OAK RIDGE
NATIONAL LABORATORY
MANAGED BY UT-BATTELLE
FOR THE DEPARTMENT OF ENERGY

ORNL/TM-2015/322

Coupling of Thermochemistry Solver THERMOCHIMICA with MOOSE/BISON

June 30, 2015

Prepared by
S. Simunovic and T.M. Besmann

UT-BATTELLE
ORNL-27 (4-00)

DOCUMENT AVAILABILITY

Reports produced after January 1, 1996, are generally available free via the U.S. Department of Energy (DOE) Information Bridge.

Web site http://www.osti.gov/bridge

Reports produced before January 1, 1996, may be purchased by members of the public from the following source.

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
Telephone 703-605-6000 (1-800-553-6847)
TDD 703-487-4639
Fax 703-605-6900
E-mail info@ntis.gov
Web site http://www.ntis.gov/support/ordernowabout.htm

Reports are available to DOE employees, DOE contractors, Energy Technology Data Exchange (ETDE) representatives, and International Nuclear Information System (INIS) representatives from the following source.

Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831
Telephone 865-576-8401
Fax 865-576-5728
E-mail reports@osti.gov
Web site http://www.osti.gov/contact.html

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ORNL/TM-2015/56467

Computer Science and Mathematics Division

COUPLING OF THERMOCHEMISTRY SOLVER THERMOCHIMICA WITH MOOSE/BISON

S. Simunovic and T. M. Besmann

Date Published: June 2015

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6283
managed by
UT-BATTELLE, LLC
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

Executive Summary

THERMOCHIMICA is a computational library for chemical thermodynamics. It determines a unique combination of phases and their compositions for a prescribed chemical composition, temperature and pressure. The solver and the underlying thermodynamic models can be used to estimate the chemical state and various constitutive and transport properties necessary for nuclear fuel performance modeling. Our NEAMS research project focuses on the development of nuclear fuel thermodynamics models and computational thermodynamics equilibrium solvers, and their implementation into Idaho National Laboratory (INL) nuclear fuel performance simulation system MOOSE/BISON. This report documents the software integration of THERMOCHIMICA with MOOSE/BISON using the current INL development framework GitLab. The THERMOCHIMICA implementation is demonstrated on modeling oxygen related phenomena in irradiated fuel, such as calculation of oxygen to metal ratio in the fluorite phase, oxygen partial pressure, oxygen chemical potential and oxygen transport. Experimental measurements from the open literature were used to validate the implemented models and illustrate functionality of the developed thermodynamics module. The calculations are based on chemical element inventory provided by neutronics, isotopic depletion, transmutation and decay calculations in the SCALE system. The burnup calculation in BISON can also be used for elemental composition although it provides only a limited set. Coupling of THERMOCHIMICA with other codes that can provide chemical composition of irradiated fuel for BISON simulations is currently being explored.

Notice: This manuscript has been authored by UT-Battelle, LLC, under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

1. Introduction

Irradiated nuclear fuels are complex multicomponent, multiphase systems. In an operating nuclear reactor, the fuel's composition continuously changes as fission products are created and actinides are consumed which happens under high temperature and composition gradients [1]. The resulting thermochemical properties strongly influence the fuel's behavior at normal operating conditions and during transients [2]. The chemical state influences physical properties and response of the fuel including thermal conductivity, heat capacity, swelling, creep, fission product volatility, etc. [3]. As LWR UO_2 fuels are burned, they generate up to sixty transuranic and fission product elements resulting in a complex chemical system [4]. In order to model the evolving chemical composition and the resulting material properties of nuclear fuels, we are developing computational thermodynamic material models and thermodynamics equilibrium solver, and implementing them in the software library THERMOCHIMICA [5]. Thermodynamic equilibrium implicitly assumes that the resulting states are time independent and not controlled by chemical kinetics. While the equilibrium state provides no direct information for transport it can provide source and sink terms and activity/vapor pressure for computing mass flux. It can also provide source terms for microstructural evolution, potential corrosion mechanisms, and gas species pressures, and support modeling species and thermal transport in fuels [1].

The THERMOCHIMICA library was incorporated into the GitHub software development system at Idaho National Laboratory [6]. It has been coupled with MOOSE (Multiphysics Object Oriented Simulation Environment) [7] -based code BISON [8-10]. MOOSE is a computational framework for development of simulation tools for solving coupled, nonlinear partial differential equations (PDEs) and BISON is a finite element method (FEM) code built on the MOOSE framework for simulation of coupled multi-physics phenomena in nuclear fuel elements and nuclear reactor engineering.

Thermodynamic coupling implementation is illustrated with modeling oxygen-related phenomena in irradiated fuel, namely, calculation of oxygen-to-metal ratio in the fluorite phase of nuclear fuel, oxygen potential and oxygen transport. The developed thermodynamic models and their implementation were verified with experimental measurements of composition and oxygen potential of high burnup fuel [11]. The verification follows the procedure established in Piro et al. [2] and in the previous NEAMS reports. The thermodynamic calculations were based on chemical element inventory provided by neutronics, isotopic depletion, transmutation and decay simulation using the SCALE [12] system for the fuel composition and power history used in the experiments [11]. The experimental data for model verification includes oxidation and local fission product concentration measurements on the same sample. The isotopic predictions using SCALE and thermochemical prediction using values and models in THERMOCHIMICA resulted in good agreement with the available experimental measurements. The burnup model [13] in BISON can also be used for elemental composition although the model provides only limited information (i.e. isotopes of U and Pu). Coupling with other codes that can provide more detailed chemical composition of irradiated fuel is currently being explored. A coupled oxygen transport problem in thermal gradient was also used for illustration of the multiphysics applicability of the thermodynamics calculations.

The THERMOCHIMICA library can be used to support modeling various physical phenomena in nuclear fuel beyond the examples provided in this report. The objective of the software implementation, models and examples documented in this report is primarily to illustrate the possibilities that the thermodynamics calculations can provide for modeling of the underlying physical mechanisms in nuclear fuel and their interactions during irradiation. The report also describes the current software interfaces to the thermodynamic solver library and a typical analysis workflow.

2. THERMOCHIMICA Software Implementation

A detailed description of the computational implementation of THERMOCHIMICA can be found in a technical report [14]. The equilibrium solver determines a unique combination of phases and their compositions at thermochemical equilibrium. The computational algorithms and software were developed at the Royal Military College of Canada and at the Oak Ridge National Laboratory and have been released in the public domain. THERMOCHIMICA routines were developed for incorporation of equilibrium thermochemical calculations into other simulation codes, with emphasis on nuclear fuel problems. Considerable research efforts have been undertaken to achieve high computational performance, primarily through advanced algorithm development, such as improved estimation techniques and domain tailored non-linear solvers [15, 16].

THERMOCHIMICA software consists of a library of material thermodynamic properties, a select set of thermodynamic models, and a numerical solver to minimize global Gibbs energy while respecting the Gibbs Phase Rule and conservation of mass. THERMOCHIMICA utilizes input parameters and a specific model framework to predict a unique combination of species and phases that are stable at thermodynamic equilibrium and then outputs material properties and boundary conditions. Specifically, the software uses input parameters such as temperature, pressure, and species mass together with a thermodynamic database such as that described above, to calculate species mole fraction, chemical potentials, thermochemical activities, moles of phases present, enthalpy, heat capacity, and Gibbs energy. The software has been designed from the beginning to handle large thermodynamic systems characteristic of irradiated nuclear fuels, thus eliminating concerns of software limitations regarding the number of chemical elements, species, pure condensed phases or solution phases.

As mentioned above, THERMOCHIMICA solves for thermodynamic equilibrium and does not take chemical kinetic information into consideration. In reality, chemical equilibrium is not achieved instantaneously. However, chemical kinetics effects are not very significant at sufficiently high temperatures, long time periods, and when atoms of the various chemical elements are randomly mixed in the system. In nuclear fuels under normal operating conditions, chemical equilibrium is achieved quickly due to the high temperatures and chemical reactions reach completion over the long time periods between refueling. Also, due to the nature of fission, the atoms of the various elements representing the transuranics and fission products are randomly mixed in irradiated nuclear fuel. This implicit assumption of local thermodynamic equilibrium must be taken into account in model development and in the solution procedures in the code.

THERMOCHIMICA is written in FORTRAN and complies with the FORTRAN 90/2003 standard. A detailed description of each subroutine in the code can be found in the Doxygen [17] documentation located in the Appendix of the THERMOCHIMICA Technical Report [14]. The documentation for each subroutine includes a description of its purpose, and a description of all input/output parameters. Descriptions of local subroutine variables are also provided. Additional information for the THERMOCHIMICA Application Programming Interface (API) for MOOSE/BISON will be updated as new functionality is required by the users or as the API implementation is evolving to improve the robustness and performance of the code. All of the programming in THERMOCHIMICA is original unless otherwise specified in the source code. In cases where subroutines have been modified from the published code, such as Numerical Recipes [18], the origins of the code have been referenced appropriately.

THERMOCHIMICA can be used for development of stand-alone programs or it can be directly

coupled to other codes. The software does not have a graphical user interface (GUI). The main purpose of the software is to be directly coupled with a multi-physics code to provide material properties and boundary conditions for various physical phenomena. Temperature, hydrostatic pressure and the mass of each chemical element must be specified as inputs. Users also have to provide a thermodynamic values and models for the systems in the form of a ChemSage data-file. ChemSage data-files can be generated by the commercial software package FactSage [19, 20] or can be manually produced.

Examples of results that can be provided as output from THERMOCHIMICA solver through its API are:

- the stable phases at equilibrium,
- the mass of solution species and phases at equilibrium,
- mole fractions of solution phase constituents,
- thermochemical activities (which are related to partial pressures for gaseous species),
- chemical potentials of solution species and phases, and
- integral Gibbs energy (referenced relative to standard state)

An example of a simple FORTRAN program that calculates thermochemical equilibrium of a simple uranium-oxygen system is given in Listing 1.

```
program thermo
  USE ModuleThermoIO
  implicit none

  ! Program variables are declared in ModuleThermoIO

  ! Initialize variables:
  dElementMass = 0D0
  INFOThermo = 0

  ! Define variable units:
  cThermoInputUnits(1) = 'K'
  cThermoInputUnits(2) = 'atm'
  cThermoInputUnits(3) = 'moles'

  ! Define thermodynamic data-file path:
  cThermoFileName = 'data/Example4.dat'

  ! Define the temperature and pressure:
  dPressure = 1D0
  dTemperature = 500D0

  ! Define the mass of each chemical element:
  dElementMass(92) = 1.00D0 ! Element ID 92 = U
  dElementMass(8)  = 2.01D0 ! Element ID 8 = O

  ! Parse the ChemSage data-file:
  call ParseCSDataFile(cThermoFileName)

  ! Call THERMOCHIMICA:
  if (INFOThermo == 0) call Thermochimica

  ! Reset variables (optional):
  if (INFOThermo == 0) call ResetThermoAll

  ! Call the debugger (optional):
  call ThermoDebug

end program thermo
```

Listing 1. Code listing for a program for simple uranium-oxygen system

3. THERMOCHIMICA Application Programming Interface for BISON

The main computer language for MOOSE/BISON is C++. In order to be able to use THERMOCHIMICA in MOOSE/BISON, it was necessary to write an API for THERMOCHIMICA that conforms to the MOOSE/BISON programming model. The API was written in FORTRAN 90. The definitions of the functions that are accessible from BISON are defined in file Thermochimica.h. The function declarations are given in Listing 2.

```
#include "Moose.h"

namespace Thermochimica
{
  extern "C"
  {
    void FORTRAN_CALL(setthermofilename)(const char*);
    void FORTRAN_CALL(setunittemperature)(const char*);
    void FORTRAN_CALL(setunitpressure)(const char*);
    void FORTRAN_CALL(setunitmass)(const char*);
    void FORTRAN_CALL(setunits)
         ( const char*, const char*, const char*);
    void FORTRAN_CALL(setstandardunits)();

    void FORTRAN_CALL(ssinitiatezrhd)();
    void FORTRAN_CALL(ssinitiateuo2px)();
    void FORTRAN_CALL(compotomratio)(char*, double*, int*);
    void FORTRAN_CALL(getmolfraction)(int*, double*, int*);
    void FORTRAN_CALL(getchemicalpotential)(int*, double*, int*);
    void FORTRAN_CALL(getelementpotential)(int*, double*, int*);
    void FORTRAN_CALL(ssparsecsdatafile)();
    void FORTRAN_CALL(ssparsesolncomponent)
         (int*m, char*, int*, double*);
    void FORTRAN_CALL(ssfindsolnspecies)
         (char*, int&, int&, double&);
    void FORTRAN_CALL(setprintresultsmode)( int* );
    void FORTRAN_CALL(appminbtomolinvol)
        ( double*, double*, double*, double*, double*, int*,
          double*, double*);
    void FORTRAN_CALL(setelementmass)( int*, double*);
    void FORTRAN_CALL(settemperaturepressure)(double*, double*);
    void FORTRAN_CALL(checkinfothermo)(int*);
    void FORTRAN_CALL(thermochimica)();
    void FORTRAN_CALL(solphaseparse)( int*, double*);
    void FORTRAN_CALL(thermodebug)();
    void FORTRAN_CALL(printresults)();
    void FORTRAN_CALL(resetinfothermo)();
    void FORTRAN_CALL(resetthermo)();
    void FORTRAN_CALL(resetthermoall)();
  }
  void ConvertToFortran(char*, std::size_t, const char*);
}
```

Listing 2. Code listing for API for THERMOCHIMICA

A brief description for each routine is given below in Table 1.

Table 1. Brief description of API in Thermochimica.h

Function Name	Description
setthermofilename	Sets input file for thermodynamics model
setunittemperature	Sets temperature unit
setunitpressure	Sets pressure unit
setunitmass	Sets mass unit
setunits	Sets all units for the problem
setstandardunits	Sets standard units for the problem
ssinitiatezrhd	Initialization (input file and units) for the Zr-H problem
ssinitiateuo2px	Initialization (input file and units) for the UO2+x problem
compotomratio	Compute oxygen to metal ration in the flourite phase
getmolfraction	Get mol fraction of the species of interest
getchemicalpotential	Get chemical potential of the species
getelementpotential	Get chemical potential of the chemical element
ssparsecsdatafile	Reads and parses thermodynamics input file. Input file and units need to be set beforehand.
ssparsesolncomponent	Returns number of gram-atoms for a chemical element in a specified solution phase component.
ssfindsolnspecies	Returns number of moles for a chemical element in a specified set of solution phase components.
setprintresultsmode	Sets print mode.
appminbtomolinvol	Converts ppm of chemical element in species composed of two chemical elements into number of moles for each element in a specified volume. May be normalized to one of the elements.
setelementmass	Sets amount of mass of a specified chemical element.
settemperaturepressure	Sets temperature and pressure for which thermochemical equilibrium is calculated.
checkinfothermo	Returns the state of the solver.
thermochimica	Main solver for thermochemical equilibrium.
thermodebug	Routine for debugging the state of the solver.
printresults	Prints the results.
resetinfothermo	Resets solver state that enables a new calculation for specified thermodynamics model.
resetthermo	Resets solver.
resetthermoall	Total reset.
ConvertToFortran	Converts string to character array for FORTRAN

The details and usage are available in the source file of the functions. The API follows the naming convention of THERMOCHIMICA subroutines. The current API implementation will be expanded with additional functions to query specific species in the results. The string-based search and indexing of the results will be replaced with indexed representation to improve computational performance.

4. Validation Examples

4.1 Modeling of the Oxidation State of the LWR Nuclear Fuel Material using THERMOCHIMICA

The isotope composition of high burnup LWR fuel was calculated using the SCALE system for the fuel and power profile of Walker et al. [11]. The irradiation history of the fuel is shown in Figure 1.

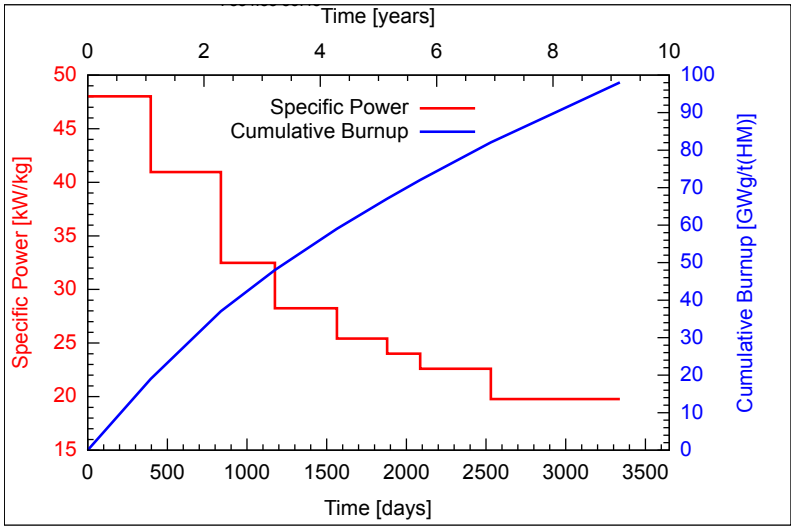


Figure 1. The irradiation history of UO_2 fuel used in the current simulation.

The SCALE simulation was used to develop a realistic element composition necessary to test the performance of the THERMOCHIMICA solver. The initial unirradiated fuel was slightly hyperstoichiometric with an enrichment of 3.5%. The outer diameter of the fuel was 9.3 mm and had 95% theoretical density. The fuel was that of a commercial Siemens 15x15 PWR fuel assembly. It was irradiated over a period of 9 power cycles to an integral burnup of 102 GWd/t(U). The SCALE 2D model configuration is shown in Figure 2.

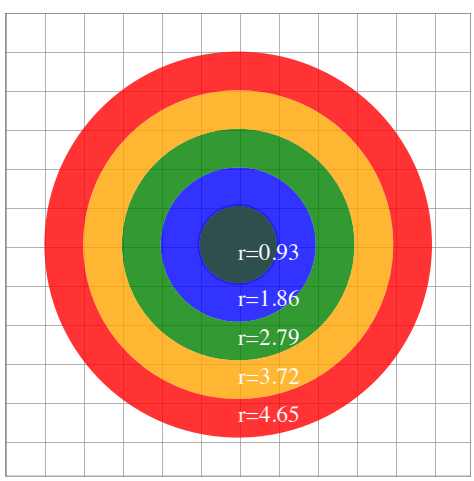


Figure 2. SCALE model geometry where r is the outer radius of the annular section in mm.

In contrast to the procedure employed in Piro et al. [2], a multi zone SCALE model was used. This approach accounts for the feedback of the self-shielding of cross sections and a new transport solution during burnup. In order to provide an adequate resolution of elemental distribution in the high burnup regions, many more rings would be needed. However, for the purpose of validation of the current model, the above configuration was considered to be sufficient. A high resolution model is currently under development and will be the subject of subsequent effort. The details of the SCALE system calculations can be found in references [2] and [12]. The resulting predicted average fission product inventory in the fuel and its comparison with experimental measurements [11] is shown in Figure 3.

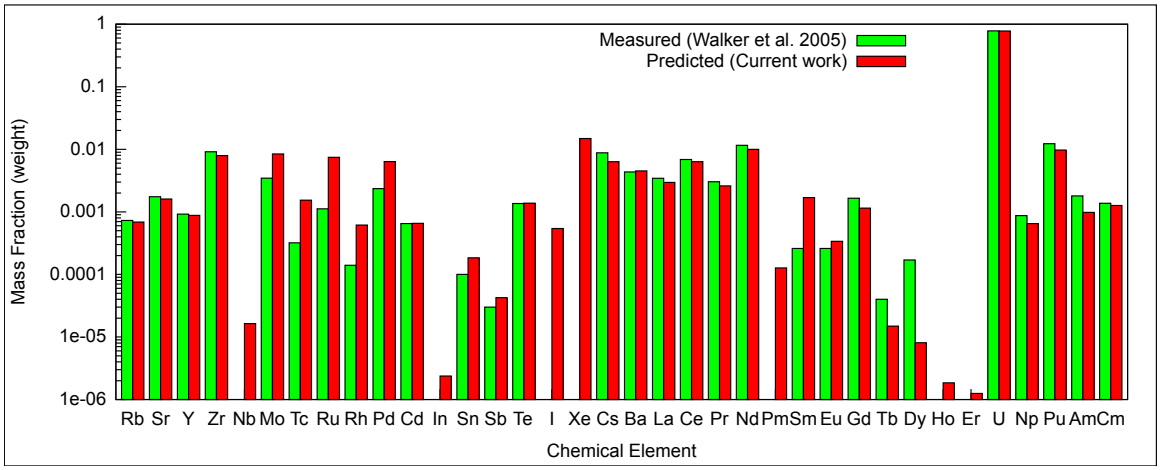


Figure 3. The predicted average chemical element inventory compared to experimental measurements [11].

The predicted results are in a good agreement with the experimental measurements. The predicted radial distribution of the element inventory is shown in Figure 4.

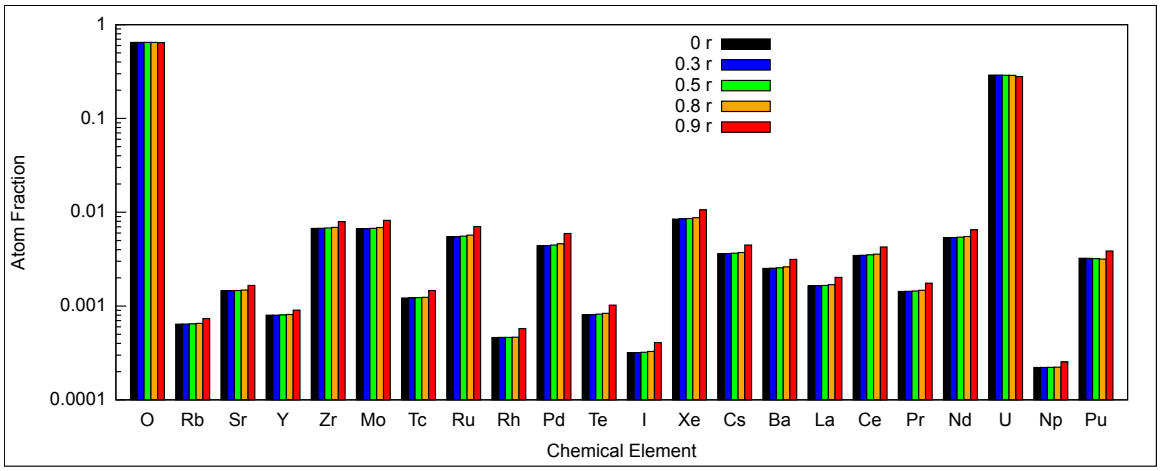


Figure 4. The predicted distribution of chemical element inventory along the radius of the pellet.

The modeling of the high burnup region, i.e., the rim region, can be enhanced by using a larger number of ring elements in the SCALE simulation.

The chemical element composition for different radial locations of nuclear fuel was used in thermodynamic calculations of the oxidation state of the fuel using the implemented subroutines in BISON. The results are summarized in Figures 5 and 6. Figure 6 shows the oxygen potential as a function of temperature for different locations in the fuel compared to experimental measurements.

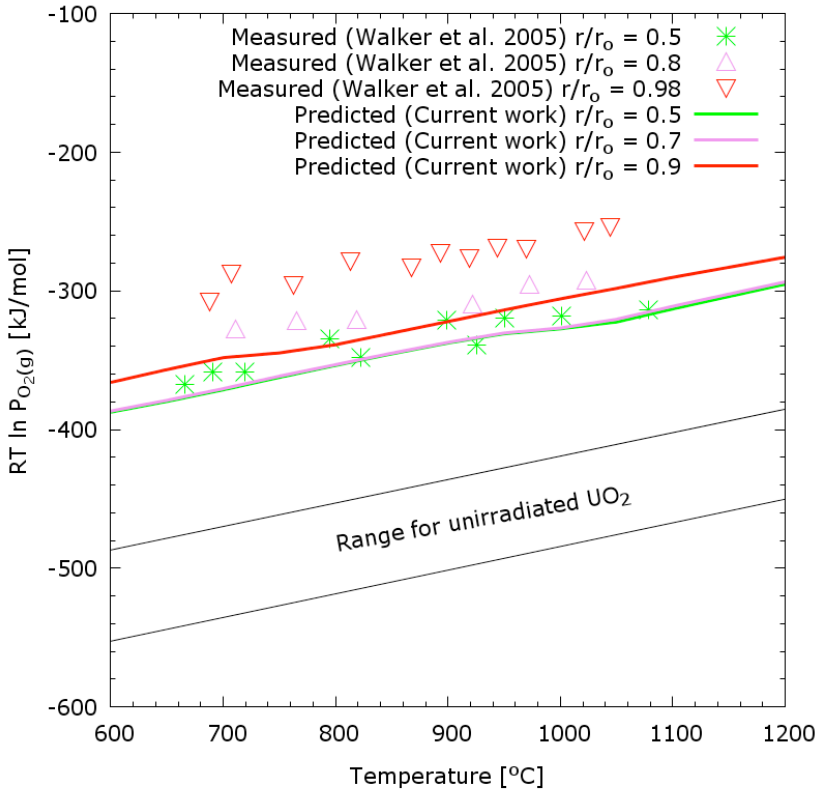


Figure 5. The predicted oxygen potential compared to experimental measurements of Walker et al. [11].

Note that the locations of the predicted values do not coincide exactly to the experimental measurements due to the specific discretization used in SCALE. Nevertheless, the predicted results are very close to those of the experiments, especially considering that the experiments had an estimated error of 10% [2]. The range for unirradiated UO_2 is shown for comparison. The predicted variation of the oxygen potential along the radius of the pellet is shown in Figure 6.

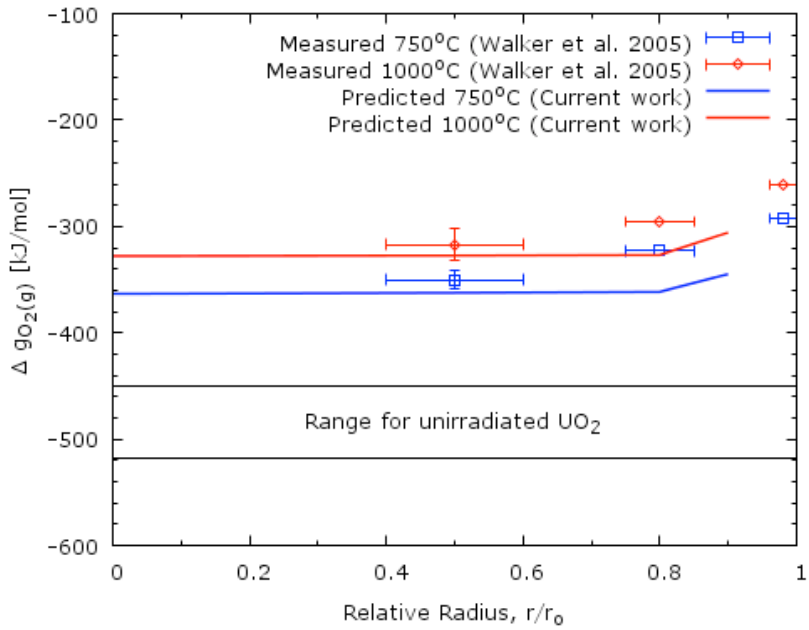


Figure 6. The predicted oxygen potential distribution along the radius of the pellet compared to experimental measurements of Walker et al. [11].

Taking into consideration that the SCALE computation only had 5 zones along the radius of the pellet, and that the burnup and thus fission products concentration increases rapidly in the sub-millimeter, rim zone in the pellet periphery, the predicted results show good agreement with experiment.

4.2 Thermodynamic State Computations in BISON using THERMOCHIMICA

New modules for oxidation-related calculations, material properties and transport were implemented in BISON. In the MOOSE object nomenclature, the objects are Initial Condition module, Material module, Kernel module and Auxiliary Kernel module. For example, initial condition module `UO2PXOxygenIc.C` calculates departure from stoichiometry for the uranium-oxygen system. Auxiliary Kernel module `OxygenThermochimicaAux.C` calculates oxygen-to-metal ratio in the fluorite phase, oxygen partial pressure, and oxygen potential in the nuclear fuel. We also illustrate multi-physics coupling between thermodynamics models, material properties, species transport and heat of diffusion. To this end, the initial condition module `UO2PXOxygenIc.C` has been coupled with the thermal material model for hyperstoichiometric fuel `ThermalUO2PX.C` and oxygen diffusion kernel `OxygenDiffusion.C`. These modules are implemented to replicate the model for coupled oxygen transport by Mihaila et al. [21, 22].

Several other kernels and initial conditions have also been implemented in the authors' version of MOOSE/BISON but they have not been included in the current source code version in the INL GitLab repository. The objective of the current code integration step is to establish and validate mechanisms and modes of THERMOCHIMICA implementation in the GitLab repository. This step involves code review by the INL code developers in order to ensure that the overall coding guidelines are followed and that the submitted code satisfies the criteria for inclusion into the overall system. This evaluation is simpler for smaller sets of submitted code, and after its acceptance, further code addition to the repository will follow. Addition of a thermodynamic solver to MOOSE/BISON constitutes a unique capability that is not readily available in similar simulation frameworks. It

significantly adds to the modeling capability and provides numerous opportunities for coupling of various physical phenomena. The purpose of the current implementation is to illustrate the possibilities more than to create a self-contained module for routine analysis.

4.3 Example Problem for Oxygen Diffusion using BISON and THERMOCHIMICA Module

The microscopic mechanisms of oxygen diffusion depend on stoichiometry in the actinide dioxides [23]. The phenomenais very complex and the thermodynamic solver can be used in combination with lower length scale models to develop better understanding of the phenomena. The model presented here and implemented in BISON is based on the deviation of stoichiometry of the UO_2 solid solution phase. The calculation of the deviation is based on the O/M ratio of the UO_2 solid solution phase [21, 22, 24]. A three sublattice CEF model is used to represent this phase. The first sublattice represents the cations on cubic coordinated sites, the second sublattice represents tetrahedrally coordinated oxygen anions (or vacancies) and the third sublattice represents octahedrally coordinated interstitial oxygen anions or vacancies. Input for the module for calculating initial condition for deviation from stoichiometry is illustrated in the snippet of the input file of Listing 3.

```
[ICs]
  [./mat_1]
    block = 1
    type = UO2PXOxygenIc
    variable = oxygen
    temp = temp
    pressure = 1.0
    U = 3.6971E+03
    O = 8.4030E+03
    Pu = 5.7050E+01
    Ba = 3.5550E+01
    Ce = 5.4720E+01
    Cs = 5.7380E+01
    I = 5.3120E+00
    La = 2.5200E+01
    Mo = 1.0080E+02
    Nd = 7.8800E+01
    Pd = 6.7630E+01
    Pr = 2.1840E+01
    Rb = 9.3000E+00
    Rh = 7.5310E+00
    Ru = 8.9420E+01
    Sr = 2.1870E+01
    Tc = 1.8530E+01
    Te = 1.2290E+01
    Xe = 1.3150E+02
    Y = 1.1550E+01
    Zr = 9.8930E+01
  [../]
[]
```

Listing 3. Input example for defining the initial composition. The information is used for calculating thermodynamic equilibrium, oxygen/metal ratio and deviation for stoichiometry

In the above example, temperature is a coupled field denoted with `temp` whereas the variable `oxygen` denotes deviation from stoichiometry in the UO_2 solid solution phase. The elemental composition used corresponds to the highest burnup in Figure 2. For a simple test cube geometry and uniform temperature and pressure as above, an initial condition calculation of the deviation from stoichiometry is shown in Figure 7. The calculated value corresponds to x in UO_{2+x} as plotted in Figure 7.

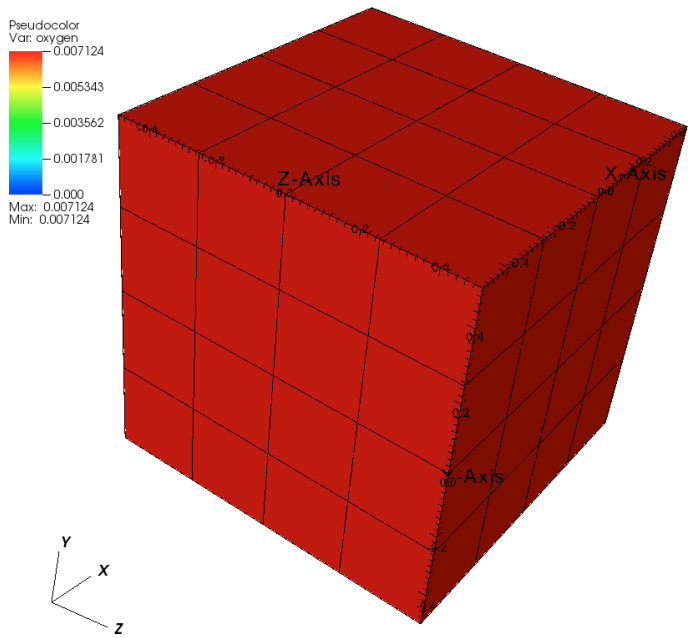


Figure 7. Initial condition calculation of the deviation from stoichiometry for the composition in Figure 3 and uniform temperature (2000K) and pressure (1 bar) in a unit (1 m) cube. No oxygen migration occurs due to constant temperature and concentration.

A simple example problem that uses thermal and oxygen transport models from the recent literature [21, 24] has been implemented in BISON. Material properties used in this model are functions of temperature and implicitly composition through the deviation of stoichiometry calculated from the O/M in the UO_2 solid solution phase. The problem is used for illustrative purposes only and for the verification of the THERMOCHIMICA implementation. The result for the problem with imposed thermal gradient and initial composition is shown in Figure 8. The trends from the results correspond to the expected values from the composition.

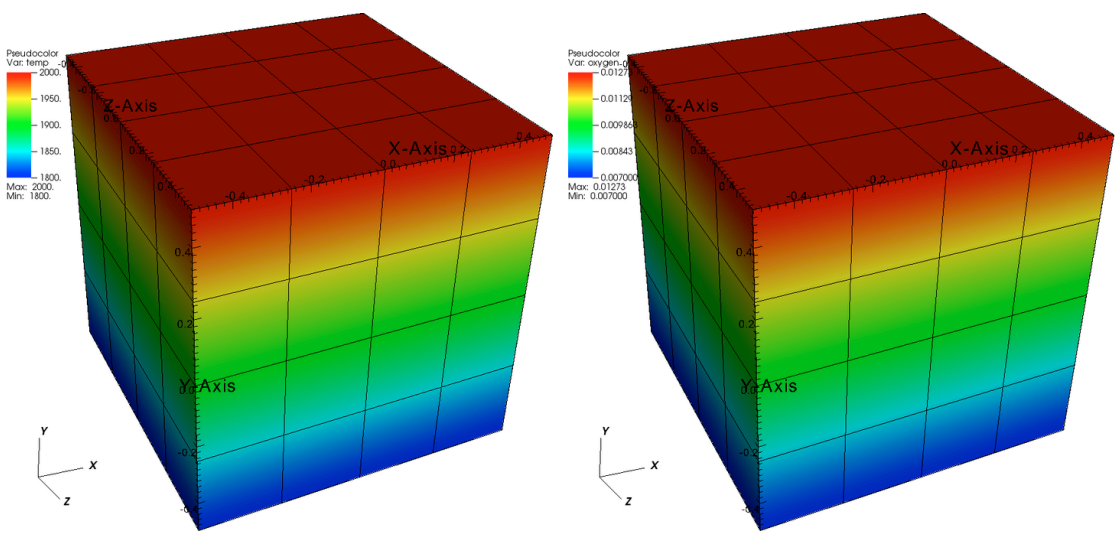


Figure 8. Temperature (left) and oxygen (right) distribution for a test problem with a temperature gradient in a unit cubic domain.

Definition of the object construction and initiation in MOOSE and BISON is fairly straightforward and is not shown in the above listing. The initial condition can be coupled to the temperature field. An added benefit of the framework is that the initial condition is valid for both regular and aux kernel variables.

4.4 Example Problem for Calculating Oxygen Thermodynamic Properties using BISON and THERMOCHIMICA Module

The main component of the implementation of the oxidation auxiliary kernel `OxygenThermochimicaAux.C` is shown in Listings 4 and 5. Listing 4 shows the constructor of the kernel. It couples temperature and provides for the calculation of oxygen potential and oxygen to metal ratio in the fuel fluorite phase.

```
template<>InputParameters validParams<OxygenThermochimicaAux>()
{
  InputParameters params = validParams<AuxKernel>();
  params.addCoupledVar("Pu", 0.0, "Pu element amount");
  params.addCoupledVar("U" , 0.0, "U element amount");
  // similarly for other elements
  params.addCoupledVar("O" , 0.0, "O element amount");
  params.addRequiredParam<int>("O2idx", "O2 index in species list");
  params.addRequiredParam<int>("O2iel", "O index in element list");
  params.addCoupledVar("temp", 1000, "Temperature");
  params.addParam<Real>("pressure" , 1.0, "Pressure");
  params.addParam<FileName>("thermofile", "THERMOCHIMICA.dat",
"Thermodynamics model file");
  params.addCoupledVar("o2m",0, "O/M aux variable to update");

  return params;
}
```

Listing 4. Constructor of `OxygenThermochimicaAux` auxiliary kernel.

The main auxiliary kernel evaluation steps are shown in Listing 5.

```
Real OxygenThermochimicaAux::computeValue()
{
  double Temperature,Pressure,dMol,O2M,dMolFraction,dIdealConstant,
    dO2PPressure;
  int iElement,idbg;
  char cPhaseIn[] = "O2ZRU_C";
  FORTRAN_CALL(Thermochimica::setthermofilename)(cThermoFileName);
  FORTRAN_CALL(Thermochimica::setstandardunits)();
  FORTRAN_CALL(Thermochimica::ssparsecsdatafile)();
  FORTRAN_CALL(Thermochimica::settemperaturepressure)
   (&Temperature, &Pressure);
  iElement=94;  dMol=_elPu[_qp];
  FORTRAN_CALL(Thermochimica::setelementmass)(&iElement, &dMol);
  // same for other elements
  // calculate thermochemical equilibrium
  FORTRAN_CALL(Thermochimica::thermochimica)();
  FORTRAN_CALL(Thermochimica::compotomratio)
    (cPhaseInFortran, &dO2M, &idbg);
  iElement=_o2idx; dMolFraction=0.0;
  FORTRAN_CALL(Thermochimica::getmolfraction)
    (&iElement, &dMolFraction, &idbg);
  _o2m[_qp] = dO2M;

  return dO2PPressure;
}
```

Listing 5. Aux kernel `OxygenThermochimicaAux` evaluation procedure.

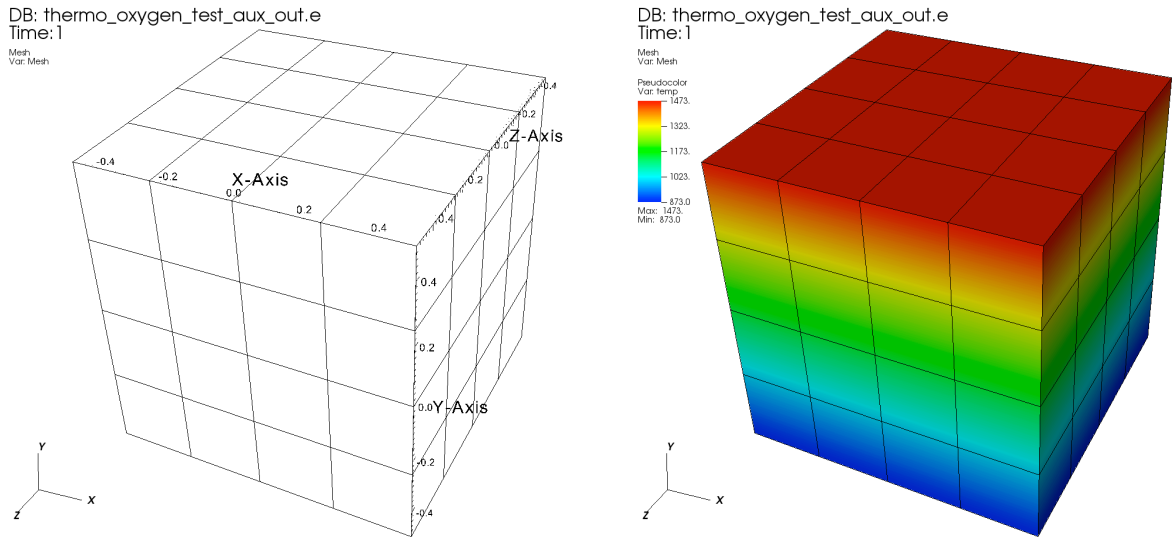
As can be seen, the oxygen potential denoted as `oxypp` is the main variable. This was imposed because the experimental data was in that form. The main variable may be changed during further development as we will concentrate on the modeling of oxygen diffusion in fuel. An example of the input for the `OxygenThermochimicaAux` is shown in Listing 6.

```
[AuxKernels]
 [./opp]
  type = OxygenThermochimicaAux
  variable = oxypp
  O = el_o
  Rb = el_rb
  Sr = el_sr
  Y = el_y
  Zr = el_zr
  Mo = el_mo
  Tc = el_tc
  Ru = el_ru
  Rh = el_rh
  Pd = el_pd
  Te = el_te
  I = el_i
  Xe = el_xe
  Cs = el_cs
  Ba = el_ba
  La = el_la
  Ce = el_ce
  Pr = el_pr
  Nd = el_nd
  U = el_u
  Pu = el_pu
  O2idx = 2
  O2iel = 21
  temp = temp
  pressure = 1.0
  thermofile = DBV8_TMB_modified.dat
  o2m = o2m
  execute_on = timestep
  [../]
[]
```

Listing 6. Example input for `OxygenThermochimicaAux` auxiliary kernel.

The parameters `O2idx` denote the index of O_2 in the species list and oxygen in the element list of the thermodynamic model. The thermodynamic model file is specified by the `thermofile` parameter. Parameter `o2m` denotes aux variable for the oxygen to metal ratio in the fluorite phase. Element concentrations are defined as separate auxiliary variables that may be provided by other BISON modules.

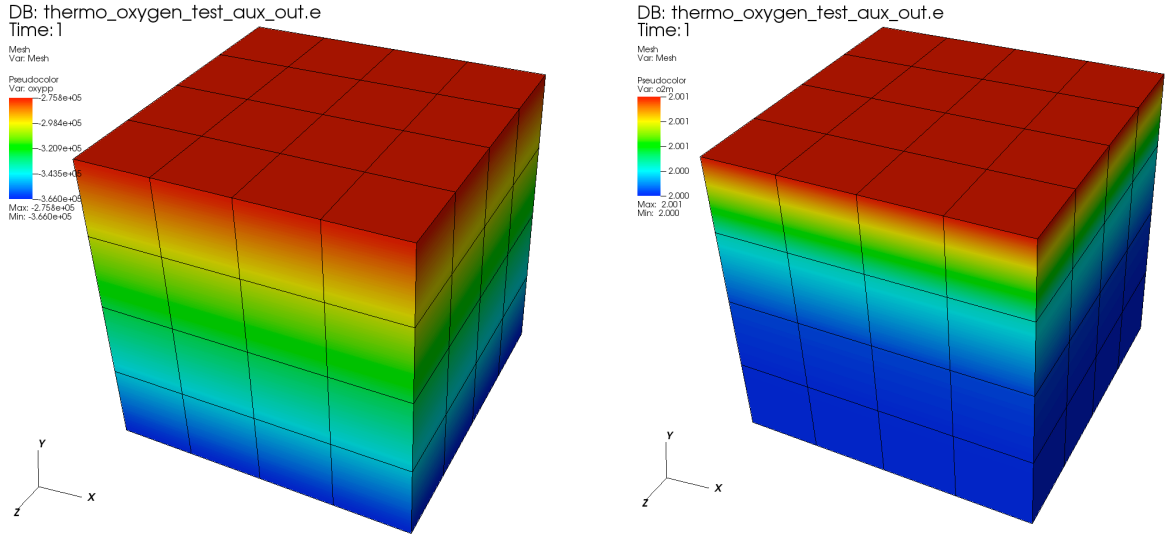
The accuracy of the developed functions and subroutines were validated using simple test and example problems available in BISON. We first show an oxidation state calculation on a simple cubic domain with dimensions given in Figure 9a. Thermal boundary conditions were applied that impose a temperature gradient on the thermal solution in the domain between 1473K and 873K as shown in Figure 9b.



(a) (b)

Figure 9. (a) Problem geometry, and (b) Temperature [K] distribution for the thermal solution

The elemental composition that was used to initiate the composition of the cubic domain corresponds to the calculated fission product inventory (Figure 4) for the outermost pellet ring in Figure 3, with the rings centerline at r/r_o=0.9. By imposing the temperature gradient between the bottom (873 K) and the top (1473 K) surface, the values of the oxygen potential across the vertical direction should recover values for at r/r_o=0.9 in Figure 6. The oxygen potential and oxygen to metal ratio in the fluorite phase computed in BISON by the `OxygenThermochimicaAux` auxiliary kernel are shown in Figures 10a and 10b, respectively.



(a) (b)

Figure 10. (a) Oxygen potential [J/mol] and (b) oxygen-to-metal ratio distribution

The two extreme values of the oxygen potential in Figure 10a match the values for the curve r/r_o=0.9

in Figure 5 for temperatures of 600 C (873 K) and 1200 C (1473 K), respectively, and thereby validate the BISON implementation of the oxidation state kernel.

Summary

The thermodynamic equilibrium solver library THERMOCHIMICA has been integrated into the MOOSE/BISON GitLab source repository system at INL. The thermodynamic solver provides information about the material composition that can be used for modeling constitutive properties and various phenomena of interest in nuclear fuel elements. The capabilities of the THERMOCHIMICA thermodynamic solver were demonstrated in simulating the composition of high burnup LWR nuclear fuel. The thermodynamics models and oxidation state computation modules implementation in MOOSE/BISON were used to simulate oxygen partial pressure values in the fuel and were compared to experiments. The implementation of the coupled oxygen transport was based on a standard approach from the literature and uses thermodynamic calculation for deriving initial conditions for the problem. The developed THERMOCHIMICA modules constitute a unique capability for modeling composition and spatial variation of compounds and phases in nuclear fuel during irradiation, and provide the framework for more accurate modeling of diffusion controlled processes in practical multicomponent systems. The incorporation of Thermochimica into the INL GitLab source repository enables its integration with other NEAMS codes and the MOOSE framework.

Acknowledgments

The authors would like to thank Markus H.A. Piro of the Canadian National Laboratory, Chalk River, Canada, for his technical assistance and continuing development of THERMOCHIMICA. The authors would like to thank James E. Banfield of GE Hitachi Nuclear Energy and Kevin T. Clarno of Oak Ridge National Laboratory for developing models for SCALE and running the simulations. The authors would also like to thank Michael J. Welland from Argonne National Laboratory, Argonne, USA for his technical comments. Research supported by the U.S. Department of Energy, Office of Nuclear Energy, Nuclear Energy Advanced Modeling and Simulation Program.

References

[1] T. M. Besmann, "Computational Thermodynamics: Application to Nuclear Materials," in *Comprehensive Nuclear Materials*. vol. 1, R. J. M. Konings, Ed., ed: Elsevier, 2012, pp. 455 - 470.

[2] M. H. A. Piro, J. Banfield, K. T. Clarno, S. Simunovic, T. M. Besmann, B. J. Lewis, *et al.*, "Coupled thermochemical, isotopic evolution and heat transfer simulations in highly irradiated UO2 nuclear fuel," *Journal of Nuclear Materials,* vol. 441, pp. 240-251, Oct 2013.

[3] D. Staicu, "Thermal Properties of Irradiated UO2 and MOX," in *Comprehensive Nuclear Materials*. vol. 2, R. J. M. Konings, Ed., "" ed: Elsevier, 2012, pp. 439 - 464.

[4] B. J. Lewis, W. T. Thompson, and F. C. Iglesias, "Fission Product Chemistry in Oxide Fuels," in *Comprehensive Nuclear Materials*. vol. 2, R. J. M. Konings, Ed., "" ed: Elsevier, 2012, pp. 515 - 546.
[5] M. H. A. Piro, S. Simunovic, T. M. Besmann, B. J. Lewis, and W. T. Thompson, "The thermochemistry library Thermochimica," *Computational Materials Science,* vol. 67, pp. 266-272, Feb 2013.
[6] Idaho National Laboratory. https://hpcgitlab.inl.gov.
[7] D. Gaston, C. Newman, G. Hansen, and D. Lebrun-Grandie, "MOOSE: A parallel computational framework for coupled systems of nonlinear equations," *Nuclear Engineering and Design,* vol. 239, pp. 1768-1778, Oct 2009.
[8] J. D. Hales, S. R. Novascone, R. L. Williamson, D. R. Gaston, and M. R. Tonks, "Solving Nonlinear Solid Mechanics Problems with the Jacobian-Free Newton Krylov Method," *Cmes-Computer Modeling in Engineering & Sciences,* vol. 84, pp. 123-153, Mar 2012.
[9] S. Novascone, B. Spencer, D. Andrs, R. Williamson, J. Hales, and D. Perez, "Results from Tight and Loose Coupled Multiphysics in Nuclear Fuels Performance Simulations using BISON," presented at the International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering, Sun Valley, Idaho, 2013.
[10] J. D. Hales, S. R. Novascone, B. W. Spencer, R. L. Williamson, G. Pastore, and D. M. Perez, "Verification of the BISON fuel performance code," *Annals of Nuclear Energy,* vol. 71, pp. 81-90, Sep 2014.
[11] C. T. Walker, V. V. Rondinella, D. Papaioannou, S. Van Winckel, W. Goll, and R. Manzel, "On the oxidation state Of UO2 nuclear fuel at a burn-up of around 100 MWd/kgHM," *Journal of Nuclear Materials,* vol. 345, pp. 192-205, Oct 2005.
[12] I. Gauld, O. Hermann, and R. Westfall, "ORIGEN Scale System Module to Calculate Fuel Depletion, Actinide Transmutation, Fission Product Buildup and Decay, and Associated Radiation Terms," 2005.
[13] K. Lassmann, C. Ocarroll, J. Vandelaar, and C. T. Walker, "The radial-distribution of plutonium in high burnup UO2 fuels," *Journal of Nuclear Materials,* vol. 208, pp. 223-231, Feb 1994.
[14] M. H. A. Piro, S. Simunovic, and T. M. Besmann, "Thermochimica User Manual v1.0," Oak Ridge National Laboratory, Oak Ridge2012.
[15] M. H. A. Piro, T. M. Besmann, S. Simunovic, B. J. Lewis, and W. T. Thompson, "Numerical verification of equilibrium thermodynamic computations in nuclear fuel performance codes," *Journal of Nuclear Materials,* vol. 414, pp. 399-407, Jul 2011.
[16] M. H. A. Piro and S. Simunovic, "Performance enhancing algorithms for computing thermodynamic equilibria," *Calphad-Computer Coupling of Phase Diagrams and Thermochemistry,* vol. 39, pp. 104-110, Dec 2012.
[17] D. van Heesch. (2015). *Doxygen Manual*. Available: http://www.doxygen.org
[18] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in Fortran 90: The Art of Parallel Scientific Computing, Volume 2*: Cambridge University Press, 1996.
[19] C. Bale, P. Chartrand, S. A. Degterov, G. Eriksson, K. Hack, R. Ben Mahfoud*, et al.*, "FactSage thermochemical software and databases," *Calphad-Computer Coupling of Phase Diagrams and Thermochemistry,* vol. 26, pp. 189-228, Jun 2002.

[20] C. W. Bale, E. Belisle, P. Chartrand, S. A. Decterov, G. Eriksson, K. Hack*, et al.*, "FactSage thermochemical software and databases - recent developments," *Calphad-Computer Coupling of Phase Diagrams and Thermochemistry,* vol. 33, pp. 295-311, Jun 2009.

[21] B. Mihaila, M. Stan, J. Ramirez, A. Zubelewicz, and P. Cristea, "Simulations of coupled heat transport, oxygen diffusion, and thermal expansion in UO2 nuclear fuel elements," *Journal of Nuclear Materials,* vol. 394, pp. 182-189, Nov 2009.

[22] B. Mihaila, M. Stan, and J. Crapps, "Impact of thermal conductivity models on the coupling of heat transport and oxygen diffusion in UO2 nuclear fuel elements," *Journal of Nuclear Materials,* vol. 430, pp. 221-228, Nov 2012.

[23] C. Gueneau, A. Chartier, and L. Van Brutzel, "Thermodynamic and Thermophysical Properties of the Actinide Oxides," in *Comprehensive Nuclear Materials*. vol. 2, R. J. M. Konings, Ed., ed: Elsevier, 2012 pp. 21 - 59.

[24] J. C. Ramirez, M. Stan, and P. Cristea, "Simulations of heat and oxygen diffusion in UO2 nuclear fuel rods," *Journal of Nuclear Materials,* vol. 359, pp. 174-184, Dec 2006.