

Demonstration of MSR Salt Property Database Operation with a Reactor Analysis Tool



**Approved for public release.
Distribution is unlimited.**

R. Salko
A. Graham
R. Lefebvre
B. Langley

September 30, 2019

DOCUMENT AVAILABILITY

Reports produced after January 1, 1996, are generally available free via US Department of Energy (DOE) SciTech Connect.

Website: www.osti.gov/

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

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
Telephone: 703-605-6000 (1-800-553-6847)
TDD: 703-487-4639
Fax: 703-605-6900
E-mail: info@ntis.gov
Website: <http://classic.ntis.gov/>

Reports are available to DOE employees, DOE contractors, Energy Technology Data Exchange representatives, and International Nuclear Information System representatives from the following source:

Office of Scientific and Technical Information
PO Box 62
Oak Ridge, TN 37831
Telephone: 865-576-8401
Fax: 865-576-5728
E-mail: report@osti.gov
Website: <http://www.osti.gov/contact.html>

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

Reactor Nuclear Systems Division

Demonstration of MSR Salt Property Database Operation with a Reactor Analysis Tool

Authors
R. Salko
A. Graham
R. Lefebvre
B. Langley

Date Published: September 30, 2019

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

CONTENTS

LIST OF FIGURES	v
LIST OF TABLES	vii
ACRONYMS	ix
ABSTRACT	1
1. INTRODUCTION	3
2. THERMOPHYSICAL DATA INTERFACE	5
2.1 Design	5
2.2 Testing	7
2.2.1 Heat exchanger test	7
2.2.2 FLiBe benchmark	11
3. VERA DEMONSTRATION	15
3.1 MSRE Model Description	15
3.2 Database File	17
3.3 Results	17
4. THERMPHYS DATABASE INTERFACE	21
4.1 Description of ThermPhys	21
4.2 Thermophysical Data Format	22
5. SUMMARY	25
6. REFERENCES	27

LIST OF FIGURES

1	Example of thermophysical property file read by the data interface class.	7
2	Diagram of the heat exchanger verification problem.	8
3	Comparison of the analytical and CTF solution for flow through a heat exchanger	9
4	Code verification of the heat exchanger problem showing that CTF converges to the analytical solution for temperature rise when specific heat is specified via the fluid property database file.	10
5	Comparison of liquid enthalpy predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.	12
6	Comparison of liquid temperature predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.	13
7	Comparison of liquid density predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.	13
8	Comparison of liquid velocity predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.	14
9	Comparison of graphite surface temperature predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.	14
10	CTF mesh generated by the quarter symmetric MSRE model with a simplified primary loop model.	15
11	MPACT model generated by the quarter symmetric MSRE model showing graphite stringers (red) with fuel salt channels (blue).	16
12	Density distribution (kg/m^3) for the coupled MPACT-CTF MSRE calculation.	18
13	Power distribution (kW/m^3) for the coupled MPACT-CTF MSRE calculation.	19
14	ThermPhys application input as highlighted within the Nuclear Energy Advanced Modeling and Simulation (NEAMS) Workbench.	21
15	ThermPhys System auto-completion in the NEAMS Workbench.	22
16	ThermPhys Reference auto-completion in the NEAMS Workbench.	22

LIST OF TABLES

1	FLiBe salt thermophysical properties	11
2	LiF-NaF-KF Salt thermophysical properties	17

ACRONYMS

ANL Argonne National Laboratory

CASL Consortium for Advanced Simulation of Light Water Reactors

HTC heat transfer coefficient

LWR light water reactor

M&S modeling and simulation

MSR molten salt reactor

MSRE Molten Salt Reactor Experiment

MSTDB Molten Salt Thermodynamic Database

NEAMS Nuclear Energy Advanced Modeling and Simulation

UQ Uncertainty Quantification

VERA Virtual Environment for Reactor Applications

WASP Workbench Analysis Sequence Processor

ABSTRACT

Successful modeling and simulation of molten salt reactors (MSRs) requires quality thermodynamic and thermophysical data for the working fluid. A goal of the Molten Salt Reactor Campaign is to develop a quality-controlled database of these properties that will be available for a multitude of different salt types. Furthermore, properties shall be defined in a standardized format that can then be used by reactor modeling and analysis tools. This report details work done to develop an approach for integrating salt properties into an MSR analysis tool. A format was defined for thermophysical properties that could be (1) produced from salt data delivered by Argonne National Laboratory (ANL) and (2) read into a reactor analysis tool. The reactor analysis tool used for this activity was VERA, which was developed within the Consortium for Advanced Simulation of Light Water Reactors. This document describes the changes implemented in VERA to accept the thermophysical properties file and the tests performed. It also demonstrates the process of reading thermophysical properties into VERA for a MSR simulation. In addition to being able to add fluid properties into the simulation tool, a process is also needed to add new salt data to the salt database. The NEAMS Workbench is being expanded with a new package called ThermPhys to provide a user interface for adding new salts to the database while performing input validation and ensuring data quality.

1. INTRODUCTION

Molten salt reactor (MSR) modeling and simulation (M&S) presents a set of unique challenges that require a multiphysics solution, with specific consideration for the chemical behavior of the salt during reactor depletion. Mass accountancy, which includes tracking of fission products, corrosion products, and salt species, is a key consideration in the successful design and operation of these reactors. Being able to predict speciation of salts and migration of operational byproducts requires quality thermodynamic data, as well as salt thermophysical properties for characterizing the thermal-hydraulic state of the salt. A report by McMurray et al. [10] describes development of the Molten Salt Thermodynamic Database (MSTDB), which defines thermodynamic properties for a collection of salts and an approach for maintaining the database. The standardized ChemSage format will be used for thermodynamic data. This document discusses development of a thermophysical data format and its application in the reactor analysis tool Virtual Environment for Reactor Applications (VERA).

VERA [7] is a reactor simulation tool that was developed as part of the Consortium for Advanced Simulation of Light Water Reactors (CASL) [1] to provide a high-fidelity multiphysics framework for light water reactor (LWR) simulation and analysis. Development of the VERA tools has been focused on addressing a series of industry challenge problems that cannot be addressed using legacy methods. Because many of the methods used in VERA are agnostic to the reactor being analyzed, and because it includes multiphysics modeling capabilities, work has been performed to extend VERA to MSR M&S capabilities [4]. Specific developments for MSR M&S activities include:

- The addition of a lumped depletion capability to account for moving and mixing the fuel
- The addition of a general multi-phase species tracking capability to track the movement of delayed neutrons, fission products, and salt components in the loop
- The addition of a salt feed/removal capability
- Improvement of neutronics and thermal-hydraulics coupling to include energy generation in the fuel salt
- Development of a package for driving the species transport calculation, including definition of salt source and removal terms
- Coupling to the equilibrium chemistry code Thermochemica
- Modeling of the Molten Salt Reactor Experiment (MSRE) experiment

Thermochemica, the equilibrium chemistry code, is currently being developed to accept the ChemSage-formatted data file to obtain its thermodynamic data. Completion of this step will allow VERA to read thermodynamic data from the salt database. The thermal-hydraulic code in VERA, CTF [11], has been modified to work with the thermophysical data interface described in this report. This report describes development of the preliminary salt thermophysical database format and data interface, presents its implementation in CTF and testing, and presents a demonstration in which thermophysical salt data provided by ANL [5] is converted to the database format and used in a VERA simulation. Furthermore, development of the ThermPhys capability in NEAMS Workbench is discussed, which provides a user interface for adding new salt properties to the database.

2. THERMOPHYSICAL DATA INTERFACE

2.1 Design

In addition to collecting and defining thermophysical salt property data, it is necessary to provide a code-level interface to the data for integration with M&S tools. To address this need, a fluid property interface class was developed that allows a user to specify temperature-dependent fluid properties for a subcooled liquid via an input file. This class was integrated into CTF to allow salt thermophysical properties to be used in VERA MSR simulations.

The class for interfacing with thermophysical fluid properties contains the following set of 1D lookup tables:

- liquid density
- liquid viscosity
- liquid specific heat
- liquid thermal conductivity
- surface tension
- enthalpy-to-temperature conversion
- temperature-to-enthalpy conversion

The class provides the following public procedures:

- | | | |
|--------------|---------------|--------------------|
| 1. get_rhol | 12. get_kv | 23. get_cpg |
| 2. get_mul | 13. get_hv | 24. get_muf |
| 3. get_cpl | 14. get_Tv | 25. get_mug |
| 4. get_kl | 15. get_drldp | 26. get_kf |
| 5. get_hl | 16. get_drldh | 27. get_kg |
| 6. get_Tl | 17. get_sigma | 28. get_Tsat |
| 7. get_drldp | 18. get_hf | 29. get_dhfdp |
| 8. get_drldh | 19. get_hg | 30. get_dhgdp |
| 9. get_rhov | 20. get_rhof | 31. get_max_hl |
| 10. get_muv | 21. get_rhog | 32. get_T_critical |
| 11. get_cpv | 22. get_cpf | 33. readPropFile |

For compatibility with two-phase modeling codes like CTF, it is necessary to include a treatment for vapor phase and saturation properties. Even though the vapor phase has an insignificant impact on the solution, it must be considered for the solution's numerical stability. Therefore, several assumptions were made when

creating this interface so that reasonable vapor-phase properties can be retrieved without requiring the data developer to provide actual vapor data, which will likely not be relevant to the modeler.

Items 1–8 are the main property procedures that accept pressure and enthalpy as input and return the thermophysical property specified after the “get” statement. The procedures perform a linear interpolation over the data found in the aforementioned 1D lookup tables. Item 7 returns the derivative of the density with respect to pressure, and item 8 returns the derivative of the density with respect to enthalpy. CTF uses Newton’s method to solve the nonlinear set of governing equations. This requires that a Jacobian matrix be built. The derivative terms are required for the Jacobian matrix, and the solution may become unstable if zero values are entered for these terms. Currently, these procedures default to calculating the derivatives from the existing water fluid property tables because all salt properties that have been gathered to date are presented as being incompressible. It should be possible to calculate the derivative of density with respect to enthalpy quite easily, but the derivative of density with respect to pressure will likely not be available; therefore, this assumption will require further investigation.

Items 9–16 are similar to Items 1–8, except that they return properties for the vapor phase. Properties for vapor density, thermal conductivity, viscosity, and specific heat simply return the liquid values implemented as wrappers around the liquid procedures. The derivative procedures are returning the values for water. The “get_hv” procedure is calling the “get_hl” procedure, but it adds an arbitrary value of 100 BTU/lbm to the result to prevent any phase enthalpy differences from becoming zero. The “get_Tv” procedure wraps the “get_Tl” procedure. Because the void is set to an insignificant value in the solution, its particular solution will have an insignificant effect on the subcooled liquid solution.

Items 17–30 handle saturation properties. The surface tension returns a constant value of 0.06 lbf/ft. Because the assumption is being made that this custom fluid property feature is only valid for single-phase flows, the critical temperature and maximum liquid enthalpy that are used as limiters in the numerical solution are set to the arbitrarily high values of 2,000 °F and 5,000 BTU/lbm. The saturation temperature is also set to an arbitrarily high value of 2,000 °F. The liquid and vapor saturation densities are set to the minimum value provided by the user. The property interface does not extrapolate properties outside the range provided by the user; instead, the edge value is used for temperatures outside the table range. Therefore, the density is essentially the value for the highest defined temperature. The liquid saturation enthalpy is obtained using the saturation temperature in the “get_hl” procedure, thus returning a “large” value. Because the latent heat of vaporization, h_{fg} , is used in many calculations, the vapor saturation enthalpy is set to an arbitrarily higher value to prevent invalid math issues in the code. The remaining saturation property procedures for specific heat, thermal conductivity, and density all use the subcooled liquid procedures with the saturation temperature. Liquid and vapor values are set to the same value. The derivative of liquid and vapor saturation enthalpy with respect to pressure do not affect numerical stability, so they are set to zero (Items 29 and 30).

Finally, Item 33 is the property reader, which reads a user-defined thermophysical property file and uses the data to instantiate the class.

Fluid properties are typically defined in terms of temperature, and temperature is usually the figure of merit in terms of code output, but many times enthalpy will be used as the working representation for fluid energy in an actual M&S tool. Therefore, both “get_Tl” and “get_hl” procedures have been provided for convenience. The user is required to provide a reference temperature and a reference enthalpy in the input file. The class then uses the following relationship to calculate enthalpy from temperature.

```

! Properties for FLiBe, consistent with functional definition in CTF
! Units are:
! density - kg/m**3
! viscosity - Pa-s
! temperature - K
! specific heat - kJ/kg/K
! thermal conductivity - W/m/K
! enthalpy - kJ/kg
prop: viscosity                2
731.2  1.9719E-02
1681.2 1.0827E-03
prop: density                 2
731.2  2056.1988
1681.2 1592.5988
prop: specific_heat           2
731.2  2.4160
1673.2 2.4160
prop: thermal_conductivity    2
731.2  0.9953
1681.2 1.4703
prop: trange                   1
731.15      1673.15

```

Figure 1. Example of thermophysical property file read by the data interface class.

$$h_T = \int_{T_{ref}}^T Cp(T) + h_{ref}. \quad (1)$$

In the equation, h_T is the enthalpy for a given temperature, T_{ref} is the user-provided reference temperature for a reference enthalpy, h_{ref} is the temperature at which the enthalpy, h_T is being evaluated, and $Cp(T)$ is the temperature-dependent specific enthalpy. A numerical integration is performed because specific heat is considered as a function of temperature. This calculation is performed for a number of temperatures to build a lookup table that can be used for fast conversions during the simulation.

An example of the thermophysical property file read by this class is shown in Figure 1. The user enters four temperature-dependent property tables for viscosity, density, specific heat, and thermal conductivity. The user must enter at least two entries in a table but more values may be entered. The temperature range (melting to boiling) must also be given. The reference enthalpy used for the temperature/enthalpy conversion will be zero at the melting temperature.

2.2 Testing

2.2.1 Heat exchanger test

Testing is performed using the heat exchanger code verification test developed in the CTF testing matrix [12]. This problem verifies the correct functioning of heat transfer between fluid and a solid object, and it can also be used to verify that the code predicts the correct enthalpy rise in the fluid that occurs due to the known fluid properties. The problem involves a single channel inside a tube-shaped, unheated conductor. The channel contains water flowing upwards. The outside of the tube is a heat transfer coefficient (HTC)/temperature boundary condition. The temperature of the inlet flow is set to a value higher than the ambient temperature, causing heat transfer out of the tube as the flow moves upwards. A diagram of this

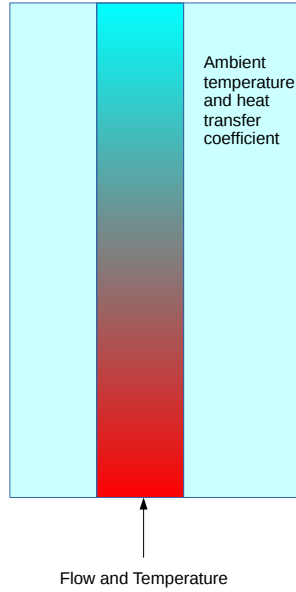


Figure 2. Diagram of the heat exchanger verification problem.

problem is shown in Figure 2. It is possible to use First Law and Newton's Law of Cooling to determine the expected temperature drop in the tube.

The boundary conditions in this model include the following:

1. Inlet temperature: 150 °C
2. Outlet pressure: 10 bar
3. Ambient temperature: 100 °C
4. Ambient HTC: 100 kW m⁻² K⁻¹
5. Inlet flow rate: 5 kg s⁻¹

The tube is 1 m long and has a diameter of 5 cm.

The specific heat is set to a constant value of 4.25 kJ kg⁻¹ through the new property interface. The other properties are set to arbitrary constant values since they do not affect the temperature prediction.

The analytical solution is formed by using First Law for an infinitesimal section of the tube:

$$q' = \dot{m}C_p \frac{dT}{dx}. \quad (2)$$

Here, q' is the linear heat rate, \dot{m} is the mass flow rate in the tube, C_p is the specific heat of the coolant, T is temperature, and x is the axial direction in the tube. The heat flux out of the tube at this location is given by Newton's Law of Cooling:

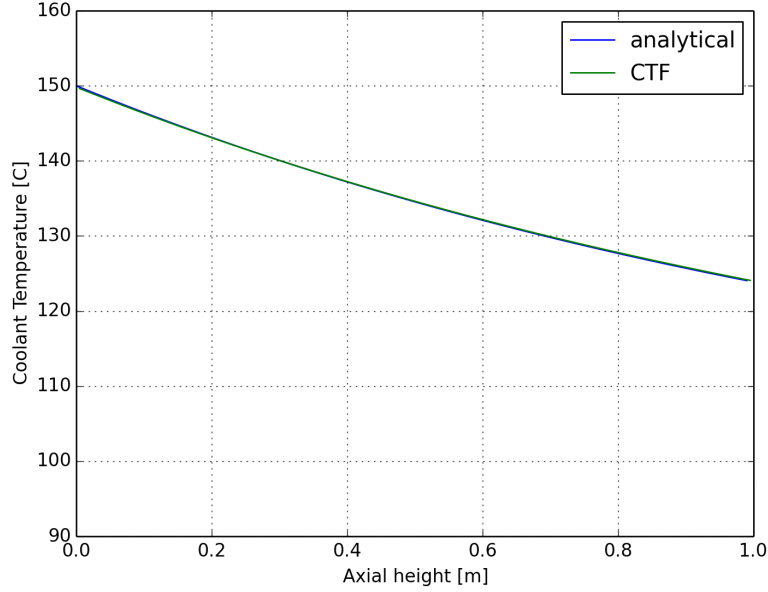


Figure 3. Comparison of the analytical and CTF solution for flow through a heat exchanger

$$q' = P_w h_\infty (T_\infty - T). \quad (3)$$

Here, P_w is the wetted perimeter of the tube, h_∞ is the HTC on the outside of the tube, and T_∞ is the ambient temperature. Substituting Eq. (3) into Eq. (2) gives a first-order, linear, ordinary differential equation:

$$\dot{m}C_p \frac{dT}{dx} + h_\infty P_w T = h_\infty T_\infty P_w. \quad (4)$$

The solution to this equation is as follows:

$$T = T_\infty + (T_{in} - T_\infty) \exp\left(\frac{h_\infty P_w}{\dot{m}C_p} x\right). \quad (5)$$

A base axial mesh of 0.01 m is employed in CTF. The tube's outside diameter is set to 5.0 cm, and the inside diameter is set to 4.9999 cm to simulate the tube being infinitely thin. The leading coefficient of the Dittus-Boelter model is set to 1,000 to simulate an infinite HTC.

Figure 3 shows that the comparison between CTF and the analytical solution is good.

To demonstrate that the solution converges to the expected analytical solution, a code verification is performed, and the error between CTF and the analytical solution is plotted against mesh size. The solution is shown to converge at the expected first-order rate (expected because CTF uses a first-order upwind differencing scheme). Figure 4 shows the results of the code verification.

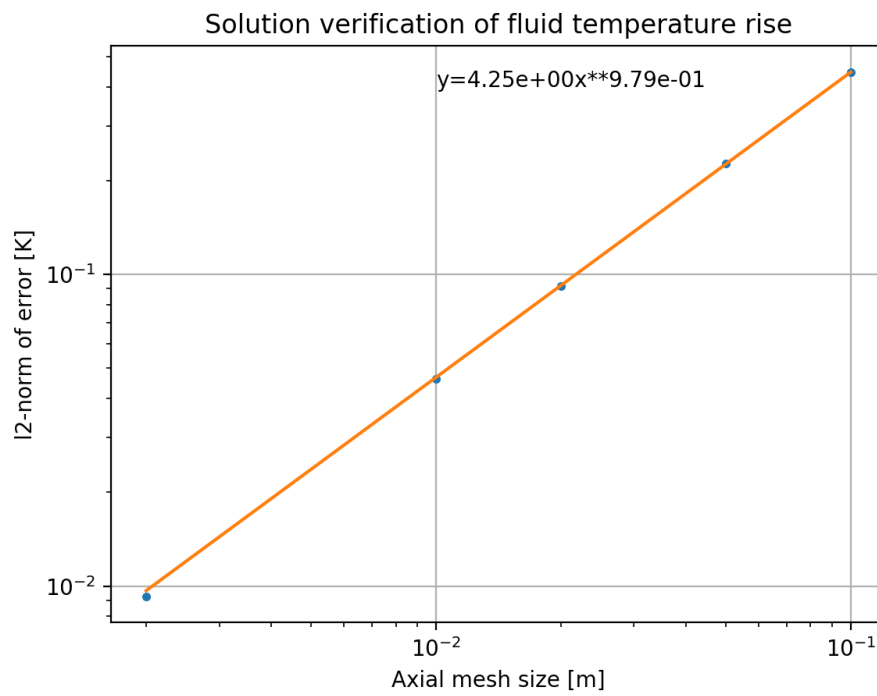


Figure 4. Code verification of the heat exchanger problem showing that CTF converges to the analytical solution for temperature rise when specific heat is specified via the fluid property database file.

2.2.2 FLiBe benchmark

In a previous activity, fluid properties for FLiBe salt were implemented into CTF [14]. Table 1 summarizes the thermophysical properties for FLiBe. These properties can be enabled in CTF by changing a flag in the input file to denote that FLiBe is the working fluid. A verification problem was set up to ensure that CTF will provide the same answer if using the internal FLiBe properties or the custom input file interface when the properties are generated using the same correlations as those in CTF. In other words, the correlations provided in Table 1 were used to create the thermophysical properties database file read by CTF.

The database file was generated by creating 98 points in the tables for density, viscosity, and thermal conductivity. The tables were evenly spaced in intervals of 10 K, ranging from the melting temperature to the boiling temperature. Only two points were used for specific heat because it is a constant.

A simple model was set up in CTF which includes four channels connected to a graphite block. There are 16 uniform axial levels, and the channel flow area, wetted perimeter, and graphite block dimensions are calculated from the MSRE facility. An inlet liquid velocity of 2 m s^{-1} is applied with an inlet temperature of 908 K and an outlet pressure of 3.447 bar. A uniform linear heat rate of 79.53 kW m^{-1} is applied to the graphite, as the heat rate cannot be applied directly to the channels through the CTF input deck at this time.

A few differences persist between the internal and input file properties:

1. The internal FLiBe properties have expressions for the derivative terms that appear in the CTF Jacobian, whereas the custom properties will fall back to using water properties, as discussed in Section 2.1.
2. The internal FLiBe properties have values for saturation enthalpy, surface tension, and salt critical temperature rather than relying on the more generic assumptions that were made for vapor properties in the custom properties.

The CTF model was run with both approaches, and the results are shown below in Figures 5–9. Results indicate a very close match in predictions, whether using the internal fluid properties or those read from the custom properties file through the new interface. This demonstrates a successful implementation of the feature. Vapor properties were not compared because different assumptions are made for these properties that may cause larger differences; however, this thermophysical property feature does not support two-phase flow, and the vapor phase presence will be insignificant in all intended applications. This particular test—in which CTF is run for the same problem using internal FLiBe properties vs. tabular properties, and subcooled liquid results are compared—was added as an automated regression test in CTF

Table 1. FLiBe salt thermophysical properties

Quantity	Equation	Units
Melting point	731.15	K
Boiling point	1673.15	K
Density	$-0.4884T + 2413.$	kg m^{-3}
Dynamic viscosity	$(1.16 \cdot 10^{-4})e^{3755./T}$	Pa s
Thermal conductivity	$0.0005T + 0.63$	$\text{W m}^{-1} \text{K}^{-1}$
Specific heat	2.416	$\text{kJ kg}^{-1} \text{K}^{-1}$

to protect this feature in the future.

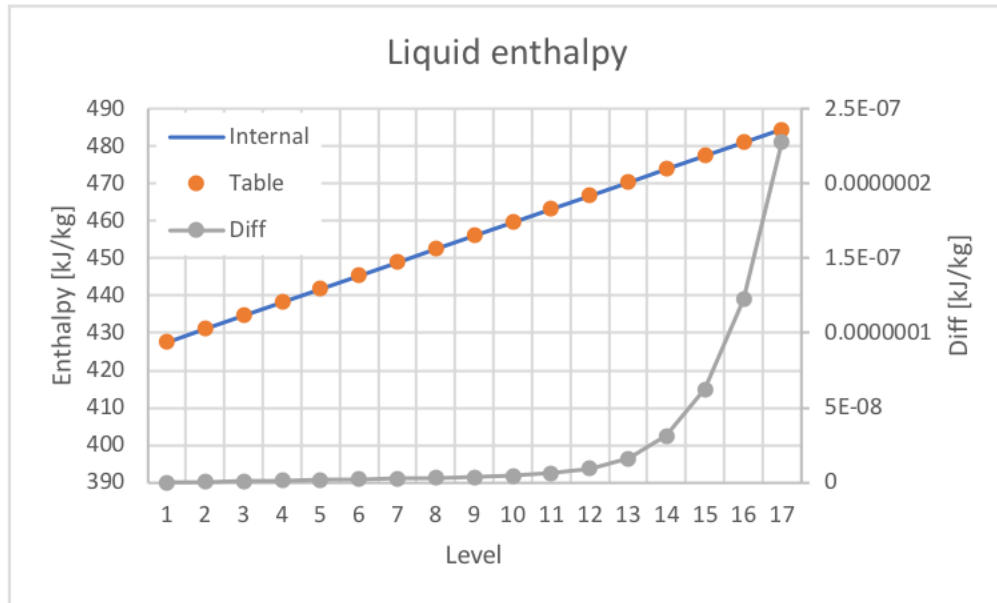


Figure 5. Comparison of liquid enthalpy predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.

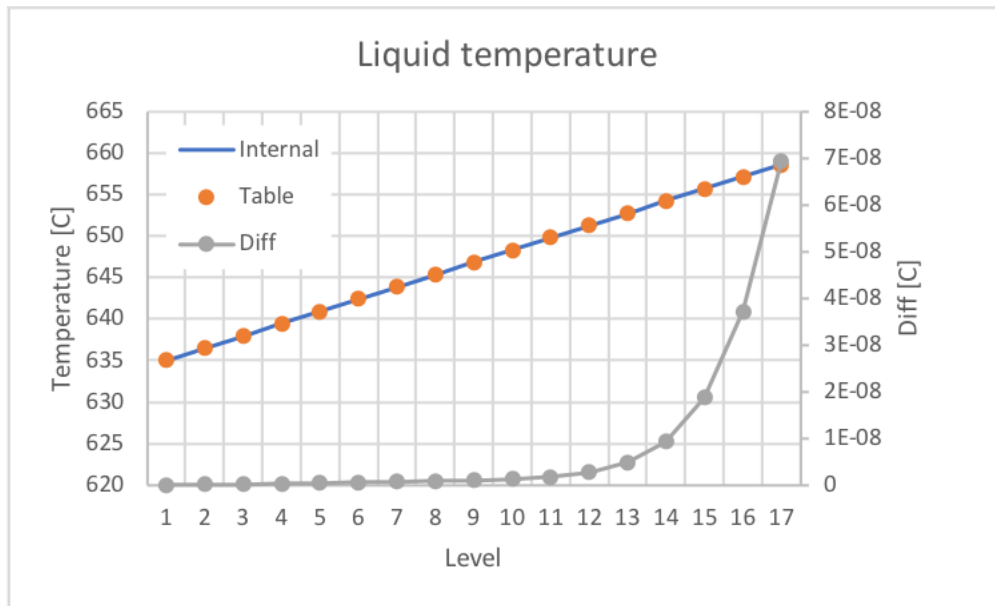


Figure 6. Comparison of liquid temperature predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.

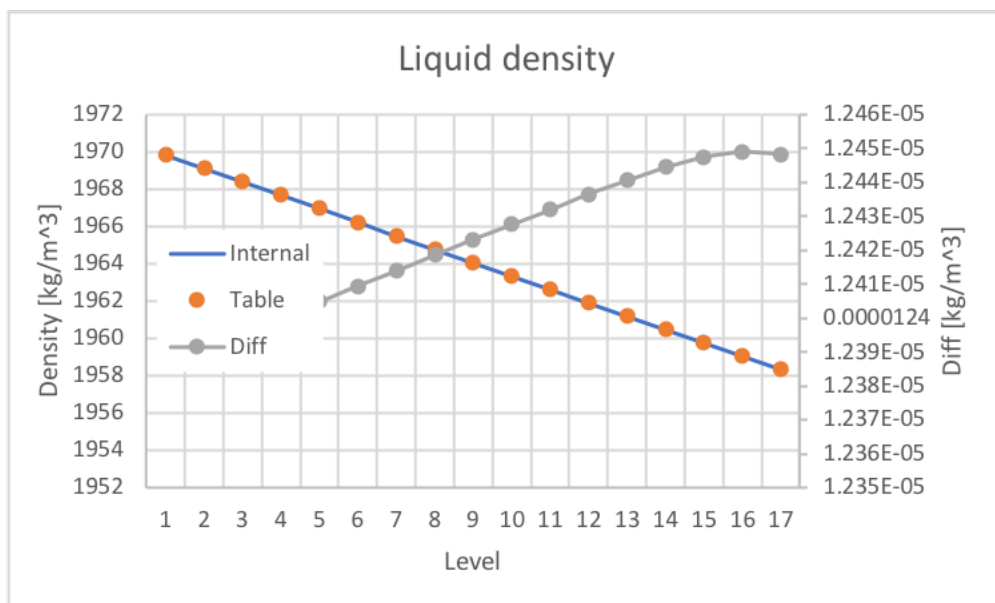


Figure 7. Comparison of liquid density predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.

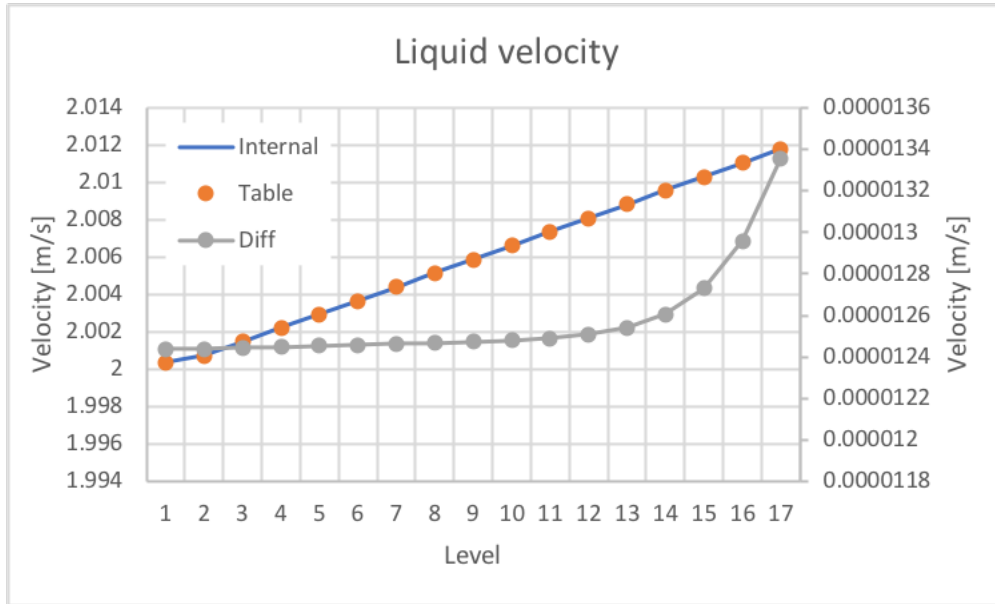


Figure 8. Comparison of liquid velocity predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.

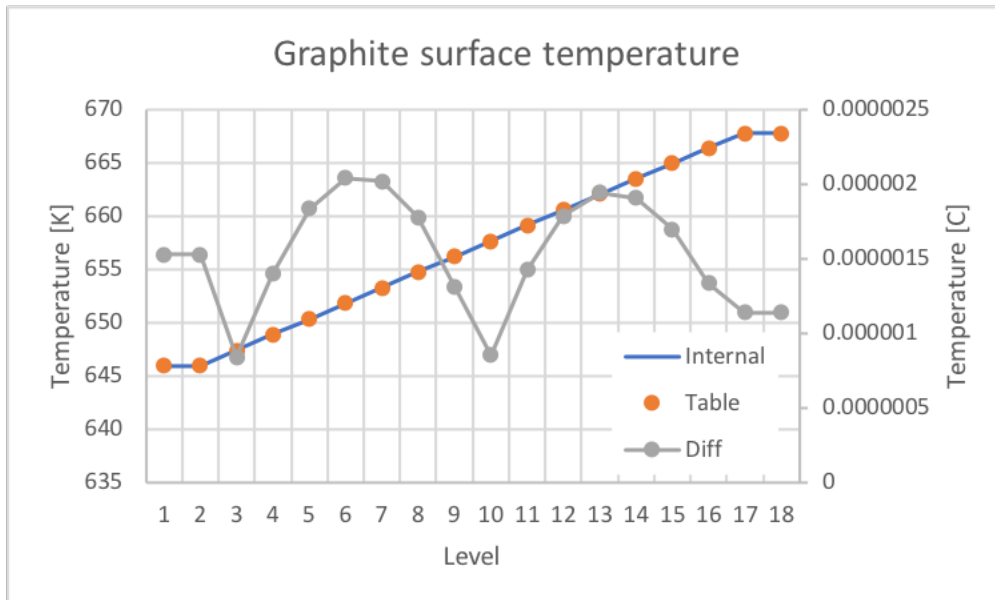


Figure 9. Comparison of graphite surface temperature predicted by CTF when using internal FLiBe properties vs. FLiBe properties read from the custom input file.

3. VERA DEMONSTRATION

3.1 MSRE Model Description

Two input files are set up to model the MSRE [6] in VERA: CTF for thermal hydraulics, and MPACT for neutron transport. The CTF model consists of the active core region and a simplified model of the primary loop, while the MPACT model only contains the active core region and the upper and lower plenum regions. For this demonstration, the MSRE was simplified so that it could be run using quarter symmetry. The actual model has control rods in the northwest, northeast, and southwest quadrants, with a sample basket in the southeast quadrant. These four positions were replaced with another graphite stringer to make the model quarter symmetric. Each code's model is described in more detail below.

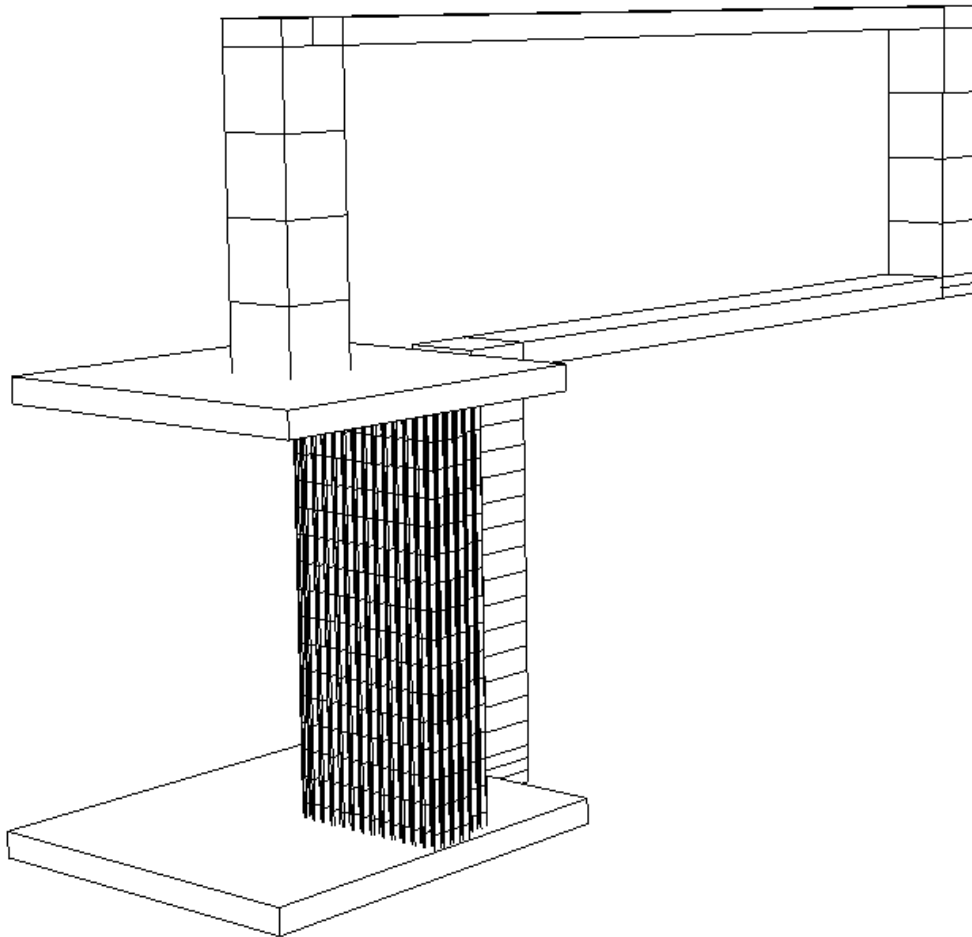


Figure 10. CTF mesh generated by the quarter symmetric MSRE model with a simplified primary loop model.

The CTF model includes 324 channels and is shown in Figure 10. Each channel contains at least 1 cell, but many have multiple cells in the axial direction. Channel 1 is a simplified model of the lower plenum region. The active core region consists of Channels 3–314. These channels all connect to Channel 1 at the bottom.

They are divided into 16 cells, each of which is 10.16 cm high, for a total height of 162.56 cm for the active core region. Channel 316 then models the upper plenum region and connects to the tops of Channels 3–314. Channel 320 connects to the top of 316 and consists of 4 levels. This channel is where the simplified model for the reactor pump is applied. Channel 320 is also where the helium injection takes place, though that is not used for these calculations. Channel 322 connects to the top of Channel 320, and it connects laterally to Channel 323, which contains the simplified model for the heat exchanger. Channels 2, 315, 317–319, 321, and 324 then form the remainder of the primary loop, connecting the heat exchanger down to the inlet plenum region.

The MPACT model only includes the lower and upper plenum regions and the active core. The model contains 162 graphite stringers. The face of each stringer was machined to contain half of a salt channel, for a total of 324 channels as in the CTF model. The upper and lower plenum regions are modeled as pure salt regions with the same cross sectional area as the active core region and the same height as in the CTF model. A slice at the centerline of the MPACT model is shown in Figure 11.

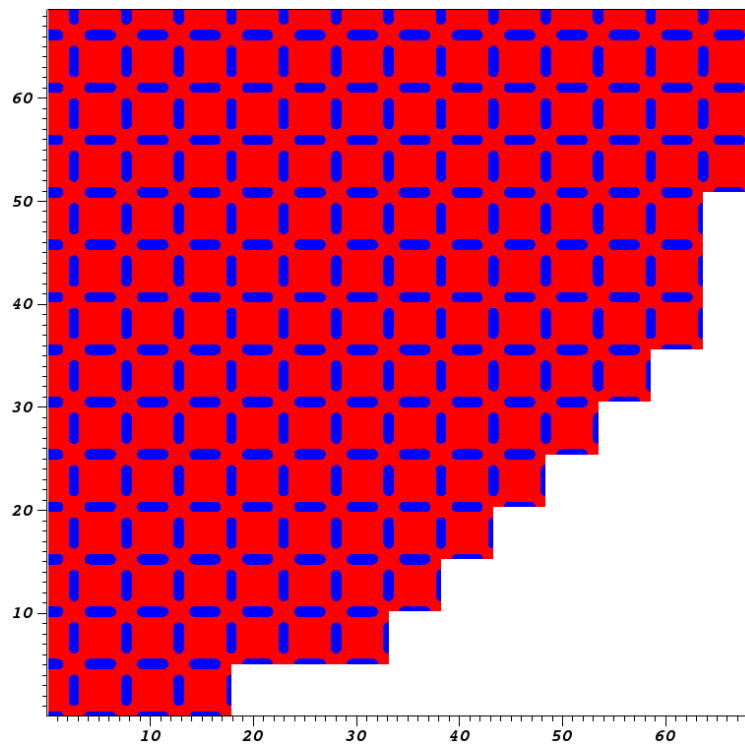


Figure 11. MPACT model generated by the quarter symmetric MSRE model showing graphite stringers (red) with fuel salt channels (blue).

During the VERA demonstration calculations, MPACT drives the coupled calculation. The name of the CTF input file is provided to MPACT, along with a set of coordinates to ensure that the models are aligned with each other since the two codes define their coordinate systems independently. MPACT then maps all of its fuel salt regions to the appropriate cells in the CTF model, allowing for the proper transfer between the two codes of volumetric power generation, salt temperature, and salt density in each cell of the CTF model.

3.2 Database File

ANL performed a literature review this year that resulted in a collection of thermophysical properties for various types of salt [5] that were delivered as a Microsoft Excel spreadsheet. One of the salt mixtures provided in this collection, LiF-NaF-KF, was selected for developing a VERA thermophysical properties database file. Thermophysical properties for this salt are summarized in Table 2. A molar mass of $0.126\,023\text{ kg mol}^{-1}$ was used for converting mols to kg. The database table was generated using a uniform temperature distribution with 113 points between the melt and boiling points.

Table 2. LiF-NaF-KF Salt thermophysical properties

Quantity	Equation	Units
Melting point	727.0	K
Boiling point	1843.2	K
Density	$2.6 - 6.2 \cdot 10^{-4}T$	g cm^{-3}
Dynamic viscosity	$2.5 \cdot 10^{-2} \exp 4.5 \cdot 10^3/T$	mPa s
Thermal conductivity	$3.6 \cdot 10^{-1} + 5.6 \cdot 10^{-4}T$	$\text{W m}^{-1} \text{K}^{-1}$
Specific heat	$9.8 \cdot 10^2 + 1.1T$	$\text{J K}^{-1} \text{mol}^{-1}$

3.3 Results

A coupled calculation of the MSRE model was run using CTF and MPACT, with the database mentioned above file used for the salt thermophysical properties in CTF. The resulting density distribution calculated by CTF using the database is shown in Figure 12. The temperature and density calculated by CTF feed into the MPACT calculations to determine the thermal hydraulic feedback to the neutron transport calculations. The resulting MPACT power distribution in the salt is shown in Figure 13. These results demonstrate the capability to run MSR models through VERA using the newly developed thermophysical properties database format.

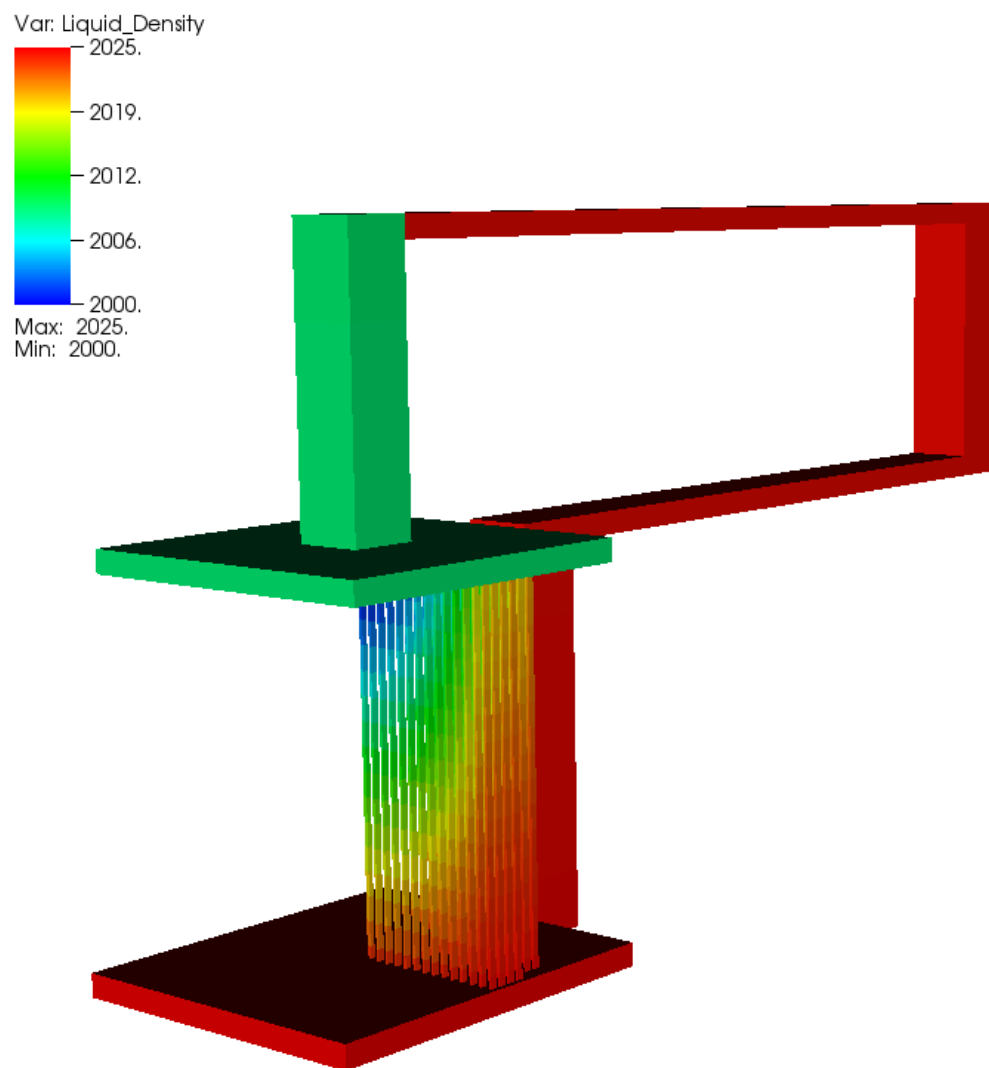


Figure 12. Density distribution (kg/m3) for the coupled MPACT-CTF MSRE calculation.**

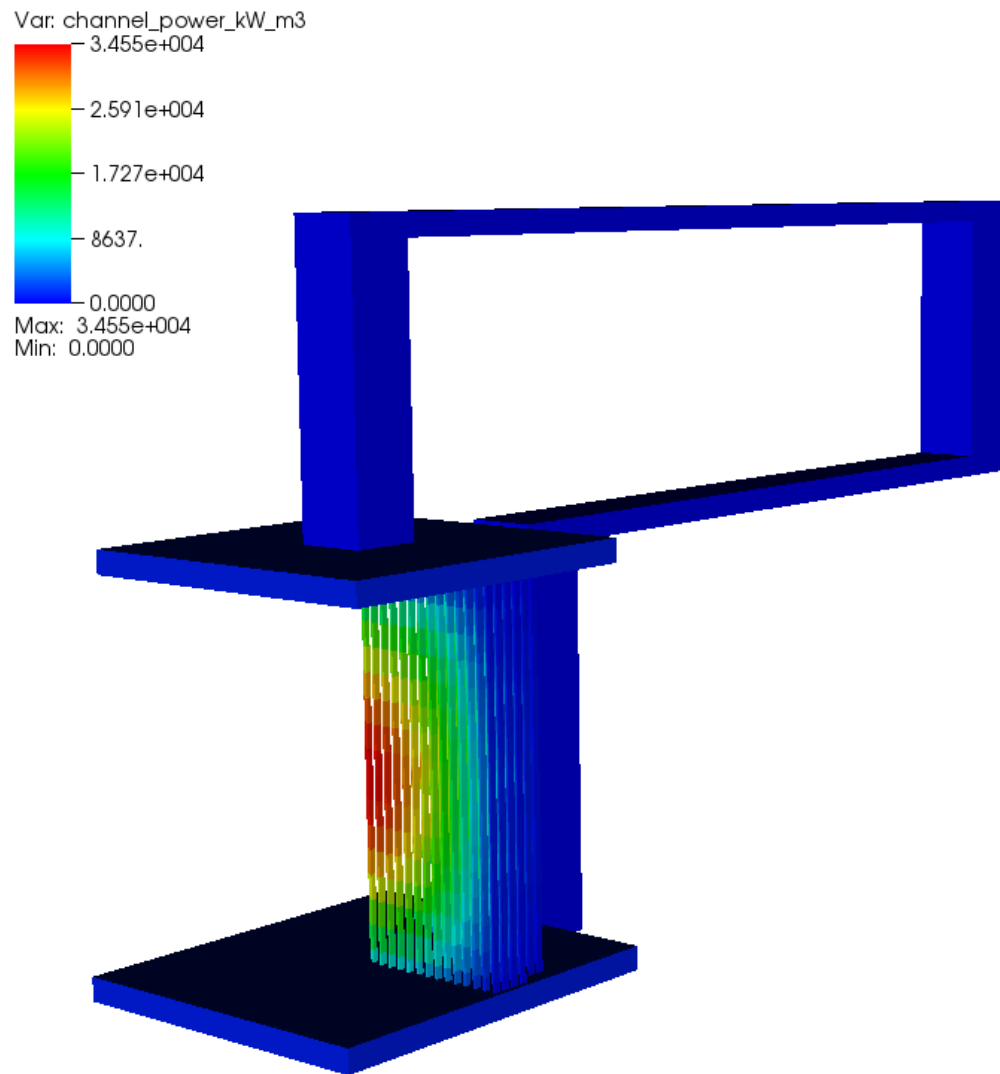


Figure 13. Power distribution (kW/m3) for the coupled MPACT-CTF MSRE calculation.**

4. THERMPHYS DATABASE INTERFACE

4.1 Description of ThermPhys

The ThermPhys interface is designed to provide a flexible, user-friendly user input experience, as well as a quality-assured data workflow compatible with the NEAMS Workbench analysis environment [8]. The user input syntax provides a human-readable format that is a more flexible, less error-prone data format than the previous approach of using Microsoft Excel spreadsheets. The data are also more accessible via processing support in the Workbench Analysis Sequence Processor (WASP) [9], and with the NEAMS Workbench integration of the Dakota Uncertainty Quantification (UQ) and optimization capabilities [13], the data are more easily configured for future sensitivity and uncertainty studies. The ThermPhys input captures all data currently reported by [5] and presents it in an intuitive format, as depicted in Figure 14. With the ThermPhys integration into the NEAMS Workbench, the user will have several additional benefits, such as input auto-completion, as depicted in Figures 15 and 16, as well as input verification. The ThermPhys application is under development and planned for release in FY20. It will process the thermophysical data and generate an HDF5 [3] -formatted file for access by any reactor analysis tool.

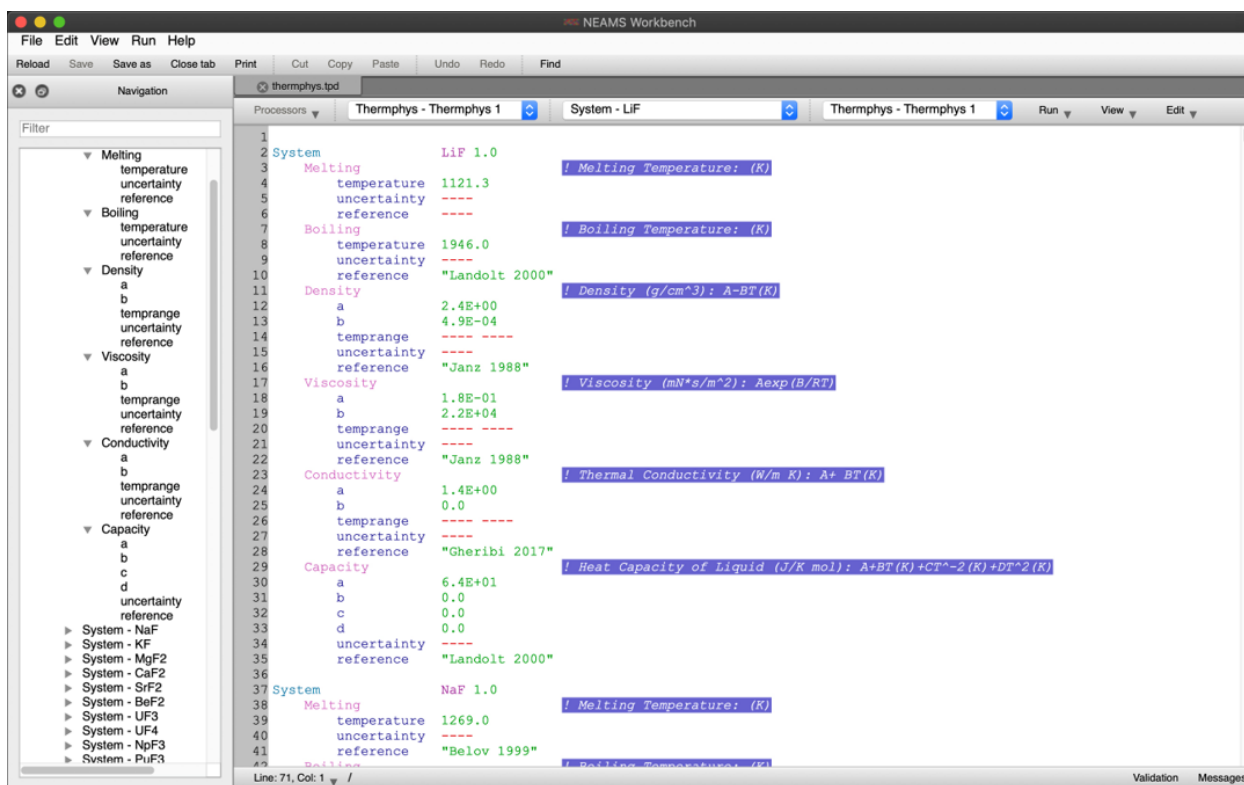


Figure 14. ThermPhys application input as highlighted within the NEAMS Workbench.

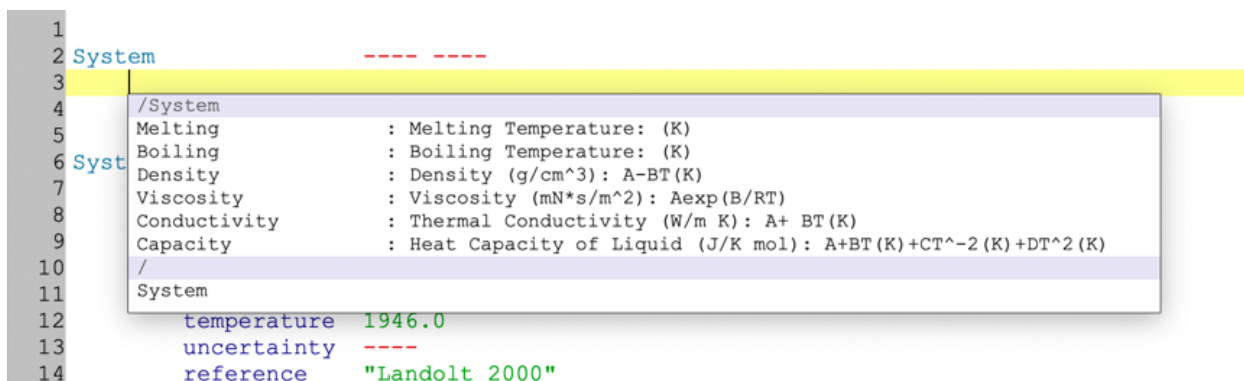


Figure 15. ThermPhys System auto-completion in the NEAMS Workbench.

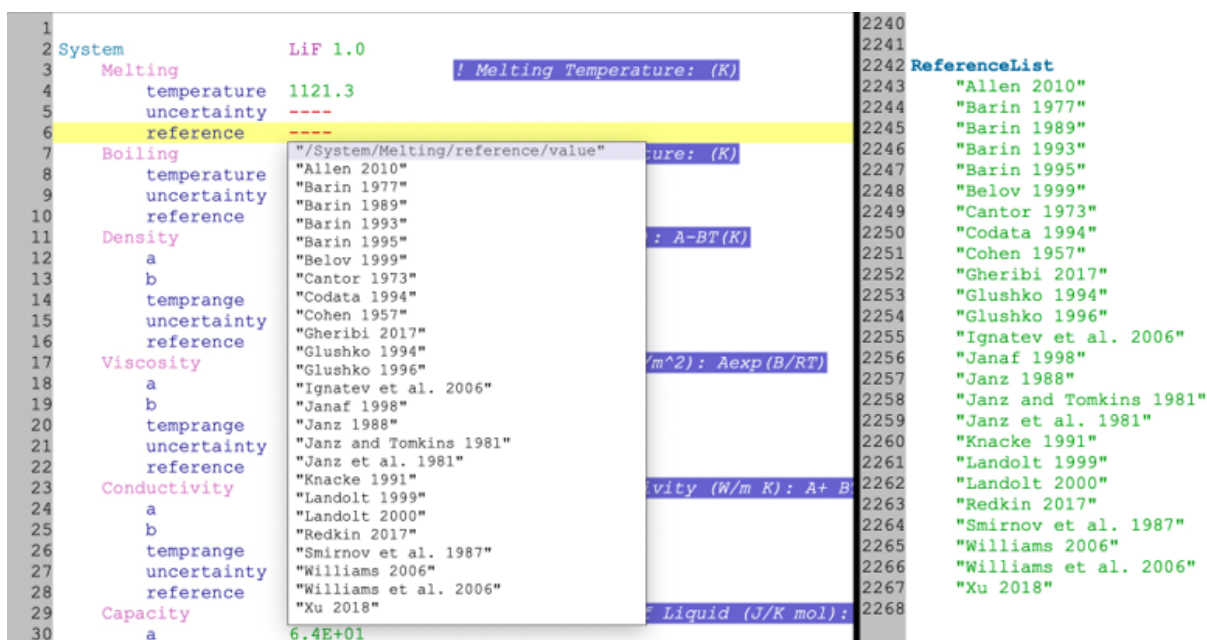


Figure 16. ThermPhys Reference auto-completion in the NEAMS Workbench.

4.2 Thermophysical Data Format

The thermophysical property reader discussed in Section 2.1 only considers the fluid properties to be temperature-dependent. This was implemented to develop a preliminary user-defined fluid property interface in the code. This interface will be improved in several ways in future work, as detailed below:

- The fluid properties will be modified to include a dependency on salt composition
- The input file format will be migrated to HDF5 for improved data compression and interfacing with the code
- The property interface class will be migrated to the open-source utility package, Futility [2], allowing for use in other simulation tools

The fluid-property interface class will be expanded to include the following procedures:

1. `get_composition`
2. `set_species`

The first item will allow the coupled code to extract the names of the salt components that exist in the database file. The second item will allow the coupled code to set the mole-percent of each component in the salt mixture. The previous fluid property procedures will still take pressure and temperature as their arguments. The “`set_species`” procedure was added separately so that the salt composition can be updated on a longer time scale, such as a depletion.

Each component of the salt ends up adding an additional dimension in the salt data table. One advantage of moving to the HDF5 format is that the data tables can be defined as multi-dimensional arrays, allowing for easier interfacing with the code. The property class will be expanded to perform multi-dimensional interpolation to work with the new datasets. It is intended that the data will be presented on a structured grid, and the following datasets will be present in the salt database file.

1. `pressure` (required): 1D vector of pressures at which thermophysical properties are being provided.
2. `temperature` (required): 1D vector of temperatures at which thermophysical properties are being provided.
3. `speciesX` (optional): 1D vector of mole percent of a species. Any number of species vectors can be provided and should be named `species1`, `species2`, ..., `speciesN`. Each provided vector adds another dimension to the property dataset array. The length of the vector defines the size of the dataset in that dimension. If provided, a species vector must have at least one value. The mole percents must be between 0.0 and 1.0.
4. `speciesNames` (optional): 1D vector of species names. If any species vectors are provided, then this is required. The names of the species must match what MPACT will pass into CTF when updating species concentrations. The size of this vector must match the number of species vectors provided. The ordering will be the same as the species vectors (e.g., entry 1 of the `speciesNames` vector will be the name of the species defined by `species1`).
5. `cp` (required): Specific heat. Shape will be 2 plus the number of species. The array will take pressure, temperature, species 1 mole percent, species 2 mole percent, ..., species N mole percent and return the specific heat for that combination. The independent values defining the state of each node in the array are given by the previously provided pressure/temperature/species vectors.
6. `thermalConductivity` (required)—thermal conductivity. This will use the same indexing scheme as `cp`.
7. `viscosity` (required): dynamic viscosity. Same indexing scheme as `cp`.
8. `density` (required): liquid density. This will use the same indexing as `cp`.
9. `Tmelt` (required): scalar value that provides the melting temperature of the salt.
10. `Tboil` (required): scalar value that provides the boiling temperature of the salt.

5. SUMMARY

This milestone report discusses the development of a thermophysical data format and its user and code interfaces. The data user interface that will allow users to enter salt properties into the database will be handled by the ThermPhys input, which is supported by the NEAMS Workbench and will be released in the next fiscal year. The code interface was integrated into the reactor analysis tool VERA, and a demonstration was performed in which data from ANL were used to perform a VERA MSR simulation. The current thermophysical data format will be expanded in the future to include salt composition effects.

6. REFERENCES

References

- [1] Consortium for Advanced Simulation of Light Water Reactors. <https://www.casl.gov>.
- [2] CASL Futility. <https://github.com/CASL/Futility>.
- [3] The HDF Support Group. <https://support.hdfgroup.org/HDF5/whatishdf5.html>.
- [4] A. Graham, B. Collins, R. Salko, R. Taylor, and C. Gentry. Development of Molten Salt Reactor Modeling and Simulation Capabilities in VERA. In *Proceedings of Global/Top Fuel 2019*, 2019.
- [5] J. Jerden. Molten Salt Thermophysical Properties Database Development: 2019 Update. Technical Report ANL/CFCT-19/6, Argonne National Laboratory, 2019.
- [6] R. Kedl. Fluid dynamic studies of the molten salt reactor experiment (msre) core. Technical Report ORNL-TM-3229, Oak Ridge National Laboratory, 1970.
- [7] B. Kochunas, D. Jabaay, S. Stimpson, A. Graham, T. Downar, B. Collins, K. Kim, W. Wieselquist, K. Clarno, J. Gehin, and S. Palmtag. VERA Core Simulator Methodology for PWR Cycle Depletion. Technical Report CASL-U-2015-0155-000, Consortium for Advanced Simulation of Light Water Reactors, 2015.
- [8] R. A. Lefebvre, B. R. Langley, P. Miller, M. Delchini, M. L. Baird, and J. P. Lefebvre. NEAMS Workbench Status and Capabilities. Technical Report ORNL/TM-2019/1314, Oak Ridge National Laboratory, 2019.
- [9] R.A. Lefebvre, B.R. Langley, and J.P. Lefebvre. Workbench Analysis Sequence Processor. Technical Report ORNL/TM-2017/698, Oak Ridge National Laboratory, 2017.
- [10] J. McMurray, T. Besmann, J. Ard, S. Utlak, and R. Lefebvre. Status of the Salt Thermodynamic Database, MSTDB. Technical Report ORNL/SPR-2019/1208, Oak Ridge National Laboratory, 2019.
- [11] R. Salko, A. Wysocki, M. Avramova, A. Toptan, N. Porter, T. Blyth, C. Dances, A. Gomez, C. Jernigan, and J. Kelly. *CTF Theory Manual*. The North Carolina State University, 2017.
- [12] R. Salko, A. Wysocki, J. Gehin, M. Avramova, A. Toptan, N. Porter, T. Blyth, C. Dances, J. Magedanz, M. Gergar, C. Gosdin, C. Jernigan, J. Kelly, and S. Palmtag. *CTF Validation and Verification*. The North Carolina State University, 2017.
- [13] Laura P. Swiler, Robert A. Lefebvre, Brandon R. Langley, and Adam B. Thompson. Integration of Dakota into the NEAMS Workbench. Technical Report SAND2017-7492, Sandia National Laboratory, 2017.
- [14] D. F. Williams, L. M. Toth, and K. T. Clarno. Assessment of Candidate Molten Salt Coolants for the Advanced High-Temperature Reactor (AHTR). Technical Report ORNL/TM-2006/69, Oak Ridge National Laboratory, 2006.