.17E-08
748	LT	72-Hf-173	172.9405	0.00E+00	8.16E-06
749	LT	72-Hf-174	173.9400	1.60E-01	1.10E-23
750	LT	72-Hf-175	174.9415	0.00E+00	1.15E-07
751	LT	72-Hf-176	175.9414	5.26E+00	0.00E+00
752	LT	72-Hf-177	176.9432	1.86E+01	0.00E+00
753	LT	72-Hf-177m	176.9432	0.00E+00	6.36E-01
754	LT	72-Hf-178	177.9437	2.73E+01	0.00E+00
755	LT	72-Hf-178m	177.9437	0.00E+00	1.73E-01
756	LT	72-Hf-179	178.9458	1.36E+01	0.00E+00
757	LT	72-Hf-179m	178.9458	0.00E+00	3.71E-02
758	LT	72-Hf-180	179.9465	3.51E+01	0.00E+00
759	LT	72-Hf-180m	179.9465	0.00E+00	3.50E-05
760	LT	72-Hf-181	180.9491	0.00E+00	1.89E-07
761	LT	72-Hf-182	181.9505	0.00E+00	2.47E-15
762	LT	73-Ta-177	176.9445	0.00E+00	3.40E-06
763	LT	73-Ta-178	177.9458	0.00E+00	1.24E-03
764	LT	73-Ta-179	178.9459	0.00E+00	1.21E-08
765	LT	73-Ta-180m	179.9475	1.20E-02	0.00E+00
766	LT	73-Ta-180	179.9475	0.00E+00	2.36E-05
767	LT	73-Ta-181	180.9480	1.00E+02	0.00E+00
768	LT	73-Ta-182	181.9501	0.00E+00	6.99E-08
769	LT	73-Ta-182m	181.9501	0.00E+00	2.45E+00
770	LT	73-Ta-183	182.9514	0.00E+00	1.57E-06
771	LT	74-W-178	177.9459	0.00E+00	3.71E-07
772	LT	74-W-180	179.9467	1.20E-01	1.22E-26
773	LT	74-W-181	180.9482	0.00E+00	6.62E-08
774	LT	74-W-182	181.9482	2.65E+01	0.00E+00
775	LT	74-W-183m	182.9502	0.00E+00	1.33E-01
776	LT	74-W-183	182.9502	1.43E+01	0.00E+00
777	LT	74-W-184	183.9509	3.06E+01	0.00E+00
778	LT	74-W-185	184.9534	0.00E+00	1.07E-07
779	LT	74-W-185m	184.9534	0.00E+00	6.92E-03
780	LT	74-W-186	185.9544	2.84E+01	1.29E-28
781	LT	74-W-187	186.9572	0.00E+00	8.02E-06
782	LT	74-W-188	187.9585	0.00E+00	1.15E-07
783	LT	74-W-189	188.9619	0.00E+00	1.08E-03
784	LT	75-Re-181	180.9501	0.00E+00	9.68E-06
785	LT	75-Re-182	181.9512	0.00E+00	3.01E-06
786	LT	75-Re-182m	181.9512	0.00E+00	1.52E-05
787	LT	75-Re-183	182.9508	0.00E+00	1.15E-07
788	LT	75-Re-184	183.9525	0.00E+00	2.27E-07
789	LT	75-Re-184m	183.9525	0.00E+00	4.75E-08

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
790	LT	75-Re-185	184.9530	3.74E+01	0.00E+00
791	LT	75-Re-186	185.9550	0.00E+00	2.16E-06
792	LT	75-Re-186m	185.9550	0.00E+00	1.10E-13
793	LT	75-Re-187	186.9557	6.26E+01	5.07E-19
794	LT	75-Re-188	187.9581	0.00E+00	1.13E-05
795	LT	75-Re-188m	187.9581	0.00E+00	6.21E-04
796	LT	75-Re-189	188.9592	0.00E+00	7.92E-06
797	LT	76-Os-182	181.9521	0.00E+00	8.82E-06
798	LT	76-Os-183	182.9531	0.00E+00	1.48E-05
799	LT	76-Os-184	183.9525	2.00E-02	0.00E+00
800	LT	76-Os-185	184.9540	0.00E+00	8.57E-08
801	LT	76-Os-186	185.9538	1.59E+00	1.10E-23
802	LT	76-Os-187	186.9557	1.96E+00	0.00E+00
803	LT	76-Os-188	187.9558	1.32E+01	0.00E+00
804	LT	76-Os-189	188.9581	1.62E+01	0.00E+00
805	LT	76-Os-189m	188.9581	0.00E+00	3.31E-05
806	LT	76-Os-190	189.9585	2.63E+01	0.00E+00
807	LT	76-Os-190m	189.9585	0.00E+00	1.17E-03
808	LT	76-Os-191	190.9609	0.00E+00	5.21E-07
809	LT	76-Os-191m	190.9609	0.00E+00	1.47E-05
810	LT	76-Os-192	191.9615	4.08E+01	0.00E+00
811	LT	76-Os-193	192.9642	0.00E+00	6.39E-06
812	LT	76-Os-194	193.9652	0.00E+00	3.66E-09
813	LT	77-Ir-185	184.9567	0.00E+00	1.34E-05
814	LT	77-Ir-186	185.9579	0.00E+00	1.16E-05
815	LT	77-Ir-188	187.9588	0.00E+00	4.64E-06
816	LT	77-Ir-189	188.9587	0.00E+00	6.08E-07
817	LT	77-Ir-189m	188.9587	0.00E+00	5.21E+01
818	LT	77-Ir-190	189.9606	0.00E+00	6.81E-07
819	LT	77-Ir-191	190.9606	3.73E+01	0.00E+00
820	LT	77-Ir-191m	190.9606	0.00E+00	1.41E-01
821	LT	77-Ir-192	191.9626	0.00E+00	1.09E-07
822	LT	77-Ir-192m	191.9626	0.00E+00	7.97E-03
823	LT	77-Ir-193	192.9629	6.27E+01	0.00E+00
824	LT	77-Ir-193m	192.9629	0.00E+00	7.62E-07
825	LT	77-Ir-194	193.9651	0.00E+00	9.99E-06
826	LT	77-Ir-194m	193.9651	0.00E+00	2.18E+01
827	LT	77-Ir-196	195.9684	0.00E+00	1.33E-02
828	LT	77-Ir-196m	195.9684	0.00E+00	1.38E-04
829	LT	78-Pt-188	187.9594	0.00E+00	7.87E-07
830	LT	78-Pt-189	188.9608	0.00E+00	1.77E-05
831	LT	78-Pt-190	189.9599	1.40E-02	3.38E-20
832	LT	78-Pt-191	190.9617	0.00E+00	2.86E-06

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
833	LT	78-Pt-192	191.9610	7.82E-01	0.00E+00
834	LT	78-Pt-193	192.9630	0.00E+00	4.39E-10
835	LT	78-Pt-193m	192.9630	0.00E+00	1.85E-06
836	LT	78-Pt-194	193.9627	3.30E+01	0.00E+00
837	LT	78-Pt-195	194.9648	3.38E+01	0.00E+00
838	LT	78-Pt-195m	194.9648	0.00E+00	2.00E-06
839	LT	78-Pt-196	195.9650	2.52E+01	0.00E+00
840	LT	78-Pt-197	196.9673	0.00E+00	9.68E-06
841	LT	78-Pt-197m	196.9673	0.00E+00	1.21E-04
842	LT	78-Pt-198	197.9679	7.16E+00	0.00E+00
843	LT	78-Pt-199	198.9706	0.00E+00	3.75E-04
844	LT	78-Pt-199m	198.9706	0.00E+00	5.10E-02
845	LT	78-Pt-200	199.9714	0.00E+00	1.53E-05
846	LT	79-Au-193	192.9642	0.00E+00	1.09E-05
847	LT	79-Au-194	193.9654	0.00E+00	5.06E-06
848	LT	79-Au-195	194.9650	0.00E+00	4.31E-08
849	LT	79-Au-195m	194.9650	0.00E+00	2.27E-02
850	LT	79-Au-196	195.9666	0.00E+00	1.30E-06
851	LT	79-Au-197	196.9666	1.00E+02	0.00E+00
852	LT	79-Au-197m	196.9666	0.00E+00	8.97E-02
853	LT	79-Au-198	197.9682	0.00E+00	2.98E-06
854	LT	79-Au-198m	197.9682	0.00E+00	3.53E-06
855	LT	79-Au-199	198.9688	0.00E+00	2.56E-06
856	LT	79-Au-200	199.9707	0.00E+00	2.39E-04
857	LT	79-Au-200m	199.9707	0.00E+00	1.03E-05
858	LT	80-Hg-193	192.9667	0.00E+00	5.07E-05
859	LT	80-Hg-193m	192.9667	0.00E+00	1.63E-05
860	LT	80-Hg-194	193.9654	0.00E+00	4.95E-11
861	LT	80-Hg-195	194.9667	0.00E+00	1.83E-05
862	LT	80-Hg-195m	194.9667	0.00E+00	4.63E-06
863	LT	80-Hg-196	195.9658	1.50E-01	0.00E+00
864	LT	80-Hg-197	196.9672	0.00E+00	3.00E-06
865	LT	80-Hg-197m	196.9672	0.00E+00	8.09E-06
866	LT	80-Hg-198	197.9668	9.97E+00	0.00E+00
867	LT	80-Hg-199	198.9683	1.69E+01	0.00E+00
868	LT	80-Hg-199m	198.9683	0.00E+00	2.71E-04
869	LT	80-Hg-200	199.9683	2.31E+01	0.00E+00
870	LT	80-Hg-201	200.9703	1.32E+01	0.00E+00
871	LT	80-Hg-202	201.9706	2.99E+01	0.00E+00
872	LT	80-Hg-203	202.9729	0.00E+00	1.72E-07
873	LT	80-Hg-204	203.9735	6.87E+00	0.00E+00
874	LT	80-Hg-205	204.9761	0.00E+00	2.25E-03
875	LT	80-Hg-206	205.9775	0.00E+00	1.39E-03

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
876	LT	81-Tl-200	199.9710	0.00E+00	7.38E-06
877	LT	81-Tl-201	200.9708	0.00E+00	2.64E-06
878	LT	81-Tl-202	201.9721	0.00E+00	6.52E-07
879	LT	81-Tl-203	202.9723	2.95E+01	0.00E+00
880	LT	81-Tl-204	203.9739	0.00E+00	5.81E-09
881	LT	81-Tl-205	204.9744	7.05E+01	0.00E+00
882	LT	81-Tl-206	205.9761	0.00E+00	2.75E-03
883	LT	81-Tl-207	206.9774	0.00E+00	2.42E-03
884	LT	82-Pb-200	199.9718	0.00E+00	8.96E-06
885	LT	82-Pb-202	201.9722	0.00E+00	4.18E-13
886	LT	82-Pb-203	202.9734	0.00E+00	3.71E-06
887	LT	82-Pb-204	203.9730	1.40E+00	1.58E-25
888	LT	82-Pb-205	204.9745	0.00E+00	1.27E-15
889	LT	82-Pb-205m	204.9745	0.00E+00	1.25E+02
890	LT	82-Pb-206	205.9745	2.41E+01	0.00E+00
891	LT	82-Pb-207	206.9759	2.21E+01	0.00E+00
892	LT	82-Pb-207m	206.9759	0.00E+00	8.60E-01
893	LT	82-Pb-208	207.9767	5.24E+01	0.00E+00
894	LT	82-Pb-209	208.9811	0.00E+00	5.92E-05
895	LT	82-Pb-210	209.9842	0.00E+00	9.89E-10
896	LT	83-Bi-205	204.9774	0.00E+00	5.24E-07
897	LT	83-Bi-206	205.9785	0.00E+00	1.29E-06
898	LT	83-Bi-207	206.9785	0.00E+00	6.96E-10
899	LT	83-Bi-208	207.9797	0.00E+00	5.97E-14
900	LT	83-Bi-209	208.9804	1.00E+02	1.16E-27
901	LT	83-Bi-210	209.9841	0.00E+00	1.60E-06
902	LT	83-Bi-210m	209.9841	0.00E+00	7.23E-15
903	LT	83-Bi-211	210.9873	0.00E+00	5.40E-03
904	LT	84-Po-206	205.9805	0.00E+00	9.12E-07
905	LT	84-Po-207	206.9816	0.00E+00	3.32E-05
906	LT	84-Po-208	207.9812	0.00E+00	7.58E-09
907	LT	84-Po-209	208.9824	0.00E+00	2.15E-10
908	LT	84-Po-210	209.9829	0.00E+00	5.80E-08
909	LT	84-Po-211	210.9866	0.00E+00	1.34E+00
910	LT	84-Po-211m	210.9866	0.00E+00	2.75E-02
911	AC	2-He-3	3.0160	0.00E+00	0.00E+00
912	AC	2-He-4	4.0026	0.00E+00	0.00E+00
913	AC	3-Li-6	6.0151	0.00E+00	0.00E+00
914	AC	3-Li-7	7.0160	0.00E+00	0.00E+00
915	AC	4-Be-7	7.0169	0.00E+00	1.51E-07
916	AC	6-C-12	12.0000	0.00E+00	0.00E+00
917	AC	80-Hg-206	205.9775	0.00E+00	1.39E-03
918	AC	81-Tl-203	202.9723	0.00E+00	0.00E+00

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
919	AC	81-Tl-205	204.9744	0.00E+00	0.00E+00
920	AC	81-Tl-206	205.9761	0.00E+00	2.75E-03
921	AC	81-Tl-207	206.9774	0.00E+00	2.42E-03
922	AC	81-Tl-208	207.9820	0.00E+00	3.78E-03
923	AC	81-Tl-209	208.9854	0.00E+00	5.25E-03
924	AC	81-Tl-210	209.9901	0.00E+00	8.89E-03
925	AC	82-Pb-203	202.9734	0.00E+00	3.71E-06
926	AC	82-Pb-204	203.9730	0.00E+00	1.58E-25
927	AC	82-Pb-205	204.9745	0.00E+00	1.27E-15
928	AC	82-Pb-206	205.9745	0.00E+00	0.00E+00
929	AC	82-Pb-207	206.9759	0.00E+00	0.00E+00
930	AC	82-Pb-207m	206.9759	0.00E+00	8.60E-01
931	AC	82-Pb-208	207.9767	0.00E+00	0.00E+00
932	AC	82-Pb-209	208.9811	0.00E+00	5.92E-05
933	AC	82-Pb-210	209.9842	0.00E+00	9.89E-10
934	AC	82-Pb-211	210.9887	0.00E+00	3.20E-04
935	AC	82-Pb-212	211.9919	0.00E+00	1.81E-05
936	AC	82-Pb-214	213.9998	0.00E+00	4.31E-04
937	AC	83-Bi-206	205.9785	0.00E+00	1.29E-06
938	AC	83-Bi-207	206.9785	0.00E+00	6.96E-10
939	AC	83-Bi-208	207.9797	0.00E+00	5.97E-14
940	AC	83-Bi-209	208.9804	0.00E+00	1.16E-27
941	AC	83-Bi-210m	209.9841	0.00E+00	7.23E-15
942	AC	83-Bi-210	209.9841	0.00E+00	1.60E-06
943	AC	83-Bi-211	210.9873	0.00E+00	5.40E-03
944	AC	83-Bi-212	211.9913	0.00E+00	1.91E-04
945	AC	83-Bi-212m	211.9913	0.00E+00	4.62E-04
946	AC	83-Bi-213	212.9944	0.00E+00	2.53E-04
947	AC	83-Bi-214	213.9987	0.00E+00	5.81E-04
948	AC	84-Po-207	206.9816	0.00E+00	3.32E-05
949	AC	84-Po-208	207.9812	0.00E+00	7.58E-09
950	AC	84-Po-209	208.9824	0.00E+00	2.15E-10
951	AC	84-Po-210	209.9829	0.00E+00	5.80E-08
952	AC	84-Po-211m	210.9866	0.00E+00	2.75E-02
953	AC	84-Po-211	210.9866	0.00E+00	1.34E+00
954	AC	84-Po-212	211.9889	0.00E+00	6.93E+02
955	AC	84-Po-213	212.9929	0.00E+00	6.93E+02
956	AC	84-Po-214	213.9952	0.00E+00	6.93E+02
957	AC	84-Po-215	214.9994	0.00E+00	3.89E+02
958	AC	84-Po-216	216.0019	0.00E+00	4.78E+00
959	AC	84-Po-218	218.0090	0.00E+00	3.73E-03
960	AC	85-At-216	216.0024	0.00E+00	6.93E+02
961	AC	85-At-217	217.0047	0.00E+00	2.15E+01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
962	AC	85-At-218	218.0087	0.00E+00	4.62E-01
963	AC	86-Rn-216	216.0003	0.00E+00	6.93E+02
964	AC	86-Rn-217	217.0039	0.00E+00	6.93E+02
965	AC	86-Rn-218	218.0056	0.00E+00	1.98E+01
966	AC	86-Rn-219	219.0095	0.00E+00	1.75E-01
967	AC	86-Rn-220	220.0114	0.00E+00	1.25E-02
968	AC	86-Rn-222	222.0176	0.00E+00	2.10E-06
969	AC	87-Fr-220	220.0123	0.00E+00	2.53E-02
970	AC	87-Fr-221	221.0143	0.00E+00	2.36E-03
971	AC	87-Fr-222	222.0175	0.00E+00	8.14E-04
972	AC	87-Fr-223	223.0197	0.00E+00	5.25E-04
973	AC	88-Ra-220	220.0110	0.00E+00	3.85E+01
974	AC	88-Ra-222	222.0154	0.00E+00	1.92E-02
975	AC	88-Ra-223	223.0185	0.00E+00	7.02E-07
976	AC	88-Ra-224	224.0202	0.00E+00	2.19E-06
977	AC	88-Ra-225	225.0236	0.00E+00	5.38E-07
978	AC	88-Ra-226	226.0254	0.00E+00	1.37E-11
979	AC	88-Ra-227	227.0292	0.00E+00	2.74E-04
980	AC	88-Ra-228	228.0311	0.00E+00	3.82E-09
981	AC	89-Ac-224	224.0217	0.00E+00	6.93E-05
982	AC	89-Ac-225	225.0232	0.00E+00	8.02E-07
983	AC	89-Ac-226	226.0261	0.00E+00	6.56E-06
984	AC	89-Ac-227	227.0278	0.00E+00	1.01E-09
985	AC	89-Ac-228	228.0310	0.00E+00	3.13E-05
986	AC	90-Th-226	226.0249	0.00E+00	3.78E-04
987	AC	90-Th-227	227.0277	0.00E+00	4.29E-07
988	AC	90-Th-228	228.0287	0.00E+00	1.15E-08
989	AC	90-Th-229	229.0318	0.00E+00	2.99E-12
990	AC	90-Th-230	230.0331	0.00E+00	2.91E-13
991	AC	90-Th-231	231.0363	0.00E+00	7.55E-06
992	AC	90-Th-232	232.0381	0.00E+00	1.56E-18
993	AC	90-Th-233	233.0416	0.00E+00	5.18E-04
994	AC	90-Th-234	234.0436	0.00E+00	3.33E-07
995	AC	91-Pa-228	228.0311	0.00E+00	8.75E-06
996	AC	91-Pa-229	229.0321	0.00E+00	5.35E-06
997	AC	91-Pa-230	230.0345	0.00E+00	4.61E-07
998	AC	91-Pa-231	231.0359	0.00E+00	6.70E-13
999	AC	91-Pa-232	232.0386	0.00E+00	6.08E-06
1000	AC	91-Pa-233	233.0403	0.00E+00	2.97E-07
1001	AC	91-Pa-234m	234.0433	0.00E+00	9.97E-03
1002	AC	91-Pa-234	234.0433	0.00E+00	2.87E-05
1003	AC	91-Pa-235	235.0454	0.00E+00	4.73E-04
1004	AC	92-U-230	230.0339	0.00E+00	3.86E-07

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1005	AC	92-U-231	231.0363	0.00E+00	1.91E-06
1006	AC	92-U-232	232.0372	0.00E+00	3.19E-10
1007	AC	92-U-233	233.0396	0.00E+00	1.38E-13
1008	AC	92-U-234	234.0410	0.00E+00	8.95E-14
1009	AC	92-U-235	235.0439	0.00E+00	3.12E-17
1010	AC	92-U-235m	235.0439	0.00E+00	4.44E-04
1011	AC	92-U-236	236.0456	0.00E+00	9.38E-16
1012	AC	92-U-237	237.0487	0.00E+00	1.19E-06
1013	AC	92-U-238	238.0508	0.00E+00	4.92E-18
1014	AC	92-U-239	239.0543	0.00E+00	4.93E-04
1015	AC	92-U-240	240.0566	0.00E+00	1.37E-05
1016	AC	92-U-241	241.0603	0.00E+00	2.31E-03
1017	AC	93-Np-234	234.0429	0.00E+00	1.82E-06
1018	AC	93-Np-235	235.0441	0.00E+00	2.03E-08
1019	AC	93-Np-236m	236.0466	0.00E+00	8.56E-06
1020	AC	93-Np-236	236.0466	0.00E+00	1.44E-13
1021	AC	93-Np-237	237.0482	0.00E+00	1.02E-14
1022	AC	93-Np-238	238.0509	0.00E+00	3.79E-06
1023	AC	93-Np-239	239.0529	0.00E+00	3.41E-06
1024	AC	93-Np-240m	240.0562	0.00E+00	1.60E-03
1025	AC	93-Np-240	240.0562	0.00E+00	1.87E-04
1026	AC	93-Np-241	241.0582	0.00E+00	8.31E-04
1027	AC	94-Pu-236	236.0461	0.00E+00	7.69E-09
1028	AC	94-Pu-237m	237.0484	0.00E+00	3.85E+00
1029	AC	94-Pu-237	237.0484	0.00E+00	1.76E-07
1030	AC	94-Pu-238	238.0496	0.00E+00	2.50E-10
1031	AC	94-Pu-239	239.0522	0.00E+00	9.11E-13
1032	AC	94-Pu-240	240.0538	0.00E+00	3.35E-12
1033	AC	94-Pu-241	241.0569	0.00E+00	1.54E-09
1034	AC	94-Pu-242	242.0587	0.00E+00	5.88E-14
1035	AC	94-Pu-243	243.0620	0.00E+00	3.89E-05
1036	AC	94-Pu-244	244.0642	0.00E+00	2.71E-16
1037	AC	94-Pu-245	245.0677	0.00E+00	1.83E-05
1038	AC	94-Pu-246	246.0702	0.00E+00	7.40E-07
1039	AC	94-Pu-247	247.0741	0.00E+00	3.53E-06
1040	AC	95-Am-239	239.0530	0.00E+00	1.62E-05
1041	AC	95-Am-240	240.0553	0.00E+00	3.79E-06
1042	AC	95-Am-241	241.0568	0.00E+00	5.08E-11
1043	AC	95-Am-242m	242.0595	0.00E+00	1.56E-10
1044	AC	95-Am-242	242.0596	0.00E+00	1.20E-05
1045	AC	95-Am-243	243.0614	0.00E+00	2.98E-12
1046	AC	95-Am-244m	244.0646	0.00E+00	4.44E-04
1047	AC	95-Am-244	244.0643	0.00E+00	1.91E-05

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1048	AC	95-Am-245	245.0665	0.00E+00	9.39E-05
1049	AC	95-Am-246	246.0698	0.00E+00	2.96E-04
1050	AC	95-Am-246m	246.0698	0.00E+00	4.62E-04
1051	AC	95-Am-247	247.0721	0.00E+00	5.02E-04
1052	AC	96-Cm-240	240.0555	0.00E+00	2.97E-07
1053	AC	96-Cm-241	241.0576	0.00E+00	2.45E-07
1054	AC	96-Cm-242	242.0588	0.00E+00	4.92E-08
1055	AC	96-Cm-243	243.0614	0.00E+00	7.55E-10
1056	AC	96-Cm-244	244.0627	0.00E+00	1.21E-09
1057	AC	96-Cm-245	245.0655	0.00E+00	2.58E-12
1058	AC	96-Cm-246	246.0672	0.00E+00	4.61E-12
1059	AC	96-Cm-247	247.0703	0.00E+00	1.41E-15
1060	AC	96-Cm-248	248.0724	0.00E+00	6.31E-14
1061	AC	96-Cm-249	249.0759	0.00E+00	1.80E-04
1062	AC	96-Cm-250	250.0784	0.00E+00	2.65E-12
1063	AC	96-Cm-251	251.0823	0.00E+00	6.88E-04
1064	AC	97-Bk-245	245.0664	0.00E+00	1.62E-06
1065	AC	97-Bk-246	246.0687	0.00E+00	4.46E-06
1066	AC	97-Bk-247	247.0703	0.00E+00	1.59E-11
1067	AC	97-Bk-248	248.0731	0.00E+00	2.44E-09
1068	AC	97-Bk-248m	248.0731	0.00E+00	8.12E-06
1069	AC	97-Bk-249	249.0750	0.00E+00	2.51E-08
1070	AC	97-Bk-250	250.0783	0.00E+00	5.99E-05
1071	AC	97-Bk-251	251.0808	0.00E+00	2.08E-04
1072	AC	98-Cf-246	246.0688	0.00E+00	5.39E-06
1073	AC	98-Cf-248	248.0722	0.00E+00	2.41E-08
1074	AC	98-Cf-249	249.0748	0.00E+00	6.26E-11
1075	AC	98-Cf-250	250.0764	0.00E+00	1.68E-09
1076	AC	98-Cf-251	251.0796	0.00E+00	2.45E-11
1077	AC	98-Cf-252	252.0816	0.00E+00	8.30E-09
1078	AC	98-Cf-253	253.0851	0.00E+00	4.50E-07
1079	AC	98-Cf-254	254.0873	0.00E+00	1.33E-07
1080	AC	98-Cf-255	255.0910	0.00E+00	1.36E-04
1081	AC	99-Es-251	251.0800	0.00E+00	5.83E-06
1082	AC	99-Es-252	252.0830	0.00E+00	1.70E-08
1083	AC	99-Es-253	253.0848	0.00E+00	3.92E-07
1084	AC	99-Es-254m	254.0880	0.00E+00	4.90E-06
1085	AC	99-Es-254	254.0880	0.00E+00	2.91E-08
1086	AC	99-Es-255	255.0903	0.00E+00	2.02E-07
1087	FP	1-H-3	3.0161	0.00E+00	1.78E-09
1088	FP	2-He-3	3.0160	0.00E+00	0.00E+00
1089	FP	2-He-4	4.0026	0.00E+00	0.00E+00
1090	FP	26-Fe-65	64.9454	0.00E+00	8.56E-01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1091	FP	27-Co-65	64.9365	0.00E+00	5.98E-01
1092	FP	28-Ni-65	64.9301	0.00E+00	7.65E-05
1093	FP	29-Cu-65	64.9278	0.00E+00	0.00E+00
1094	FP	24-Cr-66	65.9734	0.00E+00	6.93E+01
1095	FP	25-Mn-66	65.9611	0.00E+00	1.08E+01
1096	FP	26-Fe-66	65.9468	0.00E+00	1.58E+00
1097	FP	27-Co-66	65.9398	0.00E+00	3.47E+00
1098	FP	28-Ni-66	65.9291	0.00E+00	3.53E-06
1099	FP	29-Cu-66	65.9289	0.00E+00	2.26E-03
1100	FP	30-Zn-66	65.9260	0.00E+00	0.00E+00
1101	FP	31-Ga-66	65.9316	0.00E+00	2.03E-05
1102	FP	32-Ge-66	65.9338	0.00E+00	8.52E-05
1103	FP	24-Cr-67	66.9796	0.00E+00	1.39E+01
1104	FP	25-Mn-67	66.9641	0.00E+00	1.47E+01
1105	FP	26-Fe-67	66.9510	0.00E+00	1.67E+00
1106	FP	27-Co-67	66.9409	0.00E+00	1.63E+00
1107	FP	28-Ni-67	66.9316	0.00E+00	3.30E-02
1108	FP	29-Cu-67	66.9277	0.00E+00	3.11E-06
1109	FP	30-Zn-67	66.9271	0.00E+00	0.00E+00
1110	FP	31-Ga-67	66.9282	0.00E+00	2.46E-06
1111	FP	32-Ge-67	66.9327	0.00E+00	6.11E-04
1112	FP	25-Mn-68	67.9693	0.00E+00	2.48E+01
1113	FP	26-Fe-68	67.9537	0.00E+00	3.71E+00
1114	FP	27-Co-68	67.9449	0.00E+00	3.48E+00
1115	FP	28-Ni-68	67.9319	0.00E+00	2.39E-02
1116	FP	29-Cu-68	67.9296	0.00E+00	2.23E-02
1117	FP	29-Cu-68m	67.9296	0.00E+00	3.08E-03
1118	FP	30-Zn-68	67.9248	0.00E+00	0.00E+00
1119	FP	31-Ga-68	67.9280	0.00E+00	1.71E-04
1120	FP	32-Ge-68	67.9281	0.00E+00	2.96E-08
1121	FP	25-Mn-69	68.9728	0.00E+00	4.95E+01
1122	FP	26-Fe-69	68.9588	0.00E+00	6.36E+00
1123	FP	27-Co-69	68.9463	0.00E+00	3.15E+00
1124	FP	28-Ni-69	68.9356	0.00E+00	6.08E-02
1125	FP	29-Cu-69	68.9294	0.00E+00	4.05E-03
1126	FP	30-Zn-69	68.9266	0.00E+00	2.05E-04
1127	FP	30-Zn-69m	68.9266	0.00E+00	1.40E-05
1128	FP	31-Ga-69	68.9256	0.00E+00	0.00E+00
1129	FP	32-Ge-69	68.9280	0.00E+00	4.93E-06
1130	FP	33-As-69	68.9323	0.00E+00	7.59E-04
1131	FP	26-Fe-70	69.9615	0.00E+00	7.37E+00
1132	FP	27-Co-70	69.9510	0.00E+00	5.82E+00
1133	FP	28-Ni-70	69.9365	0.00E+00	1.16E-01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1134	FP	29-Cu-70	69.9324	0.00E+00	1.56E-02
1135	FP	29-Cu-70m	69.9324	0.00E+00	2.10E-02
1136	FP	30-Zn-70	69.9253	0.00E+00	0.00E+00
1137	FP	31-Ga-70	69.9260	0.00E+00	5.46E-04
1138	FP	32-Ge-70	69.9242	0.00E+00	0.00E+00
1139	FP	26-Fe-71	70.9667	0.00E+00	2.48E+01
1140	FP	27-Co-71	70.9529	0.00E+00	8.77E+00
1141	FP	28-Ni-71	70.9407	0.00E+00	2.71E-01
1142	FP	29-Cu-71	70.9327	0.00E+00	3.55E-02
1143	FP	30-Zn-71	70.9277	0.00E+00	4.72E-03
1144	FP	30-Zn-71m	70.9277	0.00E+00	4.86E-05
1145	FP	31-Ga-71	70.9247	0.00E+00	0.00E+00
1146	FP	32-Ge-71	70.9249	0.00E+00	7.02E-07
1147	FP	32-Ge-71m	70.9249	0.00E+00	3.40E+01
1148	FP	33-As-71	70.9271	0.00E+00	2.95E-06
1149	FP	26-Fe-72	71.9696	0.00E+00	6.93E+02
1150	FP	27-Co-72	71.9578	0.00E+00	1.16E+01
1151	FP	28-Ni-72	71.9421	0.00E+00	4.42E-01
1152	FP	29-Cu-72	71.9358	0.00E+00	1.05E-01
1153	FP	30-Zn-72	71.9269	0.00E+00	4.14E-06
1154	FP	31-Ga-72	71.9264	0.00E+00	1.37E-05
1155	FP	31-Ga-72m	71.9264	0.00E+00	1.75E+01
1156	FP	32-Ge-72	71.9221	0.00E+00	0.00E+00
1157	FP	33-As-72	71.9268	0.00E+00	7.41E-06
1158	FP	34-Se-72	71.9271	0.00E+00	9.55E-07
1159	FP	27-Co-73	72.9602	0.00E+00	1.69E+01
1160	FP	28-Ni-73	72.9465	0.00E+00	8.25E-01
1161	FP	29-Cu-73	72.9367	0.00E+00	1.65E-01
1162	FP	30-Zn-73	72.9298	0.00E+00	2.95E-02
1163	FP	31-Ga-73	72.9252	0.00E+00	3.96E-05
1164	FP	32-Ge-73	72.9235	0.00E+00	0.00E+00
1165	FP	32-Ge-73m	72.9235	0.00E+00	1.39E+00
1166	FP	33-As-73	72.9238	0.00E+00	9.99E-08
1167	FP	34-Se-73	72.9268	0.00E+00	2.69E-05
1168	FP	34-Se-73m	72.9268	0.00E+00	2.90E-04
1169	FP	27-Co-74	73.9654	0.00E+00	2.31E+01
1170	FP	28-Ni-74	73.9481	0.00E+00	1.02E+00
1171	FP	29-Cu-74	73.9399	0.00E+00	3.96E-01
1172	FP	30-Zn-74	73.9295	0.00E+00	7.25E-03
1173	FP	31-Ga-74	73.9269	0.00E+00	1.42E-03
1174	FP	31-Ga-74m	73.9269	0.00E+00	7.30E-02
1175	FP	32-Ge-74	73.9212	0.00E+00	0.00E+00
1176	FP	33-As-74	73.9239	0.00E+00	4.51E-07

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1177	FP	34-Se-74	73.9225	0.00E+00	0.00E+00
1178	FP	27-Co-75	74.9683	0.00E+00	2.04E+01
1179	FP	28-Ni-75	74.9529	0.00E+00	1.16E+00
1180	FP	29-Cu-75	74.9419	0.00E+00	5.66E-01
1181	FP	30-Zn-75	74.9329	0.00E+00	6.80E-02
1182	FP	31-Ga-75	74.9265	0.00E+00	5.50E-03
1183	FP	32-Ge-75	74.9229	0.00E+00	1.40E-04
1184	FP	32-Ge-75m	74.9229	0.00E+00	1.45E-02
1185	FP	33-As-75	74.9216	0.00E+00	0.00E+00
1186	FP	33-As-75m	74.9216	0.00E+00	3.93E+01
1187	FP	34-Se-75	74.9225	0.00E+00	6.70E-08
1188	FP	35-Br-75	74.9258	0.00E+00	1.19E-04
1189	FP	28-Ni-76	75.9553	0.00E+00	2.91E+00
1190	FP	29-Cu-76	75.9453	0.00E+00	1.06E+00
1191	FP	30-Zn-76	75.9333	0.00E+00	1.22E-01
1192	FP	31-Ga-76	75.9288	0.00E+00	2.13E-02
1193	FP	32-Ge-76	75.9214	0.00E+00	0.00E+00
1194	FP	33-As-76	75.9224	0.00E+00	7.34E-06
1195	FP	34-Se-76	75.9192	0.00E+00	0.00E+00
1196	FP	28-Ni-77	76.9605	0.00E+00	1.14E+01
1197	FP	29-Cu-77	76.9479	0.00E+00	1.48E+00
1198	FP	30-Zn-77	76.9370	0.00E+00	3.33E-01
1199	FP	31-Ga-77	76.9292	0.00E+00	5.25E-02
1200	FP	32-Ge-77	76.9236	0.00E+00	1.70E-05
1201	FP	32-Ge-77m	76.9236	0.00E+00	1.31E-02
1202	FP	33-As-77	76.9206	0.00E+00	4.96E-06
1203	FP	34-Se-77	76.9199	0.00E+00	0.00E+00
1204	FP	34-Se-77m	76.9199	0.00E+00	3.99E-02
1205	FP	35-Br-77	76.9214	0.00E+00	3.38E-06
1206	FP	35-Br-77m	76.9214	0.00E+00	2.70E-03
1207	FP	36-Kr-77	76.9247	0.00E+00	1.55E-04
1208	FP	28-Ni-78	77.9632	0.00E+00	6.30E+00
1209	FP	29-Cu-78	77.9520	0.00E+00	2.07E+00
1210	FP	30-Zn-78	77.9384	0.00E+00	4.72E-01
1211	FP	31-Ga-78	77.9316	0.00E+00	1.36E-01
1212	FP	32-Ge-78	77.9229	0.00E+00	1.31E-04
1213	FP	33-As-78	77.9218	0.00E+00	1.27E-04
1214	FP	34-Se-78	77.9173	0.00E+00	0.00E+00
1215	FP	35-Br-78	77.9212	0.00E+00	1.79E-03
1216	FP	36-Kr-78	77.9204	0.00E+00	0.00E+00
1217	FP	29-Cu-79	78.9546	0.00E+00	3.69E+00
1218	FP	30-Zn-79	78.9426	0.00E+00	6.97E-01
1219	FP	31-Ga-79	78.9329	0.00E+00	2.43E-01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1220	FP	32-Ge-79	78.9254	0.00E+00	3.65E-02
1221	FP	32-Ge-79m	78.9254	0.00E+00	1.78E-02
1222	FP	33-As-79	78.9210	0.00E+00	1.28E-03
1223	FP	34-Se-79	78.9185	0.00E+00	7.45E-14
1224	FP	34-Se-79m	78.9185	0.00E+00	2.95E-03
1225	FP	35-Br-79	78.9183	0.00E+00	0.00E+00
1226	FP	35-Br-79m	78.9183	0.00E+00	1.43E-01
1227	FP	36-Kr-79	78.9201	0.00E+00	5.49E-06
1228	FP	36-Kr-79m	78.9201	0.00E+00	1.39E-02
1229	FP	37-Rb-79	78.9240	0.00E+00	5.04E-04
1230	FP	29-Cu-80	79.9609	0.00E+00	4.08E+00
1231	FP	30-Zn-80	79.9443	0.00E+00	1.28E+00
1232	FP	31-Ga-80	79.9365	0.00E+00	4.14E-01
1233	FP	32-Ge-80	79.9254	0.00E+00	2.35E-02
1234	FP	33-As-80	79.9225	0.00E+00	4.56E-02
1235	FP	34-Se-80	79.9165	0.00E+00	0.00E+00
1236	FP	35-Br-80	79.9185	0.00E+00	6.53E-04
1237	FP	35-Br-80m	79.9185	0.00E+00	4.36E-05
1238	FP	36-Kr-80	79.9164	0.00E+00	0.00E+00
1239	FP	30-Zn-81	80.9505	0.00E+00	2.17E+00
1240	FP	31-Ga-81	80.9378	0.00E+00	5.70E-01
1241	FP	32-Ge-81	80.9288	0.00E+00	9.12E-02
1242	FP	32-Ge-81m	80.9288	0.00E+00	9.12E-02
1243	FP	33-As-81	80.9221	0.00E+00	2.08E-02
1244	FP	34-Se-81	80.9180	0.00E+00	6.26E-04
1245	FP	34-Se-81m	80.9180	0.00E+00	2.02E-04
1246	FP	35-Br-81	80.9163	0.00E+00	0.00E+00
1247	FP	36-Kr-81	80.9166	0.00E+00	9.59E-14
1248	FP	36-Kr-81m	80.9166	0.00E+00	5.29E-02
1249	FP	37-Rb-81	80.9190	0.00E+00	4.21E-05
1250	FP	30-Zn-82	81.9544	0.00E+00	1.33E+01
1251	FP	31-Ga-82	81.9430	0.00E+00	1.16E+00
1252	FP	32-Ge-82	81.9296	0.00E+00	1.52E-01
1253	FP	33-As-82	81.9245	0.00E+00	3.63E-02
1254	FP	33-As-82m	81.9245	0.00E+00	5.10E-02
1255	FP	34-Se-82	81.9167	0.00E+00	0.00E+00
1256	FP	35-Br-82	81.9168	0.00E+00	5.46E-06
1257	FP	35-Br-82m	81.9168	0.00E+00	1.88E-03
1258	FP	36-Kr-82	81.9135	0.00E+00	0.00E+00
1259	FP	30-Zn-83	82.9610	0.00E+00	1.61E+01
1260	FP	31-Ga-83	82.9470	0.00E+00	2.25E+00
1261	FP	32-Ge-83	82.9346	0.00E+00	3.75E-01
1262	FP	33-As-83	82.9250	0.00E+00	5.17E-02

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1263	FP	34-Se-83	82.9191	0.00E+00	5.18E-04
1264	FP	34-Se-83m	82.9191	0.00E+00	9.89E-03
1265	FP	35-Br-83	82.9152	0.00E+00	8.02E-05
1266	FP	36-Kr-83	82.9141	0.00E+00	0.00E+00
1267	FP	36-Kr-83m	82.9141	0.00E+00	1.05E-04
1268	FP	37-Rb-83	82.9151	0.00E+00	9.31E-08
1269	FP	38-Sr-83	82.9176	0.00E+00	5.94E-06
1270	FP	31-Ga-84	83.9527	0.00E+00	8.15E+00
1271	FP	32-Ge-84	83.9375	0.00E+00	7.27E-01
1272	FP	33-As-84	83.9291	0.00E+00	1.65E-01
1273	FP	34-Se-84	83.9185	0.00E+00	3.54E-03
1274	FP	35-Br-84	83.9165	0.00E+00	3.64E-04
1275	FP	35-Br-84m	83.9165	0.00E+00	1.93E-03
1276	FP	36-Kr-84	83.9115	0.00E+00	0.00E+00
1277	FP	37-Rb-84	83.9144	0.00E+00	2.44E-07
1278	FP	38-Sr-84	83.9134	0.00E+00	0.00E+00
1279	FP	31-Ga-85	84.9570	0.00E+00	1.44E+01
1280	FP	32-Ge-85	84.9430	0.00E+00	1.30E+00
1281	FP	33-As-85	84.9320	0.00E+00	3.43E-01
1282	FP	34-Se-85	84.9222	0.00E+00	2.19E-02
1283	FP	35-Br-85	84.9156	0.00E+00	3.98E-03
1284	FP	36-Kr-85	84.9125	0.00E+00	2.04E-09
1285	FP	36-Kr-85m	84.9125	0.00E+00	4.30E-05
1286	FP	37-Rb-85	84.9118	0.00E+00	0.00E+00
1287	FP	38-Sr-85	84.9129	0.00E+00	1.24E-07
1288	FP	38-Sr-85m	84.9129	0.00E+00	1.71E-04
1289	FP	39-Y-85	84.9164	0.00E+00	7.18E-05
1290	FP	31-Ga-86	85.9631	0.00E+00	2.39E+01
1291	FP	32-Ge-86	85.9465	0.00E+00	7.30E+00
1292	FP	33-As-86	85.9365	0.00E+00	7.33E-01
1293	FP	34-Se-86	85.9243	0.00E+00	4.85E-02
1294	FP	35-Br-86	85.9188	0.00E+00	1.26E-02
1295	FP	36-Kr-86	85.9106	0.00E+00	0.00E+00
1296	FP	37-Rb-86	85.9112	0.00E+00	4.31E-07
1297	FP	37-Rb-86m	85.9112	0.00E+00	1.14E-02
1298	FP	38-Sr-86	85.9093	0.00E+00	0.00E+00
1299	FP	32-Ge-87	86.9525	0.00E+00	4.95E+00
1300	FP	33-As-87	86.9399	0.00E+00	1.24E+00
1301	FP	34-Se-87	86.9285	0.00E+00	1.26E-01
1302	FP	35-Br-87	86.9207	0.00E+00	1.25E-02
1303	FP	36-Kr-87	86.9134	0.00E+00	1.51E-04
1304	FP	37-Rb-87	86.9092	0.00E+00	4.57E-19
1305	FP	38-Sr-87	86.9089	0.00E+00	0.00E+00

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1306	FP	38-Sr-87m	86.9089	0.00E+00	6.84E-05
1307	FP	39-Y-87	86.9109	0.00E+00	2.41E-06
1308	FP	39-Y-87m	86.9109	0.00E+00	1.44E-05
1309	FP	40-Zr-87	86.9148	0.00E+00	1.15E-04
1310	FP	32-Ge-88	87.9569	0.00E+00	1.05E+01
1311	FP	33-As-88	87.9449	0.00E+00	6.19E+00
1312	FP	34-Se-88	87.9314	0.00E+00	4.53E-01
1313	FP	35-Br-88	87.9241	0.00E+00	4.26E-02
1314	FP	36-Kr-88	87.9145	0.00E+00	6.78E-05
1315	FP	37-Rb-88	87.9113	0.00E+00	6.50E-04
1316	FP	38-Sr-88	87.9056	0.00E+00	0.00E+00
1317	FP	39-Y-88	87.9095	0.00E+00	7.52E-08
1318	FP	40-Zr-88	87.9102	0.00E+00	9.62E-08
1319	FP	32-Ge-89	88.9638	0.00E+00	1.78E+01
1320	FP	33-As-89	88.9494	0.00E+00	1.17E+01
1321	FP	34-Se-89	88.9364	0.00E+00	1.69E+00
1322	FP	35-Br-89	88.9264	0.00E+00	1.58E-01
1323	FP	36-Kr-89	88.9176	0.00E+00	3.67E-03
1324	FP	37-Rb-89	88.9123	0.00E+00	7.63E-04
1325	FP	38-Sr-89	88.9074	0.00E+00	1.59E-07
1326	FP	39-Y-89	88.9059	0.00E+00	0.00E+00
1327	FP	39-Y-89m	88.9059	0.00E+00	4.43E-02
1328	FP	40-Zr-89	88.9089	0.00E+00	2.46E-06
1329	FP	40-Zr-89m	88.9089	0.00E+00	2.78E-03
1330	FP	41-Nb-89	88.9134	0.00E+00	9.48E-05
1331	FP	33-As-90	89.9555	0.00E+00	1.61E+01
1332	FP	34-Se-90	89.9400	0.00E+00	4.31E+00
1333	FP	35-Br-90	89.9306	0.00E+00	3.61E-01
1334	FP	36-Kr-90	89.9195	0.00E+00	2.14E-02
1335	FP	37-Rb-90	89.9148	0.00E+00	4.39E-03
1336	FP	37-Rb-90m	89.9148	0.00E+00	2.69E-03
1337	FP	38-Sr-90	89.9077	0.00E+00	7.63E-10
1338	FP	39-Y-90	89.9072	0.00E+00	3.01E-06
1339	FP	39-Y-90m	89.9072	0.00E+00	6.04E-05
1340	FP	40-Zr-90	89.9047	0.00E+00	0.00E+00
1341	FP	40-Zr-90m	89.9047	0.00E+00	8.57E-01
1342	FP	41-Nb-90	89.9113	0.00E+00	1.32E-05
1343	FP	41-Nb-90m	89.9113	0.00E+00	3.69E-02
1344	FP	42-Mo-90	89.9139	0.00E+00	3.40E-05
1345	FP	33-As-91	90.9604	0.00E+00	1.58E+01
1346	FP	34-Se-91	90.9460	0.00E+00	2.57E+00
1347	FP	35-Br-91	90.9340	0.00E+00	1.28E+00
1348	FP	36-Kr-91	90.9234	0.00E+00	8.09E-02

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1349	FP	37-Rb-91	90.9165	0.00E+00	1.19E-02
1350	FP	38-Sr-91	90.9102	0.00E+00	2.00E-05
1351	FP	39-Y-91	90.9073	0.00E+00	1.37E-07
1352	FP	39-Y-91m	90.9073	0.00E+00	2.32E-04
1353	FP	40-Zr-91	90.9056	0.00E+00	0.00E+00
1354	FP	41-Nb-91	90.9070	0.00E+00	3.23E-11
1355	FP	41-Nb-91m	90.9070	0.00E+00	1.32E-07
1356	FP	42-Mo-91	90.9118	0.00E+00	7.46E-04
1357	FP	33-As-92	91.9668	0.00E+00	2.57E+01
1358	FP	34-Se-92	91.9499	0.00E+00	7.45E+00
1359	FP	35-Br-92	91.9393	0.00E+00	2.02E+00
1360	FP	36-Kr-92	91.9262	0.00E+00	3.77E-01
1361	FP	37-Rb-92	91.9197	0.00E+00	1.54E-01
1362	FP	38-Sr-92	91.9110	0.00E+00	7.10E-05
1363	FP	39-Y-92	91.9090	0.00E+00	5.44E-05
1364	FP	40-Zr-92	91.9050	0.00E+00	0.00E+00
1365	FP	41-Nb-92	91.9072	0.00E+00	6.33E-16
1366	FP	41-Nb-92m	91.9072	0.00E+00	7.90E-07
1367	FP	42-Mo-92	91.9068	0.00E+00	0.00E+00
1368	FP	34-Se-93	92.9563	0.00E+00	1.12E+01
1369	FP	35-Br-93	92.9430	0.00E+00	6.80E+00
1370	FP	36-Kr-93	92.9313	0.00E+00	5.39E-01
1371	FP	37-Rb-93	92.9220	0.00E+00	1.19E-01
1372	FP	38-Sr-93	92.9140	0.00E+00	1.56E-03
1373	FP	39-Y-93	92.9096	0.00E+00	1.89E-05
1374	FP	39-Y-93m	92.9096	0.00E+00	8.45E-01
1375	FP	40-Zr-93	92.9065	0.00E+00	1.44E-14
1376	FP	41-Nb-93	92.9064	0.00E+00	0.00E+00
1377	FP	41-Nb-93m	92.9064	0.00E+00	1.36E-09
1378	FP	42-Mo-93	92.9068	0.00E+00	5.49E-12
1379	FP	42-Mo-93m	92.9068	0.00E+00	2.81E-05
1380	FP	43-Tc-93	92.9102	0.00E+00	7.00E-05
1381	FP	34-Se-94	93.9605	0.00E+00	1.17E+01
1382	FP	35-Br-94	93.9487	0.00E+00	9.90E+00
1383	FP	36-Kr-94	93.9344	0.00E+00	3.27E+00
1384	FP	37-Rb-94	93.9264	0.00E+00	2.57E-01
1385	FP	38-Sr-94	93.9154	0.00E+00	9.20E-03
1386	FP	39-Y-94	93.9116	0.00E+00	6.18E-04
1387	FP	40-Zr-94	93.9063	0.00E+00	0.00E+00
1388	FP	41-Nb-94	93.9073	0.00E+00	1.08E-12
1389	FP	41-Nb-94m	93.9073	0.00E+00	1.84E-03
1390	FP	42-Mo-94	93.9051	0.00E+00	0.00E+00
1391	FP	35-Br-95	94.9529	0.00E+00	1.05E+01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1392	FP	36-Kr-95	94.9398	0.00E+00	6.08E+00
1393	FP	37-Rb-95	94.9293	0.00E+00	1.84E+00
1394	FP	38-Sr-95	94.9194	0.00E+00	2.90E-02
1395	FP	39-Y-95	94.9128	0.00E+00	1.12E-03
1396	FP	40-Zr-95	94.9080	0.00E+00	1.25E-07
1397	FP	41-Nb-95	94.9068	0.00E+00	2.29E-07
1398	FP	41-Nb-95m	94.9068	0.00E+00	2.22E-06
1399	FP	42-Mo-95	94.9058	0.00E+00	0.00E+00
1400	FP	43-Tc-95	94.9077	0.00E+00	9.63E-06
1401	FP	43-Tc-95m	94.9077	0.00E+00	1.32E-07
1402	FP	44-Ru-95	94.9104	0.00E+00	1.17E-04
1403	FP	35-Br-96	95.9585	0.00E+00	1.65E+01
1404	FP	36-Kr-96	95.9431	0.00E+00	8.66E+00
1405	FP	37-Rb-96	95.9343	0.00E+00	3.41E+00
1406	FP	38-Sr-96	95.9217	0.00E+00	6.48E-01
1407	FP	39-Y-96	95.9159	0.00E+00	1.30E-01
1408	FP	39-Y-96m	95.9159	0.00E+00	7.22E-02
1409	FP	40-Zr-96	95.9083	0.00E+00	1.10E-27
1410	FP	41-Nb-96	95.9081	0.00E+00	8.25E-06
1411	FP	42-Mo-96	95.9047	0.00E+00	0.00E+00
1412	FP	43-Tc-96	95.9079	0.00E+00	1.87E-06
1413	FP	44-Ru-96	95.9076	0.00E+00	0.00E+00
1414	FP	35-Br-97	96.9628	0.00E+00	1.73E+01
1415	FP	36-Kr-97	96.9486	0.00E+00	1.10E+01
1416	FP	37-Rb-97	96.9373	0.00E+00	4.10E+00
1417	FP	38-Sr-97	96.9261	0.00E+00	1.62E+00
1418	FP	39-Y-97	96.9181	0.00E+00	1.85E-01
1419	FP	39-Y-97m	96.9181	0.00E+00	5.92E-01
1420	FP	40-Zr-97	96.9109	0.00E+00	1.15E-05
1421	FP	41-Nb-97	96.9081	0.00E+00	1.60E-04
1422	FP	41-Nb-97m	96.9081	0.00E+00	1.18E-02
1423	FP	42-Mo-97	96.9060	0.00E+00	0.00E+00
1424	FP	43-Tc-97	96.9064	0.00E+00	5.22E-15
1425	FP	43-Tc-97m	96.9064	0.00E+00	8.82E-08
1426	FP	44-Ru-97	96.9076	0.00E+00	2.83E-06
1427	FP	36-Kr-98	97.9519	0.00E+00	1.51E+01
1428	FP	37-Rb-98	97.9418	0.00E+00	6.08E+00
1429	FP	38-Sr-98	97.9285	0.00E+00	1.06E+00
1430	FP	39-Y-98	97.9222	0.00E+00	1.26E+00
1431	FP	39-Y-98m	97.9222	0.00E+00	3.47E-01
1432	FP	40-Zr-98	97.9127	0.00E+00	2.26E-02
1433	FP	41-Nb-98	97.9103	0.00E+00	2.42E-01
1434	FP	41-Nb-98m	97.9103	0.00E+00	2.25E-04

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1435	FP	42-Mo-98	97.9054	0.00E+00	0.00E+00
1436	FP	43-Tc-98	97.9072	0.00E+00	5.23E-15
1437	FP	44-Ru-98	97.9053	0.00E+00	0.00E+00
1438	FP	36-Kr-99	98.9576	0.00E+00	2.57E+01
1439	FP	37-Rb-99	98.9454	0.00E+00	1.28E+01
1440	FP	38-Sr-99	98.9332	0.00E+00	2.57E+00
1441	FP	39-Y-99	98.9246	0.00E+00	4.72E-01
1442	FP	40-Zr-99	98.9165	0.00E+00	3.30E-01
1443	FP	41-Nb-99	98.9116	0.00E+00	4.62E-02
1444	FP	41-Nb-99m	98.9116	0.00E+00	4.62E-03
1445	FP	42-Mo-99	98.9077	0.00E+00	2.92E-06
1446	FP	43-Tc-99	98.9062	0.00E+00	1.04E-13
1447	FP	43-Tc-99m	98.9062	0.00E+00	3.21E-05
1448	FP	44-Ru-99	98.9059	0.00E+00	0.00E+00
1449	FP	45-Rh-99	98.9081	0.00E+00	4.98E-07
1450	FP	45-Rh-99m	98.9081	0.00E+00	4.10E-05
1451	FP	46-Pd-99	98.9118	0.00E+00	5.40E-04
1452	FP	36-Kr-100	99.9611	0.00E+00	9.90E+01
1453	FP	37-Rb-100	99.9499	0.00E+00	1.36E+01
1454	FP	38-Sr-100	99.9353	0.00E+00	3.43E+00
1455	FP	39-Y-100	99.9278	0.00E+00	9.43E-01
1456	FP	40-Zr-100	99.9178	0.00E+00	9.76E-02
1457	FP	41-Nb-100	99.9142	0.00E+00	4.62E-01
1458	FP	41-Nb-100m	99.9142	0.00E+00	2.32E-01
1459	FP	42-Mo-100	99.9075	0.00E+00	3.01E-27
1460	FP	43-Tc-100	99.9077	0.00E+00	4.48E-02
1461	FP	44-Ru-100	99.9042	0.00E+00	0.00E+00
1462	FP	37-Rb-101	100.9532	0.00E+00	2.17E+01
1463	FP	38-Sr-101	100.9405	0.00E+00	5.87E+00
1464	FP	39-Y-101	100.9303	0.00E+00	1.54E+00
1465	FP	40-Zr-101	100.9211	0.00E+00	3.01E-01
1466	FP	41-Nb-101	100.9153	0.00E+00	9.76E-02
1467	FP	42-Mo-101	100.9103	0.00E+00	7.91E-04
1468	FP	43-Tc-101	100.9073	0.00E+00	8.14E-04
1469	FP	44-Ru-101	100.9056	0.00E+00	0.00E+00
1470	FP	45-Rh-101	100.9062	0.00E+00	6.66E-09
1471	FP	45-Rh-101m	100.9062	0.00E+00	1.85E-06
1472	FP	46-Pd-101	100.9083	0.00E+00	2.27E-05
1473	FP	37-Rb-102	101.9589	0.00E+00	1.87E+01
1474	FP	38-Sr-102	101.9430	0.00E+00	1.00E+01
1475	FP	39-Y-102	101.9336	0.00E+00	1.93E+00
1476	FP	40-Zr-102	101.9230	0.00E+00	2.39E-01
1477	FP	41-Nb-102	101.9180	0.00E+00	1.61E-01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1478	FP	41-Nb-102m	101.9180	0.00E+00	5.33E-01
1479	FP	42-Mo-102	101.9103	0.00E+00	1.02E-03
1480	FP	43-Tc-102	101.9092	0.00E+00	1.31E-01
1481	FP	43-Tc-102m	101.9092	0.00E+00	2.66E-03
1482	FP	44-Ru-102	101.9044	0.00E+00	0.00E+00
1483	FP	45-Rh-102	101.9068	0.00E+00	3.87E-08
1484	FP	45-Rh-102m	101.9068	0.00E+00	5.87E-09
1485	FP	46-Pd-102	101.9056	0.00E+00	0.00E+00
1486	FP	38-Sr-103	102.9490	0.00E+00	1.02E+01
1487	FP	39-Y-103	102.9367	0.00E+00	3.01E+00
1488	FP	40-Zr-103	102.9266	0.00E+00	5.33E-01
1489	FP	41-Nb-103	102.9191	0.00E+00	4.62E-01
1490	FP	42-Mo-103	102.9132	0.00E+00	1.03E-02
1491	FP	43-Tc-103	102.9092	0.00E+00	1.28E-02
1492	FP	44-Ru-103	102.9063	0.00E+00	2.04E-07
1493	FP	45-Rh-103	102.9055	0.00E+00	0.00E+00
1494	FP	45-Rh-103m	102.9055	0.00E+00	2.06E-04
1495	FP	46-Pd-103	102.9061	0.00E+00	4.72E-07
1496	FP	47-Ag-103	102.9090	0.00E+00	1.76E-04
1497	FP	38-Sr-104	103.9523	0.00E+00	1.61E+01
1498	FP	39-Y-104	103.9410	0.00E+00	3.85E+00
1499	FP	40-Zr-104	103.9288	0.00E+00	5.78E-01
1500	FP	41-Nb-104	103.9225	0.00E+00	1.41E-01
1501	FP	41-Nb-104m	103.9225	0.00E+00	7.37E-01
1502	FP	42-Mo-104	103.9138	0.00E+00	1.16E-02
1503	FP	43-Tc-104	103.9115	0.00E+00	6.31E-04
1504	FP	44-Ru-104	103.9054	0.00E+00	0.00E+00
1505	FP	45-Rh-104	103.9067	0.00E+00	1.64E-02
1506	FP	45-Rh-104m	103.9067	0.00E+00	2.66E-03
1507	FP	46-Pd-104	103.9040	0.00E+00	0.00E+00
1508	FP	38-Sr-105	104.9586	0.00E+00	1.25E+01
1509	FP	39-Y-105	104.9449	0.00E+00	7.88E+00
1510	FP	40-Zr-105	104.9331	0.00E+00	1.16E+00
1511	FP	41-Nb-105	104.9239	0.00E+00	2.35E-01
1512	FP	42-Mo-105	104.9170	0.00E+00	1.95E-02
1513	FP	43-Tc-105	104.9117	0.00E+00	1.52E-03
1514	FP	44-Ru-105	104.9078	0.00E+00	4.34E-05
1515	FP	45-Rh-105	104.9057	0.00E+00	5.45E-06
1516	FP	45-Rh-105m	104.9057	0.00E+00	1.73E-02
1517	FP	46-Pd-105	104.9051	0.00E+00	0.00E+00
1518	FP	47-Ag-105	104.9065	0.00E+00	1.94E-07
1519	FP	47-Ag-105m	104.9065	0.00E+00	1.60E-03
1520	FP	48-Cd-105	104.9095	0.00E+00	2.08E-04

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1521	FP	39-Y-106	105.9498	0.00E+00	1.05E+01
1522	FP	40-Zr-106	105.9359	0.00E+00	2.57E+00
1523	FP	41-Nb-106	105.9280	0.00E+00	7.45E-01
1524	FP	42-Mo-106	105.9181	0.00E+00	7.94E-02
1525	FP	43-Tc-106	105.9144	0.00E+00	1.95E-02
1526	FP	44-Ru-106	105.9073	0.00E+00	2.16E-08
1527	FP	45-Rh-106	105.9073	0.00E+00	2.31E-02
1528	FP	45-Rh-106m	105.9073	0.00E+00	8.82E-05
1529	FP	46-Pd-106	105.9035	0.00E+00	0.00E+00
1530	FP	47-Ag-106	105.9067	0.00E+00	4.82E-04
1531	FP	47-Ag-106m	105.9067	0.00E+00	9.69E-07
1532	FP	48-Cd-106	105.9065	0.00E+00	0.00E+00
1533	FP	39-Y-107	106.9541	0.00E+00	2.31E+01
1534	FP	40-Zr-107	106.9408	0.00E+00	4.62E+00
1535	FP	41-Nb-107	106.9303	0.00E+00	2.31E+00
1536	FP	42-Mo-107	106.9217	0.00E+00	1.98E-01
1537	FP	43-Tc-107	106.9151	0.00E+00	3.27E-02
1538	FP	44-Ru-107	106.9099	0.00E+00	3.08E-03
1539	FP	45-Rh-107	106.9068	0.00E+00	5.32E-04
1540	FP	46-Pd-107	106.9051	0.00E+00	3.38E-15
1541	FP	46-Pd-107m	106.9051	0.00E+00	3.25E-02
1542	FP	47-Ag-107	106.9051	0.00E+00	0.00E+00
1543	FP	47-Ag-107m	106.9051	0.00E+00	1.56E-02
1544	FP	48-Cd-107	106.9066	0.00E+00	2.96E-05
1545	FP	49-In-107	106.9103	0.00E+00	3.57E-04
1546	FP	39-Y-108	107.9595	0.00E+00	1.44E+01
1547	FP	40-Zr-108	107.9440	0.00E+00	8.66E+00
1548	FP	41-Nb-108	107.9348	0.00E+00	3.59E+00
1549	FP	42-Mo-108	107.9234	0.00E+00	6.36E-01
1550	FP	43-Tc-108	107.9185	0.00E+00	1.34E-01
1551	FP	44-Ru-108	107.9102	0.00E+00	2.54E-03
1552	FP	45-Rh-108	107.9087	0.00E+00	4.13E-02
1553	FP	45-Rh-108m	107.9087	0.00E+00	1.93E-03
1554	FP	46-Pd-108	107.9039	0.00E+00	0.00E+00
1555	FP	47-Ag-108	107.9060	0.00E+00	4.85E-03
1556	FP	47-Ag-108m	107.9060	0.00E+00	5.01E-11
1557	FP	48-Cd-108	107.9042	0.00E+00	0.00E+00
1558	FP	40-Zr-109	108.9492	0.00E+00	5.92E+00
1559	FP	41-Nb-109	108.9376	0.00E+00	3.65E+00
1560	FP	42-Mo-109	108.9278	0.00E+00	1.31E+00
1561	FP	43-Tc-109	108.9200	0.00E+00	8.06E-01
1562	FP	44-Ru-109	108.9132	0.00E+00	2.01E-02
1563	FP	45-Rh-109	108.9087	0.00E+00	8.66E-03

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1564	FP	46-Pd-109	108.9060	0.00E+00	1.41E-05
1565	FP	46-Pd-109m	108.9060	0.00E+00	2.46E-03
1566	FP	47-Ag-109	108.9047	0.00E+00	0.00E+00
1567	FP	47-Ag-109m	108.9047	0.00E+00	1.75E-02
1568	FP	48-Cd-109	108.9050	0.00E+00	1.74E-08
1569	FP	49-In-109	108.9072	0.00E+00	4.62E-05
1570	FP	40-Zr-110	109.9529	0.00E+00	7.07E+00
1571	FP	41-Nb-110	109.9424	0.00E+00	4.08E+00
1572	FP	42-Mo-110	109.9297	0.00E+00	2.31E+00
1573	FP	43-Tc-110	109.9238	0.00E+00	7.53E-01
1574	FP	44-Ru-110	109.9141	0.00E+00	5.98E-02
1575	FP	45-Rh-110	109.9111	0.00E+00	2.17E-01
1576	FP	45-Rh-110m	109.9111	0.00E+00	2.43E-02
1577	FP	46-Pd-110	109.9052	0.00E+00	0.00E+00
1578	FP	47-Ag-110	109.9061	0.00E+00	2.82E-02
1579	FP	47-Ag-110m	109.9062	0.00E+00	3.21E-08
1580	FP	48-Cd-110	109.9030	0.00E+00	0.00E+00
1581	FP	41-Nb-111	110.9456	0.00E+00	8.66E+00
1582	FP	42-Mo-111	110.9344	0.00E+00	3.47E+00
1583	FP	43-Tc-111	110.9257	0.00E+00	2.39E+00
1584	FP	44-Ru-111	110.9177	0.00E+00	3.27E-01
1585	FP	45-Rh-111	110.9116	0.00E+00	6.30E-02
1586	FP	46-Pd-111	110.9077	0.00E+00	4.94E-04
1587	FP	46-Pd-111m	110.9077	0.00E+00	3.50E-05
1588	FP	47-Ag-111	110.9053	0.00E+00	1.08E-06
1589	FP	47-Ag-111m	110.9053	0.00E+00	1.07E-02
1590	FP	48-Cd-111	110.9042	0.00E+00	0.00E+00
1591	FP	48-Cd-111m	110.9042	0.00E+00	2.38E-04
1592	FP	49-In-111	110.9051	0.00E+00	2.86E-06
1593	FP	49-In-111m	110.9051	0.00E+00	1.50E-03
1594	FP	50-Sn-111	110.9077	0.00E+00	3.27E-04
1595	FP	41-Nb-112	111.9508	0.00E+00	1.00E+01
1596	FP	42-Mo-112	111.9368	0.00E+00	2.42E+00
1597	FP	43-Tc-112	111.9292	0.00E+00	2.48E+00
1598	FP	44-Ru-112	111.9190	0.00E+00	3.96E-01
1599	FP	45-Rh-112	111.9144	0.00E+00	3.30E-01
1600	FP	46-Pd-112	111.9073	0.00E+00	9.16E-06
1601	FP	47-Ag-112	111.9070	0.00E+00	6.15E-05
1602	FP	48-Cd-112	111.9028	0.00E+00	0.00E+00
1603	FP	49-In-112	111.9055	0.00E+00	7.72E-04
1604	FP	49-In-112m	111.9055	0.00E+00	5.62E-04
1605	FP	50-Sn-112	111.9048	0.00E+00	0.00E+00
1606	FP	41-Nb-113	112.9547	0.00E+00	2.31E+01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1607	FP	42-Mo-113	112.9419	0.00E+00	6.93E+00
1608	FP	43-Tc-113	112.9316	0.00E+00	4.33E+00
1609	FP	44-Ru-113	112.9225	0.00E+00	8.66E-01
1610	FP	45-Rh-113	112.9155	0.00E+00	2.48E-01
1611	FP	46-Pd-113	112.9101	0.00E+00	7.45E-03
1612	FP	47-Ag-113	112.9066	0.00E+00	3.59E-05
1613	FP	47-Ag-113m	112.9066	0.00E+00	1.01E-02
1614	FP	48-Cd-113	112.9044	0.00E+00	2.73E-24
1615	FP	48-Cd-113m	112.9044	0.00E+00	1.56E-09
1616	FP	49-In-113	112.9041	0.00E+00	0.00E+00
1617	FP	49-In-113m	112.9041	0.00E+00	1.16E-04
1618	FP	50-Sn-113	112.9052	0.00E+00	6.97E-08
1619	FP	50-Sn-113m	112.9052	0.00E+00	5.40E-04
1620	FP	51-Sb-113	112.9094	0.00E+00	1.73E-03
1621	FP	42-Mo-114	113.9449	0.00E+00	8.66E+00
1622	FP	43-Tc-114	113.9359	0.00E+00	4.62E+00
1623	FP	44-Ru-114	113.9243	0.00E+00	1.33E+00
1624	FP	45-Rh-114	113.9188	0.00E+00	3.75E-01
1625	FP	46-Pd-114	113.9104	0.00E+00	4.77E-03
1626	FP	47-Ag-114	113.9088	0.00E+00	1.51E-01
1627	FP	48-Cd-114	113.9034	0.00E+00	0.00E+00
1628	FP	49-In-114	113.9049	0.00E+00	9.64E-03
1629	FP	49-In-114m	113.9049	0.00E+00	1.62E-07
1630	FP	50-Sn-114	113.9028	0.00E+00	0.00E+00
1631	FP	42-Mo-115	114.9503	0.00E+00	7.53E+00
1632	FP	43-Tc-115	114.9387	0.00E+00	9.50E+00
1633	FP	44-Ru-115	114.9287	0.00E+00	9.37E-01
1634	FP	45-Rh-115	114.9203	0.00E+00	7.00E-01
1635	FP	46-Pd-115	114.9137	0.00E+00	2.77E-02
1636	FP	47-Ag-115	114.9088	0.00E+00	5.78E-04
1637	FP	47-Ag-115m	114.9088	0.00E+00	3.85E-02
1638	FP	48-Cd-115	114.9054	0.00E+00	3.60E-06
1639	FP	48-Cd-115m	114.9051	0.00E+00	1.80E-07
1640	FP	49-In-115	114.9039	0.00E+00	4.98E-23
1641	FP	49-In-115m	114.9039	0.00E+00	4.29E-05
1642	FP	50-Sn-115	114.9033	0.00E+00	0.00E+00
1643	FP	51-Sb-115	114.9066	0.00E+00	3.60E-04
1644	FP	52-Te-115	114.9119	0.00E+00	1.99E-03
1645	FP	43-Tc-116	115.9434	0.00E+00	7.70E+00
1646	FP	44-Ru-116	115.9308	0.00E+00	3.40E+00
1647	FP	45-Rh-116	115.9241	0.00E+00	1.02E+00
1648	FP	46-Pd-116	115.9142	0.00E+00	5.87E-02
1649	FP	47-Ag-116	115.9114	0.00E+00	2.92E-03

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1650	FP	47-Ag-116m	115.9114	0.00E+00	3.47E-02
1651	FP	48-Cd-116	115.9048	0.00E+00	7.09E-28
1652	FP	49-In-116	115.9053	0.00E+00	4.92E-02
1653	FP	49-In-116m	115.9053	0.00E+00	2.13E-04
1654	FP	50-Sn-116	115.9017	0.00E+00	0.00E+00
1655	FP	43-Tc-117	116.9465	0.00E+00	1.73E+01
1656	FP	44-Ru-117	116.9356	0.00E+00	4.88E+00
1657	FP	45-Rh-117	116.9260	0.00E+00	1.58E+00
1658	FP	46-Pd-117	116.9178	0.00E+00	1.61E-01
1659	FP	47-Ag-117	116.9117	0.00E+00	9.52E-03
1660	FP	47-Ag-117m	116.9117	0.00E+00	1.30E-01
1661	FP	48-Cd-117	116.9072	0.00E+00	7.73E-05
1662	FP	48-Cd-117m	116.9072	0.00E+00	5.73E-05
1663	FP	49-In-117	116.9045	0.00E+00	2.67E-04
1664	FP	49-In-117m	116.9045	0.00E+00	9.94E-05
1665	FP	50-Sn-117	116.9029	0.00E+00	0.00E+00
1666	FP	50-Sn-117m	116.9029	0.00E+00	5.90E-07
1667	FP	51-Sb-117	116.9048	0.00E+00	6.88E-05
1668	FP	52-Te-117	116.9087	0.00E+00	1.86E-04
1669	FP	43-Tc-118	117.9515	0.00E+00	1.05E+01
1670	FP	44-Ru-118	117.9378	0.00E+00	5.64E+00
1671	FP	45-Rh-118	117.9301	0.00E+00	2.61E+00
1672	FP	46-Pd-118	117.9190	0.00E+00	3.65E-01
1673	FP	47-Ag-118	117.9146	0.00E+00	1.84E-01
1674	FP	47-Ag-118m	117.9146	0.00E+00	3.47E-01
1675	FP	48-Cd-118	117.9069	0.00E+00	2.30E-04
1676	FP	49-In-118	117.9063	0.00E+00	1.39E-01
1677	FP	49-In-118m	117.9063	0.00E+00	2.60E-03
1678	FP	50-Sn-118	117.9016	0.00E+00	0.00E+00
1679	FP	51-Sb-118	117.9055	0.00E+00	3.21E-03
1680	FP	51-Sb-118m	117.9055	0.00E+00	3.85E-05
1681	FP	52-Te-118	117.9058	0.00E+00	1.34E-06
1682	FP	44-Ru-119	118.9428	0.00E+00	4.28E+00
1683	FP	45-Rh-119	118.9321	0.00E+00	4.05E+00
1684	FP	46-Pd-119	118.9231	0.00E+00	7.53E-01
1685	FP	47-Ag-119	118.9157	0.00E+00	3.30E-01
1686	FP	48-Cd-119	118.9099	0.00E+00	4.29E-03
1687	FP	48-Cd-119m	118.9099	0.00E+00	5.25E-03
1688	FP	49-In-119	118.9059	0.00E+00	4.81E-03
1689	FP	49-In-119m	118.9059	0.00E+00	6.42E-04
1690	FP	50-Sn-119	118.9033	0.00E+00	0.00E+00
1691	FP	50-Sn-119m	118.9033	0.00E+00	2.74E-08
1692	FP	51-Sb-119	118.9039	0.00E+00	5.04E-06

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1693	FP	52-Te-119	118.9064	0.00E+00	1.20E-05
1694	FP	44-Ru-120	119.9453	0.00E+00	4.65E+00
1695	FP	45-Rh-120	119.9364	0.00E+00	5.10E+00
1696	FP	46-Pd-120	119.9247	0.00E+00	1.39E+00
1697	FP	47-Ag-120	119.9188	0.00E+00	5.64E-01
1698	FP	47-Ag-120m	119.9188	0.00E+00	2.17E+00
1699	FP	48-Cd-120	119.9099	0.00E+00	1.36E-02
1700	FP	49-In-120	119.9080	0.00E+00	2.25E-01
1701	FP	49-In-120m	119.9080	0.00E+00	1.50E-02
1702	FP	50-Sn-120	119.9022	0.00E+00	0.00E+00
1703	FP	51-Sb-120	119.9051	0.00E+00	7.27E-04
1704	FP	51-Sb-120m	119.9051	0.00E+00	1.39E-06
1705	FP	52-Te-120	119.9040	0.00E+00	0.00E+00
1706	FP	45-Rh-121	120.9387	0.00E+00	4.59E+00
1707	FP	46-Pd-121	120.9289	0.00E+00	2.43E+00
1708	FP	47-Ag-121	120.9199	0.00E+00	8.89E-01
1709	FP	48-Cd-121	120.9130	0.00E+00	5.13E-02
1710	FP	48-Cd-121m	120.9130	0.00E+00	8.35E-02
1711	FP	49-In-121	120.9079	0.00E+00	3.00E-02
1712	FP	49-In-121m	120.9079	0.00E+00	2.98E-03
1713	FP	50-Sn-121	120.9042	0.00E+00	7.12E-06
1714	FP	50-Sn-121m	120.9042	0.00E+00	5.00E-10
1715	FP	51-Sb-121	120.9038	0.00E+00	0.00E+00
1716	FP	52-Te-121	120.9049	0.00E+00	4.19E-07
1717	FP	52-Te-121m	120.9049	0.00E+00	4.89E-08
1718	FP	53-I-121	120.9074	0.00E+00	9.08E-05
1719	FP	45-Rh-122	121.9432	0.00E+00	6.42E+00
1720	FP	46-Pd-122	121.9305	0.00E+00	3.96E+00
1721	FP	47-Ag-122	121.9235	0.00E+00	1.31E+00
1722	FP	47-Ag-122m	121.9235	0.00E+00	3.47E+00
1723	FP	48-Cd-122	121.9133	0.00E+00	1.32E-01
1724	FP	49-In-122	121.9103	0.00E+00	4.62E-01
1725	FP	49-In-122m	121.9103	0.00E+00	6.73E-02
1726	FP	50-Sn-122	121.9034	0.00E+00	0.00E+00
1727	FP	51-Sb-122	121.9052	0.00E+00	2.95E-06
1728	FP	51-Sb-122m	121.9052	0.00E+00	2.76E-03
1729	FP	52-Te-122	121.9030	0.00E+00	0.00E+00
1730	FP	46-Pd-123	122.9349	0.00E+00	2.84E+00
1731	FP	47-Ag-123	122.9249	0.00E+00	2.31E+00
1732	FP	48-Cd-123	122.9170	0.00E+00	3.30E-01
1733	FP	48-Cd-123m	122.9170	0.00E+00	3.81E-01
1734	FP	49-In-123	122.9104	0.00E+00	1.12E-01
1735	FP	49-In-123m	122.9104	0.00E+00	1.46E-02

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1736	FP	50-Sn-123	122.9057	0.00E+00	6.21E-08
1737	FP	50-Sn-123m	122.9057	0.00E+00	2.88E-04
1738	FP	51-Sb-123	122.9042	0.00E+00	0.00E+00
1739	FP	52-Te-123	122.9043	0.00E+00	0.00E+00
1740	FP	52-Te-123m	122.9043	0.00E+00	6.73E-08
1741	FP	53-I-123	122.9056	0.00E+00	1.46E-05
1742	FP	46-Pd-124	123.9369	0.00E+00	1.82E+01
1743	FP	47-Ag-124	123.9286	0.00E+00	4.03E+00
1744	FP	48-Cd-124	123.9176	0.00E+00	5.55E-01
1745	FP	49-In-124	123.9132	0.00E+00	2.22E-01
1746	FP	49-In-124m	123.9132	0.00E+00	1.87E-01
1747	FP	50-Sn-124	123.9053	0.00E+00	0.00E+00
1748	FP	51-Sb-124	123.9059	0.00E+00	1.33E-07
1749	FP	51-Sb-124m	123.9059	0.00E+00	7.45E-03
1750	FP	52-Te-124	123.9028	0.00E+00	0.00E+00
1751	FP	53-I-124	123.9062	0.00E+00	1.92E-06
1752	FP	54-Xe-124	123.9059	0.00E+00	0.00E+00
1753	FP	47-Ag-125	124.9304	0.00E+00	4.18E+00
1754	FP	48-Cd-125	124.9212	0.00E+00	1.02E+00
1755	FP	49-In-125	124.9136	0.00E+00	2.94E-01
1756	FP	49-In-125m	124.9136	0.00E+00	5.68E-02
1757	FP	50-Sn-125	124.9078	0.00E+00	8.32E-07
1758	FP	50-Sn-125m	124.9078	0.00E+00	1.21E-03
1759	FP	51-Sb-125	124.9053	0.00E+00	7.96E-09
1760	FP	52-Te-125	124.9044	0.00E+00	0.00E+00
1761	FP	52-Te-125m	124.9044	0.00E+00	1.40E-07
1762	FP	53-I-125	124.9046	0.00E+00	1.35E-07
1763	FP	54-Xe-125	124.9064	0.00E+00	1.14E-05
1764	FP	54-Xe-125m	124.9064	0.00E+00	1.22E-02
1765	FP	47-Ag-126	125.9345	0.00E+00	6.48E+00
1766	FP	48-Cd-126	125.9223	0.00E+00	1.35E+00
1767	FP	49-In-126	125.9165	0.00E+00	4.53E-01
1768	FP	49-In-126m	125.9165	0.00E+00	4.23E-01
1769	FP	50-Sn-126	125.9077	0.00E+00	9.55E-14
1770	FP	51-Sb-126	125.9072	0.00E+00	6.50E-07
1771	FP	51-Sb-126m	125.9072	0.00E+00	6.03E-04
1772	FP	52-Te-126	125.9033	0.00E+00	0.00E+00
1773	FP	53-I-126	125.9056	0.00E+00	6.20E-07
1774	FP	54-Xe-126	125.9043	0.00E+00	0.00E+00
1775	FP	47-Ag-127	126.9368	0.00E+00	6.36E+00
1776	FP	48-Cd-127	126.9264	0.00E+00	1.87E+00
1777	FP	49-In-127	126.9174	0.00E+00	6.36E-01
1778	FP	49-In-127m	126.9174	0.00E+00	1.89E-01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1779	FP	50-Sn-127	126.9104	0.00E+00	9.17E-05
1780	FP	50-Sn-127m	126.9104	0.00E+00	2.80E-03
1781	FP	51-Sb-127	126.9069	0.00E+00	2.08E-06
1782	FP	52-Te-127	126.9052	0.00E+00	2.06E-05
1783	FP	52-Te-127m	126.9052	0.00E+00	7.36E-08
1784	FP	53-I-127	126.9045	0.00E+00	0.00E+00
1785	FP	54-Xe-127	126.9052	0.00E+00	2.20E-07
1786	FP	54-Xe-127m	126.9052	0.00E+00	1.00E-02
1787	FP	55-Cs-127	126.9074	0.00E+00	3.08E-05
1788	FP	47-Ag-128	127.9412	0.00E+00	1.20E+01
1789	FP	48-Cd-128	127.9278	0.00E+00	2.48E+00
1790	FP	49-In-128	127.9202	0.00E+00	8.25E-01
1791	FP	49-In-128m	127.9202	0.00E+00	9.63E-01
1792	FP	50-Sn-128	127.9105	0.00E+00	1.96E-04
1793	FP	50-Sn-128m	127.9105	0.00E+00	1.07E-01
1794	FP	51-Sb-128	127.9092	0.00E+00	2.14E-05
1795	FP	51-Sb-128m	127.9092	0.00E+00	1.11E-03
1796	FP	52-Te-128	127.9045	0.00E+00	2.50E-27
1797	FP	53-I-128	127.9058	0.00E+00	4.62E-04
1798	FP	54-Xe-128	127.9035	0.00E+00	0.00E+00
1799	FP	47-Ag-129	128.9437	0.00E+00	1.51E+01
1800	FP	48-Cd-129	128.9321	0.00E+00	2.57E+00
1801	FP	49-In-129	128.9217	0.00E+00	1.14E+00
1802	FP	49-In-129m	128.9217	0.00E+00	5.64E-01
1803	FP	50-Sn-129	128.9135	0.00E+00	5.18E-03
1804	FP	50-Sn-129m	128.9135	0.00E+00	1.67E-03
1805	FP	51-Sb-129	128.9091	0.00E+00	4.38E-05
1806	FP	51-Sb-129m	128.9091	0.00E+00	6.53E-04
1807	FP	52-Te-129	128.9066	0.00E+00	1.66E-04
1808	FP	52-Te-129m	128.9074	0.00E+00	2.39E-07
1809	FP	53-I-129	128.9050	0.00E+00	1.40E-15
1810	FP	54-Xe-129	128.9048	0.00E+00	0.00E+00
1811	FP	54-Xe-129m	128.9048	0.00E+00	9.03E-07
1812	FP	55-Cs-129	128.9061	0.00E+00	6.01E-06
1813	FP	56-Ba-129	128.9087	0.00E+00	8.63E-05
1814	FP	47-Ag-130	129.9505	0.00E+00	1.39E+01
1815	FP	48-Cd-130	129.9339	0.00E+00	4.28E+00
1816	FP	49-In-130	129.9250	0.00E+00	2.39E+00
1817	FP	49-In-130m	129.9250	0.00E+00	1.28E+00
1818	FP	50-Sn-130	129.9140	0.00E+00	3.11E-03
1819	FP	50-Sn-130m	129.9140	0.00E+00	6.80E-03
1820	FP	51-Sb-130	129.9117	0.00E+00	2.92E-04
1821	FP	51-Sb-130m	129.9117	0.00E+00	1.83E-03

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1822	FP	52-Te-130	129.9062	0.00E+00	0.00E+00
1823	FP	53-I-130	129.9067	0.00E+00	1.56E-05
1824	FP	53-I-130m	129.9067	0.00E+00	1.31E-03
1825	FP	54-Xe-130	129.9035	0.00E+00	0.00E+00
1826	FP	48-Cd-131	130.9407	0.00E+00	1.02E+01
1827	FP	49-In-131	130.9268	0.00E+00	2.48E+00
1828	FP	49-In-131m	130.9268	0.00E+00	1.98E+00
1829	FP	50-Sn-131	130.9170	0.00E+00	1.24E-02
1830	FP	50-Sn-131m	130.9170	0.00E+00	1.19E-02
1831	FP	51-Sb-131	130.9120	0.00E+00	5.02E-04
1832	FP	52-Te-131	130.9085	0.00E+00	4.62E-04
1833	FP	52-Te-131m	130.9085	0.00E+00	5.79E-06
1834	FP	53-I-131	130.9061	0.00E+00	1.00E-06
1835	FP	54-Xe-131	130.9051	0.00E+00	0.00E+00
1836	FP	54-Xe-131m	130.9051	0.00E+00	6.78E-07
1837	FP	55-Cs-131	130.9055	0.00E+00	8.28E-07
1838	FP	56-Ba-131	130.9069	0.00E+00	6.98E-07
1839	FP	48-Cd-132	131.9456	0.00E+00	7.15E+00
1840	FP	49-In-132	131.9330	0.00E+00	3.35E+00
1841	FP	50-Sn-132	131.9178	0.00E+00	1.75E-02
1842	FP	51-Sb-132	131.9145	0.00E+00	4.14E-03
1843	FP	51-Sb-132m	131.9145	0.00E+00	2.82E-03
1844	FP	52-Te-132	131.9086	0.00E+00	2.50E-06
1845	FP	53-I-132	131.9080	0.00E+00	8.39E-05
1846	FP	53-I-132m	131.9080	0.00E+00	1.39E-04
1847	FP	54-Xe-132	131.9041	0.00E+00	0.00E+00
1848	FP	55-Cs-132	131.9064	0.00E+00	1.24E-06
1849	FP	56-Ba-132	131.9051	0.00E+00	0.00E+00
1850	FP	49-In-133	132.9378	0.00E+00	4.20E+00
1851	FP	50-Sn-133	132.9238	0.00E+00	4.75E-01
1852	FP	51-Sb-133	132.9153	0.00E+00	4.62E-03
1853	FP	52-Te-133	132.9110	0.00E+00	9.24E-04
1854	FP	52-Te-133m	132.9110	0.00E+00	2.09E-04
1855	FP	53-I-133	132.9078	0.00E+00	9.26E-06
1856	FP	53-I-133m	132.9078	0.00E+00	7.70E-02
1857	FP	54-Xe-133	132.9059	0.00E+00	1.53E-06
1858	FP	54-Xe-133m	132.9059	0.00E+00	3.66E-06
1859	FP	55-Cs-133	132.9055	0.00E+00	0.00E+00
1860	FP	56-Ba-133	132.9060	0.00E+00	2.09E-09
1861	FP	57-La-133	132.9082	0.00E+00	4.92E-05
1862	FP	49-In-134	133.9442	0.00E+00	4.95E+00
1863	FP	50-Sn-134	133.9283	0.00E+00	6.60E-01
1864	FP	51-Sb-134	133.9204	0.00E+00	8.89E-01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1865	FP	51-Sb-134m	133.9204	0.00E+00	6.88E-02
1866	FP	52-Te-134	133.9114	0.00E+00	2.76E-04
1867	FP	53-I-134	133.9097	0.00E+00	2.20E-04
1868	FP	53-I-134m	133.9097	0.00E+00	3.28E-03
1869	FP	54-Xe-134	133.9054	0.00E+00	0.00E+00
1870	FP	54-Xe-134m	133.9054	0.00E+00	2.39E+00
1871	FP	55-Cs-134	133.9067	0.00E+00	1.06E-08
1872	FP	55-Cs-134m	133.9067	0.00E+00	6.61E-05
1873	FP	56-Ba-134	133.9045	0.00E+00	0.00E+00
1874	FP	49-In-135	134.9493	0.00E+00	7.53E+00
1875	FP	50-Sn-135	134.9347	0.00E+00	1.31E+00
1876	FP	51-Sb-135	134.9252	0.00E+00	4.13E-01
1877	FP	52-Te-135	134.9164	0.00E+00	3.65E-02
1878	FP	53-I-135	134.9100	0.00E+00	2.93E-05
1879	FP	54-Xe-135	134.9072	0.00E+00	2.11E-05
1880	FP	54-Xe-135m	134.9072	0.00E+00	7.56E-04
1881	FP	55-Cs-135	134.9060	0.00E+00	9.55E-15
1882	FP	55-Cs-135m	134.9060	0.00E+00	2.18E-04
1883	FP	56-Ba-135	134.9057	0.00E+00	0.00E+00
1884	FP	56-Ba-135m	134.9057	0.00E+00	6.71E-06
1885	FP	57-La-135	134.9070	0.00E+00	9.87E-06
1886	FP	58-Ce-135	134.9091	0.00E+00	1.09E-05
1887	FP	50-Sn-136	135.9393	0.00E+00	2.77E+00
1888	FP	51-Sb-136	135.9303	0.00E+00	7.51E-01
1889	FP	52-Te-136	135.9201	0.00E+00	3.96E-02
1890	FP	53-I-136	135.9147	0.00E+00	8.31E-03
1891	FP	53-I-136m	135.9147	0.00E+00	1.48E-02
1892	FP	54-Xe-136	135.9072	0.00E+00	0.00E+00
1893	FP	55-Cs-136	135.9073	0.00E+00	6.10E-07
1894	FP	55-Cs-136m	135.9073	0.00E+00	3.65E-02
1895	FP	56-Ba-136	135.9046	0.00E+00	0.00E+00
1896	FP	56-Ba-136m	135.9046	0.00E+00	2.25E+00
1897	FP	50-Sn-137	136.9460	0.00E+00	3.65E+00
1898	FP	51-Sb-137	136.9353	0.00E+00	1.54E+00
1899	FP	52-Te-137	136.9253	0.00E+00	2.78E-01
1900	FP	53-I-137	136.9179	0.00E+00	2.83E-02
1901	FP	54-Xe-137	136.9116	0.00E+00	3.03E-03
1902	FP	55-Cs-137	136.9071	0.00E+00	7.30E-10
1903	FP	56-Ba-137	136.9058	0.00E+00	0.00E+00
1904	FP	56-Ba-137m	136.9058	0.00E+00	4.53E-03
1905	FP	57-La-137	136.9065	0.00E+00	3.66E-13
1906	FP	58-Ce-137	136.9078	0.00E+00	2.14E-05
1907	FP	51-Sb-138	137.9408	0.00E+00	4.13E+00

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1908	FP	52-Te-138	137.9292	0.00E+00	4.95E-01
1909	FP	53-I-138	137.9223	0.00E+00	1.11E-01
1910	FP	54-Xe-138	137.9140	0.00E+00	8.20E-04
1911	FP	55-Cs-138	137.9110	0.00E+00	3.46E-04
1912	FP	55-Cs-138m	137.9110	0.00E+00	3.97E-03
1913	FP	56-Ba-138	137.9052	0.00E+00	0.00E+00
1914	FP	57-La-138	137.9071	0.00E+00	2.15E-19
1915	FP	58-Ce-138	137.9060	0.00E+00	0.00E+00
1916	FP	51-Sb-139	138.9460	0.00E+00	5.46E+00
1917	FP	52-Te-139	138.9347	0.00E+00	2.00E+00
1918	FP	53-I-139	138.9261	0.00E+00	3.04E-01
1919	FP	54-Xe-139	138.9188	0.00E+00	1.75E-02
1920	FP	55-Cs-139	138.9134	0.00E+00	1.25E-03
1921	FP	56-Ba-139	138.9088	0.00E+00	1.39E-04
1922	FP	57-La-139	138.9064	0.00E+00	0.00E+00
1923	FP	58-Ce-139	138.9066	0.00E+00	5.83E-08
1924	FP	58-Ce-139m	138.9066	0.00E+00	1.26E-02
1925	FP	59-Pr-139	138.9089	0.00E+00	4.37E-05
1926	FP	52-Te-140	139.9388	0.00E+00	2.28E+00
1927	FP	53-I-140	139.9310	0.00E+00	8.06E-01
1928	FP	54-Xe-140	139.9216	0.00E+00	5.10E-02
1929	FP	55-Cs-140	139.9173	0.00E+00	1.09E-02
1930	FP	56-Ba-140	139.9106	0.00E+00	6.29E-07
1931	FP	57-La-140	139.9095	0.00E+00	4.78E-06
1932	FP	58-Ce-140	139.9054	0.00E+00	0.00E+00
1933	FP	59-Pr-140	139.9091	0.00E+00	3.41E-03
1934	FP	60-Nd-140	139.9095	0.00E+00	2.38E-06
1935	FP	52-Te-141	140.9447	0.00E+00	3.25E+00
1936	FP	53-I-141	140.9350	0.00E+00	1.61E+00
1937	FP	54-Xe-141	140.9267	0.00E+00	4.01E-01
1938	FP	55-Cs-141	140.9200	0.00E+00	2.79E-02
1939	FP	56-Ba-141	140.9144	0.00E+00	6.32E-04
1940	FP	57-La-141	140.9110	0.00E+00	4.91E-05
1941	FP	58-Ce-141	140.9083	0.00E+00	2.47E-07
1942	FP	59-Pr-141	140.9077	0.00E+00	0.00E+00
1943	FP	60-Nd-141	140.9096	0.00E+00	7.73E-05
1944	FP	60-Nd-141m	140.9096	0.00E+00	1.12E-02
1945	FP	61-Pm-141	140.9136	0.00E+00	5.53E-04
1946	FP	52-Te-142	141.9491	0.00E+00	3.47E+00
1947	FP	53-I-142	141.9402	0.00E+00	3.12E+00
1948	FP	54-Xe-142	141.9297	0.00E+00	5.64E-01
1949	FP	55-Cs-142	141.9243	0.00E+00	4.12E-01
1950	FP	56-Ba-142	141.9164	0.00E+00	1.09E-03

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1951	FP	57-La-142	141.9141	0.00E+00	1.27E-04
1952	FP	58-Ce-142	141.9092	0.00E+00	0.00E+00
1953	FP	59-Pr-142	141.9100	0.00E+00	1.01E-05
1954	FP	59-Pr-142m	141.9100	0.00E+00	7.91E-04
1955	FP	60-Nd-142	141.9077	0.00E+00	0.00E+00
1956	FP	53-I-143	142.9446	0.00E+00	2.34E+00
1957	FP	54-Xe-143	142.9351	0.00E+00	2.31E+00
1958	FP	55-Cs-143	142.9274	0.00E+00	3.87E-01
1959	FP	56-Ba-143	142.9206	0.00E+00	4.78E-02
1960	FP	57-La-143	142.9161	0.00E+00	8.14E-04
1961	FP	58-Ce-143	142.9124	0.00E+00	5.83E-06
1962	FP	59-Pr-143	142.9108	0.00E+00	5.91E-07
1963	FP	60-Nd-143	142.9098	0.00E+00	0.00E+00
1964	FP	61-Pm-143	142.9109	0.00E+00	3.03E-08
1965	FP	62-Sm-143	142.9146	0.00E+00	1.32E-03
1966	FP	62-Sm-143m	142.9146	0.00E+00	1.05E-02
1967	FP	53-I-144	143.9500	0.00E+00	3.57E+00
1968	FP	54-Xe-144	143.9385	0.00E+00	6.03E-01
1969	FP	55-Cs-144	143.9321	0.00E+00	6.97E-01
1970	FP	56-Ba-144	143.9229	0.00E+00	6.03E-02
1971	FP	57-La-144	143.9196	0.00E+00	1.70E-02
1972	FP	58-Ce-144	143.9137	0.00E+00	2.82E-08
1973	FP	59-Pr-144	143.9133	0.00E+00	6.69E-04
1974	FP	59-Pr-144m	143.9133	0.00E+00	1.60E-03
1975	FP	60-Nd-144	143.9101	0.00E+00	9.59E-24
1976	FP	61-Pm-144	143.9126	0.00E+00	2.21E-08
1977	FP	62-Sm-144	143.9120	0.00E+00	0.00E+00
1978	FP	54-Xe-145	144.9441	0.00E+00	3.69E+00
1979	FP	55-Cs-145	144.9355	0.00E+00	1.18E+00
1980	FP	56-Ba-145	144.9276	0.00E+00	1.61E-01
1981	FP	57-La-145	144.9216	0.00E+00	2.80E-02
1982	FP	58-Ce-145	144.9172	0.00E+00	3.84E-03
1983	FP	59-Pr-145	144.9145	0.00E+00	3.22E-05
1984	FP	60-Nd-145	144.9126	0.00E+00	0.00E+00
1985	FP	61-Pm-145	144.9128	0.00E+00	1.24E-09
1986	FP	62-Sm-145	144.9134	0.00E+00	2.36E-08
1987	FP	54-Xe-146	145.9478	0.00E+00	1.88E+00
1988	FP	55-Cs-146	145.9403	0.00E+00	2.16E+00
1989	FP	56-Ba-146	145.9302	0.00E+00	3.12E-01
1990	FP	57-La-146	145.9258	0.00E+00	1.11E-01
1991	FP	57-La-146m	145.9258	0.00E+00	6.93E-02
1992	FP	58-Ce-146	145.9188	0.00E+00	8.54E-04
1993	FP	59-Pr-146	145.9176	0.00E+00	4.78E-04

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
1994	FP	60-Nd-146	145.9131	0.00E+00	0.00E+00
1995	FP	61-Pm-146	145.9147	0.00E+00	3.97E-09
1996	FP	62-Sm-146	145.9130	0.00E+00	2.13E-16
1997	FP	54-Xe-147	146.9536	0.00E+00	6.93E+00
1998	FP	55-Cs-147	146.9442	0.00E+00	3.01E+00
1999	FP	56-Ba-147	146.9350	0.00E+00	7.75E-01
2000	FP	57-La-147	146.9282	0.00E+00	1.71E-01
2001	FP	58-Ce-147	146.9227	0.00E+00	1.23E-02
2002	FP	59-Pr-147	146.9190	0.00E+00	8.62E-04
2003	FP	60-Nd-147	146.9161	0.00E+00	7.31E-07
2004	FP	61-Pm-147	146.9151	0.00E+00	8.37E-09
2005	FP	62-Sm-147	146.9149	0.00E+00	2.07E-19
2006	FP	63-Eu-147	146.9167	0.00E+00	3.33E-07
2007	FP	64-Gd-147	146.9191	0.00E+00	5.06E-06
2008	FP	55-Cs-148	147.9492	0.00E+00	4.75E+00
2009	FP	56-Ba-148	147.9377	0.00E+00	1.13E+00
2010	FP	57-La-148	147.9322	0.00E+00	5.50E-01
2011	FP	58-Ce-148	147.9244	0.00E+00	1.24E-02
2012	FP	59-Pr-148	147.9221	0.00E+00	5.04E-03
2013	FP	59-Pr-148m	147.9221	0.00E+00	5.75E-03
2014	FP	60-Nd-148	147.9169	0.00E+00	0.00E+00
2015	FP	61-Pm-148	147.9175	0.00E+00	1.49E-06
2016	FP	61-Pm-148m	147.9207	0.00E+00	1.94E-07
2017	FP	62-Sm-148	147.9148	0.00E+00	3.14E-24
2018	FP	55-Cs-149	148.9529	0.00E+00	1.39E+01
2019	FP	56-Ba-149	148.9426	0.00E+00	2.02E+00
2020	FP	57-La-149	148.9347	0.00E+00	6.60E-01
2021	FP	58-Ce-149	148.9284	0.00E+00	1.31E-01
2022	FP	59-Pr-149	148.9237	0.00E+00	5.11E-03
2023	FP	60-Nd-149	148.9202	0.00E+00	1.11E-04
2024	FP	61-Pm-149	148.9183	0.00E+00	3.63E-06
2025	FP	62-Sm-149	148.9172	0.00E+00	0.00E+00
2026	FP	63-Eu-149	148.9179	0.00E+00	8.62E-08
2027	FP	64-Gd-149	148.9193	0.00E+00	8.65E-07
2028	FP	55-Cs-150	149.9582	0.00E+00	1.39E+01
2029	FP	56-Ba-150	149.9457	0.00E+00	2.31E+00
2030	FP	57-La-150	149.9388	0.00E+00	8.06E-01
2031	FP	58-Ce-150	149.9304	0.00E+00	1.73E-01
2032	FP	59-Pr-150	149.9267	0.00E+00	1.12E-01
2033	FP	60-Nd-150	149.9209	0.00E+00	2.78E-27
2034	FP	61-Pm-150	149.9210	0.00E+00	7.18E-05
2035	FP	62-Sm-150	149.9173	0.00E+00	0.00E+00
2036	FP	55-Cs-151	150.9622	0.00E+00	1.39E+01

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
2037	FP	56-Ba-151	150.9508	0.00E+00	2.68E+00
2038	FP	57-La-151	150.9417	0.00E+00	8.91E-01
2039	FP	58-Ce-151	150.9340	0.00E+00	3.94E-01
2040	FP	59-Pr-151	150.9283	0.00E+00	3.67E-02
2041	FP	60-Nd-151	150.9238	0.00E+00	9.29E-04
2042	FP	61-Pm-151	150.9212	0.00E+00	6.78E-06
2043	FP	62-Sm-151	150.9199	0.00E+00	2.44E-10
2044	FP	63-Eu-151	150.9198	0.00E+00	0.00E+00
2045	FP	64-Gd-151	150.9203	0.00E+00	6.47E-08
2046	FP	65-Tb-151	150.9231	0.00E+00	1.09E-05
2047	FP	56-Ba-152	151.9543	0.00E+00	3.04E+00
2048	FP	57-La-152	151.9462	0.00E+00	1.54E+00
2049	FP	58-Ce-152	151.9365	0.00E+00	4.95E-01
2050	FP	59-Pr-152	151.9315	0.00E+00	1.91E-01
2051	FP	60-Nd-152	151.9247	0.00E+00	1.01E-03
2052	FP	61-Pm-152	151.9235	0.00E+00	2.80E-03
2053	FP	61-Pm-152m	151.9235	0.00E+00	1.54E-03
2054	FP	62-Sm-152	151.9197	0.00E+00	0.00E+00
2055	FP	63-Eu-152	151.9217	0.00E+00	1.62E-09
2056	FP	63-Eu-152m	151.9217	0.00E+00	2.07E-05
2057	FP	64-Gd-152	151.9198	0.00E+00	2.03E-22
2058	FP	56-Ba-153	152.9596	0.00E+00	4.39E+00
2059	FP	57-La-153	152.9496	0.00E+00	2.03E+00
2060	FP	58-Ce-153	152.9406	0.00E+00	7.08E-01
2061	FP	59-Pr-153	152.9338	0.00E+00	1.62E-01
2062	FP	60-Nd-153	152.9277	0.00E+00	2.19E-02
2063	FP	61-Pm-153	152.9241	0.00E+00	2.20E-03
2064	FP	62-Sm-153	152.9221	0.00E+00	4.14E-06
2065	FP	63-Eu-153	152.9212	0.00E+00	0.00E+00
2066	FP	64-Gd-153	152.9218	0.00E+00	3.34E-08
2067	FP	65-Tb-153	152.9234	0.00E+00	3.43E-06
2068	FP	57-La-154	153.9545	0.00E+00	3.04E+00
2069	FP	58-Ce-154	153.9434	0.00E+00	8.94E-01
2070	FP	59-Pr-154	153.9375	0.00E+00	3.01E-01
2071	FP	60-Nd-154	153.9295	0.00E+00	2.68E-02
2072	FP	61-Pm-154	153.9265	0.00E+00	6.68E-03
2073	FP	61-Pm-154m	153.9265	0.00E+00	4.31E-03
2074	FP	62-Sm-154	153.9222	0.00E+00	0.00E+00
2075	FP	63-Eu-154	153.9230	0.00E+00	2.55E-09
2076	FP	63-Eu-154m	153.9230	0.00E+00	2.51E-04
2077	FP	64-Gd-154	153.9209	0.00E+00	0.00E+00
2078	FP	57-La-155	154.9583	0.00E+00	3.77E+00
2079	FP	58-Ce-155	154.9480	0.00E+00	1.47E+00

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
2080	FP	59-Pr-155	154.9401	0.00E+00	8.14E-01
2081	FP	60-Nd-155	154.9329	0.00E+00	7.79E-02
2082	FP	61-Pm-155	154.9281	0.00E+00	1.67E-02
2083	FP	62-Sm-155	154.9246	0.00E+00	5.18E-04
2084	FP	63-Eu-155	154.9229	0.00E+00	4.62E-09
2085	FP	64-Gd-155	154.9226	0.00E+00	0.00E+00
2086	FP	64-Gd-155m	154.9226	0.00E+00	2.17E+01
2087	FP	65-Tb-155	154.9235	0.00E+00	1.51E-06
2088	FP	66-Dy-155	154.9258	0.00E+00	1.94E-05
2089	FP	58-Ce-156	155.9513	0.00E+00	1.88E+00
2090	FP	59-Pr-156	155.9443	0.00E+00	9.46E-01
2091	FP	60-Nd-156	155.9350	0.00E+00	1.26E-01
2092	FP	61-Pm-156	155.9311	0.00E+00	2.60E-02
2093	FP	62-Sm-156	155.9255	0.00E+00	2.05E-05
2094	FP	63-Eu-156	155.9247	0.00E+00	5.28E-07
2095	FP	64-Gd-156	155.9221	0.00E+00	0.00E+00
2096	FP	65-Tb-156	155.9247	0.00E+00	1.50E-06
2097	FP	65-Tb-156m	155.9247	0.00E+00	7.89E-06
2098	FP	66-Dy-156	155.9243	0.00E+00	0.00E+00
2099	FP	58-Ce-157	156.9563	0.00E+00	2.85E+00
2100	FP	59-Pr-157	156.9474	0.00E+00	1.16E+00
2101	FP	60-Nd-157	156.9390	0.00E+00	3.64E-01
2102	FP	61-Pm-157	156.9330	0.00E+00	6.56E-02
2103	FP	62-Sm-157	156.9284	0.00E+00	1.44E-03
2104	FP	63-Eu-157	156.9254	0.00E+00	1.27E-05
2105	FP	64-Gd-157	156.9240	0.00E+00	0.00E+00
2106	FP	65-Tb-157	156.9240	0.00E+00	3.09E-10
2107	FP	66-Dy-157	156.9255	0.00E+00	2.37E-05
2108	FP	59-Pr-158	157.9520	0.00E+00	5.17E+00
2109	FP	60-Nd-158	157.9416	0.00E+00	5.21E-01
2110	FP	61-Pm-158	157.9366	0.00E+00	1.44E-01
2111	FP	62-Sm-158	157.9300	0.00E+00	2.18E-03
2112	FP	63-Eu-158	157.9279	0.00E+00	2.52E-04
2113	FP	64-Gd-158	157.9241	0.00E+00	0.00E+00
2114	FP	65-Tb-158	157.9254	0.00E+00	1.22E-10
2115	FP	65-Tb-158m	157.9254	0.00E+00	6.48E-02
2116	FP	66-Dy-158	157.9244	0.00E+00	0.00E+00
2117	FP	59-Pr-159	158.9555	0.00E+00	6.57E+00
2118	FP	60-Nd-159	158.9461	0.00E+00	8.97E-01
2119	FP	61-Pm-159	158.9390	0.00E+00	4.72E-01
2120	FP	62-Sm-159	158.9332	0.00E+00	6.10E-02
2121	FP	63-Eu-159	158.9291	0.00E+00	6.38E-04
2122	FP	64-Gd-159	158.9264	0.00E+00	1.04E-05

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
2123	FP	65-Tb-159	158.9254	0.00E+00	0.00E+00
2124	FP	66-Dy-159	158.9257	0.00E+00	5.56E-08
2125	FP	67-Ho-159	158.9277	0.00E+00	3.50E-04
2126	FP	67-Ho-159m	158.9277	0.00E+00	8.35E-02
2127	FP	60-Nd-160	159.9491	0.00E+00	1.18E+00
2128	FP	61-Pm-160	159.9430	0.00E+00	4.44E-01
2129	FP	62-Sm-160	159.9351	0.00E+00	7.22E-02
2130	FP	63-Eu-160	159.9320	0.00E+00	1.82E-02
2131	FP	64-Gd-160	159.9270	0.00E+00	0.00E+00
2132	FP	65-Tb-160	159.9272	0.00E+00	1.11E-07
2133	FP	66-Dy-160	159.9252	0.00E+00	0.00E+00
2134	FP	60-Nd-161	160.9539	0.00E+00	1.42E+00
2135	FP	61-Pm-161	160.9459	0.00E+00	6.51E-01
2136	FP	62-Sm-161	160.9388	0.00E+00	1.44E-01
2137	FP	63-Eu-161	160.9337	0.00E+00	2.67E-02
2138	FP	64-Gd-161	160.9297	0.00E+00	3.16E-03
2139	FP	65-Tb-161	160.9276	0.00E+00	1.16E-06
2140	FP	66-Dy-161	160.9269	0.00E+00	0.00E+00
2141	FP	67-Ho-161	160.9279	0.00E+00	7.76E-05
2142	FP	67-Ho-161m	160.9279	0.00E+00	1.03E-01
2143	FP	68-Er-161	160.9300	0.00E+00	6.00E-05
2144	FP	61-Pm-162	161.9503	0.00E+00	2.59E+00
2145	FP	62-Sm-162	161.9412	0.00E+00	2.89E-01
2146	FP	63-Eu-162	161.9370	0.00E+00	6.54E-02
2147	FP	64-Gd-162	161.9310	0.00E+00	1.38E-03
2148	FP	65-Tb-162	161.9295	0.00E+00	1.52E-03
2149	FP	66-Dy-162	161.9268	0.00E+00	0.00E+00
2150	FP	67-Ho-162	161.9291	0.00E+00	7.70E-04
2151	FP	67-Ho-162m	161.9291	0.00E+00	1.72E-04
2152	FP	68-Er-162	161.9288	0.00E+00	0.00E+00
2153	FP	61-Pm-163	162.9537	0.00E+00	3.47E+00
2154	FP	62-Sm-163	162.9454	0.00E+00	3.97E-01
2155	FP	63-Eu-163	162.9392	0.00E+00	9.00E-02
2156	FP	64-Gd-163	162.9340	0.00E+00	1.02E-02
2157	FP	65-Tb-163	162.9306	0.00E+00	5.92E-04
2158	FP	66-Dy-163	162.9287	0.00E+00	0.00E+00
2159	FP	67-Ho-163	162.9287	0.00E+00	4.81E-12
2160	FP	67-Ho-163m	162.9287	0.00E+00	6.36E-01
2161	FP	68-Er-163	162.9300	0.00E+00	1.54E-04
2162	FP	62-Sm-164	163.9483	0.00E+00	5.65E-01
2163	FP	63-Eu-164	163.9430	0.00E+00	2.44E-01
2164	FP	64-Gd-164	163.9359	0.00E+00	1.54E-02
2165	FP	65-Tb-164	163.9333	0.00E+00	3.85E-03

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
2166	FP	66-Dy-164	163.9292	0.00E+00	0.00E+00
2167	FP	67-Ho-164	163.9302	0.00E+00	3.98E-04
2168	FP	67-Ho-164m	163.9302	0.00E+00	3.08E-04
2169	FP	68-Er-164	163.9292	0.00E+00	0.00E+00
2170	FP	62-Sm-165	164.9530	0.00E+00	9.07E-01
2171	FP	63-Eu-165	164.9457	0.00E+00	3.01E-01
2172	FP	64-Gd-165	164.9394	0.00E+00	6.73E-02
2173	FP	65-Tb-165	164.9349	0.00E+00	5.47E-03
2174	FP	66-Dy-165	164.9317	0.00E+00	8.25E-05
2175	FP	66-Dy-165m	164.9317	0.00E+00	9.19E-03
2176	FP	67-Ho-165	164.9303	0.00E+00	0.00E+00
2177	FP	68-Er-165	164.9307	0.00E+00	1.86E-05
2178	FP	69-Tm-165	164.9324	0.00E+00	6.41E-06
2179	FP	63-Eu-166	165.9500	0.00E+00	1.73E+00
2180	FP	64-Gd-166	165.9416	0.00E+00	1.44E-01
2181	FP	65-Tb-166	165.9380	0.00E+00	2.76E-02
2182	FP	66-Dy-166	165.9328	0.00E+00	2.36E-06
2183	FP	67-Ho-166	165.9323	0.00E+00	7.18E-06
2184	FP	67-Ho-166m	165.9324	0.00E+00	1.83E-11
2185	FP	68-Er-166	165.9303	0.00E+00	0.00E+00
2186	FP	69-Tm-166	165.9335	0.00E+00	2.50E-05
2187	FP	70-Yb-166	165.9339	0.00E+00	3.40E-06
2188	FP	63-Eu-167	166.9532	0.00E+00	3.47E+00
2189	FP	64-Gd-167	166.9456	0.00E+00	2.31E-01
2190	FP	65-Tb-167	166.9400	0.00E+00	3.57E-02
2191	FP	66-Dy-167	166.9357	0.00E+00	1.86E-03
2192	FP	67-Ho-167	166.9331	0.00E+00	6.21E-05
2193	FP	68-Er-167	166.9321	0.00E+00	0.00E+00
2194	FP	68-Er-167m	166.9321	0.00E+00	3.05E-01
2195	FP	69-Tm-167	166.9328	0.00E+00	8.67E-07
2196	FP	70-Yb-167	166.9350	0.00E+00	6.60E-04
2197	FP	64-Gd-168	167.9484	0.00E+00	2.31E+00
2198	FP	65-Tb-168	167.9436	0.00E+00	8.45E-02
2199	FP	66-Dy-168	167.9371	0.00E+00	1.33E-03
2200	FP	67-Ho-168	167.9355	0.00E+00	3.86E-03
2201	FP	68-Er-168	167.9324	0.00E+00	0.00E+00
2202	FP	69-Tm-168	167.9342	0.00E+00	8.62E-08
2203	FP	70-Yb-168	167.9339	0.00E+00	0.00E+00
2204	FP	64-Gd-169	168.9529	0.00E+00	6.93E-01
2205	FP	65-Tb-169	168.9462	0.00E+00	3.47E-01
2206	FP	66-Dy-169	168.9403	0.00E+00	1.78E-02
2207	FP	67-Ho-169	168.9369	0.00E+00	2.45E-03
2208	FP	68-Er-169	168.9346	0.00E+00	8.54E-07

continues on next page

Table 10.2.13 – continued from previous page

index	sublib	nuclide	mass (g/mol)	abundance (atom%)	decay (1/s)
2209	FP	69-Tm-169	168.9342	0.00E+00	0.00E+00
2210	FP	70-Yb-169	168.9352	0.00E+00	2.51E-07
2211	FP	70-Yb-169m	168.9352	0.00E+00	1.51E-02
2212	FP	71-Lu-169	168.9377	0.00E+00	5.65E-06
2213	FP	71-Lu-169m	168.9377	0.00E+00	4.33E-03
2214	FP	65-Tb-170	169.9503	0.00E+00	2.31E-01
2215	FP	66-Dy-170	169.9424	0.00E+00	2.31E-02
2216	FP	67-Ho-170	169.9396	0.00E+00	4.19E-03
2217	FP	67-Ho-170m	169.9396	0.00E+00	1.61E-02
2218	FP	68-Er-170	169.9355	0.00E+00	0.00E+00
2219	FP	69-Tm-170	169.9358	0.00E+00	6.24E-08
2220	FP	70-Yb-170	169.9348	0.00E+00	0.00E+00
2221	FP	65-Tb-171	170.9533	0.00E+00	1.39E+00
2222	FP	66-Dy-171	170.9462	0.00E+00	1.16E-01
2223	FP	67-Ho-171	170.9415	0.00E+00	1.31E-02
2224	FP	68-Er-171	170.9380	0.00E+00	2.56E-05
2225	FP	69-Tm-171	170.9364	0.00E+00	1.14E-08
2226	FP	70-Yb-171	170.9363	0.00E+00	0.00E+00
2227	FP	71-Lu-171	170.9379	0.00E+00	9.74E-07
2228	FP	71-Lu-171m	170.9379	0.00E+00	8.77E-03
2229	FP	72-Hf-171	170.9405	0.00E+00	1.59E-05
2230	FP	66-Dy-172	171.9488	0.00E+00	2.31E-01
2231	FP	67-Ho-172	171.9448	0.00E+00	2.77E-02
2232	FP	68-Er-172	171.9394	0.00E+00	3.91E-06
2233	FP	69-Tm-172	171.9384	0.00E+00	3.03E-06
2234	FP	70-Yb-172	171.9364	0.00E+00	0.00E+00
2235	FP	71-Lu-172	171.9391	0.00E+00	1.20E-06
2236	FP	71-Lu-172m	171.9391	0.00E+00	3.12E-03
2237	FP	72-Hf-172	171.9395	0.00E+00	1.17E-08

10.2.13 DECAY RESOURCE CONTENTS

A concise representation of the data contained in the decay resource (i.e., the first 500 lines) is shown below in Example 10.2.4. Note that the actual decay resource file currently also defines the complete ORIGEN nuclide set, including the multiple versions of some nuclides by sublibrary, e.g., ^{155}Gd-155, as both a light nuclide/activation product and a fission product. The content list below contains no such duplication.

Example 10.2.4: Contents of decay resource.

```
---------------
ind: index
izzzaaa: 7-digit id composed of
i: 1-digit isomeric (metastable) state
zzz: 3-digit atomic number
aaa: 3-digit mass number
symbol: element-aaa(m) where m is first metastable state
dk: decay constant in 1/s
```

(continues on next page)

(continued from previous page)

```
Q: recoverable decay energy in MeV
fQphot: fraction of Q decay energy associated with gamma rays
mode: decay mode and yield pairs in the form
decay_mode1(isomeric_states1) yield1
decay_mode2(isomeric_states2) yield2
...
---------------
ind izzzaaa symbol dk(1/s) Q(MeV) fQphot mode(isomeric) yield
0 1001 h1 0.000e+00 0.000e+00 0.000
1 1002 h2 0.000e+00 0.000e+00 0.000
2 1003 h3 1.783e-09 5.690e-03 0.000 beta-(0) 1.000e+00
3 2003 he3 0.000e+00 0.000e+00 0.000
4 2004 he4 0.000e+00 0.000e+00 0.000
5 2005 he5 6.931e+02 8.900e-01 0.000 alpha(0) 1.000e+00
6 2006 he6 8.592e-01 1.568e+00 0.000 beta-(0) 1.000e+00
7 3006 li6 0.000e+00 0.000e+00 0.000
8 3007 li7 0.000e+00 0.000e+00 0.000
9 3008 li8 8.271e-01 9.363e+00 0.004 beta-,alpha(0) 1.000e+00
10 4007 be7 1.507e-07 4.986e-02 1.000 beta+(0) 1.000e+00
11 4008 be8 6.931e+02 9.184e-02 0.000 alpha(0) 1.000e+00
12 4009 be9 0.000e+00 0.000e+00 0.000
13 4010 be10 1.455e-14 2.026e-01 0.000 beta-(0) 1.000e+00
14 4011 be11 5.019e-02 7.501e+00 0.499 beta-(0) 9.690e-01 beta-,alpha(0)
3.100e-02
15 5010 b10 0.000e+00 0.000e+00 0.000
16 5011 b11 0.000e+00 0.000e+00 0.000
17 5012 b12 3.431e+01 6.407e+00 0.009 beta-(0) 1.000e+00
18 6012 c12 0.000e+00 0.000e+00 0.726
19 6013 c13 0.000e+00 0.000e+00 0.726
20 6014 c14 3.854e-12 4.947e-02 0.000 beta-(0) 1.000e+00
21 6015 c15 2.830e-01 6.352e+00 0.528 beta-(0) 1.000e+00
22 7013 n13 1.159e-03 1.511e+00 0.675 beta+(0) 1.000e+00
23 7014 n14 0.000e+00 0.000e+00 0.675
24 7015 n15 0.000e+00 0.000e+00 0.675
25 7016 n16 9.722e-02 7.259e+00 0.619 beta-(0) 1.000e+00 beta-,alpha(0)
1.200e-05
26 8016 o16 0.000e+00 0.000e+00 0.582
27 8017 o17 0.000e+00 0.000e+00 0.582
28 8018 o18 0.000e+00 0.000e+00 0.582
29 8019 o19 2.579e-02 2.725e+00 0.345 beta-(0) 1.000e+00
30 9019 f19 0.000e+00 0.000e+00 0.804
31 9020 f20 6.209e-02 4.115e+00 0.397 beta-(0) 1.000e+00
32 10020 ne20 0.000e+00 0.000e+00 0.515
33 10021 ne21 0.000e+00 0.000e+00 0.515
34 10022 ne22 0.000e+00 0.000e+00 0.515
35 10023 ne23 1.861e-02 2.066e+00 0.080 beta-(0) 1.000e+00
36 11022 na22 8.439e-09 2.388e+00 0.918 beta+(0) 1.000e+00
37 11023 na23 0.000e+00 0.000e+00 0.918
38 11024 na24 1.284e-05 4.677e+00 0.881 beta-(0) 1.000e+00
39 1011024 na24m 3.435e+01 4.733e-01 0.997 beta-(0) 5.000e-04 i.t.(0)
9.995e-01
40 11025 na25 1.173e-02 1.942e+00 0.224 beta-(0) 1.000e+00
41 12024 mg24 0.000e+00 0.000e+00 0.442
42 12025 mg25 0.000e+00 0.000e+00 0.442
43 12026 mg26 0.000e+00 0.000e+00 0.442
44 12027 mg27 1.221e-03 1.593e+00 0.559 beta-(0) 1.000e+00
45 12028 mg28 9.206e-06 1.531e+00 0.895 beta-(0) 1.000e+00
46 13026 al26 3.063e-14 3.119e+00 0.858 beta+(0) 1.000e+00
47 13027 al27 0.000e+00 0.000e+00 0.415
48 13028 al28 5.154e-03 3.020e+00 0.589 beta-(0) 1.000e+00
49 13029 al29 1.761e-03 2.356e+00 0.585 beta-(0) 1.000e+00
50 13030 al30 1.915e-01 5.656e+00 0.592 beta-(0) 1.000e+00
51 14028 si28 0.000e+00 0.000e+00 0.374
52 14029 si29 0.000e+00 0.000e+00 0.374
53 14030 si30 0.000e+00 0.000e+00 0.374
```

(continues on next page)

(continued from previous page)

```
54 14031 si31 7.344e-05 5.965e-01 0.001 beta-(0) 1.000e+00
55 14032 si32 1.436e-10 6.843e-02 0.000 beta-(0) 1.000e+00
56 15031 p31 0.000e+00 0.000e+00 0.415
57 15032 p32 5.625e-07 6.949e-01 0.000 beta-(0) 1.000e+00
58 15033 p33 3.166e-07 7.643e-02 0.000 beta-(0) 1.000e+00
59 15034 p34 5.576e-02 2.639e+00 0.126 beta-(0) 1.000e+00
60 16032 s32 0.000e+00 0.000e+00 0.342
61 16033 s33 0.000e+00 0.000e+00 0.342
62 16034 s34 0.000e+00 0.000e+00 0.342
63 16035 s35 9.168e-08 4.876e-02 0.000 beta-(0) 1.000e+00
64 16036 s36 0.000e+00 0.000e+00 0.000
65 16037 s37 2.288e-03 3.732e+00 0.785 beta-(0) 1.000e+00
66 17035 cl35 0.000e+00 0.000e+00 0.820
67 17036 cl36 7.297e-14 2.976e-01 0.000 beta-(0) 9.810e-01 beta+(0)
1.900e-02
68 17037 cl37 0.000e+00 0.000e+00 0.000
69 17038 cl38 3.103e-04 3.031e+00 0.498 beta-(0) 1.000e+00
70 1017038 cl38m 9.694e-01 6.714e-01 0.999 i.t.(0) 1.000e+00
71 18036 ar36 0.000e+00 0.000e+00 0.317
72 18037 ar37 2.290e-07 2.165e-03 0.106 beta+(0) 1.000e+00
73 18038 ar38 0.000e+00 0.000e+00 0.106
74 18039 ar39 8.165e-11 2.188e-01 0.000 beta-(0) 1.000e+00
75 18040 ar40 0.000e+00 0.000e+00 0.000
76 18041 ar41 1.054e-04 1.748e+00 0.734 beta-(0) 1.000e+00
77 18042 ar42 6.676e-10 2.326e-01 0.000 beta-(0) 1.000e+00
78 19039 k39 0.000e+00 0.000e+00 0.306
79 19040 k40 1.760e-17 6.553e-01 0.238 beta-(0) 8.914e-01 beta+(0)
1.086e-01
80 19041 k41 0.000e+00 0.000e+00 0.238
81 19042 k42 1.558e-05 1.709e+00 0.163 beta-(0) 1.000e+00
82 19043 k43 8.634e-06 1.282e+00 0.752 beta-(0) 1.000e+00
83 19044 k44 5.220e-04 3.825e+00 0.617 beta-(0) 1.000e+00
84 20040 ca40 0.000e+00 0.000e+00 0.285
85 20041 ca41 2.153e-13 3.112e-03 0.130 beta+(0) 1.000e+00
86 20042 ca42 0.000e+00 0.000e+00 0.130
87 20043 ca43 0.000e+00 0.000e+00 0.130
88 20044 ca44 0.000e+00 0.000e+00 0.130
89 20045 ca45 4.934e-08 7.686e-02 0.000 beta-(0) 1.000e+00 beta-(1)
1.900e-05
90 20046 ca46 0.000e+00 4.942e-01 0.000
91 20047 ca47 1.769e-06 1.347e+00 0.704 beta-(0) 1.000e+00
92 20048 ca48 9.550e-28 1.650e+00 0.014 beta-(0) 2.500e-01
beta-,beta-(0) 7.500e-01
93 20049 ca49 1.325e-03 4.042e+00 0.786 beta-(0) 1.000e+00
94 21044 sc44 4.850e-05 2.732e+00 0.782 beta+(0) 1.000e+00
95 1021044 sc44m 3.285e-06 3.073e-01 0.893 beta+(0) 1.200e-02 i.t.(0)
9.880e-01
96 21045 sc45 0.000e+00 0.000e+00 0.893
97 1021045 sc45m 2.180e+00 1.232e-02 0.056 i.t.(0) 1.000e+00
98 21046 sc46 9.575e-08 2.121e+00 0.947 beta-(0) 1.000e+00
99 1021046 sc46m 3.697e-02 1.426e-01 0.629 i.t.(0) 1.000e+00
100 21047 sc47 2.395e-06 2.713e-01 0.401 beta-(0) 1.000e+00
101 21048 sc48 4.409e-06 3.574e+00 0.938 beta-(0) 1.000e+00
102 21049 sc49 2.020e-04 8.247e-01 0.001 beta-(0) 1.000e+00
103 21050 sc50 6.762e-03 4.830e+00 0.661 beta-(0) 1.000e+00
104 22044 ti44 3.661e-10 1.504e-01 0.928 beta+(0) 1.000e+00
105 22045 ti45 6.251e-05 1.243e+00 0.700 beta+(0) 1.000e+00
106 22046 ti46 0.000e+00 0.000e+00 0.700
107 22047 ti47 0.000e+00 0.000e+00 0.700
108 22048 ti48 0.000e+00 0.000e+00 0.700
109 22049 ti49 0.000e+00 0.000e+00 0.700
110 22050 ti50 0.000e+00 0.000e+00 0.700
111 22051 ti51 2.006e-03 1.239e+00 0.298 beta-(0) 1.000e+00
112 23048 v48 5.023e-07 3.066e+00 0.952 beta+(0) 1.000e+00
113 23049 v49 2.431e-08 4.451e-03 0.199 beta+(0) 1.000e+00
```

(continues on next page)

(continued from previous page)

```
114 23050 v50 1.569e-25 1.439e+00 0.989 beta-(0) 1.700e-01 beta+(0)
8.300e-01
115 23051 v51 0.000e+00 0.000e+00 0.989
116 23052 v52 3.086e-03 2.514e+00 0.575 beta-(0) 1.000e+00
117 23053 v53 7.487e-03 2.047e+00 0.507 beta-(0) 1.000e+00
118 23054 v54 1.392e-02 5.577e+00 0.737 beta-(0) 1.000e+00
119 24048 cr48 8.930e-06 4.424e-01 0.981 beta+(0) 1.000e+00
120 24049 cr49 2.731e-04 1.658e+00 0.636 beta+(0) 1.000e+00
121 24050 cr50 0.000e+00 0.000e+00 0.636
122 24051 cr51 2.896e-07 3.668e-02 0.895 beta+(0) 1.000e+00
123 24052 cr52 0.000e+00 0.000e+00 0.895
124 24053 cr53 0.000e+00 0.000e+00 0.895
125 24054 cr54 0.000e+00 0.000e+00 0.895
126 24055 cr55 3.303e-03 1.102e+00 0.001 beta-(0) 1.000e+00
127 24066 cr66 6.931e+01 8.273e+00 0.500 beta-(0) 1.000e+00
128 24067 cr67 1.386e+01 9.576e+00 0.500 beta-(0) 1.000e+00
129 25052 mn52 1.435e-06 3.533e+00 0.979 beta+(0) 1.000e+00
130 25053 mn53 5.936e-15 5.387e-03 0.252 beta+(0) 1.000e+00
131 25054 mn54 2.571e-08 8.402e-01 0.995 beta+(0) 1.000e+00
132 25055 mn55 0.000e+00 0.000e+00 0.995
133 25056 mn56 7.466e-05 2.523e+00 0.671 beta-(0) 1.000e+00
134 25057 mn57 8.116e-03 1.205e+00 0.083 beta-(0) 1.000e+00
135 25058 mn58 2.311e-01 2.869e+00 0.020 beta-(0) 1.000e+00
136 25066 mn66 1.083e+01 8.447e+00 0.489 beta-(0) 8.912e-01
beta-,neutron(0) 1.088e-01
137 25067 mn67 1.475e+01 8.565e+00 0.500 beta-(0) 1.000e+00
138 25068 mn68 2.476e+01 8.350e+00 0.384 beta-(0) 7.469e-01
beta-,neutron(0) 2.531e-01
139 25069 mn69 4.951e+01 9.346e+00 0.500 beta-(0) 1.000e+00
140 26054 fe54 0.000e+00 0.000e+00 1.000
141 26055 fe55 8.005e-09 5.842e-03 0.281 beta+(0) 1.000e+00
142 26056 fe56 0.000e+00 0.000e+00 0.281
143 26057 fe57 0.000e+00 0.000e+00 0.281
144 26058 fe58 0.000e+00 0.000e+00 0.281
145 26059 fe59 1.803e-07 1.306e+00 0.910 beta-(0) 1.000e+00
146 26060 fe60 1.464e-14 5.018e-02 0.000 beta-(1) 1.000e+00
147 26065 fe65 8.557e-01 4.742e+00 0.359 beta-(0) 1.000e+00
148 26066 fe66 1.575e+00 4.227e+00 0.500 beta-(0) 1.000e+00
149 26067 fe67 1.666e+00 6.420e+00 0.500 beta-(0) 1.000e+00
150 26068 fe68 3.707e+00 5.856e+00 0.500 beta-(0) 1.000e+00
151 26069 fe69 6.359e+00 7.737e+00 0.500 beta-(0) 1.000e+00
152 26070 fe70 7.374e+00 6.223e+00 0.500 beta-(0) 1.000e+00
153 26071 fe71 2.476e+01 8.582e+00 0.500 beta-(0) 1.000e+00
154 26072 fe72 6.931e+02 5.658e+00 0.495 beta-(0) 7.240e-01
beta-,neutron(0) 2.760e-01
155 27055 co55 1.098e-05 2.430e+00 0.822 beta+(0) 1.000e+00
156 27056 co56 1.039e-07 3.761e+00 0.968 beta+(0) 1.000e+00
157 27057 co57 2.952e-08 1.438e-01 0.870 beta+(0) 1.000e+00
158 27058 co58 1.132e-07 1.009e+00 0.966 beta+(0) 1.000e+00
159 1027058 co58m 2.116e-05 2.474e-02 0.078 i.t.(0) 1.000e+00
160 27059 co59 0.000e+00 0.000e+00 0.078
161 27060 co60 4.167e-09 2.601e+00 0.963 beta-(0) 1.000e+00
162 1027060 co60m 1.104e-03 6.305e-02 0.105 beta-(0) 2.400e-03 i.t.(0)
9.976e-01
163 27061 co61 1.167e-04 5.639e-01 0.172 beta-(0) 1.000e+00
164 27062 co62 7.701e-03 3.237e+00 0.494 beta-(0) 1.000e+00
165 27065 co65 5.975e-01 2.772e+00 0.033 beta-(0) 1.000e+00
166 27066 co66 3.466e+00 5.890e+00 0.416 beta-(0) 1.000e+00
167 27067 co67 1.631e+00 4.460e+00 0.176 beta-(0) 1.000e+00
168 27068 co68 3.483e+00 7.497e+00 0.459 beta-(0) 1.000e+00
169 27069 co69 3.151e+00 6.651e+00 0.500 beta-(0) 1.000e+00
170 27070 co70 5.825e+00 9.047e+00 0.500 beta-(0) 1.000e+00
171 27071 co71 8.774e+00 7.629e+00 0.497 beta-(0) 9.739e-01
beta-,neutron(0) 2.610e-02
172 27072 co72 1.157e+01 8.692e+00 0.431 beta-(0) 9.420e-01
```

(continues on next page)

(continued from previous page)

```
beta-,neutron(0) 5.801e-02
173 27073 co73 1.691e+01 7.018e+00 0.261 beta-(0) 9.468e-01
beta-,neutron(0) 5.321e-02
174 27074 co74 2.311e+01 9.385e+00 0.411 beta-(0) 9.073e-01
beta-,neutron(0) 9.274e-02
175 27075 co75 2.039e+01 8.079e+00 0.271 beta-(0) 9.284e-01
beta-,neutron(0) 7.159e-02
176 28056 ni56 1.321e-06 1.728e+00 0.996 beta+(0) 1.000e+00
177 28057 ni57 5.409e-06 2.093e+00 0.925 beta+(0) 1.000e+00
178 28058 ni58 0.000e+00 0.000e+00 0.925
179 28059 ni59 2.890e-13 6.916e-03 0.338 beta+(0) 1.000e+00
180 28060 ni60 0.000e+00 0.000e+00 0.338
181 28061 ni61 0.000e+00 0.000e+00 0.338
182 28062 ni62 0.000e+00 0.000e+00 0.338
183 28063 ni63 2.170e-10 1.742e-02 0.000 beta-(0) 1.000e+00
184 28064 ni64 0.000e+00 0.000e+00 0.000
185 28065 ni65 7.649e-05 1.186e+00 0.471 beta-(0) 1.000e+00
186 28066 ni66 3.526e-06 7.333e-02 0.000 beta-(0) 1.000e+00
187 28067 ni67 3.301e-02 1.584e+00 0.028 beta-(0) 1.000e+00
188 28068 ni68 2.390e-02 1.402e+00 0.500 beta-(0) 1.000e+00
189 28069 ni69 6.080e-02 4.000e+00 0.659 beta-(0) 1.000e+00
190 28070 ni70 1.155e-01 2.509e+00 0.500 beta-(0) 1.000e+00
191 28071 ni71 2.708e-01 4.870e+00 0.500 beta-(0) 1.000e+00
192 28072 ni72 4.415e-01 3.423e+00 0.509 beta-(0) 1.000e+00
beta-,neutron(0) 1.346e-07
193 28073 ni73 8.252e-01 6.410e+00 0.694 beta-(0) 9.943e-01
beta-,neutron(0) 5.677e-03
194 28074 ni74 1.019e+00 4.630e+00 0.537 beta-(0) 9.859e-01
beta-,neutron(0) 1.406e-02
195 28075 ni75 1.155e+00 6.716e+00 0.632 beta-(0) 8.814e-01
beta-,neutron(0) 1.186e-01
196 28076 ni76 2.912e+00 5.093e+00 0.443 beta-(0) 7.584e-01
beta-,neutron(0) 2.416e-01
197 28077 ni77 1.136e+01 5.543e+00 0.455 beta-(0) 4.465e-01
beta-,neutron(0) 5.535e-01
198 28078 ni78 6.301e+00 5.438e+00 0.386 beta-(0) 6.650e-01
beta-,neutron(0) 3.350e-01
199 29062 cu62 1.194e-03 2.298e+00 0.438 beta+(0) 1.000e+00
200 29063 cu63 0.000e+00 0.000e+00 0.438
201 29064 cu64 1.516e-05 3.119e-01 0.601 beta-(0) 3.850e-01 beta+(0)
6.150e-01
202 29065 cu65 0.000e+00 0.000e+00 0.601
203 29066 cu66 2.256e-03 1.165e+00 0.084 beta-(0) 1.000e+00
204 29067 cu67 3.114e-06 2.710e-01 0.426 beta-(0) 1.000e+00
205 29068 cu68 2.229e-02 2.492e+00 0.410 beta-(0) 1.000e+00
206 1029068 cu68m 3.081e-03 1.157e+00 0.762 beta-(0) 1.600e-01 i.t.(0)
8.400e-01
207 29069 cu69 4.053e-03 1.414e+00 0.371 beta-(0) 1.000e+00
208 29070 cu70 1.558e-02 4.820e+00 0.725 beta-(0) 1.000e+00
209 1029070 cu70m 2.100e-02 2.368e+00 0.510 beta-(0) 5.200e-01 i.t.(0)
4.800e-01
210 29071 cu71 3.555e-02 3.079e+00 0.500 beta-(0) 1.000e+00
211 29072 cu72 1.045e-01 5.312e+00 0.530 beta-(0) 1.000e+00
212 29073 cu73 1.650e-01 3.192e+00 0.096 beta-(0) 9.997e-01
beta-,neutron(0) 2.900e-04
213 29074 cu74 3.961e-01 5.640e+00 0.370 beta-(0) 9.992e-01
beta-,neutron(0) 7.500e-04
214 29075 cu75 5.663e-01 4.718e+00 0.425 beta-(0) 9.650e-01
beta-,neutron(0) 3.500e-02
215 29076 cu76 1.061e+00 6.076e+00 0.274 beta-(0) 9.700e-01
beta-,neutron(0) 3.000e-02
216 29077 cu77 1.478e+00 5.866e+00 0.414 beta-(0) 8.250e-01
beta-,neutron(0) 1.750e-01
217 29078 cu78 2.069e+00 6.863e+00 0.507 beta-(0) 6.294e-01
beta-,neutron(0) 3.706e-01
```

(continues on next page)

(continued from previous page)

```
218 29079 cu79 3.687e+00 5.514e+00 0.293 beta-(0) 4.500e-01
beta-,neutron(0) 5.500e-01
219 29080 cu80 4.077e+00 7.273e+00 0.413 beta-(0) 4.212e-01
beta-,neutron(0) 5.788e-01
220 30063 zn63 3.003e-04 2.016e+00 0.544 beta+(0) 1.000e+00
221 30064 zn64 0.000e+00 0.000e+00 0.544
222 30065 zn65 3.289e-08 5.828e-01 0.988 beta+(0) 1.000e+00
223 30066 zn66 0.000e+00 0.000e+00 0.988
224 30067 zn67 0.000e+00 0.000e+00 0.988
225 30068 zn68 0.000e+00 0.000e+00 0.988
226 30069 zn69 2.048e-04 3.216e-01 0.000 beta-(0) 1.000e+00
227 1030069 zn69m 1.399e-05 4.384e-01 0.949 beta-(0) 3.300e-04 i.t.(0)
9.997e-01
228 30070 zn70 0.000e+00 4.992e-01 0.000
229 30071 zn71 4.715e-03 1.359e+00 0.232 beta-(0) 1.000e+00
230 1030071 zn71m 4.862e-05 2.070e+00 0.743 beta-(0) 1.000e+00
231 30072 zn72 4.141e-06 2.452e-01 0.609 beta-(0) 9.647e-01 beta-(1)
3.526e-02
232 30073 zn73 2.950e-02 1.875e+00 0.063 beta-(0) 1.000e+00
233 30074 zn74 7.250e-03 1.392e+00 0.595 beta-(0) 1.000e+00
234 30075 zn75 6.796e-02 3.723e+00 0.489 beta-(0) 1.000e+00
235 30076 zn76 1.216e-01 2.032e+00 0.257 beta-(0) 1.000e+00
236 30077 zn77 3.332e-01 4.388e+00 0.442 beta-(0) 1.000e+00
237 30078 zn78 4.715e-01 3.738e+00 0.409 beta-(0) 1.000e+00
238 30079 zn79 6.966e-01 6.552e+00 0.695 beta-(0) 9.870e-01
beta-,neutron(0) 1.300e-02
239 30080 zn80 1.284e+00 5.273e+00 0.660 beta-(0) 9.900e-01
beta-,neutron(0) 1.000e-02
240 30081 zn81 2.166e+00 7.093e+00 0.523 beta-(0) 9.250e-01
beta-,neutron(0) 7.500e-02
241 30082 zn82 1.333e+01 5.326e+00 0.308 beta-(0) 6.588e-01
beta-,neutron(0) 3.412e-01
242 30083 zn83 1.612e+01 6.737e+00 0.271 beta-(0) 9.186e-01
beta-,neutron(0) 8.141e-02
243 31066 ga66 2.029e-05 3.467e+00 0.713 beta+(0) 1.000e+00
244 31067 ga67 2.460e-06 1.946e-01 0.815 beta+(0) 1.000e+00
245 31068 ga68 1.706e-04 1.691e+00 0.562 beta+(0) 1.000e+00
246 31069 ga69 0.000e+00 0.000e+00 0.562
247 31070 ga70 5.465e-04 6.513e-01 0.011 beta-(0) 9.959e-01 beta+(0)
4.100e-03
248 31071 ga71 0.000e+00 0.000e+00 0.011
249 31072 ga72 1.366e-05 3.246e+00 0.853 beta-(0) 1.000e+00
250 1031072 ga72m 1.747e+01 1.197e-01 1.000 i.t.(0) 1.000e+00
251 31073 ga73 3.962e-05 7.917e-01 0.431 beta-(0) 1.477e-02 beta-(1)
9.852e-01
252 31074 ga74 1.423e-03 4.024e+00 0.754 beta-(0) 1.000e+00
253 1031074 ga74m 7.296e-02 6.468e-01 0.049 beta-(0) 2.500e-01 i.t.(0)
7.500e-01
254 31075 ga75 5.501e-03 1.690e+00 0.196 beta-(0) 9.632e-01 beta-(1)
3.676e-02
255 31076 ga76 2.126e-02 4.673e+00 0.599 beta-(0) 1.000e+00
256 31077 ga77 5.251e-02 2.764e+00 0.286 beta-(0) 1.000e+00
257 31078 ga78 1.362e-01 5.122e+00 0.496 beta-(0) 1.000e+00
258 31079 ga79 2.435e-01 4.487e+00 0.464 beta-(0) 9.495e-01 beta-(1)
4.966e-02 beta-,neutron(0) 8.900e-04
259 31080 ga80 4.136e-01 6.183e+00 0.417 beta-(0) 9.914e-01
beta-,neutron(0) 8.600e-03
260 31081 ga81 5.696e-01 4.625e+00 0.480 beta-(0) 4.662e-01 beta-(1)
4.148e-01 beta-,neutron(0) 1.190e-01
261 31082 ga82 1.157e+00 8.025e+00 0.599 beta-(0) 8.907e-01
beta-,neutron(0) 1.094e-01
262 31083 ga83 2.250e+00 5.217e+00 0.256 beta-(0) 2.567e-01
beta-,neutron(0) 7.433e-01
263 31084 ga84 8.155e+00 8.038e+00 0.433 beta-(0) 8.402e-01
beta-,neutron(0) 1.598e-01
```

(continues on next page)

(continued from previous page)

```
264 31085 ga85 1.444e+01 5.495e+00 0.126 beta-,neutron(0) 1.000e+00
265 31086 ga86 2.390e+01 6.603e+00 0.220 beta-(0) 2.375e-01
beta-,neutron(0) 7.625e-01
266 32066 ge66 8.519e-05 7.808e-01 0.875 beta+(0) 1.000e+00
267 32067 ge67 6.112e-04 2.814e+00 0.500 beta+(0) 1.000e+00
268 32068 ge68 2.961e-08 9.121e-03 0.451 beta+(0) 1.000e+00
269 32069 ge69 4.931e-06 1.071e+00 0.888 beta+(0) 1.000e+00
270 32070 ge70 0.000e+00 0.000e+00 0.888
271 32071 ge71 7.019e-07 9.092e-03 0.453 beta+(0) 1.000e+00
272 1032071 ge71m 3.396e+01 1.975e-01 0.832 i.t.(0) 1.000e+00
273 32072 ge72 0.000e+00 0.000e+00 0.832
274 32073 ge73 0.000e+00 0.000e+00 0.832
275 1032073 ge73m 1.389e+00 6.616e-02 0.171 i.t.(0) 1.000e+00
276 32074 ge74 0.000e+00 0.000e+00 0.171
277 32075 ge75 1.396e-04 4.556e-01 0.077 beta-(0) 1.000e+00
278 1032075 ge75m 1.453e-02 1.396e-01 0.415 beta-(0) 3.000e-04 i.t.(0)
9.997e-01
279 32076 ge76 0.000e+00 0.000e+00 0.415
280 32077 ge77 1.704e-05 1.729e+00 0.624 beta-(0) 1.000e+00
281 1032077 ge77m 1.310e-02 1.041e+00 0.071 beta-(0) 8.100e-01 i.t.(0)
1.900e-01
282 32078 ge78 1.313e-04 5.062e-01 0.549 beta-(0) 1.000e+00
283 32079 ge79 3.652e-02 2.023e+00 0.153 beta-(0) 1.000e+00
284 1032079 ge79m 1.777e-02 2.486e+00 0.481 beta-(0) 9.600e-01 i.t.(0)
4.000e-02
285 32080 ge80 2.350e-02 1.193e+00 0.123 beta-(0) 1.000e+00
286 32081 ge81 9.120e-02 4.257e+00 0.620 beta-(0) 1.000e+00
287 1032081 ge81m 9.120e-02 3.982e+00 0.364 beta-(0) 1.000e+00
288 32082 ge82 1.520e-01 3.550e+00 0.784 beta-(0) 1.000e+00
289 32083 ge83 3.747e-01 5.167e+00 0.421 beta-(0) 9.989e-01
beta-,neutron(0) 1.133e-03
290 32084 ge84 7.266e-01 4.122e+00 0.347 beta-(0) 8.980e-01
beta-,neutron(0) 1.020e-01
291 32085 ge85 1.296e+00 5.387e+00 0.248 beta-(0) 9.859e-01
beta-,neutron(0) 1.410e-02
292 32086 ge86 7.296e+00 5.023e+00 0.308 beta-(0) 9.480e-01
beta-,neutron(0) 5.196e-02
293 32087 ge87 4.951e+00 6.565e+00 0.376 beta-(0) 9.588e-01
beta-,neutron(0) 4.124e-02
294 32088 ge88 1.050e+01 5.797e+00 0.280 beta-(0) 9.516e-01
beta-,neutron(0) 4.838e-02
295 32089 ge89 1.777e+01 6.733e+00 0.270 beta-(0) 7.659e-01
beta-,neutron(0) 2.341e-01
296 33069 as69 7.585e-04 2.363e+00 0.484 beta+(0) 1.000e+00
297 33071 as71 2.949e-06 6.935e-01 0.832 beta+(0) 1.000e+00
298 33072 as72 7.406e-06 2.837e+00 0.634 beta+(0) 1.000e+00
299 33073 as73 9.991e-08 9.753e-03 0.479 beta+(1) 1.000e+00
300 33074 as74 4.515e-07 1.024e+00 0.741 beta-(0) 3.400e-01 beta+(0)
6.600e-01
301 33075 as75 0.000e+00 0.000e+00 0.741
302 1033075 as75m 3.934e+01 3.039e-01 0.933 i.t.(0) 1.000e+00
303 33076 as76 7.338e-06 1.482e+00 0.281 beta-(0) 1.000e+00
304 33077 as77 4.959e-06 2.337e-01 0.035 beta-(0) 9.967e-01 beta-(1)
3.329e-03
305 33078 as78 1.274e-04 2.565e+00 0.509 beta-(0) 1.000e+00
306 33079 as79 1.282e-03 8.730e-01 0.039 beta-(0) 2.369e-02 beta-(1)
9.763e-01
307 33080 as80 4.560e-02 2.730e+00 0.213 beta-(0) 1.000e+00
308 33081 as81 2.082e-02 1.805e+00 0.128 beta-(0) 9.641e-01 beta-(1)
3.589e-02
309 33082 as82 3.629e-02 3.613e+00 0.085 beta-(0) 1.000e+00
310 1033082 as82m 5.097e-02 4.936e+00 0.565 beta-(0) 1.000e+00
311 33083 as83 5.173e-02 3.761e+00 0.615 beta-(0) 1.000e+00
312 33084 as84 1.650e-01 6.351e+00 0.495 beta-(0) 9.982e-01
beta-,neutron(0) 1.800e-03
```

(continues on next page)

(continued from previous page)

```
313 33085 as85 3.430e-01 3.744e+00 0.280 beta-(0) 4.060e-01
beta-,neutron(0) 5.940e-01
314 33086 as86 7.335e-01 7.106e+00 0.553 beta-(0) 8.751e-01
beta-,neutron(0) 1.249e-01
315 33087 as87 1.238e+00 5.609e+00 0.424 beta-(0) 6.033e-01
beta-,neutron(0) 3.967e-01
316 33088 as88 6.189e+00 6.446e+00 0.389 beta-(0) 5.903e-01
beta-,neutron(0) 4.097e-01
317 33089 as89 1.175e+01 4.684e+00 0.084 beta-(0) 1.000e-05
beta-,neutron(0) 1.000e+00
318 33090 as90 1.612e+01 7.487e+00 0.358 beta-(0) 5.834e-01
beta-,neutron(0) 4.166e-01
319 33091 as91 1.575e+01 6.325e+00 0.196 beta-,neutron(0) 1.000e+00
320 33092 as92 2.567e+01 6.051e+00 0.141 beta-,neutron(0) 1.000e+00
321 34072 se72 9.551e-07 5.637e-02 0.600 beta+(0) 1.000e+00
322 34073 se73 2.693e-05 1.466e+00 0.741 beta+(0) 1.000e+00
323 1034073 se73m 2.903e-04 4.254e-01 0.617 beta+(0) 2.740e-01 i.t.(0)
7.260e-01
324 34074 se74 0.000e+00 0.000e+00 0.617
325 34075 se75 6.697e-08 4.032e-01 0.965 beta+(0) 1.000e+00
326 34076 se76 0.000e+00 0.000e+00 0.965
327 34077 se77 0.000e+00 0.000e+00 0.965
328 1034077 se77m 3.993e-02 1.620e-01 0.548 i.t.(0) 1.000e+00
329 34078 se78 0.000e+00 0.000e+00 0.548
330 34079 se79 7.446e-14 5.580e-02 0.000 beta-(0) 1.000e+00
331 1034079 se79m 2.947e-03 9.549e-02 0.144 beta-(0) 5.600e-04 i.t.(0)
9.994e-01
332 34080 se80 0.000e+00 6.628e-02 0.000
333 34081 se81 6.261e-04 6.189e-01 0.013 beta-(0) 1.000e+00
334 1034081 se81m 2.017e-04 1.029e-01 0.175 beta-(0) 5.100e-04 i.t.(0)
9.995e-01
335 34082 se82 0.000e+00 0.000e+00 0.175
336 34083 se83 5.180e-04 3.006e+00 0.844 beta-(0) 1.000e+00
337 1034083 se83m 9.888e-03 2.254e+00 0.436 beta-(0) 1.000e+00
338 34084 se84 3.544e-03 9.486e-01 0.430 beta-(0) 1.000e+00
339 34085 se85 2.187e-02 3.923e+00 0.524 beta-(0) 1.000e+00
340 34086 se86 4.847e-02 3.601e+00 0.631 beta-(0) 1.000e+00
341 34087 se87 1.260e-01 3.769e+00 0.176 beta-(0) 9.980e-01
beta-,neutron(0) 2.000e-03
342 34088 se88 4.530e-01 3.558e+00 0.238 beta-(0) 9.933e-01
beta-,neutron(0) 6.700e-03
343 34089 se89 1.691e+00 4.662e+00 0.214 beta-(0) 9.220e-01
beta-,neutron(0) 7.800e-02
344 34090 se90 4.305e+00 4.159e+00 0.199 beta-(0) 9.917e-01
beta-,neutron(0) 8.259e-03
345 34091 se91 2.567e+00 5.849e+00 0.268 beta-(0) 9.875e-01
beta-,neutron(0) 1.251e-02
346 34092 se92 7.453e+00 5.127e+00 0.265 beta-(0) 9.805e-01
beta-,neutron(0) 1.948e-02
347 34093 se93 1.118e+01 6.361e+00 0.387 beta-(0) 6.779e-01
beta-,neutron(0) 3.221e-01
348 34094 se94 1.175e+01 5.388e+00 0.219 beta-(0) 7.137e-01
beta-,neutron(0) 2.863e-01
349 35075 br75 1.195e-04 1.764e+00 0.689 beta+(0) 1.000e+00
350 35076 br76 1.189e-05 3.430e+00 0.810 beta+(0) 1.000e+00
351 35077 br77 3.376e-06 3.301e-01 0.972 beta+(0) 1.000e+00
352 1035077 br77m 2.699e-03 1.058e-01 0.184 i.t.(0) 1.000e+00
353 35078 br78 1.791e-03 2.058e+00 0.502 beta+(0) 1.000e+00
354 35079 br79 0.000e+00 0.000e+00 0.502
355 1035079 br79m 1.426e-01 2.073e-01 0.768 i.t.(0) 1.000e+00
356 35080 br80 6.534e-04 8.015e-01 0.094 beta-(0) 9.170e-01 beta+(0)
8.300e-02
357 1035080 br80m 4.356e-05 8.543e-02 0.283 i.t.(0) 1.000e+00
358 35081 br81 0.000e+00 0.000e+00 0.283
359 35082 br82 5.457e-06 2.806e+00 0.948 beta-(0) 1.000e+00
```

(continues on next page)

(continued from previous page)

```
360 1035082 br82m 1.885e-03 7.818e-02 0.103 beta-(0) 2.400e-02 i.t.(0)
9.760e-01
361 35083 br83 8.022e-05 3.371e-01 0.020 beta-(0) 2.403e-04 beta-(1)
9.998e-01
362 35084 br84 3.637e-04 2.987e+00 0.588 beta-(0) 1.000e+00
363 1035084 br84m 1.925e-03 3.655e+00 0.757 beta-(0) 1.000e+00
364 35085 br85 3.983e-03 1.106e+00 0.055 beta-(0) 1.626e-03 beta-(1)
9.984e-01
365 35086 br86 1.260e-02 5.241e+00 0.629 beta-(0) 1.000e+00
366 35087 br87 1.246e-02 5.007e+00 0.668 beta-(0) 9.740e-01
beta-,neutron(0) 2.600e-02
367 35088 br88 4.255e-02 5.543e+00 0.565 beta-(0) 9.342e-01
beta-,neutron(0) 6.580e-02
368 35089 br89 1.575e-01 4.262e+00 0.382 beta-(0) 8.620e-01
beta-,neutron(0) 1.380e-01
369 35090 br90 3.610e-01 4.837e+00 0.308 beta-(0) 7.480e-01
beta-,neutron(0) 2.520e-01
370 35091 br91 1.281e+00 4.502e+00 0.089 beta-(0) 8.000e-01
beta-,neutron(0) 2.000e-01
371 35092 br92 2.021e+00 6.976e+00 0.380 beta-(0) 8.644e-01
beta-,neutron(0) 1.356e-01
372 35093 br93 6.796e+00 5.392e+00 0.158 beta-(0) 7.338e-01
beta-,neutron(0) 2.662e-01
373 35094 br94 9.902e+00 5.164e+00 0.240 beta-(0) 1.309e-01
beta-,neutron(0) 8.691e-01
374 35095 br95 1.050e+01 5.181e+00 0.154 beta-(0) 1.670e-03
beta-,neutron(0) 9.983e-01
375 35096 br96 1.650e+01 6.836e+00 0.302 beta-(0) 4.227e-01
beta-,neutron(0) 5.773e-01
376 35097 br97 1.733e+01 6.143e+00 0.252 beta-,neutron(0) 1.000e+00
377 36076 kr76 1.301e-05 4.240e-01 0.964 beta+(0) 1.000e+00
378 36077 kr77 1.553e-04 1.722e+00 0.604 beta+(0) 1.000e+00
379 36078 kr78 0.000e+00 0.000e+00 0.604
380 36079 kr79 5.495e-06 2.786e-01 0.915 beta+(0) 1.000e+00
381 1036079 kr79m 1.386e-02 1.302e-01 0.313 i.t.(0) 1.000e+00
382 36080 kr80 0.000e+00 0.000e+00 0.313
383 36081 kr81 9.592e-14 1.206e-02 0.588 beta+(0) 1.000e+00
```

10.2.14 REACTION RESOURCE CONTENTS

A concise representation of the data contained in the reaction resource is shown below in Example 10.2.5. This type of output may be generated from any AMPX multigroup library using the PALEALE utility.

Example 10.2.5: Contents of reaction resource.

```
JEFF-3.0/A Neutron Activation File
J-Ch Sublet, A J Koning, R A Forrest and J Kopecky March 2003
238-group Library developed by I C Gauld and D Wiarda
Updated: 01/11/11
1001 h-1 h1 125 jeff-31 rel0 rev31 mod0
1002 h-2 h2 128 jeff-31 rel0 rev31 mod0
1003 h-3 h3 131 jeff-31 rel0 rev31 mod0
2003 he-3 he3 225 jeff-31 rel0 rev31 mod0
3006 li-6 li6 325 jeff-31 rel0 rev31 mod0
3007 li-7 li7 328 jeff-31 rel0 rev31 mod0
4007 be-7 be7 419 jeff-31 rel0 rev31 mod0
4009 be-9 be9 425 jeff-31 rel0 rev31 mod0
4010 be-10 be10 428 jeff-31 rel0 rev31 mod0
5010 b-10 b10 525 jeff-31 rel0 rev31 mod0
5011 b-11 b11 528 jeff-31 rel0 rev31 mod0
6012 c-12 c12 625 jeff-31 rel0 rev31 mod0
6013 c-13 c13 628 jeff-31 rel0 rev31 mod0
6014 c-14 c14 631 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
7014 n-14 n14 725 jeff-31 rel0 rev31 mod0
7015 n-15 n15 728 jeff-31 rel0 rev31 mod0
8016 o-16 o16 825 jeff-31 rel0 rev31 mod0
8017 o-17 o17 828 jeff-31 rel0 rev31 mod0
8018 o-18 o18 831 jeff-31 rel0 rev31 mod0
9019 f-19 f19 925 jeff-31 rel0 rev31 mod0
10020 ne-20 ne20 1025 jeff-31 rel0 rev31 mod0
10021 ne-21 ne21 1028 jeff-31 rel0 rev31 mod0
10022 ne-22 ne22 1031 jeff-31 rel0 rev31 mod0
11022 na-22 na22 1122 jeff-31 rel0 rev31 mod0
11023 na-23 na23 1125 jeff-31 rel0 rev31 mod0
11024 na-24 na24 1128 jeff-31 rel0 rev31 mod0
12024 mg-24 mg24 1225 jeff-31 rel0 rev31 mod0
12025 mg-25 mg25 1228 jeff-31 rel0 rev31 mod0
12026 mg-26 mg26 1231 jeff-31 rel0 rev31 mod0
12028 mg-28 mg28 1237 jeff-31 rel0 rev31 mod0
13026 al-26 al26 1322 jeff-31 rel0 rev31 mod0
13027 al-27 al27 1325 jeff-31 rel0 rev31 mod0
14028 si-28 si28 1425 jeff-31 rel0 rev31 mod0
14029 si-29 si29 1428 jeff-31 rel0 rev31 mod0
14030 si-30 si30 1431 jeff-31 rel0 rev31 mod0
14031 si-31 si31 1434 jeff-31 rel0 rev31 mod0
14032 si-32 si32 1437 jeff-31 rel0 rev31 mod0
15031 p-31 p31 1525 jeff-31 rel0 rev31 mod0
15032 p-32 p32 1528 jeff-31 rel0 rev31 mod0
15033 p-33 p33 1531 jeff-31 rel0 rev31 mod0
16032 s-32 s32 1625 jeff-31 rel0 rev31 mod0
16033 s-33 s33 1628 jeff-31 rel0 rev31 mod0
16034 s-34 s34 1631 jeff-31 rel0 rev31 mod0
16035 s-35 s35 1634 jeff-31 rel0 rev31 mod0
16036 s-36 s36 1637 jeff-31 rel0 rev31 mod0
17035 cl-35 cl35 1725 jeff-31 rel0 rev31 mod0
17036 cl-36 cl36 1728 jeff-31 rel0 rev31 mod0
17037 cl-37 cl37 1731 jeff-31 rel0 rev31 mod0
18036 ar-36 ar36 1825 jeff-31 rel0 rev31 mod0
18037 ar-37 ar37 1828 jeff-31 rel0 rev31 mod0
18038 ar-38 ar38 1831 jeff-31 rel0 rev31 mod0
18039 ar-39 ar39 1834 jeff-31 rel0 rev31 mod0
18040 ar-40 ar40 1837 jeff-31 rel0 rev31 mod0
18041 ar-41 ar41 1840 jeff-31 rel0 rev31 mod0
18042 ar-42 ar42 1843 jeff-31 rel0 rev31 mod0
19039 k-39 k39 1925 jeff-31 rel0 rev31 mod0
19040 k-40 k40 1928 jeff-31 rel0 rev31 mod0
19041 k-41 k41 1931 jeff-31 rel0 rev31 mod0
19042 k-42 k42 1934 jeff-31 rel0 rev31 mod0
19043 k-43 k43 1937 jeff-31 rel0 rev31 mod0
20040 ca-40 ca40 2025 jeff-31 rel0 rev31 mod0
20041 ca-41 ca41 2028 jeff-31 rel0 rev31 mod0
20042 ca-42 ca42 2031 jeff-31 rel0 rev31 mod0
20043 ca-43 ca43 2034 jeff-31 rel0 rev31 mod0
20044 ca-44 ca44 2037 jeff-31 rel0 rev31 mod0
20045 ca-45 ca45 2040 jeff-31 rel0 rev31 mod0
20046 ca-46 ca46 2043 jeff-31 rel0 rev31 mod0
20047 ca-47 ca47 2046 jeff-31 rel0 rev31 mod0
20048 ca-48 ca48 2049 jeff-31 rel0 rev31 mod0
21045 sc-45 sc45 2125 jeff-31 rel0 rev31 mod0
21046 sc-46 sc46 2128 jeff-31 rel0 rev31 mod0
21047 sc-47 sc47 2131 jeff-31 rel0 rev31 mod0
21048 sc-48 sc48 2134 jeff-31 rel0 rev31 mod0
21601 21601 sc44m1 2123 jeff-31 rel0 rev31 mod0
22044 ti-44 ti44 2219 jeff-31 rel0 rev31 mod0
22045 ti-45 ti45 2222 jeff-31 rel0 rev31 mod0
22046 ti-46 ti46 2225 jeff-31 rel0 rev31 mod0
22047 ti-47 ti47 2228 jeff-31 rel0 rev31 mod0
22048 ti-48 ti48 2231 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
22049 ti-49 ti49 2234 jeff-31 rel0 rev31 mod0
22050 ti-50 ti50 2237 jeff-31 rel0 rev31 mod0
23048 v-48 v48 2319 jeff-31 rel0 rev31 mod0
23049 v-49 v49 2322 jeff-31 rel0 rev31 mod0
23050 v-50 v50 2325 jeff-31 rel0 rev31 mod0
23051 v-51 v51 2328 jeff-31 rel0 rev31 mod0
24048 cr-48 cr48 2419 jeff-31 rel0 rev31 mod0
24050 cr-50 cr50 2425 jeff-31 rel0 rev31 mod0
24051 cr-51 cr51 2428 jeff-31 rel0 rev31 mod0
24052 cr-52 cr52 2431 jeff-31 rel0 rev31 mod0
24053 cr-53 cr53 2434 jeff-31 rel0 rev31 mod0
24054 cr-54 cr54 2437 jeff-31 rel0 rev31 mod0
25052 mn-52 mn52 2516 jeff-31 rel0 rev31 mod0
25053 mn-53 mn53 2519 jeff-31 rel0 rev31 mod0
25054 mn-54 mn54 2522 jeff-31 rel0 rev31 mod0
25055 mn-55 mn55 2525 jeff-31 rel0 rev31 mod0
26054 fe-54 fe54 2625 jeff-31 rel0 rev31 mod0
26055 fe-55 fe55 2628 jeff-31 rel0 rev31 mod0
26056 fe-56 fe56 2631 jeff-31 rel0 rev31 mod0
26057 fe-57 fe57 2634 jeff-31 rel0 rev31 mod0
26058 fe-58 fe58 2637 jeff-31 rel0 rev31 mod0
26059 fe-59 fe59 2640 jeff-31 rel0 rev31 mod0
26060 fe-60 fe60 2643 jeff-31 rel0 rev31 mod0
27055 co-55 co55 2713 jeff-31 rel0 rev31 mod0
27056 co-56 co56 2716 jeff-31 rel0 rev31 mod0
27057 co-57 co57 2719 jeff-31 rel0 rev31 mod0
27058 co-58 co58 2722 jeff-31 rel0 rev31 mod0
27059 co-59 co59 2725 jeff-31 rel0 rev31 mod0
27060 co-60 co60 2728 jeff-31 rel0 rev31 mod0
27601 co-58m co58m1 2723 jeff-31 rel0 rev31 mod0
28056 ni-56 ni56 2819 jeff-31 rel0 rev31 mod0
28057 ni-57 ni57 2822 jeff-31 rel0 rev31 mod0
28058 ni-58 ni58 2825 jeff-31 rel0 rev31 mod0
28059 ni-59 ni59 2828 jeff-31 rel0 rev31 mod0
28060 ni-60 ni60 2831 jeff-31 rel0 rev31 mod0
28061 ni-61 ni61 2834 jeff-31 rel0 rev31 mod0
28062 ni-62 ni62 2837 jeff-31 rel0 rev31 mod0
28063 ni-63 ni63 2840 jeff-31 rel0 rev31 mod0
28064 ni-64 ni64 2843 jeff-31 rel0 rev31 mod0
28066 ni-66 ni66 2849 jeff-31 rel0 rev31 mod0
29063 cu-63 cu63 2925 jeff-31 rel0 rev31 mod0
29064 cu-64 cu64 2928 jeff-31 rel0 rev31 mod0
29065 cu-65 cu65 2931 jeff-31 rel0 rev31 mod0
29067 cu-67 cu67 2937 jeff-31 rel0 rev31 mod0
30064 zn-64 zn64 3025 jeff-31 rel0 rev31 mod0
30065 zn-65 zn65 3028 jeff-31 rel0 rev31 mod0
30066 zn-66 zn66 3031 jeff-31 rel0 rev31 mod0
30067 zn-67 zn67 3034 jeff-31 rel0 rev31 mod0
30068 zn-68 zn68 3037 jeff-31 rel0 rev31 mod0
30070 zn-70 zn70 3043 jeff-31 rel0 rev31 mod0
30072 zn-72 zn72 3049 jeff-31 rel0 rev31 mod0
30601 30601 zn69m1 3041 jeff-31 rel0 rev31 mod0
31067 ga-67 ga67 3119 jeff-31 rel0 rev31 mod0
31069 ga-69 ga69 3125 jeff-31 rel0 rev31 mod0
31071 ga-71 ga71 3131 jeff-31 rel0 rev31 mod0
31072 ga-72 ga72 3134 jeff-31 rel0 rev31 mod0
32068 ge-68 ge68 3219 jeff-31 rel0 rev31 mod0
32069 ge-69 ge69 3222 jeff-31 rel0 rev31 mod0
32070 ge-70 ge70 3225 jeff-31 rel0 rev31 mod0
32071 ge-71 ge71 3228 jeff-31 rel0 rev31 mod0
32072 ge-72 ge72 3231 jeff-31 rel0 rev31 mod0
32073 ge-73 ge73 3234 jeff-31 rel0 rev31 mod0
32074 ge-74 ge74 3237 jeff-31 rel0 rev31 mod0
32076 ge-76 ge76 3243 jeff-31 rel0 rev31 mod0
32077 ge-77 ge77 3246 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
33071 as-71 as71 3313 jeff-31 rel0 rev31 mod0
33072 as-72 as72 3316 jeff-31 rel0 rev31 mod0
33073 as-73 as73 3319 jeff-31 rel0 rev31 mod0
33074 as-74 as74 3322 jeff-31 rel0 rev31 mod0
33075 as-75 as75 3325 jeff-31 rel0 rev31 mod0
33076 as-76 as76 3328 jeff-31 rel0 rev31 mod0
33077 as-77 as77 3331 jeff-31 rel0 rev31 mod0
34072 se-72 se72 3419 jeff-31 rel0 rev31 mod0
34073 se-73 se73 3422 jeff-31 rel0 rev31 mod0
34074 se-74 se74 3425 jeff-31 rel0 rev31 mod0
34075 se-75 se75 3428 jeff-31 rel0 rev31 mod0
34076 se-76 se76 3431 jeff-31 rel0 rev31 mod0
34077 se-77 se77 3434 jeff-31 rel0 rev31 mod0
34078 se-78 se78 3437 jeff-31 rel0 rev31 mod0
34079 se-79 se79 3440 jeff-31 rel0 rev31 mod0
34080 se-80 se80 3443 jeff-31 rel0 rev31 mod0
34082 se-82 se82 3449 jeff-31 rel0 rev31 mod0
35076 br-76 br76 3516 jeff-31 rel0 rev31 mod0
35077 br-77 br77 3519 jeff-31 rel0 rev31 mod0
35079 br-79 br79 3525 jeff-31 rel0 rev31 mod0
35081 br-81 br81 3531 jeff-31 rel0 rev31 mod0
35082 br-82 br82 3534 jeff-31 rel0 rev31 mod0
36076 kr-76 kr76 3619 jeff-31 rel0 rev31 mod0
36078 kr-78 kr78 3625 jeff-31 rel0 rev31 mod0
36079 kr-79 kr79 3628 jeff-31 rel0 rev31 mod0
36080 kr-80 kr80 3631 jeff-31 rel0 rev31 mod0
36081 kr-81 kr81 3634 jeff-31 rel0 rev31 mod0
36082 kr-82 kr82 3637 jeff-31 rel0 rev31 mod0
36083 kr-83 kr83 3640 jeff-31 rel0 rev31 mod0
36084 kr-84 kr84 3643 jeff-31 rel0 rev31 mod0
36085 kr-85 kr85 3646 jeff-31 rel0 rev31 mod0
36086 kr-86 kr86 3649 jeff-31 rel0 rev31 mod0
37083 rb-83 rb83 3719 jeff-31 rel0 rev31 mod0
37084 rb-84 rb84 3722 jeff-31 rel0 rev31 mod0
37085 rb-85 rb85 3725 jeff-31 rel0 rev31 mod0
37086 rb-86 rb86 3728 jeff-31 rel0 rev31 mod0
37087 rb-87 rb87 3731 jeff-31 rel0 rev31 mod0
38082 sr-82 sr82 3819 jeff-31 rel0 rev31 mod0
38083 sr-83 sr83 3822 jeff-31 rel0 rev31 mod0
38084 sr-84 sr84 3825 jeff-31 rel0 rev31 mod0
38085 sr-85 sr85 3828 jeff-31 rel0 rev31 mod0
38086 sr-86 sr86 3831 jeff-31 rel0 rev31 mod0
38087 sr-87 sr87 3834 jeff-31 rel0 rev31 mod0
38088 sr-88 sr88 3837 jeff-31 rel0 rev31 mod0
38089 sr-89 sr89 3840 jeff-31 rel0 rev31 mod0
38090 sr-90 sr90 3843 jeff-31 rel0 rev31 mod0
39086 y-86 y86 3916 jeff-31 rel0 rev31 mod0
39087 y-87 y87 3919 jeff-31 rel0 rev31 mod0
39088 y-88 y88 3922 jeff-31 rel0 rev31 mod0
39089 y-89 y89 3925 jeff-31 rel0 rev31 mod0
39090 y-90 y90 3928 jeff-31 rel0 rev31 mod0
39091 y-91 y91 3931 jeff-31 rel0 rev31 mod0
39601 39601 y87m1 3920 jeff-31 rel0 rev31 mod0
40086 zr-86 zr86 4013 jeff-31 rel0 rev31 mod0
40088 zr-88 zr88 4019 jeff-31 rel0 rev31 mod0
40089 zr-89 zr89 4022 jeff-31 rel0 rev31 mod0
40090 zr-90 zr90 4025 jeff-31 rel0 rev31 mod0
40091 zr-91 zr91 4028 jeff-31 rel0 rev31 mod0
40092 zr-92 zr92 4031 jeff-31 rel0 rev31 mod0
40093 zr-93 zr93 4034 jeff-31 rel0 rev31 mod0
40094 zr-94 zr94 4037 jeff-31 rel0 rev31 mod0
40095 zr-95 zr95 4040 jeff-31 rel0 rev31 mod0
40096 zr-96 zr96 4043 jeff-31 rel0 rev31 mod0
40097 zr-97 zr97 4046 jeff-31 rel0 rev31 mod0
41090 nb-90 nb90 4116 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
41091 nb-91 nb91 4119 jeff-31 rel0 rev31 mod0
41092 nb-92 nb92 4122 jeff-31 rel0 rev31 mod0
41093 nb-93 nb93 4125 jeff-31 rel0 rev31 mod0
41094 nb-94 nb94 4128 jeff-31 rel0 rev31 mod0
41095 nb-95 nb95 4131 jeff-31 rel0 rev31 mod0
41096 nb-96 nb96 4134 jeff-31 rel0 rev31 mod0
41601 41601 nb91m1 4120 jeff-31 rel0 rev31 mod0
41611 41611 nb92m1 4123 jeff-31 rel0 rev31 mod0
41621 41621 nb93m1 4126 jeff-31 rel0 rev31 mod0
41631 41631 nb95m1 4132 jeff-31 rel0 rev31 mod0
42092 mo-92 mo92 4225 jeff-31 rel0 rev31 mod0
42093 mo-93 mo93 4228 jeff-31 rel0 rev31 mod0
42094 mo-94 mo94 4231 jeff-31 rel0 rev31 mod0
42095 mo-95 mo95 4234 jeff-31 rel0 rev31 mod0
42096 mo-96 mo96 4237 jeff-31 rel0 rev31 mod0
42097 mo-97 mo97 4240 jeff-31 rel0 rev31 mod0
42098 mo-98 mo98 4243 jeff-31 rel0 rev31 mod0
42099 mo-99 mo99 4246 jeff-31 rel0 rev31 mod0
42100 mo-100 mo100 4249 jeff-31 rel0 rev31 mod0
43095 tc-95 tc95 4313 jeff-31 rel0 rev31 mod0
43096 tc-96 tc96 4316 jeff-31 rel0 rev31 mod0
43097 tc-97 tc97 4319 jeff-31 rel0 rev31 mod0
43098 tc-98 tc98 4322 jeff-31 rel0 rev31 mod0
43099 tc-99 tc99 4325 jeff-31 rel0 rev31 mod0
43601 43601 tc95m1 4314 jeff-31 rel0 rev31 mod0
43611 43611 tc97m1 4320 jeff-31 rel0 rev31 mod0
44096 ru-96 ru96 4425 jeff-31 rel0 rev31 mod0
44097 ru-97 ru97 4428 jeff-31 rel0 rev31 mod0
44098 ru-98 ru98 4431 jeff-31 rel0 rev31 mod0
44099 ru-99 ru99 4434 jeff-31 rel0 rev31 mod0
44100 ru-100 ru100 4437 jeff-31 rel0 rev31 mod0
44101 ru-101 ru101 4440 jeff-31 rel0 rev31 mod0
44102 ru-102 ru102 4443 jeff-31 rel0 rev31 mod0
44103 ru-103 ru103 4446 jeff-31 rel0 rev31 mod0
44104 ru-104 ru104 4449 jeff-31 rel0 rev31 mod0
44105 ru-105 ru105 4452 jeff-31 rel0 rev31 mod0
44106 ru-106 ru106 4455 jeff-31 rel0 rev31 mod0
45099 rh-99 rh99 4513 jeff-31 rel0 rev31 mod0
45100 rh-100 rh100 4516 jeff-31 rel0 rev31 mod0
45101 rh-101 rh101 4519 jeff-31 rel0 rev31 mod0
45102 rh-102 rh102 4522 jeff-31 rel0 rev31 mod0
45103 rh-103 rh103 4525 jeff-31 rel0 rev31 mod0
45105 rh-105 rh105 4531 jeff-31 rel0 rev31 mod0
45601 45601 rh99m1 4514 jeff-31 rel0 rev31 mod0
45611 45611 rh101m1 4520 jeff-31 rel0 rev31 mod0
45621 45621 rh102m1 4523 jeff-31 rel0 rev31 mod0
46100 pd-100 pd100 4619 jeff-31 rel0 rev31 mod0
46101 pd-101 pd101 4622 jeff-31 rel0 rev31 mod0
46102 pd-102 pd102 4625 jeff-31 rel0 rev31 mod0
46103 pd-103 pd103 4628 jeff-31 rel0 rev31 mod0
46104 pd-104 pd104 4631 jeff-31 rel0 rev31 mod0
46105 pd-105 pd105 4634 jeff-31 rel0 rev31 mod0
46106 pd-106 pd106 4637 jeff-31 rel0 rev31 mod0
46107 pd-107 pd107 4640 jeff-31 rel0 rev31 mod0
46108 pd-108 pd108 4643 jeff-31 rel0 rev31 mod0
46109 pd-109 pd109 4646 jeff-31 rel0 rev31 mod0
46110 pd-110 pd110 4649 jeff-31 rel0 rev31 mod0
46112 pd-112 pd112 4655 jeff-31 rel0 rev31 mod0
47105 ag-105 ag105 4719 jeff-31 rel0 rev31 mod0
47107 ag-107 ag107 4725 jeff-31 rel0 rev31 mod0
47109 ag-109 ag109 4731 jeff-31 rel0 rev31 mod0
47111 ag-111 ag111 4737 jeff-31 rel0 rev31 mod0
47601 ag-110m ag106m1 4723 jeff-31 rel0 rev31 mod0
47611 47611 ag108m1 4729 jeff-31 rel0 rev31 mod0
47621 47621 ag110m1 4735 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
48106 cd-106 cd106 4825 jeff-31 rel0 rev31 mod0
48108 cd-108 cd108 4831 jeff-31 rel0 rev31 mod0
48109 cd-109 cd109 4834 jeff-31 rel0 rev31 mod0
48110 cd-110 cd110 4837 jeff-31 rel0 rev31 mod0
48111 cd-111 cd111 4840 jeff-31 rel0 rev31 mod0
48112 cd-112 cd112 4843 jeff-31 rel0 rev31 mod0
48113 cd-113 cd113 4846 jeff-31 rel0 rev31 mod0
48114 cd-114 cd114 4849 jeff-31 rel0 rev31 mod0
48115 cd-115 cd115 4852 jeff-31 rel0 rev31 mod0
48116 cd-116 cd116 4855 jeff-31 rel0 rev31 mod0
48601 cd-115m cd113m1 4847 jeff-31 rel0 rev31 mod0
48611 48611 cd115m1 4853 jeff-31 rel0 rev31 mod0
49111 in-111 in111 4919 jeff-31 rel0 rev31 mod0
49113 in-113 in113 4925 jeff-31 rel0 rev31 mod0
49115 in-115 in115 4931 jeff-31 rel0 rev31 mod0
49601 49601 in114m1 4929 jeff-31 rel0 rev31 mod0
50112 sn-112 sn112 5025 jeff-31 rel0 rev31 mod0
50113 sn-113 sn113 5028 jeff-31 rel0 rev31 mod0
50114 sn-114 sn114 5031 jeff-31 rel0 rev31 mod0
50115 sn-115 sn115 5034 jeff-31 rel0 rev31 mod0
50116 sn-116 sn116 5037 jeff-31 rel0 rev31 mod0
50117 sn-117 sn117 5040 jeff-31 rel0 rev31 mod0
50118 sn-118 sn118 5043 jeff-31 rel0 rev31 mod0
50119 sn-119 sn119 5046 jeff-31 rel0 rev31 mod0
50120 sn-120 sn120 5049 jeff-31 rel0 rev31 mod0
50121 sn-121 sn121 5052 jeff-31 rel0 rev31 mod0
50122 sn-122 sn122 5055 jeff-31 rel0 rev31 mod0
50123 sn-123 sn123 5058 jeff-31 rel0 rev31 mod0
50124 sn-124 sn124 5061 jeff-31 rel0 rev31 mod0
50125 sn-125 sn125 5064 jeff-31 rel0 rev31 mod0
50126 sn-126 sn126 5067 jeff-31 rel0 rev31 mod0
50601 50601 sn117m1 5041 jeff-31 rel0 rev31 mod0
50611 50611 sn119m1 5047 jeff-31 rel0 rev31 mod0
50621 50621 sn121m1 5053 jeff-31 rel0 rev31 mod0
51119 sb-119 sb119 5119 jeff-31 rel0 rev31 mod0
51121 sb-121 sb121 5125 jeff-31 rel0 rev31 mod0
51122 sb-122 sb122 5128 jeff-31 rel0 rev31 mod0
51123 sb-123 sb123 5131 jeff-31 rel0 rev31 mod0
51124 sb-124 sb124 5134 jeff-31 rel0 rev31 mod0
51125 sb-125 sb125 5137 jeff-31 rel0 rev31 mod0
51126 sb-126 sb126 5140 jeff-31 rel0 rev31 mod0
51127 sb-127 sb127 5143 jeff-31 rel0 rev31 mod0
51601 51601 sb120m1 5123 jeff-31 rel0 rev31 mod0
52118 te-118 te118 5219 jeff-31 rel0 rev31 mod0
52119 te-119 te119 5222 jeff-31 rel0 rev31 mod0
52120 te-120 te120 5225 jeff-31 rel0 rev31 mod0
52121 te-121 te121 5228 jeff-31 rel0 rev31 mod0
52122 te-122 te122 5231 jeff-31 rel0 rev31 mod0
52123 te-123 te123 5234 jeff-31 rel0 rev31 mod0
52124 te-124 te124 5237 jeff-31 rel0 rev31 mod0
52125 te-125 te125 5240 jeff-31 rel0 rev31 mod0
52126 te-126 te126 5243 jeff-31 rel0 rev31 mod0
52127 te-127 te127 5246 jeff-31 rel0 rev31 mod0
52128 te-128 te128 5249 jeff-31 rel0 rev31 mod0
52129 te-129 te129 5252 jeff-31 rel0 rev31 mod0
52130 te-130 te130 5255 jeff-31 rel0 rev31 mod0
52132 te-132 te132 5261 jeff-31 rel0 rev31 mod0
52601 te-127m te119m1 5223 jeff-31 rel0 rev31 mod0
52611 te-129m te121m1 5229 jeff-31 rel0 rev31 mod0
52621 52621 te123m1 5235 jeff-31 rel0 rev31 mod0
52631 52631 te125m1 5241 jeff-31 rel0 rev31 mod0
52641 52641 te127m1 5247 jeff-31 rel0 rev31 mod0
52651 52651 te129m1 5253 jeff-31 rel0 rev31 mod0
52661 52661 te131m1 5259 jeff-31 rel0 rev31 mod0
53123 i-123 i123 5313 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
53124 i-124 i124 5316 jeff-31 rel0 rev31 mod0
53125 i-125 i125 5319 jeff-31 rel0 rev31 mod0
53126 i-126 i126 5322 jeff-31 rel0 rev31 mod0
53127 i-127 i127 5325 jeff-31 rel0 rev31 mod0
53128 i-128 i128 5328 jeff-31 rel0 rev31 mod0
53129 i-129 i129 5331 jeff-31 rel0 rev31 mod0
53130 i-130 i130 5334 jeff-31 rel0 rev31 mod0
53131 i-131 i131 5337 jeff-31 rel0 rev31 mod0
53133 i-133 i133 5343 jeff-31 rel0 rev31 mod0
54122 xe-122 xe122 5419 jeff-31 rel0 rev31 mod0
54124 xe-124 xe124 5425 jeff-31 rel0 rev31 mod0
54125 xe-125 xe125 5428 jeff-31 rel0 rev31 mod0
54126 xe-126 xe126 5431 jeff-31 rel0 rev31 mod0
54127 xe-127 xe127 5434 jeff-31 rel0 rev31 mod0
54128 xe-128 xe128 5437 jeff-31 rel0 rev31 mod0
54129 xe-129 xe129 5440 jeff-31 rel0 rev31 mod0
54130 xe-130 xe130 5443 jeff-31 rel0 rev31 mod0
54131 xe-131 xe131 5446 jeff-31 rel0 rev31 mod0
54132 xe-132 xe132 5449 jeff-31 rel0 rev31 mod0
54133 xe-133 xe133 5452 jeff-31 rel0 rev31 mod0
54134 xe-134 xe134 5455 jeff-31 rel0 rev31 mod0
54135 xe-135 xe135 5458 jeff-31 rel0 rev31 mod0
54136 xe-136 xe136 5461 jeff-31 rel0 rev31 mod0
54601 54601 xe129m1 5441 jeff-31 rel0 rev31 mod0
54611 54611 xe131m1 5447 jeff-31 rel0 rev31 mod0
54621 54621 xe133m1 5453 jeff-31 rel0 rev31 mod0
55129 cs-129 cs129 5513 jeff-31 rel0 rev31 mod0
55131 cs-131 cs131 5519 jeff-31 rel0 rev31 mod0
55132 cs-132 cs132 5522 jeff-31 rel0 rev31 mod0
55133 cs-133 cs133 5525 jeff-31 rel0 rev31 mod0
55134 cs-134 cs134 5528 jeff-31 rel0 rev31 mod0
55135 cs-135 cs135 5531 jeff-31 rel0 rev31 mod0
55136 cs-136 cs136 5534 jeff-31 rel0 rev31 mod0
55137 cs-137 cs137 5537 jeff-31 rel0 rev31 mod0
56128 ba-128 ba128 5619 jeff-31 rel0 rev31 mod0
56129 ba-129 ba129 5622 jeff-31 rel0 rev31 mod0
56130 ba-130 ba130 5625 jeff-31 rel0 rev31 mod0
56131 ba-131 ba131 5628 jeff-31 rel0 rev31 mod0
56132 ba-132 ba132 5631 jeff-31 rel0 rev31 mod0
56133 ba-133 ba133 5634 jeff-31 rel0 rev31 mod0
56134 ba-134 ba134 5637 jeff-31 rel0 rev31 mod0
56135 ba-135 ba135 5640 jeff-31 rel0 rev31 mod0
56136 ba-136 ba136 5643 jeff-31 rel0 rev31 mod0
56137 ba-137 ba137 5646 jeff-31 rel0 rev31 mod0
56138 ba-138 ba138 5649 jeff-31 rel0 rev31 mod0
56139 ba-139 ba139 5652 jeff-31 rel0 rev31 mod0
56140 ba-140 ba140 5655 jeff-31 rel0 rev31 mod0
56601 56601 ba133m1 5635 jeff-31 rel0 rev31 mod0
56611 56611 ba135m1 5641 jeff-31 rel0 rev31 mod0
57135 la-135 la135 5716 jeff-31 rel0 rev31 mod0
57137 la-137 la137 5722 jeff-31 rel0 rev31 mod0
57138 la-138 la138 5725 jeff-31 rel0 rev31 mod0
57139 la-139 la139 5728 jeff-31 rel0 rev31 mod0
57140 la-140 la140 5731 jeff-31 rel0 rev31 mod0
57141 la-141 la141 5734 jeff-31 rel0 rev31 mod0
58134 ce-134 ce134 5819 jeff-31 rel0 rev31 mod0
58135 ce-135 ce135 5822 jeff-31 rel0 rev31 mod0
58136 ce-136 ce136 5825 jeff-31 rel0 rev31 mod0
58138 ce-138 ce138 5831 jeff-31 rel0 rev31 mod0
58139 ce-139 ce139 5834 jeff-31 rel0 rev31 mod0
58140 ce-140 ce140 5837 jeff-31 rel0 rev31 mod0
58141 ce-141 ce141 5840 jeff-31 rel0 rev31 mod0
58142 ce-142 ce142 5843 jeff-31 rel0 rev31 mod0
58143 ce-143 ce143 5846 jeff-31 rel0 rev31 mod0
58144 ce-144 ce144 5849 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
58601 58601 ce137m1 5829 jeff-31 rel0 rev31 mod0
59141 pr-141 pr141 5925 jeff-31 rel0 rev31 mod0
59142 pr-142 pr142 5928 jeff-31 rel0 rev31 mod0
59143 pr-143 pr143 5931 jeff-31 rel0 rev31 mod0
60140 nd-140 nd140 6019 jeff-31 rel0 rev31 mod0
60141 nd-141 nd141 6022 jeff-31 rel0 rev31 mod0
60142 nd-142 nd142 6025 jeff-31 rel0 rev31 mod0
60143 nd-143 nd143 6028 jeff-31 rel0 rev31 mod0
60144 nd-144 nd144 6031 jeff-31 rel0 rev31 mod0
60145 nd-145 nd145 6034 jeff-31 rel0 rev31 mod0
60146 nd-146 nd146 6037 jeff-31 rel0 rev31 mod0
60147 nd-147 nd147 6040 jeff-31 rel0 rev31 mod0
60148 nd-148 nd148 6043 jeff-31 rel0 rev31 mod0
60149 nd-149 nd149 6046 jeff-31 rel0 rev31 mod0
60150 nd-150 nd150 6049 jeff-31 rel0 rev31 mod0
61143 pm-143 pm143 6137 jeff-31 rel0 rev31 mod0
61144 pm-144 pm144 6140 jeff-31 rel0 rev31 mod0
61145 pm-145 pm145 6143 jeff-31 rel0 rev31 mod0
61146 pm-146 pm146 6146 jeff-31 rel0 rev31 mod0
61147 pm-147 pm147 6149 jeff-31 rel0 rev31 mod0
61148 pm-148 pm148 6152 jeff-31 rel0 rev31 mod0
61149 pm-149 pm149 6155 jeff-31 rel0 rev31 mod0
61150 pm-150 pm150 6158 jeff-31 rel0 rev31 mod0
61151 pm-151 pm151 6161 jeff-31 rel0 rev31 mod0
61601 pm-148m pm148m1 6153 jeff-31 rel0 rev31 mod0
62144 sm-144 sm144 6225 jeff-31 rel0 rev31 mod0
62145 sm-145 sm145 6228 jeff-31 rel0 rev31 mod0
62146 sm-146 sm146 6231 jeff-31 rel0 rev31 mod0
62147 sm-147 sm147 6234 jeff-31 rel0 rev31 mod0
62148 sm-148 sm148 6237 jeff-31 rel0 rev31 mod0
62149 sm-149 sm149 6240 jeff-31 rel0 rev31 mod0
62150 sm-150 sm150 6243 jeff-31 rel0 rev31 mod0
62151 sm-151 sm151 6246 jeff-31 rel0 rev31 mod0
62152 sm-152 sm152 6249 jeff-31 rel0 rev31 mod0
62153 sm-153 sm153 6252 jeff-31 rel0 rev31 mod0
62154 sm-154 sm154 6255 jeff-31 rel0 rev31 mod0
63145 eu-145 eu145 6307 jeff-31 rel0 rev31 mod0
63146 eu-146 eu146 6310 jeff-31 rel0 rev31 mod0
63147 eu-147 eu147 6313 jeff-31 rel0 rev31 mod0
63148 eu-148 eu148 6316 jeff-31 rel0 rev31 mod0
63149 eu-149 eu149 6319 jeff-31 rel0 rev31 mod0
63150 eu-150 eu150 6322 jeff-31 rel0 rev31 mod0
63151 eu-151 eu151 6325 jeff-31 rel0 rev31 mod0
63152 eu-152 eu152 6328 jeff-31 rel0 rev31 mod0
63153 eu-153 eu153 6331 jeff-31 rel0 rev31 mod0
63154 eu-154 eu154 6334 jeff-31 rel0 rev31 mod0
63155 eu-155 eu155 6337 jeff-31 rel0 rev31 mod0
63156 eu-156 eu156 6340 jeff-31 rel0 rev31 mod0
63157 eu-157 eu157 6343 jeff-31 rel0 rev31 mod0
63601 63601 eu150m1 6323 jeff-31 rel0 rev31 mod0
63611 63611 eu152m1 6329 jeff-31 rel0 rev31 mod0
64146 gd-146 gd146 6407 jeff-31 rel0 rev31 mod0
64147 gd-147 gd147 6410 jeff-31 rel0 rev31 mod0
64148 gd-148 gd148 6413 jeff-31 rel0 rev31 mod0
64149 gd-149 gd149 6416 jeff-31 rel0 rev31 mod0
64150 gd-150 gd150 6419 jeff-31 rel0 rev31 mod0
64151 gd-151 gd151 6422 jeff-31 rel0 rev31 mod0
64152 gd-152 gd152 6425 jeff-31 rel0 rev31 mod0
64153 gd-153 gd153 6428 jeff-31 rel0 rev31 mod0
64154 gd-154 gd154 6431 jeff-31 rel0 rev31 mod0
64155 gd-155 gd155 6434 jeff-31 rel0 rev31 mod0
64156 gd-156 gd156 6437 jeff-31 rel0 rev31 mod0
64157 gd-157 gd157 6440 jeff-31 rel0 rev31 mod0
64158 gd-158 gd158 6443 jeff-31 rel0 rev31 mod0
64159 gd-159 gd159 6446 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
64160 gd-160 gd160 6449 jeff-31 rel0 rev31 mod0
65151 tb-151 tb151 6501 jeff-31 rel0 rev31 mod0
65152 tb-152 tb152 6504 jeff-31 rel0 rev31 mod0
65153 tb-153 tb153 6507 jeff-31 rel0 rev31 mod0
65154 tb-154 tb154 6510 jeff-31 rel0 rev31 mod0
65155 tb-155 tb155 6513 jeff-31 rel0 rev31 mod0
65156 tb-156 tb156 6516 jeff-31 rel0 rev31 mod0
65157 tb-157 tb157 6519 jeff-31 rel0 rev31 mod0
65158 tb-158 tb158 6522 jeff-31 rel0 rev31 mod0
65159 tb-159 tb159 6525 jeff-31 rel0 rev31 mod0
65160 tb-160 tb160 6528 jeff-31 rel0 rev31 mod0
65161 tb-161 tb161 6531 jeff-31 rel0 rev31 mod0
65601 65601 tb156m1 6517 jeff-31 rel0 rev31 mod0
65602 65602 tb154m2 6512 jeff-31 rel0 rev31 mod0
65612 65612 tb156m2 6518 jeff-31 rel0 rev31 mod0
66154 dy-154 dy154 6619 jeff-31 rel0 rev31 mod0
66155 dy-155 dy155 6622 jeff-31 rel0 rev31 mod0
66156 dy-156 dy156 6625 jeff-31 rel0 rev31 mod0
66157 dy-157 dy157 6628 jeff-31 rel0 rev31 mod0
66158 dy-158 dy158 6631 jeff-31 rel0 rev31 mod0
66159 dy-159 dy159 6634 jeff-31 rel0 rev31 mod0
66160 dy-160 dy160 6637 jeff-31 rel0 rev31 mod0
66161 dy-161 dy161 6640 jeff-31 rel0 rev31 mod0
66162 dy-162 dy162 6643 jeff-31 rel0 rev31 mod0
66163 dy-163 dy163 6646 jeff-31 rel0 rev31 mod0
66164 dy-164 dy164 6649 jeff-31 rel0 rev31 mod0
66165 dy-165 dy165 6652 jeff-31 rel0 rev31 mod0
66166 dy-166 dy166 6655 jeff-31 rel0 rev31 mod0
67163 ho-163 ho163 6719 jeff-31 rel0 rev31 mod0
67164 ho-164 ho164 6722 jeff-31 rel0 rev31 mod0
67165 ho-165 ho165 6725 jeff-31 rel0 rev31 mod0
67166 ho-166 ho166 6728 jeff-31 rel0 rev31 mod0
67601 ho-166m ho164m1 6723 jeff-31 rel0 rev31 mod0
67611 67611 ho166m1 6729 jeff-31 rel0 rev31 mod0
68160 er-160 er160 6819 jeff-31 rel0 rev31 mod0
68161 er-161 er161 6822 jeff-31 rel0 rev31 mod0
68162 er-162 er162 6825 jeff-31 rel0 rev31 mod0
68164 er-164 er164 6831 jeff-31 rel0 rev31 mod0
68165 er-165 er165 6834 jeff-31 rel0 rev31 mod0
68166 er-166 er166 6837 jeff-31 rel0 rev31 mod0
68167 er-167 er167 6840 jeff-31 rel0 rev31 mod0
68168 er-168 er168 6843 jeff-31 rel0 rev31 mod0
68169 er-169 er169 6846 jeff-31 rel0 rev31 mod0
68170 er-170 er170 6849 jeff-31 rel0 rev31 mod0
68171 er-171 er171 6852 jeff-31 rel0 rev31 mod0
68172 er-172 er172 6855 jeff-31 rel0 rev31 mod0
69165 tm-165 tm165 6913 jeff-31 rel0 rev31 mod0
69166 tm-166 tm166 6916 jeff-31 rel0 rev31 mod0
69167 tm-167 tm167 6919 jeff-31 rel0 rev31 mod0
69168 tm-168 tm168 6922 jeff-31 rel0 rev31 mod0
69169 tm-169 tm169 6925 jeff-31 rel0 rev31 mod0
69170 tm-170 tm170 6928 jeff-31 rel0 rev31 mod0
69171 tm-171 tm171 6931 jeff-31 rel0 rev31 mod0
69172 tm-172 tm172 6934 jeff-31 rel0 rev31 mod0
70166 yb-166 yb166 7019 jeff-31 rel0 rev31 mod0
70168 yb-168 yb168 7025 jeff-31 rel0 rev31 mod0
70169 yb-169 yb169 7028 jeff-31 rel0 rev31 mod0
70170 yb-170 yb170 7031 jeff-31 rel0 rev31 mod0
70171 yb-171 yb171 7034 jeff-31 rel0 rev31 mod0
70172 yb-172 yb172 7037 jeff-31 rel0 rev31 mod0
70173 yb-173 yb173 7040 jeff-31 rel0 rev31 mod0
70174 yb-174 yb174 7043 jeff-31 rel0 rev31 mod0
70175 yb-175 yb175 7046 jeff-31 rel0 rev31 mod0
70176 yb-176 yb176 7049 jeff-31 rel0 rev31 mod0
71169 lu-169 lu169 7107 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
71170 lu-170 lu170 7110 jeff-31 rel0 rev31 mod0
71171 lu-171 lu171 7113 jeff-31 rel0 rev31 mod0
71172 lu-172 lu172 7116 jeff-31 rel0 rev31 mod0
71173 lu-173 lu173 7119 jeff-31 rel0 rev31 mod0
71174 lu-174 lu174 7122 jeff-31 rel0 rev31 mod0
71175 lu-175 lu175 7125 jeff-31 rel0 rev31 mod0
71176 lu-176 lu176 7128 jeff-31 rel0 rev31 mod0
71177 lu-177 lu177 7131 jeff-31 rel0 rev31 mod0
71601 71601 lu174m1 7123 jeff-31 rel0 rev31 mod0
71611 71611 lu177m1 7132 jeff-31 rel0 rev31 mod0
72170 hf-170 hf170 7213 jeff-31 rel0 rev31 mod0
72171 hf-171 hf171 7216 jeff-31 rel0 rev31 mod0
72172 hf-172 hf172 7219 jeff-31 rel0 rev31 mod0
72173 hf-173 hf173 7222 jeff-31 rel0 rev31 mod0
72174 hf-174 hf174 7225 jeff-31 rel0 rev31 mod0
72175 hf-175 hf175 7228 jeff-31 rel0 rev31 mod0
72176 hf-176 hf176 7231 jeff-31 rel0 rev31 mod0
72177 hf-177 hf177 7234 jeff-31 rel0 rev31 mod0
72178 hf-178 hf178 7237 jeff-31 rel0 rev31 mod0
72179 hf-179 hf179 7240 jeff-31 rel0 rev31 mod0
72180 hf-180 hf180 7243 jeff-31 rel0 rev31 mod0
72181 hf-181 hf181 7246 jeff-31 rel0 rev31 mod0
72182 hf-182 hf182 7249 jeff-31 rel0 rev31 mod0
72601 72601 hf180m1 7244 jeff-31 rel0 rev31 mod0
72602 72602 hf178m2 7239 jeff-31 rel0 rev31 mod0
72612 72612 hf179m2 7242 jeff-31 rel0 rev31 mod0
73177 ta-177 ta177 7316 jeff-31 rel0 rev31 mod0
73179 ta-179 ta179 7322 jeff-31 rel0 rev31 mod0
73180 ta-180 ta180 7325 jeff-31 rel0 rev31 mod0
73181 ta-181 ta181 7328 jeff-31 rel0 rev31 mod0
73182 ta-182 ta182 7331 jeff-31 rel0 rev31 mod0
73183 ta-183 ta183 7334 jeff-31 rel0 rev31 mod0
73601 73601 ta180m1 7326 jeff-31 rel0 rev31 mod0
74178 w-178 w178 7419 jeff-31 rel0 rev31 mod0
74180 w-180 w180 7425 jeff-31 rel0 rev31 mod0
74181 w-181 w181 7428 jeff-31 rel0 rev31 mod0
74182 w-182 w182 7431 jeff-31 rel0 rev31 mod0
74183 w-183 w183 7434 jeff-31 rel0 rev31 mod0
74184 w-184 w184 7437 jeff-31 rel0 rev31 mod0
74185 w-185 w185 7440 jeff-31 rel0 rev31 mod0
74186 w-186 w186 7443 jeff-31 rel0 rev31 mod0
74187 w-187 w187 7446 jeff-31 rel0 rev31 mod0
74188 w-188 w188 7449 jeff-31 rel0 rev31 mod0
75181 re-181 re181 7513 jeff-31 rel0 rev31 mod0
75182 re-182 re182 7516 jeff-31 rel0 rev31 mod0
75183 re-183 re183 7519 jeff-31 rel0 rev31 mod0
75184 re-184 re184 7522 jeff-31 rel0 rev31 mod0
75185 re-185 re185 7525 jeff-31 rel0 rev31 mod0
75186 re-186 re186 7528 jeff-31 rel0 rev31 mod0
75187 re-187 re187 7531 jeff-31 rel0 rev31 mod0
75188 re-188 re188 7534 jeff-31 rel0 rev31 mod0
75189 re-189 re189 7537 jeff-31 rel0 rev31 mod0
75601 75601 re182m1 7517 jeff-31 rel0 rev31 mod0
75611 75611 re184m1 7523 jeff-31 rel0 rev31 mod0
75621 75621 re186m1 7529 jeff-31 rel0 rev31 mod0
76182 os-182 os182 7619 jeff-31 rel0 rev31 mod0
76183 os-183 os183 7622 jeff-31 rel0 rev31 mod0
76184 os-184 os184 7625 jeff-31 rel0 rev31 mod0
76185 os-185 os185 7628 jeff-31 rel0 rev31 mod0
76186 os-186 os186 7631 jeff-31 rel0 rev31 mod0
76187 os-187 os187 7634 jeff-31 rel0 rev31 mod0
76188 os-188 os188 7637 jeff-31 rel0 rev31 mod0
76189 os-189 os189 7640 jeff-31 rel0 rev31 mod0
76190 os-190 os190 7643 jeff-31 rel0 rev31 mod0
76191 os-191 os191 7646 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
76192 os-192 os192 7649 jeff-31 rel0 rev31 mod0
76193 os-193 os193 7652 jeff-31 rel0 rev31 mod0
76194 os-194 os194 7655 jeff-31 rel0 rev31 mod0
76601 76601 os191m1 7647 jeff-31 rel0 rev31 mod0
77185 ir-185 ir185 7707 jeff-31 rel0 rev31 mod0
77186 ir-186 ir186 7710 jeff-31 rel0 rev31 mod0
77188 ir-188 ir188 7716 jeff-31 rel0 rev31 mod0
77189 ir-189 ir189 7719 jeff-31 rel0 rev31 mod0
77190 ir-190 ir190 7722 jeff-31 rel0 rev31 mod0
77191 ir-191 ir191 7725 jeff-31 rel0 rev31 mod0
77192 ir-192 ir192 7728 jeff-31 rel0 rev31 mod0
77193 ir-193 ir193 7731 jeff-31 rel0 rev31 mod0
77194 ir-194 ir194 7734 jeff-31 rel0 rev31 mod0
77601 77601 ir193m1 7732 jeff-31 rel0 rev31 mod0
77602 77602 ir192m2 7730 jeff-31 rel0 rev31 mod0
77611 77611 ir194m1 7735 jeff-31 rel0 rev31 mod0
77621 77621 ir196m1 7741 jeff-31 rel0 rev31 mod0
78188 pt-188 pt188 7819 jeff-31 rel0 rev31 mod0
78189 pt-189 pt189 7822 jeff-31 rel0 rev31 mod0
78190 pt-190 pt190 7825 jeff-31 rel0 rev31 mod0
78191 pt-191 pt191 7828 jeff-31 rel0 rev31 mod0
78192 pt-192 pt192 7831 jeff-31 rel0 rev31 mod0
78193 pt-193 pt193 7834 jeff-31 rel0 rev31 mod0
78194 pt-194 pt194 7837 jeff-31 rel0 rev31 mod0
78195 pt-195 pt195 7840 jeff-31 rel0 rev31 mod0
78196 pt-196 pt196 7843 jeff-31 rel0 rev31 mod0
78197 pt-197 pt197 7846 jeff-31 rel0 rev31 mod0
78198 pt-198 pt198 7849 jeff-31 rel0 rev31 mod0
78200 pt-200 pt200 7855 jeff-31 rel0 rev31 mod0
78202 pt-202 pt202 7861 jeff-31 rel0 rev31 mod0
78601 78601 pt193m1 7835 jeff-31 rel0 rev31 mod0
78611 78611 pt195m1 7841 jeff-31 rel0 rev31 mod0
79193 au-193 au193 7913 jeff-31 rel0 rev31 mod0
79194 au-194 au194 7916 jeff-31 rel0 rev31 mod0
79195 au-195 au195 7919 jeff-31 rel0 rev31 mod0
79196 au-196 au196 7922 jeff-31 rel0 rev31 mod0
79197 au-197 au197 7925 jeff-31 rel0 rev31 mod0
79198 au-198 au198 7928 jeff-31 rel0 rev31 mod0
79199 au-199 au199 7931 jeff-31 rel0 rev31 mod0
79601 79601 au198m1 7929 jeff-31 rel0 rev31 mod0
79611 79611 au200m1 7935 jeff-31 rel0 rev31 mod0
80193 hg-193 hg193 8016 jeff-31 rel0 rev31 mod0
80194 hg-194 hg194 8019 jeff-31 rel0 rev31 mod0
80195 hg-195 hg195 8022 jeff-31 rel0 rev31 mod0
80196 hg-196 hg196 8025 jeff-31 rel0 rev31 mod0
80197 hg-197 hg197 8028 jeff-31 rel0 rev31 mod0
80198 hg-198 hg198 8031 jeff-31 rel0 rev31 mod0
80199 hg-199 hg199 8034 jeff-31 rel0 rev31 mod0
80200 hg-200 hg200 8037 jeff-31 rel0 rev31 mod0
80201 hg-201 hg201 8040 jeff-31 rel0 rev31 mod0
80202 hg-202 hg202 8043 jeff-31 rel0 rev31 mod0
80203 hg-203 hg203 8046 jeff-31 rel0 rev31 mod0
80204 hg-204 hg204 8049 jeff-31 rel0 rev31 mod0
80601 80601 hg193m1 8017 jeff-31 rel0 rev31 mod0
80611 80611 hg195m1 8023 jeff-31 rel0 rev31 mod0
80621 80621 hg197m1 8029 jeff-31 rel0 rev31 mod0
81200 tl-200 tl200 8116 jeff-31 rel0 rev31 mod0
81201 tl-201 tl201 8119 jeff-31 rel0 rev31 mod0
81202 tl-202 tl202 8122 jeff-31 rel0 rev31 mod0
81203 tl-203 tl203 8125 jeff-31 rel0 rev31 mod0
81204 tl-204 tl204 8128 jeff-31 rel0 rev31 mod0
81205 tl-205 tl205 8131 jeff-31 rel0 rev31 mod0
82200 pb-200 pb200 8213 jeff-31 rel0 rev31 mod0
82202 pb-202 pb202 8219 jeff-31 rel0 rev31 mod0
82203 pb-203 pb203 8222 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
82204 pb-204 pb204 8225 jeff-31 rel0 rev31 mod0
82205 pb-205 pb205 8228 jeff-31 rel0 rev31 mod0
82206 pb-206 pb206 8231 jeff-31 rel0 rev31 mod0
82207 pb-207 pb207 8234 jeff-31 rel0 rev31 mod0
82208 pb-208 pb208 8237 jeff-31 rel0 rev31 mod0
82209 pb-209 pb209 8240 jeff-31 rel0 rev31 mod0
82210 pb-210 pb210 8243 jeff-31 rel0 rev31 mod0
83203 bi-203 bi203 8307 jeff-31 rel0 rev31 mod0
83205 bi-205 bi205 8313 jeff-31 rel0 rev31 mod0
83206 bi-206 bi206 8316 jeff-31 rel0 rev31 mod0
83207 bi-207 bi207 8319 jeff-31 rel0 rev31 mod0
83208 bi-208 bi208 8322 jeff-31 rel0 rev31 mod0
83209 bi-209 bi209 8325 jeff-31 rel0 rev31 mod0
83210 bi-210 bi210 8328 jeff-31 rel0 rev31 mod0
83601 83601 bi210m1 8329 jeff-31 rel0 rev31 mod0
84206 po-206 po206 8425 jeff-31 rel0 rev31 mod0
84207 po-207 po207 8428 jeff-31 rel0 rev31 mod0
84208 po-208 po208 8431 jeff-31 rel0 rev31 mod0
84209 po-209 po209 8434 jeff-31 rel0 rev31 mod0
84210 po-210 po210 8437 jeff-31 rel0 rev31 mod0
86211 rn-211 rn211 8625 jeff-31 rel0 rev31 mod0
86222 rn-222 rn222 8658 jeff-31 rel0 rev31 mod0
88223 ra-223 ra223 8825 jeff-31 rel0 rev31 mod0
88224 ra-224 ra224 8828 jeff-31 rel0 rev31 mod0
88225 ra-225 ra225 8831 jeff-31 rel0 rev31 mod0
88226 ra-226 ra226 8834 jeff-31 rel0 rev31 mod0
88228 ra-228 ra228 8840 jeff-31 rel0 rev31 mod0
89225 ac-225 ac225 8925 jeff-31 rel0 rev31 mod0
89226 ac-226 ac226 8928 jeff-31 rel0 rev31 mod0
89227 ac-227 ac227 8931 jeff-31 rel0 rev31 mod0
90227 th-227 th227 9025 jeff-31 rel0 rev31 mod0
90228 th-228 th228 9028 jeff-31 rel0 rev31 mod0
90229 th-229 th229 9031 jeff-31 rel0 rev31 mod0
90230 th-230 th230 9034 jeff-31 rel0 rev31 mod0
90231 th-231 th231 9037 jeff-31 rel0 rev31 mod0
90232 th-232 th232 9040 jeff-31 rel0 rev31 mod0
90234 th-234 th234 9046 jeff-31 rel0 rev31 mod0
91228 pa-228 pa228 9122 jeff-31 rel0 rev31 mod0
91229 pa-229 pa229 9125 jeff-31 rel0 rev31 mod0
91230 pa-230 pa230 9128 jeff-31 rel0 rev31 mod0
91231 pa-231 pa231 9131 jeff-31 rel0 rev31 mod0
91232 pa-232 pa232 9134 jeff-31 rel0 rev31 mod0
91233 pa-233 pa233 9137 jeff-31 rel0 rev31 mod0
92230 u-230 u230 9213 jeff-31 rel0 rev31 mod0
92231 u-231 u231 9216 jeff-31 rel0 rev31 mod0
92232 u-232 u232 9219 jeff-31 rel0 rev31 mod0
92233 u-233 u233 9222 jeff-31 rel0 rev31 mod0
92234 u-234 u234 9225 jeff-31 rel0 rev31 mod0
92235 u-235 u235 9228 jeff-31 rel0 rev31 mod0
92236 u-236 u236 9231 jeff-31 rel0 rev31 mod0
92237 u-237 u237 9234 jeff-31 rel0 rev31 mod0
92238 u-238 u238 9237 jeff-31 rel0 rev31 mod0
92240 u-240 u240 9243 jeff-31 rel0 rev31 mod0
93234 np-234 np234 9337 jeff-31 rel0 rev31 mod0
93235 np-235 np235 9340 jeff-31 rel0 rev31 mod0
93236 np-236 np236 9343 jeff-31 rel0 rev31 mod0
93237 np-237 np237 9346 jeff-31 rel0 rev31 mod0
93238 np-238 np238 9349 jeff-31 rel0 rev31 mod0
93239 np-239 np239 9352 endf-70/jeff-31 rel0 rev31 mod0
94236 pu-236 pu236 9428 jeff-31 rel0 rev31 mod0
94237 pu-237 pu237 9431 jeff-31 rel0 rev31 mod0
94238 pu-238 pu238 9434 jeff-31 rel0 rev31 mod0
94239 pu-239 pu239 9437 jeff-31 rel0 rev31 mod0
94240 pu-240 pu240 9440 jeff-31 rel0 rev31 mod0
94241 pu-241 pu241 9443 jeff-31 rel0 rev31 mod0
```

(continues on next page)

(continued from previous page)

```
94242 pu-242 pu242 9446 jeff-31 rel0 rev31 mod0
94244 pu-244 pu244 9452 jeff-31 rel0 rev31 mod0
94246 pu-246 pu246 9458 jeff-31 rel0 rev31 mod0
94247 pu-247 pu247 9461 jeff-31 rel0 rev31 mod0
95240 am-240 am240 9540 jeff-31 rel0 rev31 mod0
95241 am-241 am241 9543 jeff-31 rel0 rev31 mod0
95242 am-242 am242 9546 jeff-31 rel0 rev31 mod0
95243 am-243 am243 9549 jeff-31 rel0 rev31 mod0
95601 am-242m am242m1 9547 jeff-31 rel0 rev31 mod0
96240 cm-240 cm240 9625 jeff-31 rel0 rev31 mod0
96241 cm-241 cm241 9628 jeff-31 rel0 rev31 mod0
96242 cm-242 cm242 9631 jeff-31 rel0 rev31 mod0
96243 cm-243 cm243 9634 jeff-31 rel0 rev31 mod0
96244 cm-244 cm244 9637 jeff-31 rel0 rev31 mod0
96245 cm-245 cm245 9640 jeff-31 rel0 rev31 mod0
96246 cm-246 cm246 9643 jeff-31 rel0 rev31 mod0
96247 cm-247 cm247 9646 jeff-31 rel0 rev31 mod0
96248 cm-248 cm248 9649 jeff-31 rel0 rev31 mod0
96249 cm-249 cm249 9652 jeff-31 rel0 rev31 mod0
96250 cm-250 cm250 9655 jeff-31 rel0 rev31 mod0
97245 bk-245 bk245 9740 jeff-31 rel0 rev31 mod0
97246 bk-246 bk246 9743 jeff-31 rel0 rev31 mod0
97247 bk-247 bk247 9746 jeff-31 rel0 rev31 mod0
97248 bk-248 bk248 9749 jeff-31 rel0 rev31 mod0
97249 bk-249 bk249 9752 jeff-31 rel0 rev31 mod0
97250 bk-250 bk250 9755 jeff-31 rel0 rev31 mod0
97601 97601 bk248m1 9750 jeff-31 rel0 rev31 mod0
98246 cf-246 cf246 9843 jeff-31 rel0 rev31 mod0
98248 cf-248 cf248 9849 jeff-31 rel0 rev31 mod0
98249 cf-249 cf249 9852 jeff-31 rel0 rev31 mod0
98250 cf-250 cf250 9855 jeff-31 rel0 rev31 mod0
98251 cf-251 cf251 9858 jeff-31 rel0 rev31 mod0
98252 cf-252 cf252 9861 jeff-31 rel0 rev31 mod0
98253 cf-253 cf253 9864 jeff-31 rel0 rev31 mod0
98254 cf-254 cf254 9867 jeff-31 rel0 rev31 mod0
99251 es-251 es251 9911 jeff-31 rel0 rev31 mod0
99252 es-252 es252 9912 jeff-31 rel0 rev31 mod0
99253 es-253 es253 9913 jeff-31 rel0 rev31 mod0
99254 es-254 es254 9914 jeff-31 rel0 rev31 mod0
99255 es-255 es255 9916 jeff-31 rel0 rev31 mod0
99601 99601 es254m1 9915 jeff-31 rel0 rev31 mod0
100252 fm-252 fm252 9933 jeff-31 rel0 rev31 mod0
100253 fm-253 fm253 9934 jeff-31 rel0 rev31 mod0
100255 fm-255 fm255 9936 jeff-31 rel0 rev31 mod0
100257 fm-257 fm257 9938 jeff-31 rel0 rev31 mod0
```

10.3 SCALE NUCLEAR DATA COVARIANCE LIBRARY

A. Holcomb, D. Wiarda, M. L. Williams, G. Arbanas, and B. L. Broadhead

ABSTRACT

An updated cross section covariance library has been created for use with the sensitivity and uncertainty modules in SCALE 6.3. The data have been assembled from a variety sources, including high-fidelity covariance evaluations from ENDF/B-VIII.0 as well as approximate uncertainties obtained from a collaborative project performed by Brookhaven National Laboratory, Los Alamos National Laboratory, and Oak Ridge National Laboratory. This document describes the assumptions in generating the data, the library contents, and processing procedure for the SCALE 56-group and 252-group covariance libraries. The ENDF/B-VII.1 covariance libraries provided with SCALE 6.3 are identical to the libraries released with SCALE 6.2. The SCALE 44-group covariance library distributed with SCALE 6.0 and SCALE 6.1 is retained for backwards compatibility.

ACKNOWLEDGMENT

We gratefully acknowledge the sponsorship of the US Department of Energy Nuclear Criticality Safety Program in the development of the SCALE 6.3 covariance libraries.

10.3.1 INTRODUCTION

The SCALE 6.3 covariance library suite contains covariance libraries based on ENDF/B-VIII.0, ENDF/B-VII.1, and pre-ENDF/B-VII.0. For each ENDF release, if uncertainty information is not available for all neutron targets, then the corresponding SCALE covariance library is patched such that each library has some estimate of the uncertainty data for every neutron target in the library. The ENDF/B-VIII.0 covariance library [XSLibDAB18] contains at least some covariance data for 252 nuclides and the ENDF/B-VII.1 library [COVLIBCHO+11] has data for 187 nuclides. Both the ENDF/B-VIII.0 and ENDF/B-VII.1 covariance libraries have been supplemented by the previous SCALE 6.1 covariance data for the ~305 nuclides missing covariance data in ENDF/B-VIII.0 and ~215 nuclides missing covariance data in ENDF/B-VII.1. The ENDF/B-VII.1 uncertainties were modified for a few nuclides, as described in Sect. 10.3.2.3. The ENDF/B-VIII.0 covariance data are available in a 56-group format, and the ENDF/B-VII.1 covariance library is in both a 56-group and 252-group format. These covariance libraries were generated for compatibility with the ENDF/B-VII.1 cross section libraries distributed with SCALE 6.3, and they may also be applied for the 238-group ENDF/B-VII.0 library provided with SCALE 6.2. The previous SCALE 6.0 and SCALE 6.1 44-group library (44groupcov) was based on older covariance data and is retained in SCALE 6.3 for backwards compatibility. However, the 56- and 252-group covariance libraries (56groupcov7.1 and 252groupcov7.1) are now recommended for all applications. The 56-group library-which is default for SCALE uncertainty analysis-and the 252 fine-group library generally produce similar results, except for some threshold reactions such as (n,2n). The 252-group library may be used to improve uncertainty estimates from these types of data, but it takes more execution time than the default 56-group library. Because the 56- and 252-group covariance data in many cases are based on newer uncertainty evaluations than the previous 44-group library, some differences will occur between these sets of results.

The covariance data correspond to relative uncertainties assembled from a variety of sources, including evaluations from ENDF/B-VIII.0, ENDF/B-VII.1, ENDF/B-VI, and approximated uncertainties from a collaborative project performed by Brookhaven National Laboratory (BNL), Los Alamos National Laboratory (LANL), and Oak Ridge National Laboratory (ORNL). Because SCALE uncertainty data come from several different sources, the application of a single generic covariance library to all multigroup cross section libraries raises questions about consistency with any given data evaluation. In reality, much of the approximate uncertainty data in the library is based on simplifying approximations that do not depend on specific ENDF evaluations and thus can be applied to all cross section libraries within the limitations of the assumed methodology. In other cases in which a covariance evaluation has been taken from a specific nuclear data file (e.g., ENDF/B-VIII.-,ENDF/B-VII.1, ENDF/B-VI, or JENDL), it is assumed that the same *relative* (rather than *absolute*) uncertainties can be applied to all cross section libraries, even if these are not strictly consistent with the nuclear data evaluations. The assumption is partially justified by the fact that different evaluations often use many of the same experimental measurements since there is a limited amount of this information available. In some cases, older data evaluations have been carried over into the newer ENDF versions. Also, because many important nuclear data are now known rather well, newer evaluations in many instances correspond to rather modest variations from previous ones and are expected to lie within the earlier uncertainties.

No inherently "true" uncertainty can be defined for nuclear data. For example, in theory, two independent evaluations could produce similar nuclear data with very different uncertainties. Differences in nuclear data evaluations directly impact calculations that can be affirmed by comparisons with benchmark experiments;

but there is no established procedure to quantify the reliability of uncertainty estimates. In general, the SCALE covariance library should be viewed as a best-estimate assessment of data uncertainties based upon the specific methodology described in the following section. While this methodology is not unique and other approaches could have been used, the SCALE covariance library is a reasonable representation of the nuclear data uncertainties for most applications given the current lack of information. Furthermore, it is the only available comprehensive library that has been created in a well-defined, systematic manner.

10.3.2 COVARIANCE DATA DESCRIPTION

10.3.2.1 Evaluated covariances from nuclear data files

A rigorous, modern evaluation of nuclear data typically uses a regression algorithm that adjusts parameters in a nuclear physics model (e.g., Reich-Moore resonance formula, optical model, etc.), to fit a set of differential experimental measurements that have various sources of statistical and systematic uncertainties [COVLIBLLD+06]. Information from the regression analysis of the model parameters can be propagated to uncertainties and correlations in the evaluated differential data. In this manner, the differential nuclear data and covariances are consistent and are coupled together by evaluation processes. Unfortunately, only a limited number of cross section evaluations have produced high-fidelity covariances in this rigorous manner. All other nuclear data uncertainties must be estimated from approximations in which the uncertainty assessment is decoupled from the original evaluation procedure.

The SCALE covariance library is based on several different uncertainty approximations with varying degrees of fidelity relative to the actual nuclear data evaluation. The library includes high-fidelity evaluated covariances obtained from the latest ENDF releases whenever available. As discussed in Sect. 10.3.1, it is assumed that covariances taken from one data evaluation, such as ENDF/B-VII.1, can also be applied to other evaluations of the same data, such as ENDF/B-VIII.0. If this is done judiciously for cases in which the nuclear data evaluations are similar, then the covariances taken from one source should be a reasonable representation of uncertainties for the other evaluations.

10.3.2.2 Approximate covariance data

At the other end of the spectrum from high fidelity data, low-fidelity (lo-fi) covariances are defined to be those estimated independently of a specific data evaluation. The approximate covariance data in SCALE are based on results from a collaborative project funded by the US Department of Energy Nuclear Criticality Safety Program to generate lo-fi covariances over the energy range from 10^{-5} eV to 20 MeV for materials without covariances in ENDF/B-VII.1. Nuclear data experts at BNL, LANL, and ORNL devised simple procedures to estimate data uncertainties in the absence of high fidelity covariance evaluations. The result of this project is a set of covariance data in ENDF/B file 33 format that can be processed into multigroup covariances [COVLIBLKH+08]. Some of these data were later revised and included in ENDF/B-VII.1, while others were carried over from SCALE 6.1 to the SCALE 6.3 library. In this documentation, these data are known as BLO (BNL-LANL-ORNL) uncertainty data, which were generated as described below.

ORNL used uncertainties in integral experiment measurements of thermal cross sections, resonance integrals, and potential cross sections to approximate the standard deviations of capture, fission, and elastic scattering reactions for the thermal (<0.5 eV) and resonance ranges (0.5 eV- 5 keV). Full energy correlation was assumed for the covariances within each of these respective ranges [COVLIBWBDR07, COVLIBWR08] This procedure was originally introduced for the approximate uncertainty data in SCALE 5.1. However, the current version includes updated integral measurement uncertainties, using the more recent values tabulated by Mughabghab in the *Atlas of Neutron Resonances* [COVLIBMug06]. The lo-fi relative uncertainty is computed as the absolute uncertainty in the integral parameter (i.e., thermal cross section or resonance

integral) taken from the *Atlas*, divided by the average of the measured parameter and the calculated value computed from ENDF/B-VII differential data:

$$U = \frac{\Delta_I}{0.5 \times (X_I + X_D)}, \tag{10.3.1}$$

where:

U is the relative lo-fi uncertainty included in SCALE,

Δ_I is the absolute uncertainty in the integral measurement (obtained from Mughabghab), and

X_I and X_D are the measured and computed (from ENDF/B differential data) integral parameter values, respectively.

In some cases the integral measurement value from the Mughabghab *Atlas*[6] and the corresponding value computed from the ENDF/B-VII differential evaluation are inconsistent-defined here as having a difference greater than two standard deviations in the measured and computed integral parameters. In these cases, the lo-fi relative standard deviation is defined as half the difference relative to the average of the measured and calculated values:

$$U = \frac{|X_I - X_D|}{X_I + X_D};\ \text{for}\ |X_I - X_D| > 2\Delta_I. \tag{10.3.2}$$

In some instances this expression may exceed 100%. For these cases, a 100% uncertainty was assigned. Also, the *Atlas* does not include uncertainties in integral measurements for several isotopes, which typically are not of great interest for most applications. In this case the integral uncertainty was defined as a +/-5 in the least significant digit for these materials; e.g., 1.23 is assign an uncertainty of +/- 5E-3.

BNL and LANL provided estimates in the fast energy range from 5 keV to 20 MeV for covariances of capture, fission, elastic, inelastic, (n,2n) cross sections, and prompt nubar. BNL used optical model calculations with estimated uncertainties in model parameters to compute covariances in the fast range for about 300 structural isotopes, fission products, and non-fissionable heavy nuclei. Estimated uncertainties in model parameters were based on previous work and expert judgment [COVLIBPHO09]. Covariances for 14 actinide isotopes were obtained from earlier work performed by BNL for Subgroup-26 (SG-26) [COVLIBRHOM07]. The SG-26 actinide covariances cover the full energy range, including thermal, resonance, and fast regions. If the thermal data uncertainties estimated by the SG-26 approach exceed the thermal uncertainty given in reference 6, the thermal data covariances are represented by ORNL's integral uncertainty technique.

LANL produced covariances in the fast range for an additional 47 actinide materials. The LANL actinide covariances were based on empirical estimates of nuclear reaction models [COVLIBKTY+08]. Full energy range covariances were also produced by LANL for 16 light isotopes ranging from hydrogen to fluorine [COVLIBHal08]. These included high fidelity covariances from R-matrix analyses for ^{1}H, ^{6}Li, and ^{10}B, along with lo-fi uncertainties for the other materials, based on approximations such as least-squares fitting to experimental data, statistical model calculations at higher energies, or sometimes simply best-judgment estimation [COVLIBLKH+08].

10.3.2.3 Modifications to covariance data

In generating earlier covariance libraries, some omissions or inconsistencies were identified and corrected in the current covariance library:

- If the absolute correlation is larger than 1, it is set to 1.
- If a relative uncertainty is larger than 1, it is set to 1.

- If cross section data exist but covariance data do not span the entire range, then the diagonal element for the higher energy groups is repeated for the lower energy groups.
- If total inelastic scattering covariance is not supplied, it is calculated from the uncertainties in the discrete level inelastic data.
- If total nubar covariance is not supplied, it is calculated from the the prompt and delayed nubar uncertainties

A few inconsistencies were found in the ENDF/B-VII.1 uncertainty data, and these were modified for the SCALE 6.3 covariance library [COVLIBWIMR14]. The corrections were also conveyed to the National Nuclear Data Center, where they were added to the ENDF/A file for possible inclusion in the future release of ENDF/B-VIII.1. These modifications are summarized below:

(a) ^{235}U thermal nubar: standard deviation was decreased from 0.7% to 0.3% in energy range from 0.0 to 0.5 eV, consistent with JENDL-3.3.

(b) ^{239}Pu thermal nubar: standard deviation was increased from 0.01% to 0.15% in energy range from 0.0 to 0.01 eV, consistent with ENDF/B-VII.1 uncertainty at 0.01 eV.

(c) H thermal capture: standard deviation reduced from 2.5% to 0.2%, consistent with Williams and Rearden 2008 [COVLIBWR08],

(d) ^{103}Rh thermal capture: reduced from ~4% to 1.04%, consistent with Williams and Rearden 2008 [COVLIBWR08].

(e) ^{151}Sm thermal capture: modified to ~1.8%, consistent with Williams and Rearden 2008 [COVLIBWR08].

(f) ^{147}Pm: standard deviation was reduced from 24% to 5% in the energy range 0.5–5000 eV, consistent with the quoted resonance integral uncertainty in Williams and Rearden 2008 [COVLIBWR08].

Several modifications were also made to the uncertainties obtained from the original BLO data used in SCALE 6.1. The energy boundary between the thermal and resonance covariance blocks was modified from 0.5 to 0.625 eV in order to coincide with a 56-group boundary. The BLO lo-fi data do not include thermal or resonance range uncertainties for isotope reactions that do not have integral uncertainties given in the Mughabghab text [COVLIBMug06]. These occur mainly for relatively unimportant data such as elastic cross sections of several fission products. Therefore in these cases the uncertainties were estimated using different approaches. For example, the thermal data uncertainty was sometimes used to represent the epithermal uncertainty if it was not available in the Mughabghab tabulation, and sometimes the high-energy uncertainty was extended to lower energies. The uncertainty in the ^{149}Sm resonance capture integral is not provided in the 2006 edition of Mughabghab's text, so it was set to the value of 5.7%, which was obtained from an earlier tabulation by Mughabghab [COVLIBMug03].

10.3.2.4 Covariance data for fission spectra

As of ENDF/B-VII.1, covariance matrices are now provided for the fission exit energy distribution. The data are given as a function of incident energy. The incident energy grid is very broad, and the exit energy distribution is constant over a given incident energy group. Since the COVERX library file only allows one multigroup fission spectrum (χ) covariance matrix per nuclide, the exit energy spectrum is used for the average energy of fission. If ν is nubar, f is fission, and w is the appropriate flux, then the average energy of fission is calculated as:

$$10^7 \exp\left(-\frac{\sum \nu f w \frac{1}{2}\left(\log\left(\frac{10^7}{E_{g1}}\right)+\log\left(\frac{10^7}{E_{g2}}\right)\right)}{\sum \nu f w}\right) \tag{10.3.3}$$

where the sum is over all groups and E_{g1} and E_{g2} are the group boundaries for group g. ENDF/B-VII.1 provides covariance data for exit energy distributions for 64 nuclides. This includes all nuclides for which fission spectrum (χ) covariance matrices where provided in the previous covariance library. Some additional χ-covariance matrices were taken from JENDL-4.0. The new 56-group and 252-group fission spectrum covariances are more complete and significantly improved compared to the earlier 44-group chi uncertainty data, which were based on the Watt fission spectrum in ENDF/B-V. (see Sect. 10.3.5).

10.3.3 MULTIGROUP COVARIANCE PROCESSING

Covariance data were processed with the AMPX code PUFF-IV [COVLIBWD06]. PUFF-IV has major improvements in the treatment of the resolved and unresolved resonance parameter uncertainties over previous code versions [COVLIBWALD08]. All nuclides with resonance parameter uncertainty files were processed with the full sensitivity option in PUFF-IV.

10.3.4 CONTENTS OF THE SCALE 6.3 COVARIANCE LIBRARY

The SCALE covariance library provides uncertainty data in 56- and 252-group formats for a total of 456 materials in ENDF/B-VII.1 and 587 materials in ENDF/B-VIII.0, including some duplication for materials with multiple thermal scattering kernels. Table 10.3.1 describes the contents of the library using the following nomenclature:

1. ENDF/B-VII.1: evaluated covariance data released with ENDF/B-VII.1
2. ENDF/B-VII.2-prelim: recently evaluated data proposed for future release of ENDF/B-VII.2
3. ENDF/B-VI: evaluated covariance data released with ENDF/B-VI
4. BLO approximate data: lo-fi covariances from BLO project
5. SG-26: approximate covariances from WPEC Subgroup-26
6. JENDL-4.0: evaluated covariance data released with JENDL-4.0

Several covariance evaluations include cross correlations between reactions. These are summarized in Table 10.3.2.

Table 10.3.1: Contents of SCALE 6.3 covariance libraries.

SCALE name	SCALE ID	Data source	Comment
ac-225	89225	ENDF/B-VII.1	
ac-226	89226	ENDF/B-VII.1	
ac-227	89227	ENDF/B-VII.1	
ag-107	47107	BLO approximation data	
ag-109	47109	ENDF/B-VII.1	
ag-110m	1047110	BLO approximation data	
ag-111	47111	BLO approximation data	
al-27	13027	ENDF/B-VII.1	
albound	1013027	ENDF/B-VII.1	Duplicate of al-27
am-240	95240	ENDF/B-VII.1	
am-241	95241	ENDF/B-VII.1 χ covariance JENDL-4.0	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
am-242	95242	SG-26 χ covariance JENDL-4.0	Thermal uncertainty replaced by Mughabghab value
am-242m	1095242	ENDF/B-VII.1	
am-243	95243	ENDF/B-VII.1 χ covariance JENDL-4.0	
am-244	95244	BLO approximation data χ covariance JENDL-4.0	
am-244m	1095244	BLO approximation data	
ar-36	18036	BLO approximation data	
ar-38	18038	BLO approximation data	
ar-40	18040	BLO approximation data	
as-74	33074	BLO approximation data	
as-75	33075	BLO approximation data	
au-197	79197	ENDF/B-VII.1	
b-10	5010	ENDF/B-VII.1	
b-11	5011	ENDF/B-VII.1	
ba-130	56130	BLO approximation data	
ba-132	56132	BLO approximation data	
ba-133	56133	BLO approximation data	
ba-134	56134	BLO approximation data	
ba-135	56135	BLO approximation data	
ba-136	56136	BLO approximation data	
ba-137	56137	BLO approximation data	
ba-138	56138	BLO approximation data	
ba-140	56140	BLO approximation data	
be-7	4007	BLO approximation data	
be-9	4009	ENDF/B-VII.1	
be-beo	5004009	ENDF/B-VII.1	Duplicate of be-9
bebound	3004009	ENDF/B-VII.1	Duplicate of be-9
bi-209	83209	ENDF/B-VII.1	
bk-245	97245	ENDF/B-VII.1	
bk-246	97246	ENDF/B-VII.1	
bk-247	97247	ENDF/B-VII.1	
bk-248	97248	ENDF/B-VII.1	
bk-249	97249	ENDF/B-VII.1	
bk-250	97250	ENDF/B-VII.1	
br-79	35079	BLO approximation data	
br-81	35081	BLO approximation data	
c	6000	ENDF/B-VII.1	
ca	20000	BLO approximation dataca	
ca-40	20040	BLO approximation data	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
ca-42	20042	BLO approximation data	
ca-43	20043	BLO approximation data	
ca-44	20044	BLO approximation data	
ca-46	20046	BLO approximation data	
ca-48	20048	BLO approximation data	
cd	48000	BLO approximation data	
cd-106	48106	BLO approximation data	
cd-108	48108	BLO approximation data	
cd-110	48110	BLO approximation data	
cd-111	48111	BLO approximation data	
cd-112	48112	BLO approximation data	
cd-113	48113	BLO approximation data	
cd-114	48114	BLO approximation data	
cd-115m	1048115	BLO approximation data	
cd-116	48116	BLO approximation data	
ce-136	58136	BLO approximation data	
ce-138	58138	BLO approximation data	
ce-139	58139	BLO approximation data	
ce-140	58140	BLO approximation data	
ce-141	58141	ENDF/B-VII.1	
ce-142	58142	BLO approximation data	
ce-143	58143	BLO approximation data	
ce-144	58144	BLO approximation data	
cf-246	98246	ENDF/B-VII.1	
cf-248	98248	ENDF/B-VII.1	
cf-249	98249	ENDF/B-VII.1	
cf-250	98250	ENDF/B-VII.1	
cf-251	98251	ENDF/B-VII.1	
cf-252	98252	ENDF/B-VII.1	
cf-253	98253	ENDF/B-VII.1	
cf-254	98254	ENDF/B-VII.1	
cl	17000	BLO approximation data	
cl-35	17035	ENDF/B-VII.1	
cl-37	17037	ENDF/B-VII.1	
cm-240	96240	ENDF/B-VII.1	
cm-241	96241	ENDF/B-VII.1	
cm-242	96242	ENDF/B-VII.1	
cm-243	96243	ENDF/B-VII.1	
cm-244	96244	ENDF/B-VII.1	
cm-245	96245	ENDF/B-VII.1	
cm-246	96246	ENDF/B-VII.1	
cm-247	96247	ENDF/B-VII.1	
cm-248	96248	ENDF/B-VII.1	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
cm-249	96249	ENDF/B-VII.1	
cm-250	96250	ENDF/B-VII.1	
co-58	27058	BLO approximation data	
co-58m	1027058	BLO approximation data	
co-59	27059	ENDF/B-VII.1	
cr-50	24050	ENDF/B-VII.1	
cr-52	24052	ENDF/B-VII.1	
cr-53	24053	ENDF/B-VII.1	
cr-54	24054	ENDF/B-VII.1	
cs-133	55133	ENDF/B-VII.1	
cs-134	55134	BLO approximation data	
cs-135	55135	ENDF/B-VII.1	
cs-136	55136	BLO approximation data	
cs-137	55137	BLO approximation data	
cu-63	29063	ENDF/B-VI	
cu-65	29065	ENDF/B-VI	
d	1002	ENDF/B-VII.1	Duplicate of h-2
d-cryo_ortho	4001002	ENDF/B-VII.1	Duplicate of h-2
d-cryo_para	5001002	ENDF/B-VII.1	Duplicate of h-2
dfreegas	8001002	ENDF/B-VII.1	
dy-156	66156	BLO approximation data	
dy-158	66158	BLO approximation data	
dy-160	66160	BLO approximation data	
dy-161	66161	BLO approximation data	
dy-162	66162	BLO approximation data	
dy-163	66163	BLO approximation data	
dy-164	66164	BLO approximation data	
er-162	68162	BLO approximation data	
er-164	68164	BLO approximation data	
er-166	68166	ENDF/B-VII.1	
er-167	68167	ENDF/B-VII.1	
er-168	68168	ENDF/B-VII.1	
er-170	68170	ENDF/B-VII.1	
es-251	99251	ENDF/B-VII.1	
es-252	99252	ENDF/B-VII.1	
es-253	99253	ENDF/B-VII.1	
es-254	99254	ENDF/B-VII.1	
es-254m	1099254	ENDF/B-VII.1	
es-255	99255	ENDF/B-VII.1	
eu-151	63151	BLO approximation data	
eu-152	63152	BLO approximation data	
eu-153	63153	ENDF/B-VII.1	
eu-154	63154	BLO approximation data	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
eu-155	63155	ENDF/B-VII.1	Uses ENDF/B-VII.1 data uncertainty in the thermal range for MT=102
eu-156	63156	BLO approximation data	
eu-157	63157	BLO approximation data	
f-19	9019	ENDF/B-VII.1	
fe-54	26054	ENDF/B-VII.1	
fe-56	26056	ENDF/B-VII.1	
fe-57	26057	ENDF/B-VII.1	
fe-58	26058	ENDF/B-VI	
febound	1026000	ENDF/B-VII.1	Duplicate of fe-56
fm-255	100255	ENDF/B-VII.1	
ga	31000	BLO approximation data	
ga-69	31069	BLO approximation data	
ga-71	31071	BLO approximation data	
gd-152	64152	ENDF/B-VII.1	
gd-153	64153	ENDF/B-VII.1	
gd-154	64154	ENDF/B-VII.1	
gd-155	64155	ENDF/B-VII.1	
gd-156	64156	ENDF/B-VII.1	
gd-157	64157	ENDF/B-VII.1	
gd-158	64158	ENDF/B-VII.1	
gd-160	64160	ENDF/B-VII.1	
ge-70	32070	BLO approximation data	
ge-72	32072	BLO approximation data	
ge-73	32073	BLO approximation data	
ge-74	32074	BLO approximation data	
ge-76	32076	BLO approximation data	
graphite	3006000	ENDF/B-VII.1	Duplicate of c
h	1001	ENDF/B-VII.2 prelim	Duplicate of h1
h-3	1003	BLO approximation data	
h-benzene	6001001	ENDF/B-VII.2 prelim	Duplicate of h-1
h-benzene	5006000	ENDF/B-VII.1	Duplicate of c
h-cryo_ortho	4001001	ENDF/B-VII.2 prelim	Duplicate of h-1
h-cryo_para	5001001	ENDF/B-VII.2 prelim	Duplicate of h-1
h-liquid_ch4	1001001	ENDF/B-VII.2 prelim	Duplicate of h-1
h-poly	9001001	ENDF/B-VII.2 prelim	Duplicate of h-1
h-solid_ch4	2001001	ENDF/B-VII.2 prelim	Duplicate of h-1
h-zrh2	7001001	ENDF/B-VII.2 prelim	Duplicate of h-1
he-3	2003	BLO approximation data	
he-4	2004	ENDF/B-VII.1	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
hf	72000	BLO approximation data	
hf-174	72174	BLO approximation data	
hf-176	72176	BLO approximation data	
hf-177	72177	BLO approximation data	
hf-178	72178	BLO approximation data	
hf-179	72179	BLO approximation data	
hf-180	72180	BLO approximation data	
hfreegas	8001001	ENDF/B-VII.2 prelim	
hg-196	80196	BLO approximation data	
hg-198	80198	BLO approximation data	
hg-199	80199	BLO approximation data	
hg-200	80200	BLO approximation data	
hg-201	80201	BLO approximation data	
hg-202	80202	BLO approximation data	
hg-204	80204	BLO approximation data	
ho-165	67165	BLO approximation data	
ho-166m	1067166	BLO approximation data	
i-127	53127	ENDF/B-VII.1	
i-129	53129	ENDF/B-VII.1	
i-130	53130	BLO approximation data	
i-131	53131	BLO approximation data	
i-135	53135	BLO approximation data	
in	49000	ENDF/B-VI	
in-113	49113	BLO approximation data	
in-115	49115	BLO approximation data	
ir-191	77191	ENDF/B-VII.1	
ir-193	77193	ENDF/B-VII.1	
k	19000	BLO approximation data	
k-39	19039	ENDF/B-VII.1	
k-40	19040	BLO approximation data	
k-41	19041	ENDF/B-VII.1	
kr-78	36078	BLO approximation data	
kr-80	36080	BLO approximation data	
kr-82	36082	BLO approximation data	
kr-83	36083	BLO approximation data	
kr-84	36084	BLO approximation data	
kr-85	36085	BLO approximation data	
kr-86	36086	BLO approximation data	
la-138	57138	BLO approximation data	
la-139	57139	ENDF/B-VII.1	
la-140	57140	BLO approximation data	
li-6	3006	ENDF/B-VII.1	
li-7	3007	ENDF/B-VII.1	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
lu-175	71175	BLO approximation data	
lu-176	71176	BLO approximation data	
mg	12000	BLO approximation data	
mg-24	12024	ENDF/B-VII.1	
mg-25	12025	ENDF/B-VII.1	
mg-26	12026	ENDF/B-VII.1	
mn-55	25055	ENDF/B-VII.1	
mo	42000	BLO approximation data	
mo-100	42100	ENDF/B-VII.1	
mo-92	42092	ENDF/B-VII.1	
mo-94	42094	ENDF/B-VII.1	
mo-95	42095	ENDF/B-VII.1	
mo-96	42096	ENDF/B-VII.1	
mo-97	42097	ENDF/B-VII.1	
mo-98	42098	ENDF/B-VII.1	
mo-99	42099	BLO approximation data	
n-14	7014	BLO approximation data	
n-15	7015	ENDF/B-VII.1	
na-23	11023	ENDF/B-VII.1	
nb-93	41093	ENDF/B-VI	
nb-94	41094	BLO approximation data	
nb-95	41095	ENDF/B-VII.1	
nd-142	60142	BLO approximation data	
nd-143	60143	ENDF/B-VII.1	
nd-144	60144	BLO approximation data	
nd-145	60145	ENDF/B-VII.1	
nd-146	60146	ENDF/B-VII.1	
nd-147	60147	BLO approximation data	
nd-148	60148	ENDF/B-VII.1	
nd-148	60148	BLO approximation data	
nd-150	60150	BLO approximation data	
ni-58	28058	ENDF/B-VII.1	
ni-59	28059	BLO approximation data	
ni-60	28060	ENDF/B-VII.1	
ni-61	28061	ENDF/B-VI	
ni-62	28062	ENDF/B-VI	
ni-64	28064	ENDF/B-VI	
np-234	93234	ENDF/B-VII.1	
np-235	93235	ENDF/B-VII.1	
np-236	93236	ENDF/B-VII.1	
np-237	93237	ENDF/B-VII.1 χ covariance JENDL-4.0	
np-238	93238	ENDF/B-VII.1	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
np-239	93239	ENDF/B-VII.1	
o-16	8016	ENDF/B-VII.1	
o-17	8017	BLO approximation data	
o-beo	5008016	ENDF/B-VII.1	Duplicate of o-16
o-uo2	1008016	ENDF/B-VII.1	Duplicate of o-16
p-31	15031	BLO approximation data	
pa-229	91229	ENDF/B-VII.1	
pa-230	91230	ENDF/B-VII.1	
pa-231	91231	BLO approximation data χ covariance JENDL-4.0	
pa-232	91232	ENDF/B-VII.1	
pa-233	91233	BLO approximation data χ covariance JENDL-4.0	
pb-204	82204	ENDF/B-VII.1	
pb-206	82206	ENDF/B-VII.1	
pb-207	82207	ENDF/B-VII.1	
pb-208	82208	ENDF/B-VII.1	
pd-102	46102	BLO approximation data	
pd-104	46104	BLO approximation data	
pd-105	46105	ENDF/B-VII.1	
pd-106	46106	ENDF/B-VII.1	
pd-107	46107	ENDF/B-VII.1	
pd-108	46108	ENDF/B-VII.1	
pd-110	46110	BLO approximation data	
pm-147	61147	ENDF/B-VII.1	Thermal and resonance range uncertainty values from Mughabghab
pm-148	61148	BLO approximation data	
pm-148m	1061148	BLO approximation data	
pm-149	61149	BLO approximation data	
pm-151	61151	BLO approximation data	
pr-141	59141	ENDF/B-VII.1	
pr-142	59142	BLO approximation data	
pr-143	59143	BLO approximation data	
pu-236	94236	ENDF/B-VII.1	
pu-237	94237	ENDF/B-VII.1	
pu-238	94238	ENDF/B-VII.1	
pu-239	94239	ENDF/B-VII.2 prelim	
pu-240	94240	ENDF/B-VII.1	
pu-241	94241	ENDF/B-VII.1 χ covariance JENDL-4.0	
pu-242	94242	ENDF/B-VII.1	
pu-243	94243	BLO approximation data	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
pu-244	94244	ENDF/B-VII.1	
pu-246	94246	ENDF/B-VII.1	
rb-85	37085	BLO approximation data	
rb-86	37086	BLO approximation data	
rb-87	37087	BLO approximation data	
re-185	75185	ENDF/B-VI	
re-187	75187	ENDF/B-VI	
rh-103	45103	ENDF/B-VII.1	Uses ENDF/B-VII.1 data uncertainty in the thermal range for MT=102
rh-105	45105	BLO approximation data	
ru-100	44100	BLO approximation data	
ru-101	44101	ENDF/B-VII.1	
ru-102	44102	ENDF/B-VII.1	
ru-103	44103	ENDF/B-VII.1	
ru-104	44104	ENDF/B-VII.1	
ru-105	44105	BLO approximation data	
ru-106	44106	ENDF/B-VII.1	
ru-96	44096	BLO approximation data	
ru-98	44098	BLO approximation data	
ru-99	44099	BLO approximation data	
s	16000	BLO approximation data	
s-32	16032	BLO approximation data	
s-33	16033	BLO approximation data	
s-34	16034	BLO approximation data	
s-36	16036	BLO approximation data	
sb-121	51121	BLO approximation data	
sb-123	51123	BLO approximation data	
sb-124	51124	BLO approximation data	
sb-125	51125	BLO approximation data	
sb-126	51126	BLO approximation data	
sc-45	21045	ENDF/B-VI	
se-74	34074	BLO approximation data	
se-76	34076	BLO approximation data	
se-77	34077	BLO approximation data	
se-78	34078	BLO approximation data	
se-79	34079	BLO approximation data	
se-80	34080	BLO approximation data	
se-82	34082	BLO approximation data	
si	14000	ENDF/B-VI	
si-28	14028	ENDF/B-VII.1	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
si-29	14029	ENDF/B-VII.1	
si-30	14030	ENDF/B-VII.1	
si-28 in SiO_2	14728	ENDF/B-VII.1	Duplicate of si-28
si-29 in SiO_2	14729	ENDF/B-0VII.1	Duplicate of si-29
si-30 in SiO_2	14730	ENDF/B-VII.1	Duplicate of si-30
sm-144	62144	BLO approximation data	
sm-147	62147	BLO approximation data	
sm-148	62148	BLO approximation data	
sm-149	62149	ENDF/B-VII.1	Uses ENDF/B-VII.1 data uncertainty in the thermal range for MT=102
sm-149	62149	BLO approximation data	
sm-150	62150	BLO approximation data	
sm-151	62151	ENDF/B-VII.1	Uses ENDF/B-VII.1 data uncertainty in the thermal range for MT=102
sm-152	62152	ENDF/B-VII.1	
sm-153	62153	BLO approximation data	
sm-154	62154	BLO approximation data	
sn-112	50112	BLO approximation data	
sn-113	50113	BLO approximation data	
sn-114	50114	BLO approximation data	
sn-115	50115	BLO approximation data	
sn-116	50116	BLO approximation data	
sn-117	50117	BLO approximation data	
sn-118	50118	BLO approximation data	
sn-119	50119	BLO approximation data	
sn-120	50120	BLO approximation data	
sn-122	50122	BLO approximation data	
sn-123	50123	BLO approximation data	
sn-124	50124	BLO approximation data	
sn-125	50125	BLO approximation data	
sn-126	50126	BLO approximation data	
sr-84	38084	BLO approximation data	
sr-86	38086	BLO approximation data	
sr-87	38087	BLO approximation data	
sr-88	38088	BLO approximation data	
sr-89	38089	BLO approximation data	
sr-90	38090	BLO approximation data	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
ta-181	73181	BLO approximation data	
ta-182	73182	BLO approximation data	
tb-159	65159	BLO approximation data	
tb-160	65160	BLO approximation data	
tc-99	43099	ENDF/B-VII.1	
te-120	52120	BLO approximation data	
te-122	52122	BLO approximation data	
te-123	52123	BLO approximation data	
te-124	52124	BLO approximation data	
te-125	52125	BLO approximation data	
te-126	52126	BLO approximation data	
te-127m	1052127	BLO approximation data	
te-128	52128	BLO approximation data	
te-129m	1052129	BLO approximation data	
te-130	52130	BLO approximation data	
te-132	52132	BLO approximation data	
th-227	90227	ENDF/B-VII.1	
th-228	90228	ENDF/B-VII.1	
th-229	90229	ENDF/B-VII.1	
th-230	90230	ENDF/B-VII.1	
th-231	90231	ENDF/B-VII.1	
th-232	90232	ENDF/B-VII.1 χ covariance JENDL-4.0	
th-233	90233	ENDF/B-VII.1	
th-234	90234	ENDF/B-VII.1	
ti	22000	BLO approximation data	
ti-46	22046	ENDF/B-VII.1	
ti-47	22047	ENDF/B-VII.1	
ti-48	22048	ENDF/B-VII.1	
ti-49	22049	ENDF/B-VII.1	
ti-50	22050	ENDF/B-VII.1	
tl-203	81203	ENDF/B-VII.1	
tl-205	81205	ENDF/B-VII.1	
tm-169	69169	ENDF/B-VII.1	
tm-170	69170	ENDF/B-VII.1	
u-230	92230	ENDF/B-VII.1	
u-231	92231	ENDF/B-VII.1	
u-232	92232	ENDF/B-VII.1	
u-233	92233	ENDF/B-VII.1 χ covariance JENDL-4.0	
u-234	92234	ENDF/B-VII.1	
u-235	92235	ENDF/B-VII.2 prelim	
u-236	92236	ENDF/B-VII.1	

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
u-237	92237	BLO approximation data	
u-238	92238	ENDF/B-VII.1	
u-239	92239	BLO approximation data	
u-240	92240	BLO approximation data	
u-241	92241	BLO approximation data	
u-uo2	1092235	ENDF/B-VII.1	Duplicate of u-235
v	23000	BLO approximation data	
v-50	23050	BLO approximation data	Duplicate of v
v-51	23051	BLO approximation data	Duplicate of v
w	74000	BLO approximation data	
w-180	74180	ENDF/B-VII.1	
w-182	74182	ENDF/B-VII.1	
w-183	74183	ENDF/B-VII.1	
w-184	74184	ENDF/B-VII.1	
w-186	74186	ENDF/B-VII.1	
xe-123	54123	BLO approximation data	
xe-124	54124	BLO approximation data	
xe-126	54126	BLO approximation data	
xe-128	54128	BLO approximation data	
xe-129	54129	BLO approximation data	
xe-130	54130	BLO approximation data	
xe-131	54131	ENDF/B-VII.1	
xe-132	54132	ENDF/B-VII.1	
xe-133	54133	BLO approximation data	
xe-134	54134	ENDF/B-VII.1	
xe-135	54135	BLO approximation data	
xe-136	54136	BLO approximation data	
y-89	39089	ENDF/B-VII.1	
y-90	39090	BLO approximation data	
y-91	39091	BLO approximation data	
zr	40000	BLO approximation data	
zr-90	40090	ENDF/B-VII.1	
zr-91	40091	ENDF/B-VII.1	
zr-92	40092	ENDF/B-VII.1	
zr-93	40093	ENDF/B-VII.1	
zr-94	40094	ENDF/B-VII.1	
zr-95	40095	ENDF/B-VII.1	
zr-96	40096	ENDF/B-VII.1	
zr-90-zr5h8	1040090	ENDF/B-VII.1	Duplicate of zr-90
zr-91-zr5h8	1040091	ENDF/B-VII.1	Duplicate of zr-91
zr-92-zr5h8	1040092	ENDF/B-VII.1	Duplicate of zr-92
zr-93-zr5h8	1040093	ENDF/B-VII.1	Duplicate of zr-93
zr-94-zr5h8	1040094	ENDF/B-VII.1	Duplicate of zr-94

continues on next page

Table 10.3.1 – continued from previous page

SCALE name	SCALE ID	Data source	Comment
zr-95-zr5h8	1040095	ENDF/B-VII.1	Duplicate of zr-95
zr-96-zr5h8	1040096	ENDF/B-VII.1	Duplicate of zr-96

Table 10.3.2: Covariance data with cross-correlations between nuclide reactions.

Nuclide 1	Reaction 1	Nuclide 2	Reaction 2
^{239}Pu	Fission	^{6}Li	Triton production
^{239}Pu	Fission	^{197}Au	Capture
^{239}Pu	Fission	^{235}U	Fission
^{239}Pu	Fission	^{238}U	Fission
^{235}U	Fission	^{197}Au	Capture
^{235}U	Fission	^{6}Li	Triton production
^{238}U	Capture	^{197}Au	Capture
^{238}U	Capture	^{235}U	Fission

10.3.5 SCALE 6.1 44-GROUP COVARIANCE LIBRARY

The older 44-group covariance library distributed with SCALE 6.0 and SCALE 6.1 is included with this distribution for backwards compatibility. The 44-group covariance library provides uncertainty data for a total of 401 materials, including some duplication for materials with multiple thermal scattering kernels. However, the 44-group library was created prior to the official release of ENDF/B-VII.1. Therefore, it is recommended that the 56- or 252-group covariances be used rather than the 44-group. As discussed in Sect. 10.3.1, it is assumed that covariances taken from one data evaluation such as ENDF/B-VI or JENDL-3.3 can also be applied to other evaluations of the same data, such as ENDF/B-VII. If this is done judiciously for cases in which the nuclear data evaluations are similar, then the covariances taken from one source should be a reasonable representation of uncertainties for the other evaluations. Among the materials in the SCALE 44-group library with covariances taken from high-fidelity nuclear data evaluations are the following:

a) ENDF/B-VII evaluations *(includes both VII.0 and pre-release covariances proposed for VII.1, but no official ENDF/B-VII.1)*:

Au, ^{209}Bi, ^{59}Co, ^{152,154,155,156}Gd, ^{191,193}I, ^{7}Li, ^{23}Na, ^{93}Nb, ^{58}Ni, ^{99}Tc,^{232}Th, ^{48}Ti, ^{239}Pu, ^{233,235,238}U,V

(b) ENDF/B-VI evaluations:

Al, ^{241}Am, ^{10}B, ^{12}C, ^{50,52,53,54}Cr, ^{63,65}Cu, ^{54,56,57}Fe, In, ^{55}Mn, ^{60,61,62,64}Ni, ^{206,207,208}Pb, ^{242}Pu, ^{28,29}Si

(c) JENDL-3.3 evaluations:

^{11}B, ^{1}H, ^{16}O, ^{240,241}Pu

Two modifications were also made to the ENDF/B-VII evaluated nubar covariances. These nubar uncertainties are believed to be more realistic. The ENDF/B-VII.0 ^{235}U thermal nubar uncertainty of 0.71% was revised to the JENDL-3.3 value of 0.31%. In addition, the thermal nubar certainty in the pre-released ENDF/B-VII.1 ^{233}U evaluation was modified to the value in a recent ORNL data evaluation [COVLIBLWRD08]. This ORNL ^{233}U cross section evaluation also provided the thermal and resonance cross sections for the prereleased

ENDF/B-VII.1 data. The ENDF/B-VII.1 pre-release nubar data for ^{239}Pu was incomplete when the 44-group covariance library was generated, so ^{239}Pu nubar data are included from ENDF/B-V, the most current data available at that time. This value is much higher than the current estimated uncertainty in ^{239}Pu nubar. The basic ENDF/B uncertainty files that were changed are described in Table 10.3.3.

Several modifications were also made to the uncertainties obtained from the BLO data. The BLO thermal uncertainties for ^{1}H capture and elastic and for ^{16}O elastic were modified to the JENDL-3.3 values of 0.5% and 0.1%, respectively. Similarly, the uncertainty in the ^{10}B (n,alpha) thermal cross section was modified to the ENDF/B-VI value of about 0.2%, since this is more consistent with the Mughabghab integral uncertainty. The uncertainty in the ^{149}Sm resonance capture integral is not provided in the 2006 edition of Mughabghab's text; therefore it was set to the value of 5.7% which was obtained from an earlier tabulation by Mughabghab [COVLIBMug03].

Table 10.3.3: Summary of changes made to covariance evaluations for the 44-group library.

ENDF/B-VII.1 pre-release ^{239}Pu	Data were incomplete at time of library generation, so ENDF/B-V data were used for nubar.
ENDF/B-VII ^{235}U	Thermal nubar modified to JENDL-3.3 value
ENDF/B-VII ^{233}U	Thermal nubar modified to value from ORNL internal evaluation
ENDF/B-VI ^{241}Am	Thermal uncertainties were added to total cross section (set equal to capture uncertainties)
ENDF/B-VI ^{28}Si, ^{29}Si, ^{30}Si, ^{206}Pb, ^{57}Fe	In elastic scatter uncertainty, corrected cross reference to MT=102 from original value of MT=1.02
ENDF/B-VI ^{208}Pb, ^{207}Pb	Removed MT=3 due to inconsistency with other MT values, resulting in very large uncertainty predictions

At the time of the preparation of the 44-group covariance library, ENDF/B did not provide fission spectra uncertainty estimates. The methodology used to construct these data for the 44-group covariance library is described in Broadhead and Wagschal [COVLIBBW04]. In this approach, the fission spectrum is represented as either a Watt or Maxwellian distribution. These energy distributions are widely used to represent fission spectra and have been commonly employed in many ENDF/B evaluations. For example, Watt and Maxwellian expressions were used almost exclusively to describe fission spectra in ENDF/B-V and also for many ENDF/B-VI evaluations. More recent evaluations for some important fissionable nuclides have replaced the simple Watt and Maxwellian analytical expressions by distributions such as the Madland-Nix spectrum obtained from more phenomenological nuclear fission models. However, it is assumed here that uncertainties based on an appropriate Watt or Maxwellian representation of the fission spectrum can be transferred to the actual fission spectra contained in the different multigroup cross section libraries.

The methodology in Broadhead and Wagschal [COVLIBBW04] determines energy-dependent covariances from uncertainties and correlations in the *a* and *b* parameters for the Watt spectrum or the *T* parameter for a Maxwellian spectrum, appearing the analytical expressions given below:

Watt Spectrum: $\chi(E) = \frac{e^{-E/a}}{I} \sinh(\sqrt{bE})$

Maxwellian Spectrum: $\chi(E) = \frac{\sqrt{E}e^{-E/T}}{I}$

In these expressions, the parameter "I" is the normalization factor required to normalize the integrated spectrum to unity. The value of "I" is fixed by the values of the other parameters. Due to the normalization

constraint, the fission spectrum covariance includes anti-correlations. The assumed fission spectra parameters and uncertainties are given in Maerker, Marable, and Wagschal 1980 [COVLIBMMW80] and in Howerton and Doyas 1971 [COVLIBHD71].

Table 10.3.4 shows that fission spectra covariances are not provided for all fissionable materials in the SCALE multigroup cross sections. Table 10.3.5 lists the fissionable nuclides without fission spectra covariances on the 44-group covariance library.

Table 10.3.4: Source of fission spectrum parameters and uncertainties

Watt spectrum	a or T	b	Source of parameters	σ_a or σ_T (%)	σ_b (%)	Source of uncertainty
^{235}U	0.988	2.249	ENDF/B-V	1.2	5.9	TANS[16]
^{238}U	0.881	3.401	ENDF/B-V	1.2	5.9	TANS[16]
^{233}U	0.977	2.546	ENDF/B-V	1.2	5.9	TANS[16]
^{239}Pu	0.966	2.842	ENDF/B-V	1.2	5.9	TANS[16]
^{232}Th	1.0888	1.6871	ENDF/B-V	1.2	5.9	TANS[16]
^{252}Cf	1.025	2.926	ENDF/B-V	1.2	5.9	TANS[16]
Maxwellian Spectrum						
^{238}Pu	1.330	-	ENDF/B-V	3.01	-	NSE[17]
^{240}Pu	1.346	-	ENDF/B-V	2.97	-	NSE[17]
^{241}Pu	1.3597	-	ENDF/B-V	2.50	-	NSE[17]
^{242}Pu	1.337	-	ENDF/B-V	5.24	-	NSE[17]

Table 10.3.5: Fissionable nuclides with missing fission spectrum uncertainty data in covariance library.

^{241}Am	^{244}Cm	^{238}Pu
^{242}Am	^{245}Cm	^{243}Pu
^{243}Am	^{246}Cm	^{244}Pu
^{249}Bk	^{247}Cm	^{230}Th
^{249}Cf	^{248}Cm	^{232}U
^{250}Cf	^{237}Np	^{234}U
^{251}Cf	^{238}Np	^{236}U
^{253}Cf	^{239}Np	^{237}U
^{242}Cm	^{231}Pa	
^{243}Cm	^{233}Pa	

Table 10.3.6 describes the contents of the library using the following nomenclature:

1. ENDF/B-VII.0: evaluated covariance data released with ENDF/B-VII.0
2. ENDF/B-VII-p: recently evaluated data proposed for future release of ENDF/B-VII.1
3. ENDF/B-VI: evaluated covariance data released with ENDF/B-VI
4. JENDL-3.3: evaluated covariance data in JENDL-3.3
5. BLO approximate data: lo-fi covariances from BLO project

6. BLO LANL evaluation: LANL R-matrix evaluation from BLO project
7. SG-26: approximate covariances from WPEC Subgroup-26

Table 10.3.6: Contents of SCALE 6.1 44-group covariance library.

SCALE name	Data source	Comments
ac-225	BLO approximate data	
ac-226	BLO approximate data	
ac-227	BLO approximate data	
ag-107	BLO approximate data	
ag-109	BLO approximate data	
ag-110m	BLO approximate data	
ag-111	BLO approximate data	
al-27	ENDF/B-VI	
am-241	ENDF/B-VI	MT=452 added corrections for total and elastic)
am-242	SG-26	Thermal uncertainty replaced by Mughabghab value
am-242m	SG-26	Thermal uncertainty replaced by Mughabghab value
am-243	BLO approximate data	
am-244	BLO approximate data	
am-244m	BLO approximate data	
ar-36	BLO approximate data	
ar-38	BLO approximate data	
ar-40	BLO approximate data	
as-74	BLO approximate data	
as-75	BLO approximate data	
au-197	ENDF/B-VII-p	Pre-released evaluation proposed for ENDF/B-VII.1
b-10	BLO LANL evaluation +ENDF/B-VI	LANL high-fidelity covariance, with ENDF/B-VI for thermal
b-11	JENDL 3.3	
ba-130	BLO approximate data	
ba-132	BLO approximate data	
ba-133	BLO approximate data	
ba-135	BLO approximate data	
ba-136	BLO approximate data	
ba-137	BLO approximate data	
ba-138	BLO approximate data	
ba-140	BLO approximate data	
be-7	BLO approximate data	
be-9	BLO approximate data	
Bebound	BLO approximate data	Duplicate of ^{9}Be
bi-209	ENDF/B-VII-p	Pre-released evaluation proposed for ENDF/B-VII.1

continues on next page

Table 10.3.6 – continued from previous page

bk-249	BLO approximate data	
bk-250	BLO approximate data	
br-79	BLO approximate data	
br-81	BLO approximate data	
C	ENDF/B-VI	
C-graphite	ENDF/B-VI	Duplicate of carbon
Ca	BLO approximate data	
ca-40	BLO approximate data	
ca-42	BLO approximate data	
ca-43	BLO approximate data	
ca-44	BLO approximate data	
ca-46	BLO approximate data	
ca-48	BLO approximate data	
Cd	BLO approximate data	
cd-106	BLO approximate data	
cd-108	BLO approximate data	
cd-110	BLO approximate data	
cd-111	BLO approximate data	
cd-112	BLO approximate data	
cd-113	BLO approximate data	
cd-114	BLO approximate data	
cd-115m	BLO approximate data	
cd-116	BLO approximate data	
cd-136	BLO approximate data	
cd-138	BLO approximate data	
cd-139	BLO approximate data	
cd-140	BLO approximate data	
cd-141	BLO approximate data	
cd-142	BLO approximate data	
ce-143	BLO approximate data	
ce-144	BLO approximate data	
cf-249	BLO approximate data	
cf-250	BLO approximate data	
cf-251	BLO approximate data	
cf-252	BLO approximate data	
cf-253	BLO approximate data	
cf-254	BLO approximate data	
Cl	BLO approximate data	
cl-35	BLO approximate data	
cl-37	BLO approximate data	
cm-241	BLO approximate data	Thermal uncertainty
cm-242	SG-26	Mughabghab value
cm-243	SG-26	Thermal uncertainty replaced by Mughabghab value

continues on next page

Table 10.3.6 – continued from previous page

cm-244	SG-26	Thermal uncertainty replaced by Mughabghab value
cm-245	SG-26	Thermal uncertainty replaced by Mughabghab value
cm-246	BLO approximate data	
cm-247	BLO approximate data	
cm-248	BLO approximate data	
cm-249	BLO approximate data	
cm-250	BLO approximate data	
co-58 co-58m co-59	BLO approximate data BLO approximate data ENDF/B-VII-p	Pre-released evaluation proposed for ENDF/B-VII.1
cr-50	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
cr-52	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
cr-53	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
cr-54	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
cs-133	BLO approximate data	
cs-134	BLO approximate data	
cs-135	BLO approximate data	
cs-136	BLO approximate data	
cs-137	BLO approximate data	
cu-63	ENDF/B-VI	
cu-65	ENDF/B-VI	
dy-156	BLO approximate data	
dy-158	BLO approximate data	

continues on next page

Table 10.3.6 – continued from previous page

dy-160	BLO approximate data	
dy-161	BLO approximate data	
dy-162	BLO approximate data	
dy-163	BLO approximate data	
dy-164	BLO approximate data	
er-162	BLO approximate data	
er-164	BLO approximate data	
er-166	BLO approximate data	
er-167	BLO approximate data	
er-168	BLO approximate data	
er-170	BLO approximate data	
es-253	BLO approximate data	
es-254	BLO approximate data	
es-255	BLO approximate data	
eu-151	BLO approximate data	
eu-152	BLO approximate data	
eu-153	BLO approximate data	
eu-154	BLO approximate data	
eu-155 eu-156 eu-157	BLO approximate data BLO approximate data BLO approximate data	
f-19	BLO approximate data	
fe-54	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
fe-56	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
fe-57	ENDF/B-VI	Error in file corrected LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.

continues on next page

Table 10.3.6 – continued from previous page

fe-58 fm-255	ENDF/B-VI BLO approximate data	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results. New material not in previous SCALE 5.1 covariance libraries.
Ga ga-69 ga-71	BLO approximate data BLO approximate data BLO approximate data	
gd-152 gd-153	ENDF/B-VII.0 BLO approximate data	
gd-154	ENDF/B-VII.0	
gd-155	ENDF/B-VII.0	
gd-156	ENDF/B-VII.0	
gd-157	ENDF/B-VII.0	
gd-158	ENDF/B-VII.0	
gd-160	ENDF/B-VII.0	
ge-70	BLO approximate data	
ge-72	BLO approximate data	
ge-73	BLO approximate data	
ge-74	BLO approximate data	
ge-76	BLO approximate data	
h-1	BLO LANL evaluation +JENDL 3.3	LANL covariance above 5 keV; JENDL values below 5 keV
h-ZrH	BLO LANL evaluation +JENDL 3.	Duplicate of ^{1}H
h-poly	BLO LANL evaluation +JENDL 3.	Duplicate of ^{1}H
Hfreegas	BLO LANL evaluation +JENDL 3.	Duplicate of ^{1}H
h-2	BLO approximate data	
Dfreegas	BLO approximate data	Duplicate of ^{2}H
h-3	BLO approximate data	
he-3	BLO approximate data	
he-4	BLO approximate data	
Hf	BLO approximate data	
hf-174	BLO approximate data	
hf-176	BLO approximate data	
fh-177	BLO approximate data	
hf-178	BLO approximate data	
hf-179	BLO approximate data	
hf-180	BLO approximate data	
hg-196	BLO approximate data	
hg-198	BLO approximate data	

continues on next page

Table 10.3.6 – continued from previous page

hg-199	BLO approximate data	
hg-200	BLO approximate data	
hg-201	BLO approximate data	
hg-202	BLO approximate data	
hg-204	BLO approximate data	
ho-165	BLO approximate data	
i-127	BLO approximate data	
i-129	BLO approximate data	
i-130	BLO approximate data	
i-131	BLO approximate data	
i-135	BLO approximate data	
In	ENDF/B-VI	
in-113 in-115	BLO approximate data BLO approximate data	
ir-191	ENDF/B-VII.0	
ir-193	ENDF/B-VII.0	
K	BLO approximate data	
k-39	BLO approximate data	
k-40	BLO approximate data	
k-41	BLO approximate data	
kr-78	BLO approximate data	
kr-80	BLO approximate data	
kr-82	BLO approximate data	
kr-83	BLO approximate data	
kr-84	BLO approximate data	
kr-85	BLO approximate data	
kr-86	BLO approximate data	
la-138 la-139	BLO approximate data BLO approximate data	
la-140	BLO approximate data	
li-6	BLO-LANL evaluation	
li-7	ENDF/B-VII.0	
lu-175	BLO approximate data	
lu-176	BLO approximate data	
Mg	BLO approximate data	
mg-24	BLO approximate data	
mg-25	BLO approximate data	
mg-26	BLO approximate data	
mn-55	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.

continues on next page

Table 10.3.6 – continued from previous page

Mo	BLO approximate data	
mo-92	BLO approximate data	
mo-94	BLO approximate data	
mo-95	BLO approximate data	
mo-96	BLO approximate data	
mo-97	BLO approximate data	
mo-98 mo-99 mo-100	BLO approximate data BLO approximate data BLO approximate data	
n-14	BLO approximate data	
n-15	BLO approximate data	
na-23	ENDF/B-VII-p	Pre-released evaluation proposed for ENDF/B-VII.1
nb-93	ENDF/B-VII-p	Pre-released evaluation proposed for ENDF/B-VII.1
nb-94	BLO approximate data	
nb-95	BLO approximate data	
nd-142	BLO approximate data	
nd-143	BLO approximate data	
nd-144	BLO approximate data	
nd-145	BLO approximate data	
nd-146	BLO approximate data	
nd-147	BLO approximate data	
nd-148	BLO approximate data	
nd-150	BLO approximate data	
ni-58 ni-59	ENDF/B-VII-p BLO approximate data	Pre-released evaluation proposed for ENDF/B-VII.1
ni-60	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
ni-61	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
ni-62	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.

continues on next page

Table 10.3.6 – continued from previous page

ni-64	ENDF/B-VI	LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
np-235 np-236 np-237	BLO approximate data BLO approximate data SG-26	Thermal uncertainty replaced by Mughabghab value
np-238	BLO approximate data	
np-239	BLO approximate data	
o-16	JENDL 3.3+BLO	BLO covariances from LANL used above 5 keV
o-17	BLO approximate data	
p-31	BLO approximate data	
pa-231 pa-232	BLO approximate data BLO approximate data	
pa-233	BLO approximate data	
pb-204 pb-206	BLO approximate data ENDF/B-VI	Error in file corrected
pb-207	ENDF/B-VI	MT=3 removed, Error in file corrected
bp-208	ENDF/B-VI	MT=3 removed, Error in file corrected
pd-102	BLO approximate data	
pd-104	BLO approximate data	
pd-105	BLO approximate data	
pd-106	BLO approximate data	
pd-107	BLO approximate data	
pd-108	BLO approximate data	
pd-110	BLO approximate data	
pm-147	BLO approximate data	
pm-148	BLO approximate data	
pm-148m	BLO approximate data	
pm-149	BLO approximate data	
pm-151	BLO approximate data	
pr-141	BLO approximate data	
pr-142	BLO approximate data	
pr-143	BLO approximate data	
pu-236 pu-237 pu-238	BLO approximate data BLO approximate data SG-26	Thermal uncertainty replaced by Mughabghab value
pu-239	ENDF/B-VII-p	Pre-released evaluation proposed for ENDF/B-VII.1; nubar data from ENDF/B-V Cross nuclide-to-nuclide matrices present; covariances due to fission cross sections / nubar for each nuclide Table 10.3.2

continues on next page

Table 10.3.6 – continued from previous page

pu-240	JENDL 3.3	Cross nuclide-to-nuclide matrices present; covariances due to fission cross sections / nubar for each nuclide (Table 10.3.2
pu-241	JENDL 3.3	Cross nuclide-to-nuclide matrices present; covariances due to fission cross sections / nubar for each nuclide (Table 10.3.2)
pu-242	ENDF/B-VI	
pu-243	BLO approximate data	
pu-244	BLO approximate data	
pu-246	BLO approximate data	
rb-85	BLO approximate data	
rb-86	BLO approximate data	
rb-87	BLO approximate data	
re-185	ENDF/B-VI	MT=2 added from Mughabghab. LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
re-187	ENDF/B-VI	MT=2 added from Mughabghab. LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
rh-103	BLO approximate data	
rh-105	BLO approximate data	
ru-96	BLO approximate data	
ru-98	BLO approximate data	
ru-103	BLO approximate data	
ru-99	BLO approximate data	
ru-100	BLO approximate data	
ru-101	BLO approximate data	
ru-102	BLO approximate data	
ru-104	BLO approximate data	
ru-105	BLO approximate data	
ru-106	BLO approximate data	
S	BLO approximate data	
s-32	BLO approximate data	
s-33	BLO approximate data	
s-34	BLO approximate data	
s-36	BLO approximate data	
sb-123	BLO approximate data	
sb-124	BLO approximate data	
sb-125	BLO approximate data	

continues on next page

Table 10.3.6 – continued from previous page

sb-126	BLO approximate data	
sc-45	ENDF/B-VI	
se-74	BLO approximate data	
se-76	BLO approximate data	
se-77	BLO approximate data	
se-78 se-79	BLO approximate data BLO approximate data	
se-80	BLO approximate data	
se-82	BLO approximate data	
Si	ENDF/B-VI	
si-28	ENDF/B-VI	Error in file corrected LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
si-29	ENDF/B-VI	Error in file corrected LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
si-30	ENDF/B-VI	Error in file corrected LB=8 representation caused problematic representation of cross section uncertainty due to use of fine energy group structure. Tests were performed to determine how to handle this problem. LB=8 data were removed in the final results.
sm-144	BLO approximate data	
sm-147	BLO approximate data	
sm-148	BLO approximate data	
sm-149	BLO approximate data	Resonance range uncertainty from Kawano 2008
sm-150	BLO approximate data	
sm-151	BLO approximate data	
sm-152	BLO approximate data	
sm-153	BLO approximate data	
sm-154	BLO approximate data	
sn-112	BLO approximate data	
sn-113	BLO approximate data	
sn-114	BLO approximate data	
sn-115	BLO approximate data	
sn-116	BLO approximate data	
sn-117	BLO approximate data	
sn-118	BLO approximate data	

continues on next page

Table 10.3.6 – continued from previous page

sn-119	BLO approximate data	
sn-120	BLO approximate data	
sn-122	BLO approximate data	
sn-123	BLO approximate data	
sn-124	BLO approximate data	
sn-125	BLO approximate data	
sr-84	BLO approximate data	
sr-86	BLO approximate data	
sr-87	BLO approximate data	
sr-88	BLO approximate data	
sr-89	BLO approximate data	
sr-90	BLO approximate data	
ta-181	BLO approximate data	
ta-182	BLO approximate data	
tb-159	BLO approximate data	
tb-160	BLO approximate data	
tc-99	ENDF/B-VII.0	
te-120	BLO approximate data	
te-122	BLO approximate data	
te-123	BLO approximate data	
te-124	BLO approximate data	
te-125	BLO approximate data	
te-126	BLO approximate data	
te-127m	BLO approximate data	
te-128	BLO approximate data	
te-129m	BLO approximate data	
te-130	BLO approximate data	
th-227	BLO approximate data	
th-228	BLO approximate data	
th-229	BLO approximate data	
th-230	BLO approximate data	
th-232	ENDF/B-VII.0	
th-233	BLO approximate data	Cross nuclide-to-nuclide matrices present; covariances due to fission cross sections / nubar for each nuclide (Table 10.3.2) Pre-released evaluation proposed for ENDF/B-VII.1
th-234	BLO approximate data	
Ti	BLO approximate data	
ti-46	BLO approximate data	
ti-47	BLO approximate data	
ti-48	ENDF/B-VII-p	
ti-49	BLO approximate data	
ti-50	BLO approximate data	
u-232	BLO approximate data	

continues on next page

Table 10.3.6 – continued from previous page

u-233	ENDF/B-VII-p	Pre-released evaluation proposed for ENDF/B-VII.1; nubar uncertainty from Ref. 14Cross nuclide-to-nuclide matrices present; covariances due to fission cross sections / nubar for each nuclide (Table 10.3.2).
u-234	SG-26	Thermal uncertainty replaced by Mughabghab value
u-235	ENDF/B-VII-p	Pre-released evaluation proposed for ENDF/B-VII.1; nubar uncertainty from JENDL-3.1 Cross nuclide-to-nuclide matrices present; covariances due to fission cross sections / nubar for each nuclide (Table 10.3.2).
u-236	SG-26	Thermal uncertainty replaced by Mughabghab value
u-237	BLO approximate data	
u-238 u-239 u-240 u-241	ENDF/B-VII-p BLO approximate data BLO approximate data BLO approximate data	Pre-released evaluation proposed for ENDF/B-VII.1 Cross nuclide-to-nuclide matrices present; covariances due to fission cross sections / nubar for each nuclide (Table 10.3.2).
V	ENDF/B-VII-p	Pre-released evaluation proposed for ENDF/B-VII.1
W	BLO approximate data	
w-182	BLO approximate data	
w-183	BLO approximate data	
w-184	BLO approximate data	
w-186	BLO approximate data	
xe-123	BLO approximate data	
xe-124	BLO approximate data	
xe-126	BLO approximate data	
xe-128	BLO approximate data	
xe-129	BLO approximate data	
xe-130	BLO approximate data	
xe-131	BLO approximate data	
xe-132	BLO approximate data	
xe-134	BLO approximate data	
xe-135	BLO approximate data	
xe-136	BLO approximate data	
y-89	ENDF/B-VI	
y-90	BLO approximate data	
y-91	BLO approximate data	
Zr	BLO approximate data	
zr-90	BLO approximate data	
zr-91	BLO approximate data	

continues on next page

Table 10.3.6 – continued from previous page

zr-92	BLO approximate data	
zr-93	BLO approximate data	
zr-94	BLO approximate data	
zr-95	BLO approximate data	
zr-96	BLO approximate data	

Table 10.3.7: Covariance data with cross correlations between nuclide reactions.

Nuclide 1	Reaction 1	Nuclide 2	Reaction 2
^{240}Pu	Fission	^{239}Pu	Fission
^{240}Pu	Fission	^{233}U	Fission
^{240}Pu	Fission	^{238}U	Fission
^{241}Pu	Fission	^{239}Pu	Fission
^{241}Pu	Fission	^{240}Pu	Fission
^{241}Pu	Fission	^{233}U	Fission
^{241}Pu	Fission	^{235}U	Fission
^{241}Pu	Fission	^{238}U	Fission
^{235}U	Fission	^{240}Pu	Fission

11. UTILITY MODULES FOR SCALE LIBRARIES

M.L. Williams

Historically the SCALE Code System has included a number of standalone, executable utility modules developed mainly for performing operations on various types of multigroup nuclear data libraries and on the standard composition library used in SCALE. In legacy versions of SCALE, some utility modules were directly executed during computational sequences; but in the modern version of SCALE the standalone executables are no longer used in computation sequences — their functions have been incorporated into a more unified and integrated framework. Several of the earlier utility modules were deprecated in the SCALE 6.2 release; but others are retained and are still quite useful for nuclear data manipulations, editing, and checking. Most of these utilities are taken directly from the AMPX code system. Sect. 11.1 provides a listing of the AMPX modules distributed with SCALE, along with a brief description of each module's function. User input for these modules can be found in the AMPX documentation that accompanies SCALE. Several additional utility modules included in the SCALE code package are also described in this section. Finally, many AMPX and SCALE utility modules use the FIDO input processing procedure, and this is described in the FIDO chapter.

11.1 AMPX LIBRARY UTILITY MODULES

D. Wiarda, A. Holcomb, L. M. Petrie

Abstract

The purpose of this section is to document selected AMPX modules that can benefit the analyst interested in editing, converting, or combining cross-section libraries normally used by the SCALE system modules. The input description for these codes is provided in the documentation of the AMPX nuclear data processing code system that is distributed with SCALE package.

11.1.1 INTRODUCTION

AMPX is a modular system [UTILS-WWCD15] that generates continuous energy (CE) and multigroup (MG) cross section data from evaluated nuclear data files such as ENDF/B. All the nuclear data libraries distributed with SCALE have been processed using AMPX. In addition to data processing modules, AMPX also includes a number of useful utility modules for checking, manipulating, and editing the libraries in SCALE. This section lists and briefly describes some of the AMPX utility codes that may be useful to SCALE users. Input instructions for these codes can be found the AMPX code documentation, which is distributed with the SCALE code package. Additional AMPX modules of interest may also found in the documentation.

11.1.2 AJAX: MODULE TO MERGE, COLLECT, ASSEMBLE, REORDER, JOIN, AND/OR COPY SELECTED DATA FROM AMPX MASTER LIBRARIES

AJAX (**A**utomatic **J**oining of **A**MPX **X**-Sections) is a module to combine data from different AMPX libraries. Options are provided to allow merging from any number of files.

11.1.3 ALPO: MODULE TO CONVERT AMPX LIBRARIES INTO ANISN FORMAT

ALPO (**A**NISN **L**IBRARY **P**RODUCTION **O**PTION) is a module for converting AMPX working libraries into the library format used by the legacy discrete ordinates transport codes ANISN and DORT/TORT contained in the DOORS package cite:*rhoades_doors_1998*.

11.1.4 CADILLAC: MODULE TO MERGE MULTIPLE COVARIANCE DATA FILES

CADILLAC (**C**ombine **A**ll **D**ata **I**dentifiers **L**isted in **L**ogical **A**MPX **C**overx-format) is a module that can be used to combine multiple covariance data files in COVERX format into a single covariance data file. The material IDs can be changed as needed by the user.

11.1.5 COGNAC: MODULE TO CONVERT COVARIANCE DATA FILES IN COVERX FORMAT

COGNAC (**C**onversion **O**perations for **G**roup-dependent **N**uclides in **A**MPX **C**overx-format) is a module that can be used to convert a single COVERX-formatted data file from bcd format to binary. Also, COGNAC can be used to convert from binary to bcd, binary to binary, and bcd to bcd.

11.1.6 LAVA: MODULE TO MAKE AN AMPX WORKING LIBRARY FROM AN ANISN LIBRARY

LAVA (**L**et **A**NISN **V**isit **A**MPX) is a module that can convert an ANISN formatted library (neutron, gamma, or coupled neutron-gamma) to an AMPX working library that can be used in XSDRNPM.

11.1.7 MALOCS: MODULE TO COLLAPSE AMPX MASTER CROSS-SECTION LIBRARIES

MALOCS (**M**iniature **A**MPX **L**ibrary **O**f **C**ross **S**ections) is a module to collapse AMPX master cross-section libraries. The module can be used to collapse neutron, gamma-ray, or coupled neutron-gamma master libraries.

11.1.8 PALEALE: MODULE TO LIST INFORMATION FROM AMPX LIBRARIES

PALEALE lists selected data by nuclide, reaction, data-type from AMPX master and working libraries.

11.1.9 RADE: MODULE TO CHECK AMPX CROSS-SECTION LIBRARIES

RADE (**R**ancid **A**MPX **D**ata **E**xposer) is provided to check AMPX- and ANISN-formatted multigroup libraries. It will check neutron, gamma, or coupled neutron-gamma libraries.

11.1.10 TOC: MODULE TO PRINT AN AMPX LIBRARY TABLE OF CONTENTS

Program TOC is a utility program to print a sorted table of contents of an AMPX cross section library. It is designed to be run interactively, with the cross section library specified as the argument.

11.2 WORKER: SCALE SYSTEM MODULE FOR CREATING AND MODIFYING WORKING-FORMAT LIBRARIES

L. M. Petrie

ABSTRACT

WORKER is a standalone utility module used to convert AMPX master-formatted or working-formatted multigroup cross section libraries into a single working library for SCALE transport calculations performed by XSDRNPM, KENO V.a and KENO-VI, and other modules. Beginning with SCALE 6.2, WORKER is no longer used for calculations performed with SCALE sequences because this function has been absorbed into

the XSProc module. This document gives instructions on how to use the WORKER program as a standalone module.

ACKNOWLEDGMENTS

Several former ORNL staff made valuable contributions to the WORKER development. The authors acknowledge the contributions of former ORNL staff members S. Goluoglu, N. M. Greene, and D. F. Hollenbach.

11.2.1 INTRODUCTION

Historically the AMPX nuclear data processing system [WORKER1] has defined two formats for the multigroup (MG) libraries used in SCALE. The "master" formatted library contains more general information than normally required for radiation transport calculations. A master library includes such information as 2D transfer arrays for all inelastic levels, temperature-dependent thermal-scattering kernels, and Bondarenko self-shielding data. A "working" formatted library contains only the subset of these data needed for radiation transport calculations. A working library contains only a single, combined 2D transfer array [i.e., sum of elastic, all discrete inelastic levels, (n,2n), etc.], and the temperature-dependent data normally have been interpolated to the appropriate temperatures. In SCALE 6.2, more general formats for nuclear data libraries are supported, and the MG libraries distributed with SCALE are no longer restricted by the inflexible Master library format. However the WORKER module is retained in SCALE 6.2 for manipulation of legacy formatted libraries.

WORKER is a standalone program that reads input nuclear data library files and produces a single working library file that can be read by transport modules such as XSDRNPM, KENO-VI, NEWT, CENTRM, and Monaco. Prior to SCALE 6.2, the distributed MG libraries were stored in master format, and WORKER was executed during every MG sequence to read the master library file and write an output working library file subsequently read by the transport codes run by the sequence. In modern versions of AMPX and SCALE (e.g., SCALE 6.2 and later), the differences in the master and working formats are superfluous, and the functions of WORKER are now performed in memory by the XSProc module. Nevertheless, WORKER may still be useful for manipulation of external library files and for standalone execution of SCALE modules such as CENTRM and XSDRNPM.

11.2.2 AMPX LIBRARY FORMAT CONVERSION

The primary function of WORKER is to convert data from an initial AMPX MG library file, with any format supported by SCALE, into the AMPX working library format. The nuclear data files distributed with SCALE provide MG data that are truly problem-independent and that can be tailored at runtime for a particular application. These library files must carry data at a sufficient level of detail to allow satisfying many of the less-common but very powerful analyses such as cross section sensitivity studies and coupled neutron-gamma transport calculations. The nuclear data libraries include shielding factors used by the Bondarenko self-shielding method. Temperature dependence of the shielding factors and the thermal scattering kernel data is allowed. Any number of scattering processes [e.g., elastic, discrete-level inelastic, continuum inelastic, (n,2n), etc.] can be included to any degree of anisotropic representation. In short, there is too much detail to require transport codes to process this library.

WORKER processes and combines the data mentioned above into a form ready for use by MG particle transport programs. The final output is an AMPX working library file containing two types of data: group-averaged 1D cross sections for an arbitrary number of processes for neutrons and/or gamma rays, and total 2D transfer matrices (i.e., summed over the scattering types) for neutrons and/or gamma rays. In some cases, the transfer matrices on the master library are temperature-dependent. WORKER performs linear interpolation to the temperature specified for the nuclide, but it does not extrapolate outside the range of the data. WORKER also has the option of producing an AMPX master library of the selected nuclides interpolated to the specified

temperature(s). For resonance nuclides, the elastic scattering matrix is scaled uniformly to make the matrix consistent with the self-shielded 1D values. The $P_\ell(\ell > 0)$ matrices are scaled by the amount required for the P_0 matrix. For processes involving multiple-exit neutrons [e.g., (n,2n), (n,3n), etc.], WORKER multiplies by the appropriate multiplicity before adding to the total transfer matrix. In the case of coupled neutron-gamma libraries, gamma yields are sometimes expressed in "yield" units, thereby requiring a multiplication by a cross section before their introduction into the total transfer matrix. (This scheme allows one to produce self-shielded gamma production cross sections.)

11.2.3 WORKER INPUT SPECIFICATIONS

The following is a description of WORKER input requirements and input-output device requirements.

11.2.3.1 Input parameters

WORKER uses FIDO-type input. The number of items to be input in an array is shown in brackets, and default values are given in parentheses. If N1 and/or N7 are negative, a *direct access* master file will be assumed on that unit. Note that direct access master files may use up to five different units. The initial values of N1 or N7 will be used to determine the starting unit number for the first file, and the unit number will be incremented by 1 for each additional file.

Data block 1

-1$$ Option [1]

1. NFISFOT – No Fission Photon Option (0)

   0 – add fission photons to the transfer array

   1 – do not add fission photons to the transfer array

0$ Logical Assignments [6]. This array is input only if a user needs to modify default values.

1. N1 – Input Master Cross section Library (1)
2. N2 – Input Working Cross section Library (2)
3. N4 – Output Working Cross section Library produced by WORKER (4)
4. N5 – Scratch (18)
5. N6 – Scratch (19)
6. N7 – Output Temperature Interpolated Master Library (0)

1$ Integer Parameters [8]

1. NUCM – Number of nuclides to be read from the input library, N1. (0)
2. NUCW – Number of nuclides to be read from the input working library, N2. (0)
3. IPRT – Output print option trigger. (2)

   -2 – no cross section edits

   -1 – edit reaction cross sections

   >-1 – edit reaction cross sections and transfer arrays through order IPRT
4. IMST – Flag to copy entire Master Library. (0) (0 / >0 = only listed nuclides/all nuclides)
5. IWRK – Flag to copy entire Working Library. (0) (0 / >0 = only listed nuclides/all nuclides)

6. N1A – Sequence number of the filename for unit N1. (1)
7. N2A – Sequence number of the filename for unit N2. (1)
8. N4A – Sequence number of the filename for unit N4. (1)

Data block 2

WORKER can combine data from multiple input libraries to make the merged output working library on logical Unit N4. This output working library contains either the entire library specified or only those nuclides selected on the specified library. If IMST is set greater than ZERO, the entire input library is copied to the output working library. If IWRK is set greater than ZERO, the entire input working library is copied to the output working library. If there is no data in the `4$$` or `5$$` arrays, the nuclide ID numbers listed in the `2$$` and `3$$` arrays remain unchanged. For selected master libraries, nuclides are by default selected at 300 K unless data is provided in the 6** array.

2$$

ID numbers from the input library on N1 of nuclides to be placed on the output working library. If this array is not present and N1 is specified or if IMST > 0, then all nuclides on N1 will be copied to the output working library.

3$$

ID numbers from the input library on N2 of nuclides to be placed on the output working Library. If this array is not present and N2 is specified or if IWRK > 0, then all nuclides on N2 will be copied to the output working library.

4$$

New ID numbers for nuclides from the input master library on N1 to be placed on the output working Library. If this array is not present and N1 is specified, then all nuclides on N1 that are to be copied to the output working library will retain their original ID numbers. These correspond on a one-to-one basis with the ID numbers entered in the 2$$ array.

5$$

New ID numbers for nuclides from the input library on N2 to be placed on the output working library. If this array is not present and N2 is specified, then all nuclides on N2 that are to be copied to the output working library will retain their original ID numbers. These correspond on a one-to-one basis with the ID numbers entered in the $$ array.

6**

Thermal-Scattering kernel temperatures (K) for nuclides selected. Scattering kernels are sometimes provided at several temperatures for a nuclide on a master library. To get data at the specified input temperature, WORKER will interpolate between temperature data. For temperatures above the maximum or below the minimum temperature, WORKER will not extrapolate but instead will use the maximum or minimum temperature data. This array has no effect for sets of data with zero or one thermal kernel.

7$$

MT number of the incoherent thermal-scattering kernel. These allow selecting a thermal-scattering kernel with an MT (identifying) number other than the default (1007).

8$$

Mixture numbers associated with the ID numbers in the 2$$ array.

9$$

Mixture numbers associated with the ID numbers in the 3$$ array.

T - Terminate Block 2.

11.2.3.2 Abbreviated input description

Users who become familiar with the values required by WORKER will become dissatisfied with having to use a detailed input description. The description that follows is intended to serve as a "skeleton" guide for these users:

Data Block 1

-1$$ Option (1)

1. NFISFOT – Fission photon flag

0$$ Logical Assignments (6)

1. NT1 – Input library (1)
2. NT2 – Working In (2)
3. NT4 – Working Out (4)
4. NT5 – Scratch (18)
5. NT6 – Scratch (19)
6. NT7 – Master Out (0)

1$ Integer Parameters [5]

1. NMT – number from master
2. NWT – number from working
3. IPRT – cross section print option
4. IMST – flag to copy entire master library
5. IWRK – flag to copy entire working library
6. N1A – flag to append integer to master library file
7. N2A – flag to append integer to working library file
8. N4A – flag to append integer to output library file

T Terminate Block 1.

Data Block 2

2$$ Identifiers of nuclides on input library (NUCM)

3$$ Identifiers of nuclides on working library (NUCW)

4$$ New identifiers for nuclides from input Library (NUCM)

5$$ New Identifiers for nuclides from working Library (NUCW)

6** Thermal Kernel Temperatures (NUCM)

7$$ MTs for incoherent thermal scattering matrices

8$$ Mixture numbers associated with identifiers in the `2$$` (NUCM)

9$$ Mixture numbers associated with identifiers in the `3$$` (NUCW)

T Terminate Block 2

11.2.3.3 Input/output assignments

WORKER typically requires the following input-output devices during an execution.

Logical Number	Purpose
NT1 (1)	Input Cross section Library
NT2 (2)	Previously Prepared Working/Weighted Library
NT4 (4)	New Working Library
NT5 (18)	Scratch Unit
NT6 (19)	Scratch Unit
NT7 (0)	Temperature Interpolated Master Library
5	Record Input (when run outside of SCALE)
6	Printed Output

11.2.4 SAMPLE PROBLEM

A sample problem includes two calls to WORKER to represent different capabilities. The input assumes there is a master library available on Unit 84 that contains at least the following five nuclides-1001, 8016, 13027, 92235, and 92238-and a working library that contains the same nuclides on Unit 70.

11.2.4.1 Sample problem input

Example 11.2.1 shows the input for the sample problem. The first call to WORKER copies five nuclides from the master library on Unit 84 to the working library on Unit 75. The `0$$` array specifies reading a master library on Unit 84 and writing a working library on Unit 75. The `1$$` array specifies selecting five nuclides that are read from the master library. The `2$$` array lists the five nuclides requested from the master library: hydrogen (1001), oxygen (8016), aluminum (13027), ^{235}U (92235), and ^{238}U (92238). The `4$$` array specifies the new nuclide ID numbers. In this problem, the `4$$` array is not needed since the ID numbers do not change. The 6** array specifies selecting the thermal scattering kernel at 300 K for any nuclide having multiple-scattering kernels.

The second call to WORKER combines a master and working library. The `0$$` array specifies reading a master library on Unit 70 and a working library on Unit 75 and writing a new working library on Unit 79. The `1$$` array specifies selecting five nuclides that are read from the master library and five nuclides that are read from the working library. The `2$$` array lists the five nuclides requested from the master library: hydrogen (1001), oxygen (8016), aluminum (13027), ^{235}U (92235), and ^{238}U (92238). The `4$$` array specifies the new nuclide ID numbers for the master library nuclides. The `3$$` array lists the five nuclides requested from the working library: hydrogen (1001), oxygen (8016), aluminum (13027), ^{235}U (92235), and ^{238}U (92238). The `5$$` array specifies the new nuclide ID numbers for the working library nuclides. The 6** array specifies selecting the thermal scattering kernel at 600 K for any nuclide from the MASTER library having multiple-scattering kernels.

Example 11.2.1: Sample problem input.

```
=WORKER
  0$$ 84  0  75  E
  1$$  5  0  -2  -1  -1   E   T
  2$$ 1001 8016 13027 92238 92235  E
  4$$ 1001 8016 13027 92238 92235  E
  6** 300  300  300   300   300    E   T
END
```

(continues on next page)

(continued from previous page)

```
=WORKER
  0$$ 70  75  79  E
  1$$  5  5  -2  -1  -1   E   T
  2$$ 1001 8016 13027 92238 92235  E
  3$$ 1001 8016 13027 92238 92235  E
  4$$ 3001001 3008016 2013027 1092238 1092235  E
  5$$ 6001001 6008016 5013027 4092238 4092235  E
  6** 600  600  600   600   600    E   T
END
```

11.2.4.2 Reference

11.3 COMPOZ DATA GUIDE

J. R. Knight[1] *and L. M. Petrie*

ABSTRACT

The COMPOZ program used to create the Standard Composition Library is described. Of particular importance is documentation of the COMPOZ input data file structure. Knowledge of the file structure allows users to edit the data file and subsequently create their own site-specific composition library.

ACKNOWLEDGMENT

This work was originally funded by the Office of Nuclear Material Safety and Safeguards of the U.S. Nuclear Regulatory Commission.

11.3.1 INTRODUCTION

COMPOZ is the program that creates (writes) the SCALE Standard Composition Library. Data are input in free form. A text data file containing the input to COMPOZ (and the Standard Composition Library) is available with the SCALE system. Execution of COMPOZ using this data file creates the Standard Composition Library currently available with the SCALE package. This section provides documentation of the data file structure. Knowledge of the data file structure allows users to edit the data file and subsequently create their own site-specific or user-specific composition library.

COMPOZ is intended to create or make *permanent changes* to and/or to print the composition library and should not be used for any other purpose. To avoid confusion with the Standard Composition Library provided with SCALE, it is strongly recommended that only *new* keywords and compositions be used in any site-specific or user-specific library.

11.3.2 INPUT DATA DESCRIPTION

COMPOZ input data are entered in free form. All data must be followed by at least one blank. The COMPOZ input data file contains *five* data records or blocks:

1. COMPOZ mode flag selects whether a new standard composition library will be created, or an old standard composition library will be listed. Only if a new library is being created are the following data records entered. A new library is created with a filename of "xfile089". If an old library is being dumped as an ASCII file, it will be named "_sclN...N" where N...N is an 18 digit sequence number that is incremented starting from 0 for each library dumped in the same directory.
2. The header record contains the library identification, a set of parameters describing the size of the library, and library title with 80 characters per line.

[1] Formerly with Oak Ridge National Laboratory.

3. The standard composition table contains the name, theoretical density, number of elements, and other information about each standard composition. Individual nuclides, mixtures, and compounds are all included in the table.
4. The nuclide information table contains the nuclide identification number, atomic mass, and resonance energy cross sections.
5. The isotopic distribution table contains the nuclide identification number and the atom percent of each isotope used in specifying the default enrichment.

Note: For executing COMPOZ via SCALE, an =COMPOZ is required in the first eight columns of a record preceding the mode flag, and an END is required in the first three columns of a record inserted after the last data record. If debug output is desired, then use =COMPOZ PRINTDEBUG to execute compoz.

11.3.2.1 COMPOZ mode selector

1. LGEN =

   0 – create a new library and list it

   1 – list an existing library

   >1 – list an existing library and write an ASCII input file.

If LGEN is 0, then input the following data to create a new standard composition library.

11.3.2.2 Library heading information

1. IDT – library identification number
2. TITLE – 1 line of 80 characters used to identify the library

11.3.2.3 Standard composition table

1. SCID – Composition name, maximum of 12 characters.
2. ROTH – Theoretical density, gm/cm^3.
3. ICP –

   0 for a mixture,

   1 for a compound.
4. NCZA – Element or nuclide ID
5. ATPM – Weight percent if ICP = 0. Number of atoms per molecule if ICP = 1.
6. END – Keyword END to terminate this standard composition.

For each composition, items 4 and 5 are repeated until all components of the composition are described. Items 1 through 6 are entered in a similar fashion for all compositions. After all the standard compositions are read, terminate the table with an END [label], where [label] represents an optional label.

11.3.2.4 Nuclide information table

1. NZA – Nuclide ID. This should be the mass number + 1000 * the atomic number.
2. AM – Atomic mass, C-12 scale.
3. SIGS – Resonance energy scattering cross section, barns.
4. SIGT – Resonance energy total cross section, barns.
5. NU*SIGF – Resonance energy nu*sigf cross section, barns.

The resonance energy cross sections are averaged over the appropriate energy range for the nuclide. Items 1–5 are repeated for all nuclides. After all nuclides are entered, terminate the nuclide table with an END [label].

11.3.2.5 Isotopic distribution table

1. NZN – 1000 * atomic number of variable isotope elements.
2. ISZA – Isotope ID.
3. ABWP – Default abundance, atom percent.
4. END – Keyword END to terminate this isotopic specification.

The default abundance is generally the naturally occurring abundance. For each element, items 2 and 3 are repeated until 100% total abundance is described, making a set for this element. The next element is described in the same fashion in the next set, etc. After all isotopic distributions are entered, terminate the isotopic distribution table with an END [label].

11.3.3 SAMPLE PROBLEM

The following sample problem first lists the SCALE standard composition library, then creates a new, short standard composition library, then lists and outputs an ASCII copy of this new library, and finally copies this new copy back to the output directory.

```
  =compoz
'  print the current standard composition library
    1
end
=compoz
'  create a new standard composition library
    0
'  library identification number
  101
'  library title
scale-X standard composition library
'  standard composition table
'  all nuclide IDs here must be in the nuclide table
    h               1.0000    0        1001 100.0000 end
    o               1.0000    0        8016 100.0000 end
    u              19.0500    0       92000 100.0000 end
    h2o             0.9982    1        1001    2
                                       8016    1 end
    uo2            10.9600    1       92000    1
                                       8016    2 end
'  end of standard composition table
   end stdcmp
'  nuclide table
'    ID       AWR     SigmaS      SigmaT      nuSigmaF
   1001   1.00783    20.38087    20.38782     0.00000
   1002   2.01410     3.39486     3.39487     0.00000
```

(continues on next page)

(continued from previous page)

```
   8016  15.99491     3.88696     3.88696     0.00000
   8017  16.99913     3.74000     3.74501     0.00000
   8018  17.99916     3.79000     3.79000     0.00000
  92233 233.03964    12.46693    37.62292   100.78482
  92234 234.04095    12.18716    16.09542     2.66969
  92235 235.04393    11.90249    35.22383    90.23152
  92236 236.04556    12.27302    14.93351     1.33334
  92237 237.04874    14.24581    24.68619     1.93695
  92238 238.05080    12.32636    14.62708     0.65970
'  end of nuclide table
    end nuclides
'  isotope distribution table
'  all nuclide IDs here must be in the nuclide table
   1000   1001  99.9885
          1002   0.0115 end
   8000   8016  99.7570
          8017   0.0380
          8018   0.2050 end
  92000  92234   0.0054
         92235   0.7204
         92238  99.2742 end
'  end of isotope distribution table
    end isotopes
'  end of compoz input
end
=compoz
'  print and create an ASCII copy of the current standard composition library
'  (created in the previous step)
  2
end
=shell
 # copy the ASCII copy back to the output directory
  copy_file _scl000000000000000000 ${OUTBASE}.stdcmplib
end
```

11.4 ICE: MODULE TO MIX MULTIGROUP CROSS SECTIONS

N. M. Greene,[*] *L. M. Petrie, S. K. Fraley*[*] [ice1]

ABSTRACT

ICE is a legacy SCALE utility program that reads microscopic cross sections from an AMPX working library and uses input mixture number densities to produce macroscopic cross sections, which are written to an output file in the AMPX working library format. User input is entered with the FIDO procedures.

11.4.1 INTRODUCTION

ICE (**I**ntermixed **C**ross Sections **E**ffortlessly) is a legacy SCALE utility program that reads microscopic cross sections from an AMPX working library and uses input mixture number densities to produce macroscopic cross sections, which are written to an AMPX working library output file. The code was originally developed to allow efficient cross section mixing with minimum user effort and with reduced core storage requirements. The SCALE version of ICE is the latest in a series [UTILS-Fra76] of versions of the program.

In previous versions of SCALE several sequences employed the ICE module as a component in the self-shielding procedure; however in modern sequences the functionality of ICE has been replaced by new routines in the XSProc module. The ICE module is retained in the modern version of SCALE mainly for use as a standalone executable module to compute macroscopic cross sections and to provide backward compatibility with legacy inputs.

11.4.2 CROSS SECTION MIXING EXPRESSIONS

The mixing operations in ICE use the simple expressions presented below.

11.4.2.1 Cross-section mixing for AMPX libraries

For the options that produce AMPX working libraries, the mixing of cross sections involves a very simple summing of constituent values times a number density for the constituent, that is, Σ, a macroscopic value, is determined by

$$\Sigma = \sum_j N_j \sigma_j \tag{11.4.1}$$

where the j are the individual nuclides in the mixture whose number density and microscopic cross sections are N_j and σ_j, respectively.

The only exceptions to the above rule are for fissionable mixtures where the number of neutrons per fission, ν_g, or a fission spectrum, χ_g, is required:

$$\nu_g = \frac{\sum_j N_j \nu_{gj} \sigma_{gj}}{\sum_j N_j \sigma_{fgj}} \tag{11.4.2}$$

χ_g is defined as the fraction of the fission neutrons produced by the mixture which fall in group g. By definition,

$$\sum_g \chi_g = 1.0 \tag{11.4.3}$$

ICE uses the following scheme to determine χ. First, terms F_g are determined by

$$F_g = \sum_j N_j \chi_{g,j} \sum_{g'} \overline{\nu\sigma}_{fg',j} \hat{\phi}_{g'}, \tag{11.4.4}$$

where VO fg, j is the average of the product of ν times $\sigma 0_f$ for the nuclide, $\chi_{g,j}$ is the nuclide fission spectrum, and $\hat{\phi}_g$ is an estimate for the integrated flux in group g'. Once the F_g are determined, χ_g is determined by normalizing the sum of F_g to unity.

In many AMPX libraries, the integrals of the spectrum used to determine the multigroup values are carried on the library for each nuclide. ICE uses this nuclide-dependent spectrum to determine χ_g. This option should be exercised with caution, however, for no attempt is made to ensure that the individual spectra are consistently normalized.

11.4.3 INPUT INSTRUCTIONS

The input to ICE uses the FIDO schemes described in the FIDO chapter. In the descriptions, the number of entries expected in an array is given in square brackets.

****

Card A (20A4)

Title card

Data Block 1

-1$ Direct-Access Specifications [4]

1. NB8 No longer used.
2. NL8 No longer used.
3. NB9 No longer used.
4. NL9 No longer used.

0$ Logical Unit Specifications for Various Cross-Section Libraries [5]

1. INTAPE Input AMPX working library unit; default 4.
2. IOT1 Output AMPX working library unit; default 3.
3. IOT2 No longer used.
4. IOT3 No longer used.
5. IOT4 No longer used.

1$ Problem Size and Major Options [7]

1. MIX Number of cross-section mixtures to be made.
2. NMIX Number of mixing operations (elements times density operations) to be performed.
3. IFLAG(1) Set greater than ten if AMPX working library output desired
4. IFLAG(2) No longer used..
5. IFLAG(3) No longer used..
6. IFLAG(4) No longer used.
7. KOPT No longer used.

T - Terminate Block 1

Data Block 2

2$ [NMIX]

1. (KM(I),I=1,NMIX) Mixture numbers in the mixture specification table – values range from 1 to MIX.

3$ [NMIX]

1. (KE(I),I=1,NMIX) Element identifiers for the mixture specification table.

4* [NMIX]

1. (RHO(I),I=1,NMIX) Atom densities for the mixture specification table.

5$ [MIX]

1. (NCOEF(I),I=1,MIX) Number of Legendre coefficients, including P_o, to be mixed for each mixture.

6* [NG+4]

No longer used.

12$ [NMIX]

`1.(NUCMX(I),I=1,NMIX) Element mixture identifiers for the mixture specification table.

7$ No longer used.

T - Terminate Data Block 2

Data Block 3

8$ [MIX] Required only if IFLAG(1) > 0

1. (MID(I),I=1,MIX) Mixture ID numbers for AMPX working library; default (MID(I)=I,I=1,MIX)

9$ [N] No longer used.

.

.

10 No longer used.

11 No longer used.

T - Terminate Data Block 3

11.4.4 SAMPLE PROBLEM

A simple case has been selected to demonstrate the use of ICE. In this case, it is desired to produce mixture cross sections for UO_2 and H_2O using basic data from ENDF version 7 238 group SCALE library. Information pertinent to the basic data is given in the following table:

Nuclide	Identifier	Order of Scattering
^{235}U	92235	5
^{238}U	92238	5
O	8016	5
H	1001	5

The atom densities to be used are:

UO_2

N(^{235}U) = 0.01 atoms/(barn-cm)

N(^{238}U) = 0.04 atoms/(barn-cm)

N(O) = 0.08 atoms/(barn-cm)

Water

N(H) = 0.06 atoms/(barn-cm)

N(O) = 0.03 atoms/(barn-cm)

In the sample case, we have elected to make an AMPX working library on logical 61,

We have selected further to identify UO_2 with a 111 on the AMPX working library.

CSAS-MG PARM=CHECK is run to set up the master library, then WORKER is run to produce a working library for ICE.

A listing of the input follows:

```
=csas-mg   parm=(check)
cross sections for ice sample problem
v7-238
read composition
  atom   1 1 4  1001 1  8016 1  92235 1  92238 1  end atom
end composition
end
=ice
sample ice problem
0$$  4 61 62 63 64
1$$ 2 5 13 13 13 13 2  1t
2$$ 3r1 2r2
3$$ 92235 92238 8016 1001 8016
4** 0.01  0.04  0.08 0.06 0.03
5$$ 1 2
12$$  f1
 2t
8$$ 111 222  9$$ 1 2 3  11$$ 100 1111 2222  3t
End
```

11.4.4.1 Reference

11.5 FIDO INPUT SYSTEM

L. M. Petrie

ABSTRACT

This document provides a description of the FIDO input system being used in conjunction with several SCALE functional modules. The FIDO system is a widely used method of entering or modifying large data arrays with minimum effort. Special advantage is taken of patterns of repetition or symmetry whenever possible.

ACKNOWLEDGMENTS

This document was funded by the Office of Nuclear Material Safety and Safeguards, U.S. Nuclear Regulatory Commission, for inclusion within the SCALE manual.

11.5.1 INTRODUCTION

The FIDO input method is specially devised to allow entering or modifying large data arrays with minimum effort. Advantage is taken of patterns of repetition or symmetry wherever possible. The FIDO system was patterned after the input method used with the FLOCO coding system at Los Alamos and was first applied to the DTF-II code. Since that time, numerous features requested by users have been added, a free-field option has been developed, and the application of FIDO has spread to innumerable codes. Starting with SCALE 5, the FIDO routines have been converted to Fortran 90, and the requirement that arrays be held in a large container array has been removed.

The data are entered in units called "arrays." An array comprises a group of contiguous storage locations that are to be filled with data at the same time. These arrays usually correspond on a one-to-one basis with Fortran arrays used in the program. A group of one or more arrays read with a single call to the FIDO package forms a "block," and a special delimiter is required to signify the end of each block. Arrays within a block may be read in any order with respect to each other, but an array belonging to one block must not be shifted to another. The same array can be entered repeatedly within the same block. For example, an array could be filled with "0" using a special option, and then a few scattered locations could be changed by reading in a new set of data for that array. If no entries to the arrays in a block are required, the delimiter alone satisfies the input requirement.

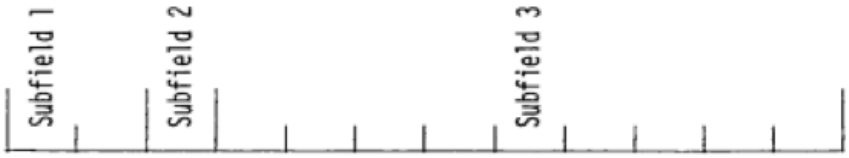
Three major types of input are available: fixed-field input, free-field input, and user-field input.

11.5.2 FIXED-FIELD INPUT

The fixed-field input option is documented here for completeness.

Note: The use of fixed-field input is NOT recommended. Use the free-field input option documented in Sect. 11.5.3.

Each record is divided into six 12-column data fields, each of which is divided into three subfields. The following sketch illustrates a typical data field. The three subfields always comprise 2, 1, and 9 columns, respectively.



To begin the first array of a block, an array originator field is placed in any field on a record:

Subfield 1: An integer array identifier < 100 specifying the data array to be read in.

Subfield 2: An array-type indicator:

"$" if the array is integer data

"*" if the array is real data

"#" if the array is double-precision data

Subfield 3: Blank

Data are then placed in successive fields until the required number of entries has been accounted for.

In entering data, it is convenient to think of an "index" or "pointer" as a designator that is under the control of the user and which specifies the position in the array into which the next data entry is to go. The pointer is always positioned at array location #1 by entering the array originator field. The pointer subsequently moves according to the data operator chosen. Blank fields are a special case in that they do not cause any data modification and do not move the pointer.

A data field has the following form:

Subfield 1: The data numerator, an integer <100. We refer to this entry as N_1 in the following discussion.

Subfield 2: One of the special data operators listed below.

Subfield 3: A nine-character data entry, to be read in F9.0 format. It will be converted to an integer if the array is a "$" array or if a special array operator such as Q is being used. Note that an exponent is permissible but not required. Likewise, a decimal is permissible but not required. If no decimal is supplied, it is assumed to be immediately to the left of the exponent, if any; and otherwise to the right of the last column. This entry is referred to as N3 in the following discussion.

A list of data operators and their effect on the array being input follows:

"Blank"

indicates a single entry of data. The data entry in the third subfield is entered in the location indicated by the pointer, and the pointer is advanced by one. However, an entirely blank field is ignored.

"+" or "-"

indicates exponentiation. The data entry in the third field is entered and multiplied by $10^{\pm N_1}$ where N_1 is the data numerator in the first subfield, given the sign indicated by the data operator itself. The pointer advances by one. In cases where an exponent is needed, this option allows the entering of more significant figures than the blank option.

"&"

has the same effect as "+."

"R"

indicates that the data entry is to be repeated N_1 times. The pointer advances by N_1. The entry 5R1 is equivalent to 1 1 1 1 1.

"I"

indicates linear interpolation. The data numerator, N_1, indicates the number of interpolated points to be supplied. The data entry in the third subfield N_3 is entered, followed by Nj interpolated entries equally spaced between that value and the data entry found in the third subfield of the next nonblank field. The pointer is advanced by $N_1 + 1$. The field following an "I" field is than processed normally, according to its own data operator. The "I" entry is especially valuable for specifying a spatial mesh. For example, the entry 3I 10 50 is equivalent to 10 20 30 40 50. In "$" arrays, interpolated values will be rounded to the nearest integer.

"L"

indicates logarithmic interpolation. The effect is the same as that of "I" except that the resulting data are evenly separated in log-space. This feature is especially convenient for specifying an energy mesh. For example, the entry 3L 1 1+4 is equivalent to 1 10 100 1000 10000.

"Q"

is used to repeat sequences of numbers. The length of the sequence is given by the third subfield, N_3. The sequence of N_3 entries is to be repeated N_1 times. The pointer advances by $N_1 * N_3$. If either N_1 or N_3 is 0, then a sequence of $N_1 + N_3$ is repeated one time only, and the pointer advances by $N_1 + N_3$. This feature is especially valuable for geometry specification.

The "N" option

has the same effect as "Q," except that the order of the sequence is reversed each time it is entered. This feature is valuable for the type of symmetry possessed by S_n quadrature coefficients.

"M"

has the same effect as "N," except that the sign of each entry in the sequence is reversed each time the sequence is entered. For example, the entries

1 2 3 2M2

would be equivalent to

1 2 3 -3 -2 2 3.

This option is also useful in entering discrete ordinates angular quadrature coefficients.

"Z"

causes $N_1 + N_3$ locations to be set at 0. The pointer is advanced by $N_1 + N_3$.

"C"

causes the position of the last array entered to be printed. This is the position of the pointer, less 1. The pointer is not moved.

"O"

causes the print trigger to be changed. The trigger is originally off. Successive "O" fields turn it on and off alternately. When the trigger is on, each record is listed as it is read.

"S"

indicates that the pointer is to skip N_1 positions leaving those array positions unchanged. If the third subfield is blank, the pointer is advanced by N_1. If the third subfield is nonblank, that data entry is entered following the skip, and the pointer is advanced by $N_1 + 1$.

"A"

moves the pointer to the position, N_3 specified in the third subfield.

"F"

fills the remainder of the array with the datum entered in the third subfield. For example, F9 will fill all positions of the array with a value of 9.

"E"

skips over the remainder of the array. The array length criterion is always satisfied by an E, no matter how many entries have been specified. No more entries to an array may be given following an "E," except that data entry may be restarted with an "A."

The reading of data to an array is terminated when a new array origin field is supplied, or when the block is terminated. If an incorrect number of positions has been filled, an error edit is given; and a flag is set which will later abort execution of the problem. FIDO then continues with the next array if an array origin was read. Otherwise, control is returned to the calling program.

A block termination consists of a field having "T" in the second subfield. Entries following "T" on a record are ignored, and control is returned from FIDO to the calling program.

Comment records can be entered within a block by placing an apostrophe (') in column 1. Then columns 2–80 will be listed, with column 2 being used for printer carriage control. Such records have no effect on the data array or pointer.

11.5.3 FREE-FIELD INPUT

With free-field input, data are written without fixed restrictions as to field and subfield size and positioning on the record. The options used with fixed-field input are available, although some are slightly restricted in form. In general, fewer data records are required for a problem, the interpreting print is easier to read, a record listing is more intelligible, the records are easier to enter, and certain common data entry errors are tolerated without affecting the problem. Data arrays using fixed- and free-field input can be intermingled at will within a given block.

The concept of three subfields per field is still applicable to free-field input; but if no entry for a field is required, no space for it need be left. Only columns 1–72 may be used, as with fixed-field input. A field may not be split across records.

The array originator field can begin in any position. The array identifiers and type indicators are used as in fixed-field input. The type indicator is entered twice to designate free-field input (i.e., "$$," "**," or "##"). The blank third subfield required in fixed-field input is not required. For example,

31**

indicates that array 31, a real-data array, will follow in free-field format.

Data fields may follow the array origin field immediately. The data field entries are identical to the fixed-field entries with the following restrictions:

1. Any number of blanks may separate fields, but at least one blank must follow a third subfield entry if one is used.

2. If both first- and second-subfield entries are used, no blanks may separate them (i.e., 24S, but not 24 S).

3. Numbers written with exponents must not have imbedded blanks (i.e., 1.0E+4, 1.0-E4, 1.0+4, or even 1+4, but *not* 1.0 E4). A zero should never be entered with an exponent. For example, 0.00 - 5 or 0.00E - 5 will be interpreted as $- 5 \times 10^{-2}$.

4. In third-subfield data entries only 9 digits, including the decimal but not including the exponent field, can be used (i.e., 123456.89E07, but *not* 123456.789E07).

5. The Z entry must be of the form: 738Z, *not* Z738 or 738 Z.

6. The + or - data operators are not needed and are not available.

7. The Q, N, and M entries are restricted: 3Q4, 1N4, M4, but *not* 4Q, 4N, or 4M.

11.5.4 USER-FIELD INPUT

If the user follows the array identifier in the array originator field with the character "U" or "V," the input format is to be specified by the user. If "U" is specified, the FORTRAN FORMAT to be used must be supplied in columns 1–72 of the next record. The format must be enclosed by the usual parentheses. Then the data for the entire array must follow on successive records. The rules of ordinary FORTRAN input as to exponents, blanks, etc., apply. If the array data do not fill the last record, the remainder must be left blank.

"V" has the same effect as "U," except that the format read in the last preceding "U" array is used.

11.5.5 CHARACTER INPUT

If the user wishes to enter character data into an array, at least three options are available. The user may specify an arbitrary format using a "U" and reading in the format. The user may follow the array identifier by a "/." The next two entries into subfield 3 specify the beginning and ending indices in the array into which data will be read. The character data are then read starting with the next data record in an 18A4 format if going to a real or integer array, and 9AB if going to a double precision array.

Finally, the user may specify the array as a free-form "*" array and then specify the data entries as "nH" character data where n specifies how many characters follow H.

11.6 MALOCS2: MODULE TO COLLAPSE AMPX MASTER CROSS SECTION LIBRARIES

L.M. Petrie

11.6.1 INTRODUCTION

MALOCS2 (**M**iniature **A**MPX **L**ibrary **O**f **C**ross **S**ections) is a module to collapse AMPX master cross-section libraries. The SCALE MALOCS2 module is an extension of the AMPX module MALOCS. MALOCS2 provides capability to read the collapsing spectrum from the output flux file produced by XSDRNPM, and also has extended options for collapsing Legendre moments of the 2D elastic scattering matrix. The module can be used to collapse neutron, gamma-ray, or coupled neutron-gamma master libraries.

11.6.2 MALOCS INPUT DATA

broadfilename
filename of the collapsed library [no default]

crosssectionprint
cross section printing option [none]

none - don't print any cross sections

onedxsecs - print the 1D cross sections

twodxsecs N - print the 2D cross sections through Legendre order N

epsilon
epsilon for when to print invalid moment messages[0.05]

finefilename
filename of the input library [no default]

fluxfilename
filename of an xsdrn flux file to be used in the collapse [no default]

numgammagroups
the number of fine gamma groups [no default]

gammacollapse
the broad group by fine group collapse structure for the gammas

must come after "numgammagroups"

latticezones
identifies the zones to be used as fuel, gap, clad, and moderator [1,2,3,4]

max2dweightorder
maximum Legendre order to be collapsed [max Legendre order of the nuclide]

numneutrongroups
the number of fine neutron groups [no default]

neutroncollapse
the broad group by fine group collapse structure for the neutrons

must come after "numneutrongroups"

printepsilon
not used [2.0D-6]

problemfilename
filename of the xsdrn data file that corresponds to the flux file [no default]

sigmatotalpl
flag to turn on doing a within group correction using the Pl weighted sigma total

'y' or 'yes' is true, anything else is false [true]

updatechi
flag to turn on updating the total chi

'y' or 'yes' is true, anything else is false [true]

validate2ds
flag to validate the Legendre moments of the collapsed 2D cross sections

'y' or 'yes' is true, anything else is false [true]

weighttype
type of weighting to be done

innercell - cell weight over a subset of the zones

innercell is followed by the largest zone number in the innercell

cell - cell weight over the whole cell

zone - weight each zone independently

region - cell weight each nuclide over only the zones it is in

default is region

wgtsource
source of the weighting flux

nuclideflux - use the flux from the nuclide on the fine group library

inputflux - read a flux from input

[default is to use an xsdrn flux]

end
terminates input stream

11.6.3 MALOCS EXAMPLE PROBLEM

The following problem shortens the 56 group library to just the nuclides that will be used to run a fixed source, 1-D discrete ordinates calculation of a void sphere with a neutron source in it, surrounded by a sphere of water, and then surrounded by an iron sphere. The flux from the discrete ordinates problem is then used to collapse the short library to 14 groups using a zone collapse method. Finally, the collapsed library is listed showing the nuclides on it, and copied back to the input directory.

```
=shell
  ln -s ${DATA}/scale.rev04.xn56v7.1 ft51f001
end
=ajax
  0$$ 52 e
  1$$ 1  1t
```

(continues on next page)

(continued from previous page)

```
  2$$ 51 8  2t
  3$$ 1001 1002 8016 8017 26054 26056 26057 26058 3t
end
=csas1   parm=bonami
generate a flx file to be used to collapse a library
v7-56n
read composition
  iron  1 1.0 293.0  end iron
  water 2 1.0 293.0  end water
end composition
read celldata
  multiregion spherical end
  0 1.0  2 10.0  1 15.0  end zone
  moredata
    ievt=0 iqm=1 ntd=61 fwr=62 source(1)=15
    0.2 0.2 0.2 0.5 0.5 0.5 0.5 0.5 0.5 0.2 0.2 0.2 0.05 0.05 0.05
  end moredata
end celldata
end
=malocs2
' the input fine group cross section library to be collapsed
  finefilename=ft52f001
' the output collapsed cross section library
  broadfilename=ft53f001
' the file with the fluxes from xsdrn to be used to collapse the XSs
  fluxfilename=ft62f001
' the file containing the description of the xsdrn problem
  problemfilename=ft61f001
' number of fine neutron groups
  numneutrongroups=56
' fine group to broad group correspondence array
  neutroncollapse
  4r1 4r2 4r3 4r4 4r5 4r6 4r7 4r8 4r9 4r10 4r11 4r12 4r13 4r14
' type of weighting to be used in doing the collapse
  weighttype=zone
  end
end
=paleale
  0$$ 53 e 1$$ 0 1t
end
=shell
  cp ft53f001 ${OUTDIR}
end
```

12. INSTALLING AND BUILDING SCALE

12.1 SYSTEM REQUIREMENTS

12.1.1 SUPPORTED OPERATING SYSTEMS

- Linux 64-bit (RHEL 6 or newer)
- Mac OS X (Darwin) 10.11.6 or newer
- Windows 7, 64-bit or newer

12.1.2 DISK SPACE REQUIREMENTS

You must have at least 180 GB free on your computer when you begin a SCALE 6.3 install! Once setup files are removed, the final disk usage for SCALE 6.3 will be ~110 GB.

You will need to download ~62 GB of data to begin the SCALE install:

- SCALE data pak installer (61 GB)
- SCALE code installer for your operating system (<1 GB)

Once these files are available locally on your machine, proceed with the installation instructions below.

Note: Once unpacked and installed, the SCALE data and code will be ~110 GB, which will increase the total disk space usage to ~171 GB. After verifying the install was successful, you may delete the data pak and code installers to free your disk of the ~62 GB in setup files, bringing the final install size to ~110GB.

12.1.3 MEMORY

Minimum: 8 GB (per CPU for parallel calculations).

Note: SCALE calculations vary widely with the amount of memory required. It is always possible to increase the complexity of a model and exceed the amount of memory available. Large TRITON-KENO 3D models can easily require 32 GB of memory.

12.1.4 JAVA REQUIREMENTS

Java 1.8 or newer.

12.2 INSTALLATION INSTRUCTIONS

The SCALE 6.3.0 installation has been divided into three components: 64-bit pre-compiled binaries, source code, and the data. The precompiled binaries are available in platform-familiar installers: NSIS on Windows, Drag-N-Drop bundle on Mac, and Tar GZip on Linux. For deployments that include source code, it is available as a zip file, and the data are available as a self-extracting IzPack installer. Other systems may require a custom build of SCALE from source code using the build instructions provided below.

Pre-Compiled Binaries

Note: After the Binaries are installed, the nuclear data still must be installed. After completing the binary installation according to the instructions in this section, please follow the instructions for SCALE 6.3 Data.

Windows

Double-click the Windows installation file "SCALE-6.3.0-setup.exe", agree to the terms, and the following "Welcome" dialog will be shown.

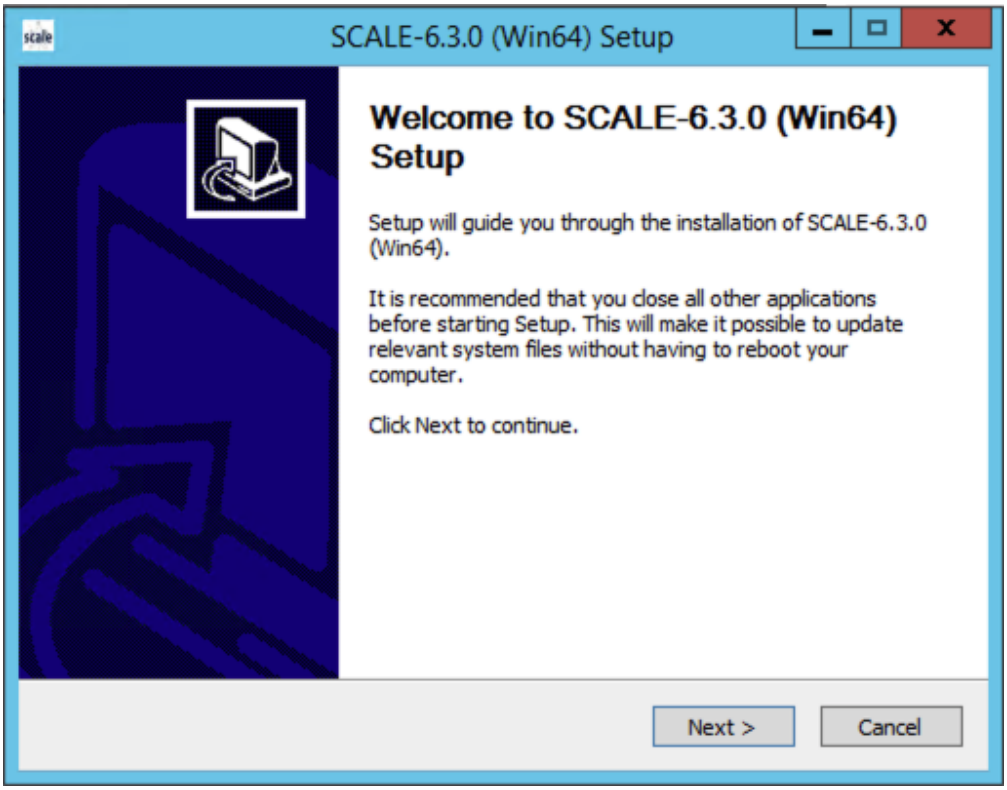


Fig. 12.2.1: Example Windows installation start screen.

Follow the prompts through the install. There are two user decisions. The first is regarding the install location, as shown below. Generally, the "C:SCALE-6.3.0" is sufficient. (Do not install SCALE to the "Program Files" directory, as it has insufficient default privileges.)

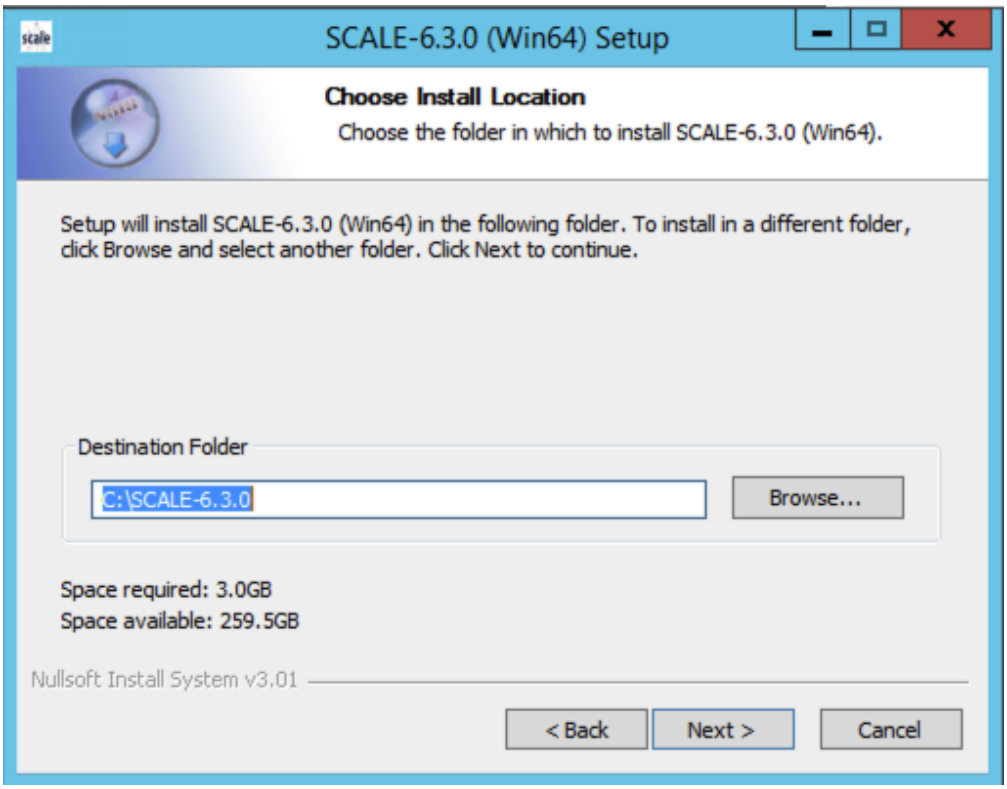


Fig. 12.2.2: Example Windows install location.

The next user decision is to designate the name of the SCALE 6.3.0 application folder within the Windows start menu. The default is “SCALE 6.3.0 (Win64)”; however, changing this name will not affect the installation in any way.

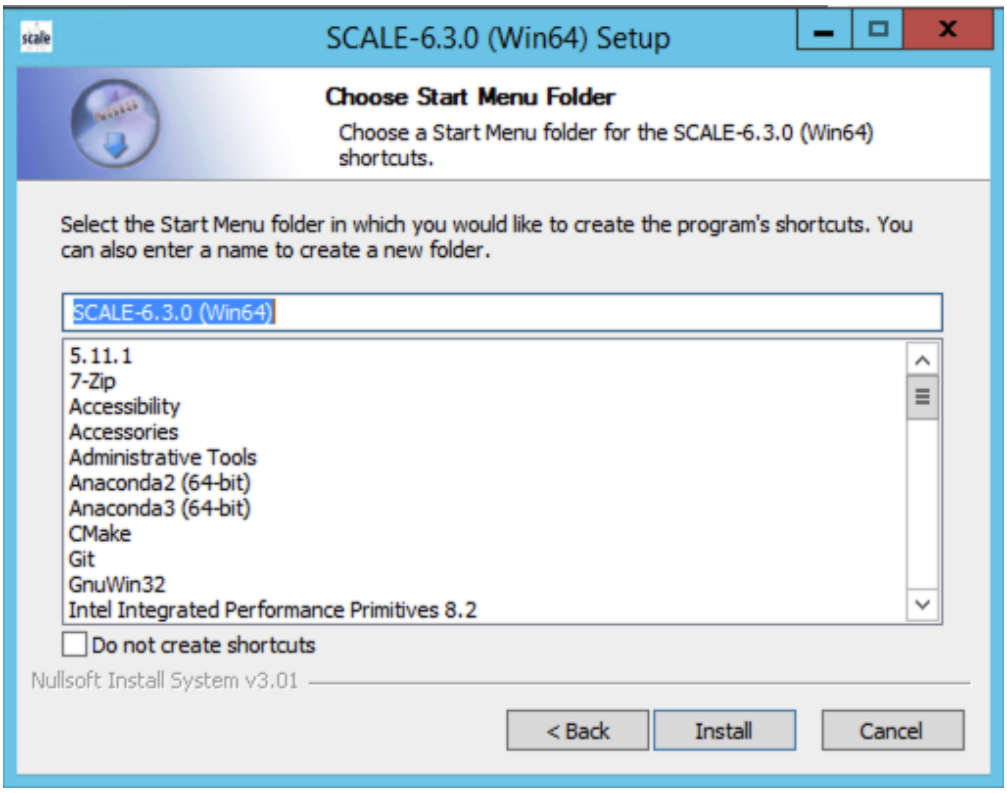


Fig. 12.2.3: Example Windows start menu folder selection.

The installation can take several minutes. Note that Norton Antivirus on Windows requires an exception for the SCALE Runtime Environment program, scalerte.exe. If using Norton Antivirus, please establish this exception prior to testing; otherwise, SCALE 6.3.0 will not be able to execute. Once installation is complete, follow the instructions given below to install SCALE 6.3 Data.

Mac OSX

Double click the Mac OSX installation file “SCALE-6.3.0-setup.dmg”; after a few moments, the following screen will open.

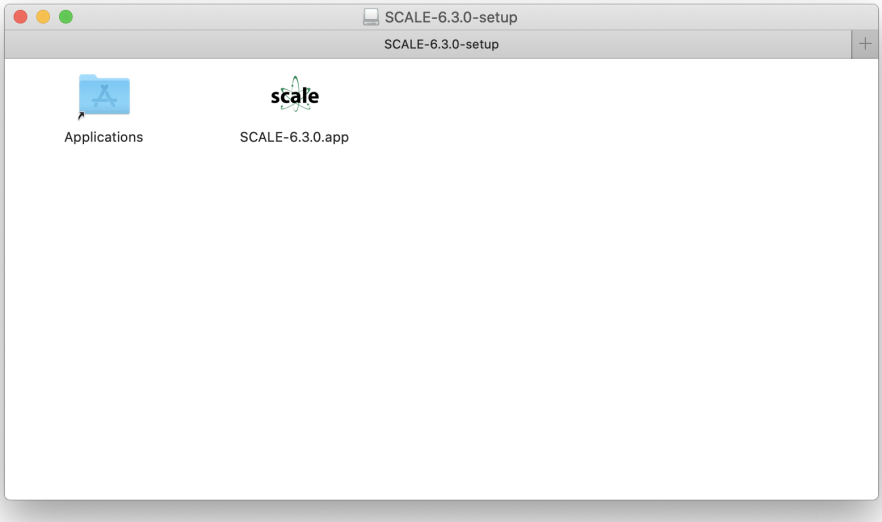


Fig. 12.2.4: Example Mac OS X bundle contents.

Drag the "SCALE-6.3.0.app" icon onto "Applications" to install SCALE-6.3.0. Once installation is complete, follow the instructions given below to install SCALE 6.3 Data.

Linux

Copy the SCALE-6.3.0.tar.gz to the installation directory. Invoke 'tar -xzf SCALE-6.3.0.tar.gz' to create the SCALE-6.3.0 directory.

```
[hts@hts-rhel-pc] ~/codes $ ls *.tar.gz
SCALE-6.3.0.tar.gz
[hts@hts-rhel-pc] ~/codes $ tar xf SCALE-6.3.0.tar.gz
[hts@hts-rhel-pc] ~/codes $ cd SCALE-6.3.0/
[hts@hts-rhel-pc] ~/codes/SCALE-6.3.0 $ ls
Ampx             bin                 cmds  etc                 help    include   lib      Meshview        plugins  regression    samples               uslstats
AmpxRunResources.xml  CentrmOptions.xml  docs  FileNameAliases.txt  images  Javapeno  LICENSE  MoreOptions.xml  python   release-info  ScaleRunResources.xml  vibe
```

Fig. 12.2.5: Example Linux Tar GZip.

Once installation is complete, follow the instructions given below to install SCALE 6.3 Data.

Source Code

If you received the source code version, you can unzip SCALE-6.3.0-Source.zip file to any location on your computer.

For example, on a Windows platform, double click the SCALE-6.3.0-Source.zip file. This will extract the files into the "SCALE-6.3.4-Source" directory.

On Linux and Mac OS X, create a destination directory for the source and copy the SCALE-6.3.0-Source.zip into that directory.

Change to the directory and invoke unzip `SCALE-6.3.0-Source.zip` to deploy the source code.

12.2.1 SCALE 6.3 DATA

Note that if you have SCALE 6.3 data installed from a prior SCALE 6.3 installation, you can copy or link the data into the SCALE 6.3.0 directory. To link preinstalled SCALE 6.3 data, open a command prompt and change the directory to the SCALE 6.3.0 installation directory:

```
(C:\SCALE-6.3.0, /Applications/SCALE-6.3.0/Contents/Resources, etc.).
```

On Windows, the following will link the SCALE 6.3 data into the SCALE 6.3.0 installation. (This may require administrative privileges.)

```
mklink /D data C:\SCALE-6.3\data
```

On OS X, the following will link the SCALE 6.3 data into the SCALE 6.3.0 installation:

```
ln -s /Applications/SCALE-6.3.app/Contents/Resources/data data
```

On Linux, the following will link the SCALE 6.3 data into the SCALE 6.3.0 installation:

```
ln -s /scale/scale6.3/data data
```

To begin installation of SCALE 6.3 data**, copy the scale-6.3-data-setup.jar to your local disk. Double-click this jar file. If the installer does not start**, then bring up a command prompt or terminal window and issue the following command: `java -jar scale-6.3-data-setup.jar` in the location where the installer jar file was copied.

After launching the installer, a dialog should appear as shown below.

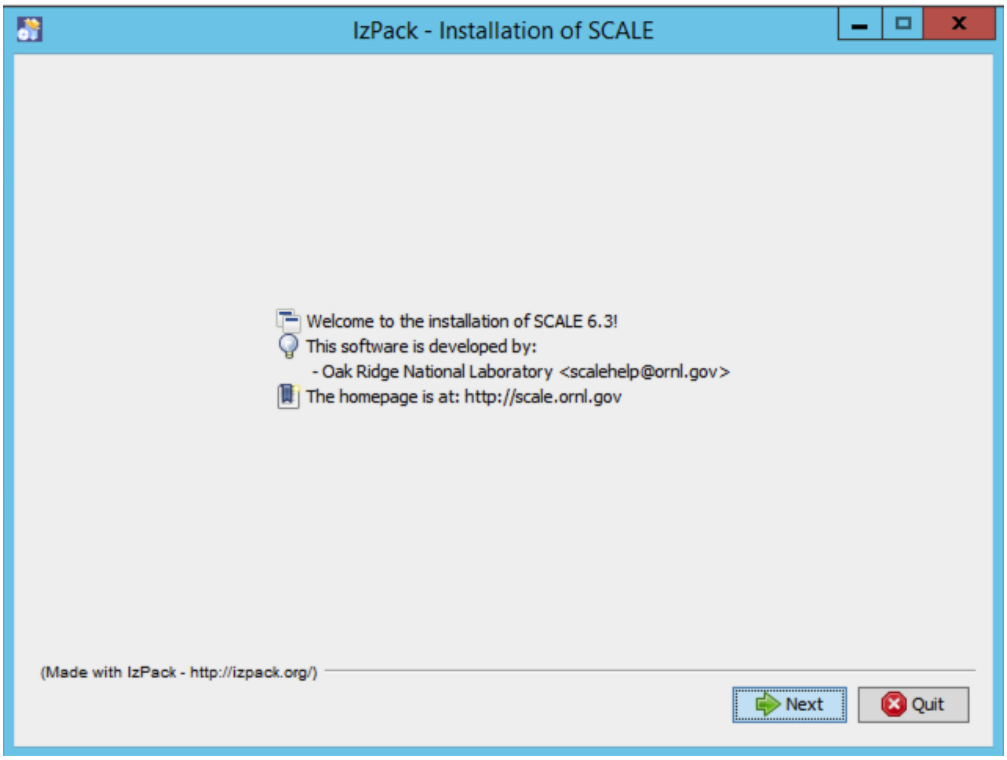


Fig. 12.2.6: SCALE 6.3 data installer welcome dialog.

Continue by pressing *Next*.

You will be prompted to review and accept the terms of the license agreement.

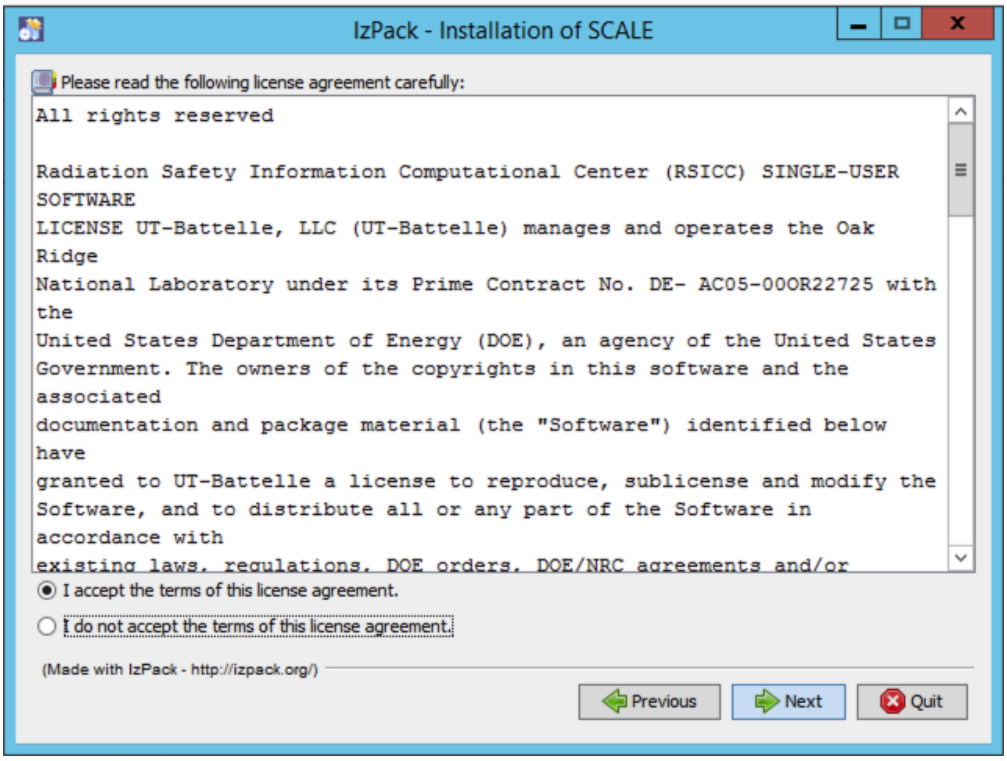


Fig. 12.2.7: SCALE license agreement.

To proceed, check to indicate your acceptance of the terms, and press next. You will be prompted to choose the destination of your installation.

For Windows users, the recommended installation path is inside your SCALE-6.3.0 directory:

```
c:\SCALE-6.3.0
```

For Linux, a typical location is:

```
/scale/scale6.3.0
```

For Mac, a typical location is:

```
/Applications/SCALE-6.3.0.app/Contents/Resources
```

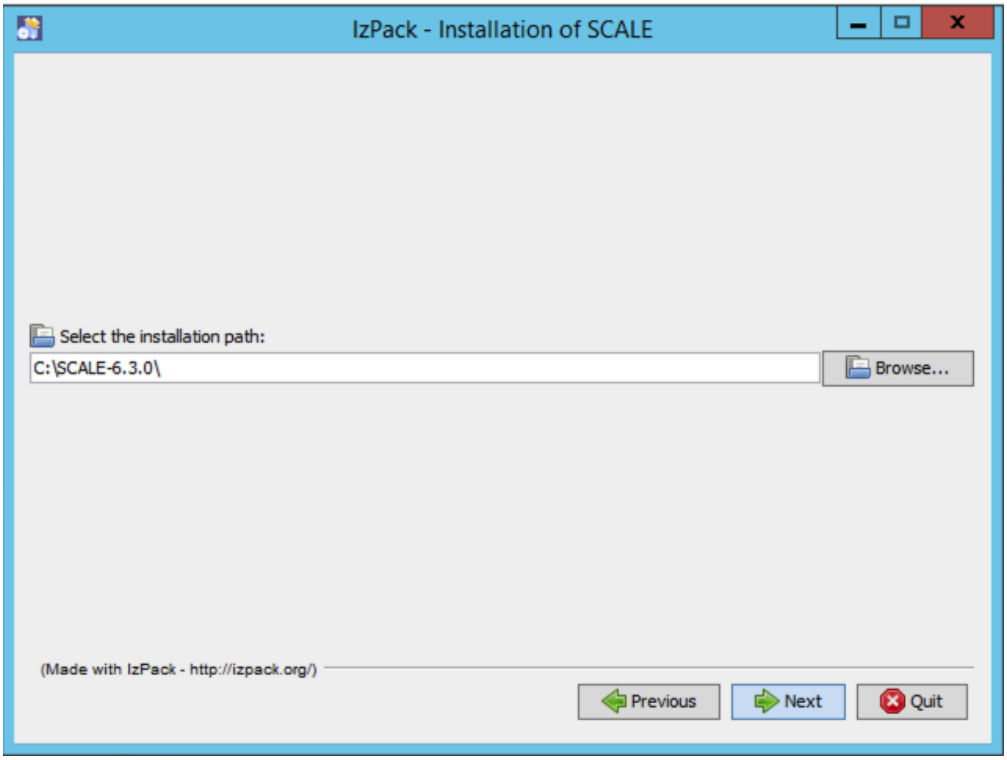


Fig. 12.2.8: Example setting typical data installation path on Windows.

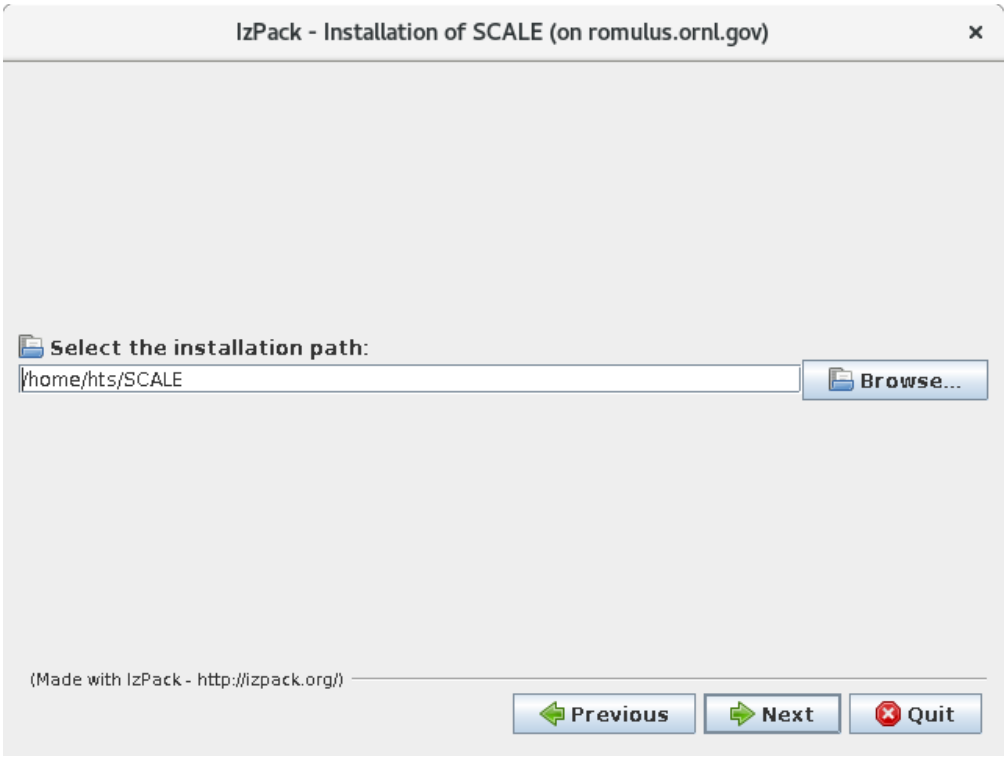


Fig. 12.2.9: Example setting a typical data installation path on Linux.

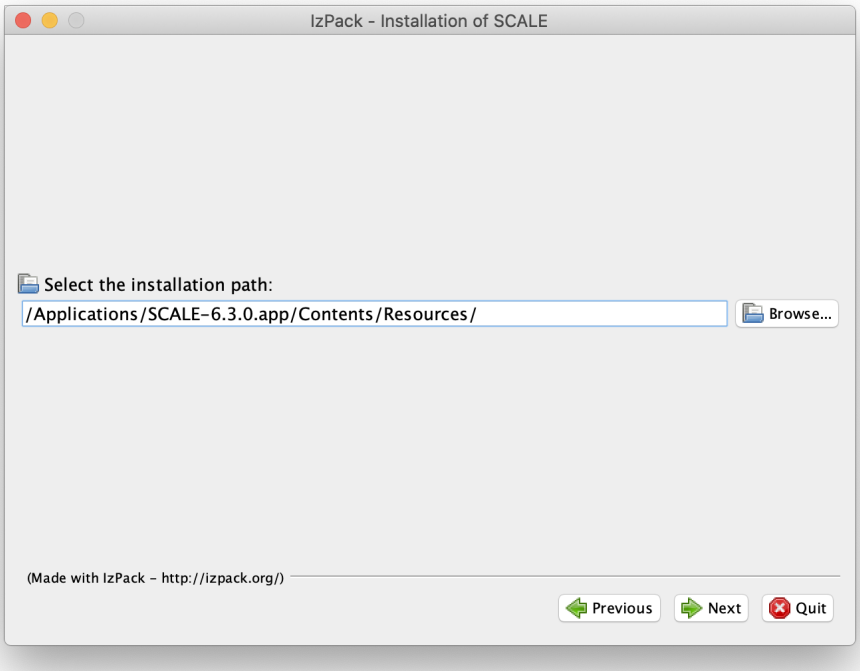


Fig. 12.2.10: Example setting a typical data installation path on Mac OS X.

For some systems, the installer will not be allowed to create a new directory. If you encounter a message like the one shown below, simply create the directory manually (e.g., using Windows Explorer), and then return to the installer and continue.

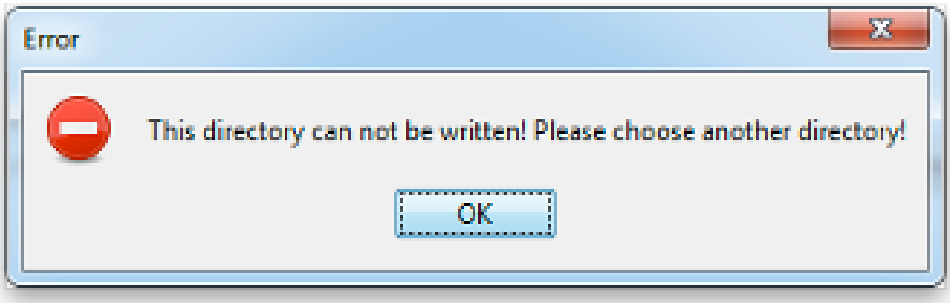


Fig. 12.2.11: Data installer directory creation error message.

In most cases, the installer will present a prompt to confirm the creation of a new directory; if that target directory is correct, then you can simply press *OK*. If the directory was manually created, the installer will notify you that it is about to overwrite any previous contents. Press *Yes* to continue.

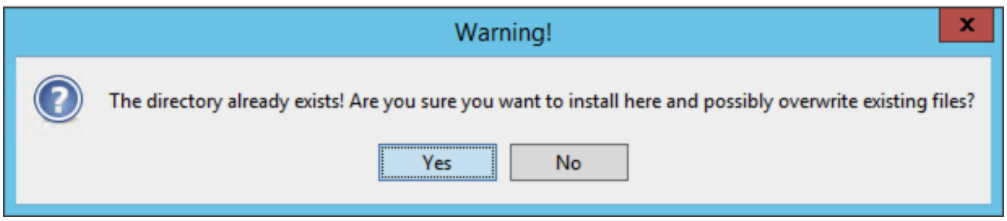


Fig. 12.2.12: Directory overwrite dialog.

The installer then presents a dialog for available data sets you may wish to install.

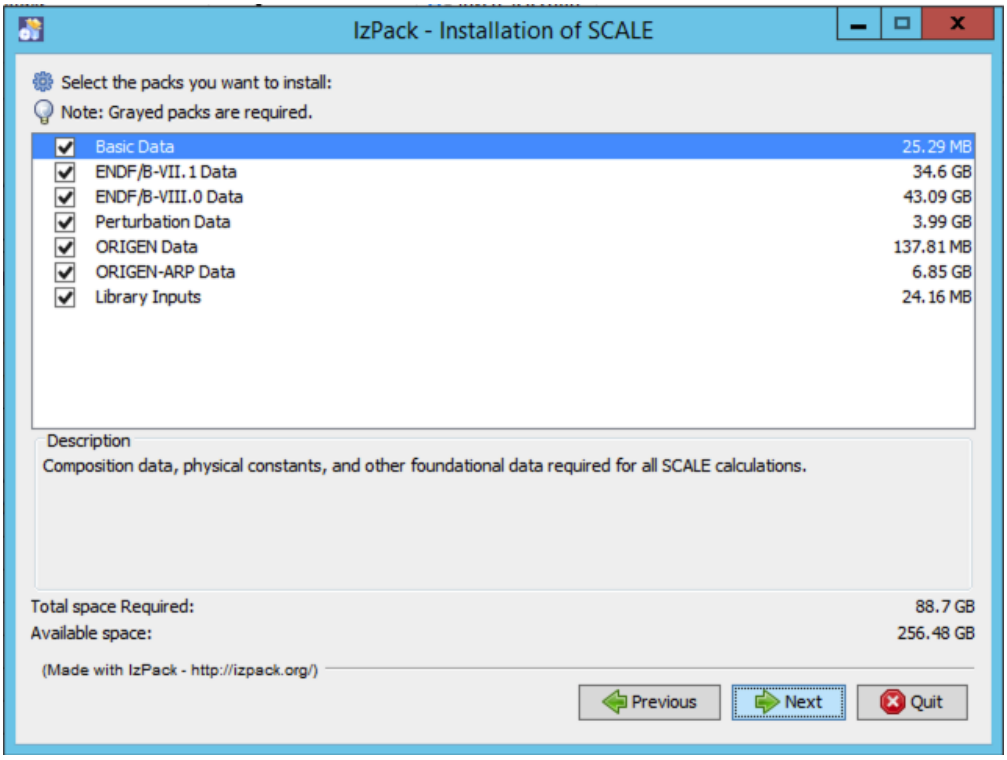


Fig. 12.2.13: Data selection dialog.

Please select the data needed to perform your desired SCALE calculations.

- Basic Data — Composition data, physical constants, and other foundational data required for all SCALE calculations
- ENDF/B-VII.1 Data — Nuclear Data from ENDF/B-VII.1 for CE and MG calculations, coupled n-gamma data from ENDF/B-VII.1 for MG calculations, and ENDF/B-VII.1 neutron covariance data (recommended for most calculations; required to run the SCALE sample problems)
- ENDF/B-VIII.0 Data — Nuclear Data from ENDF/B-VIII.0 for CE and MG calculations
- Perturbation data — 1,000 samples of data each for ENDF/B-VII.1 neutron covariance data, fission product yield covariance data, and radioactive decay covariance data (required for Sampler calculations perturbing nuclear data)

- ORIGEN Data — Activation, depletion and decay data from ENDF/B-VII.1 and JEFF 3.0/A (required for activation, depletion and decay calculations)
- ORIGEN reactor libraries — Pre-generated ORIGEN reactor libraries for many fuel types (required for spent fuel characterization and source terms calculations)

Next, the installer may ask for the location of the data .pak files that are unpacked during the installation process. Please direct the installer to the location of these data files, which are part of the SCALE distribution and may be on the distribution media or copied to a local directory as shown below.

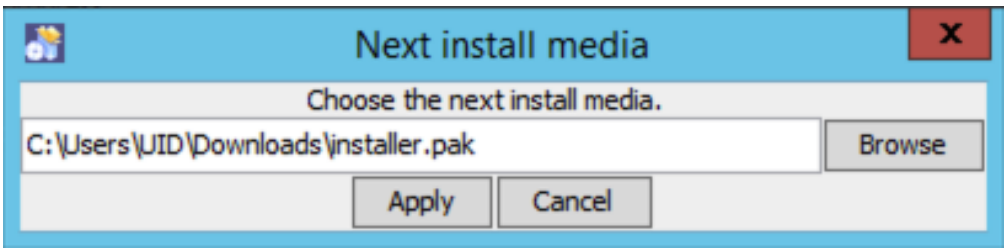


Fig. 12.2.14: Media selection dialog.

The installer will display the progress of the installation. If installing from a DVD set, part way through the installation, you will be prompted for `installer.pak.1`. Please insert the PAK.1 disk and click *Apply*.

The installation from PAK.1 will complete and prompt for `installer.pak.2`. Please insert the PAK.2 disk and click *Apply*.

PAK.2 will finish. Proceed with all DVDs or files in the delivery until you have completed your installation of SCALE 6.3 data.

12.3 RUNNING SCALE

SCALE is run by using the Fulcrum user interface or by invoking the SCALE Runtime Environment, *scalerte*, from the command line. Note that this release does not include the previous interfaces GeeWiz or OrigenArp for running SCALE on Windows, as Fulcrum should be used instead. Additionally, previous shortcuts like *runscale* are not implemented.

12.3.1 RUNNING SCALE FROM FULCRUM

The most convenient way to run SCALE from a desktop is by launching Fulcrum. The Fulcrum executable is provided in the *bin* directory where SCALE was installed (e.g., `C:\SCALE-6.3.0\bin\Fulcrum.exe`). Fulcrum includes an online help document to assist users with its many features, and it includes links to the user manual and primers.

12.3.1.1 Windows

For a Windows installation, launch Fulcrum from either the start menu or the shortcut provided in the *SCALE-6.3.0* folder on the desktop.

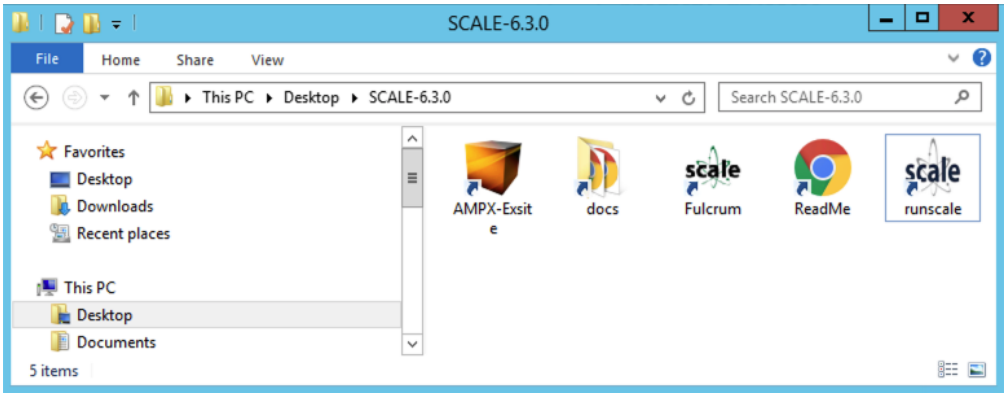


Fig. 12.3.1: Example Windows desktop SCALE shortcuts folder.

12.3.1.2 Linux

For a Linux installation, launch Fulcrum directly from the bin directory of the SCALE installation (e.g., `/scale/scale6.3.0/bin/Fulcrum`).

12.3.1.3 Mac OS X

For a Mac installation, launch Fulcrum by executing the *SCALE-6.3.0* app in the Applications directory.

12.3.2 RUNNING SCALE FROM THE COMMAND LINE

Using the command line, SCALE can be executed using the *scalerte* command from the *bin* directory inside the SCALE installation. Paths for each platform are shown below based on recommended installation directories. Your directory may differ based on the installation.

12.3.2.1 Windows

```
C:\\SCALE-6.3.0\bin\scalerte.exe
```

Note: Note that Norton Antivirus may require an exception for scalerte.exe on Windows to prevent the executable from being quarantined.

12.3.2.2 Linux

```
/scale/scale6.3.0/bin/scalerte
```

12.3.2.3 Mac

```
/Applications/SCALE-6.3.0.app/Contents/Resources/bin/scalerte
```

The SCALE runtime environment provides several command line options. The usage is as follows.

```
scalerte [options] inputfile(s) [options] [inputfile(s)]
```

Where `inputfile(s)` are one or more files or file patterns (`test.inp`, or `test*.inp`, etc.).

Where *options* are:

`-a`: Specify alias file.

```
-a path/to/aliasesfile
```

-f: Add hostname to output filename. Produces inputfile.hostname.out

-h: Print this information as a help message.

-I: Number of threads to use for MPI/OpenMP directives. -I 4

-m: Print information messages as SCALE executes.

-M: Specify a machine names file for SCALE parallel capabilities.

```
-M /path/to/machine/names/file
```

-n: Nice level on Nix systems, ignored on Windows. Default: -n 2

-N: Number of MPI processes to run. -N 20

-o: Overrides the default *inputfile.out* output name. The *.out* extension is appended by scalerte, so there is no need to specify the extension.

```
-o path/to/outputfile
```

Note: If the *path/to/outputfile* already exists, it will be deleted. If this option is specified while in stack mode (multiple input files), the value provided is prepended to the inputfile's basename.

```
scalerte triton* -o myout results in output names myout.triton*.out
```

-z: Add date to the output filenames.

Produces output files in the form of

```
inputfile.yyyy.MM.ddThh.mm.ss.out
inputfile.yyyy.MM.ddThh.mm.ss.msg
inputfile.yyyy.MM.ddThh.mm.ss.etc
```

Where:

- yyyy- is the year of execution.
- MM- is the month of execution.
- dd- is the day of execution.
- hh- is the hour of execution.
- mm- is the minute of execution.
- ss- is the second of execution.

12.3.3 EXAMPLE INVOCATION

For users familiar with previous invocations of the SCALE batch script, this usage is no longer valid. The only valid entry point is scalerte. Scalerte can be executed from anywhere with a fully qualified path such as `C:\SCALE-6.3.0\bin\scalerte` or `/scale/scale6.3.0/bin/scalerte`, for example.

Invoke SCALE on a single input file named *HelloWorld.inp.*

```
scalerte HelloWorld
```

or

```
scalerte HelloWorld.inp
```

Invoke SCALE on all input files patterned *HelloWorld*.inp.*

```
scalerte HelloWorld*.inp
```

Invoke SCALE on all input files patterned *HelloWorld*.inp* and print runtime messages to the console.

```
scalerte HelloWorld*.inp -m
```

Invoke SCALE on all input files patterned *HelloWorld*.inp* and include hostname and date/time in the output file's name.

```
scalerte HelloWorld*.inp -fz
```

or

```
scalerte HelloWorld*.inp -f -z
```

Invoke SCALE on *HelloWorld.inp* and rename output to be *MyHello.out.*

```
scalerte HelloWorld -o MyHello
```

or

```
scalerte HelloWorld.inp -o MyHello
```

Invoke SCALE on all files patterned *HelloWorld*.inp* and rename output to be *MyHelloWorld*.out*.

Note: When SCALE is run in stack mode (multiple inputs), the output override is prepended to the input file's name.

```
scalerte HelloWorld*.inp -o My
```

Invoke SCALE on *HelloWorld.inp* and keep the working directory.

```
scalerte HelloWorld.inp -r
```

Invoke SCALE on *HelloWorld.inp* and override and keep the working directory.

```
scalerte HelloWorld.inp -r -T myHelloWorldTempDir
```

Invoke SCALE on *HelloWorld.inp* and specify the number of threads to be 4.

```
scalerte HelloWorld.inp -I 4
```

12.3.4 SCALE VARIABLES

This section describes the environment variable used within *scalerte*. These variables can be accessed through SCALE's *shell* module to populate the working directory and/or to return SCALE-generated files that were not returned by *scalerte*.

Shell is used to perform system commands inside a problem before, after, or between explicitly called modules. It is usually used in SCALE to link a file from one name and place to another name and place, to delete files or directories, and to move or copy files. The use of shell in an input has the following form:

```
=shell
 *** System Commands ***
end
```

where *System Commands* are any UNIX or DOS command(s). This is particularly useful in tandem with the following environment variables when files not automatically returned are desired. The following example shell command copies the x16 file back alongside the output file.

```
=shell
cp x16 ${OUTBASENAME}.x16
end
```

Please note the syntax of Nix and Windows environment variables, ${VAR} and %VAR% respectively, are interchangeable, as *scalerte* and *shell* understand both.

Below are the seven primary locations known by *scalerte*:

- The user's home directory, HOME

  *Nix systems, ${HOME}, /home/uid

  *Windows, %HOME%, C:Usersuid

- The directory of SCALE, SCALE

  *Nix systems, ${SCALE}, location of user's installation—typically `/scale/scale#`

  *Windows, %SCALE%, location of user's installation—typically C:SCALE#

- The directory of the input file, INPDIR

  *Nix systems, ${INPDIR}

  *Windows, %INPDIR%

- The directory of the output file, OUTDIR, which by default is the same as INPDIR, because the output file is written next to the input file

  *Nix systems, ${OUTDIR}

  *Windows, %OUTDIR%

- The directory from which SCALE was invoked, the return directory, RTNDIR, which is the directory your console will return to upon completion

  *Nix systems, ${RTNDIR}

  *Windows, %RTNDIR%

- The directory that contains the SCALE data, DATA

  *Nix systems, ${DATA}

  *Windows, %DATA%

- The working directory for a given input file, TMPDIR, or shorthand TMP

  *Nix systems, ${TMPDIR}, ${TMP}

  *Windows, %TMPDIR%, %TMP%

There are several secondary locations in the SCALE directory tree. These are as follows:

- The directory containing the platform-specific compiled programs, PGMDIR, or legacy PGM_DIR:

  *Nix systems, ${PGMDIR}, ${PGM_DIR}

  *Windows, %PGMDIR%, %PGM_DIR%

Lastly, there are several environment variables provided for convenience and/or that are associated with output data that can be useful.

- The directory containing the ORIGEN data files, ORIGENDIR

  *Nix systems, ${ORIGENDIR}

  *Windows, %ORIGENDIR%

- The base name of the input file, BASENAME, which is the name of the input file without both absolute path and extension

  *Nix systems, ${BASENAME}, or ${CASE_NAME}

  *Windows, %BASENAME%, or %CASE_NAME%

- The base name of the output file, OUTBASENAME, or legacy CASE_NAME, which is the name of the output file without both absolute path and extension

  *Nix systems, ${OUTBASENAME}, or ${CASE_NAME}

  *Windows, %OUTBASENAME%, or %CASE_NAME%

- The base name of the output file, OUTBASE, which is the absolute name of the output file without file extension

  *Nix systems, ${OUTBASE}

  *Windows, %OUTBASE%

- The absolute path to the input file, INPUTFILE

  *Nix systems, ${INPUTFILE}

  *Windows, %INPUTFILE%.

- The absolute path to the output file, OUTFILE.

  *Nix systems, ${OUTFILE}

  *Windows, %OUTFILE%

- The directory containing USLSTATS output, USLDIR (If USLSTATS data were output, then these data would be located in OUTDIROUTBASENAME.uslstats directory.)

  *Nix systems, ${USLDIR}

  *Windows, %USLDIR%

- The directory containing CENTRM output, CENTRMDIR (If CENTRM data were output, then these data would be located in OUTDIROUTBASENAME.centrmfiles directory.)

  *Nix systems, $CENTRMDIR

  *Windows, %CENTRMDIR%

- The directory containing XSDRNPM output, XSDRNDIR (If XSDRNPM data were output, then these data would be located in OUTDIROUTBASENAME.xsdrnfiles directory.)

  *Nix systems, $XSDRNDIR

  *Windows, %XSDRNDIR%

- The platform-specific file separator, FS, which is either backslash () on Windows, or forward slash (/) on Nix systems

  *Nix systems, ${FS}

  *Windows, %FS%

12.3.5 PARALLEL EXECUTION CAPABILITY

SCALE 6.3 contains four modules and sequences that have distributed memory (MPI) parallelism: KENO V.a, KENO-VI, Sampler, and ORIGAMI. However, the binary executable files distributed with SCALE do not have MPI enabled. To run one of these codes in parallel, the user must first build SCALE with MPI enabled (see build instructions in this guide). Control modules like CSAS6, T6-DEPL, and TSUNAMI-3D-K6 automatically initiate the parallel version of KENO–VI in a parallel SCALE build if the user provides the required arguments as summarized below. When running a standalone code in parallel (such as KENO-VI), a "%" prefix is required on the sequence specification record in the input file (e.g., =%keno-vi). Parallel code execution is available on Linux and Mac systems but is not available on Windows PCs.

The executable binary code distributed with SCALE only enables serial calculations. If MPI parallelism is desired, then the source code must be compiled with MPI support enabled for the platform and configuration where the code will be executed in parallel. The SCALE build configuration sets some variables (*SCALEMPI*, *SCALECMDS*, and *MPIRUN*), depending on the third-party MPI package for the SCALE driver, which prepares two different execution environments with these variable sets for both serial and parallel code execution.

Executing SCALE in parallel is initiated by *scalerte* when the user provides the necessary MPI command line arguments with the two options, *–N* and *–M*, which specify the number of MPI processes and machine names, respectively. To run an input called `HelloWorld.inp` from an MPI build on SCALE on two nodes specified in the machine file *mach*, use the following command:

```
scalerte -N 2 -M mach HelloWorld.inp
```

With these options, *scalerte* sets *NTASKS* and *MACHINEFILE* variables, depending on the user request, and passes them to the SCALE driver. During the parallel code execution process, the driver invokes the MPIRUN wrapper to run the parallel functional module across MPI processes. An environment variable *MPIARGS* is available for the driver to pass some user-defined MPI options to MPIRUN wrapper.

12.4 SCALE SAMPLE PROBLEMS

The SCALE sample problems are designed by the SCALE developers to verify the installation and functionality of SCALE relative to the expected results. Users are urged to run the sample problems to verify the proper installation of SCALE. ORNL has provided a set of reference results from each sample problem against which the results of each installation can be compared.

The sample problems are most easily run through Fulcrum by selecting *Run SCALE Verification* from the *Run* menu on the main menu bar. This will run all of the sample problems sequentially and will present any differences from the ORNL-generated results.

The SCALE runtime environment, *scalerte*, which is described in the subsequent section, has a built-in scripting interface to allow for invocation of several groupings of sample problems, including individual problems, problems for a specific module, subsets of modules, or all samples. The sample problems will print a message indicating the sample problem currently running, followed by the differences between ORNL-generated results and the newly generated results for the sample problem(s). Note that there may be up to an hour's delay before messages are printed to the console. The final message will indicate that the "Process finished with a 0 return code . . . ." If no differences or minimal differences are noted for a particular sample problem, then SCALE has been properly installed for the tested functionality.

The set of sample problems is located in the samples directory of the SCALE installation.

To run all sample problems, execute the following command:

```
scalerte @samples/samples
```

To invoke sample problems for a particular module or sequence, use

```
scalerte @samples/samples modulename
```

where `modulename` is the name of the module or sequence to test, e.g., *centrm*.

To invoke a single sample problem, do the following.

```
scalerte @samples/samples problemname
```

where `problemname` is the name of the specific sample problem to test, e.g., `centrm-pwr.inp`.

The above *modulename* and *problemname* examples can be combined and repeated to exercise sets of sample problems of interest. For example, run *csas* and *origen* sequence sample problems in addition to the *centrm-pwr.inp* sample.

```
scalerte @samples/samples csas centrm-pwr.inp origen
```

As the sample problems execute, feedback will be provided to the screen such as that shown in the example below. If no differences or only small differences are reported between the ORNL results and the currently generated results for each sample, then SCALE has been properly installed and configured for the tested functionality.

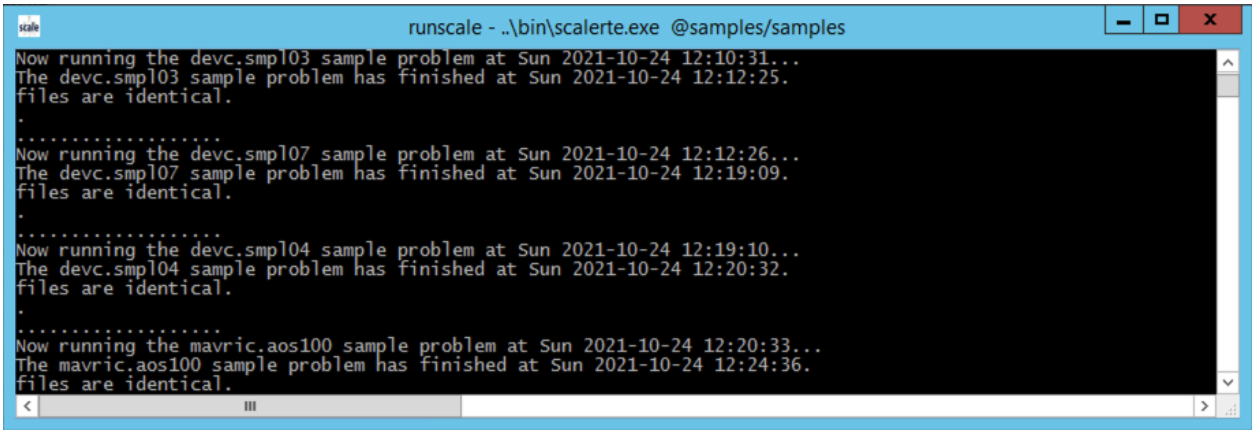


Fig. 12.4.1: Sample problem output messages.

For any problems or questions, please contact scalehelp@ornl.gov.

12.5 BUILD INSTRUCTIONS

These instructions are only for those who wish to recompile their SCALE binaries. If you are running SCALE using the precompiled binary executable files distributed with SCALE, then please disregard this section.

For SCALE 6.3 and subsequent updates, the build configuration has been completely renewed relative to all previous versions. The SCALE build is now based on CMake from KitWare, which supports a consistent experience on Linux, Mac, and Windows.

12.5.1 OVERVIEW

There are four main steps to create and install SCALE binaries

1. Install compilers and third-party libraries (TPLs).
2. Configure CMake to generate a native build tree.
3. Compile all executables and libraries.
4. Install to deploy all executables into a configuration ready for execution.

12.5.2 REQUIRED RESOURCES

SCALE requires the following programs in order to compile:

1. Fortran Compiler
    i. Windows – Intel Fortran compiler 15.0
    ii. Linux - Intel Fortran compiler 14.1+/GNU gfortran 4.8.3+ compiler
    iii. Mac OS – GNU gfortran 4.8.5+ compiler
2. C/C++ Compiler
    i. Windows – Intel C/C++ compiler 15.0.=
    ii. Linux – Intel C/C++ compiler 14.1+ / GNU g++/gcc 4.8.3+ compiler
    iii. Mac OS – GNU g++/gcc 4.8.5+ compiler

3. CMake 3.13+ - Platform independent build configuration for Linux, Darwin, and Windows

SCALE requires the following libraries, which are not distributed with SCALE:

LAPACK libraries:module

a. lapack

b. blas

You must modify the LAPACK variable in configure_scale_gnu.sh to point to your installation of LAPACK prior to configuration.

The following optional package is required by SCALE if MPI support is desired:

> OpenMPI-1.8.1+ - Optional for SCALE build with MPI support on Linux and Mac OS (currently, SCALE does not support MPI for Windows)

12.5.3 MAC OSX RESOURCES

Mac OSX does not come preloaded with the necessary resources for compiling SCALE. Macports provides a simple mechanism for retrieving almost all of the required resources necessary to compile SCALE on a Mac. This process has only been tested on Mac OSX 10.11.6.

You can download the latest macports from http://www.macports.org/. You will need XCode (free from the App Store) with commandline tools installed for macports to work.

Important: You will need administrative privileges to install macports and the necessary resources.

Please see macports instructions for xcode installation/requirements. http://guide.macports.org/#installing.xcode

After installing these resources, a new shell session is necessary to access the new tools.

..note:: If your rsync port is blocked by the firewall (as is the case at ORNL), you may synchronize over http by updating your */opt/local/etc/macports/sources.conf* file to change the line:

```
rsync://rsync.macports.org/release/tarballs/ports.tar [default]
```

to

```
http://www.macports.org/files/ports.tar.gz [default]
```

The following commands will install all necessary resources.

If the rsync port is blocked, use

```
$> sudo port -d sync
```

If the rsync port is open, use

```
$>sudo port selfupdate
```

Regardless of rsync status, execute the following commands:

```
$>sudo port install gcc48
$>sudo port select gcc mp-gcc48
$>sudo port install qt4-mac
```

These commands upgrade the default compilers from gcc 4.2.2 to gcc 4.8.3, and they install qt4.8.6.

12.6 CMAKE CONFIGURATION

CMakeLists.txt files can be found throughout SCALE. From the SCALE root directory, these CMakeLists.txt files create a tree of included directories called the SOURCE TREE. Namely, the source directories are

```
packages/AmpxLib

XSProc

ScaleLib

etc.
```

To configure a build, call cmake on the root CMakeLists.txt file, namely `scale_dir/CMakeLists.txt`. CMake takes your source tree and creates a BUILD TREE. The build tree contains or will contain the build configuration, the Make or NMake files, and all compilation output: object files, archive libraries, and binary executables.

SCALE requires several TPLs, specifically, QT and Lapack. These TPLs must be specified at the time of configuration. For ease of use, configuration scripts for every supported platform are available in the scripts directory of the source code. These scripts describe the necessary variables to define. A user's modifications to these scripts should be limited to the path to the root directories for the TPLs.

12.6.1 RECOMMENDED CONFIGURATION PROCEDURE

After unpacking the source code, navigate to the root scale directory (<SCALE_SRC_ROOT>. You will see CMakeLists.txt, PackagesList.cmake, and CTestConfig.cmake. This is the root of the source tree, to which you will point CMake. This example demonstrates creating build trees for multiple configurations for your working copy.

12.7 LINUX AND MAC CONFIGURATION

- **Make Build Directory**

```
mkdir build

mkdir -p build/intel \**this could be any directory

or

mkdir -p build/gcc
```

- **CMake Initialization**

  Copy the cmake script from the scripts directory to your build directory. Update the cmake script with your TPL specifications.

  - Serial SCALE (without MPI support)

```
cp <SCALE_SRC_ROOT>/script/configure_scale_gnu.sh
build/gcc

chmod u+x build/gcc/configure_scale_gnu.sh
```

- Parallel SCALE (with MPI support)

```
cp script/configure_scale_mpi.sh build/gcc

chmod u+x build/gcc/configure_scale_mpi.sh
```

- **Create Your Configuration**

  Create your configuration by pointing this script at the source tree root

```
cd build/gcc
```

  Edit the CMakelists.txt file in <SCALE_SRC_ROOT>/Trilinos/packages/anasazi/src/ so as to comment out the following lines:

```
ASSERT_DEFINED(Anasazi_ENABLE_ThyraEpetraAdapters)

ASSERT_DEFINED(Anasazi_ENABLE_ThyraCore)

ASSERT_DEFINED(Anasazi_ENABLE_Tpetra)
```

  – Serial SCALE (without MPI support)

```
./configure_scale_gnu.sh ../..
```

  – Parallel SCALE (with MPI support)

```
./configure_scale_mpi.sh ../..
```

/* **Configuration Output….** */

**If you add or remove source files from the source tree, CMake will NOT see these modifications unless a CMake file is modified (CMakeLists.txt,PackagesList.cmake,etc…).

CMake will re-evaluate the entire source tree when any CMake file has been modified. If CMake does not pick up the addition/removal of sources files, then the easiest way to update the build tree is to "touch" any CMake file in the source tree.

12.7.1 WINDOWS CONFIGURATION

Instructions are not documented here. Please contact scalehelp@ornl.gov for assistance.

12.7.2 COMPILATION

Every library and executable is a TARGET. Calling make on Linux and Mac and nmake on Windows from the root of your build tree (<SCALE_SRC_ROOT>/build/gcc from the previous example) will build ALL targets. There are two options when building specific targets. For example, MavricUtilities contains a number of executables: mtadder, mtaverager, and so on.

You may invoke make mtadder to build mtadder alone. Alternatively, you can build it from build/gcc, cd packages/MavricUtilities and make to compile ALL targets in mavricUtilities.

12.7.3 COMPILATION FLAGS

You may modify `CMAKE_Fortran_FLAGS`, `CMAKE_C_FLAGS`, and `CMAKE_CXX_FLAGS` on the cmake command invocation line.

12.7.4 INSTALLATION

CMake provides the install target, which installs all binaries from the current directory down. For example, `make install'` from `build/gcc` will install any targets, which would install aim, mavric, mtadder, mtaverager, etc., while `cd packages/Mavric` and `make install` will only install targets declared in the Mavric directory.

***Note that `make install` re-evaluates the build for all dependencies. For example, package A depends on packages B and D. Package D depends on package E. Therefore, `make install` for package A would result in packages E, D, and B being re-evaluated and rebuilt if necessary. Thus, if you know you want to build and install, you can save time building by simply doing a `make install`. An alternative is `make install/fast`, which will skip. . . ..

REFERENCES

[IntroJED+20] Seth R. Johnson, Thomas M. Evans, Gregory G. Davidson, Steven P. Hamilton, Tara M. Pandya, Katherine E. Royston, and Elliott D. Biondo. Omnibus User Manual. Technical Report ORNL/TM-2018/1073, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2020.

[IntroPJE+16] T. M. Pandya, S. R. Johnson, Evans, T.M., G. G. Davidson, S. P. Hamilton, and A. T. Godfrey. Implementation, capabilities, and benchmarking of shift, a massively parallel monte carlo radiation transport code. *Journal of Computational Physics*, 308:239–272, 2016. Publisher: Elsevier.

[CSAS5Mih99] J. T. Mihalczo. Brief summary of unreflected and unmoderated cylindrical critical experiments with oralloy at Oak Ridge. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1999.

[CSAS5Tho64] J. T. Thomas. CRITICAL THREE-DIMENSIONAL ARRAYS OF NEUTRON-INTERACTING UNITS. PART II. U (93.2) METAL. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1964.

[CSAS5Tho73] Joseph T. Thomas. Critical three-dimensional arrays of U (93.2)-metal cylinders. *Nuclear Science and Engineering*, 52(3):350–359, 1973. Publisher: Taylor & Francis.

[CSAS-ShiftESSC10] Thomas M. Evans, Alissa S. Stafford, Rachel N. Slaybaugh, and Kevin T. Clarno. Denovo: A new three-dimensional parallel discrete ordinates code in SCALE. *Nuclear technology*, 171(2):171–200, 2010.

[CSAS-ShiftPJE+16] T. M. Pandya, S. R. Johnson, Evans, T.M., G. G. Davidson, S. P. Hamilton, and A. T. Godfrey. Implementation, capabilities, and benchmarking of shift, a massively parallel monte carlo radiation transport code. *Journal of Computational Physics*, 308:239–272, 2016. Publisher: Elsevier.

[DEVCESSC10] Thomas M. Evans, Alissa S. Stafford, Rachel N. Slaybaugh, and Kevin T. Clarno. Denovo: A new three-dimensional parallel discrete ordinates code in SCALE. *Nuclear technology*, 171(2):171–200, 2010.

[DEVCGPJD+11] Sedat Goluoglu, Lester M. Petrie Jr, Michael E. Dunn, Daniel F. Hollenbach, and Bradley T. Rearden. Monte Carlo criticality methods and analysis capabilities in SCALE. *Nuclear Technology*, 174(2):214–235, 2011.

[DEVCIPE+09] Ahmad M. Ibrahim, Douglas E. Peplow, Thomas M. Evans, John C. Wagner, and Paul PH Wilson. Improving the Mesh Generation Capabilities in the SCALE Hybrid Shielding Analysis Sequence. In *Transactions of the American Nuclear Society*, volume 100, 302. 2009.

[DEVCJoh13] Seth R. Johnson. Fast mix table construction for material discretization. In *Proceedings of the 2013 International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering-M and C 2013*. 2013.

[CritSafetyESSC10] Thomas M. Evans, Alissa S. Stafford, Rachel N. Slaybaugh, and Kevin T. Clarno. Denovo: A new three-dimensional parallel discrete ordinates code in SCALE. *Nuclear technology*, 171(2):171–200, 2010.

[CritSafetyGPJD+11] Sedat Goluoglu, Lester M. Petrie Jr, Michael E. Dunn, Daniel F. Hollenbach, and Bradley T. Rearden. Monte Carlo criticality methods and analysis capabilities in SCALE. *Nuclear Technology*, 174(2):214–235, 2011.

[CritSafetyPJE+16] T. M. Pandya, S. R. Johnson, Evans, T.M., G. G. Davidson, S. P. Hamilton, and A. T. Godfrey. Implementation, capabilities, and benchmarking of shift, a massively parallel monte carlo radiation transport code. *Journal of Computational Physics*, 308:239–272, 2016. Publisher: Elsevier.

[CritSafetyPPJ09] Douglas E. Peplow and Lester M. Petrie Jr. Criticality Accident Alarm System Modeling with SCALE. In *Proceedings of the 2009 International Conference on Advances in Mathematics, Computational Methods, and Reactor Physics*. 5 2009. URL: http://inis.iaea.org/Search/search.aspx?orig_q=RN:40080894 (visited on 2020-08-10).

[CritSafetyUMN97] Taro Ueki, Takamasa Mori, and Masayuki Nakagawa. Error estimations and their biases in Monte Carlo eigenvalue calculations. *Nuclear science and engineering*, 125(1):1–11, 1997. Publisher: Taylor & Francis.

[TRITONBPBR17] B. R. Betzler, J. J. Powers, N. R. Brown, and B. T. Rearden. Molten Salt Reactor Tools in SCALE. In *ANS Mathematics & Computation Topical Meeting, Apr. 16–20*. Jeju, Korea, 2017.

[TRITONBPW17] B. R. Betzler, J. J. Powers, and A. Worrall. Molten salt reactor neutronics and fuel cycle modeling and simulation with scale. *Annals of Nuclear Energy*, 101:489–503, 2017. doi:10.1016/j.anucene.2016.11.040.

[TRITONDB11] M.D. DeHart and S. M. Bowman. Reactor physics methods and analysis capabilities in scale. *Nuclear Technology*, 174(2):196–213, 2011. doi:10.13182/NT174-196.

[TRITONVBWF20] P. V. Vicente, B. R. Betzler, W. A. Wieselquist, and M. Fratoni. Modeling Molten Salt Reactor Fission Product Removal with SCALE. Technical Report ORNL/TM-2019/1418, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2020.

[POLARISBS96] S. M. Bowman and T. Suto. Scale-4 analysis of pressurized water reactor critical configurations: Volume 5, North Anna Unit 1 Cycle 5. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1996.

[POLARISGod14] Andrew T. Godfrey. VERA core physics benchmark progression problem specifications. *Consortium for Advanced Simulation of LWRs*, 2014.

[POLARISJWM+21] Matthew A. Jessee, William A. Wieselquist, Ugur Mertyurek, Kang Seog Kim, Thomas M. Evans, Steven P. Hamilton, and Cole Gentry. Lattice physics calculations using the embedded self-shielding method in polaris, part i: methods and implementation. *Annals of Nuclear Energy*, 150:107830, 2021.

[POLARISRE13] B. Roque and M. Erlund. International Comparison of a Depletion Calculation Benchmark on Fuel Cycle Issues, Results from Phase 1 on UOx Fuels. Technical Report, NEA/NSC/DOC, 2013.

[POLARISWK12] Mark L. Williams and Kang Seog Kim. The embedded self-shielding method. In *PHYSOR 2012*. 2012.

[MAVRIC-BK98] Randal S. Baker and Kenneth R. Koch. An Sn algorithm for the massively parallel CM-200 computer. *Nuclear Science and Engineering*, 128(3):312–320, 1998. Publisher: Taylor & Francis.

[MAVRIC-GV91] R. P. Gardner and K. Verghese. Monte Carlo nuclear well logging benchmark problems with preliminary intercomparison results. *Nuclear Geophysics*, 5(4):429–438, 1991.

[MAVRIC-HW03] Alireza Haghighat and John C. Wagner. Monte Carlo variance reduction with deterministic importance functions. *Progress in Nuclear Energy*, 42(1):25–53, 2003. Publisher: Elsevier.

[MAVRIC-IPE+09] Ahmad M. Ibrahim, Douglas E. Peplow, Thomas M. Evans, John C. Wagner, and Paul PH Wilson. Improving the Mesh Generation Capabilities in the SCALE Hybrid Shielding Analysis Sequence. In *Transactions of the American Nuclear Society*, volume 100, 302. 2009.

[MAVRIC-Joh13] Seth R. Johnson. Fast mix table construction for material discretization. In *Proceedings of the 2013 International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering-M and C 2013*. 2013.

[MAVRIC-Lat71] K. D. Lathrop. Remedies for ray effects. *Nuclear Science and Engineering*, 45(3):255–268, 1971. Publisher: Taylor & Francis.

[MAVRIC-Lat68] Kaye D. Lathrop. Ray effects in discrete ordinates equations. *Nuclear Science and Engineering*, 32(3):357–369, 1968. Publisher: Taylor & Francis.

[MAVRIC-Pep11] Douglas E. Peplow. Monte Carlo shielding analysis capabilities with MAVRIC. *Nuclear Technology*, 174(2):289–313, 2011. Publisher: Taylor & Francis.

[MAVRIC-RC98] W. A. Rhoades and R. L. Childs. DOORS 3.2, one-, two-, three-dimensional discrete ordinates neutron/photon transport code system. Technical Report Radiation Shielding Information Center package CCC-650, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1998.

[MAVRIC-RS97] Wayne A. Rhoades and D. B. Simpson. The TORT three-dimensional discrete ordinates neutron/photon transport code (TORT version 3). Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1997.

[MAVRIC-Wag97] John C. Wagner. *Acceleration of Monte Carlo shielding calculations with an automated variance reduction technique and parallel processing*. PhD Thesis, Pennsylvania State University, 1997.

[MAVRIC-Wag02] John C. Wagner. An automated deterministic variance reduction generator for Monte Carlo shielding applications. In *Proceedings of the American Nuclear Society 12th Biennial RPSD Topical Meeting*, 14–18. Citeseer, 2002.

[MAVRIC-WBP07] John C. Wagner, Edward D. Blakeman, and Douglas E. Peplow. Forward-weighted CADIS method for global variance reduction. In *Transactions of the American Nuclear Society*, volume 97, 630. 2007.

[MAVRIC-WH98] John C. Wagner and Alireza Haghighat. Automated variance reduction of Monte Carlo shielding calculations using the discrete ordinates adjoint function. *Nuclear Science and Engineering*, 128(2):186–208, 1998. Publisher: Taylor & Francis.

[MAVRIC-WPE20] John C. Wagner, Douglas E. Peplow, and Thomas M. Evans. Automated Variance Reduction Applied to Nuclear Well-Logging Problems. *Nuclear Science and Engineering*, 2020.

[MAVRIC-WPM14] John C. Wagner, Douglas E. Peplow, and Scott W. Mosher. FW-CADIS Method for Global and Regional Variance Reduction of Monte Carlo Radiation Transport Calculations. *Nuclear Science and Engineering*, 176(1):37–57, January 2014. URL: https://www.tandfonline.com/doi/full/10.13182/NSE12-33 (visited on 2020-08-18), doi:10.13182/NSE12-33.

[MAVRIC-WDP+09] Dorothea Wiarda, Michael E. Dunn, Douglas E. Peplow, Thomas M. Miller, and Hatice Akkurt. Development and Testing of Endf/b-vi. 8 and Endf/b-vii. 0 Coupled Neutron-gamma Libraries for Scale 6. Technical Report ORNL/TM-2008/047, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2009.

[MAVRIC-WH03] James M. Willenbring and Michael Allen Heroux. Trilinos users guide. Technical Report SAND2003-2952, Sandia National Laboratories, 2003.

[CAASMP13a] T.M. Miller and D.E. Peplow. Corrected user guidance to perform three-dimensional criticality accident alarm system modeling with SCALE. In *Transactions of the American Nuclear Society*, volume 108, 498–501. 1 2013.

[CAASMP13b] Thomas Martin Miller and Douglas E. Peplow. Guidance Detailing Methods to Calculate Criticality Accident Alarm Detector Response and Coverage. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1 2013. URL: https://www.osti.gov/biblio/1095654 (visited on 2020-08-10).

[CAASMP13c] Thomas Martin Miller and Douglas E. Peplow. Guide to Performing Computational Analysis of Criticality Accident Alarm Systems. Technical Report ORNL/TM-2013/211, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 9 2013. URL: https://www.osti.gov/biblio/1136788 (visited on 2020-08-10), doi:10.2172/1136788.

[CAASPW12] Douglas Peplow and Larry Wetzel. Methods for Detector Placement and Analysis of Criticality Accident Alarm Systems. In *Transactions of the American Nuclear Society*, volume 106. January 2012.

[CAASPLMP10] Douglas E. Peplow and Jr Lester M. Petrie. Criticality Accident Alarm System Modeling Made Easy with SCALE 6.1. In *Transactions of the American Nuclear Society*, volume 102, 297–299. 6 2010. URL: http://epubs.ans.org/?a=10467 (visited on 2020-08-10).

[CAASPPJ09] Douglas E. Peplow and Lester M. Petrie Jr. Criticality Accident Alarm System Modeling with SCALE. In *Proceedings of the 2009 International Conference on Advances in Mathematics, Computational Methods, and Reactor Physics*. 5 2009. URL: http://inis.iaea.org/Search/search.aspx?orig_q=RN:40080894 (visited on 2020-08-10).

[CAASWODonnellN12] L.L. Wetzel, B. O'Donnell, and W.D. Newmyer. CAAS detector placement using MAVRIC. In *Transactions of the American Nuclear Society*, volume 107, 644–647. 1 2012.

[MAV-AppC-Bec09] T. L. Becker. *Hybrid Monte Carlo/Deterministic Methods for Deep-Penetration Problems*. PhD Thesis, University of Michigan, 2009.

[MAV-AppC-BL09] T. L. Becker and E. W. Larsen. The application of weight windows to 'global'monte carlo problems. In *Proc. Conf. Advances in Mathematics, Computational Methods, and Reactor Physics*. American Nuclear Society Saratoga Springs, New York, 2009.

[MAV-AppC-CL01] Marc A. Cooper and Edward W. Larsen. Automated weight windows for global Monte Carlo particle transport calculations. *Nuclear Science and Engineering*, 137(1):1–13, 2001. Publisher: Taylor & Francis.

[MAV-AppC-ESSC10] Thomas M. Evans, Alissa S. Stafford, Rachel N. Slaybaugh, and Kevin T. Clarno. Denovo: A new three-dimensional parallel discrete ordinates code in SCALE. *Nuclear technology*, 171(2):171–200, 2010.

[MAV-AppC-Pep13] Douglas E. Peplow. Comparison of hybrid methods for global variance reduction in shielding calculations. In *Proceedings of the 2013 International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering-M and C 2013*. 2013.

[MAV-AppC-PME12] Douglas E. Peplow, Scott W. Mosher, and Thomas M. Evans. Consistent Adjoint Driven Importance Sampling Using Space, Energy and Angle. Technical Report ORNL/TM-2012/7, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2012.

[MAV-AppC-PMEW10] Douglas E. Peplow, Scott W. Mosher, Thomas M. Evans, and J. C. Wagner. Hybrid Monte Carlo/Deterministic Methods for Streaming/Beam Problems. In *American Nuclear Society Radiation Protection and Shielding Division 2010 Topical Meeting, Las Vegas, Nevada*. 2010.

[MAV-AppC-VRUS97] K. A. Van Riper, T. J. Urbatsch, and P. D. Soran. AVATAR – Automatic variance reduction in Monte Carlo calculations. Technical Report LA-UR-97-919; CONF-971005-10, Los Alamos National Laboratory, Los Alamos, NM (USA), May 1997. URL: https://www.osti.gov/biblio/527548 (visited on 2020-08-11).

[MAV-AppC-VWVdEH11] A. J. Van Wijk, Gert Van den Eynde, and J. E. Hoogenboom. An easy to implement global variance reduction procedure for MCNP. *Annals of Nuclear Energy*, 38(11):2496–2503, 2011. Publisher: Elsevier.

[ORIGEN-BA67] S. J. Ball and R. K. Adams. MATEXP: A General Purpose Digital Computer Program for Solving Ordinary Differential Equations by the Matrix Exponential Method. Technical Report ORNL/TM-1933, Union Carbide Corporation (Nuclear Division), Oak Ridge National Laboratory, 8 1967.

[ORIGEN-Bat10] H. Bateman. The Solution of Differential Equations Occurring in the Theory of Radioactive Transformations. *Proc. Cambridge Phil. Soc.*, 15:423, 1910.

[ORIGEN-Bel73] M. J. Bell. ORIGEN B-The ORNL Isotope Generation and Depletion Code . Technical Report ORNL-4628 (CCC-217), Union Carbide Corporation (Nuclear Division), Oak Ridge National Laboratory, 5 1973.

[ORIGEN-DJPB86] D. J. Pellarin, W. L. Matney and N. E. Bibler. :math:`\left (\alpha ,n \right )` neutron emission from dwpf glass. Technical Report DPST-86-212, Savannah River Laboratory, 1 1986. URL: https://www.osti.gov/servlets/purl/780500-Z8ooL7/native/.

[ORIGEN-Gau03] I. C. Gauld. MOX Cross-Section Libraries for ORIGEN-ARP. Technical Report ORNL/TM-2003/2, UT-Battelle, LLC, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 7 2003.

[ORIGEN-IA11] A. E. Isotalo and P. A. Aarnio. Comparison of depletion algorithms for large systems of nuclides. *Annals of Nuclear Energy*, 38(2):261 – 268, 2011. URL: http://www.sciencedirect.com/science/article/pii/S0306454910003889, doi:https://doi.org/10.1016/j.anucene.2010.10.019.

[ORIGEN-LL67] L. Lapidus and R. Luus. *Optimal Control of Engineering Processes*, pages 45–49. Blaisdell Publishing Co., Waltham, MA, 1967.

[ORIGEN-LHBP98] L. C. Leal, O. W. Hermann, S. M. Bowman, and C. V. Parks. ARP: Automatic Rapid Process for the Generation of Problem-Dependent SAS2H/ORIGEN-S Cross-Section Libraries. Technical Report ORNL/TM-13584, Lockheed Martin Energy Research Corporation, Oak Ridge National Laboratory, 4 1998.

[ORIGEN-ORN11] ORNL. SCALE: A Comprehensive Modeling and Simulation Suite for Nuclear Safety Analysis and Design". Technical Report ORNL/TM-2005/39, Version 6.1, Oak Ridge National Laboratory, Oak Ridge, TN, 6 2011.

[ORIGEN-Pus11] Maria Pusa. Rational Approximations to the Matrix Exponential in Burnup Calculations. *Nuclear Science and Engineering*, 169(2):155–167, 2011. URL: https://doi.org/10.13182/NSE10-81, doi:10.13182/NSE10-81.

[ORIGEN-Pus13] Maria Pusa. *Numerical Methods for Nuclear Fuel Burnup Calculations*. PhD thesis, Aalto University, 2013.

[ORIGEN-PL10] Maria Pusa and Jaakko Leppänen. Computing the Matrix Exponential in Burnup Calculations. *Nuclear Science and Engineering*, 164(2):140–150, 2010. doi:10.13182/NSE09-14.

[ORIGEN-Sho00] E. F. Shores. Data Updates for the SOURCES-4A Computer Code. Technical Report LA-UR-00-5016, Los Alamos National Laboratory, 10 2000. SOURCES-4C available from the Radiation Safety Information Computational Center (RSICC) as code package C00661.

[ORIGEN-Von62] D. R. Vondy. Development of a General Method of Explicit Solution to the Nuclide Chain Equations for Digital Machine Calculations. Technical Report ORNL/TM-361, Union Carbide Corporation (Nuclear Division), Oak Ridge National Laboratory, 10 1962.

[ORIGEN-Wil86] Mark L. Williams. *Perturbation theory for nuclear reactor analysis*. CRC Press, Inc., 1986.

[ORIGEN-WPC+99] W. B. Wilson, R. T. Perry, W. S. Charlton, T. A. Parish, G. P. Estes, T. H. Brown, E. D. Arthur, M. Bozoian, T. R. England, D. G. Madland, and J. E. Stewart. SOURCES 4A: Code for Calculating :math:`\left (\alpha ,n\right )`, Spontaneous Fission, and Delayed Neutron Sources and Spectra. Technical Report LA-13639-MS, Los Alamos National Laboratory, 9 1999.

[ORIGEN-WPS+83] W. B. Wilson, R. T. Pery, J. E. Stewart, T. R. England, D. G. Madland, and E. D. Arthur. Development of the SOURCES Code and Data Library for the Calculation of Neutron Sources and Spectra from :math:`\left ( \alpha ,n \right )` Reactions, Spontaneous Fission, and :math:`beta^-` Delayed Neutrons. Technical Report LA-9841-PR, Los Alamos National Laboratory, Los Alamos, NM, 1983.

[ORIGEN-WA02] George Wolberg and Itzik Alfy. An energy-minimization framework for monotonic cubic spline interpolation. *Journal of Computational and Applied Mathematics*, 143:145–188, 6 2002.

[ORIGEN-REFed68] R. E. Funderlic (ed.). The programmer's handbook-a compendium of numerical analysis utility programs. Technical Report AEC Research and Development Report K-1729, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2 1968.

[ORIGEN-RX-LIBSCSHW21] Riley M. Cumberland, Ryan T. Sweet, Ugur Mertyurek Robert Hall, and William A. Wieselquist. Isotopic and Fuel Lattice Parameter Trends in Extended Enrichment and Higher Burnup LWR Fuel, Volume II: BWR. Technical Report ORNL/TM-2020/1835, Oak Ridge National Laboratory, Oak Ridge, TN (USA), March 2021. doi:10.2172/1782042.

[ORIGEN-RX-LIBSHCWS21] Bob Hall, Riley Cumberland, William Wieselquist, and Ryan Sweet. Isotopic and Fuel Lattice Parameter Trends in Extended Enrichment and Higher Burnup LWR Fuel Vol I: PWR fuel. Technical Report ORNL/TM-2020/1833, Oak Ridge National Laboratory, Oak Ridge, TN (USA), February 2021. doi:10.2172/1779134.

[ORIGAMIOWHR94] O. W. Hermann, C. V. Parks and J. P. Renier. Technical Support for a Proposed Decay Heat Guide Using SAS2H/ORIGEN Data. Technical Report NUREG/CR-5625 (ORNL-6698), Martin Marietta Energy Systems, Inc., Oak Ridge National Laboratory, 9 1994.

[ORIGAMIRGI10] G. Radulescu, I. C. Gauld, and G. Ilas. SCALE 5.1 Predictions of PWR Spent Nuclear Fuel Isotopic Compositions. Technical Report ORNL/TM-2010/44, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 3 2010.

[ORIGAMIRGlW12] G. Radulescu, I. C. Gauld, G. llas, and J. C. Wagner. An Approach for Validating Actinide and Fission Product Burnup Credit Criticality Safety Analyses-Isotopic Composition Predictions. Technical Report NUREG/CR-7108, ORNL/TM-2011/509, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 4 2012.

[ORIGAMISGRT12] S. Skutnik, I.C. Gauld, C.E. Romano, and H.R. Trellue. Creating NDA 'Working Standards' through High-Fidelity Spent Fuel Modeling. In *Proceedings of the Institute of Nuclear Materials Management, 53rd Annual Meeting*. Orlando, FL, 7 2012.

[ORIGAMISHDLG13] S. Skutnik, F. Havlůj, D. Lago, and I. Gauld. Development of an Object-Oriented ORIGEN for Advanced Nuclear Fuel Modeling Applications. In *International Conference on Mathematics and Computational Methods Applied to Nuclear Science & Engineering (M&C 2013)*. Sun Valley, ID, 5 2013.

[TS-1D-AKPZ74] H. Alter, R. Kidman, R. B. abd Labauv, R. Protsik, and B. A. Zolotar. Cross Section Evaluation Working Group Benchmark Specification. Technical Report BNL 19302, National Neutron Cross Section Center (ENDF-202), Brookhaven National Laboratory , Upton, NY (USA), 11 1974.

[TS-1D-CLP86] Wilfred C., N. F. Landers, and L. M. Petrie. Validation of KENO V.a Comparison with Critical Experiments. Technical Report ORNL/CSD/TM-238, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 12 1986.

[TS-1D-IAKS07] Kalcko Ivanov, M. Avramova, Ivan Aleksander Kodeli, and E. Sartori. *Benchmark for uncertainty analysis in modeling (UAM) for design, operation and safety analysis of LWRs*. Citeseer, 2007.

[3d1] H. Hurwitz, Jr., "*A Note on the Theory of Danger Coefficients*, Tech. Rep. KAPL-98, Knolls Atomic Power Laboratory, 1948.

[3d2] B. C. Kiedrowski, "Adjoint Weighting for Continuous-Energy Monte Carlo Radiation Transport,", doctoral dissertation, University of Wisconsin (2009).

[3d3] C. M. Perfetti, "Advanced Monte Carlo Methods for Eigenvalue Sensitivity Coefficient Calculation,", doctoral dissertation, University of Michigan (2012).

[3d4] K. Banerjee, W. M. Martin, "Kernel Density Estimate Monte Carlo Global Flux Tallies," *Proc*eedings *of the International Conference on Mathematics, Computational Methods & Reactor Physics (M&C 2009)*, Saratoga Springs, New York, May 3–7, 2009.

[3d5] C. M. Perfetti, B. T. Rearden, "Continuous-Energy Monte Carlo Methods for Calculating Generalized Response Sensitivities using TSUNAMI-3D," *Proceedings of the 2014 International Conference on the Physics of Reactors (PHYSOR 2014),* Kyoto, Japan, September 28–October 3, 2014.

[3d6] C. M. Perfetti, B. T. Rearden, "Performance Enhancements to the SCALE TSUNAMI-3D Generalized Response Sensitivity Capability," *Trans. Am. Nucl. Soc.* **112** (2014).

[3d7] M. L. Williams, "Perturbation Theory for Nuclear Reactor Analysis," in Y. Ronen, *CRC Handbook of Nuclear Reactor Calculations: Volume III*, Boca Raton, Florida: CRC Press, Inc. (1986).

[3d8] M. L. Williams, "Generalized Contributon Response Theory," *Nucl. Sci. & Engr.*, **108**, pp. 355–383 (1991).

[3d9] M. L. Williams, *Equations for Contributon Eigenvalue Solution*. Unpublished document (2007).

[3d10] C. M. Perfetti, B. T. Rearden, "Metrics for Diagnosing Undersampling in Monte Carlo Tally Estimates," *Proceedings of M&C 2015*, Nashville, Tennessee, April 19–23, 2015

[3d11] *International Handbook of Evaluated Criticality Safety Benchmark Experiments*, Nuclear Energy Agency Nuclear Science Committee of the Organization for Economic Co-operation and Development, NEA/NSC/DOC(95)03 (2010). Available at: http://icsbep.inl.gov/handbook.shtml

[refTsip1] B. L. Broadhead, B. T. Rearden, C. M. Hopper, J. J. Wagschal, and C. V. Parks, "Sensitivity- and Uncertainty-Based Criticality Safety Validation Techniques," Nucl. Sci. Eng., 146, 340–366 (2004).

[refTsip2] S. Golouglu, C. M. Hopper, and B. T. Rearden, "Extended Interpretation of Sensitivity Data for Benchmark Areas of Applicability," Trans. Am. Nuc. Soc., 88, 77–79 (2003).

[refTsip3] S. Goluoglu, K. R. Elam, B. T. Rearden, B. L. Broadhead, and C. M. Hopper, Sensitivity Analysis Applied to the Validation of the 10B Capture Reaction in Nuclear Fuel Casks, NUREG/CR-6845 (ORNL/TM-2004/48) U.S. Nuclear Regulatory Commission, Oak Ridge National Laboratory, August 2004.

[refTsip4] B. L. Broadhead, C. M. Hopper, K. R. Elam, B. T. Rearden, and R. L. Childs, "Criticality Safety Applications of S/U Validation Methods," Trans. Am. Nucl. Soc. 83, 107–113 (2000).

[refTsip5] M. E. Dunn, PUFF-III: A Code for Processing ENDF Uncertainty Data Into Multigroup Covariance Matrices, ORNL/TM-1999/235 (NUREG/CR-6650), U.S. Nuclear Regulatory Commission, Oak Ridge National Laboratory, June 2000.

[refTsip6] J. J. Lichtenwalter, S. M. Bowman, M. D. DeHart, and C. M. Hopper, Criticality Benchmark Guide for Light-Water-Reactor Fuel in Transportation and Storage Packages, NUREG/CR-6361 (ORNL/TM-13211), U.S. Nuclear Regulatory Commission, Oak Ridge National Laboratory, March 1997.

[SAMPLER-BWAW22] Friederike Bostelmann, Dorothea Wiarda, Goran Arbanas, and William A. Wieselquist. Extension of SCALE/Sampler's sensitivity analysis. *Annals of Nuclear Energy*, 165:108641, 2022. doi:10.1016/j.anucene.2021.108641.

[SAMPLER-BHNS10] O. Buss, A. Hoefer, J. C. Neuber, and M. Schmid. Hierarchical Monte-Carlo Approach to Bias Estimation for Criticality Safety Calculations . In *Proceedings of PHYSOR 2010, Advances in Reactor Physics to Power the Nuclear Renaissance*. Pittsburgh, PA (USA), 5 2010.

[SAMPLER-HMAK20] Dongli Huang, Ugur Mertyurek, and Hany Abdel-Khalik. Verification of the sensitivity and uncertainty-based criticality safety validation techniques: ornl's scale case study. *Nuclear Engineering and Design*, 361:110571, 2020. doi:https://doi.org/10.1016/j.nucengdes.2020.110571.

[SAMPLER-KHK94] B. Krzykacz, E. Hofer, and M. Kloos. A Software System for Probabilistic Uncertainty and Sensitivity Analysis of Results from Computer Models. In *Proc. International Conference on Probabilistic Safety Assessment and Management (PSAM-II)*. San Diego, CA (USA), 1994.

[SAMPLER-PFG15] M. T. Pigni, M. W. Francis, and I. C. Gauld. Investigation of Inconsistent ENDF/B-VII.1 Independent and Cumulative Fission Product Yields with Proposed Revisions. *Nuclear Data Sheets*, 123:231–236, 2015. doi:https://doi.org/10.1016/j.nds.2014.12.040.

[SAMPLER-RWJ+11] B. T. Rearden, M. L. Williams, M. A. Jessee, D. E. Mueller, and D. A. Wiarda. Sensitivity and Uncertainty Analysis Capabilities and Data in SCALE. *Nuclear Technology*, 2(174):236–288, 2011.

[SAMPLER-WIJ+13] M. L. Williams, G. Ilas, M. A. Jessee, B. T. Rearden, D. Wiarda, W. Zwermann, L. Gallner, M. Klein, B. Krzykacz-Hausmann, and A. Pautz. A Statistical Sampling Method for Uncertainty Analysis with SCALE and XSUSA. *Nuclear Technology*, pages 515–526, 2013.

[SAMSBev69] P. Bevington. *R.(1969) Data Reduction and Error Analysis for the Physical Sciences*. New York: McGraw-Hill, 1969.

[SAMSCom99] NEA Nuclear Science Committe. *International Handbook of Evaluated Criticality Safety Benchmarks Experiments*. OECD Nuclear Energy Agency, 1999.

[SAMSDun00] M. E. Dunn. PUFF-III: A Code for Processing ENDF Uncertainty Data into Multigroup Covariance Matrices. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2000.

[SAMSGan67] A. Gandini. A generalized perturbation method for bi-linear functionals of the real and adjoint neutron fluxes. *Journal of Nuclear Energy*, 21(10):755–765, 1967. Publisher: Pergamon.

[SAMSHTK84] Akihiro Hara, Toshikazu Takeda, and Yasuyuki Kikuchi. SAGEP: two-dimensional sensitivity analysis code based on generalized perturbation theory. Technical Report, Japan Atomic Energy Research Inst., 1984.

[SAMSLWM+81] J. L. Lucius, C. R. Weisbin, J. H. Marable, J. D. Drischler, R. Q. Wright, and J. E. White. Users manual for the FORSS sensitivity and uncertainty analysis code system. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1981.

[SAMSObl76] E. M. Oblow. Sensitivity theory from a differential viewpoint. *Nuclear Science and Engineering*, 59(2):187–189, 1976. Publisher: Taylor & Francis.

[SAMSSJ72] Weston M. Stacey Jr. Variational estimates and generalized perturbation theory for the ratios of linear and bilinear functionals. *Journal of Mathematical Physics*, 13(8):1119–1125, 1972. Publisher: American Institute of Physics.

[SAMSUsa64] L. N. Usachev. Perturbation theory for the breeding ratio and for other number ratios pertaining to various reactor processes. *Journal of Nuclear Energy. Parts A/B. Reactor Science and Technology*, 18(10):571–583, 1964. Publisher: Elsevier.

[SAMSWML+76] C. R. Weisbin, J. H. Marable, J. L. Lucius, E. M. Oblow, F. R. Mynatt, R. W. Peelle, and F. G. Perey. Application of FORSS sensitivity and uncertainty methodology to fast reactor benchmark analysis. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1976.

[SAMSWOM+78] C. R. Weisbin, E. M. Oblow, J. H. Marable, R. W. Peelle, and J. L. Lucius. Application of sensitivity and uncertainty methodology to fast reactor integral experiment analysis. *Nuclear Science and Engineering*, 66(3):307–333, 1978. Publisher: Taylor & Francis.

[SAMSWBP01] M. L. Williams, B. L. Broadhead, and C. V. Parks. Eigenvalue sensitivity theory for resonance-shielded cross sections. *Nuclear Science and Engineering*, 138(2):177–191, 2001.

[SAMSWil86] Mark L. Williams. *Perturbation theory for nuclear reactor analysis*. CRC Press, Inc., 1986.

[SAMSNationalNDCenter74] National Nuclear Data Center. *Cross Section Evaluation Working Group Benchmark Specifications*. Brookhaven National Laboratory, 1974.

[TSARWGC06] Mark L. Williams, Jess C. Gehin, and Kevin T. Clarno. Sensitivity analysis of reactivity responses using one-dimensional discrete ordinates and three-dimensional Monte Carlo methods. In *C135. pdf in Proc. of PHYSOR–2006, American Nuclear Society Topical Meeting on Reactor Physics: Advances in Nuclear Analysis and Simulation*. 2006.

[TSARWR08] Mark L. Williams and Bradley T. Rearden. SCALE-6 Sensitivity/uncertainty methods and covariance data. *Nuclear Data Sheets*, 109(12):2796–2800, 2008.

[TSURFER-Bri06] J. Blair Briggs. International handbook of evaluated criticality safety benchmark experiments. *Nuclear Energy Agency, NEA/NSC/DOC (95)*, 3:l, 2006.

[TSURFER-BHCP99] B. L. Broadhead, C. M. Hopper, R. L. Childs, and C. V. Parks. Sensitivity and uncertainty analyses applied to criticality safety validation, methods development. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1999.

[TSURFER-BHP99] B. L. Broadhead, C. M. Hopper, and C. V. Parks. Sensitivity and uncertainty analyses applied to criticality safety validation. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1999.

[TSURFER-BRH+04] B. L. Broadhead, B. T. Rearden, C. M. Hopper, J. J. Wagschal, and C. V. Parks. Sensitivity-and uncertainty-based criticality safety validation techniques. *Nuclear science and engineering*, 146(3):340–366, 2004.

[TSURFER-Gol04] Sedat Goluoglu. Sensitivity Analysis Applied to the Validation of the 10 B Capture Reaction in Nuclear Fuel Casks. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2004.

[TSURFER-Hwa88] R. N. Hwang. Topics in data adjustment theory and applications. In *Proceedings of the Specialists' Meeting on the Application of Critical Experiments and Operating Data to Core Design via Formal Methods of Cross Section Data Adjustment*. Jackson Hole, WY (USA), 9 1988.

[TSURFER-JRCMPR16] E. Juárez Ruiz, R. Cortes Maldonado, and F. Pérez Rodríguez. Relationship between the Inverses of a Matrix and a Submatrix. *Computación y Sistemas*, 20(2):251–262, July 2016. URL: http://www.cys.cic.ipn.mx/ojs/index.php/CyS/article/view/2083 (visited on 2020-08-10), doi:10.13053/cys-20-2-2083.

[TSURFER-LKH+08] R. C. Little, T. Kawano, G. D. Hale, M. T. Pigni, M. Herman, P. Obložinsky, M. L. Williams, M. E. Dunn, G. Arbanas, and D. Wiarda. Low-fidelity Covariance Project. *Nuclear Data Sheets*, 109(12):2828–2833, 2008. Publisher: Elsevier.

[TSURFER-MWB81] R. E. Maerker, J. J. Wagschal, and B. L. Broadhead. Development and demonstration of an advanced methodology for LWR dosimetry applications. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1981.

[TSURFER-PRR+74] A. Pazy, G. Rakavy, I. Reiss, J. J. Wagschal, Atara Ya'ari, and Y. Yeivin. The role of integral data in neutron cross-section evaluation. *Nuclear Science and Engineering*, 55(3):280–295, 1974. Publisher: Taylor & Francis.

[TSURFER-PC88] W. P. Poenitz and P. J. Collins. Utilization of experimental integral data for the adjustment and uncertainty evaluation of reactor design quantities. In *Proceedings of the Specialists' Meeting on the Application of Critical Experiments and Operating Data to Core Design via Formal Methods of Cross Section Data Adjustment*. Jackson Hole, WY (USA), 9 1988.

[TSURFER-TTYS88] T. Takeda, M. Takamoto, A. Yoshimura, and K. Shirakata. Adjustment of JENDL-2 Cross Sections and Prediction Accuracy for FBR Core Parameters Using Jupiter Integral Data. In *Proceedings of the Specialists' Meeting on the Application of Critical Experiments and Operating Data to Core Design via Formal Methods of Cross Section Data Adjustment*. Jackson Hole, WY (USA), 9 1988.

[TSURFER-WML+76] C. R. Weisbin, J. H. Marable, J. L. Lucius, E. M. Oblow, F. R. Mynatt, R. W. Peelle, and F. G. Perey. Application of FORSS sensitivity and uncertainty methodology to fast reactor benchmark analysis. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1976.

[TSURFER-WBP01] M. L. Williams, B. L. Broadhead, and C. V. Parks. Eigenvalue sensitivity theory for resonance-shielded cross sections. *Nuclear Science and Engineering*, 138(2):177–191, 2001.

[TSURFER-Wil86] Mark L. Williams. *Perturbation theory for nuclear reactor analysis*. CRC Press, Inc., 1986.

[TSURFER-WR08] Mark L. Williams and Bradley T. Rearden. SCALE-6 Sensitivity/uncertainty methods and covariance data. *Nuclear Data Sheets*, 109(12):2796–2800, 2008.

[TSURFER-YWMW80] Y. Yeivin, J. J. Wagschal, J. H. Marable, and C. R. Weisbin. Relative Consistency of ENDF/B-IV and-V with Fast Reactor Benchmarks. In *Nuclear Cross Sections for Technology: Proceedings of the International Conference on Nuclear Cross Sections for Technology, Held at the University of Tennessee, Knoxville, TN, October 22-26, 1979*, volume 594, 182. US Department of Commerce, National Bureau of Standards, 10 1980.

[SU-BRH+04] B. L. Broadhead, B. T. Rearden, C. M. Hopper, J. J. Wagschal, and C. V. Parks. Sensitivity- and uncertainty-based criticality safety validation techniques. *Nuclear science and engineering*, 146(3):340–366, 2004.

[SU-LBDH97] J. J. Lichtenwalter, S. M. Bowman, M. D. DeHart, and C. M. Hopper. Criticality Benchmark Guide for Light-Water-Reactor Fuel in Transportation and Storage Packages. Technical Report, Nuclear Regulatory Commission, 1997.

[SU-MR15] W. J. Marshall and B. T. Rearden. Determination of Experimental Correlations Using the Sampler Sequence within SCALE 6.2. In *ICNC 2015*. Charlotte, NC (USA), 2015.

[SU-RMB+09] B. T. Rearden, D. E. Mueller, S. M. Bowman, R. D. Busch, and S. J. Emerson. TSUNAMI Primer: A Primer for Sensitivity/Uncertainty Calculations with SCALE. Technical Report ORNL/TM-2009/027, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2009.

[SU-RWJ+11] B. T. Rearden, M. L. Williams, M. A. Jessee, D. E. Mueller, and D. A. Wiarda. Sensitivity and Uncertainty Analysis Capabilities and Data in SCALE. *Nuclear Technology*, 2(174):236–288, 2011.

[SU-WIJ+13] M. L. Williams, G. Ilas, M. A. Jessee, B. T. Rearden, D. Wiarda, W. Zwermann, L. Gallner, M. Klein, B. Krzykacz-Hausmann, and A. Pautz. A Statistical Sampling Method for Uncertainty Analysis with SCALE and XSUSA. *Nuclear Technology*, pages 515–526, 2013.

[SU-AmericanNSSCWGA8198] American Nuclear Society Standards Committee Working Group ANS-8.1. *American National Standard for Nuclear Criticality Safety in Operations with Fissionable Materials outside Reactors*. Number ANSI/ANS-8.1-1998. American Nuclear Society, 1998.

[SU-AmericanNSSCWGA82407] American Nuclear Society Standards Committee Working Group ANS-8.24. *American National Standard for Validation of Neutron Transport Methods for Nuclear Criticality Safety Calculations*. Number ANSI/ANS-8.24-2007. American Nuclear Society, 2007.

[XSProc-IB64] Igor Ilich Bondarenko. *Group constants for nuclear reactor calculations*. Consultants Bureau, 1964.

[XSProc-RLL+15] Bradley T. Rearden, Robert A. Lefebvre, Jordan P. Lefebvre, Kevin T. Clarno, Mark A. Williams, Lester M. Petrie, and Ugur Mertyurek. Modernization enhancements in SCALE 6.2. In *PHYSOR 2014*. Kyoto, Japan, 2015.

[XSProc-Wil11] Mark L. Williams. Resonance self-shielding methodologies in SCALE 6. *Nuclear Technology*, 174(2):149–168, May 2011. URL: https://doi.org/10.13182/NT09-104, doi:10.13182/NT09-104.

[XSProc-WA95] Mark L. Williams and Mehdi Asgari. Computation of continuous-energy neutron spectra with discrete ordinates transport theory. *Nuclear Science and Engineering*, 121(2):173–201, 1995. Publisher: Taylor & Francis.

[STDCOMPAWT03] Georges Audi, A. H. Wapstra, and C. Thibault. The AME2003 atomic mass evaluation:(II). Tables, graphs and references. *Nuclear physics A*, 729(1):337–676, 2003. Publisher: Elsevier.

[STDCOMPBDLDB+05] J. K. Böhlke, J. R. De Laeter, P. De Bievre, H. Hidaka, H. S. Peiser, K. J. R. Rosman, and P. D. P. Taylor. Isotopic compositions of the elements, 2001. *Journal of Physical and Chemical Reference Data*, 34(1):57–67, 2005. Publisher: American Institute of Physics for the National Institute of Standards and ....

[STDCOMPWea01] Robert C. Weast, editor. *CRC Handbook of Chemistry and Physics 64th Ed. 1983-1984*. CRC, Boca Raton, Fla, 4th edition edition, October 2001. ISBN 978-0-8493-0464-4.

[BONAMIDYB77] W. J. Davis, M. B. Yarbrough, and A. B. Bortz. SPHINX, a one-dimensional diffusion and transport nuclear cross section processing code. Technical Report Westinghouse Advanced Reactors Division report WARD-XS-3045-17, Westinghouse, 8 1977.

[BONAMIGC62] Rubin Goldstein and E. Richard Cohen. Theory of resonance absorption of neutrons. *Nuclear Science and Engineering*, 13(2):132–140, 1962. Publisher: Taylor & Francis.

[BONAMIGre82] N. M. Greene. Method for interpolating in Bondarenko factor tables and other functions. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1982.

[BONAMIIB64] Igor Ilich Bondarenko. *Group constants for nuclear reactor calculations*. Consultants Bureau, 1964.

[BONAMIKid74] R. B. Kidman. Improved F-factor interpolation scheme for 1DX. In *Transactions of the American Nuclear Society*, volume 18. 1974.

[BONAMIKHW21] Kang Seog Kim, Andrew M. Holcomb, and William A. Wieselquist. Spatially Dependent Resonance Self-Shielding Capability for Non-uniform Temperature Profile in SCALE-6.3 XSProc-BONAMI. In *M&C 2021*. 2021.

[BONAMILam66] John R. Lamarsh. *Introduction to nuclear reactor theory*. Volume 3. Addison-Wesley Reading, Massachusetts, 1966.

[BONAMILHJ65] D. C. Leslie, J. G. Hill, and A. Jonsson. Improvements to the Theory of Resonance Escape in Heterogeneous Fuel: I. Regular Arrays of Fuel Rods. *Nuclear Science and Engineering*, 22(1):78–86, 1965. Publisher: Taylor & Francis.

[BONAMIOtt64] John M. Otter. Escape Probability Approximations in Lumped Resonance Absorbers. Technical Report, Atomics International. Div. of North American Aviation, Inc., Canoga Park ..., 1964.

[BONAMISeg81] M. Segev. Interpolation of resonance integrals. *Nuclear Science and Engineering*, 79(1):113–118, 1981. Publisher: Taylor & Francis.

[BONAMIWWCD15] Dorothea Wiarda, Mark L. Williams, Cihangir Celik, and Michael E. Dunn. AMPX: A Modern Cross Section Processing System for Generating Nuclear Data Libraries. Technical Report, Oak Ridge National Laboratory, Charlotte, NC (USA), 9 2015.

[CENTRM-BG70] George I. Bell and Samuel Glasstone. Nuclear reactor theory. Technical Report, US Atomic Energy Commission, Washington, DC (United States), 1970.

[CENTRM-Car64] Ingvar Carlvik. A method for calculating collision probabilities in general cylindrical geometry and applications to flux distributions and Dancoff factors. Technical Report, Aktiebolaget Atomenergi, Stockholm (Sweden), 1964.

[CENTRM-HF71] H. C. Honeck and D. R. Finch. FLANGE-II, A Code to Process Thermal Neutron Data from an ENDF/B Tape. Technical Report, Savannah River Laboratory, 1971.

[CENTRM-KW12] Kang Seog Kim and Mark L. Williams. The Method of Characteristics for 2-D Multigroup and Pointwise Transport Calculations in SCALE/CENTRM. In *PHYSOR 2012*. 2012.

[CENTRM-Rob81] G. S. Robinson. Notes on the AAEC Version of XLACS-II. *Informal Memo (9/3/81)*, 1981.

[CENTRM-Wil00] Mark L. Williams. Submoment expansion of neutron-scattering sources. *Nuclear science and engineering*, 134(2):218–226, 2000. Publisher: Taylor & Francis.

[CENTRM-WA95] Mark L. Williams and Mehdi Asgari. Computation of continuous-energy neutron spectra with discrete ordinates transport theory. *Nuclear Science and Engineering*, 121(2):173–201, 1995. Publisher: Taylor & Francis.

[PMCBG70] George I. Bell and Samuel Glasstone. Nuclear reactor theory. Technical Report, US Atomic Energy Commission, Washington, DC (United States), 1970.

[PMCBuc78] J.A. Bucholz. A Method of Generating Multigroup Transfer Matrices Using an Analytical Angular Integration Free of Truncation Error. Technical Report ORNL/NUREG/CSD-8, Union Carbide Corporation (Nuclear Division), Oak Ridge National Laboratory, Oak Ridge, TN (USA), 9 1978.

[PMCWA95] Mark L. Williams and Mehdi Asgari. Computation of continuous-energy neutron spectra with discrete ordinates transport theory. *Nuclear Science and Engineering*, 121(2):173–201, 1995. Publisher: Taylor & Francis.

[CRAWDADHCML16] Shane WD Hart, Cihangir Celik, G. Ivan Maldonado, and Luiz Leal. Creation of problem-dependent Doppler-broadened cross sections in the KENO Monte Carlo code. *Annals of Nuclear Energy*, 88:49–56, 2016.

[MAT-GW05] Sedat Goluoglu and Mark L. Williams. Modeling doubly heterogeneous systems in scale. In *Transactions of the American Nuclear Society*, volume 93, 963. 2005.

[MAT-IB64] Igor Ilich Bondarenko. *Group constants for nuclear reactor calculations*. Consultants Bureau, 1964.

[MAT-StammlerA83] Rudi JJ Stamm'ler and Máximo Julio Abbate. *Methods of steady-state reactor physics in nuclear design*. Volume 111. Academic Press London, 1983.

[MAT-WWCD15] Dorothea Wiarda, Mark L. Williams, Cihangir Celik, and Michael E. Dunn. AMPX: A Modern Cross Section Processing System for Generating Nuclear Data Libraries. Technical Report, Oak Ridge National Laboratory, Charlotte, NC (USA), 9 2015.

[MAT-Wil11] Mark L. Williams. Resonance self-shielding methodologies in SCALE 6. *Nuclear Technology*, 174(2):149–168, May 2011. URL: https://doi.org/10.13182/NT09-104, doi:10.13182/NT09-104.

[MAT-WA95] Mark L. Williams and Mehdi Asgari. Computation of continuous-energy neutron spectra with discrete ordinates transport theory. *Nuclear Science and Engineering*, 121(2):173–201, 1995. Publisher: Taylor & Francis.

[KENO-Appendix-CBKH+77] Clco C. Byers, Jerry J. Koelling, Gordon E. Hansen, David R. Smith, and Howard R. Dyer. Critical measurements of a water-reflected enriched uranium sphere. In *Transactions of the American Nuclear Society*, volume 27. 1977.

[KENO-Appendix-CIM64] D. C. Irving and John T. Mihalczo. Monte Carlo Calculations for Enriched Uranium Metal Assemblies. In *Transactions of the American Nuclear Society*, volume 7. 1964.

[KENO-Appendix-CJoh66] E. B. Johnson. Critical Parameters of Uranium (1.95) Metal Cylindrical Annuli. In *Transactions of the American Nuclear Society*, volume 9. 1966.

[KENO-Appendix-CMLTH93] J. T. Mihalczo, J. J. Lynn, J. R. Taylor, and G. E. Hansen. Measurements with an unreflected uranium (93.2%) metal sphere. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1993.

[KENO-Appendix-CTho64] J. T. Thomas. CRITICAL THREE-DIMENSIONAL ARRAYS OF NEUTRON-INTERACTING UNITS. PART II. U (93.2) METAL. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1964.

[KENO-Appendix-CTho73] Joseph T. Thomas. Critical three-dimensional arrays of U (93.2)-metal cylinders. *Nuclear Science and Engineering*, 52(3):350–359, 1973. Publisher: Taylor & Francis.

[KENO-Appendix-CWT69] G. E. Whitesides and J. T. Thomas. USE OF DIFFERENTIAL CURRENT ALBEDOS IN MONTE CARLO CRITICALITY CALCULATIONS. Technical Report, Union Carbide Corp., Oak Ridge, Tenn., 1969.

[BDL09] B. Becker, R. Dagan, and G. Lohnert. Proof and implementation of the stochastic formula for ideal gas, energy dependent scattering kernel. *Annals of Nuclear Energy*, 36(4):470–474, 2009. Publisher: Elsevier.

[Dem99a] L. Demaret. Accurate determination of confidence intervals in Monte Carlo eigenvalue calculations. In *ICNC'99, Sixth Int. Conf. on Nuclear Criticality Safety, Versailles, France, Sep. 20-24. 1999*, volume 1, 66–80. 1999.

[Dem99b] L. Demaret. Accurate determination of confidence intervals in Monte Carlo eigenvalue calculations. In *ICNC'99, Sixth Int. Conf. on Nuclear Criticality Safety, Versailles, France, Sep. 20-24. 1999*, volume 1, 66–80. 1999.

[DG02] M. E. Dunn and N. M. Greene. AMPX-2000: A Cross-Section Processing System for Generating Nuclear Data for Criticality Safety Applications. In *Transactions of the American Nuclear Society*, volume 86, 118–119. 2002.

[Hof82] T. J. Hoffman. The Optimization of Russian Roulette Parameters for KENO Va. Technical Report ORNL/TM-7539, Union Carbide Corp., Nucl. Div., Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1982.

[ODellA87] R. Douglas O'Dell and Raymond E. Alcouffe. Transport calculations for nuclear analyses: Theory and guidelines for effective use of transport codes. Technical Report, Los Alamos National Laboratory, Los Alamos, NM (USA), 1987.

[Tho64] J. T. Thomas. CRITICAL THREE-DIMENSIONAL ARRAYS OF NEUTRON-INTERACTING UNITS. PART II. U (93.2) METAL. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1964.

[Tho73] Joseph T. Thomas. Critical three-dimensional arrays of U (93.2)-metal cylinders. *Nuclear Science and Engineering*, 52(3):350–359, 1973. Publisher: Taylor & Francis.

[Uek05] Taro Ueki. Information theory and undersampling diagnostics for Monte Carlo simulation of nuclear criticality. *Nuclear Science and Engineering*, 151(3):283–292, 2005. Publisher: Taylor & Francis.

[Whi71] G. E. Whitesides. A Difficulty in Computing the k-effective of the World. In *Transactions of the American Nuclear Society*, volume 14, 680. 1971.

[MONACO-FS81] F. H. Froehner and R. R. Spencer. Method for sampling from fission neutron energy spectra. Technical Report ORNL/TM-7631, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1981.

[MONACO-JTN+97] B. Jansky, Z. Turzík, E. Novák, J. Kyncl, F. Cvachovec, and P. Tiller. Comparison of measured and calculated neutron transmission through heavy water for 252Cf source placed in the center of 30 cm diameter sphere. *Annals of Nuclear Energy*, 24(15):1189–1212, 1997.

[MONACO-KI11] Brian C. Kiedrowski and Ahmad Ibrahim. Evaluating the efficiency of estimating numerous Monte Carlo tallies. Technical Report, Los Alamos National Laboratory, Los Alamos, NM (USA), 2011.

[MONACO-KS11] Brian C. Kiedrowski and Clell J. Solomon. Statistical assessment of numerous Monte Carlo tallies. Technical Report, Los Alamos National Laboratory, Los Alamos, NM (USA), 2011.

[MONACO-PFB97] S. P. Pederson, R. A. Forster, and T. E. Booth. Confidence interval procedures for Monte Carlo transport simulations. *Nuclear science and engineering*, 127(1):54–77, 1997. Publisher: Taylor & Francis.

[MONACO-SWA93] E. Sajo, M. L. Williams, and M. Asgari. Comparison of Measured and Calculated Neutron Transmission through Steel for a 252Cf source. *Annals of Nuclear Energy*, 20(9):585–604, 1993.

[MONACO-UOA92] K. Ueki, A. Ohashi, and Y. Anayama. Neutron Shielding Ability of KRAFTON N2 - Mannan - KRAFTON N2 Sandwich-type Material and Others. In *New Horizons in Radiation Protection and Shielding*, 130–137. Pasco, WA (USA), April 1992.

[Monte-CarloGPJD+11] Sedat Goluoglu, Lester M. Petrie Jr, Michael E. Dunn, Daniel F. Hollenbach, and Bradley T. Rearden. Monte Carlo criticality methods and analysis capabilities in SCALE. *Nuclear Technology*, 174(2):214–235, 2011.

[Monte-CarloHMGR13] S. Hart, G. Ivan Maldonado, Sedat Goluoglu, and Brad Rearden. Implementation of the Doppler broadening rejection correction in KENO. In *Transactions of the American Nuclear Society*, volume 108, 423–426. 2013.

[Monte-CarloPep11] Douglas E. Peplow. Monte Carlo shielding analysis capabilities with MAVRIC. *Nuclear Technology*, 174(2):289–313, 2011. Publisher: Taylor & Francis.

[Monte-CarloRPP+14] Bradley T. Rearden, L. M. Petrie, Douglas E. Peplow, Kursat B. Bekar, Dorothea Wiarda, Cihangir Celik, Christopher M. Perfetti, Ahmad M. Ibrahim, S. W. D. Hart, and Michael E. Dunn. Monte Carlo capabilities of the SCALE code system. In *SNA+ MC 2013-Joint International Conference on Supercomputing in Nuclear Applications+ Monte Carlo*, 06007. EDP Sciences, 2014.

[Monte-CarloUB05] Taro Ueki and Forrest B. Brown. Stationarity modeling and informatics-based diagnostics in Monte Carlo criticality calculations. *Nuclear Science and Engineering*, 149(1):38–50, 2005. Publisher: Taylor & Francis.

[Monte-CarloUMN97] Taro Ueki, Takamasa Mori, and Masayuki Nakagawa. Error estimations and their biases in Monte Carlo eigenvalue calculations. *Nuclear science and engineering*, 125(1):1–11, 1997. Publisher: Taylor & Francis.

[XSDAld63] Berni Alder. *Methods in Computational Physics. V. 1*. Academic Press, 1963.

[XSDCL65] B. G. Carlson and K. D. Lathrop. Discrete ordinates angular quadrature of the neutron transport equation. Technical Report LA-3186, Los Alamos National Laboratory, Los Alamos, NM (USA), 1965.

[XSDCar74] B. M. Carmichael. Standard interface files and procedures for reactor physics codes, version III. Technical Report, Los Alamos Scientific Laboratory, Los Alamos, NM (USA), 1974.

[XSDEme75] B. B. Emett. The MORSE Monte Carlo Radiation Transport Code. Technical Report ORNL-4972, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1975.

[XSDEJ67] Ward W. Engle Jr. A User's Manual for ANISN: A One Dimensional Discrete Ordinates Code with Anisotropic Scattering. Technical Report, Oak Ridge Gaseous Diffusion Plant, Oak Ridge, TN (USA), 3 1967. Available from Radiation Shielding Information Center as CCC-82/ANISN.

[XSDGCJ69] N. M. Greene and C. W. Craven Jr. XSDRN: A DISCRETE ORDINATES SPECTRAL AVERAGING CODE. Technical Report, Union Carbide Corp. / Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1969.

[XSDHTS76] H. Henryson, B. J. Toppel, and C. G. Stenberg. MC 2-2: a code to calculate fast neutron spectra and multigroup cross sections. Technical Report, Argonne National Laboratory, 1976.

[XSDHof82] T. J. Hoffman. The Optimization of Russian Roulette Parameters for KENO Va. Technical Report ORNL/TM-7539, Union Carbide Corp., Nucl. Div., Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1982.

[XSDHT82] T. J. Hoffman and J. S. Tang. XSDRNPM-S biasing of MORSE-SGC/S shipping-cask calculations. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1982.

[XSDHon61] Henry C. Honeck. THERMOS. A Thermalization Transport Theory Code for Reactor Lattice Calculations. Technical Report, Brookhaven National Laboratory, Upton, NY, 1961.

[XSDJD63] G. D. Joanou and J. S. Dudek. GAM-II. AB \$ sub 3\$ CODE FOR THE CALCULATION OF FAST-NEUTRON SPECTRA AND ASSOCIATED MULTIGROUP CONSTANTS. Technical Report, General Atomic Div., General Dynamics Corp., San Diego, Calif., 1963.

[XSDOls65] Tom Olsen. Void Streaming in Sn Calculations. *Nuclear Science and Engineering (US)*, 1965.

[XSDREJ77] W. A. Rhoades and W. W. Engle Jr. New weighted-difference formulation for discrete-ordinates calculations. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1977.

[XSDRSCEJ79] W. A. Rhoades, D. B. Simpson, R. L. Childs, and W. W. Engle Jr. DOT-IV two-dimensional discrete ordinates transport code with space-dependent mesh and quadrature. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1979.

[XSDTREJ80] E. T. Tomlinson, W. A. Rhoades, and W. W. Engle Jr. Flux extrapolation models used in the DOT IV discrete ordinates neutron transport code. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1980.

[XSDWRHB79] C. R. Weisbin, R. W. Roussin, H. R. Hendrickson, and E. W. Bryant. Review of the theory and application of sensitivity and uncertainty analysis. Proceedings of a seminar-workshop, Oak Ridge, Tennessee. Technical Report, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 1979.

[XSDWWCD15] D. Wiarda, M. L. Williams, C. Celik, and M. E. Dunn. AMPX-2000: A cross-section processing system for generating nuclear data for criticality safety applications. In *Proc. Int. Conf. Nuclear Criticality Safety (ICNC 2015), Charlotte, North Carolina*. 2015.

[XSDWil86] Mark L. Williams. *Perturbation theory for nuclear reactor analysis*. CRC Press, Inc., 1986.

[NEWTAL81] R. E. Alcouffe and E. W. Larsen. Review of characteristic methods used to solve the linear transport equation. Technical Report, Los Alamos Scientific Laboratory, Los Alamos, NM (USA), 1981.

[NEWTALMJW79] R. E. Alcouffe, E. W. Larsen, W. F. Miller Jr, and B. R. Wienke. Computational efficiency of numerical methods for the multigroup, discrete-ordinates neutron transport equations: the slab geometry case. *Nuclear Science and Engineering*, 71(2):111–127, 1979. Publisher: Taylor & Francis.

[NEWTCL65] B. G. Carlson and K. D. Lathrop. Discrete ordinates angular quadrature of the neutron transport equation. Technical Report LA-3186, Los Alamos National Laboratory, Los Alamos, NM (USA), 1965.

[NEWTCar70] Bengt G. Carlson. TRANSPORT THEORY: DISCRETE ORDINATES QUADRATURE OVER THE UNIT SPHERE. Technical Report, Los Alamos Scientific Laboratory, Los Alamos, NM (USA), 1970.

[NEWTDeh92] Mark David Dehart. A discrete ordinates approximation to the neutron transport equation applied to generalized geometries. Technical Report, Westinghouse Savannah River Co., Aiken, SC (United States), 1992.

[NEWTDH76] James J. Duderstadt and Louis J. Hamilton. *Nuclear Reactor Analysis* . Wiley, 1976. ISBN 978-0471223634.

[NEWTEJ67] Ward W. Engle Jr. A User's Manual for ANISN: A One Dimensional Discrete Ordinates Code with Anisotropic Scattering. Technical Report, Oak Ridge Gaseous Diffusion Plant, Oak Ridge, TN (USA), 3 1967. Available from Radiation Shielding Information Center as CCC-82/ANISN.

[NEWTHil76] F. B. Hildebrand. *Advanced Calculus for Applications*. Volume 511. Prentice-Hall, Inc., 1976.

[NEWTLA81] E. W. Larsen and R. E. Alcouffe. Linear characteristic method for spatially discretizing the discrete ordinates equations in (X, Y)-geometry. Technical Report, Los Alamos Scientific Lab., NM (USA), 1981.

[NEWTLat68] K. D. Lathrop. Spatial differencing of the two-dimensional transport equation. *Report GA-8746, Gulf General Atomic*, 1968.

[NEWTLat69] K. D. Lathrop. Spatial differencing of the transport equation: positivity vs. accuracy. *Journal of computational physics*, 4(4):475–498, 1969. Publisher: Elsevier.

[NEWTLB73] K. D. Lathrop and F. W. Brinkley. TWOTRAN-II: An interfaced, exportable version of the TWOTRAN code for two-dimensional transport. Technical Report, Los Alamos Scientific Lab., N. Mex.(USA), 1973.

[NEWTLee62] Clarence E. Lee. *The discrete Sn approximation to transport theory*. Volume 2595. Los Alamos Scientific Laboratory of the University of California, 1962.

[NEWTLM84] E. E. Lewis and W. F. Miller. *"Computational Methods of Neutron Transport"*. John Wiley & Sons, Inc., New York, 1984.

[NEWTZDX+08] Zhaopeng Zhong, Thomas J. Downar, Yunlin Xu, Mark D. DeHart, and Kevin T. Clarno. Implementation of two-level coarse-mesh finite difference acceleration in an arbitrary geometry, two-dimensional discrete ordinates transport method. *Nuclear science and engineering*, 158(3):289–298, 2008. Publisher: Taylor & Francis.

[DTESSC10] Thomas M. Evans, Alissa S. Stafford, Rachel N. Slaybaugh, and Kevin T. Clarno. Denovo: A new three-dimensional parallel discrete ordinates code in SCALE. *Nuclear technology*, 171(2):171–200, 2010.

[XSLibCHO+11] M.B. Chadwick, M. Herman, P. Obložinský, M.E. Dunn, Y. Danon, A.C. Kahler, D.L. Smith, B. Pritychenko, G. Arbanas, R. Arcilla, R. Brewer, D.A. Brown, R. Capote, A.D. Carlson, Y.S. Cho, H. Derrien, K. Guber, G.M. Hale, S. Hoblit, S. Holloway, T.D. Johnson, T. Kawano, B.C. Kiedrowski, H. Kim, S. Kunieda, N.M. Larson, L. Leal, J.P. Lestone, R.C. Little, E.A. McCutchan, R.E. MacFarlane, M. MacInnes, C.M. Mattoon, R.D. McKnight, S.F. Mughabghab, G.P.A. Nobre, G. Palmiotti, A. Palumbo, M.T. Pigni, V.G. Pronyaev, R.O. Sayer, A.A. Sonzogni, N.C. Summers, P. Talou, I.J. Thompson, A. Trkov, R.L. Vogt, S.C. van der Marck, A. Wallner, M.C. White, D. Wiarda, and P.G. Young. ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data. *Nuclear Data Sheets*, 112(12):2887–2996, 2011. URL: https://www.sciencedirect.com/science/article/pii/S009037521100113X, doi:https://doi.org/10.1016/j.nds.2011.11.002.

[XSLibDAB18] et al. D.A. Brown. ENDF/B-VIII.0: The 8th Major Release of the Nuclear Reaction Data Library with CIELO-project Cross Sections, New Standards and Thermal Scattering Data. *Nuclear Data Sheets*, 148(12):1–142, 2018. URL: https://www.sciencedirect.com/science/article/pii/S0090375218300206, doi:https://doi.org/10.1016/j.nds.2018.02.001.

[XSLibDL04] M. E. Dunn and L. C. Leal. Calculating probability tables for the unresolved-resonance region using Monte Carlo methods. *Nuclear science and engineering*, 148(1):30–42, 2004. Publisher: Taylor & Francis.

[XSLibHCML16] Shane WD Hart, Cihangir Celik, G. Ivan Maldonado, and Luiz Leal. Creation of problem-dependent Doppler-broadened cross sections in the KENO Monte Carlo code. *Annals of Nuclear Energy*, 88:49–56, 2016.

[XSLibWDP+09] Dorothea Wiarda, Michael E. Dunn, Douglas E. Peplow, Thomas M. Miller, and Hatice Akkurt. Development and Testing of Endf/b-vi. 8 and Endf/b-vii. 0 Coupled Neutron-gamma Libraries for Scale 6. Technical Report ORNL/TM-2008/047, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2009.

[XSLibWWCD15] Dorothea Wiarda, Mark L. Williams, Cihangir Celik, and Michael E. Dunn. AMPX: A Modern Cross Section Processing System for Generating Nuclear Data Libraries. Technical Report, Oak Ridge National Laboratory, Charlotte, NC (USA), 9 2015.

[XS1] The “residual” is the remainder after the reaction specified by MT has taken place (for example, A-1 after an n,2n reaction on target A). This “residual” may break up further if LR>0.

[XS2] Note that the partial fission cross sections are not defined for incident charged particles.

[Origen-Data-Resources10c] Code of Federal Regulations, Title 10, Part 20.

[Origen-Data-ResourcesBC79] J. K. Bair and J. Gomez Del Campo. Neutron Yields from Alpha Particle Bombardment. *Nuclear Science and Engineering*, 71:18, 1979.

[Origen-Data-ResourcesBH73] J. K. Bair and F. X. Hass. Total Neutron Yield from the Reaction ^{13}C $\left (\alpha ,n \right) ^16O$ and ^{17,18}O $\left (\alpha ,n \right) ^20,21$Ne. *Phys. Rev. C*, 7:1356, 1973.

[Origen-Data-ResourcesCHO+11] M.B. Chadwick, M. Herman, P. Obložinský, M.E. Dunn, Y. Danon, A.C. Kahler, D.L. Smith, B. Pritychenko, G. Arbanas, R. Arcilla, R. Brewer, D.A. Brown, R. Capote, A.D. Carlson, Y.S. Cho, H. Derrien, K. Guber, G.M. Hale, S. Hoblit, S. Holloway, T.D. Johnson, T. Kawano, B.C. Kiedrowski, H. Kim, S. Kunieda, N.M. Larson, L. Leal, J.P. Lestone, R.C. Little, E.A. McCutchan, R.E. MacFarlane, M. MacInnes, C.M. Mattoon, R.D. McKnight, S.F. Mughabghab, G.P.A. Nobre, G. Palmiotti, A. Palumbo, M.T. Pigni, V.G. Pronyaev, R.O. Sayer, A.A. Sonzogni, N.C. Summers, P. Talou, I.J. Thompson, A. Trkov, R.L. Vogt, S.C. van der Marck, A. Wallner, M.C. White, D. Wiarda, and P.G. Young. ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data. *Nuclear Data Sheets*, 112(12):2887–2996, 2011. URL: https://www.sciencedirect.com/science/article/pii/S009037521100113X, doi:https://doi.org/10.1016/j.nds.2011.11.002.

[Origen-Data-ResourcesCOH+06] M.B. Chadwick, P. Obložinský, M. Herman, N.M. Greene, R.D. McKnight, D.L. Smith, P.G. Young, R.E. MacFarlane, G.M. Hale, S.C. Frankle, A.C. Kahler, T. Kawano, R.C. Little, D.G. Madland, P. Moller, R.D. Mosteller, P.R. Page, P. Talou, H. Trellue, M.C. White, W.B. Wilson, R. Arcilla, C.L. Dunford, S.F. Mughabghab, B. Pritychenko, D. Rochman, A.A. Sonzogni, C.R. Lubitz, T.H. Trumbull, J.P. Weinman, D.A. Brown, D.E. Cullen, D.P. Heinrichs, D.P. McNabb, H. Derrien, M.E. Dunn, N.M. Larson, L.C. Leal, A.D. Carlson, R.C. Block, J.B. Briggs, E.T. Cheng, H.C. Huria, M.L. Zerkle, K.S. Kozier, A. Courcelle, V. Pronyaev, and S.C. van der Marck. ENDF/B-VII.0: Next Generation Evaluated Nuclear Data Library for Nuclear Science and Technology. *Nuclear Data Sheets*, 107(12):2931–3060, 2006. URL: https://www.sciencedirect.com/science/article/pii/S0090375206000871, doi:https://doi.org/10.1016/j.nds.2006.11.001.

[Origen-Data-ResourcesCHG79] A. G. Croft, R. L. Haese, and N. B. Grove. Updated Decay and Photon Libraries for the ORIGEN Code. Technical Report ORNL/TM-6055, Union Carbide Corporation (Nuclear Division), Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2 1979.

[Origen-Data-ResourcesDSF73] L. T. Dillman, W. S. Snyder, and M. R. Ford. Nuclear Data Compilation of Utility in Medical and Biological Applications. In *Proceedings of the Symposium on Application of Nuclear Data in Science and Technology*, volume 2, 529—539. 3 1973.

[Origen-Data-ResourcesDG02] M. E. Dunn and N. M. Greene. AMPX-2000: A Cross-Section Processing System for Generating Nuclear Data for Criticality Safety Applications. In *Transactions of the American Nuclear Society*, volume 86, 118–119. 2002.

[Origen-Data-ResourcesFKS02] R. A. Forrest, J. Kopecky, and J-Ch. Sublet. The European Activation File: EAF-2003 Cross Section Library. Technical Report UKAEA FUS-486, EURATOM/UKAEA Fusion, 12 2002.

[Origen-Data-ResourcesFWP15] M. W. Francis, C. F. Weber, and M. T. Pigni. Reactor Fuel Isotopics and Code Validation for Nuclear Applications. Technical Report ORNL TM-2014/464, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 2015.

[Origen-Data-ResourcesGZ75] K. W. Geiger and L. Van Der Zwain. An Evaluation of the ^{9}Be $\left (\alpha ,n \right)$ Cross Section Data. *Nucl. Instr. and Methods*, 131:315, 1975.

[Origen-Data-ResourcesGM59] J. H. Gibbons and R. L. Macklin. Total Neutron Yields from Light Elements Under Proton and Alpha Bombardment. *Physical Review*, 114:571, 1959.

[Origen-Data-ResourcesGM71] N. B. Gove and M. J. Martin. Log-f Tables for Beta Decay. *Nuclear Data Tables*, 10:206, 1971.

[Origen-Data-ResourcesGum54] R. O. Gumprecht. Current status of the isotopic purity of hanford plutonium. Technical Report Hanford Technical Record, HW-33874, Hanford Works, Richland, WA, 1954. URL: https://www.osti.gov/biblio/88862-current-status-isotopic-purity-hanford-plutonium.

[Origen-Data-ResourcesIGW11] Eugene C. Fortune IV, Ian C. Gauld, and C.-K. Chris Wang. Gamma Dose Rate Near a New :sup:`252` Cf Brachytherapy Source. *Nuclear Technology*, 175(1):73–76, 2011. URL: https://doi.org/10.13182/NT11-A12272, doi:10.13182/NT11-A12272.

[Origen-Data-ResourcesLS71] D. L. Lessor and R. E. Schenter. Neutron Spectra from $\left (\alpha ,n \right)$ Reaction in Plutonium Compounds Calculated from Hauser-Feshbach Reaction Theory. Technical Report BNWL-B-109, Brookhaven National Laboratory, Upton, NY, 1971.

[Origen-Data-ResourcesMBM78] M. Balakrishnan, S. Kailas and M. K. Mehta. A Study of the Reaction `math:^19F \left ( \alpha ,n \right ) ^22Na` in the Bombarding Energy Range 2.6 to 5.1 MeV. *Pramana*, 10:329, 1978.

[Origen-Data-ResourcesPW81] R. T. Perry and W. B. Wilson. Neutron Production from :math:`\left ( \alpha ,n \right )` Reactions and Spontaneous Fission in ThO :sub:`2`, UO :sub:`2`, and (U,Pu)O :sub:`2` Fuels. Technical Report LA-8869-MS, Los Alamos National Laboratory, Los Alamos, NM, 1981.

[Origen-Data-ResourcesPFG15] M. T. Pigni, M. W. Francis, and I. C. Gauld. Investigation of Inconsistent ENDF/B-VII.1 Independent and Cumulative Fission Product Yields with Proposed Revisions. *Nuclear Data Sheets*, 123:231–236, 2015. doi:https://doi.org/10.1016/j.nds.2014.12.040.

[Origen-Data-ResourcesSho00] E. F. Shores. Data Updates for the SOURCES-4A Computer Code. Technical Report LA-UR-00-5016, Los Alamos National Laboratory, 10 2000. SOURCES-4C available from the Radiation Safety Information Computational Center (RSICC) as code package C00661.

[Origen-Data-ResourcesSKFK03] J.-Ch. Sublet, A. J. Koning, R. A. Forrest, and J. Kopecky. The JEFF-3.0/A Neutron Activation File-EAF-2003 into ENDF-6 Format. Technical Report JEFDOC-982, Commissariat à l'Energie Atomique, France, 11 2003.

[Origen-Data-ResourcesWPS+83] W. B. Wilson, R. T. Pery, J. E. Stewart, T. R. England, D. G. Madland, and E. D. Arthur. Development of the SOURCES Code and Data Library for the Calculation of Neutron Sources and Spectra from :math:`\left ( \alpha ,n \right )` Reactions, Spontaneous Fission, and :math:`beta^-` Delayed Neutrons. Technical Report LA-9841-PR, Los Alamos National Laboratory, Los Alamos, NM, 1983.

[Origen-Data-ResourcesWFHZ75] S. E. Woosley, W. A. Fowler, J. A. Holmes, and B. A. Zimmerman. Tables of Thermonuclear Reaction Rate Data for Intermediate Mass Nuclei . Technical Report Preprint OAP-422, Kellogg Radiation Laboratory, California Institute of Technology, Pasadena, CA, 1975. unpublished.

[Origen-Data-ResourcesZie77] J. F. Zieger. Helium Stopping Power and Ranges in All Elemental Matter. In *The Stopping Power and Ranges of Ions in Matter Series*, volume 4. Pergamon Press, New York, 1977.

[Origen-Data-ResourcesUEPAEPA93] US Environmental Protection Agency (EPA) . External Exposure to Radionuclides in Air, Water, and Soil. Technical Report Federal Guidance Report 12, US Environmental Protection Agency, Washington, DC, 1993.

[Origen-Data-ResourcesInternationalCoRPICRP77] International Commission on Radiological Protection (ICRP). Age-Dependent Doses to Members of the Public from Intake of Radionuclides: Part 5, Compilation of Ingestion and Inhalation Dose Coefficients. In *ICRP Publication 72*. Pergamon Press, New York, 1977.

[Origen-Data-ResourcesNuclearEAgency00] Nuclear Energy Agency. The JEF-2.2 Nuclear Data Library. Technical Report JEFF Report 17, Organisation For Economic Co-Operation And Development, 4 2000.

[COVLIBBW04] Bryan L. Broadhead and J. J. Wagschal. The Fission Spectrum Uncertainty. *PHYSOR 2004–The Physics of Fuel Cycles and Advanced Nuclear Systems: Global Developments*, pages 25–29, 2004.

[COVLIBCHO+11] M.B. Chadwick, M. Herman, P. Obložinský, M.E. Dunn, Y. Danon, A.C. Kahler, D.L. Smith, B. Pritychenko, G. Arbanas, R. Arcilla, R. Brewer, D.A. Brown, R. Capote, A.D. Carlson, Y.S. Cho, H. Derrien, K. Guber, G.M. Hale, S. Hoblit, S. Holloway, T.D. Johnson, T. Kawano, B.C. Kiedrowski, H. Kim, S. Kunieda, N.M. Larson, L. Leal, J.P. Lestone, R.C. Little, E.A. McCutchan, R.E. MacFarlane, M. MacInnes, C.M. Mattoon, R.D. McKnight, S.F. Mughabghab, G.P.A. Nobre, G. Palmiotti, A. Palumbo, M.T. Pigni, V.G. Pronyaev, R.O. Sayer, A.A. Sonzogni, N.C. Summers, P. Talou, I.J. Thompson, A. Trkov, R.L. Vogt, S.C. van der Marck, A. Wallner, M.C. White, D. Wiarda, and P.G. Young. ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data. *Nuclear Data Sheets*, 112(12):2887–2996, 2011. URL: https://www.sciencedirect.com/science/article/pii/S009037521100113X, doi:https://doi.org/10.1016/j.nds.2011.11.002.

[COVLIBHal08] Gerald Hale. Covariances from light-element r-martix analyses. Technical Report, Los Alamos National Laboratory, Los Alamos, NM (USA), 2008.

[COVLIBHD71] R. J. Howerton and R. J. Doyas. Fission temperatures as a function of the average number of neutrons from fission. *Nuclear Science and Engineering*, 46(3):414–416, 1971. Publisher: Taylor & Francis.

[COVLIBKTY+08] T. Kawano, P. Talou, P. G. Young, G. Hale, M. B. Chadwick, and R. C. Little. Evaluation of covariances for actinides and light elements at LANL. *Nuclear Data Sheets*, 109(12):2817–2821, 2008.

[COVLIBLLD+06] N. M. Larson, L. C. Leal, H. Derrien, G. Arbanas, R. O. Sayer, and D. Wiarda. A systematic description of the generation of covariance matrices. In *PHYSOR-2006: ANS Topical Meeting on Reactor Physics*, 10–14. 2006.

[COVLIBLWRD08] Luiz C. Leal, D. Wiarda, Bradley T. Rearden, and Herve Derrien. 233 U Cross-Section and Covariance Data Update for SCALE 5.1 Libraries. *ORNL/TM-2007/115, Oak Ridge National Laboratory, Oak Ridge, Tenn*, 2008.

[COVLIBLKH+08] R. C. Little, T. Kawano, G. D. Hale, M. T. Pigni, M. Herman, P. Obložinsky, M. L. Williams, M. E. Dunn, G. Arbanas, and D. Wiarda. Low-fidelity Covariance Project. *Nuclear Data Sheets*, 109(12):2828–2833, 2008. Publisher: Elsevier.

[COVLIBMMW80] R. E. Maerker, J. H. Marable, and J. J. Wagschal. Estimation of the uncertainties in the ENDF/BV Uranium 235 fission spectrum. In *Transactions of the American Nuclear Society*, volume 35. 1980.

[COVLIBMug03] S. F. Mughabghab. Thermal neutron capture cross sections resonance integrals and g-factors. *International Atomic Energy Agency*, 2003.

[COVLIBMug06] Said F. Mughabghab. *Atlas of Neutron Resonances: Resonance Parameters and Thermal Cross Sections. Z= 1-100*. Elsevier, 2006.

[COVLIBPHO09] M. T. Pigni, M. Herman, and P. Obložinsky. Extensive Set of Cross-Section Covariance Estimates in the Fast Neutron Region. *Nuclear science and engineering*, 162(1):25–40, 2009. Publisher: Taylor & Francis.

[COVLIBRHOM07] D. Rochman, M. Herman, P. Oblozinsky, and S. F. Mughabghab. Preliminary cross-section covariances for WPEC Subgroup 26. Technical Report BNL-77407-2007-IR, Brookhaven National Laboratory, Upton, NY (USA), 2007.

[COVLIBWD06] D. Wiarda and M. E. Dunn. PUFF-IV: A Code for Processing ENDF Uncertainty Data into Multigroup Covariance Matrices. Technical Report ORNL/TM-2006/147, Oak Ridge National Laboratory, Oak Ridge, TN (USA), 10 2006.

[COVLIBWALD08] Dorothea Wiarda, Goran Arbanas, L. Leal, and Michael E. Dunn. Recent advances with the AMPX covariance processing capabilities in PUFF-IV. *Nuclear Data Sheets*, 109(12):2791–2795, 2008. Publisher: Elsevier.

[COVLIBWBDR07] Mark L. Williams, Bryan L. Broadhead, Michael E. Dunn, and Bradley T. Rearden. Approximate Techniques for Representing Nuclear Data Uncertainties. In *Eighth International Topical Meeting on Nuclear Applications and Utilization of Accelerators (AccApp'07)," Pocatello, ID (July 30-Aug. 2, 2007)*. 7 2007.

[COVLIBWIMR14] Mark L. Williams, Germina Ilas, William J. Marshall, and Bradley T. Rearden. Applications of nuclear data covariances to criticality safety and spent fuel characterization. *Nuclear Data Sheets*, 118:341–345, 2014.

[COVLIBWR08] Mark L. Williams and Bradley T. Rearden. SCALE-6 Sensitivity/uncertainty methods and covariance data. *Nuclear Data Sheets*, 109(12):2796–2800, 2008.

[SCALE-DATA-WWCD15] Dorothea Wiarda, Mark L. Williams, Cihangir Celik, and Michael E. Dunn. AMPX: A Modern Cross Section Processing System for Generating Nuclear Data Libraries. Technical Report, Oak Ridge National Laboratory, Charlotte, NC (USA), 9 2015.

[WORKER1] D. Wiarda, M. L. Williams, C. Celik, and M. E. Dunn, "AMPX: A Modern Cross Section Processing System For Generating Nuclear Data Libraries," *Proceedings of International Conference on Nuclear Criticality Safety,* Charlotte, N.C., Sept. 13-17, 2015.

[ice1] * Formerly with Oak Ridge National Laboratory.

[UTILS-Fra76] S. K. Fraley. User's guide for ice. Technical Report ORNL/CSD/TM-9, Union Carbide Corporation (Nuclear Division), Oak Ridge National Laboratory, 7 1976.

[UTILS-WWCD15] Dorothea Wiarda, Mark L. Williams, Cihangir Celik, and Michael E. Dunn. AMPX: A Modern Cross Section Processing System for Generating Nuclear Data Libraries. Technical Report, Oak Ridge National Laboratory, Charlotte, NC (USA), 9 2015.