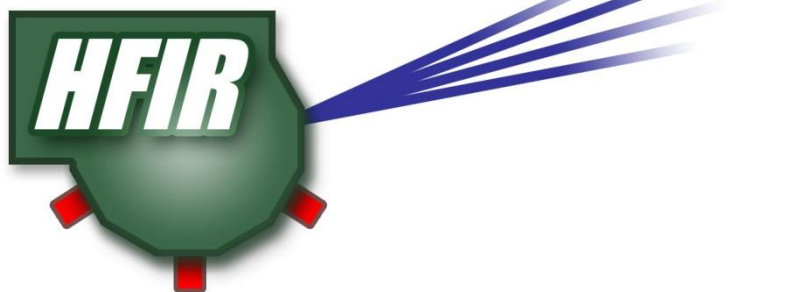OAK RIDGE
NATIONAL LABORATORY

MANAGED BY UT-BATTELLE
FOR THE DEPARTMENT OF ENERGY

Two-Dimensional Thermal Hydraulic Analysis and Benchmark in Support of HFIR LEU Conversion using COMSOL

September 2010

J. D. Freels
I. T. Bodey
R. V. Arimilli
K. T. Lowe

UT-BATTELLE

ORNL-27 (4-00)

TWO DIMENSIONAL THERMAL HYDRAULIC ANALYSIS AND BENCHMARK IN SUPPORT OF HFIR LEU CONVERSION USING COMSOL

J. D. Freels
I. T. Bodey*
R. V. Arimilli*
K. T. Lowe†

*University of Tennessee, Knoxville, TN
†Bettis Atomic Power Laboratory, West Mifflin, PA

September 2010

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (cont'd)

NOMENCLATURE

English Letters

A	Cross Sectional Area, m^2
C_p	Isobaric Specific Heat Capacity, J/kg/K
C_μ	Turbulent Viscosity Coefficient
CV	Control Volume
D_h	Hydraulic Diameter of Involute Flow Channel, m
E_{enter}	System Inlet Energy, W
E_{exit}	System Outlet Energy, W
E_{gen}	Internally Generated Thermal Energy, W
E_{st}	Energy Stored in the System, W
F	Body Force, N/m^3
f	Fraction of the Reactor Power Deposited In the Fuel Elements
$\mathbf{I}$	Identity Tensor
$\mathbf{k}$	Thermal Conductivity Tensor, W/m/K
k	Thermal Conductivity Scalar, W/m/K
k_{clad}	Thermal Conductivity of the Clad, W/m/K
k_{fluid}	Thermal Conductivity of the fluid, W/m/K
k_{fuel}	Thermal Conductivity of the Fuel, W/m/K
k_T	Turbulent Thermal Conductivity, W/m/K
k_{turb}	Turbulent Kinetic Energy, m/s^2
p	Thermodynamic Pressure, Pa
Pr_T	Turbulent Prandtl Number
Q	Total Power of the Core, MW
q	Thermal Power, W
q''	Heat Flux, W/m^2
q'''	Power Density or Thermal Energy Generation Rate, W/m^3
q_{plate}	Power Dissipated per Plate, W
q_s	Production/Absorption Coefficient, $W/m^3/K$
R_b	Base Radius of the Involute Origin, m
s	Arc Length, m
T	Transpose of a Matrix
T	Temperature, K
T_{bulk}	Bulk Water Temperature, K
T_f	Fluid Temperature, K
T_s	Clad Surface Temperature, K
u	Fluid Velocity Magnitude, m/s
V	Volume, m^3
$V_{i,j}$	Control Volume Element, m^3
V_{clad}	Volume of the clad, m^3
V_{fuel}	Volume of the Fuel, m^3
z	Axial Position, m

Greek Letters

ε	Dissipation of Turbulent Kinetic Energy
η	Fluid Viscosity, Pa-s
η_T	Turbulent Viscosity, Pa-s
θ	Angle Subtended Between Radial Nodes
ρ	Density, kg/m^3
φ'	Unmodified Multiplier Array
φ	Modified Multiplier Array

ABSTRACT

The research documented herein was funded by a research contract between the Research Reactors Division (RRD) of Oak Ridge National Laboratory (ORNL) and the University of Tennessee, Knoxville (UTK) Mechanical, Aerospace and Biomedical Engineering Department (MABE). The research was governed by a statement of work (SOW) which clearly defines nine specific tasks. This report is outlined to follow and document the results of each of these nine specific tasks. The primary goal of this phase of the research is to demonstrate, through verification and validation methods, that COMSOL is a viable simulation tool for thermal-hydraulic modeling of the High Flux Isotope Reactor (HFIR) core. A secondary goal of this two-dimensional phase of the research is to establish methodology and data base libraries that are also needed in the full three-dimensional COMSOL simulation to follow. COMSOL version 3.5a was used for all of the models presented throughout this report.

1. VERIFICATION OF EXISTING COMSOL MODELS FROM PRIOR RESEARCH

The first item to be addressed in the SOW was to verify that the model files submitted by The University of Tennessee's Nuclear Engineering Department (UTNE) will reproduce the results found in ORNL/TM-2008/188 (TM). Two sets of models from UTNE were analyzed. The first set of models were delivered with the TM at the time it was published. Upon verification testing of these models it was found that the current (1st set) model results did not match the results presented in the TM. Subsequently, after requesting from UTNE why there was a discrepancy, a new set of models (2nd set) were provided that matched the results in the TM. Nevertheless, even with this second set of models, there were still verification issues with the re-execution of nine of these models. Upon re-execution of these nine models, error messages occurred related to issues with the thermal variables used in the boundary conditions. Table 1.1 below provides a list of the equivocal files.

Table 1.1. Problem files present in ORNL/TM-2008/188

File Name as Submitted by UTNE	Re-execution Issue
kipsitest12v1.mph	Will not re-execute as delivered due to thermal variable evaluation failure.
PowerProfilev1.mph	Will not re-execute as delivered due to thermal variable evaluation failure.
3dsolution_random_meshv1.mph	Will not re-execute as delivered due to thermal variable evaluation failure.
3D_mapped_mesh_v1.mph	Failure to evaluate thermal variables
hotstripe_3Da.mph	Will not re-execute as delivered due to thermal variable evaluation failure.
hotspot_3D_v2a.mph	Will not re-execute as delivered due to thermal variable evaluation failure.
004thickhotspot_2Dv1.mph	Results are different after re-execution
3sidedflowchannel_3D_3v1.mph	All domains are not meshed. Will not re-execute.
kepsitest10v1.mph	Will not re-execute as delivered

One reason for the failure of these models to re-execute could be due to the use of different versions of COMSOL in the creation of the files (v3.4) and that used by MABE in the re-execution process (v3.5a). Another possible reason could be due to a redundant set of equations used in the setup of the models. Upon evaluation of the models in Section 4 of ORNL/TM-2008/188, it was discovered that the application modes being used were redundant and that the boundary conditions were incorrect.

The application modes chosen for these models are the Weakly Compressible Navier-Stokes equation set, the Convection and Conduction equation set, and the General Heat Transfer module. The redundancy occurs with the use of the Convection and Conduction equation set in conjunction with the General Heat Transfer module. The General Heat Transfer module is inherently a conduction analysis equation set in which the user has the ability to initialize convection. The General Heat Transfer governing equation has the form

$$\nabla \cdot (-k\nabla T) = q''' + q_s T - \rho C_p \vec{u} \cdot \nabla T \qquad (1.1)$$

where k is the thermal conductivity of the material, T is the temperature field, q''' is the power density, q_s is the production/ absorption coefficient, ρ is the fluid density, C_p is the specific heat at constant pressure, and u is the fluid velocity. The use of the Weakly Compressible Navier-Stokes equation set is incorrect in that it does not model the turbulent aspect of the flow.

The Reynolds number for the flow is defined as

$$Re_{D_h} = \frac{\rho u D_h}{\eta} \tag{1.2}$$

where ρ = 980.28 kg/m^3 [2] is the density of the fluid, u = 15.88 m/s [4] is the velocity of the flow in the coolant channel, D_h = 0.0025 m is the hydraulic diameter, and η = 0.5693×10^{-3} Pa*s [7] is the dynamic viscosity. With these values Re_{Dh} = 68359, which is in the turbulent regime for enclosed channel flow. Furthermore, the use of a constant density, as reported in ORNL/TM-2008/188, negates the characteristic that distinguishes the Weakly-Compressible set from the Incompressible Navier-Stokes equation set.

2. VERIFICATION OF SSHTC INPUTS

Four sources are required to verify the inputs given in the models appearing in Section 4 of the ORNL/TM-2008/188 (hereafter referred to as the "TM"): (1) The materials referenced in the RELAP5 input report [5]. (2) The Steady-State Heat Transfer Code (SSHTC)[4]. (3-4) Two differing sets of data files provided by the UTK Nuclear Engineering Department (UTNE) as part of the deliverables with the TM (FY 2008 research on this topic). The UTNE models were submitted at two separate times. The initial models did not match those reported in the TM, while the second set of models did match the reported results.

The tables below condense the material properties used in all four model sets. Table 2.1 shows that the original files submitted by UTNE were more consistent with input related to the SSHTC. This observation is based on the thermal conductivity values for the clad and fuel. Some properties used in the SSHTC were not used in the initial studies with COMSOL. Likewise, some COMSOL models have more properties specified than the SSHTC code. Water is listed twice because some models used constant properties and others used variable properties. Reporting of these values does not represent validation of the property's accuracy. This information only reflects that which is present in the TM model setup. Empty entries in Table 2.1 indicate values that have not been independently verified.

Table 2.1. Comparison of SSHTC properties to UTNE model parameters[1]

Material & Properties	Units	UTNE Set 1	SSHTC
Clad			
k	W/m/K	167.9	167.88
C_p	J/kg/K	888	
ρ	kg/m^3	2700	
Fuel			
k	W/m/K	46.9	
C_p	J/kg/K	266	
ρ	kg/m^3	19000	
Water			
k	W/m/K	COMSOL function	1.73073*(-0.55178+0.0056853*T-1.06554e-5*T^2+5.94048e-9*T^3)
C_p	J/kg/K	COMSOL function	1289.28+24.9397*T-0.073528*T^2+7.4278e-5*T^3
ρ	kg/m^3	COMSOL function	895.26+1.04524*T-.0023485*T^2
η	Pa*s	COMSOL function	.000413379*(352*(9/5*T-459.688)^-1.162)
Beohmite			
k	W/m/K		2.25
ρ	kg/m^3		3020
Water Constant			
k	W/m/K	0.668	
C_p	J/kg/K	4185	
ρ	kg/m^3	973	
η	Pa*s	3.65E-04	
Segregated Fuel			
k	W/m/K		13
Inner Fuel			
k	W/m/K		132.06
Outer Fuel			
k	W/m/K		110.07
U_3O_8 Segregation			
k	W/m/K		46.9
Poison			
k	W/m/K	151.1	

---

[1] T^2 should be interpreted as T^2; that is the symbol "^" means "raised to the power of"

The values reported in Table 2.2 for the RELAP5 HFIR model are portions of the data set for that particular property. The portions presented are used to show the similarity between RELAP5 and the UTNE COMSOL models. This table is not meant to be a complete summary of the values used in the RELAP5 HFIR model. Table 2.2 shows that the material properties used in the updated UTNE models, which are the models used in the TM report, compare closely to the properties used in the RELAP5 HFIR model. This conclusion is based primarily from clad properties and comparing UTNE fuel properties with SSHTC inner fuel properties. The properties of water used in the RELAP5 HFIR models are not explicitly documented in the HFIR model description, but are generically available to the RELAP5 code upon code installation and have been extensively validated in a long history of research. A review and tabulation of this RELAP5 internal library is beyond the scope of this research. It should be noted however, that the interface of this RELAP5 library to the COMSOL material library is certainly possible if such an interface is desirable in the future for consistency of the data libraries. In addition, COMSOL functions for water as temperature dependent polynomial expressions are shown in Table 2.3 at atmospheric pressure. Empty entries in Table 2.2 indicate values that have not been independently verified or are obtained from another source (internally-stored water properties in RELAP5 for example).

Table 2.2. Comparison of updated UTNE model properties with RELAP5 inputs

Material & Properties	Units	UTNE Set 2	HFIR RELAP5 model
Clad			
k	$W/m/K$	181.3	181.3 @ 301 K
C_p	$J/kg/K$	888	900.16 @ 300 K
ρ	kg/m^3	2700	2710
Fuel			
k	$W/m/K$	176.95	
C_p	$J/kg/K$	266	
ρ	kg/m^3	19000	
Water			
k	$W/m/K$	COMSOL function	
C_p	$J/kg/K$	COMSOL function	
ρ	kg/m^3	COMSOL function	16.018*(62.455+6.8649e-3*(9/5*T-460)-1.2013e-4*(9/5*T-460)^2)
η	$Pa*s$	COMSOL function	
Beohmite			
k	$W/m/K$		2.25
ρ	kg/m^3		3020
C_p	$J/kg/K$		795.5 @ 300K
Water Constant			
k	$W/m/K$	0.65414 @ 333K	
C_p	$J/kg/K$	4185 @ 333K	
ρ	kg/m^3	982.91 @ 333K	
η	$Pa*s$	0.00046416@333K	
Inner Fuel			
k	$W/m/K$		177.2 @ 300 K
C_p	$J/kg/K$		762 @ 300K
ρ	kg/m^3		2821
Outer Fuel			
k	$W/m/K$		159.07 @300K
C_p	$J/kg/K$		724.32@ 300K
ρ	kg/m^3		2984
Poison			
k	$W/m/K$	151.1	

Table 2.3. Internal COMSOL temperature dependent water properties

Material & Properties	Units	COMSOL Functions	Temperature Range
Water			
k	*W/m/K*	$-0.869083936 + 0.00894880345\,T - 1.583363\text{E-}5\,T^2 + 7.97542359\text{E-}9\,T^3$	273.15 – 553.75
C_p	*J/kg/K*	$1.20101471\text{E}4 - 80.4072879\,T + 3.09866854\text{E-}1\,T^2 - 5.38186884\text{E-}4\,T^3 + 3.62536437\text{E-}7\,T^4$	273.15 – 553.75
ρ	*kg/m^3*	$8.38466135\text{E}2 + 1.40050603\,T - 3.01123760\text{E-}3\,T^2 + 3.71822313\text{E-}7\,T^3$	273.15 – 553.75
η	*Pa*s*	$1.3799566804 - 2.1224019151\text{E-}2\,T + 1.3604562827\text{E-}4\,T^2 - 4.6454090319\text{E-}7\,T^3 + 8.9042735735\text{E-}10\,T^4 - 9.0790692686\text{E-}13\,T^5 + 3.8457331488\text{E-}16\,T^6$	273.15 – 413.15
η	*Pa*s*	$4.01235783\text{E-}3 - 2.10746715\text{E-}5\,T + 3.85772275\text{E-}8\,T^2 - 2.39730284\text{E-}11\,T^3$	413.15 – 553.75

3. COMPARISON OF COMSOL WITH AN ANALYTIC SOLUTION

Two problems were chosen as simple validation models with which to determine the accuracy of COMSOL: 1) a one dimensional, steady-state, constant property, thermal conduction problem and 2) a two dimensional, steady-state, constant property, thermal conduction problem. Problem 1 lends itself to an analytical solution which is used as the base case for the comparison calculations with the COMSOL model. Problem 2 uses a COMSOL mesh independent solution (i.e., results do not change with increased mesh density) as the base case for comparison.

Problem 1:

Figure 3.1 below shows the physical setup for problem 1 and problem 2. The difference between problems 1 and 2 lies in the conditions on the boundaries normal to the x-direction. All of the properties listed in Fig. 3.1 were found in the TM. The fuel and cladding are assumed to be in perfect thermal contact. $y = 0$ is the symmetry axis for the fuel and is represented as an adiabatic boundary due to the volumetric generation term in the fuel. For problem 1, no thermal gradients exist in the x direction in either the fuel or cladding. The y_3 boundary is isothermal. As shown in Fig. 3.1, q''' is the power density, k is the thermal conductivity of the material, and q'' is the interfacial heat flux.
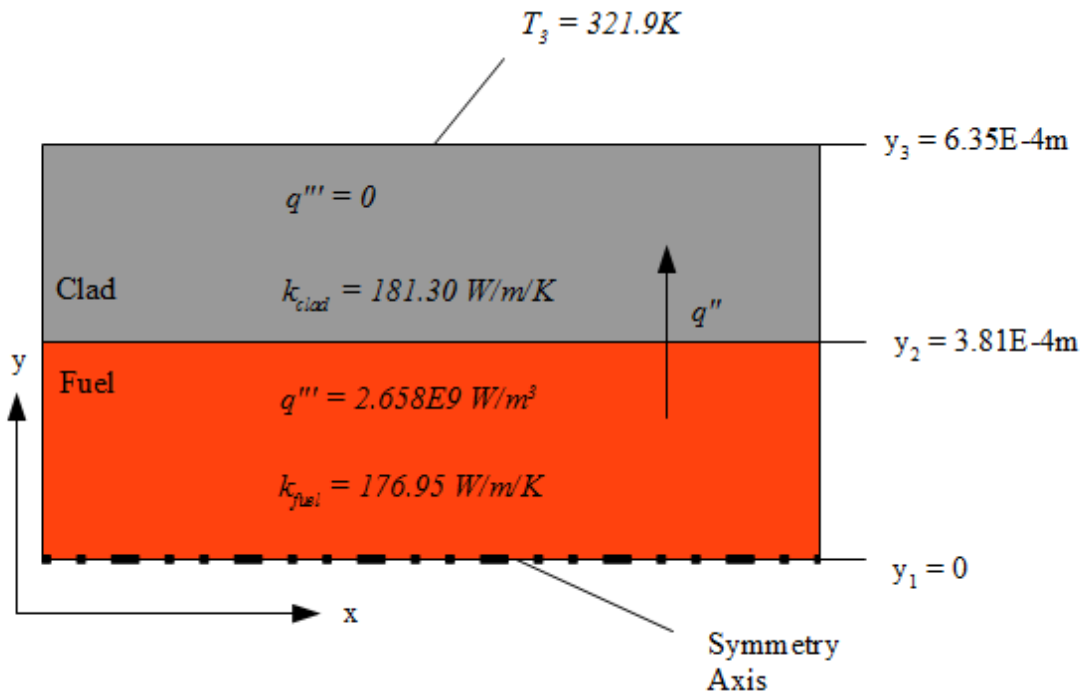
Fig. 3.1. Schematic representation of simple validation problem for COMSOL typical of HFIR fuel heat plate conduction.

3.1 ANALYTICAL FORMULATION AND SOLUTION

The physical setup depicted in Fig. 3.1 is represented mathematically by the thermal energy diffusion equation

$$\frac{d}{dx}\left(-k\frac{dT}{dx}\right) = q''' \tag{3.1}$$

where k is the thermal conductivity of the material, T is the temperature field and q''' is the power density. Assuming constant properties, Equation 3.1 may be simplified to

$$k\frac{d^2T}{dx^2} = -q''' \tag{3.2}$$

which is characterized as Poisson's equation in one dimension with a thermal source. The solution of Equation 3.2 is parabolic with the source term being the coefficient of the quadratic term in the solution and the remaining coefficients being determined by both the source term and the boundary conditions for that domain. Equation 3.2 is the governing equation for both the fuel domain and the clad domain; however, the governing equation for the clad may be simplified even further to the homogeneous case due to the lack of a thermal source in that domain. The governing equation for the clad domain is

$$\frac{d^2T}{dx^2} = 0 \tag{3.3}$$

which is characterized as Laplace's equation in one dimension. The solution to Laplace's equation is a line whose coefficients are determined solely by the boundary conditions for that domain.
The complete problem formulation for the temperature field in the clad is

$$\frac{d^2T}{dx^2} = 0$$

$$\frac{dT}{dx}(y_2) = \frac{-q}{k_{clad}A} \tag{3.4}$$

$$T(y_3) = 321.9K$$

where A is the cross sectional area of the interface between the clad and the fuel. The complete formulation for the temperature field in the fuel is

$$\frac{d^2T}{dx^2} = \frac{-q}{k_{fuel}V_{fuel}}$$

$$\frac{dT}{dx}(y_1) = 0 \tag{3.5}$$

$$T(y_2) = T_2$$

Given the formulation of equation sets 3.4 and 3.5, the temperature field is

$$T(y) = \frac{q}{2k_{fuel}V_{fuel}}(y_2^2 - y^2) + T_2 \qquad \text{for } \{y_1 \leq y < y_2\} \tag{3.6}$$

and

$$T(y) = \frac{qy_2}{k_{clad}V_{clad}}(y_3 - y) + T_3 \qquad \text{for } \{y_2 \leq y \leq y_3\}. \tag{3.7}$$

From Equations 3.6 and 3.7, the adiabatic boundary temperature and the interfacial temperature may be determined by using values of parameters shown in Fig. 3.1, and are calculated to be 324.41 K and 323.32 K respectively.

3.2   COMSOL MODEL RESULTS

The geometry shown in Fig. 3.1 above was created in COMSOL and all of the properties found in the physical setup were input into the model. The number of mesh elements was varied from an extremely coarse mesh structure consisting of two elements to a relatively fine mesh consisting of 376,000 elements. The meshing procedure began with a mapped mesh (MM) consisting of two elements. The number of elements was then doubled in the MM scheme until a subsequent MM refinement would exceed the default extremely coarse free mesh element number. At that point the meshing scheme was switched to free mesh for the remainder of the study.

The temperature field calculated by COMSOL is shown in Fig. 3.2.
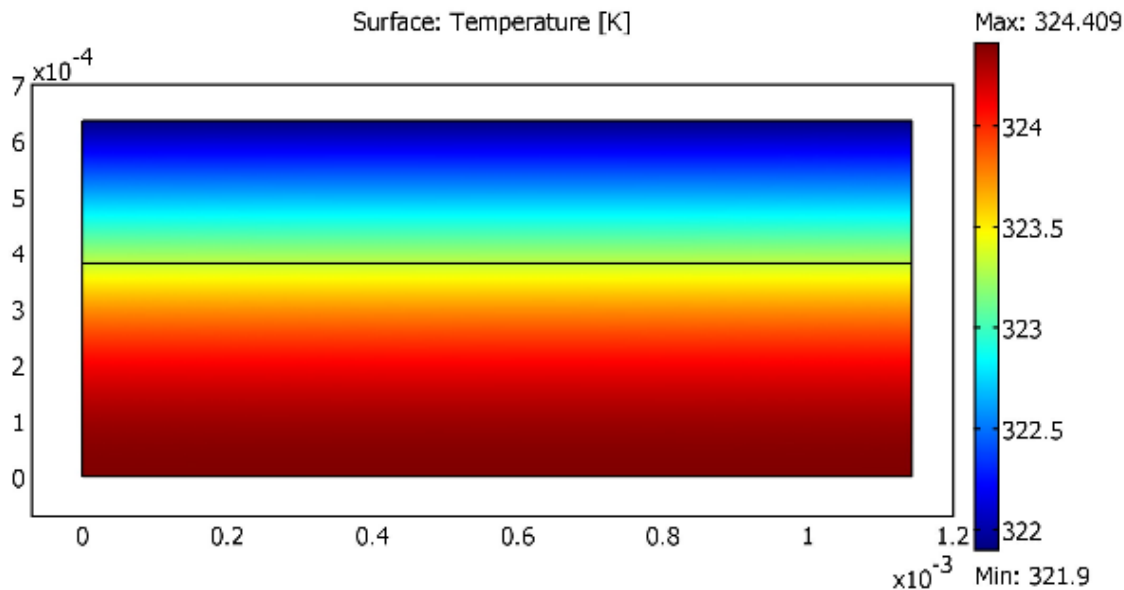
Fig. 3.2. COMSOL solution for equation sets 3.4 and 3.5.

The reader is able to verify in Fig. 3.2 that the adiabatic boundary temperature is 324.409 K. Using the boundary integration feature of COMSOL, the interfacial temperature was found to be 323.319 K. Therefore, the relative error in the results provided by COMSOL and the analytical solution was within 3.1×10^{-4} %.

In an effort to determine the minimum number of elements that provides a reliable finite-element method (FEM) solution, the number of mesh elements was varied and solutions were found for each mesh structure. In tandem with the variation in the number of mesh elements, the type of element was also varied. Three different types of Lagrange elements were considered: (1) Linear, (2) Quadratic, and (3) Cubic. One different feature between these element types is in the number of solution nodes per element. For the linear elements, each vertex of the element represents a node. The quadratic elements contain intermediate nodes between the vertex nodes of the element and the cubic elements contain still more intermediate nodes. Fig. 3.3 below shows an example of a triangular mesh structure consisting of 17 elements. For each mesh structure chosen, three different solutions are obtained by varying the element type as described above.
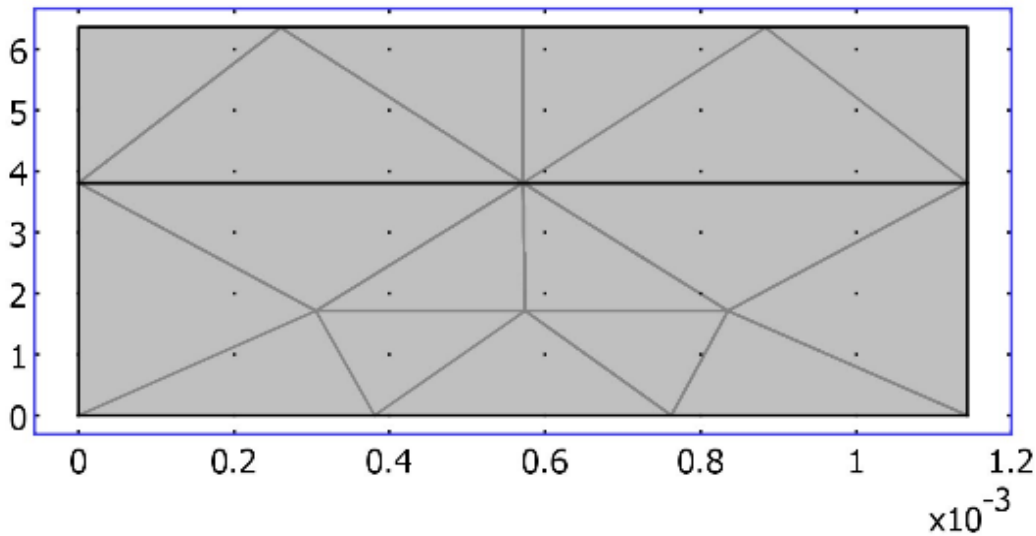
Fig. 3.3. Mesh structure consisting of 17 elements.

An energy norm calculation was conducted using the quantity "gradT_htgh" from the COMSOL post-processing menu which may be written as

$$\sqrt{\vec{\nabla}T \cdot \vec{\nabla}T} \tag{3.8}$$

Since the problem considered here is essentially 1-D, the energy norm computed is $\approx \frac{dT}{dy}$ and is "a suitable quantity for computing the energy norm" per ref. 3. The square of this quantity is integrated over the entire subdomain. This result was then compared with the same computed quantity from the analytical solution. A plot of this result is presented as an "energy norm" in Fig. 3.4 below. The reader may verify from Fig. 3.4 that the error in both the quadratic and cubic FEM solutions have converged to a constant value, nearly zero in this case, with a minimum amount of elements while the linear type solution requires a relatively large number of elements, approximately 10,000, to achieve the same error.

It was not initially obvious why these results were obtained. Because the analytical solution yields a

quadratic and linear temperature profile in y, Equations 3.6 and 3.7 respectively, both the quadratic and cubic finite-element interpolants used in COMSOL yield the exact solution to this problem giving the observed zero error. Therefore, in order to demonstrate the quadratic and cubic convergence rate, a problem with a more complex solution must be solved.
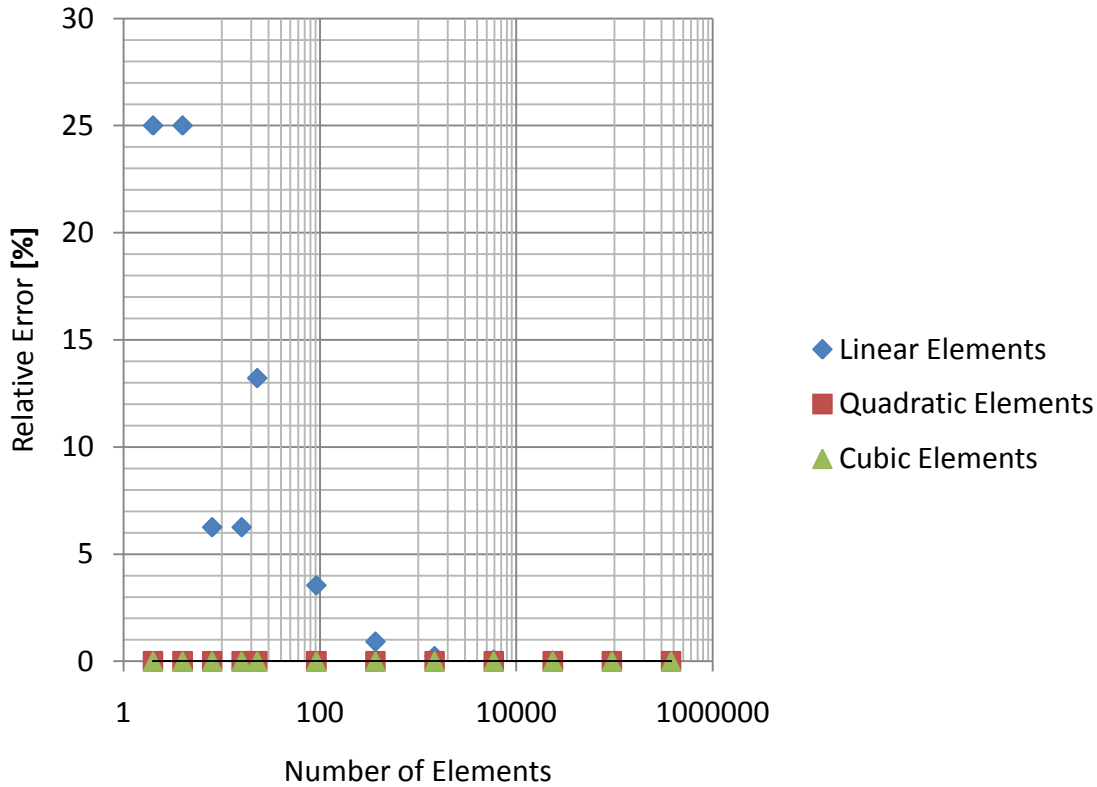
Fig. 3.4. Problem 1: Relative error in the energy norm as a function of element number.

Problem 2:

Problem 2 has the same physical setup as shown in Fig. 3.1, with the exception of isothermal conditions, 300 K, at the boundaries normal to the x-direction. With these boundary conditions the problem becomes two-dimensional. The analysis was performed using the same mesh structures as the 1D problem. The same procedure was followed as described in the problem 1 development above (i.e., the element type was used as a parameter for each generated mesh). No analytical solution was established for this problem. Therefore, the relative error for all element types, in Fig. 3.5, is calculated using a mesh independent solution as the "correct" solution. The results of this analysis are shown in Fig. 3.5.
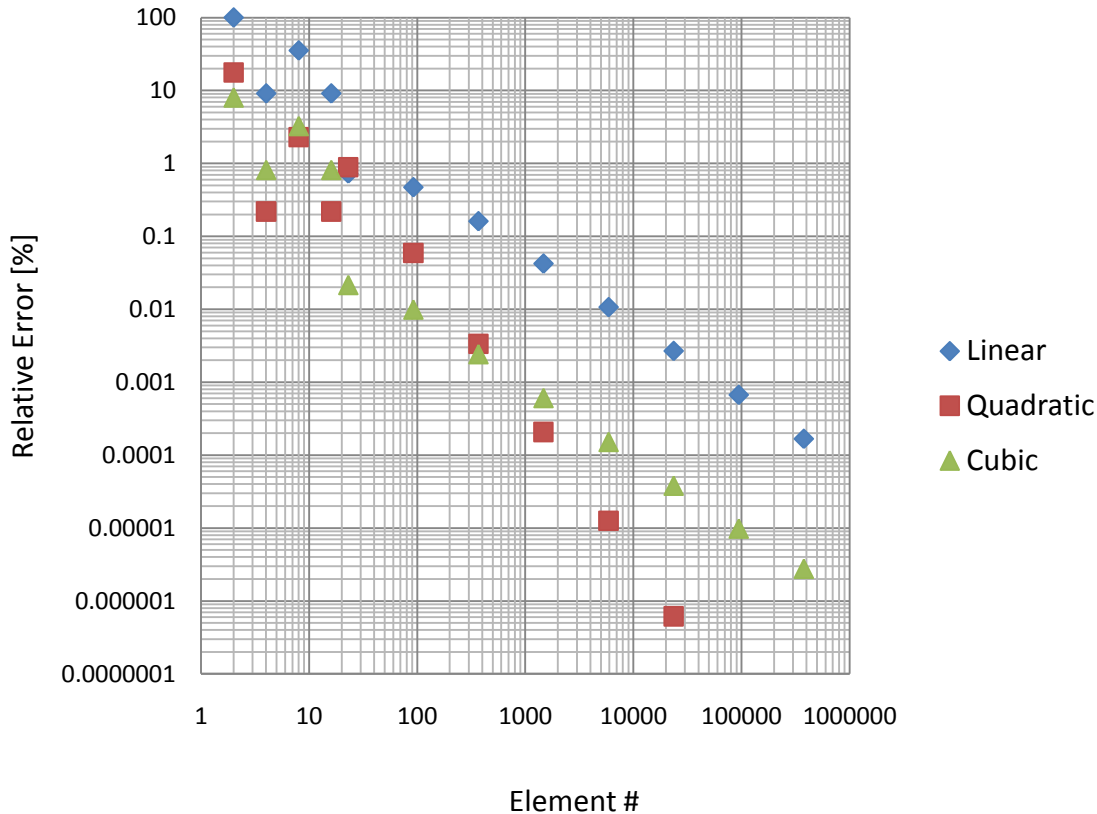
Fig. 3.5. Problem 2: Relative error in the energy norm as a function of element number.

For both the quadratic and cubic element convergence curves, the last point (near zero error) is not plotted since it is assumed equal to the actual (truth) solution and zero cannot be plotted on a logarithmic axis.

This analysis shows that for this linear problem, the COMSOL solution to equation sets 3.4 and 3.5 are mesh independent for both the quadratic and cubic element types over the range of elements chosen for the solution. Further, the COMSOL user may select a level of accuracy for heat conduction problems of this type without unnecessary mesh density and computer resources. The COMSOL user should also keep in mind that using higher-order elements (quadratic and cubic in this case, but even higher-order elements are available) will indeed increase the accuracy for a smaller number of finite elements. However, the tradeoff for this increased accuracy is a larger bandwidth in the solution matrix which, in turn, also drives up the computer resources required to obtain a solution. Therefore, there is a delicate balance in choosing element order, number of elements, computer resources, and desired solution accuracy that is found by the experienced COMSOL analyst. The COMSOL code defaults to quadratic elements for all the models that are to be used in this research. As the number of elements increases, this level of FEM interpolation is not necessary, and it is cost effective to reduce the finite-element order down to a linear basis to reduce computer resources while maintaining solution accuracy. Alternatively, it is often advantageous to maintain a mesh design without going through additional efforts to re-mesh to increase accuracy. The COMSOL analyst has the distinct advantage to simply increase the finite-element interpolation order (example: from linear to quadratic, or quadratic to cubic) and then automatically obtain a dramatic increase in solution accuracy without changing the mesh. This is an extremely powerful feature of COMSOL that is only found in a truly finite-element based simulation code.

**4. COMPARISON OF SSHTC FLUID PROPERTIES WITH THE ASME STEAM TABLES
AND DEVELOPMENT OF TEMPERATURE AND PRESSURE-DEPENDENT
WATER TABLES FOR USE AS A COMSOL LIBRARY**

For this task, a computer library package[10] obtained from the National Institute of Standards and
Technology (NIST) which contains the water properties formerly referred to as the American Society of
Mechanical Engineers (ASME) Steam Tables. Originally published as the ASME steam tables, the
package is now released by NIST. Both the present-day NIST and formerly-known-as ASME tables
reference the International Association for the Properties of Water and Steam (IAPWS) formulations. For
comparison, a mean pressure of 2.9680 MPa (430.6 psia) is chosen from the NIST tables. This pressure is
based on the core inlet pressure and typical pressure drop across the core reported in RELAP5. Figure 4.1
shows the variance in thermal conductivity, specific heat, and density between the SSHTC and NIST
properties. The temperature range is from 0 °C to the boiling point. All SSHTC properties are within 2%
of the NIST properties. In the range of temperatures reported in the SSHTC, the deviation from NIST is
less than ±0.5%.
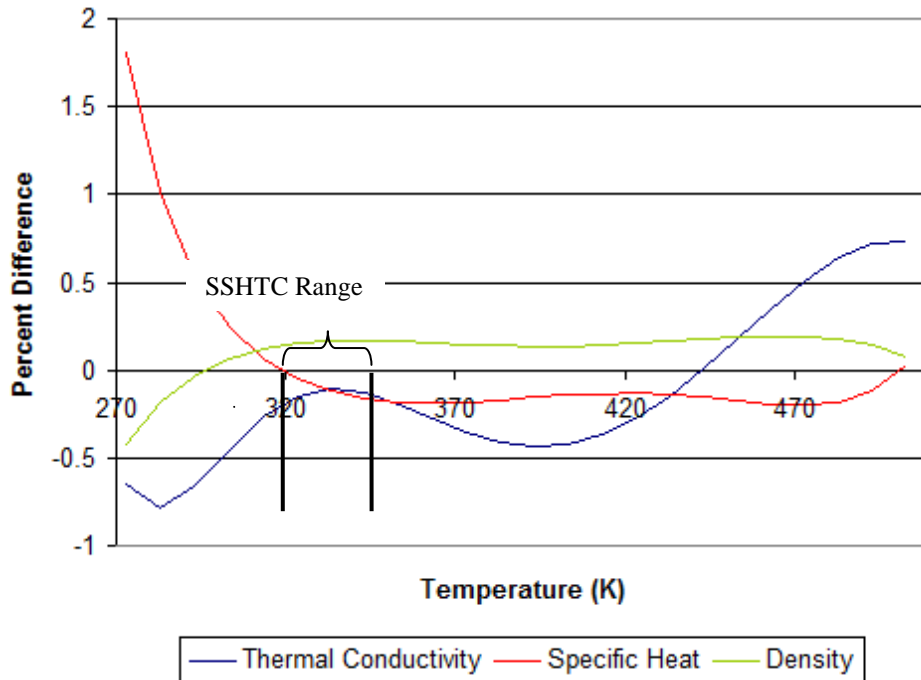
**Fig. 4.1. Comparison of Steady State Heat Transfer Code and NIST temperature-
dependent properties of water at 2.968 Mpa.**

The variance in viscosity is shown separately in Fig. 4.2. It has a large deviation (45%) at low
temperatures and approaches 5% error at the higher temperature range. The pressure at which the SSHTC
properties were evaluated is not explicitly stated. If the SSHTC properties are at atmospheric pressure,
then the natural conclusion is that the increase in operating pressure most affects the viscosity of water as
compared to the other intrinsic properties.
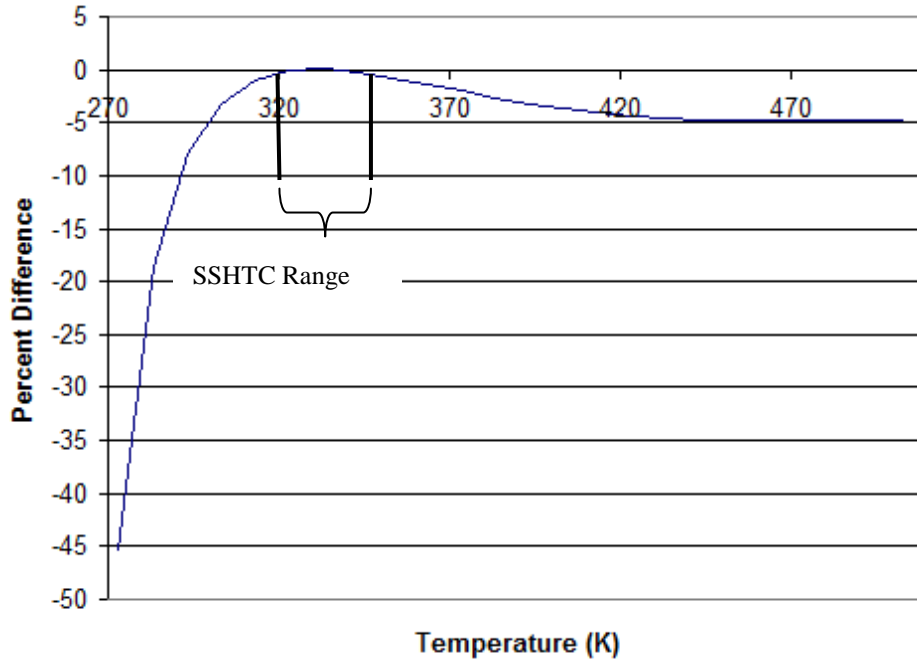
Fig. 4.2. Variation in viscosity of water reported in SSHTC as compared to NIST at 2.968 MPa.

With respect to the data in both Fig. 4.1 and Fig. 4.2, the difference could be attributed to the increased accuracy and improvements in the ASME steam tables over the years since the SSHTC was originally developed (1967).

Defining a COMSOL material library is similar to defining functions within COMSOL. Once the library is created, the graphical user interface (GUI) has a tool to define analytical functions, polynomials, and tabular data. However this tool is limited to one independent variable. To create a look-up table based on two parameters in the present version of COMSOL (v3.5a), the *.xml library file must the manually edited. The format for the data structure is shown in Fig. 4.3. The x variable(s) are defined as temperature in this case. The y variable(s) are defined as pressure. These arrays form a matrix where the y data provides the row index, and the x data provides the column index. The data is then entered by row. There is no need to denote progression to the next row. The data string is a continuous list.

Fig. 4.3. Schematic of data structure for the COMSOL material library with two independent variables.

Example 1 below shows a section of the *.xml code for defining the data structure in Fig. 4.3. The indices on the data values are noted in typical (row, column) format.

Example 1. *Text entry for .xml coding.*

```
<x type="stringarray">
{"T1","T2","T3",…,"Tn"}
 </x>
<y type="stringarray">
       {"P1","P2",...,"Pm"}
</y>
<data type="stringarray">
{"value(1,1)","value(1,2)",…,"value(1,n)","value(2,1)",…,"value(2,n)","value(m,1)",…,"value(m,n)"}
</data>
```

5. INVESTIGATION OF THE NECESSITY OF THE USE OF EXTRUSION COUPLING VARIABLES[2]

Extrusion coupling variables are used to map information from one domain to another. It had been previously stated [7] that extrusion coupling variables were necessary to accurately model the temperature field between the fuel and clad as well as between the clad and the coolant. A new model was created to investigate this issue which consists of one half of a fuel plate coupled with one half of a coolant channel as shown below in Fig. 5.1. An entrance and exit region was also added to more accurately portray the influence of the fuel plate on the fluid. The scale has been greatly exaggerated for visualization. The leading edge of the fuel plate is filleted to reduce the pressure drop due to form drag [4]. The radius of curvature of the fillet is 6.35×10^{-4} m (0.025") as shown on the fuel plate design drawings [8]. The material properties used in this model are constant to reduce central processing unit (CPU) time and random access memory (RAM) consumption in solving the model.
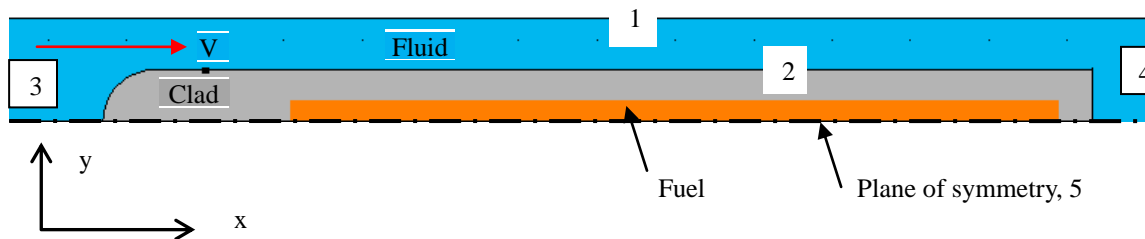
Fig. 5.1. Geometry of 2-D model used in COMSOL showing boundary numbers for later reference.

5.1   MESH STRUCTURE

The HFIR fuel plate structure is very thin, 1.27×10^{-3} m (0.050") thick, relative to its height, 0.6096 m (24") tall, which produces a very high aspect ratio of 480. This high aspect ratio requires the meshing scheme to be considered more carefully. An optimal mesh will have sufficient density at the clad-coolant interface and fluid exit boundary while the interior of the flow channel and solid will have minimum density in the axial direction.

COMSOL recommends a mapped mesh, which is a grid like mesh, for thin structures, however the curved geometry of the leading edge of the fuel plate restricts the use of this type of mesh. The free mesh is easiest to set up on this geometry with COMSOL. The interaction at the clad-coolant interface requires a dense mesh to capture the thermal-fluid boundary layer physics properly. Thus the number of elements is very large while the quality of the mesh is poor. Figure 5.2 shows a triangular mesh, with 3.56×10^5 elements overall, at the leading edge of the fuel plate which extends 1 mm.

---

[2]More information can be found on the use of extrusion coupling variables in Appendix A. Ref: COMSOL Multiphysics User's Guide.
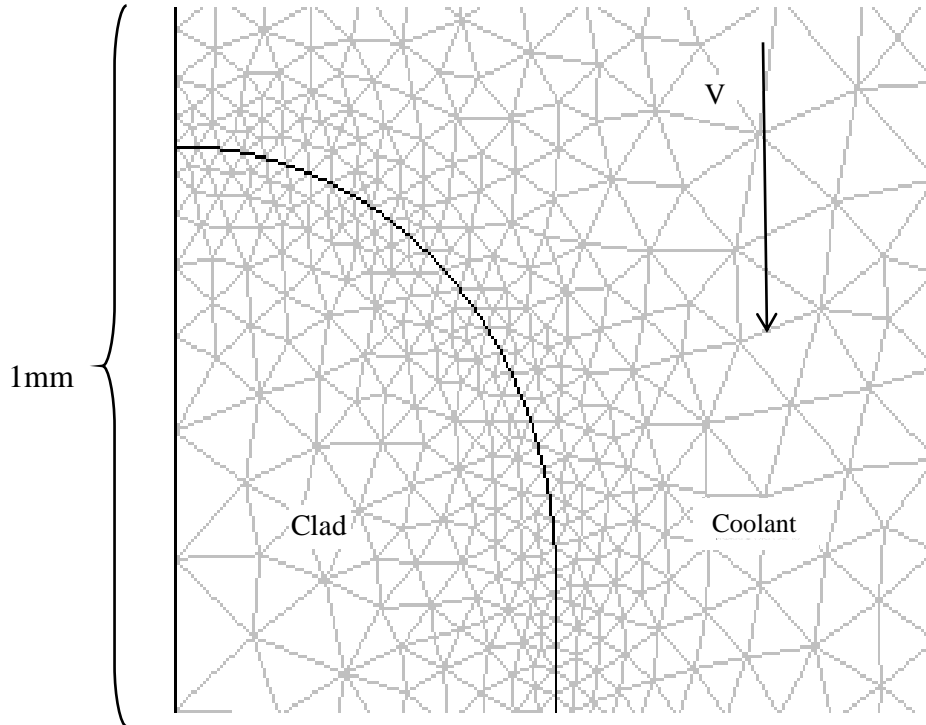
Fig. 5.2. Free mesh structure consisting of 3.66×10^5 elements.

The reader should take notice that the quality of the mesh is poor indeed (the mesh quality is printed as a COMSOL diagnostic from the output of the "mesh statistics") and produces large errors in the energy balance of the solid-fluid interaction (to be discussed in Section 5.3), on the order of 34%.

COMSOL provides a boundary layer meshing (BLM) option which, for 2-D simulations, creates rectangular elements along a solid-fluid interface and triangular elements in the remaining portions of the fluid domain. The user has control over the selection of the number of layers in the mesh and the density of these layers outward from the solid-fluid boundary. The BLM also displays its superiority in the elemental axial length. This feature reduces the overall number of elements while providing sufficient mesh density at the clad-coolant interface. Figure 5.3 below depicts the BLM at the leading edge as well.
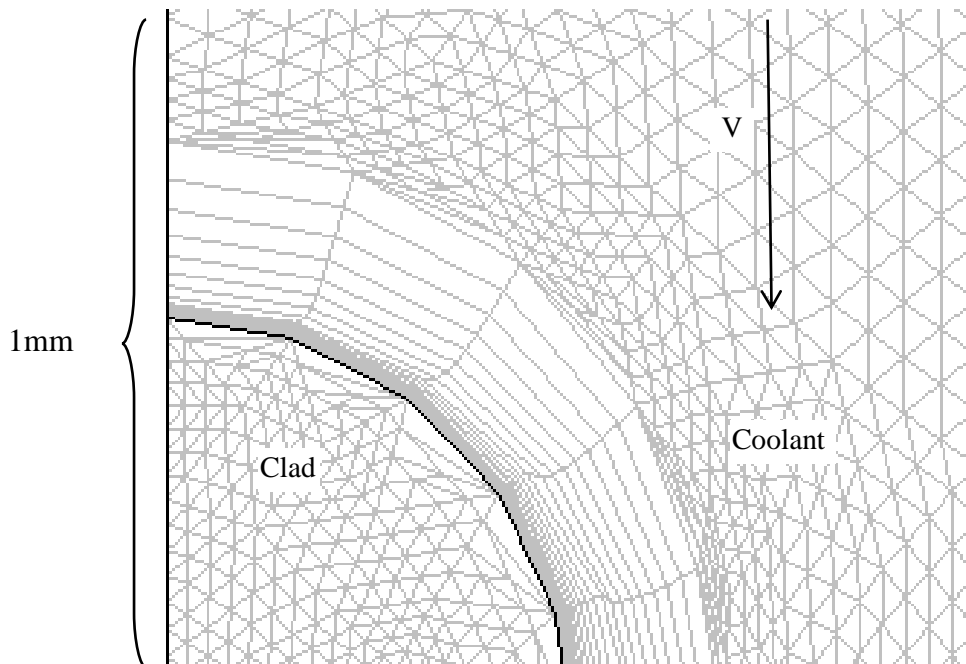
Fig. 5.3. BLM consisting of 2.36×10^5 elements.

One immediately notices the significant increase in the mesh density of the BLM at the clad-coolant interface relative to that shown in Fig. 5.2 above for a free mesh. Also, the overall number of elements has decreased by more than 35%. With this mesh, the physics is better represented and as a result the relative error in the energy balance of the solid-fluid interaction was below 3.4% which is a significant improvement over previous meshes.

Although not shown here, further improvement in the mesh design may be achieved in COMSOL by meshing domains independently of each other. In this manner, the mesh in the solid region is not altered by the mesh contents in the fluid region. This may be achieved by specifying only a single subdomain be meshed, or by using the assembly feature during geometry creation.

5.2   MODEL PHYSICS

The multiphysics application mode used in the present research was the thermal-fluid interaction, turbulent, non-isothermal flow equation set. The k-ε equation set was chosen to model the turbulent flow. COMSOL provides two turbulent flow equation sets, the k-ε and the k-ω. The k-ε model was preferred due to previous experience with this modeling equation set (k-ω is more applicable to rotational flows, but can also be applied here as an alternative. Often the k-ε solution is used as a initial condition for the k-ω solution). The k-ε turbulent equation set is coupled with the turbulent energy equation and the solid energy equation for this model.

In three dimensions, the mass and momentum portion of the k-ε equation set consists of four equations. The Reynolds stress term is described by the use of an eddy viscosity, η_T, that depends on the turbulent kinetic energy, k, and the rate of dissipation of that energy, ε. Thus an additional two equations are used to determine k and ε. The continuity equation is

$$\nabla \cdot (\rho \vec{u}) = 0 \tag{5.1}$$

where ρ is the density of the fluid and u is the velocity field.

The turbulent momentum equation is

$$\rho(\vec{u} \cdot \nabla)\vec{u} = \nabla \cdot [-p\boldsymbol{I} + (\eta + \eta_T)(\nabla \vec{u} + (\nabla \vec{u})^T - (2/3)(\nabla \cdot \vec{u})\boldsymbol{I} - (2/3)\rho k_{turb}\boldsymbol{I})] \\ + \vec{F} \tag{5.2}$$

where k_{turb}, is the turbulent kinetic energy not to be confused with the thermal conductivity, $\boldsymbol{I}$ is the identity tensor, and η_T is the turbulent viscosity.

The turbulent kinetic energy equation is

$$\rho \vec{u} \cdot \nabla k_{turb} = \nabla \cdot \left[\eta + \frac{\eta_T}{\sigma_k}\nabla k_{turb}\right] + \eta_T P(\vec{u}) - \frac{2\rho k_{turb}}{3}\nabla \cdot \vec{u} - \rho\varepsilon \tag{5.3}$$

where ε is the dissipation of turbulent kinetic energy and σ_k is a constant equal to 1. The dissipation of turbulent kinetic energy is governed by

$$\rho \vec{u} \cdot \nabla \varepsilon = \nabla \cdot \left[\eta + \frac{\eta_T}{\sigma_\varepsilon}\nabla\varepsilon\right] + \frac{C_{\varepsilon_1}\varepsilon}{k_{turb}}\left[\eta_T P(\vec{u}) - \frac{2\rho k_{turb}}{3}\nabla \cdot \vec{u}\right] - \frac{C_{\varepsilon_2}\rho\eta^2}{k_{turb}} \tag{5.4}$$

where $C_{\varepsilon 1}$ and $C_{\varepsilon 2}$ are constants having values of 1.44 and 1.92 respectively and σ_ε is equal to 1.3. The vector valued function P is defined as

$$P(\vec{u}) = \nabla \vec{u} : [(\nabla \vec{u})^T + \nabla \vec{u}] - \frac{2}{3}(\nabla \cdot \vec{u})^2 \tag{5.5}$$

where the first term is the tensor product of the dyads and the second term is the square of the divergence of the velocity field. The turbulent viscosity is defined as

$$\eta_T = \rho C_\mu \frac{k_{turb}^2}{\varepsilon} \tag{5.6}$$

where C_μ is equal to 0.09. The turbulent viscosity is a property of the flow, not of the fluid. The boundary conditions used for the flow equations listed above, equations 5.1- 5.4, are given below in Table 5.1.

Table 5.1. Turbulent flow boundary conditions

Boundary Number	Condition
1	Symmetry
2	Logarithmic Wall Function $y^+ = 30$
3	$u_{in} = 15.88$ m/s
4	$p_{out} = 0$ Pa

The turbulent thermal energy is governed by the equation

$$\nabla \cdot \left[-\left(k_{fluid} + k_T\right)\nabla T_f \right] = q''' + q_s T_f - \rho C_p \vec{u} \cdot \nabla T_f \tag{5.7}$$

where T_f is the fluid temperature and k_T is the turbulent thermal conductivity of the flow due to turbulent mixing. The constitutive equation for the turbulent thermal conductivity is

$$k_T = \frac{C_p \eta_T}{Pr_T} \tag{5.8}$$

The boundary conditions used for the determination of the temperature field within the fluid are given below in Table 5.2.

Table 5.2. Boundary conditions for thermal energy transport through the fluid

Boundary Number	Condition
1	Adiabatic
2	$T = T_s$
3	$T = T_{in}$
4	Convective Flux

The thermal energy in the solid (i.e., the fuel and clad domains) is governed by

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = \frac{-q'''}{k} \tag{5.9}$$

with the proper choices for both k and q depending on which domain one happens to be in. Also, the temperature field T, in this domain, is the temperature within the solid, T_s. The boundary conditions used for the determination of the temperature field within the solid are given below in Table 5.3.

Table 5.3. Boundary conditions for thermal energy diffusion through the solid

Boundary Number	Condition
2	q_0 = - qwf_htgh
5	Adiabatic
Fuel-Clad	Continuity

All of the physical aspects of the model have been established. The solution (i.e., flow field and thermal field) to the above stated problem is shown below in Fig. 5.4. Note that the scale of the model has been augmented in order to visualize the solution over the entire geometry.
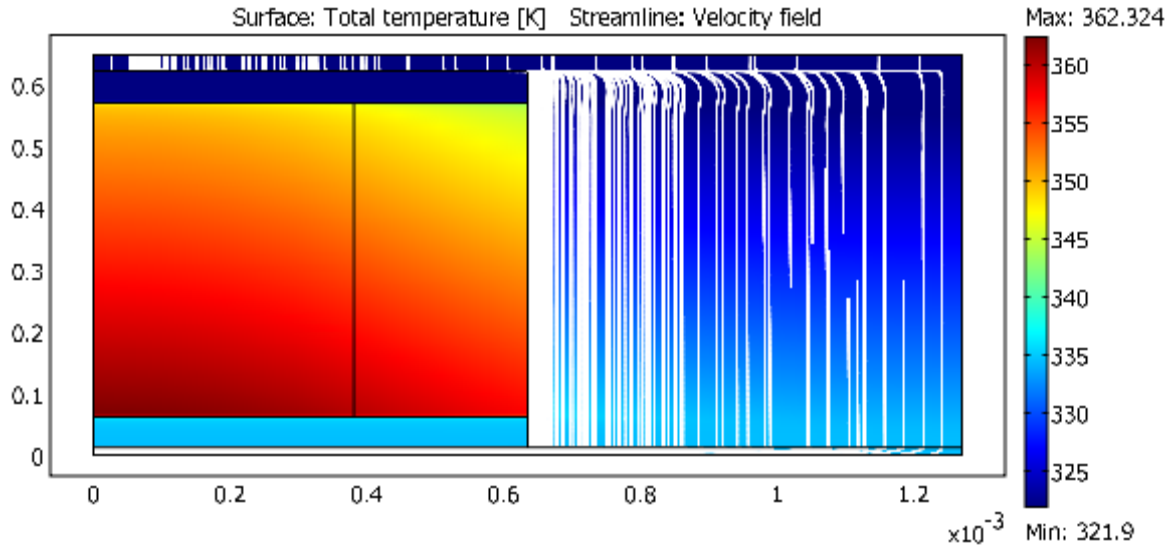
Fig. 5.4. Thermal and fluid velocity field solutions for the governing equations and boundary conditions described in this section.

The orientation of the COMSOL model, Fig. 5.4, is rotated 90° relative to that depicted in Fig. 5.1 as in the actual HFIR system (fuel vertically oriented). From Fig. 5.4, it appears that the physics are properly represented. The solid temperatures are highest at the fuel symmetry plane near the trailing edge of the fuel plate. This is expected due to the inverse relationship of the film coefficient along a heated distance (i.e., the film coefficient decreases along the axial length towards the exit). The model results are used to verify mass and energy balances addressed in the next section.

5.3   MODEL POST-PROCESSING

The conservation of the overall thermal energy and the fluid mass is crucial to the quality assurance of the model. The mass flow rate is defined as

$$\dot{m} = \iint \rho u \, dA \tag{5.10}$$

where dA is the elemental cross sectional area of the flow channel. Since no sources or sinks reside in the flow channel, the mass flow rate must remain constant from the inlet to the outlet. COMSOL has a built in integration feature allowing the user to integrate both predefined and user defined quantities along the boundaries. Integrating the product of the density and velocity, ρu over the fluid inlet boundary yields a value of 19.95192 m^2/s and integrating the density- velocity product over the fluid outlet boundary gives 19.79586 m^2/s. Comparing these two values the relative error in the mass flow rate is 0.7822%.

The balance of the thermal energy must be checked in both the solid domain and the fluid domain. To this end the First Law of Thermodynamics is employed.

$$E_{st} = E_{enter} - E_{exit} + E_{gen} \tag{5.11}$$

The Solid Domain Energy Balance:

With $E_{st} = 0$, for the steady state model, Equation 5.11 may be written as

$$\iiint (\nabla \cdot q'')dV = \oiint q'' \cdot \hat{n}\, dA \tag{5.12}$$

which is the Divergence Theorem applied to the thermal energy flux. From Gauss's Law one obtains

$$\nabla \cdot q'' = q''' \tag{5.13}$$

where q is the power. Combining Equations 5.12 and 5.13 yields

$$\iiint q'''dV = \oiint q'' \cdot \hat{n}\, dA \tag{5.14}$$

This equation is used to calculate the numerical error in the conservation of energy in the solid domain. This same equation is used for the fluid domain, however, the flux integral must be modified.

The integrated thermal source term, in Equation 5.14, within the fuel has a value of 1.0289×10^6 W/m, while the integrated heat flux at the fuel boundary is 1.0272×10^6 W/m. Thus the relative error in Equation 5.14 over the fuel is 0.17%. Comparing the integrated thermal source with the integrated heat flux over the clad boundary, 1.0288×10^6 W/m, yields a relative error of 0.012%. Thus the overall energy is well balanced in the solid domain.

The Fluid Domain Energy Balance:

The fluid domain requires a bit more analysis to evaluate the error in the solution. Referring back to Equation 5.11, the situation in the fluid requires that $E_{st} = E_{gen} = 0$. Thus the energy equation for the fluid is $E_{enter} = E_{exit}$. The fluid has an inherent energy associated with the flow of mass and the fluid temperature. This energy is advected into the region of interest and is described by

$$q = \dot{m}C_p T_{bulk} = \iint \rho C_p u T_f dA \tag{5.15}$$

Equation 5.15 may be used in the calculation of the energy advected out of the domain as well. In this case, T increases in the flow direction as a result of the introduction of thermal energy from the fuel source into the fluid. From this argument it is reasonable to expect that the exiting advected energy will be larger than the entering advected energy by an amount equal to that entering the flow due to the thermal source. Thus the formulation of the First Law for the fluid domain is

$$\oiint q'' \cdot \hat{n}\, dA + \left[\iint \rho u C_p T_f\, dA \right]_{inlet} = \left[\iint \rho u C_p T_f\, dA \right]_{outlet} \tag{5.16}$$

By combining Equation 5.14 and 5.16, the equation for the energy conservation error in the fluid is found to be

$$\iiint q''' \, dV + \left[\iint \rho u C_p T_f\, dA \right]_{inlet} = \left[\iint \rho u C_p T_f\, dA \right]_{outlet} \tag{5.17}$$

COMSOL was used to evaluate the energy advection terms in Equation 5.17 yielding 2.772357×10^7 W/m

and 2.672972×10^7 W/m respectively. The difference in the advected energy terms is 9.93850×10^5 W/m, which when compared to the integrated thermal source term yields a relative error of 3.39%. The low relative errors in the conservation of mass and energy builds confidence in the model created. Given that the model was successful without the use of the extrusion coupling variables, it can be concluded that they are unnecessary.

6. CONFIRMATION OF THE USE OF ADIABATIC CONDITIONS AT THE LEADING AND TRAILING EDGES OF THE FUEL PLATE

The SSHTC is a steady state, 1-D, thermal-hydraulic code[3]. The solution lattice used in the SSHTC, shown below in Fig. 6.1, is a grid pattern similar to the mapped mesh structure described in the previous section. This mesh is laid over the clad-coolant interface.

The thermal energy flux leaving the clad surface at position i,j is given by

$$\frac{q}{A_{i,j}} = \frac{Qf\varphi_{i,j}}{A} \tag{6.1}$$

as described by equation 7 of the SSHTC description [1], where Q is the reactor power level, f is the fraction of reactor power deposited in the fuel element, $\varphi_{i,j}$ is the power density distribution multiplier at position i,j, and A is the local clad surface area. Equation 6.1 does not allow for the axial nor the span wise flow of thermal energy and thus adiabatic conditions are obtained at the boundaries of each local elemental area. From this argument the reader may conclude that the extreme boundaries of the fuel plate will also exhibit an adiabatic condition. As a result of Equation 6.1, the thermal conductivity tensor within the solid is reduced to its 1-D form.

$$k = \begin{bmatrix} 0 & 0 & 0 \\ 0 & k_{yy} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{6.2}$$

This tensor representation is taken with reference to the coordinate orientation of Fig. 5.1 above. Because the SSHTC is a one-dimensional, lumped-parameter code, the true geometry of the fuel plate used in the SSHTC does not explicitly incorporate the filleted leading edge, but the form-loss effect of a pressure drop caused by the rounded leading edge is certainly incorporated. Furthermore, the SSHTC does not explicitly model the involute shape of the fuel plate, but rather, approximates the actual arc length of the plated by based on a flat plate model.

---

[4]1-D in the sense that no axial or radial (span-wise) heat conduction is allowed within the solid domains. All heat generated goes through the clad to the fluid.
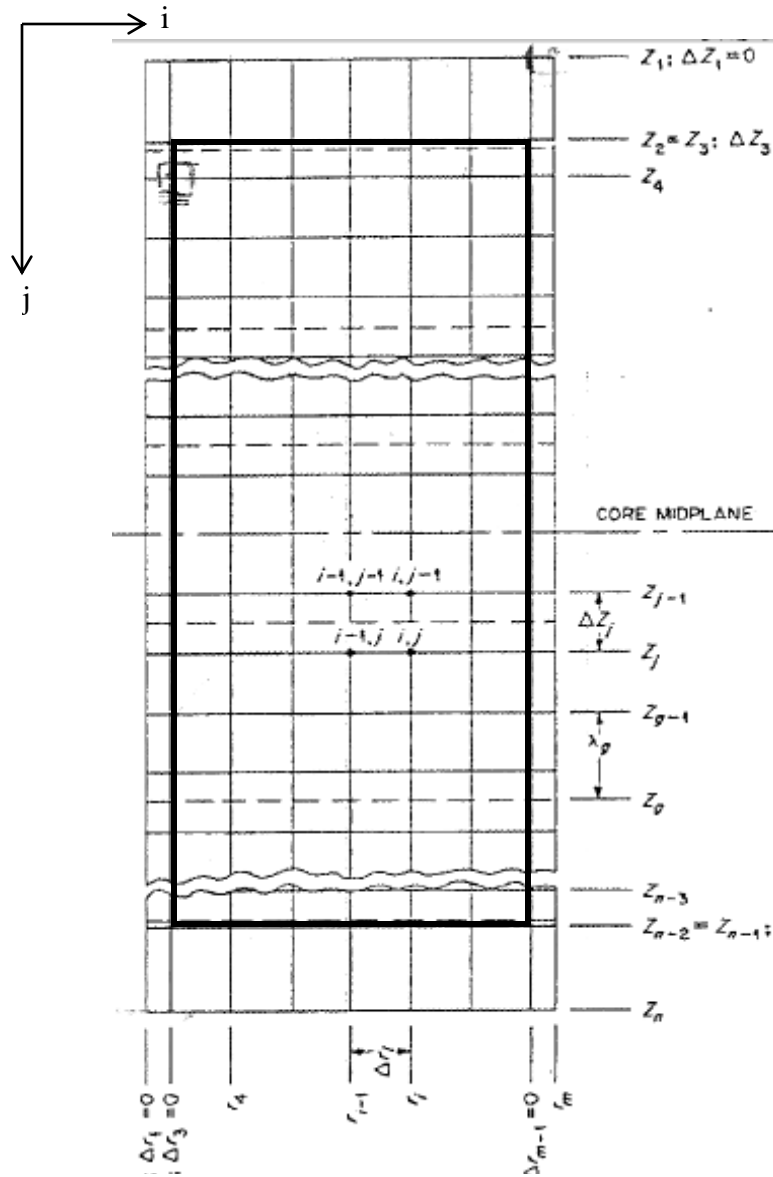
Fig. 6.1. Mesh structure used in the SSHTC.

(The region enclosed by the heavy outlined rectangle
represents the fueled portion of the fuel plate. Positions
within the mesh are indicated by i,j. Drawing is Fig. 5 of
Ref. 4.)

7. DETERMINATION OF A POWER DENSITY PROFILE FOR USE IN COMSOL

7.1   COMPARISON OF THE INTEGRAL POWER GENERATION USED IN REFERENCE 7 WITH THE SSHTC

From the initial COMSOL research of ref. 7, hereafter referred to as "ref-7", axial multipliers are used to describe the distribution of power/heat load in HEU fuel plate of HFIR. The weights are multiplied by the volumetric heat generation rate to achieve a generation rate for individual subdomains in COMSOL. The heat generation is applied to 2D models that represent vertical cross-sections of the fuel plate. At the start of the studies documented here, several questions remained unclear about the multipliers used: did they accurately represent the heat load in the fuel plate, and did they result in an overall power balance for the prescribed heat load?

First, the volumetric heat load is established. The ref-7 report uses 80.7 MW as the total heat load in the fuel plates. The RELAP5 report [5] establishes a balance among the heat load in the fuel plates and other heat structures to result in an overall thermal load of 85 MW for the HFIR HEU reactor.

The volume of active fuel, fuel-filler combination, is incorrectly stated in ref-7 as 0.030355 m^3. The corrected active fuel volume is 0.015181 m^3. Details of this correction are in Appendix B. This volume represents the active volume of fuel in the plate and spreads the heat over the full thickness of the plate minus the clad, a length of 0.000762 m (0.030" nominal fuel thickness). Note, if the fuel and filler are being modeled separately, then a new fuel volume-specific heat load should be calculated. With these parameters, the average volumetric heat load is 5.3159×10^9 W/m^3.

The axial multipliers used in ref-7 are shown in Table 7.1. It is noted that the fuel plates are nominally 0.6096 m (24") long. The outer 0.0508 m (2") on the top and bottom of the plate does not contain fuel. Thus, the zero position is actually 0.0508 m down from the top of the fuel plate. The source of these multipliers is from recent output from the SSHTC developed at HFIR[1].

Table 7.1. Axial heat load multipliers cited in TM report

Position – (m)	Multiplier
0.000	0.678
0.014	0.722
0.034	0.815
0.054	0.924
0.074	1.031
0.094	1.130
0.114	1.227
0.134	1.312
0.154	1.387
0.174	1.447
0.194	1.493
0.214	1.520
0.234	1.532
0.254	1.533
0.274	1.523
0.294	1.494
0.314	1.448
0.334	1.384
0.354	1.312
0.374	1.235
0.394	1.148
0.414	1.050
0.434	0.944
0.454	0.819
0.474	0.709
0.494	0.706
0.508	0.703

The first analysis of this data is to determine the magnitude of power prescribed by this distribution.   This numerical calculation represents the integral power generation. The axial weights are multiplied by the volumetric heat generation and by the control volume for each axial position. The control volume is an involute prism with the height taken half way between axial nodes, the width of the fuel (0.000762 m), and the arc length of the active fuel section (0.079502 m [3.13"]). The sum of the power contributions over the axial control volumes results in 194,000 W per plate. The average multiplier is 1.186. The average is determined by calculating the average volumetric heat generation for the inner fuel and dividing by the average generation for the whole core. On a volume basis, the inner fuel section contains 34% of the total fuel. On a heat load basis, the inner fuel section contains 41% of the total generation. No analysis of the outer fuel section contribution is cited in ref. 7 to clarify the validity of the axial distribution.

The source of the axial multipliers is the SSHTC input found in the McClain report[1]. This data is reproduced in Table 7.2. This input also defines the axial spacing and multipliers for the code. All dimensions in this input are in inches. However, the input also defines radial nodes. By comparison, the multipliers used in ref. 7 are only the axial multipliers for the first active radial node.

Table 7.2. SSHTC power density multipliers and nodal positioning data

```
Inner element
DR(I,1), I = 1,7 and I = 8,M
DR is the radial increment for the node.
    0.          0.0895     0.          0.3386     0.3937     0.3937
0.3937
    0.3937     0.2362     0.          0.0794
DZ(J,1) 10 per card, except for 83 which has J = 31, N.
DZ is the axial increment for the node
 0.    2.000 0.    0.5512 0.7874 0.7874 0.7874 0.7874 0.7874
0.7874
0.7874 0.7874 0.7874 0.7874 0.7874 0.7874 0.7874 0.7874 0.7874
0.7874
0.7874 0.7874 0.7874 0.7874 0.7874 0.7874 0.7874 0.7874 0.5512 0.
Inner element relative power density.  Core power is 85MW.  You
would have to convert the numbers below to power
density in the fuel meat, which means making use of the thickness
of meat at each radial location.
0       0       0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0       0       0
0       0       0.678 1.109 1.379 1.515 1.470 1.344 1.186 0       0
0       0       0.722 0.741 0.738 0.771 0.809 0.830 0.837 0       0
0       0       0.815 0.753 0.738 0.767 0.802 0.830 0.824 0       0
0       0       0.924 0.848 0.810 0.821 0.850 0.889 0.893 0       0
0       0       1.031 0.952 0.904 0.917 0.939 0.980 0.988 0       0
0       0       1.130 1.043 0.990 1.007 1.027 1.067 1.077 0       0
0       0       1.227 1.126 1.068 1.089 1.107 1.146 1.162 0       0
0       0       1.312 1.198 1.137 1.163 1.179 1.217 1.241 0       0
0       0       1.387 1.264 1.200 1.233 1.241 1.280 1.316 0       0
0       0       1.447 1.322 1.255 1.285 1.294 1.334 1.383 0       0
0       0       1.493 1.372 1.306 1.332 1.339 1.380 1.445 0       0
0       0       1.520 1.398 1.338 1.369 1.372 1.410 1.478 0       0
0       0       1.532 1.408 1.350 1.386 1.388 1.422 1.490 0       0
0       0       1.533 1.403 1.349 1.386 1.388 1.420 1.485 0       0
0       0       1.523 1.393 1.335 1.372 1.375 1.418 1.464 0       0
0       0       1.494 1.368 1.305 1.342 1.345 1.380 1.425 0       0
0       0       1.448 1.324 1.256 1.289 1.295 1.330 1.372 0       0
0       0       1.384 1.264 1.193 1.221 1.235 1.283 1.312 0       0
0       0       1.312 1.195 1.127 1.150 1.167 1.212 1.248 0       0
0       0       1.235 1.120 1.057 1.077 1.097 1.144 1.177 0       0
0       0       1.148 1.039 .982  1.000 1.023 1.069 1.098 0       0
0       0       1.050 0.952 .903  .920  0.945 0.988 1.011 0       0
0       0       0.944 0.859 .820  .839  0.864 0.904 0.916 0       0
0       0       0.819 0.762 .735  .755  0.783 0.816 0.812 0       0
0       0       0.709 0.668 .667  .698  0.735 0.762 0.753 0       0
0       0       0.706 0.678 .680  .713  0.749 0.760 0.762 0       0
0       0       0.703 1.028 1.231 1.342 1.319 1.216 1.078 0       0
```

The SSHTC develops a 2-D grid that defines the heat generation on an axial and radial basis, as shown in Fig. 6.1. A comparison of these multipliers is shown in Fig. 7.1.
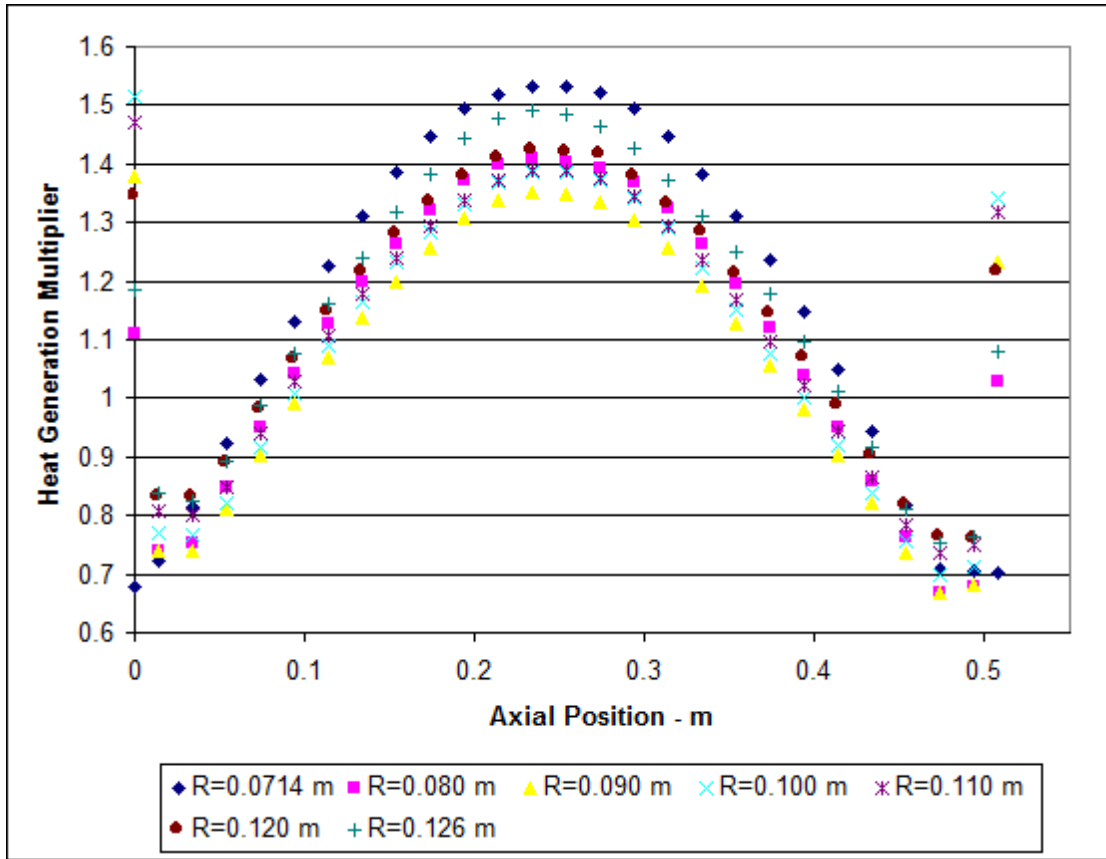
Fig. 7.1. Axial multipliers for the inner fuel from Table 9.

The first radius has the general shape of a parabola. The multipliers start and end with the lowest value and have the highest value in the middle. All other radial locations have more of a "W" shape with lower peak values in the center.

7.2 MULTIPLIER ARRAY NORMALIZATION PROCEDURE OUTLINED IN THE SSHTC AND RESULTANT POWER DENSITY BASED ON FUEL AND FILLER GENERATION

The SSHTC requires a "normalization" of the multiplier array shown above in Table 7.2, which is common to all fuel plates. The scaled array has the following form

$$\varphi_{i,j} = \frac{\varphi'_{i,j}}{\Phi} \tag{7.1}$$

where $\varphi_{i,j}$ is the scaled array, $\varphi'_{i,j}$ is the array shown in Table 7.2, and Φ is defined in the SSHTC as

$$\Phi = \frac{M_{inner\ element} + M_{outer\ element}}{N_{inner\ element} + N_{outer\ element}} \tag{7.2}$$

before defining M and N in the above equation, it is convenient to define the incremental arc length.

From the SSHTC, the incremental arc length is

$$\Delta s_i = \frac{\Delta r_i}{2R}\left[\Delta r_i + 2\left(R_b + \sum_{l=1}^{i-1}\Delta r_l\right)\right]$$ (7.3)

where r is the radial position and R_b is the radial position of the involute origin. Equation 7.3 can be applied to both inner and outer fuel elements. With the incremental arc length defined, M and N are defined as

$$M = 2R_b\sum_{i=4}^{m-2}\sum_{j=4}^{n-2}\left[\frac{\varphi'_{i-1,j-1} + \varphi'_{i,j-1} + \varphi'_{i-1,j} + \varphi'_{i,j}}{4}\right]\Delta s_i\Delta z_j$$ (7.4)

$$N = 2R_b\sum_{i=4}^{m-2}\sum_{j=4}^{n-2}\Delta s_i\Delta z_j$$ (7.5)

MATLAB was used to perform the above calculations and thus to scale the multipliers, φ'. The results of this procedure are shown below in Table 7.3. Note that Table 7.3 omits the zeros shown in Table 7.2 which represent unfueled regions. Table 7.3 represents φ in Equation 7.1. By inspection of the entries of Table 7.3, it is obvious that the sum of the array's components is greater than 1. This array will significantly increase the power generated per plate and thus in the entire HFIR core. As a result of this discrepancy, the values presented in Table 7.3 are modified to conserve the total core thermal energy output. (i.e., the sum of the individual nodal power contributions must equal the power dissipated in each plate). The SSHTC also performs this normalization albeit the normalization is to the entered reactor power rather than to the normalized core-average power density of 1.0. To this end, a constant modifier was calculated by

$$Modifier = \frac{1}{\sum\varphi_{i,j}}$$ (7.6)

This number is then multiplied by the individual entries, $\varphi_{i,j}$, the result of which is shown in Table 7.4.

Table 7.3. Scaled multiplier array calculated using the procedure outlined in the SSHTC

0.7107	1.1625	1.4455	1.5881	1.5409	1.4088	1.2432
0.7568	0.7767	0.7736	0.8082	0.8480	0.8700	0.8774
0.8543	0.7893	0.7736	0.8040	0.8407	0.8700	0.8637
0.9686	0.8889	0.8491	0.8606	0.8910	0.9319	0.9361
1.0807	0.9979	0.9476	0.9612	0.9843	1.0273	1.0356
1.1845	1.0933	1.0377	1.0556	1.0765	1.1184	1.1289
1.2862	1.1803	1.1195	1.1415	1.1604	1.2013	1.2180
1.3753	1.2558	1.1918	1.2191	1.2358	1.2757	1.3008
1.4539	1.3249	1.2579	1.2925	1.3008	1.3417	1.3795
1.5168	1.3857	1.3155	1.3470	1.3564	1.3983	1.4497
1.5650	1.4382	1.3690	1.3962	1.4036	1.4465	1.5147
1.5933	1.4654	1.4025	1.4350	1.4382	1.4780	1.5493
1.6059	1.4759	1.4151	1.4528	1.4549	1.4906	1.5618
1.6069	1.4706	1.4140	1.4528	1.4549	1.4885	1.5566
1.5964	1.4602	1.3994	1.4382	1.4413	1.4864	1.5346
1.5660	1.4340	1.3679	1.4067	1.4099	1.4465	1.4937
1.5178	1.3878	1.3166	1.3512	1.3574	1.3941	1.4382
1.4507	1.3249	1.2505	1.2799	1.2945	1.3449	1.3753
1.3753	1.2526	1.1813	1.2055	1.2233	1.2704	1.3082
1.2945	1.1740	1.1080	1.1289	1.1499	1.1992	1.2338
1.2034	1.0891	1.0294	1.0482	1.0723	1.1205	1.1509
1.1006	0.9979	0.9465	0.9644	0.9906	1.0356	1.0597
0.9895	0.9004	0.8595	0.8795	0.9057	0.9476	0.9602
0.8585	0.7987	0.7704	0.7914	0.8208	0.8553	0.8512
0.7432	0.7002	0.6992	0.7317	0.7704	0.7987	0.7893
0.7400	0.7107	0.7128	0.7474	0.7851	0.7966	0.7987
0.7369	1.0776	1.2904	1.4067	1.3826	1.2746	1.1300

Table 7.4. Unit normalized set of scaled multipliers

0.0034	0.0056	0.0069	0.0076	0.0074	0.0068	0.0060
0.0036	0.0037	0.0037	0.0039	0.0041	0.0042	0.0042
0.0041	0.0038	0.0037	0.0039	0.0040	0.0042	0.0041
0.0047	0.0043	0.0041	0.0041	0.0043	0.0045	0.0045
0.0052	0.0048	0.0046	0.0046	0.0047	0.0049	0.0050
0.0057	0.0053	0.0050	0.0051	0.0052	0.0054	0.0054
0.0062	0.0057	0.0054	0.0055	0.0056	0.0058	0.0058
0.0066	0.0060	0.0057	0.0059	0.0059	0.0061	0.0062
0.0070	0.0064	0.0060	0.0062	0.0062	0.0064	0.0066
0.0073	0.0067	0.0063	0.0065	0.0065	0.0067	0.0070
0.0075	0.0069	0.0066	0.0067	0.0067	0.0069	0.0073
0.0077	0.0070	0.0067	0.0069	0.0069	0.0071	0.0074
0.0077	0.0071	0.0068	0.0070	0.0070	0.0072	0.0075
0.0077	0.0071	0.0068	0.0070	0.0070	0.0071	0.0075
0.0077	0.0070	0.0067	0.0069	0.0069	0.0071	0.0074
0.0075	0.0069	0.0066	0.0068	0.0068	0.0069	0.0072
0.0073	0.0067	0.0063	0.0065	0.0065	0.0067	0.0069
0.0070	0.0064	0.0060	0.0061	0.0062	0.0065	0.0066
0.0066	0.0060	0.0057	0.0058	0.0059	0.0061	0.0063
0.0062	0.0056	0.0053	0.0054	0.0055	0.0058	0.0059
0.0058	0.0052	0.0049	0.0050	0.0051	0.0054	0.0055
0.0053	0.0048	0.0045	0.0046	0.0048	0.0050	0.0051
0.0048	0.0043	0.0041	0.0042	0.0043	0.0046	0.0046
0.0041	0.0038	0.0037	0.0038	0.0039	0.0041	0.0041
0.0036	0.0034	0.0034	0.0035	0.0037	0.0038	0.0038
0.0036	0.0034	0.0034	0.0036	0.0038	0.0038	0.0038
0.0035	0.0052	0.0062	0.0068	0.0066	0.0061	0.0054

The sum of all of the entries in Table 7.4 is indeed 1.0 A plot of the unit normalized multipliers presented in Table 7.4 is shown in Fig. 7.2.
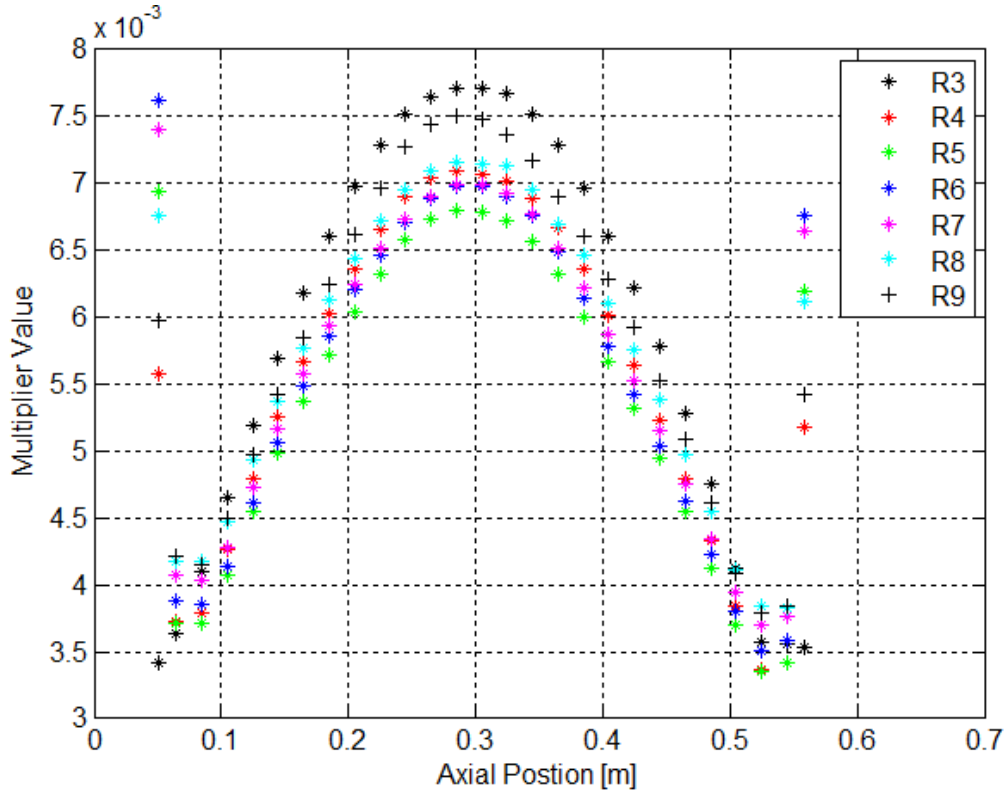
Fig. 7.2. The overall shape of the multiplier profile is preserved.
(Only the magnitude has been changed.)

Now that the multipliers are properly normalized, a power density profile may be created for use in COMSOL. Total core power values of $Q = 80.7$ MW and 85 MW were used in this analysis. To calculate a thermal energy generation rate, the active fuel volume of the core must be found. The RELAP5 code provides estimates for the fuel-filler volume for both the inner and outer core on a per plate basis as 3.03×10^{-5} m^3 and 2.71×10^{-5} m^3, respectively. The total generating volume of fuel in the core is found by

$$V_{total} = 171V_{inner} + 369V_{outer} \tag{7.7}$$

which for the above given values of inner and outer active volumes yields a value of 1.5181×10^{-2} m^3. The total power density of the core is calculated by

$$PowerDensity = \frac{Q}{V_{total}} \tag{7.8}$$

Thus the total generation rate for the HFIR core operating at 80.7 MW is 5.3159×10^9 W/m^3. Note that the remainder of the power (85.0-80.7=4.3 Mw) is modeled in the RELAP5 HFIR model as being generated and dissipated outside the fuel meat. From Equation 7.8, the power contributions from both the outer and inner core are calculated by using

$$q_{inner\ or\ outer} = \frac{Q}{V_{total}}V_{inner\ or\ outer} \tag{7.9}$$

From Equation 7.9 the contribution from the outer core is found to be 5.3158×10^7 W, which accounts for

65.87 % of the total core power, while the contribution of the inner core is $2.7543{\times}10^7$ W accounting for 34.13 % of the total. On a per plate basis, one inner plate dissipates $1.6107{\times}10^5$ W and one outer plate dissipates $1.4406{\times}10^5$ W. These values assume equal generation among all plates in the specified region (i.e., symmetry in the azimuthal direction.)

To create a power density profile for an inner fuel plate based on the normalized set of multipliers, the power generation of the core is multiplied by each entry in the array shown in Table 7.4 above. This distributes the power density through the fueled region in accordance with the profiles determined by the neutronics calculations done at HFIR. The same procedure for the multipliers can be carried out for the outer fuel plate as well.   The power density profile associated with the sixth radial position of the plate using the normalized multipliers for both the inner and outer fuel plates is shown below in Fig. 7.3.
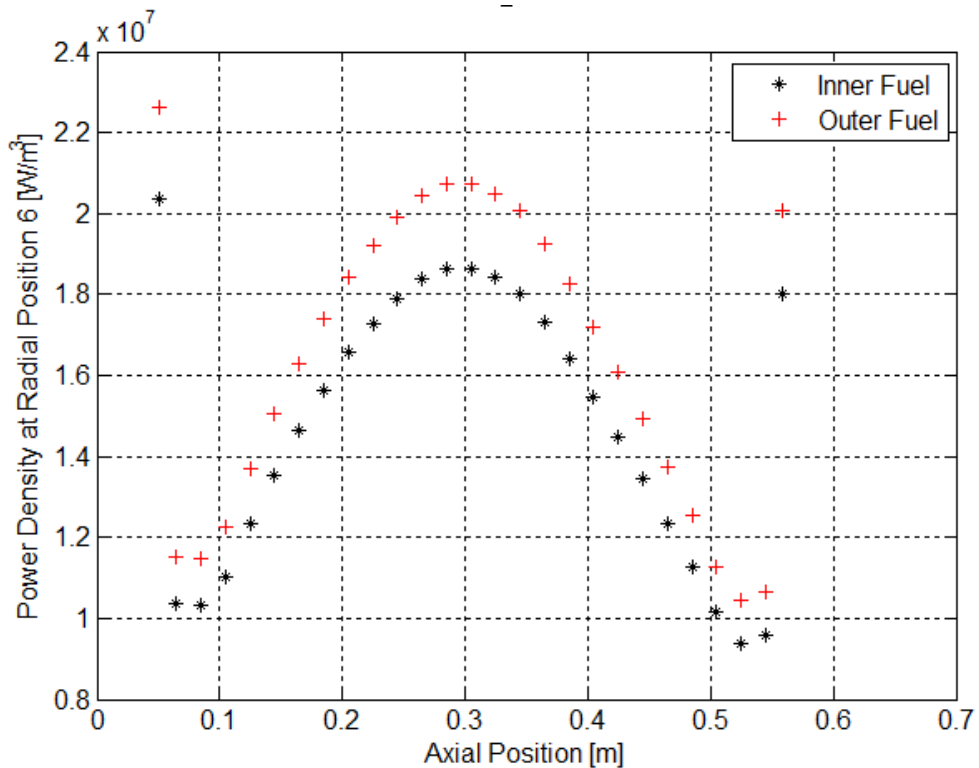
Fig. 7.3. Power density profile at radial Position 6 as a function of axial position using normalized multipliers.

(This power density is based on the fuel generating an equal amount of thermal energy in each plate, i.e. azimuthal symmetry.)

For comparison (and to resolve confusion arising in earlier studies), the power density profile generated for the 80.7 MW core using the SSHTC normalized multipliers in Table 7.3 is two orders of magnitude larger than that shown in Fig. 7.3. This scaled power density distribution is shown in Fig. 7.4 below.
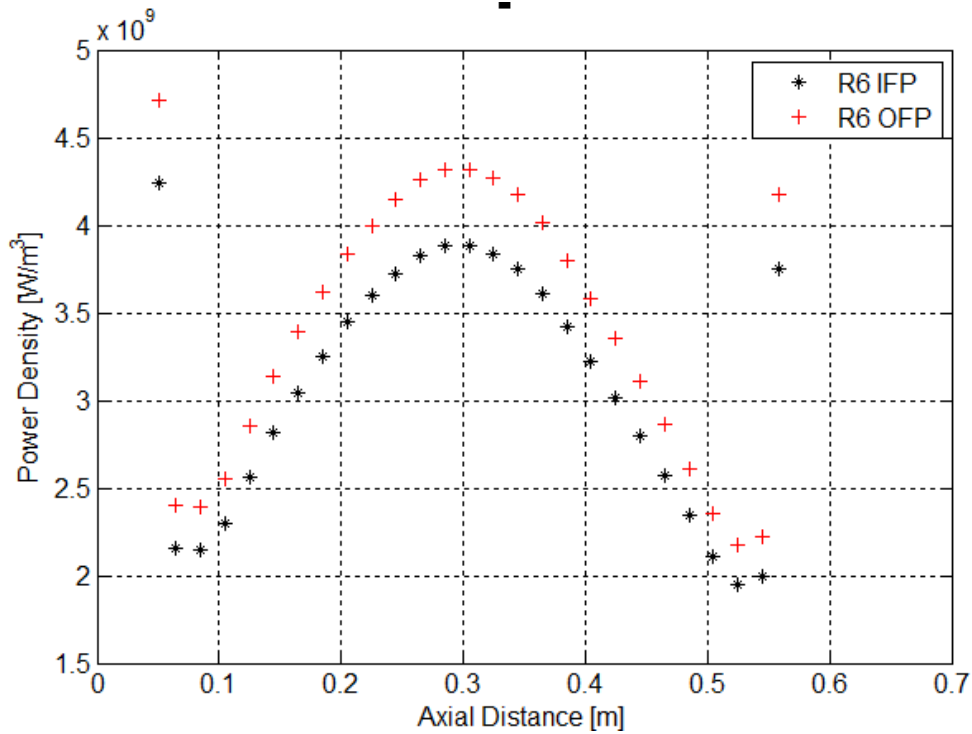
Fig. 7.4. Power density distribution for axial Position 6 using multipliers in Table 7.3.

The same procedure is followed for calculating the power density distribution for a total core power of Q = 85MW. The power density of the core is found to be 5.5990×10^9 W/m^3. From this power density, the power contributions from both the inner and outer core are 5.5989×10^7 W and 2.9010×10^7 W, respectively. Each plate in the inner core dissipates 1.6965×10^5 W, while each outer plate dissipates 1.5173×10^5 W.

The power density associated with the sixth radial position for the 85 MW core using the normalized multiplier array is shown below in Fig. 7.5. Again, the scaled multipliers produce a distribution that is two orders of magnitude larger than that in Fig. 7.5. There is only a slight shift up in power density for the 85 MW core relative to the distribution seen in Fig. 7.4.
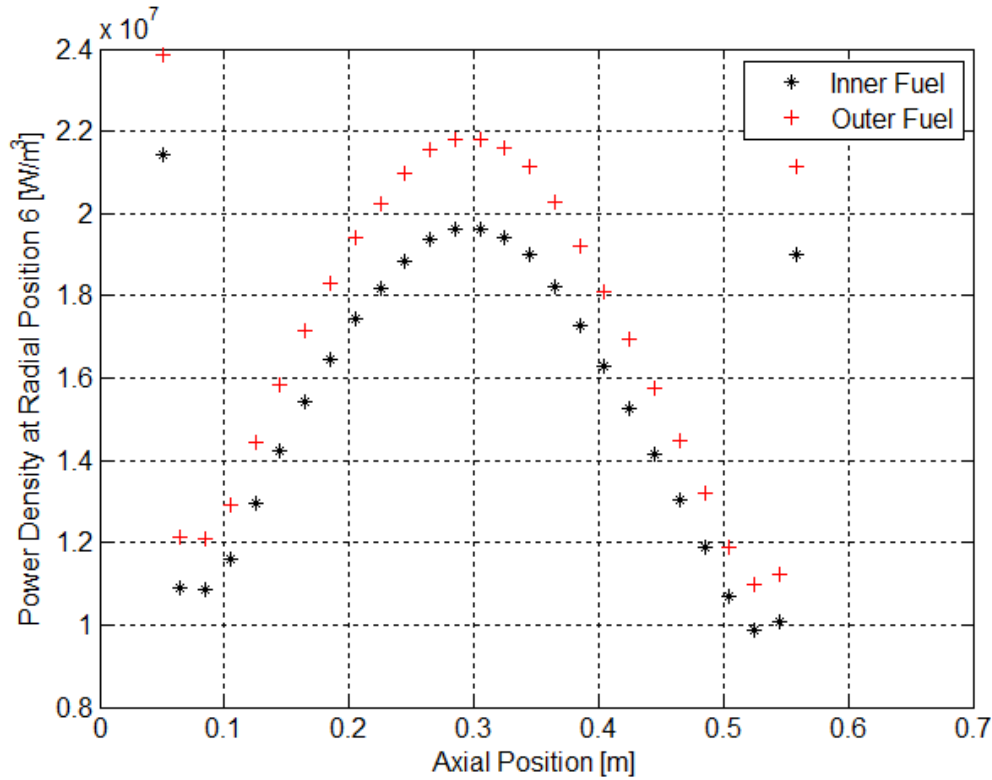
Fig. 7.5. Power density profile at radial Position 6 for the 85 MW core using normalized multipliers.

7.3 VOLUMETRIC AVERAGING OF THE SSHTC MULTIPLIER ARRAY AND RESULTANT POWER DENSITY FOR FUEL AND FILLER GENERATION

To average the multipliers at varying radial positions for each axial position, several manipulations are required. The data are given in terms of radial position which is not a dimension that can be directly used to determine the control volume for each node in the grid. The radial position must be translated into an arc length relative to the generating origin for the involute. The arc lengths can then be used to define the control volumes and average the multipliers on a volumetric basis.

To find the arc length from the origin of the involute to a radial position (R), the angle between the base radius (R_b) and the specified radius must be solved in Equation 7.10. The base radius is the inner radius of the respective involute shown in Table 7.5. The geometric information is from the RELAP5 report [5]. The equations were taken from a calculation review of involute generation by D. Cook [4].

Table 7.5. Radii of inner and outer fuel sections

	Inner	Outer
Ri (m)	0.06913	0.1492
Ro (m)	0.128	0.2111

$$\theta = \sqrt{\frac{R^2}{R_b} - 1} \qquad (7.10)$$

The angle can then be used in Equation 7.11 to find the arc distance from the involute origin.

$$\Delta s = \frac{R_b \theta^2}{2}$$ (7.11)

The control volume for each internal grid point is then defined as

$$CV = \frac{\Delta s_{i+1} - \Delta s_{i-1}}{2} \times \frac{z_{j+1} - z_{j-1}}{2} \times 0.000762$$ (7.12)

where i,j represents a nodal position in Fig. 6.1, z represents an axial position, and 0.000762 m is the thickness of the fuel-filler combination. Appropriate half and quarter volumes are calculated for edge and corner grid points, respectively.

The control volume is then multiplied by the weight for a given node. The radial products are summed and divided by the total volume of the involute prism that encompasses all the radial nodes for the axial location. This quotient is the volume averaged multiplier for the given axial node. Table 7.6 shows the results for the inner and outer fuel sections.

Table 7.6. Volume averaged multipliers for inner and outer involute

Axial location (m)	Inner multiplier	Outer multiplier
0	1.156	1.400
0.014	0.781	0.638
0.034	0.785	0.617
0.054	0.852	0.654
0.074	0.947	0.745
0.094	1.036	0.847
0.114	1.118	0.943
0.134	1.191	1.031
0.154	1.257	1.113
0.174	1.312	1.191
0.194	1.361	1.262
0.214	1.392	1.319
0.234	1.406	1.350
0.254	1.404	1.350
0.274	1.393	1.320
0.294	1.362	1.266
0.314	1.312	1.191
0.334	1.252	1.106
0.354	1.183	1.019
0.374	1.112	0.928
0.394	1.035	0.833
0.414	0.953	0.734
0.434	0.867	0.628
0.454	0.777	0.526
0.474	0.712	0.471
0.494	0.721	0.474
0.508	1.196	0.868

Before these axial multipliers are used, an energy balance must be checked. Similar to the process followed for the first check with the data in Table 7.1, the total heat generation in the inner and outer sections is calculated. For the outer fuel section there are 369 plates with an active fuel arc length of 0.071374 m (2.81").

The inner fuel section is generating 30.97 MW, and the outer fuel section is generating 51.6 MW for a total of 82.6 MW. The total energy balance is 2.4% higher than the nominal generation of 80.7 MW. The difference is attributed to the round off error. As reference information, the value of 82.6 MW is essentially the value contained in the HFIR safety analysis report (SAR) indicating that 97.5% of the energy is deposited in the assembly (82.9 MW). Any discrepancy between the documented HFIR RELAP5 model power distribution and the tabulated power distribution as documented by the HFIR SAR is outside the scope of this research.

Several conclusions can be drawn from this analysis. First, the fuel sections do not distribute the power in accordance with the fuel volume ratios. The inner plate generates 37.5% of the power but only contains 34 % of the fuel. The inner fuel volume percentage is 34.1% when using the volumes calculated in the finite volume method as compared to the volume from the RELAP5 source. The total active fuel volume for the HEU core is 4.5% larger than the volume calculated from the RELAP5 reference volumes.

The second conclusion is that the axial distribution for the inner fuel plate must be considered in tandem with the distribution for the outer fuel plate. Only considering one or the other does not provide enough information to justify the power distribution.

To correct for the 2.4% error in the energy balance, the axial multipliers can be scaled to preserve the percentage that each section contributes to the total power. The inner section should generate 37.5% of 80.7 MW which is 30.26 MW. This output is 0.977 of 30.97 MW. Therefore, the axial multipliers can be scaled by 0.977. The scaling factor is also the same for the outer fuel section multipliers. This scaled set of multipliers is shown in Table 7.7.

Table 7.7. Scaled axial multipliers for inner and outer plates

Axial location (m)	Inner multiplier	Outer multiplier
0	1.130	1.369
0.014	0.763	0.623
0.034	0.767	0.603
0.054	0.832	0.639
0.074	0.925	0.729
0.094	1.012	0.828
0.114	1.092	0.922
0.134	1.163	1.008
0.154	1.228	1.088
0.174	1.282	1.164
0.194	1.329	1.233
0.214	1.360	1.289
0.234	1.373	1.319
0.254	1.372	1.319
0.274	1.361	1.291
0.294	1.330	1.237
0.314	1.282	1.164
0.334	1.223	1.081
0.354	1.156	0.996
0.374	1.086	0.907
0.394	1.011	0.814
0.414	0.931	0.717
0.434	0.847	0.614
0.454	0.759	0.514
0.474	0.695	0.460
0.494	0.705	0.463
0.508	1.169	0.849

A comparison of the volume average axial multipliers for the inner plate and the assumed distribution in ref-7 is shown in Fig. 7.6. The volume averaged multipliers account for the up-turn in power at the leading and trailing edge of the fuel plate. The center peak of the volume averaged multiplier is lower and the total power for the inner plate is about 10% less. The average multiplier for the volume averaged method is 1.081 compared to 1.186 for the distribution used in ref-7.
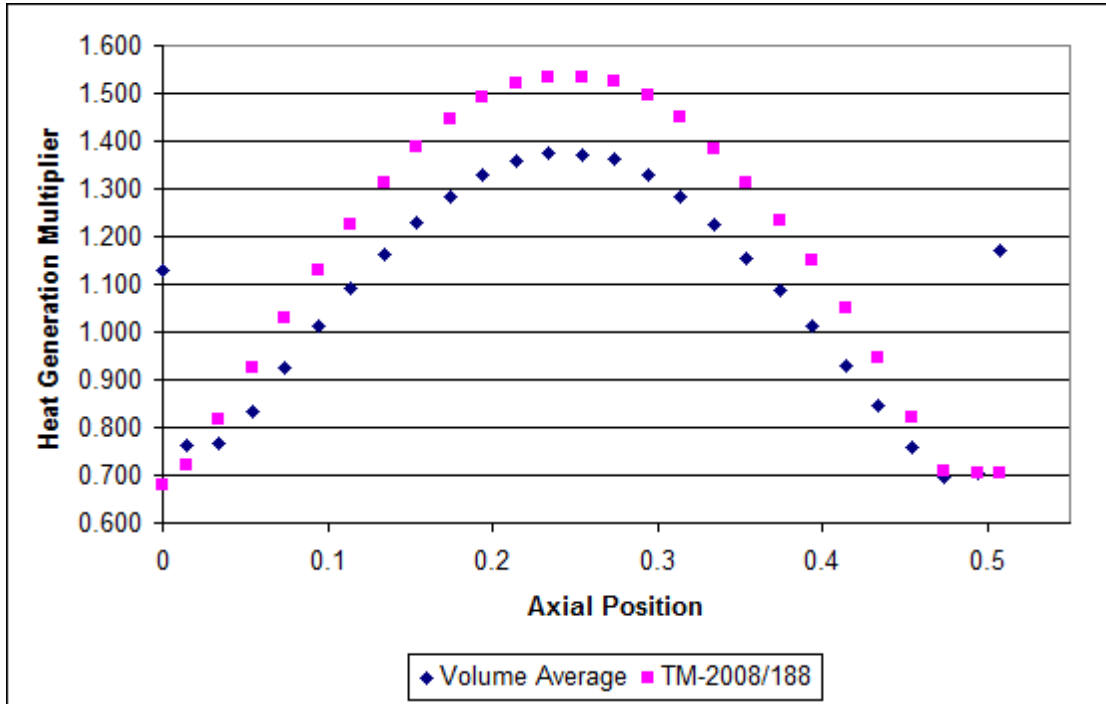
Fig. 7.6. Comparison of volume averaged and TM report multipliers.

These differences are significant and indicate a need to establish a format for transmittal of geometry and power density from neutronics software to the 2D representation in the COMSOL code.

7.4   MULTIPLIER MODIFICATION AND RESULTANT POWER DENSITY BASED ON U_3O_8-AL CERMET GENERATION ONLY

It has been mentioned earlier in this report that a power density analysis should be carried out where only the uranium fuel, hereafter referred to as the fuel, is generating thermal energy. This type of analysis presents many challenges as the fuel has a contour, on one side, relative to the involute fuel plate as shown in Fig. 7.7. Previous analysis used both the fuel and the filler as generating material which presented a constant cross sectional area, and therefore, a simpler volume calculation.

The calculation of the fuel volume involves integration of the fuel contour. Thus some analytic function of the contour must be found before integration is possible. Fortunately, coordinate data for the contour exists relative to the involute for both inner and outer fuel plates in HFIR drawings D-42114 [8] and D-42122 [9], representing the inner and outer fuel plates respectively. Both sets of coordinate data were imported into MATLAB and fitted by cubic spline interpolation. The coordinate data for the inner fuel and outer fuel and their cubic spline interpolants are shown below in Figs. 7.8 and 7.9, respectively.

Fig. 7.7. Involute arc of the fuel plate.

(Notice that the fuel and filler combination has a constant cross sectional area, while the fuel itself has a contour relative to the involute arc.)

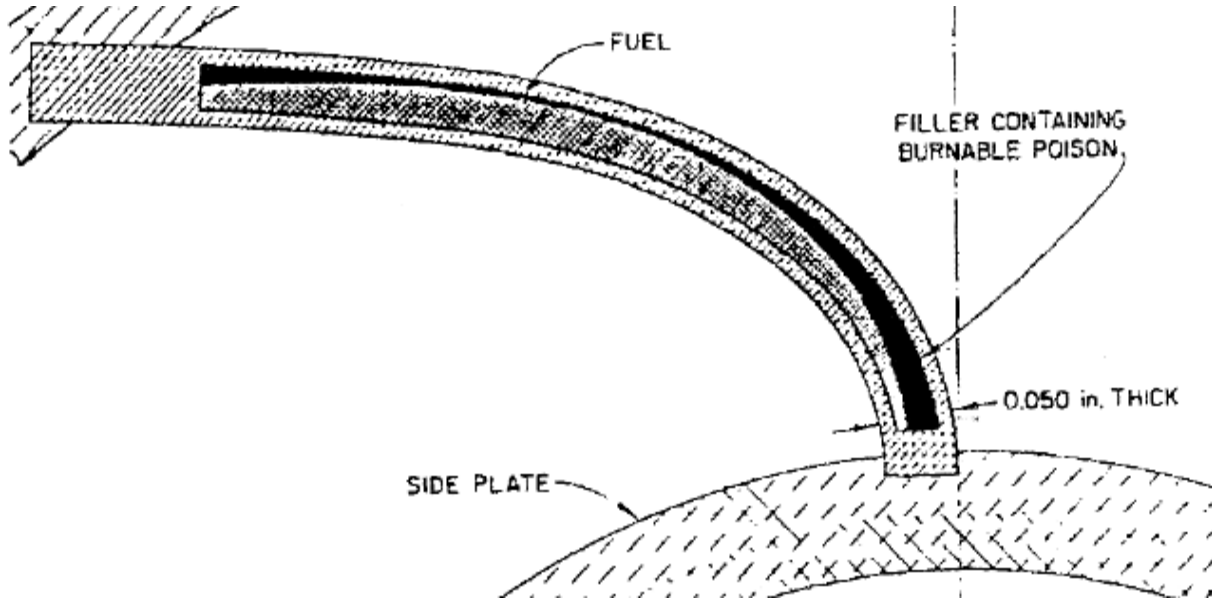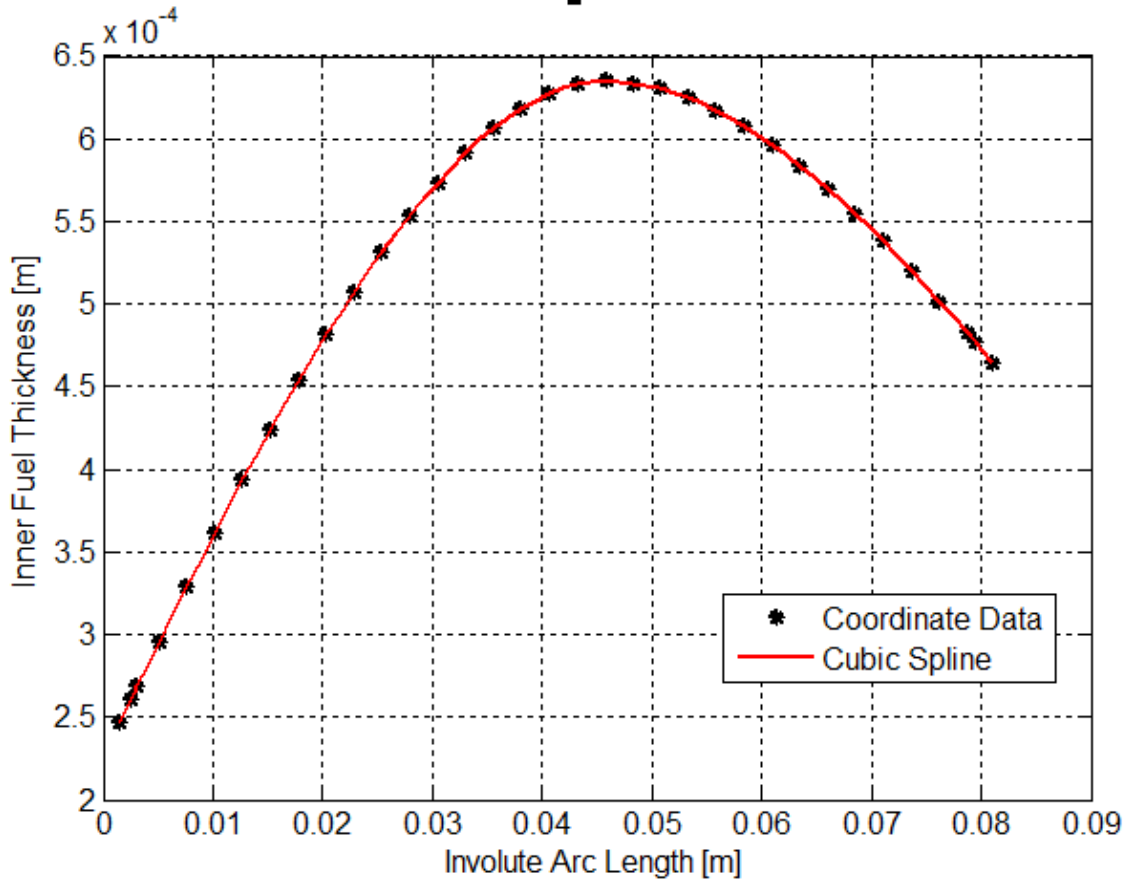**Fig. 7.8. Coordinate data for the inner fuel contour relative to the involute and its cubic spline interpolant.**
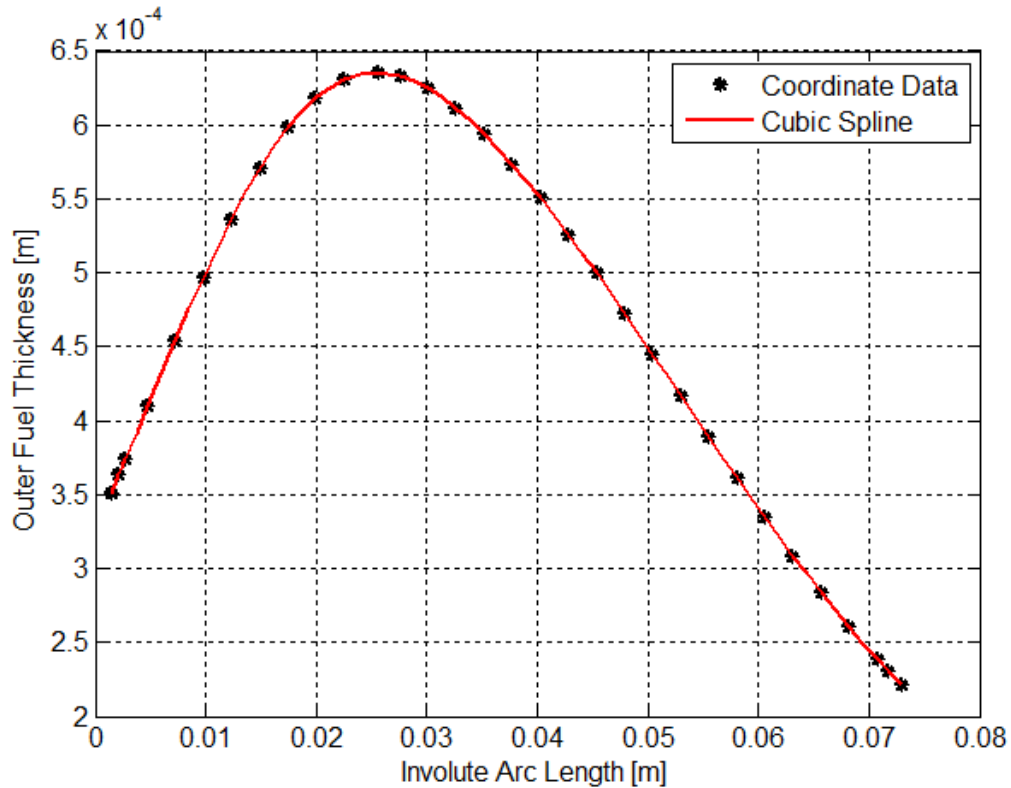
Fig. 7.9. Coordinate data for the outer fuel contour relative to the involute and its cubic spline interpolant.

From Figs. 7.8 and 7.9, it is obvious that the cubic splines do indeed fit the coordinate data very well and thus one should be confident that the integration of these functions will represent the true (at least to within manufacturing tolerances) cross sectional area of the fuel with minimal error. The integration of the cubic splines was carried out numerically using Simpson's rule between two adjacent data points over the entire contour. The cross sectional area was then multiplied[4] by the fueled length of the core, 0.508 m, which yields an inner fuel volume of 0.00364 m^3 and an outer fuel volume of 0.00639 m^3. The combination of these two volumes gives the total active fuel volume of the core to be 0.01003m^3. The active fuel volume per plate is 2.1287×10^{-5} m^3 and 1.7317×10^{-5} m^3 for the inner and outer plates, respectively.

For completeness, this analysis was carried out for both the an 80.7 MW core, as modeled in the RELAP5 model of HFIR, and then again for a 85 MW core as a sensitivity study. To properly represent the total thermal energy load of the core it is necessary to consider both inner and outer fuel contributions to this total.

From Equation 7.8, the generation rate of the 80.7 MW core is 8.0459×10^9 W/m^3. The contribution of the inner core to the total core power is 2.9287×10^7 W or 36.29%, while the contribution from the outer core

---

[4] This assumes that the taper at each end of the fuel region of the plate shown in drawings D-42114/RJ and D-42122/RJ is such that it is manufactured perfectly and the taper is zero. This assumption is consistent with that used in the 3D analysis for the HEU fuel whereby the fuel has the same contour from top to bottom and no axial grading. This assumption must be validated for the LEU fuel design once the manufacturing process is established .

is found to be 5.1413×10⁷ W or 63.71% of the total core power. On a per-plate basis the inner and outer fuel plates produce 1.7127×10⁵ W and 1.3933×10⁵ W, respectively. Similar calculations are performed on this information as was done in Section 7.2.

The power density multiplies both a normalized set of multipliers and a modified set. The normalized set is the same as that presented in Table 7.4 above. The modification procedure is on a control volume basis similar to that outlined in the previous section with the exception that the sum of the elemental control volumes makes up the volume of the fuel not the fuel-filler combination.

The power density distribution at radial position 6 using the normalized set of multipliers is shown below in Fig. 7.10.
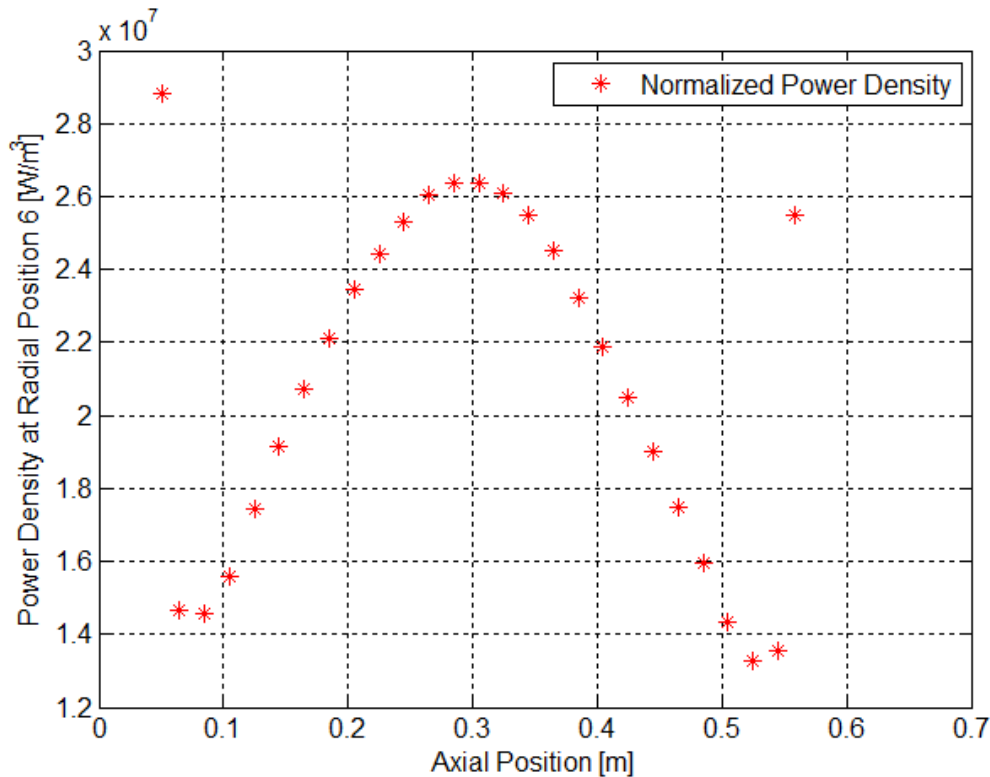
Fig. 7.10. U₃O₈ normalized power density profile at radial Position 6 for the 80.7 MW core.

The multiplier modification procedure uses the power density found by Equation 7.8. The power density is distributed through the use of the multiplier array in Table 7.2. Using the nodal position data supplied in the SSHTC, congruent with Fig. 6.1, elemental control volumes were created around each node. Each nodal power density is multiplied by its appropriate control volume to give an elemental contribution to the total power of the plate denoted as $q_{multiplier\ total}$ in Equation 7.13,

$$q_{multiplier\ total} = \sum_{i=1}^{n}\sum_{j=1}^{m} q_{i,j} = \frac{q_{plate}}{V_{plate\ fuel}}\sum_{i=1}^{n}\sum_{j=1}^{m} \varphi'_{i,j} V_{i,j} \tag{7.13}$$

noting that q_{plate} for an inner element plate is different from the value for an outer element plate. As discussed earlier, the sum of the array entries, $\varphi_{i,j}$, in Table 7.2 is larger than 1. Thus one expects the sum

in Equation 7.13 to be larger than q_{plate} and thus the core total to be larger than Q.

The calculation of the appropriate multiplier scaling factor is accomplished by taking the ratio of the thermal energy dissipated by each plate to Equation 7.13. This value then multiplies φ' to give a scaled set φ. Thus the total core power is preserved. Figure 7.11 below depicts the modified power density profile for radial position 6 of an inner fuel plate.
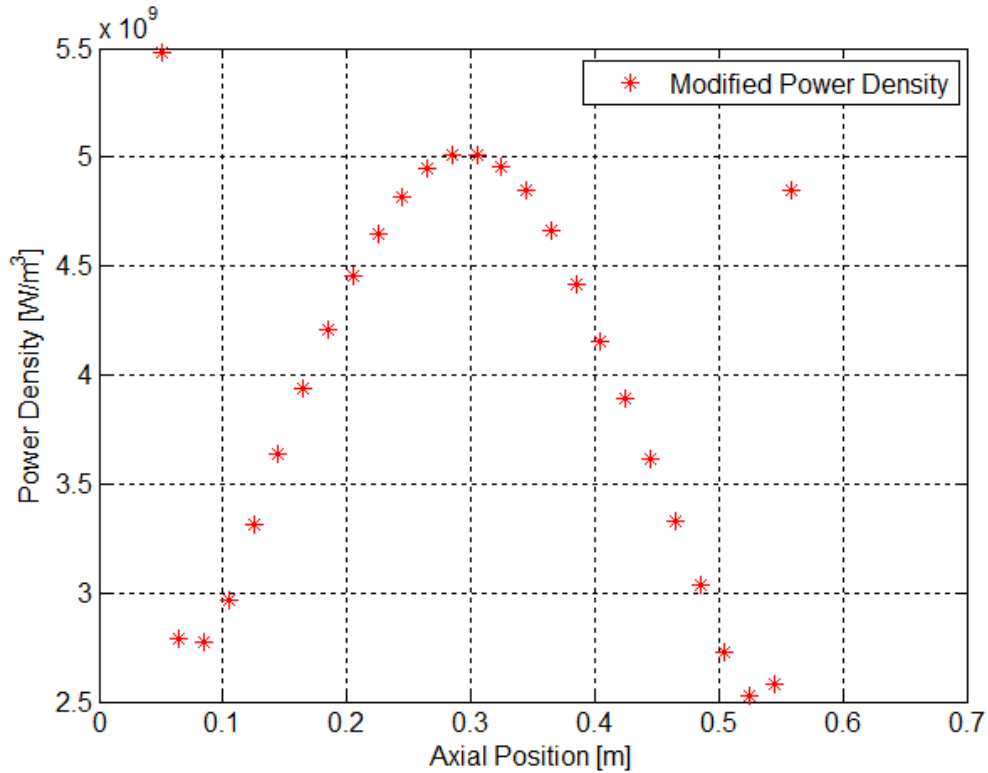
Fig. 7.11. U$_3$O$_8$ modified power density profile at radial Position 6 for the 80.7 MW core.

For the 85 MW core, the generation of thermal energy is 8.4749×10^9 W/m^3, which yields an outer core contribution of 5.4153×10^7 W and an inner core contribution of 3.0848×10^7 W. On a per plate basis the outer core dissipates 1.4676×10^5 W per plate and the inner core dissipates 1.8039×10^5 W per plate.

With this information, both the normalized power density distribution and the scaled power density distribution can be calculated using the procedures of this sub section. These distributed power densities are shown below in Figs. 7.12 and 7.13, respectively.
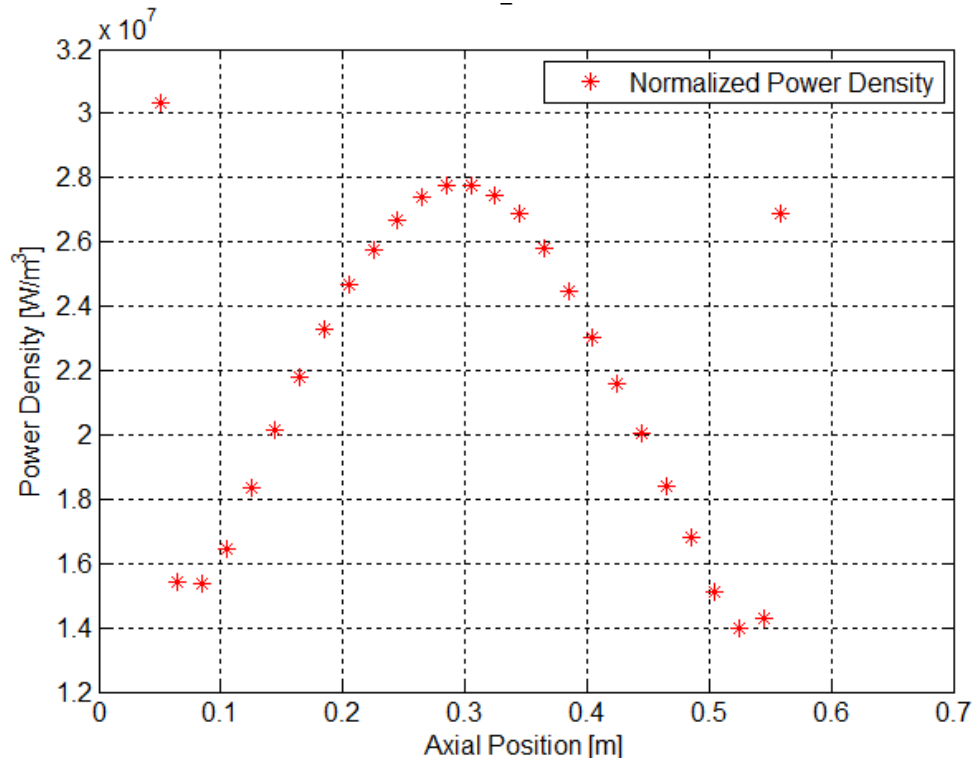
Fig. 7.12. U₃O₈ **normalized power density distribution for the 85 MW core at radial Position 6.**
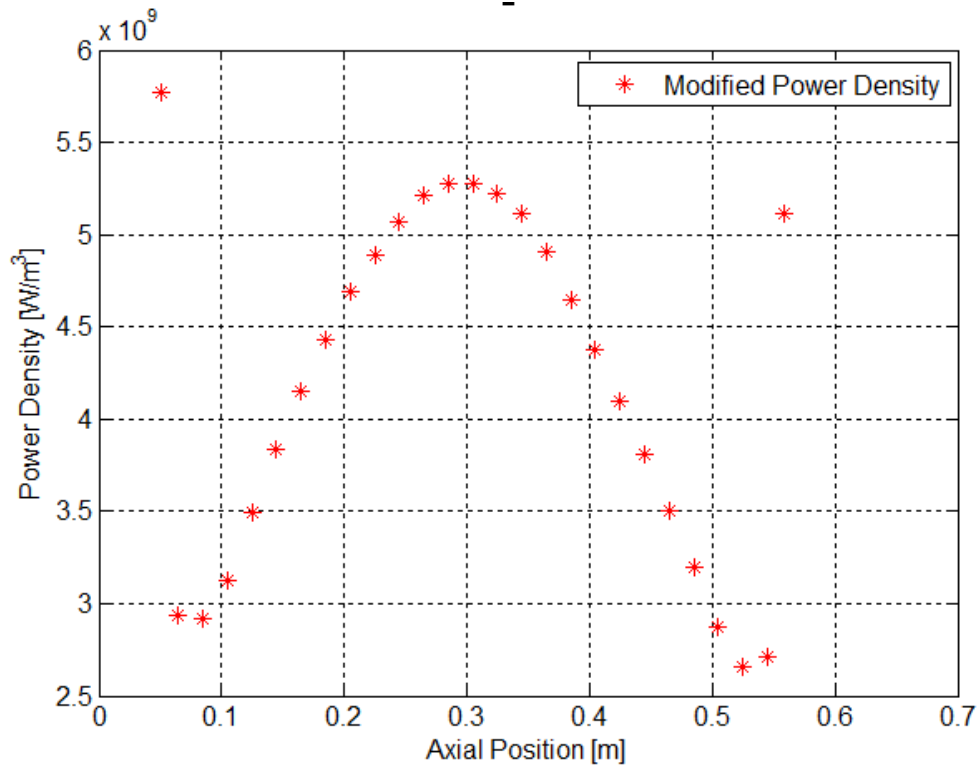
**Fig. 7.13. U$_3$O$_8$ scaled power density distribution at radial Position 6
for the 85 MW core.**

The SSHTC code is very complex and the documentation that accompanies the code is not sufficient by today's standards. Therefore, examining the output of the SSHTC code alone will not enable the analyst to fully comprehend all the details that are needed to arrive at a consistent approach between the COMSOL model and the SSHTC. To compound the situation, it must be realized that the SSHTC code FORTRAN source contains a minimal level of descriptive comments.

8. IMPLEMENTATION OF THE POWER DENSITY PROFILE WITHIN A SINGLE SUBDOMAIN IN COMSOL

The model described in ref-7 separates the fuel region into 27 separate subdomains to accommodate the power density profile discussed in Section 7.1 above. Each independent subdomain has an axial length equal to the distance between two adjacent nodes shown in Fig. 6.1 above. This approach is questionable and it has been found that it is more efficient to create a single subdomain and implement a distributed power density in that single subdomain. To this end a single fuel domain was created and the power density profiles of the last section were entered as thermal source terms in the COMSOL environment.

COMSOL accommodates user defined data in tabular or analytic (i.e., continuous or piecewise) form. Because the COMSOL input data was created from solution vectors, for a specific radial location, in MATLAB, only tabular data entry into the COMSOL environment will be considered here[5].

The normalized and modified power densities of Section 7 are added to the COMSOL data interpolation algorithm via a text file created from the MATLAB output. The appropriate syntax used in the text files is as follows:

%Grid
x_1 x_2 x_3 • • • x_n

%Data
y_1 y_2 y_3 • • • y_n

where the Grid is the independent variable to be represented and the Data is the dependent variable to be represented. Thus for this specific case (i.e., plate segment at radial position 6), x is the axial position of the jth node and y is the power density at that node. Notice that both solution vectors have the same number of entries.

Once the data is read in from the text file, several options for representation are available. Given the 1-D character of the SSHTC model, nearest neighbor interpolation is chosen which produces a constant magnitude of the input over a region specific to the input akin to the zero axial diffusion assumption of the SSHTC. An extrapolation option is also given to project beyond the bounds of the data, which is set to zero in this case given that the clad generates no thermal energy. The profile created by COMSOL for the power density at the sixth radial position of the 80.7 MW core inner plate is shown below in Fig. 8.1.

---

[5]For more information on user defined data entry into the COMSOL environment see Appendix C.
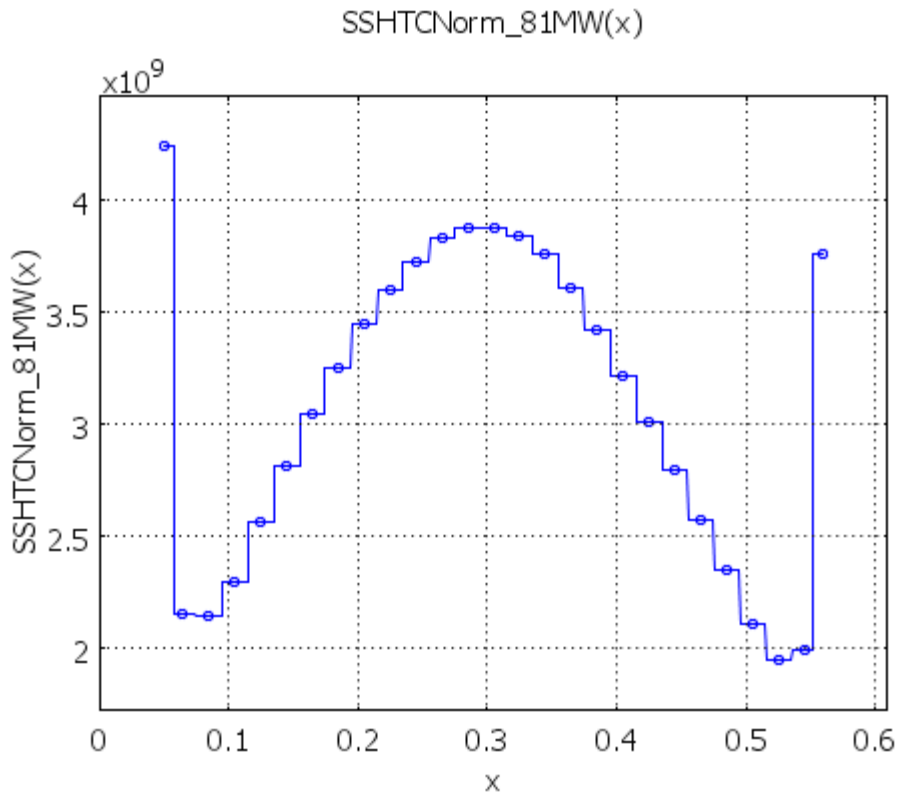
Fig. 8.1. COMSOL nearest neighbor interpolation of the normalized array power density profile for the 80.7 MW core.

As seen in Section 7, the profiles for all core loads and multiplier manipulations retain their relative shape, thus all profiles will have the shape shown in Fig. 8.1 and vary only in magnitude. Text files are created for all core loads and multiplier manipulations and used in their respective COMSOL thermal model for comparison with the output of the SSHTC.

9. COMPARISON OF THE SSHTC-BASED COMSOL THERMAL-HYDRAULIC MODELS WITH THE SSHTC

The SSHTC output file, *reference.output,* was used to provide information as input into COMSOL and as an information base for comparison. The SSHTC outputs a film coefficient, at the clad-coolant interface, through the use of the modified Hausen Nusselt number correlation. Figure 9.1 below shows the SSHTC output for the film coefficient as a function of axial position. There is an apparent discontinuity in the film coefficient caused by the input power profile which has been altered by either (1) a step change in the power density due to the position of the safety plates relative to the control cylinder at the simulated point in time of the reactor cycle, or (2) a conservatively imposed power factor (U_4 and U_5) which has been artificially introduced in the SSHTC analysis to evaluate the worst-case effect of power shifts on the structural integrity of the fuel elements, or (3) both (1) and (2). [The typical values used for U_4 are 0.9 for j=1,16 and 1.12 for j=17,31. The typical values used for U_5 are 1.1 for j=1,16 and 0.88 for j=17,31] This type of conservative imposition of the power factor is not used in the present analysis, but could certainly be evaluated with COMSOL in future analyses.

The sixth radial position of the fuel plate is modeled in all cases in this section. The COMSOL models differ only in the power density profile within the fuel domain. This radial position was chosen because of the minimal influence the filler material has on the results, see Fig. 7.7
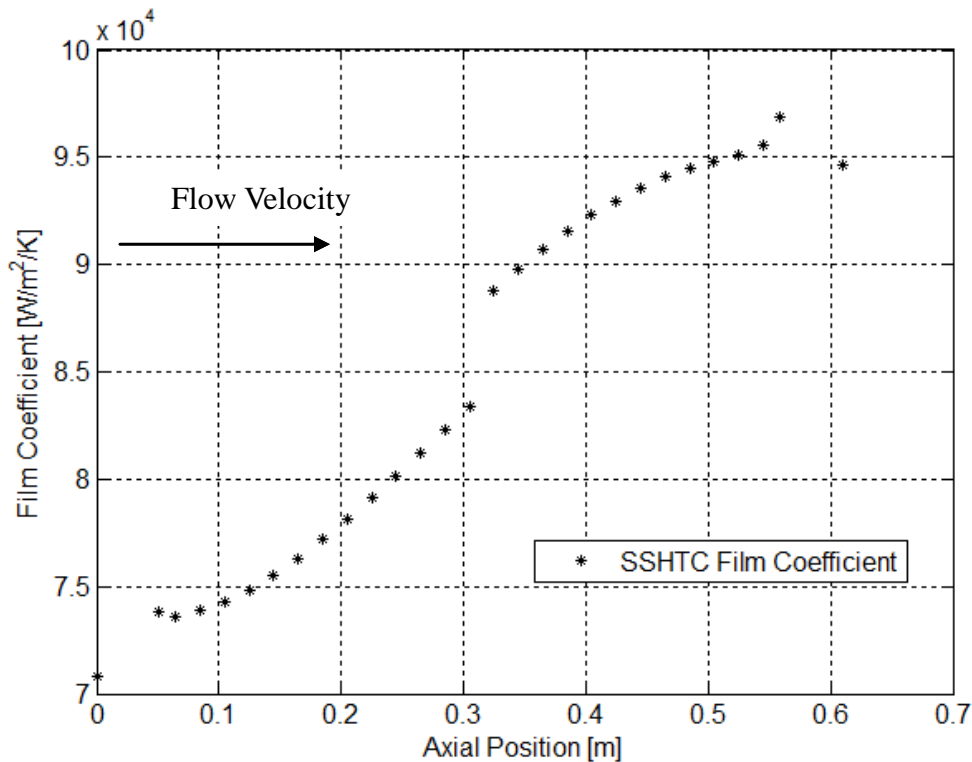
Fig. 9.1. SSHTC film coefficient at radial Position 6 as a function of axial position used in COMSOL.

The SSHTC output file also provides the bulk water temperature, at all positions *i,j,* for the coolant through the enclosed flow channel. This bulk water temperature distribution for the axial strip at the sixth radial position is shown in Fig. 9.2.
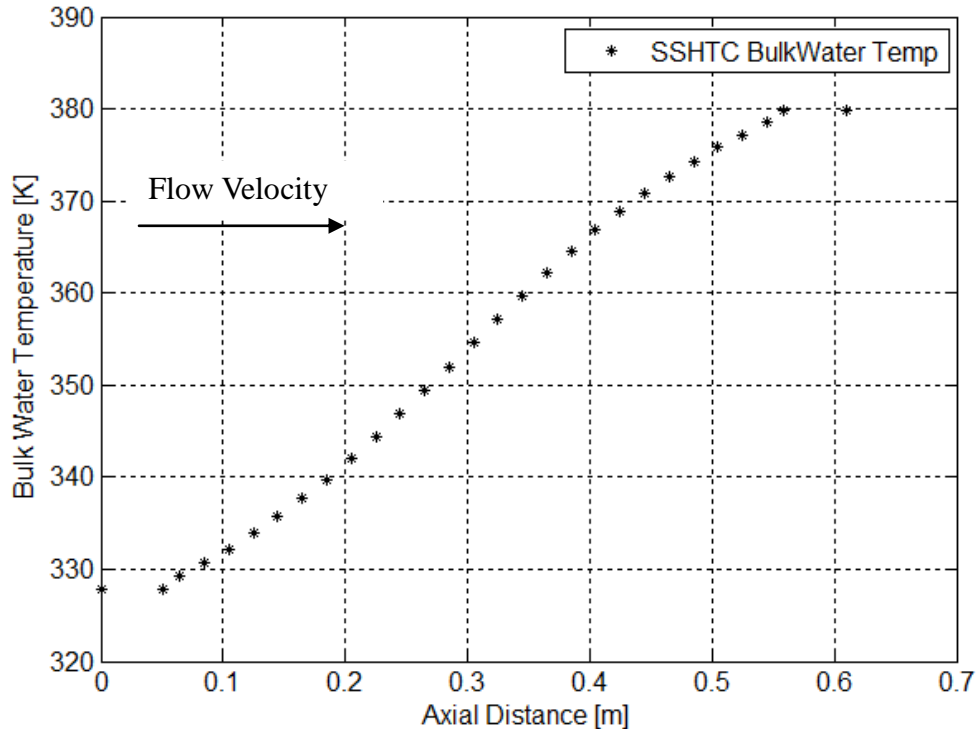
**Fig. 9.2. SSHTC calculated bulk water temperature at radial Position 6
as a function of axial position used in COMSOL.**

The SSHTC numerical results of both the film coefficient and the bulk water temperature distribution are
used as tabular data for input to the COMSOL models in an effort to more closely mimic the SSHTC.

Figure 6.1 shows the solution lattice used by the SSHTC. Again this lattice is imposed on the clad surface
and does not account for any phenomena upstream or downstream of the fuel plate. Given that the film
coefficient and bulk water temperature are specified, it is unnecessary and counterproductive to consider
fluid flow in the validation of COMSOL models. Thus the thermal models created for comparison have
been reduced to conduction heat transfer analysis in the solid fuel, filler, and clad regions for each axial
node of the strip at radial position 6 with 3 global adiabatic boundaries and a single global convective
boundary. The fuel-clad boundaries are given the default continuity condition, $G_{in} = G_{out}$, with G being
any applicable general physical property.

To mimic the SSHTC, it is necessary to construct only the fuel and clad (i.e., solid) domains in
COMSOL. The archetype for the COMSOL thermal models is shown below in Fig. 9.3.
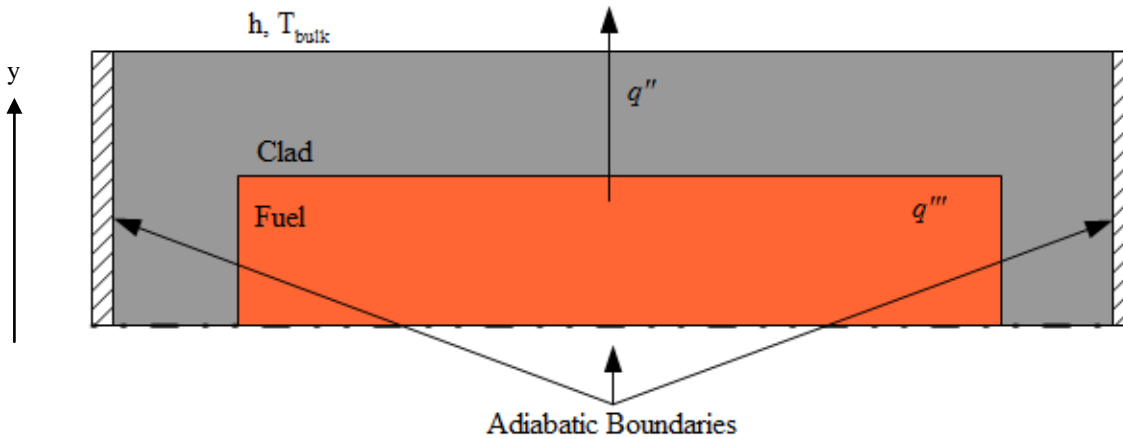
Fig. 9.3. COMSOL thermal model archetype.

> The scale has been greatly exaggerated for visualization. As discussed in Section 3 the aspect ratio, length to width, is 480.

Figure 9.3 shows the geometry and the appropriate boundary conditions used in the comparison of COMSOL with the SSHTC. The leading edge of the clad is not filleted as it was in previous models since that modification serves no purpose here. While this model represents only one axial nodal strip of the fuel plate at a specific radial location, it encompasses all axial aspects of the modeling strategy of the SSHTC for that specific radial location. This axial strip model is permissible solely because the thermal conductivity tensor assumed in the SSHTC has the form presented in Equation 6.2. Had there been additional entries in Equation 6.2, the diffusion of thermal energy would be multi-directional and Fig. 9.3 would not be representative of that effort.

The quantity used for the comparison of the two codes is the clad surface heat flux for an average plate. These values are pulled from *reference.output* and plotted against the COMSOL output for the same quantity. To avoid confusing the reader, the models that are to be compared will be compartmentalized into cases as shown in the Table 9.1.

Table 9.1. Model configuration list

Case	Description
A	80.7 MW core, Normalized Multipliers
B	80.7 MW core, SSHTC Modified Multipliers
C	85 MW core, SSHTC Modified Multipliers
D	85 MW core, U_3O_8 Fuel Generation Only

9.1   NORMALIZED MULTIPLIER ARRAY POWER DENSITY: CASE A

The power density used here is that depicted in Fig. 7.3. Figure 9.4 below shows the temperature field through the fuel plate.
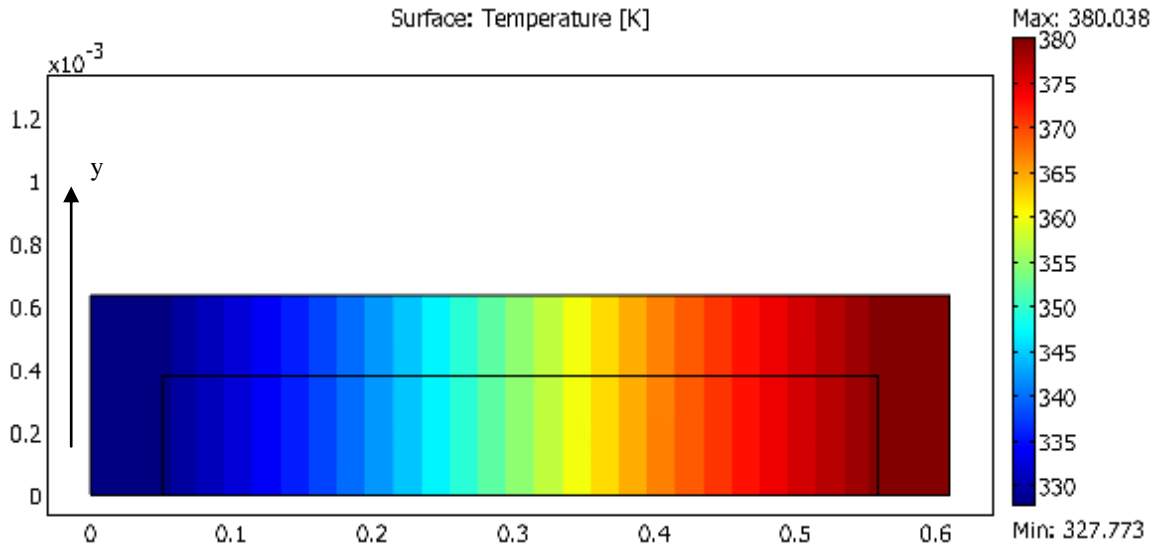
**Fig. 9.4. COMSOL temperature output for the 80.7 MW core
using the normalized set of multipliers.**

The quasi-isothermal rectangular cells observed in Fig. 9.4 are a direct result of the nearest neighbor interpolation used to describe the power density. The error in the conservation of energy is 0.021% found by Equation 5.13. Thus the model provides a sufficiently accurate representation of the energy. This model does not provide sufficient thermal energy to the clad surface as is witnessed in the surface heat flux shown below in Fig. 9.5.
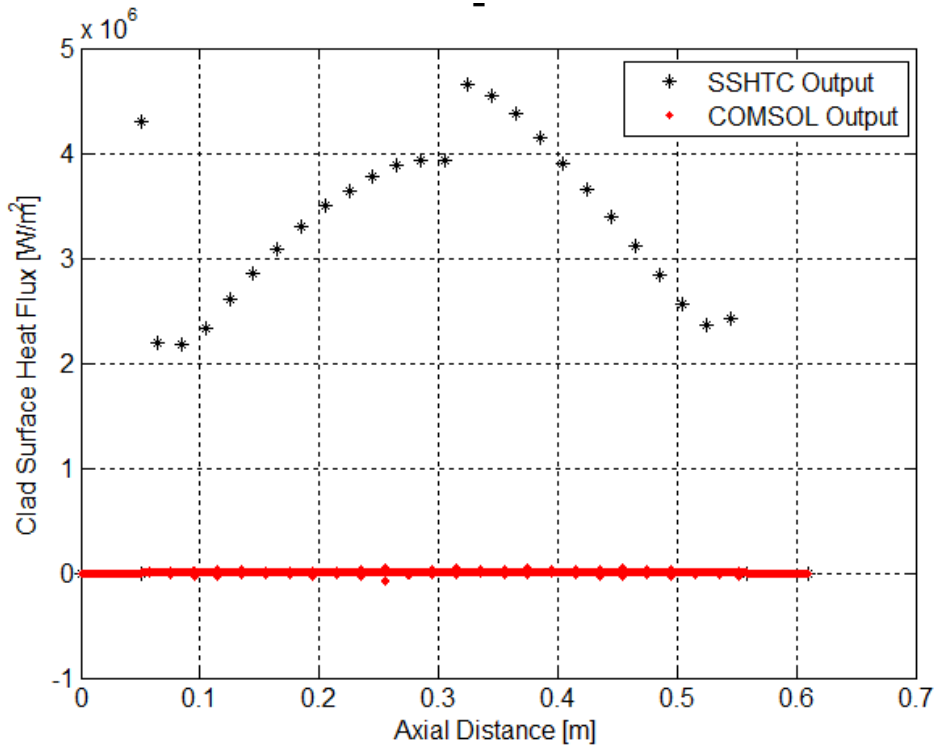
Fig. 9.5. Clad surface heat flux produced by COMSOL and the SSHTC using normalized multipliers.

This model does not provide sufficient thermal energy to the clad surface as is witnessed in the surface heat flux shown below in Fig. 9.5. The discrepancy between the heat flux in this model and that of the SSHTC is approximately 2 orders of magnitude. The scales are so different that the profile of the thermal energy flux from the COMSOL model is not observed. At this point, it is assumed that the normalized set of multipliers is to blame for such a large difference. As seen in Section 7.2, increasing the overall core power to 85 MW does not significantly change the power density using the unit normalized multipliers. Thus this set of multipliers and all thermal information calculated from it will no longer be considered

9.2 MODIFIED MULTIPLIER ARRAY USING SSHTC PROCEDURE: CASES B AND C

These models use the power density profiles produced by the multiplier array shown in Table 7.3.

Case B:

Figure 9.6 below shows the temperature field due to this power density for the 80.7 MW core.
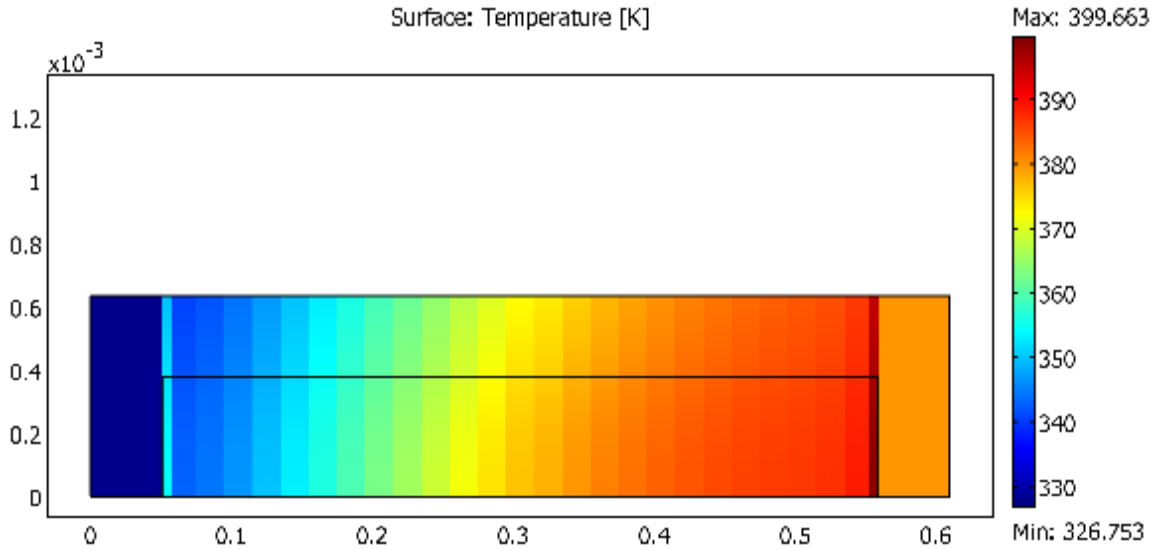
**Fig. 9.6. Temperature field found by COMSOL using the SSHTC multiplier
modification procedure and the 80.7 MW core.**

The change in temperature in the y direction is more pronounced than in the previous model. This is a direct result of increasing the power density with an unchanged bulk temperature distribution. While the modified multiplier set produces more favorable results than the normalized set, the magnitude of the power density is still entirely too low as shown in Fig. 9.8.

Case C:

Increasing the power will, of course, increase the magnitude of the power density. Thus the 85 MW core power density, found by the multiplier array in Table 7.3, is used. The COMSOL results produced by the 85 MW core is shown below in Fig. 9.7.
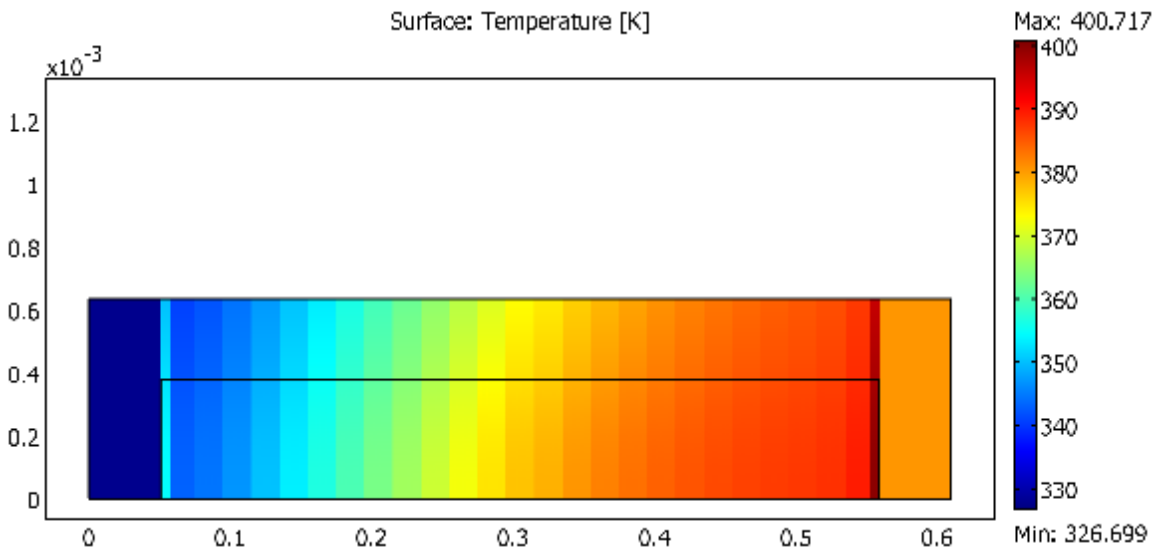
Fig. 9.7. Temperature field for the 85 MW core.

Page 55 of 82

Aside from the slight temperature increase no other change is observed. The clad surface heat flux produced by this model is shown in Fig. 9.8 as well. While this increased power density does have the desired effect it still does not match that of the SSHTC.
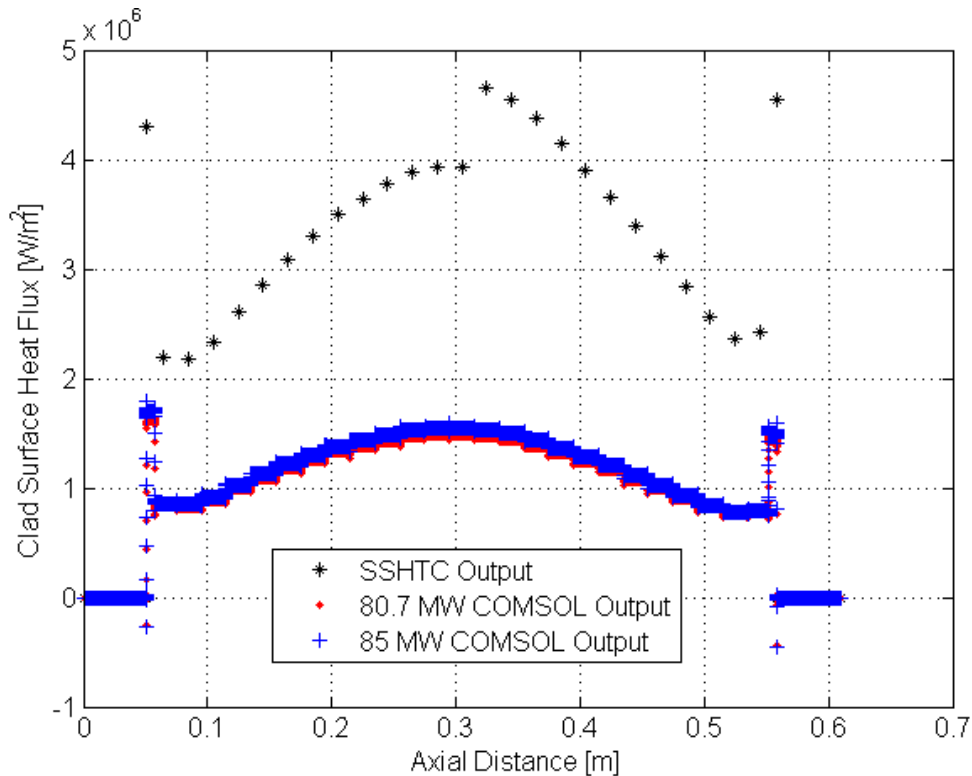
Fig. 9.8. Clad surface thermal energy flux for the 80.7 MW core, the 85 MW core and the SSHTC surface heat flux at radial position 6 for the inner plate.

9.3   POWER DENSITY DUE TO U₃O₈ FUEL VOLUME ONLY: CASE D

The power densities used in this section were found by the procedure outlined in Section 7.4. Because the uranium fuel is producing thermal energy, the COMSOL geometry needed to incorporate a separate subdomain to represent the filler material. The filler material was assumed to have the same thermal conductivity as the uranium fuel, which is a conservative assumption, and zero thermal energy generation. The filler material is aluminum alloy Al 1100. A thermal conductivity value of 218 W/m/K was found for a constant temperature of 25℃. Given the relatively high temperatures found in the HFIR, it did not seem appropriate to include these values in the model. The actual material properties for Al 1100 will be used in later models. The results provided by COMSOL for the 85 MW core using this multiplier modification procedure is shown in Fig. 9.9. The 85 MW core is chosen based on the previous underestimated heat fluxes.
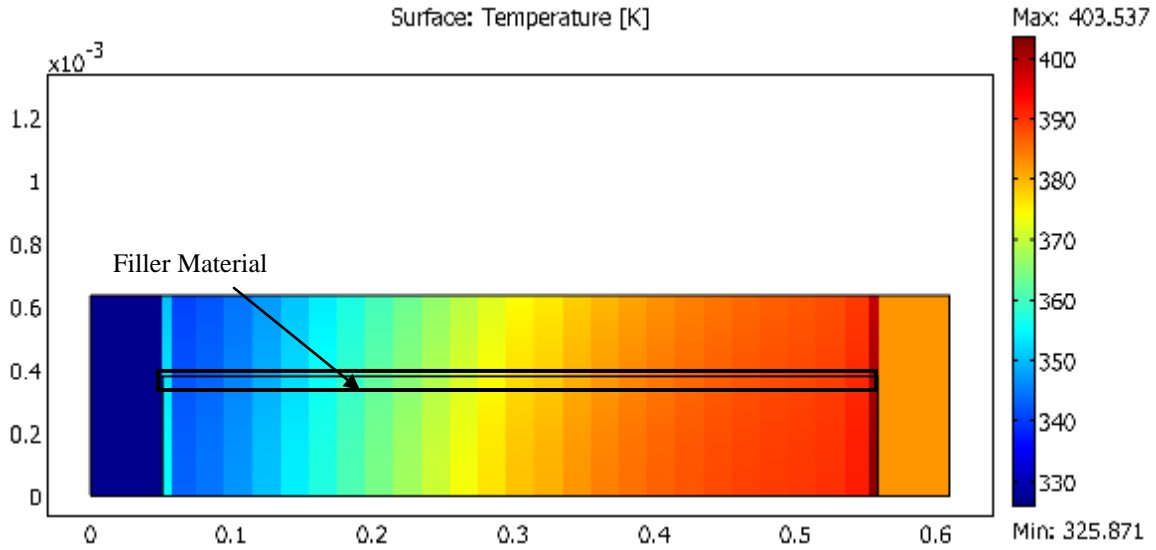
Fig. 9.9. COMSOL generated temperature field for the 85 MW core where only the uranium fuel is generating thermal energy.

The exported image does not resolve the separate filler material domain, thus it has been artificially imposed as a rectangle in Fig. 9.9 to show location and relative size. The global temperature has risen slightly from 400 K to 403 K. However, the heat flux is still too low as shown in Fig. 9.10.
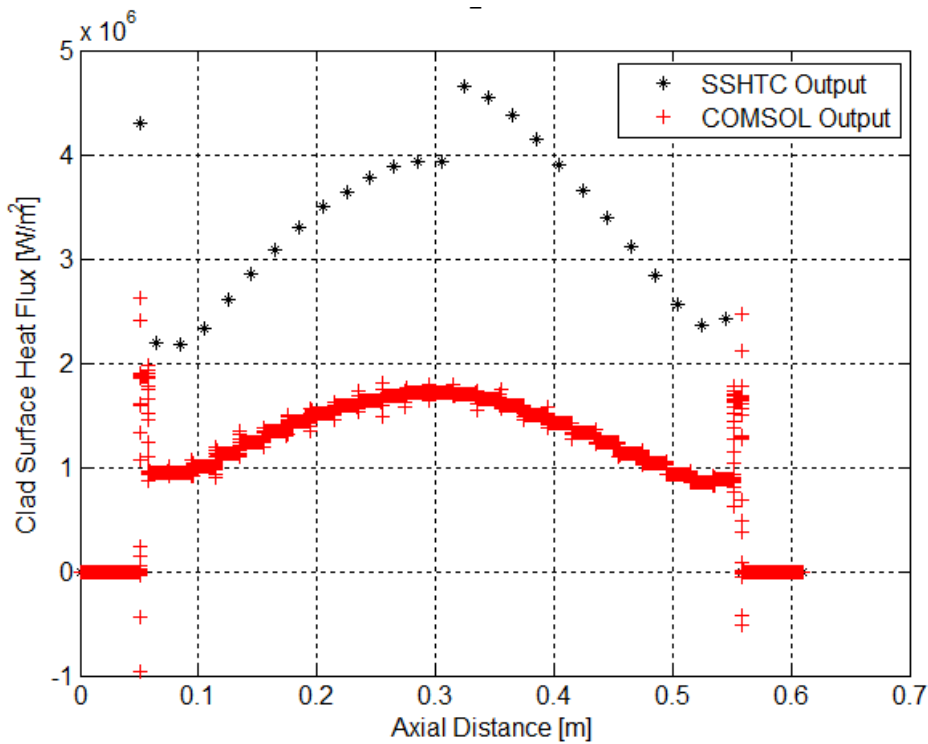
Fig. 9.10. Clad surface heat flux for the 85 MW core.

This case provides the heat flux closest to that given by the SSHTC, that is relative to the other models introduced in this section.

It seems reasonable to assume that the lack of agreement in the clad surface heat fluxes may be due to incorrect usage of the multipliers though it is noted that the SSHTC normalizes the multipliers so that the derived core power is equal to the value input by the user. Further investigation into these issues is necessary for a thorough understanding of the SSHTC.

Even though the clad surface heat flux calculated from COMSOL has been too low in magnitude relative to the SSHTC, the general shape of the heat flux distribution is similar except for the large change in the axial center presented by the SSHTC. This similarity in profile shows that COMSOL is indeed capable of mimicking the SSHTC. The similarity is not limited to surface heat flux. The surface temperature too shares a similar profile to the SSHTC surface temperature as shown in Fig. 9.11.
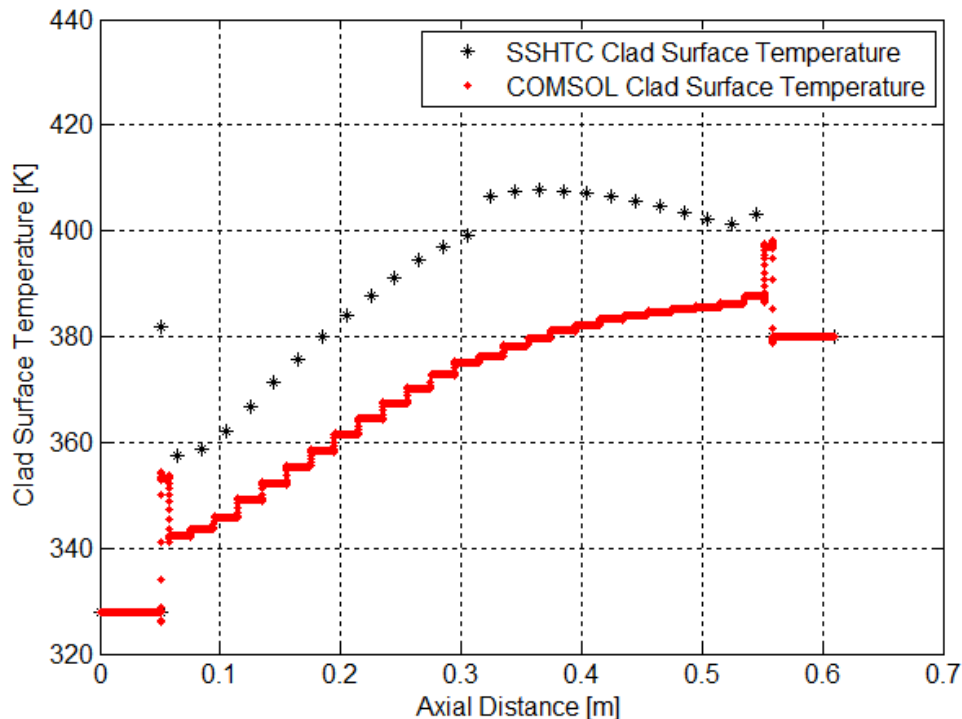
Fig. 9.11. Clad surface temperature profile for the 85 MW core.

Given the unidirectional thermal conductivity and insulated leading and trailing boundaries, the SSHTC temperatures would be higher than the actual surface temperatures occurring in the reactor core, thus the SSHTC is conservative relative to the surface heat flux and temperature. COMSOL has multi-dimensional capabilities and is not limited to such restrictive assumptions such as uni-directional thermal conductivity and specific film coefficients.

9.4 DISCRETIZED FUEL POWER DENSITY BASED ON INDIVIDUAL SURFACE HEAT FLUXES FROM THE SSHTC OUTPUT

It has been established that the SSHTC assumes a unidirectional heat flux from the fuel to the clad surface. Again, this implies adiabatic boundaries in the axial and span-wise directions around each node in Fig. 6.1. This one dimensional characteristic allows one to calculate a power density magnitude corresponding to each clad surface heat flux value given in the *reference.output* documentation. In

previous methods these values were integrated over the clad surface, thus averaging out any abrupt changes in the surface heat flux and the surface temperature. This method, however, accounts for the abrupt changes by taking the discrete surface heat flux values and calculating a discretized power density distribution.

The determination of the implied thermal energy generation rate, for the fuel, was calculated from the surface heat flux data given in *reference.output,* called *QA*. This approach is different from those attempted before in the sense that the heat flux data from *reference.output* is not being integrated. Instead the values, *QA*, are being used individually in the calculation of individual thermal energy generation rates at locations congruent with the axial displacements given in the SSHTC.

The SSHTC allows the thermal energy generated in the fuel to be transported to the fluid in a direction normal to the clad surface only (i.e., no thermal gradients exist in the axial or span-wise directions of the fuel plate). Also, the SSHTC does not treat the involute shape of the fuel plate. Instead, it flattens out the involute into a rectangular geometry (See section 6 of this document). This feature, along with the unidirectional heat transfer, allows one to easily calculate the generation rate in the fuel from the surface heat flux information mentioned above.

The thermal field, within all of the domains in the fuel plate, is governed, in general, by Poisson's equation given in Equation 3.1

$$\frac{d}{dx}\left(-k\frac{dT}{dx}\right) = q''' \tag{3.1}$$

The SSHTC treats the thermal conductivity tensor as unidirectional and constant, thus only one constant entry is maintained which corresponds to the direction normal to the clad surface as stated in Equation 6.2. This allows a simplification to be employed resulting in Equations 3.2 and 3.3, the heat equation for the fuel and clad, respectively.

$$\left(-k\frac{d^2T}{dx^2}\right) = q''' \tag{3.2}$$

$$\left(-k\frac{d^2T}{dx^2}\right) = 0; \tag{3.3}$$

Equation 3.3 may be rearranged to solve for the concavity of the thermal field

$$\frac{d^2T}{dx^2} = \frac{q'''}{-k} \tag{9.1}$$

The heat flux at any point in the fuel may be determined by integrating Equation 9.1. First the second derivative is split up

$$\frac{d}{dx}\left(\frac{dT}{dx}\right) = \frac{q'''}{-k} \tag{9.2}$$

Integration of Equation 9.2 with respect to *x* gives the following result

$$\frac{dT}{dx} = \frac{q'''}{-k} x + C_1 \tag{9.3}$$

Given the adiabatic condition at the fuel centerline, C_1 is found to be zero. From Fourier's Law the magnitude of the heat flux for the specific SSHTC conditions is found by

$$q'' = -k \frac{dT}{dx} \tag{9.4}$$

Substituting Equation 9.3 into Equation 9.4 gives

$$q'' = q'''x \tag{9.5}$$

Thus the heat flux in the fuel is a linearly increasing function of position normal to the clad surface. The situation is different in the clad. Since the thermal field in the clad is governed by Equation 3.3, the temperature profile is linear. This is due to the zero thermal energy generation in that domain. The same analysis can be performed in the clad as was done in the fuel domain. Integrating Equation 3.3, one finds

$$\frac{dT}{dx} = C_2 \tag{9.6}$$

Employing Fourier's Law once again we find that the heat flux through the clad is constant

$$q_2'' = C_2 \tag{9.7}$$

and must equal the clad surface heat flux reported in *reference.output, QA*. Now the heat flux at the fuel-clad interface is known.

As was shown, the heat flux through the fuel is a linearly increasing function of position, and the boundary conditions are known (i.e., $q'' = 0$ at the fuel centerline and $q'' = QA$ at the fuel-clad interface). Now recalling Gauss's Law

$$\nabla \cdot q'' = q''' \tag{5.13}$$

one notices that in the case of the SSHTC Gauss's Law is reduced to

$$\frac{dq''}{dx} = q''' \tag{9.8}$$

which for linearly varying heat fluxes is exactly equal to

$$\frac{q''(x + \Delta x) - q''(x)}{\Delta x} = q''' \tag{9.9}$$

More specifically, for the SSHTC representation, Equation 9.9 is reduced to

$$\frac{QA}{\Delta x} = q''' \tag{9.10}$$

This value is computed for each axial *QA* in *reference.output* at radial position 6 and entered into

COMSOL in tabular form. The model geometry does not incorporate a flow field due to the user specified bulk water temperature and film coefficient data used as inputs into the COMSOL environment from the *reference.output* documentation. Once again, this is strictly a conduction model.
The results are remarkably similar. Figure 9.12 shows a comparison of the heat flux data from *reference.output* and COMSOL.
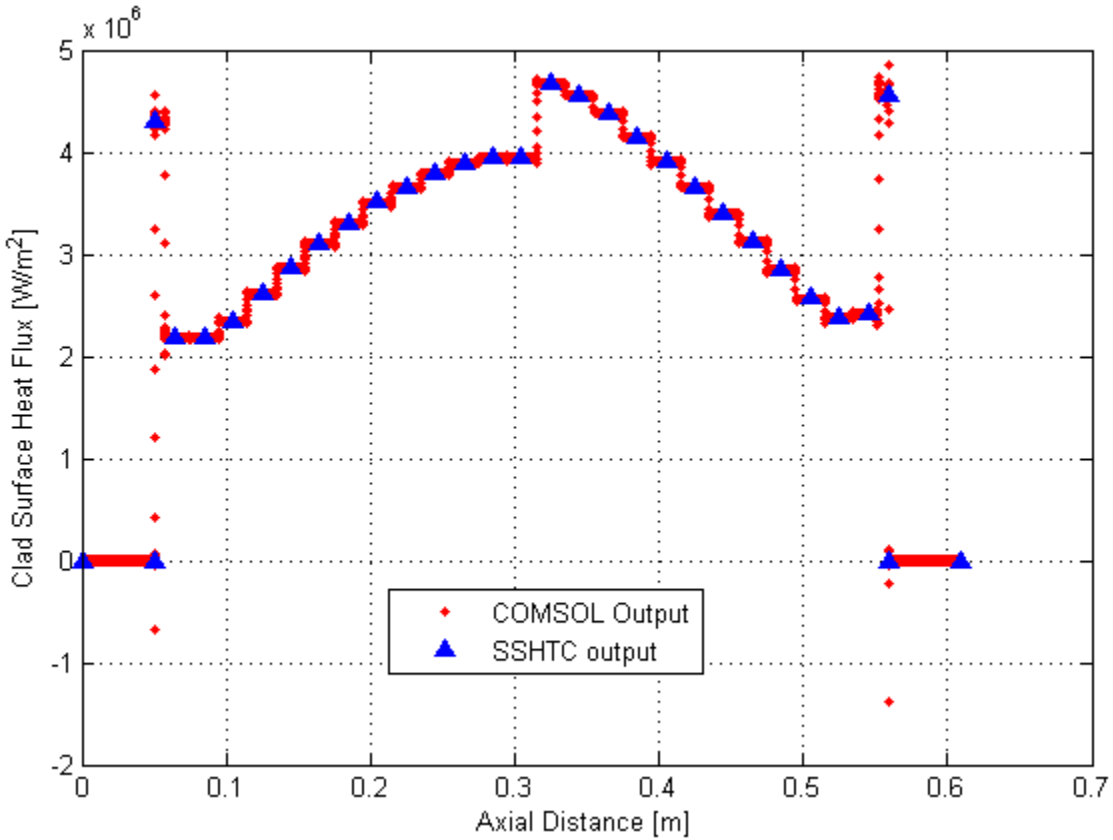
Fig. 9.12. Comparison of SSHTC clad surface heat flux output and COMSOL clad surface heat flux using the power density determination method of the present section.

COMSOL predicts heat fluxes that are very close to those presented in the *reference.output* documentation. Figure 9.13 shows the companion clad surface temperatures from the *reference.output* documentation and the COMSOL model.
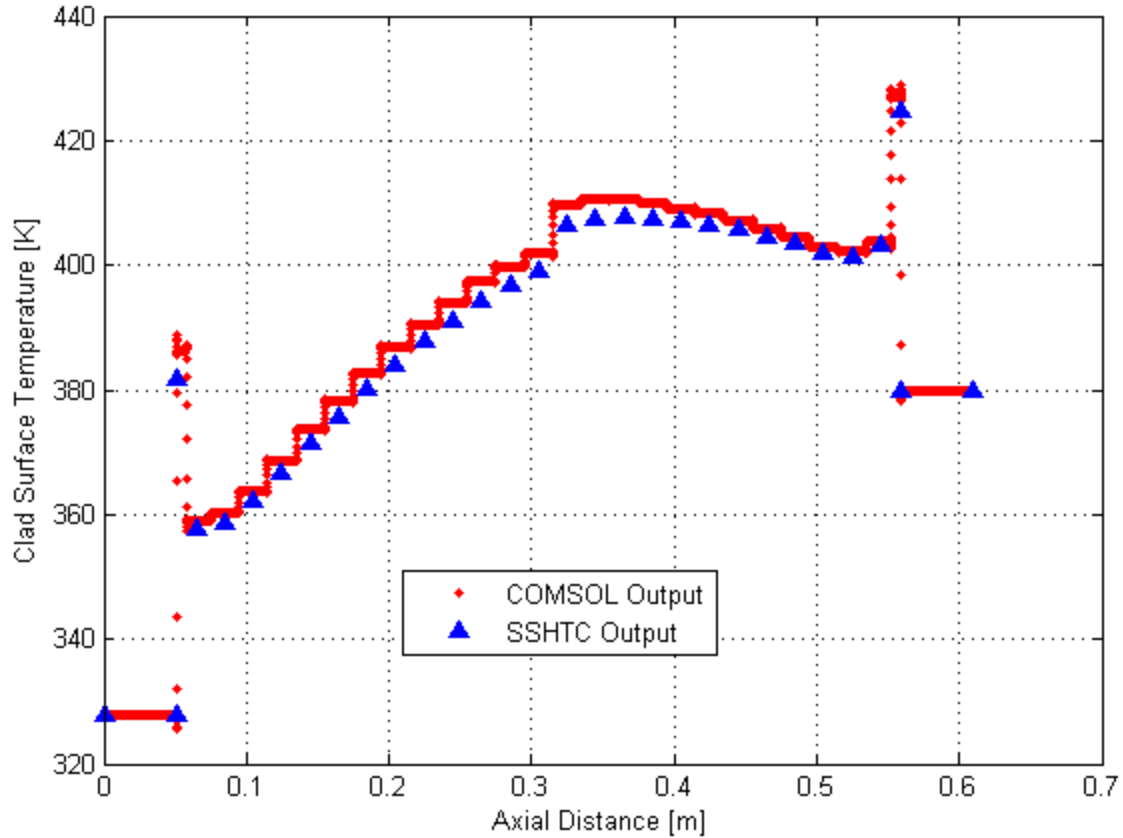
Fig. 9.13. Comparison of SSHTC clad surface temperature output and COMSOL clad surface temperature using the power density determination method of the present section.

COMSOL slightly over-predicts the surface temperature, but this should not be cause for concern as COMSOL is producing conservative results.

10. COMSOL BASED 2D THERMAL-FLUID MODEL COMPARED WITH THE SSHTC OUTPUT.

In Section 9, the SSHTC based COMSOL models were used for comparison with the SSHTC output. These models are one dimensional in the sense that no axial or radial conduction is allowed within the solid domains. All generated heat goes through the clad to the fluid. In this section, a 2D COMSOL model is developed that allows diffusion in the solid both axially and radially through the plate. The power density used is similar to that used in Fig. 9.12 with the exception of the method of interpolation. For this model, it was desirable to be as physically accurate as possible. Thus cubic spline interpolation was employed. In effect, with this model, the restrictions of specifying a film coefficient and bulk water temperature are lifted. These quantities will be part of the solution (output) of the 2D model.

A geometrically similar model to that shown in Fig. 5.4 was created that incorporates a filleted leading edge to the fuel plate and both an entrance and exit length. The exit length has been significantly extended so that a uniform pressure distribution along the exit boundary is realized. This is more consistent with the physical setup of the HFIR fuel plate-coolant channel combination. The power density profile discussed in Section 9.4 was also used in that model. The solution to the thermal-fluid interaction for that model is shown in Fig. 10.1.
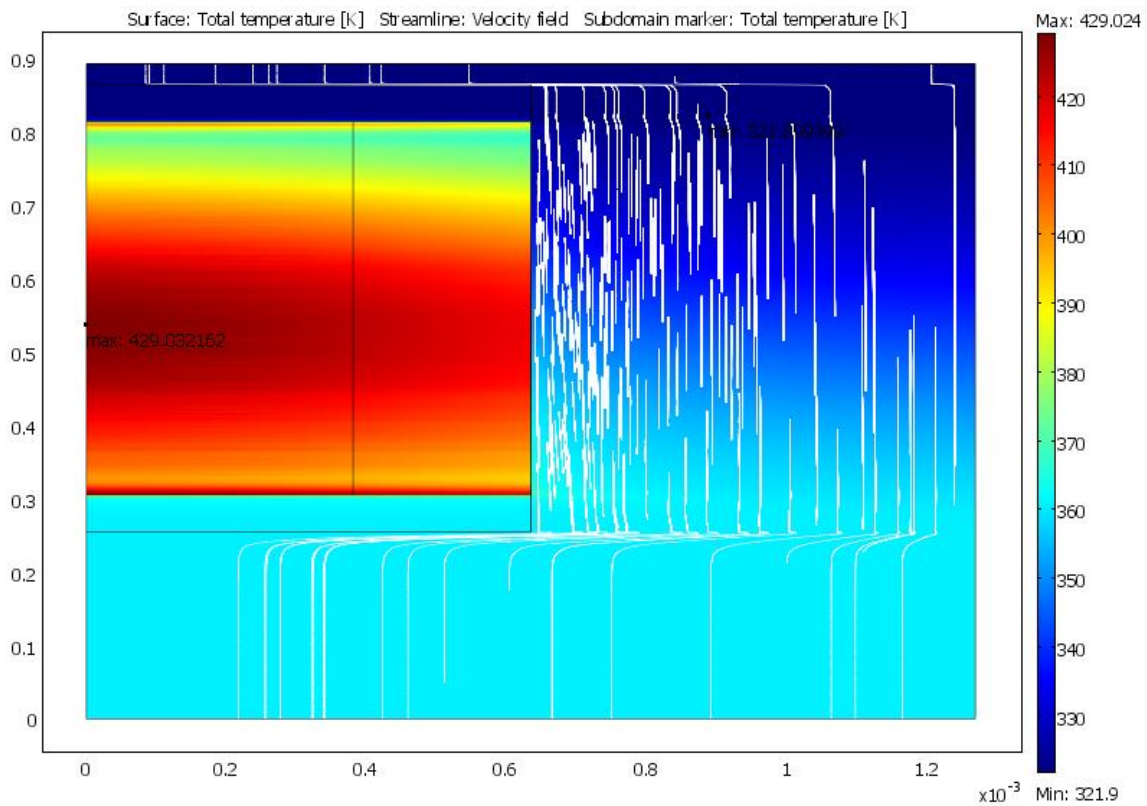
Fig. 10.1. 2D solution using the power density profile found with uranium fuel alone generating the heat and none by the filler material.

The model results shown by Fig. 10.1, as part of the solution process, includes the effects of developing momentum and thermal boundary layers to determine the surface temperature distribution without specifying the heat transfer coefficient. As a result, the film coefficient and the bulk water temperature may be attributed to the flow physics calculations. This model also incorporates the variable properties that were set up as a COMSOL library discussed in Section 4. The surface heat flux from both the SSHTC output and the 2D model are shown in Fig. 10.2. Figure 10.2 shows the effect of specifying the wall offset, a necessary condition in COMSOL 3.5a. The valid range for the wall offset is

$$30 \leq y^+ \leq 100 \,.$$
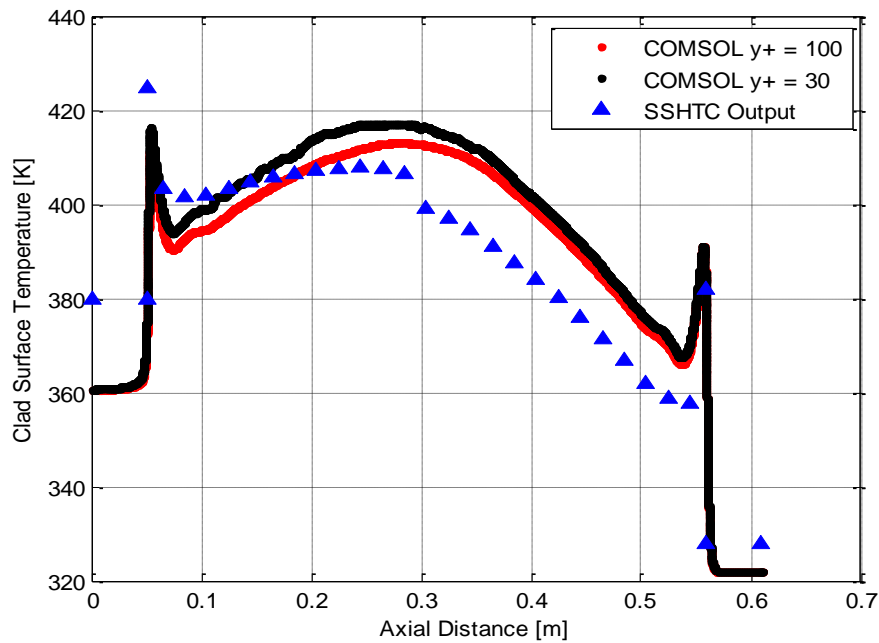
Fig. 10.2. Clad surface temperature profile using the 1D SSHTC restrictive conditions and the more physical 2D conditions.

Figure 10.2 provides much insight regarding the temperature distribution of the fuel plate using a more physical representation. One notices the significant difference in the temperature profiles, between the SSHTC and the 2D COMSOL results, caused by the use of an isotropic thermal conductivity. For example, the position of the peak temperature has moved slightly downstream of the center of the plate. This has a significant effect on the deformation profile of the fuel plate caused by thermal expansion, and as a result on the location of potential flow channel necking. The location of the highest temperature in the SSHTC output are close to the trailing edge of the fuel plate. Also, the local temperature maximum occurring around $x = 0.55\text{m}$ has increased. Thus the use of the more physical thermal conductivity acts to distribute the heat more uniformly in the fuel plate.

The global error in the conservation of energy for the COMSOL model was found to be 0.367%. While there exists a 0.018% error in global mass conservation.

11. REFERENCES

1. Cook, D. H., *HFIR Fuel Plate Involute Geometry (C-HFIR-2009-004 Rev 0)*, 2009.

2. "Engineering Toolbox" Available at: http://www.engineeringtoolbox.com/ 12/14/2009

3. Freels, J. D., *Implementation of Oak Ridge National Laboratory Software Quality Assurance Requirements for COMSOL 3.4*, COMSOL Conference, Boston, 2008.

4. H. A. McLain, *HFIR Fuel Element Steady State Heat Transfer Analysis, Revised Version*, ORNL/TM-1904, Oak Ridge National Laboratory, Oak Ridge, Tennessee, December 1967 as appended by T. E. Cole, L. F. Parsly, and W. E. Thomas, Revisions to the HFIR Steady State Heat Transfer Analysis Code, ORNL/CF-85/68, April 7, 1986.

5. Morris, D. G. and Wendel, M. W., *High Flux Isotope Reactor System Relap5 Input Model (ORNL/TM-11647)*, 1993.

6. White, F. M.: *Fluid Mechanics*, 5th ed., McGraw-Hill, New York, 2003.

7. Tschaepe, L.; Ruggles, A. E., Freels, J. D.; Primm, R. T.: *Evaluation of HFIR LEU Fuel Using the COMSOL Multiphysics Platform (ORNL/TM-2008/188), 2009*

8. HFIR engineering drawing D-42114, revision J, HFIR Inner Fuel Element Fuel Plate Loading Details, 12/15/2005.

9. HFIR engineering drawing D-42122, revision J, HFIR Outer Fuel Element Fuel Plate Loading Details, 12/15/2005.

10. NIST "Thermophysical Properties of Fluid Systems" Available at: http://webbook.nist.gov/chemistry/fluid

APPENDIX A

Extrusion Coupling Variables

(COMSOL 3.5a User Manual)

Extrusion Coupling Variables

An *extrusion coupling variable* maps values from the source domain to the destination domain. When the domains are of the same space dimension, you typically have a point-wise mapping. When the destination domain has higher dimension than the source domain, the mapping is done by extruding point-wise values to the higher dimensions. You can define the transformation between the source and destination in two ways: as a *linear transformation* or as a *general transformation*.

LINEAR TRANSFORMATION

The linear transformation maps between domains of the same dimension. The domains can exist in geometries of different space dimensions. For example, you can couple from edges in 2D to those in 3D; you can also couple 2D subdomains to 3D faces. In these cases you obviously need geometries of different space dimensions for the source and destination. You define the linear transformation by specifying points in both the source and destination.

To define an extrusion coupling variable with linear transformation, first go to the **Options** menu and point to **Extrusion Coupling Variables**. Then click **Subdomain Variables**, **Boundary Variables**, **Edge Variables**, or **Point Variables**, depending on the intended source for the coupling variable.
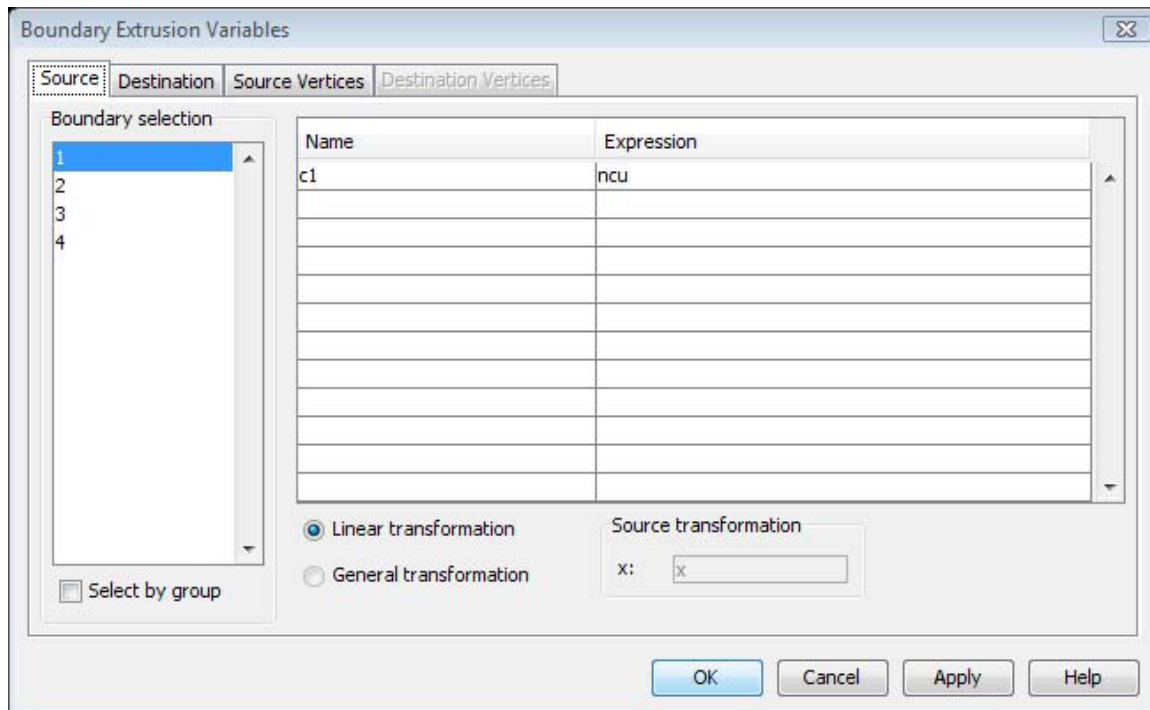
Fig. 4-51: Specifying the sources for the boundary extrusion coupling variables (linear transformation).

In the dialog box, select source domains for the coupling variable in the domain selection list (in the previous example it is the **Boundary selection** list) or by clicking on domains in the drawing area. Make an entry for the coupling variable in the **Name** edit field, and then provide an expression on these domains in the **Expression** edit field. Make sure that you have selected the **Linear transformation** button.

Now click the **Destination** tab. First select the geometry of the destination domain from the domains in the **Geometry** list. You can use the same geometry for both source and destination. The dimension of the destination domain is the same as the dimension of the source domain, regardless of the geometry dimensions. COMSOL Multiphysics transforms the expression on the source domains to the selected destination domains.
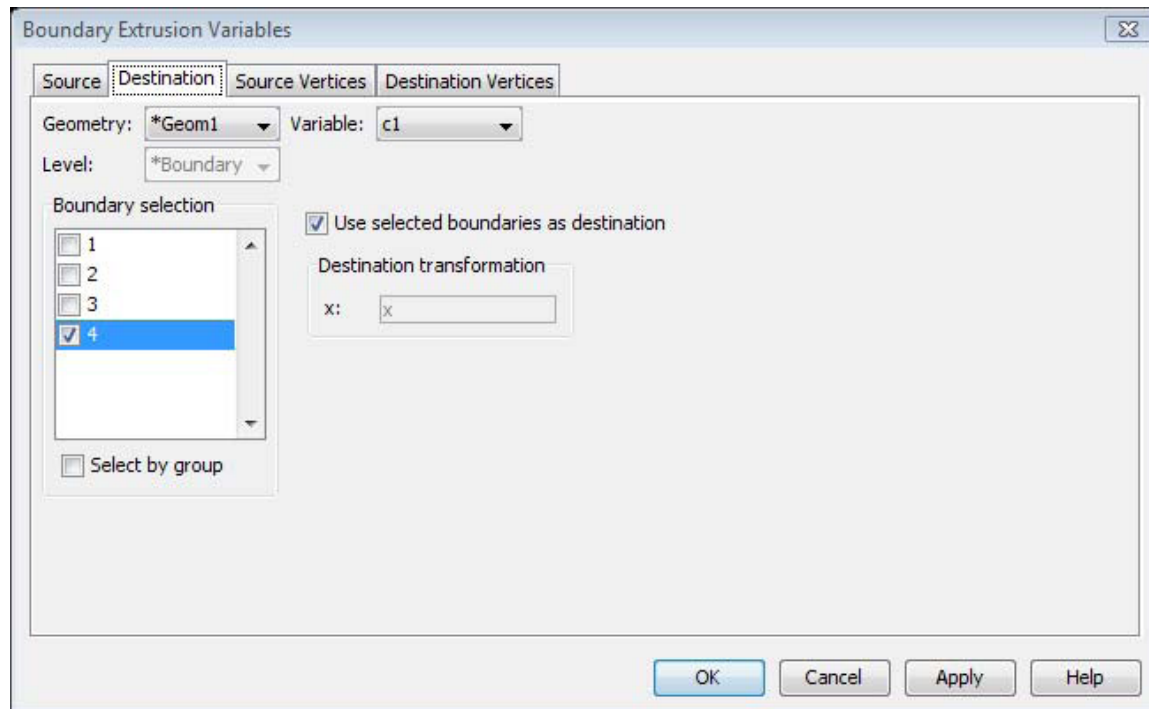
Fig. 4-52: Specifying the destinations for the boundary extrusion coupling variables (linear transformation).

You can change the variable when working with the destination page. The default variable is the one selected on the source page.

Continue by specifying the linear transformation by giving a set of points in the source geometry and in the destination geometry. This works in exactly the same way as for periodic boundary conditions. See "Linear Transformation from Source to Destination" on page 260 for more information.
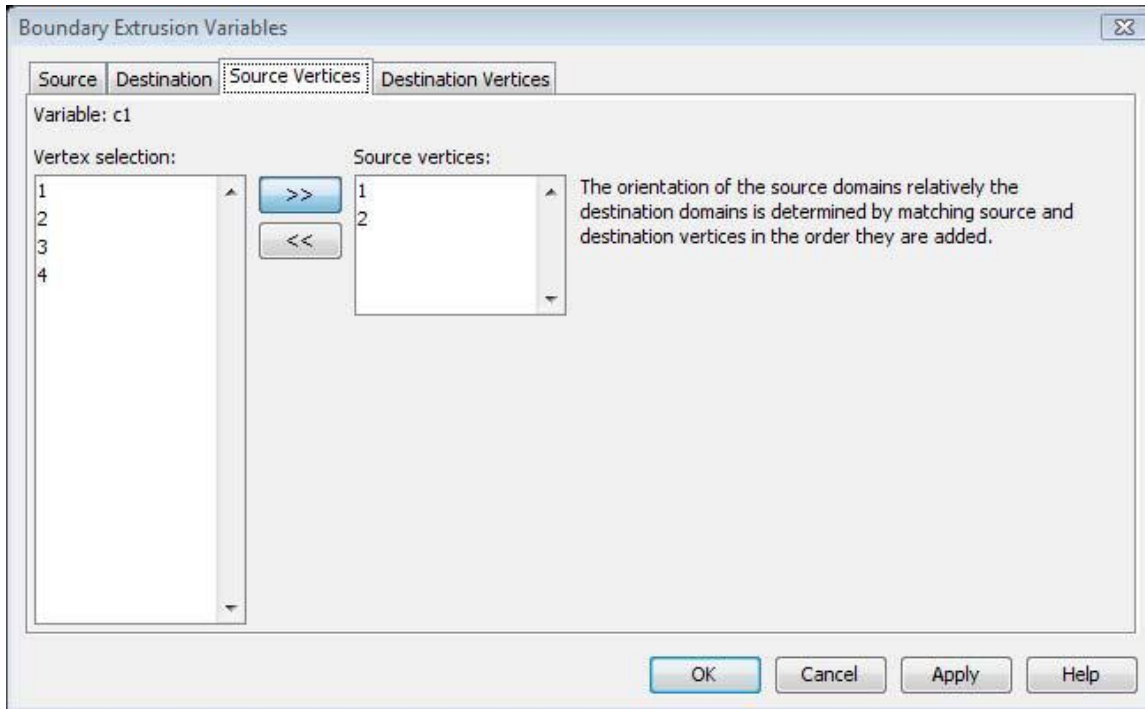
Fig. 4-53: Specifying the source vertices for the boundary extrusion coupling variables.

Select vertices in the **Vertex selection** list and move them by clicking the right arrow button to the **Source vertices** or **Destination vertices** list, respectively.

Together, the vertices on the source geometry and destination geometry define a linear transformation.

GENERAL TRANSFORMATION

The *general transformation* of the extrusion coupling variable defines a more general transformation between source and destination than the linear transformation. Specifically, when the destination domain has more space dimensions than the source domain, the variable performs extrusion of values.

The definition of any extrusion coupling variable involves two mesh transformations, which are important to understand. The *source transformation* is a one-to-one mapping that maps the mesh of the physical source domain to an *intermediate mesh* embedded in a space of the same dimension as the source. The *destination transformation* is a mapping from the destination domain, where the value of the variable is defined, to the same space that contains the intermediate mesh.

When you request the value of the coupling variable somewhere in the destination domain, the software transforms the destination points using the destination transformation. It compares the resulting coordinates to the elements in the intermediate mesh to find corresponding locations in the physical source domain. This means that the source transformation must be inverted but not the destination transformation. The latter can in fact be noninvertible, which is, for example, the case for a linear extrusion.

To avoid the need to solve a nonlinear system of equations for every destination point, the software assumes that the source transformation is linear on each element of the intermediate mesh. In practice, the transformation is often trivial and leaves the coordinates unchanged, but it can also rescale, stretch, bend, or reflect the mesh.

Choose the geometry of the source domain by opening the dialog box from that particular geometry. To define an extrusion coupling variable with general transformation, go to the **Options** menu and select **Extrusion Coupling Variable**. Then select **Subdomain**, **Boundary**, **Edge**, or **Point**, depending on the intended source for the coupling variable.
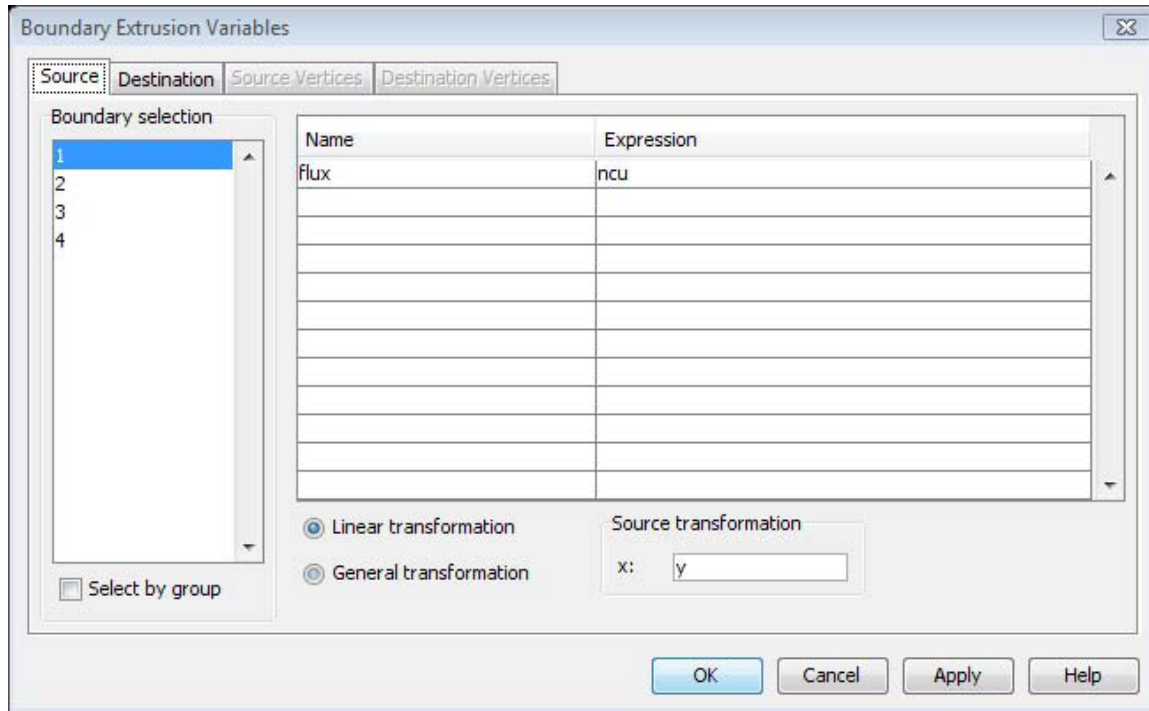
Fig. 4-55: Specifying the sources for the boundary extrusion coupling variables (general transformation).

In the dialog box, select source domains for the coupling variable in the domain selection list; in the example above it is the **Boundary selection** list. Make an entry for the coupling variable in the **Name** edit field, and then provide an expression for the selected source domains in the **Expression** edit field. Make sure that you have selected the **General transformation** button.
Now define the source transformation that maps the source domain to the intermediate domain of the same dimension. The source transformation has the same number of fields as the dimension of the source domain. You can use expressions containing space coordinates in the source geometry when defining the transformation. Remember that the transformation must be approximately linear within each mesh element.

Now click the **Destination** tab. First select the **Geometry** and then select the domain type. You can change the variable when working with the destination page. The default variable is the one selected on the source page.
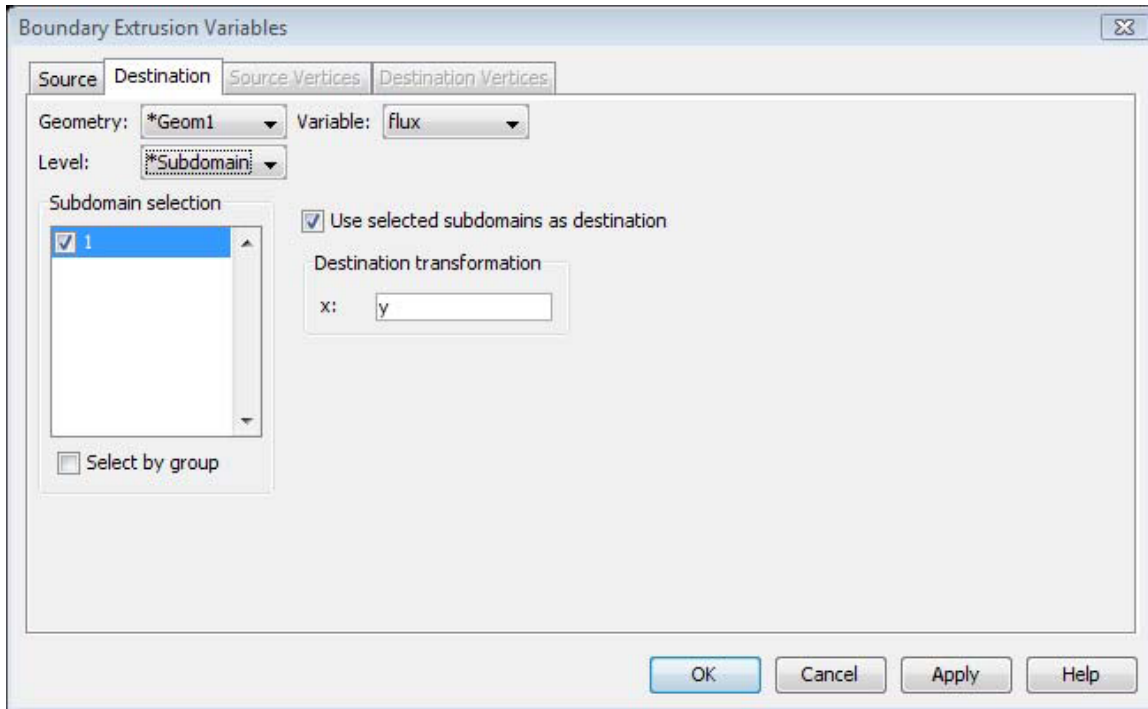
Fig. 4-56: Specifying the destinations for the boundary extrusion coupling variables (general transformation).

Specify the destination transformation. The destination transformation maps coordinates in the destination domain to the intermediate mesh. The destination transformation has the same number of fields as the dimension of the source domain. When defining the transformation it is permissible to use expressions containing space coordinates in the destination geometry. You can also specify an arbitrary transformation, which can be highly nonlinear or noninvertible.

APPENDIX B

Calculation of Volumetric Heat Generation Rate

TM-2008/188

The volumetric heat generation number cited in the TM-2008/188 is 2.65×10^9 W/m^3. This is generated from a total heat load in the fuel section of 80.7 MW as reported in the RELAP5 report by Morris and Wendel. The fuel volume is reported as 0.030355 m^3; no justification is presented for this volume.

RELAP5 (Morris and Wendel)

Geometry parameters reported in this report are:

Total active fuel heat transfer area: 39.84 m^2

Active heat transfer area for one inner plate: 0.0792 m^2

Active heat transfer area for one outer plates 0.0712 m^2

Volume of fuel meat in one inner plate: 3.03×10^{-5} m^3

Volume of fuel meat in one outer plate: 2.71×10^{-5} m^3

Total inner plates: 171

Total outer plates 369

MABE Analysis

Method 1: Estimate

From the fuel plate drawings, the following geometric parameters are determined:

Nominal active fuel plate length: 20 in = .508 m

Lumped Fuel width of fuel and filler: 0.03 in = 7.62×10^{-4} m

Active inner involute arc length: 3.13 in = 0.079502 m

Active outer involute arc length: 2.81 in = 0.071374 m

Inner fuel volume = $7.62 \times 10^{-4} \times 0.079502 \times .508 \times 171 = 0.005263$ m^3

Outer fuel volume = $7.62 \times 10^{-4} \times 0.071374 \times .508 \times 369 = 0.010195$ m^3

Total fuel volume = 0.015458 m^3

Method 2: RELAP5 Volume

Use RELAP5 volumes

Inner fuel volume: $3.03 \times 10^{-5} \times 171 = 0.0051813$ m^3
Outer fuel volume: $2.71 \times 10^{-5} \times 369 = 0.009999$ m^3

Total fuel volume = 0.015181 m^3

Method 3: RELAP5 Area

From RELAP5 active area. Total areas must be divided by 2 to give the active area for one side of the plate

Inner fuel volume: $0.0792/2 \times 171 \times 7.62 \times 10^{-4} = 0.0051600$ m^3
Outer fuel volume: $0.0712/2 \times 369 \times 7.62 \times 10^{-4} = 0.01001$ m^3

Total fuel volume = 0.015170 m^3

Method 4: SolidWorks

A CAD model of the fuel section was drawn using solid works. The involute is formed by a series of defined points connected by splines. The software will calculate the volume. In this case, the volume is of the fuel and filler from the inner radius to the outer radius.

Volume of inner fuel in one plate: 3.2413×10^{-5} m^3
Volume of inner fuel: $171 \times 3.2413 \times 10^{-5} = 0.0055426$ m^3

The outer plate has not been drawn but the inner plate result is within 7% of Method 2.

Method 5: Finite Volume

The SSHTC code defines axial and radial nodes for the inner and outer plate. The radial nodes can be translated into arc lengths. The control volumes can be calculated and summed.

Inner plate: 3.615×10^{-5} m^3 x $171 = 0.005413$ m^3
Outer plate: 2.833×10^{-5} m^3 x $369 = 0.010455$ m^3

Total active fuel volume: 0.015868 m^3

Where is the mistake between the TM and RELAP5 information:

If the total active area in RELAP5 (39.84 m^2) is multiplied by the thickness of the lumped fuel meat section:

$39.84 \times 7.62 \times 10^{-4} = 0.030358$ m^3

The mistake in this calculation is that the active area accounts for both sides of the plate.

Method 6: RELAP5 Area total

The active area should be divided by two to result in a fuel volume estimation of 0.015179 m^3.

The correction is that the reported volumetric heat generation should be doubled to 5.316×10^9 W/m^3.

A summary of the different methods is shown in the table. The RELAP5 Active Volume (Method 2) is used as the base of comparison.

	Inner Plate Volume - m^3	% Error	Outer Plate Volume - m^3	% Error	Total Volume - m^3	% Error
Method 1	0.005263	1.58	0.010195	1.96	0.015458	1.82
Method 2	0.005181		0.009999		0.015181	
Method 3	0.00516	0.41	0.01001	0.11	0.01517	0.07
Method 4	0.005543	6.97				
Method 5	0.005413	4.47	0.010455	4.56	0.015868	4.53
Method 6					0.015179	0.01

To determine the volume of the active fuel section, the best recommendation is to use the RELAP5 volumes (Method 2). The RELAP5 areas can be used and result in little error that is mostly contributed to round off (Method 3 and 6). The estimate in Method 1 is within 2% of the RELAP5 calculations. A more careful calculation of the involute cross-sectional area is needed but Method 1 provides a reasonable estimate. Using SolidWorks (Method 4) accounts for the inactive area between the fuel and the wall of the inner channel. Method 5 over predicts the volume but maintains the inner and outer volume ratios. More divisions in the model radial nodes are needed to better determine the active fuel volume.

APPENDIX C

Using Functions Based on Interpolated Data

(COMSOL 3.5a User Manual)

To define functions based on interpolated data, use the **Functions** dialog box, which you open from the **Options** menu. The interpolation data can be both structured, (defined on a grid) or unstructured (defined on a generic point cloud). To define a new interpolation function, click the **New** button to open the **New Function** dialog box. Specify a name for the function, then click the **Interpolation** button. Next choose a method for entering data (a user-defined table or data from a file or MATLAB workspace) and, if applicable, a data source, then click **OK**. For file data, select the **Store data in model** check box to load the data into the model and make it independent of the original data file.
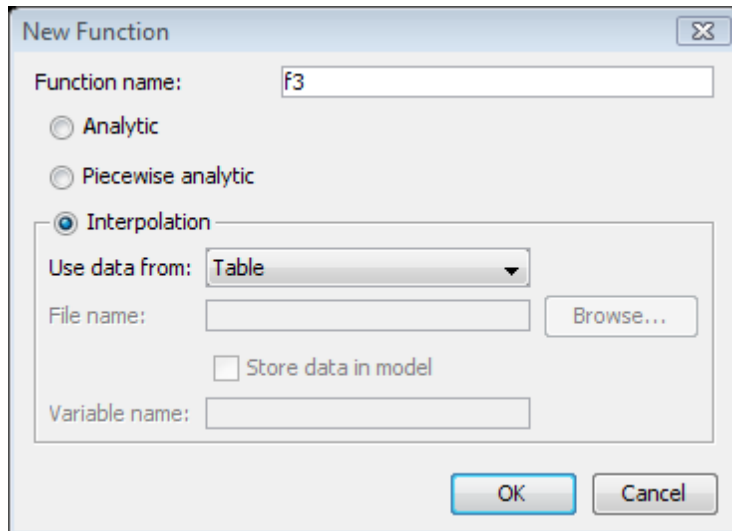
Fig. 4-4: The New Function dialog box, used here to specify an interpolation function.

For functions of one variable, you can choose between the following interpolation methods:

•Nearest neighbor

•Linear

•Piecewise cubic

•Cubic spline

For functions of more than one variable COMSOL Multiphysics supports only the nearest-neighbor and linear interpolation methods. Piecewise-cubic interpolation is a method using a piecewise-cubic Hermite polynomial with continuous first derivatives. It preserves the shape of the data and respects monotonicity. The cubic-spline method also performs interpolation with a piecewise cubic polynomial. Here, even second derivatives are continuous; however, the method does not necessarily respect monotonicity.

Also specify how COMSOL Multiphysics treats arguments that fall outside the grid or mesh of points. There are four extrapolation methods to choose from for structured interpolation:

•Constant

•Interpolation function

•Linear

•Specific number

Unstructured interpolation supports using a constant or a specific number only.

Constant means that the extrapolation method uses the value from the closest point inside the grid (for structured interpolation) or the value from the closest mesh element (for unstructured interpolation). The interpolation function evaluates the polynomial from the closest grid point at the actual point where a value is requested. Linear means that the function is linear outside the grid with a continuous function value and continuous first derivative at the boundary of the grid. This extrapolation method can only be used together with the interpolation methods piecewise cubic and cubic spline. By selecting the specific number option, you can assign a single value, usually zero or NaN, to all points outside the grid or mesh.

The **Use space coordinates as default function arguments** check box, when selected, makes it possible to use the space coordinates as default function arguments. Thus, if the space coordinates are x, y, and z, selecting this check box for a function myfun allows you to write myfun instead of myfun(x,y,z) when using the function. You can use this option only for geometries of the same dimension as the function. The complete version (myfun(arg1,arg2,arg3)) is always available. For functions of one variable you can enter the data directly into a lookup table (see Fig. 4-5).

Figure 4-5: The Functions dialog box, used here to edit an interpolation function.
To get a graph of the interpolation function, click the **Plot** button. The plot includes extrapolation (the red dashed lines outside of the interval for *x*) to show how the extrapolation method works (see Fig. 4-6).
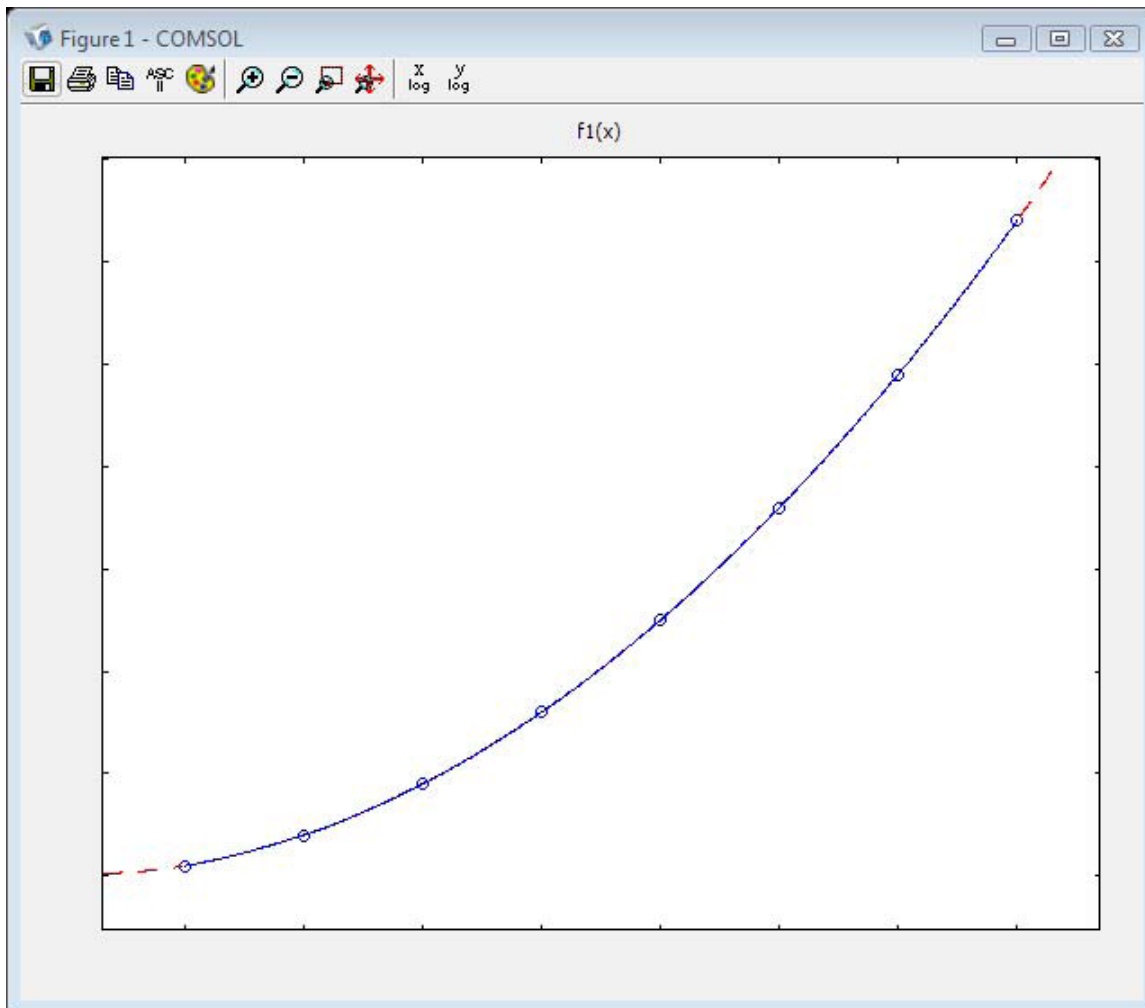
Fig. 4-6: A plot of the function f1(x). Notice the dashed red lines that indicate the extrapolation.

For functions of one to three variables, you can retrieve data from a text file or MATLAB. The interpolation functions support the three file formats that can be exported from **Export > Postprocessing Data** on the **File** menu: the **Grid, data** format on a regular grid, the **Coordinates, data** format, and the **Nodes, elements data** format, described in more detail in "Exporting Postprocessing Data to a File" on page 489.

When the data is stored in a text file, the function normally only stores the name of the text file and reads data from the file during the analysis. Select the **Store data in model** check box when creating the function to load the data into the model and make it independent of the original file.

- Nodes, elements, data:

%Coordinates

One to three columns containing x, y (optional) and z   (optional)

%Elements

Triangulation where each row contains the row indices of the points in the Coordinates section that make up one element(triangular in 2D, tetrahedral in 3D) %Data (funname)

Column of data values for each point

It is possible to include more than one function in the file as long as a %Data header separates them one from the other.

- Grid, data:

% Grid

x grid points separated by spaces

y grid points separated by spaces (optional)

z grid points separated by spaces (optional)

% Data

Data values separated by spaces

Each row contains values for different *x* grid points for fixed values of *y* and *z*. The rows first increase the *y* grid value and then the *z* grid value. The grid points can also represent another independent variable that the data values depend on. For example, the "grid points" can be temperature values and the data values the thermal conductivity at these temperatures (see the example in "Interpolation of External Data and Nonlinear Materials" on page 236). It is important to use a comment line starting with % to separate the grid points or other interpolation points and the data values that are associated with these coordinates or interpolation points. You can use the postwriteinterpfile function to create a file of this format from MATLAB.

It is possible to include more than one function in the file as long as a %Data header separates them one from the other.

-Coordinates, data:

%Header (optional)

Columns containing x, y (optional) and z (optional) followed by data columns.

You can include function names in the header. In that case, the input columns must be labeled with x, y, and z, respectively, depending on input dimension. For example, a file with the following content creates two 2D functions named myfun1 and myfun2:

% x y myfun1 myfun2

0 0 0.12 0.34

0 1 0.52 1.50

1 0 0.67 0.91

If the file does not include any header to indicate the function dimension, the software assumes that it is identical to the largest geometry dimension present in the model. A file with four columns, for example, is interpreted as one 3D function in a 3D model, two 2D functions in a 2D model, and three 1D functions in a 1D model.

INTERNAL DISTRIBUTION

1. K. J. Beierschmitt (*beierschmitt@ornl.gov*)
2. C. A. Blue (*blueca@ornl.gov*)
3. I. T. Bodey (bodeyit@ornl.gov)
4. S. E. Burnette (*burnettese@ornl.gov*)
5. D. C. Christensen (*christensend@ornl.gov*)
6. D. H. Cook *(dhc@ornl.gov)*
7. R. A. Crone (*cronera@ornl.gov*)
8. R. J. Ellis (*ellisrj@ornl.gov*)
9. J. D. Freels *(freelsjd@ornl.gov)*
10. J. C. Gehin (*gehinjc@ornl.gov*)
11. G. Ilas (*ilasg@ornl.gov*)
12. P. K. Jain (jainpk@ornl.gov
13. B. C. Jolly (jollybc@ornl.gov)
14. J. H. Miller (*millerjh2@ornl.gov*)
15. C. V. Parks (*parkscv@ornl.gov)*
16. R. T. Primm III (*primmrtiii@ornl.gov*)
17. R. R. Rawl (*rawlrr@ornl.gov*)
18. D. G. Renfro (*renfrodg@ornl.gov)*
19. A. W. Riedy (*riedyaw@ornl.gov*)
20. J. E. Rushton (*rushtonje@ornl.gov*)
21. L. J. Satkowiak (*satkowiaklj@ornl.gov*)
22. K. A. Smith (*smithka@ornl.gov*)
23. R. L. Snipes (*snipesrl@ornl.gov*)
24. S. J. Zinkle (*zinklesj@ornl.gov*)
25. J. C. Wagner (wagnerjc@ornl.gov)
26. G. L. Yoder Jr. (yodergljr@ornl.gov)
27. ORNL Laboratory Records (*hamrindr@ornl.gov*)

EXTERNAL DISTRIBUTION

28. A. Adams, U.S. Nuclear Regulatory Commission, One White Flint North, 11555 Rockville Pike, Rockville, Maryland 20852-2738 (*axa@nrc.gov*).
29. R. V. Arimilli, 235 Dougherty Engineering Building, University of Tennessee, Knoxville, TN 37996-2030 (arimilli@utk.edu).
30. R. A. Butler, Director, Research Reactor Center, 1513 Research Park Drive, Columbia, MO 65211 (*ButlerRa@missouri.edu*).
31. G. S. Chang, Idaho National Laboratory, P.O. Box 1625, Idaho Falls, ID 83415-3885 (*gray.chang@inl.gov*).
32. D. Chong, NA-212, U.S. Department of Energy, 1000 Independence Avenue SW, Washington, DC 20585 (*Daniel.Chong@nnsa.doe.gov*).
33. H. E. Clark, U.S. Department of Energy Oak Ridge Office, P.O. Box 2001, Oak Ridge, TN 37831 (*hkc@ornl.gov*).
34. D. Diamond, Brookhaven National Laboratory, P.O. Box 5000, Upton, NY 11973-5000 (*diamond@bnl.gov*).
35. E. Fontes, COMSOL AB, Tegnergatan 23, SE-111 40, Stockholm, Sweden (ed@comsol.com).
36. H. D. Gougar, Manager, Fission & Fusion Systems, INL, P.O. Box 1625, MS 3860, Idaho Falls, ID 83415-3860 (*goughd@inl.gov*).
37. D. M. Hewitt, Idaho National Laboratory, P.O. Box 1625, Idaho Falls, ID 83415-3750 (*Dana.Hewitt@inl.gov*).
38. L. Jollay, BWXT/Y-12, Y-12 National Security Complex, P.O. Box 2009, Oak Ridge, TN 37831-8245 (*jollayl@y12.doe.gov*).

39. Roger Klaffky, Office of Basic Energy Sciences, U.S. Department of Energy, 1000 Independence Avenue SW, Washington, DC 20585 (*roger.klaffky@science.doe.gov*).
40. D. Kutikkad, Assistant Reactor Manager-Physics, University of Missouri Research Reactor Facility, Columbia, MO 65211 (*kutikkadk@missouri.edu*).
41. K. T. Lowe, Bettis Atomic Power Laboratory, Bechtel Bettis, Inc, Mail Stop 36E, P. O. Box 79, West Mifflin, PA, 15122 (lowekt@bettis.gov) or (klowe1@gmail.com) if not available.
42. Pedro Mantano, Office of Basic Energy Sciences, U.S. Department of Energy, 1000 Independence Ave., SW, Washington, DC 20585 (*pedro.montano@science.doe.gov*).
43. J. Matos, Argonne National Laboratory, 9700 S. Cass Avenue, Argonne, IL 60439 (*jim.matos@anl.gov*).
44. C. McKibben, University of Missouri Research Reactor Facility, Columbia, MO 65211 (*mckibben@missouri.edu*).
45. Johnny O. Moore, U. S. Department of Energy, (*moorejo@ornl.gov*).
46. T. Newton, MIT Nuclear Reactor Laboratory, 138 Albany St., Cambridge, MA 02139 (*tnewton@mit.edu*).
47. Sean O'Kelly, NIST Center for Neutron Research, 100 Bureau Drive, Stop 8560, Gaithersburg, MD 20899-8560 (*sean.okelly@nist.gov*).
48. W. Richards, NIST Center for Neutron Research, 100 Bureau Drive, Stop 8561, Gaithersburg, MD 20899-8561 (*wade.richards@nist.gov*).
49. W. C. Richardson, BWXT Technology, Inc., 2016 Mount Athos Rd., Lynchburg, VA 24504 (*WCRichardson@bwxt.com*).
50. J. Roglans, Argonne National Laboratory, 9700 S. Cass Avenue, Argonne, IL 60439 (*roglans@anl.gov*).
51. N. Rom, COMSOL AB, Tegnérgatan 23, SE-111 40 Stockholm, Sweden (niklas@comsol.com).
52. A. E. Ruggles, Department of Nuclear Engineering, 315 Pasqua Engineering Building, Knoxville, TN 37996-2300 (aruggles@utk.edu).
53. B. Sjodin, COMSOL, Inc., New England Executive Park, Suite 350, Burlington, MA 01803 (bjorn@comsol.com).
54. D. Smith, COMSOL Inc., 1 New England Executive Park, Suite 350, Burlington, MA 01803 (dan.smith@comsol.com).
55. J. Snelgrove, Argonne National Laboratory, 9700 S. Cass Avenue, Argonne, IL 60439 (*jimsnelgrove@anl.gov*).
56. Carol Sohn, of Basic Energy Sciences, U.S. Department of Energy (*carol.sohn@pnso.science.doe.gov*).
57. P. Staples, NA-212, U.S. Department of Energy, 1000 Independence Avenue SW, Washington, DC 20585 (*Parrish.Staples@nnsa.doe.gov*).
58. John Stevens, Argonne National Laboratory, 9700 S. Cass Avenue, Argonne, IL 60439 (*johnstevens@anl.gov*).
59. Daniel M. Wachs, MFC 791 B-147, Idaho National Laboratory, P.O. Box 6188, Idaho Falls, ID 83415 (*Daniel.Wachs@inl.gov*).
60. R. E. Williams, NIST Center for Neutron Research, 100 Bureau Drive, Stop 8560, Gaithersburg, MD 20899-8560 (*robert.williams@nist.gov*)
61. Eric C Woolstenhulme, P.O. Box 1625, Idaho Falls, ID 83415-3750 (*Eric.Woolstenhulme@inl.gov*).