



Experience with DireWolf for Heat-Pipe Microreactor Analysis to Support Early Demonstration

June 2020

Changing the World's Energy Future

Travis Lange
Idaho National Laboratory



DISCLAIMER

This information was prepared as an account of work sponsored by an agency of the U.S. Government. Neither the U.S. Government nor any agency thereof, nor any of their employees, makes any warranty, expressed 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. References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

Experience with DireWolf for Heat-Pipe Microreactor Analysis to Support Early Demonstration

Travis Lange

June 2020

**Idaho National Laboratory
Idaho Falls, Idaho 83415**

<http://www.inl.gov>

**Prepared for the
U.S. Department of Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517**

Page intentionally left blank

ABSTRACT

Development of microreactor designs is gaining popularity for distributed generation and remote grid applications. The new number of advanced reactor designs under development is driving a need for advanced modeling and simulation to analyze these designs. A new code suite for microreactor analysis is being developed at Idaho National Laboratory called DireWolf. DireWolf is a MOOSE application that leverages the MOOSE framework to couple reactor physics, fuel performance, heat transfer, and heat pipe performance. A number of initial analyses have been performed using coupled and standalone physics modules in DireWolf that are relevant to heat-pipe microreactor analysis. The experience of using DireWolf within the microreactor program to date is documented below. Successful applications with both coupled and standalone physics modules have been performed and the limitations encountered are discussed.

Page intentionally left blank

CONTENTS

ABSTRACT	iii
ACRONYMS.....	vii
Acknowledgements.....	viii
1. INTRODUCTION AND BACKGROUND	10
1.1 Modeling and Simulation to support early demonstration.....	10
2. DIREWOLF CODE OVERVIEW.....	10
2.1 MOOSE	11
2.2 Rattlesnake.....	11
2.3 MAMMOTH.....	12
2.4 BISON	12
2.5 Sockeye.....	12
2.6 Magpie	14
2.7 CUBIT	14
3. EXPERIENCE WITH DireWolf FOR HEAT PIPE MICROREACTOR ANALYSIS	15
3.1 Description of Models	15
3.2 Workflow of Multiphysics DireWolf Simulations	17
3.3 Observations on DireWolf Usage	18
4. Summary	20
5. REFERENCES	20

FIGURES

Figure 1. DireWolf components and coupling with data transfer through MOOSE framework.....	11
Figure 2. Cross section of heat pipe modeled with Sockeye.	13
Figure 3. Different scale-length models used in Magpie for irradiation damage production modeling.....	14
Figure 4. Engineering drawing of single-heat-pipe monolith used for preliminary modeling in DireWolf (dimensions in inches).	15
Figure 5. Single heat pipe monolith mesh.	16
Figure 6. Engineering drawing of 37-heat-pipe monolith used for preliminary modeling in DireWolf.....	16
Figure 7. 37-heat-pipe monolith mesh created in Cubit.	17
Figure 8. Workflow for coupled, multiphysics heat pipe microreactor analysis using DireWolf.....	18
Figure 9. 37-heat-pipe monolith temperature distribution (in Kelvin) from BISON standalone.	19
Figure 10. MOOSE multi-app simulation solution of heat transfer and heat pipe thermal transfer for single-heat-pipe (left) and 37-heat-pipe (right) monoliths.	19

Page intentionally left blank

ACRONYMS

1D	One-dimensional
2D	Two-dimensional
3D	Three-dimensional
FEM	Finite element methods
GW	Giga Watt
INL	Idaho National Laboratory
JFNK	Jacobian-Free Newton Krylov
LANL	Los Alamos National Laboratory
MOOSE	Multiphysics Object Oriented Simulation Environment
MW	Mega Watt
NEAMS	Nuclear Energy Advanced Modeling and Simulation

Acknowledgements

The work performed herein could not have been performed without the significant professional support of the MOOSE developers under the support of the NEAMS program. I would like to acknowledge and thank the developers for their support in particular expertise and knowledge for aspects of the MOOSE framework and tools:

- Dr. Josh Hansel – for support in creating heat pipe models for simulation in Sockeye simulations. Thanks to Josh for sharing knowledge in the operations and behavior of heat pipes.
- Dr. David Andrs – for support in obtaining a working understanding of the MOOSE framework and incorporated physics modules. David also supported the linking of sub-apps into a MOOSE multi-app model for coupled simulations.
- Dr. Vincent Laboure – for support in understanding the neutronic models and workflows currently used in coupled reactor analysis.
- Mr. William Hoffman – for support in developing BISON inputs for standalone and coupled simulations. And for support in generating meshes in Cubit for use in MOOSE simulations.
- Dr. Sebastian Schunert – for help in understanding the development and coupling of Magpie into coupled reactor physics simulations and associated workflow.

I would also like to acknowledge and thank Dr. Piyush Sabharwall for his management and direction of this research process of exercising DireWolf for heat-pipe microreactor analysis.

Thanks to Dr. Jess Gehin for his support in directing and advising the scope and direction of the research in order to fulfill our mission in this milestone.

Page intentionally left blank

Experience with DireWolf for Heat-Pipe Microreactor Analysis to Support Early Demonstration

1. INTRODUCTION AND BACKGROUND

Many new designs are being considered to advance nuclear power production with different power capacities, safety features, and physical sizes. Today's nuclear power fleet is mainly comprised of large light-water reactors with power capacities over 1 GW thermal energy with many being around 3 GW thermal energy. As an alternative to large stationary power reactors with sites and exclusion zones more compact, reactors have been and are being designed to take advantage of economic and technical characteristics of smaller reactors. Microreactors are one classification of smaller reactors with a power capacity of less than 20 MW thermal energy. Microreactors are designed to be easily transportable, manufacturable in a factory, and simple in neutronic design to enable fully or semi-autonomous operation [1]. These characteristics allow it to be used in many special purpose applications where reliable distributed power generation is necessary, such as in forward military bases, remote communities, and emergency response for disaster scenarios [2].

Although there are many meaningful applications for reliable, safe, and transportable microreactors, with a few historical exceptions none have been built to date. This means that there is no operating experience to know the true characteristics of these microreactor designs, other than historical reactor operation data. This is where advanced modeling and simulation of coupled multiphysics tools can bridge the gap without risking large investments to build untested microreactor designs.

The DOE microreactor program is collaborating with the Nuclear Energy Advanced Modeling and Simulation (NEAMS) program on the development, use and validation of modeling and simulation tools for microreactors. The work described in this report documents recent efforts on using the NEAMS tools within the microreactor program.

1.1 Modeling and Simulation to Support Early Demonstration

Research and development in advanced three-dimensional (3D) modeling and simulation of nuclear reactors has increased in the last decade as low-cost supercomputing has enabled researchers and academics to implement physics models that were computationally expensive just a few years ago. The advancement of supercomputing in research laboratories has paved the way for the technology to penetrate the market into industry applications. Nuclear fuel vendors can now invest in supercomputers that can be utilized to take advantage of advanced modeling and simulation tools.

The objective of this study is to exercise the capabilities of these advanced tools, which are under ongoing development to assess their readiness level for modeling and simulation of heat-pipe-cooled microreactor analysis. The goal of this current assessment is to use these tools to analyze a heat-pipe-cooled microreactor to understand the current performance and capabilities of the tools and to provide feedback to the software developers.

2. DIREWOLF CODE OVERVIEW

DireWolf is a Multiphysics Object Oriented Simulation Environment (MOOSE) based application that is being developed by NEAMS for microreactor analysis. It contains the following MOOSE-based applications: Rattlesnake for radiation transport, MAMMOTH for reactor physics, BISON for fuel and materials performance, and Sockeye for heat pipe performance [3]. These applications are then coupled together through the MOOSE framework using the multi-app and transfer systems. 3DireWolf is under active development. This report will focus on applying the state-of-the-art DireWolf code to determine the extent of its capabilities and usability in heat-pipe heat transfer analysis.

2.1 MOOSE

The NEAMS program is dedicated to developing capabilities for advanced modeling of simulation for nuclear energy. One of the core capabilities used to achieve their goals of advanced modeling and simulation is MOOSE. MOOSE is a framework that provides the solution engine for multiple physics over multiple length and time scales. MOOSE is a finite element framework written in C++ that allows for flexible and modular applications to be built on top of it. MOOSE manages data transfer between additional physics modules called “apps.” MOOSE also contains some fundamental physics packages such as tensor mechanics, heat conduction, and porous flow among other capabilities [4,5,6].

MOOSE contains the numerical solvers necessary for these physics packages. A parallel nonlinear solver, known as the Jacobian-Free Newton Krylov (JFNK) method, is implemented in MOOSE to effectively couple the systems of equations used in the multiple physics packages.

MOOSE is the framework that couples other physics packages together resulting in an internally coupled, multiphysics simulation. The set of physics packages applied to heat-pipe-cooled microreactor simulation is collectively known as DireWolf [7]. DireWolf uses the transfers and multi-app systems from MOOSE to couple the individual apps together. Each application can be executed on its own or with another one in a fully coupled multiphysics simulation. In this work a graphical depiction was generated of the different components of DireWolf along with the relevant coupling data transferred between each application and is shown in Figure 1.

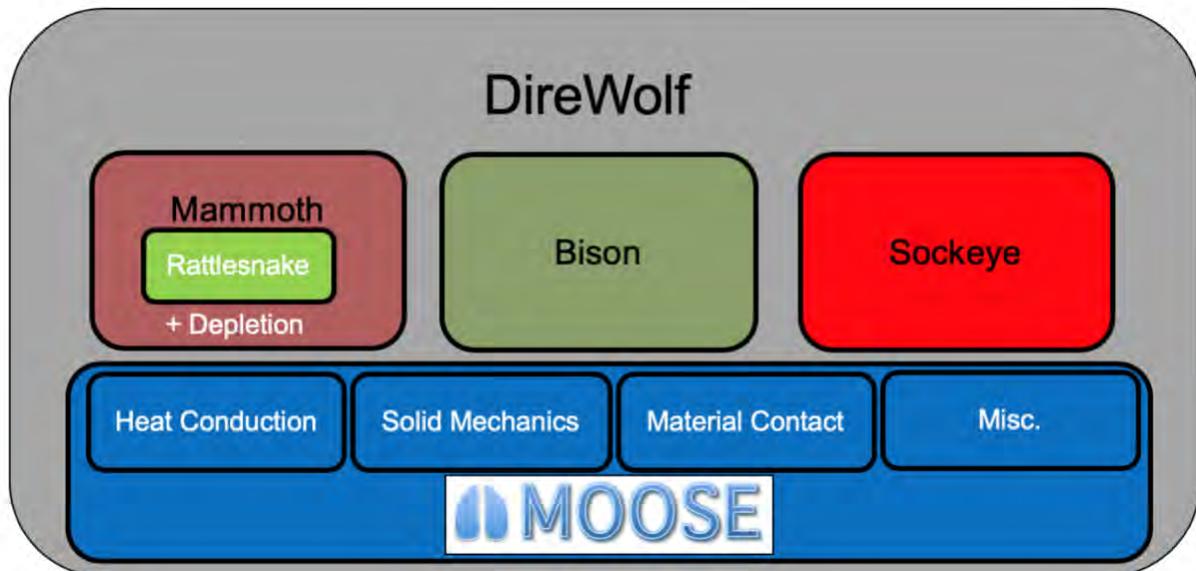


Figure 1. DireWolf components and coupling with data transfer through MOOSE framework.

2.2 Rattlesnake

Rattlesnake is an application built on the MOOSE framework to solve 3D, steady-state and time-dependent radiation transport [8]. In a nuclear reactor, neutrons are the primary particle of importance to track their direction, energy, and material interactions. Secondary particles of interest for radiation transport calculations are photons and electrons. The interaction of neutrons with materials, particularly fuel, are calculated in terms of reaction rates that feed into calculations for reactor criticality, heat generation and transport, and material evolution and performance.

Rattlesnake is a deterministic radiation transport solver that can use multiple transport schemes including multigroup diffusion, spherical harmonics, along with first and second-order S_n transport

methods. Rattlesnake is capable of handling highly nonlinear coupling through the MOOSE framework to analyze strong reactor transients along with other nonlinear dependencies from other coupled physics codes. Rattlesnake uses the linear Boltzmann equation to solve k-eigenvalue problems with multigroup transport. Rattlesnake solves the radiation transport using finite element methods (FEM). This requires a two-dimensional (2D) or 3D mesh of the geometry to solve spatially dependent quantities [9]. The neutronics mesh used by Rattlesnake in MOOSE can be independent from the mesh used by the other sub-apps, with some limitations. Rattlesnake is also capable of using multiple transport schemes within different regions of its own mesh. This allows for efficient solving in different parts of the mesh depending on their material and geometric complexity.

2.3 MAMMOTH

MAMMOTH is an application built on MOOSE for reactor physics designed to provide geometry-agnostic analysis of a wide variety of reactor concepts. MAMMOTH is also built on top of Rattlesnake, meaning that MAMMOTH inherits all of Rattlesnake's radiation transport capabilities while adding its own set of physics. MAMMOTH can solve steady-state and transient core performance, fuel depletion and performance, irradiation effects on the core internals and reactor pressure vessel, criticality and decay heat calculations, and reprocessing and non-destructive post-irradiation examination [10].

MAMMOTH leverages Rattlesnake as a sub-app for radiation transport and adds cross-section storage, self-shielding and cross-section generation, microscopic and macroscopic depletion, decay heat calculations, and homogenization equivalence. MAMMOTH manages cross-section data, which is generated with another code, usually Serpent [11], that uses Monte Carlo to generate the macroscopic, multigroup cross sections [12].

MAMMOTH has been applied previously in a coupled multiphysics simulation and analysis for the High-temperature Test Reactor [13].

2.4 BISON

BISON is a finite element fuel performance application built on the MOOSE framework. BISON solves material physics for both fuel and non-fuel materials. The fuel models can be solved for pellets, TRISO particles, metallic rods, and plates [14]. BISON solves 2D axisymmetric and 3D geometries with coupled thermo-mechanics and species diffusion. The fuel models include fission product swelling, densification, thermal and irradiation creep, fracture, fission gas release, and thermal properties dependent on temperature and burnup. Non-fuel material models in BISON include plasticity, irradiation growth, and thermal and irradiation creep calculations. For fuel pellets inside cylindrical metallic clad, traditional fuel in light water reactors, there are models in BISON to calculate gap heat transfer, mechanical contact, and the evolution of the gap/plenum pressure with plenum volume, gas temperature, and fission gas addition [15].

BISON can be coupled with the mesoscale-length fuel performance code MARMOT to perform multiscale fuel performance. BISON has already been demonstrated with a single microreactor assembly [1].

2.5 Sockeye

Sockeye is a MOOSE-based application that calculates heat pipe performance and operating limits. Sockeye uses governing equations for two-phase flow that derive from the seven equations used in RELAP-7 to model thermal-hydraulic systems. Sockeye takes these flow models and applies them to specific heat pipe geometries with a focus on liquid metal working fluids for high-temperature applications such as microreactors. However, lower temperature working fluids should work since they

are governed by the same physical equations, but they have not been tested and are not supported directly at this time [16].

Sockeye is currently being developed to analyze “conventional” heat pipe geometries that consist of an outer cylindrical cladding with an interior concentric wick with a central flow annulus for vapor transport. A depiction of the geometry assumed in Sockeye is shown in Figure 2 [16].

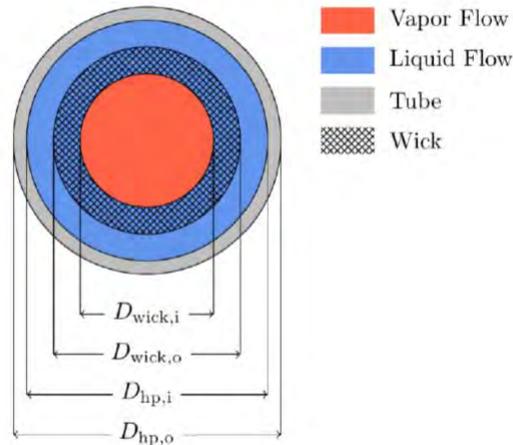


Figure 2. Cross section of heat pipe modeled with Sockeye.

The wick structure is assumed to be homogenous and annular, such as sintered metal or a type of screen mesh. More complex designs, such as arterial wicks, are not included in Sockeye.

Sockeye models two-phase flow inside the heat pipe with a 1D approximation for normal operation. Sockeye is capable of 2D heat transfer through the heat pipe cladding that links with the 1D flow channel model. Heat pipe startup is approximated with an aggregate heat transfer model that conserves the energy balance in a lumped approximation. A more realistic three-phase startup model is being developed in Sockeye but is not yet ready. Sockeye can be coupled to other MOOSE apps through heat transfer boundary conditions between applications [3].

Sockeye is under development to calculate heat pipe operating limits, including the sonic limit, the entrainment limit, the wicking limit, and the boiling limit. These operating limits cannot yet be calculated with Sockeye and are planned to be implemented in future work.

Sockeye currently incorporates physics to capture the effects of heat pipe operating limits, including the sonic limit, the entrainment limit, the wicking limit, and the boiling limit. However, Sockeye has yet to demonstrate validation with analytic operational limit models. Development is planned for special models for entrainment and boiling.

The verification and validation of Sockeye is limited. An initial demonstration and benchmarking have been performed against a simplified 1D heat pipe code developed at Los Alamos National Laboratory [17].

Once completed, Sockeye will provide necessary information for the transient and steady-state operation of a heat-pipe-cooled microreactor design through the heat pipe performance and limits.

2.6 Magpie

Magpie is a MOOSE-based application that calculates irradiation damage production and evolution on multiple length scales. Magpie's initial capability was modeling nano-scale ballistic mixing. It can simulate steady-state recoil cascades leading to phase separation in immiscible alloys. This capability has been expanded to the engineering and micron scales across the finite element mesh used in the MOOSE framework. A depiction of these scale length models is shown in Figure 3. The nanometer scale simulations are the most accurate tracking multiple generations of collisions and energy deposition. However, the nanometer scale simulations are very expensive computationally and are not needed for every simulation. The engineering scale simulations track partial cascades and approximate the energy deposition below a threshold, which significantly decreases computational expense while maintaining accuracy of the high-energy part of the cascade. The micron scale simulations are the least computationally expensive but only considering the primary recoils [18].

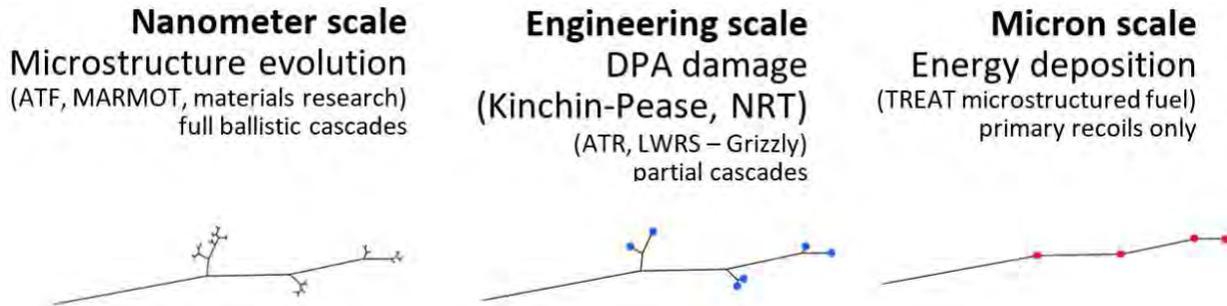


Figure 3. Different scale length models used in Magpie for irradiation damage production modeling.

Magpie currently uses a Binary Collision Monte Carlo model for simulating energy deposition and transfer in damage cascades. This model tracks individual collisions and energy deposition between particles. This can be very computationally expensive. A significant part of the computational effort is spent at low-energy interactions at the end of the cascade. To speed up the calculations, the cascade trajectories are truncated below a threshold energy in which the energy deposited will not likely exit the current finite element of the mesh that it is in [19]. This allows for quicker simulations when the larger length scale is of primary interest.

Magpie can be coupled with the radiation transport code Rattlesnake through the MOOSE multi-app process. This allows for calculation of local irradiation damage of materials on the neutronic mesh.

Magpie will be used to calculate the irradiation damage (measured in displacements per atom [DPA]) of materials in the heat-pipe-cooled microreactor design. This information will be used to analyze the useful lifetime of materials in the reactor and damage rates over the lifetime to ensure acceptable material performance and behavior.

2.7 CUBIT

Cubit is a meshing tool that is external to MOOSE. Cubit is developed and maintained at Sandia National Laboratory [20]. It is a flexible and powerful toolkit for the generation of 2D and 3D meshes and geometries for finite element analysis [21]. Cubit can be executed through command line interface or through a graphical user interface which allows a flexible user experience with or without visualization of the mesh as it is generated. Cubit is well documented online with a user-manual and tutorials available for mesh generation. Cubit is available to U.S. government contractors for free but must be purchased as a separate software package, known as Trelis, for commercial or academic use.

MOOSE currently has an internal Mesh Generator system that is capable of meshing cores by generating multiple mesh generators and linking them together in an arbitrarily complex manner.

However, MOOSE can also use meshes generated by Cubit. This allows a user to select their preferred mesh generation software. Cubit was used for the generation of meshes in this work.

3. EXPERIENCE WITH DireWolf FOR HEAT-PIPE MICROREACTOR ANALYSIS

Preliminary analysis to exercise the DireWolf code and to define a workflow for coupled, multiphysics modeling and simulation of heat-pipe microreactor designs has been performed. The goal of the initial applications of DireWolf is to test the standalone and coupled, Multiphysics capabilities ensuring convergence and that results are not non-physical. Two models consisting of heat pipes, heater rods, and a monolith block have been generated that correspond to planned experiments at INL in the Microreactor Agile Nonnuclear Experimental Testbed facility. The models are described in more detail below.

3.1 Description of Models

The initial model used in exercising standalone and coupled portions of DireWolf consists of a single heat pipe surrounded by six heater rods in a hexagonal monolith housing. A radial view of the monolith is shown in Figure 4. The initial analysis exercises the coupling between BISON for heat transfer and Sockeye for heat pipe performance.

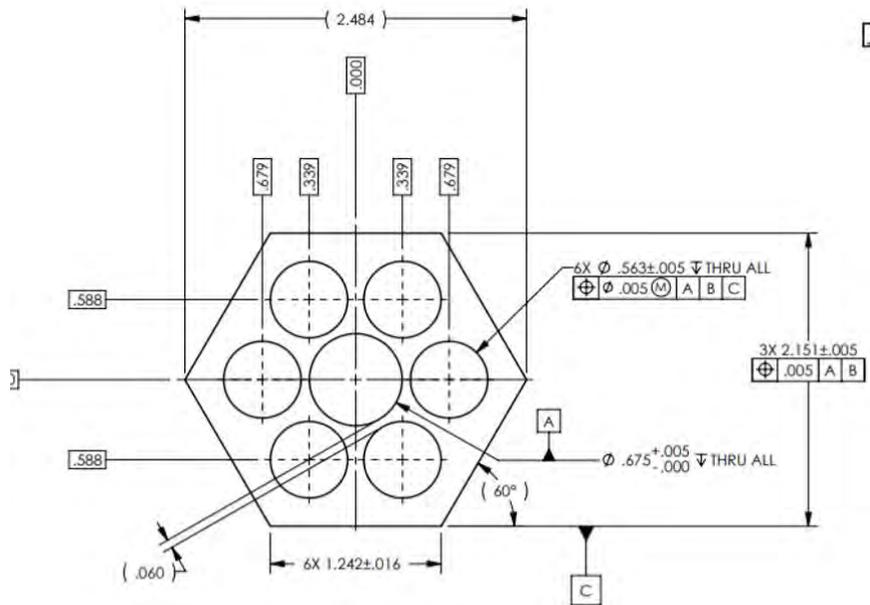


Figure 4. Engineering drawing of single heat pipe monolith used for preliminary modeling in DireWolf (dimensions in inches).

A mesh was generated using Cubit as an input into the BISON heat transfer simulation. The completed mesh for the single-heat pipe monolith with heater rods is depicted in Figure 5.

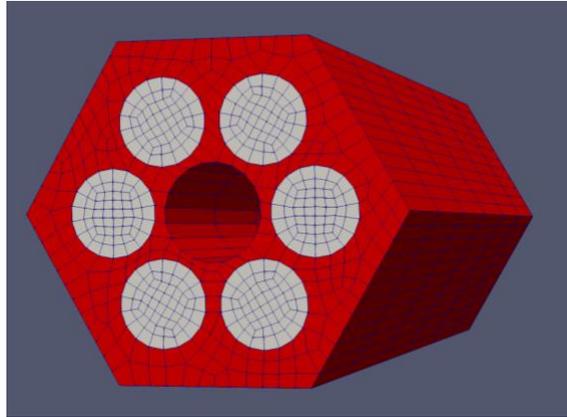


Figure 5. Single heat pipe monolith mesh.

A 37-heat-pipe monolith model was also generated that corresponds to a second phase of heat pipe experiments planned at INL. The multiphysics simulation for this model incorporates testing multiple instances of Sockeye for multiple heat pipe solutions into a single simulation. The geometry for the 37-heat-pipe monolith is shown in Figure 6.

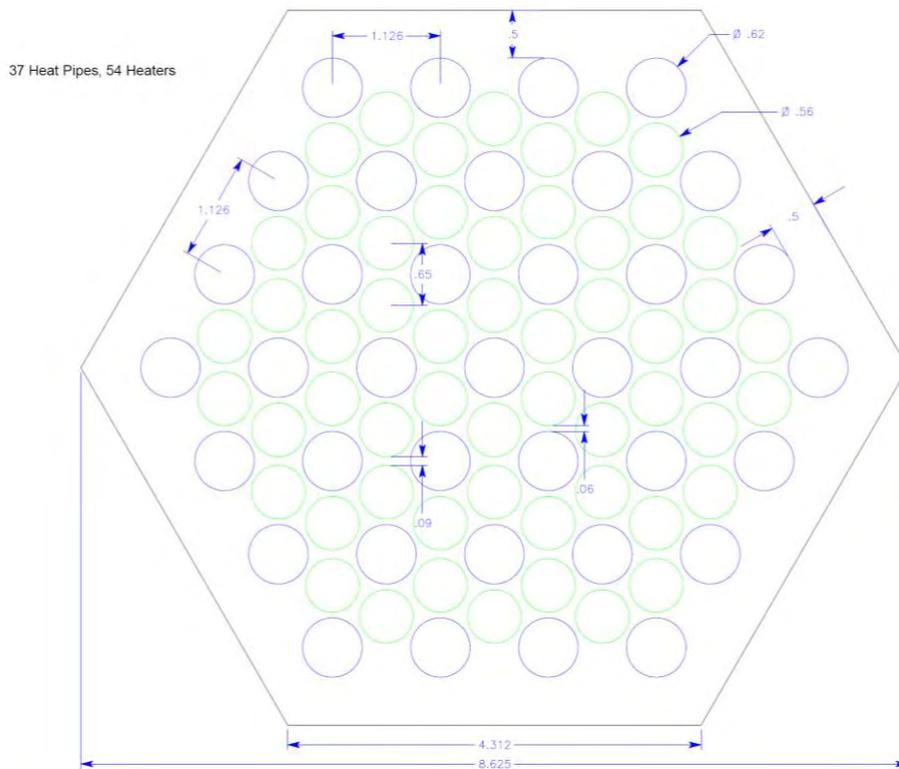


Figure 6. Engineering drawing of 37-heat-pipe monolith used for preliminary modeling in DireWolf.

A Cubit 3D mesh was generated for the geometry for finite element analysis in MOOSE. The mesh is shown in Figure 7 that includes heater rods (multi-colored) and holes for heat pipes.

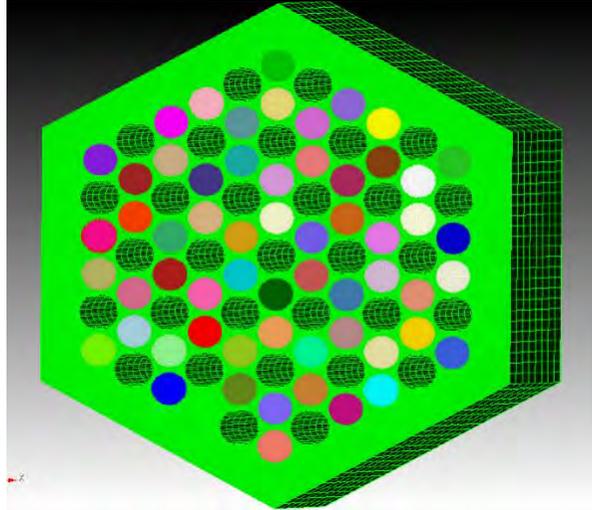


Figure 7. 37-heat-pipe monolith mesh created in Cubit.

3.2 Workflow of Multiphysics DireWolf Simulations

A workflow diagram with the required inputs and generated outputs is shown in Figure 8 **Error! Reference source not found.** A few caveats must be disclosed; this may not be the only workflow available for DireWolf applications, but represents the workflow used in this work. Depending on the simplicity of the mesh, the internal meshing capability may be used. But for geometries of interest, an external meshing toolkit is likely needed. The workflow is constantly under development and is improving all the time. However, the workflow shown in Figure 8 **Error! Reference source not found.** is a snapshot of the required processes at the time of this analysis.

The workflow begins by generating the geometries and meshes for the reactor design to be analyzed. This will likely require an external meshing tool kit, such as Cubit, to capture the requisite details of the reactor design. Two separate meshes must be generated: one for the neutronic analysis and one for the thermal/materials analysis. This is not strictly necessary, but the fine resolution needed for the heat transfer would make the neutronic solve computationally untenable. Therefore, a course neutronic mesh is needed to be coupled to a finer structural mesh.

As mentioned earlier, each standalone application in DireWolf (e.g., MAMMOTH, BISON, Sockeye, and Magpie) requires its own individual MOOSE-based inputs. Each input file shares a common syntax but has some unique input requirements as well. In practice, the user received assistance from each developer to generate the input required for the individual applications. A coupled neutronics, heat transfer, heat pipe performance, and irradiation damage analysis in DireWolf currently uses four MOOSE-based inputs and two meshes. Many of the files shown in the workflow diagram are outputs from one application and used as an input in another application. The input file requirements will change over time as DireWolf is developed and optimized for reactor analysis. Currently DireWolf has exposed the full flexibility and capability of the code that requires the aforementioned multiple input files.

Additional steps must be taken to generate a cross-section library for MAMMOTH. The GRIFFIN code that is being developed to replace MAMMOTH will have internal cross-section generation capability, but this capability is still under development. Currently cross sections are generated through an external Monte Carlo neutronics code called Serpent. Serpent is used independently in reactor analysis and requires its own model generation and inputs. An additional “include” file is created by the user and added to a standard Serpent input to have Serpent generate the correct output files that will be post-processed into the cross-section library for MAMMOTH. After generating multiple Serpent output files for cross sections at different reactor states (usually different temperatures), the Serpent outputs are

post-processed with the “ISO_XML” script that has its own input file to direct the postprocessing. Finally, the cross-section library is generated and can be pointed to the MAMMOTH input file for radiation transport simulations.

DireWolf is the driving application where microreactor-specific processes are housed. DireWolf creates instances of the other applications to solve their individual physics and communicates their results to be shared with the other applications. A given simulation may only require a single application instance (MAMMOTH and BISON), while others may need multiple instances of the application created during the simulation (Sockeye and Magpie). Usually, each instance of an application generates its own output file. Therefore, in this simulation there is one MAMMOTH output, one BISON output, multiple Sockeye outputs (reflecting the total number of heat pipes in the model), and multiple Magpie outputs. There is an internal data processing capability in MOOSE to pass results generated by individual applications to the master application where various mathematical operations can be performed on the data to generate a single output with processed results. The internal data processing instructions must be included in the input files before simulation execution to generate the desired processed results. The individual outputs may also be analyzed separately or post-processed to be analyzed collectively. Additionally, there are two software packages, Peacock and Chigger, that can process all of the outputs simultaneously.

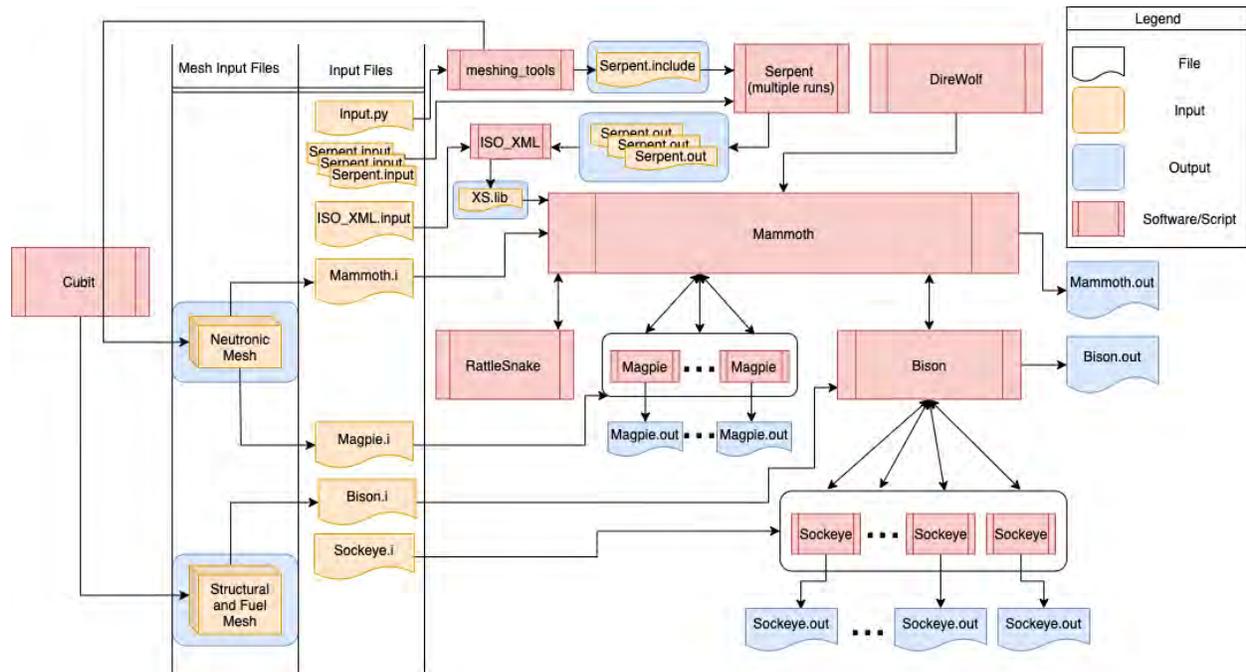


Figure 8. Workflow for coupled, multiphysics heat-pipe microreactor analysis using DireWolf.

3.3 Observations on DireWolf Usage

Successful simulations were performed for both standalone and coupled simulations for heat pipe performance and heat transfer in the single and 37 heat pipe models. Due to time limitations the mesh and input files required for radiation transport or reactor physics simulations were not completed. The results for the successful standalone simulation of heat transfer in BISON for the 37 heat-pipe monolith is shown in Figure 9.

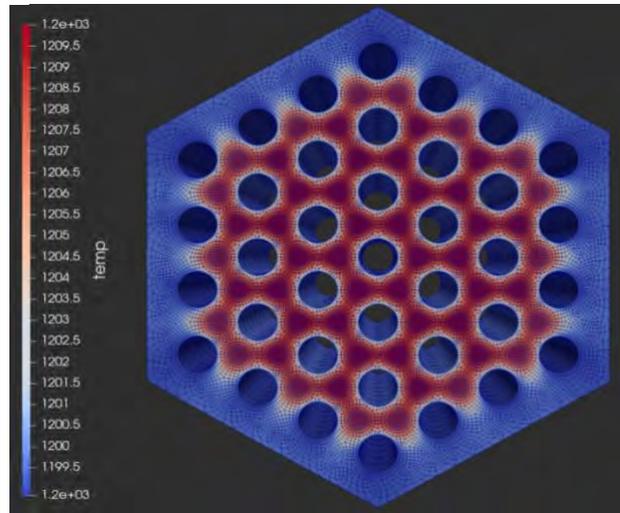


Figure 9. 37-heat-pipe monolith temperature distribution (in Kelvin) from BISON standalone.

Coupled multiphysics simulations were successfully performed for both single heat pipe and 37 heat pipe models. The single heat pipe model was simulated first to exercise the multiphysics coupling with a single instance of Sockeye. Then the 37-heat-pipe model would be simulated with multiphysics coupling to test the parallel solving of multiple heat pipes in a single model.

The results of the initial multiphysics applications is shown in Figure 10. Both models converged successfully with no obvious errors in the solution. This is a successful initial application of the coupled, nonnuclear physics modules in DireWolf.

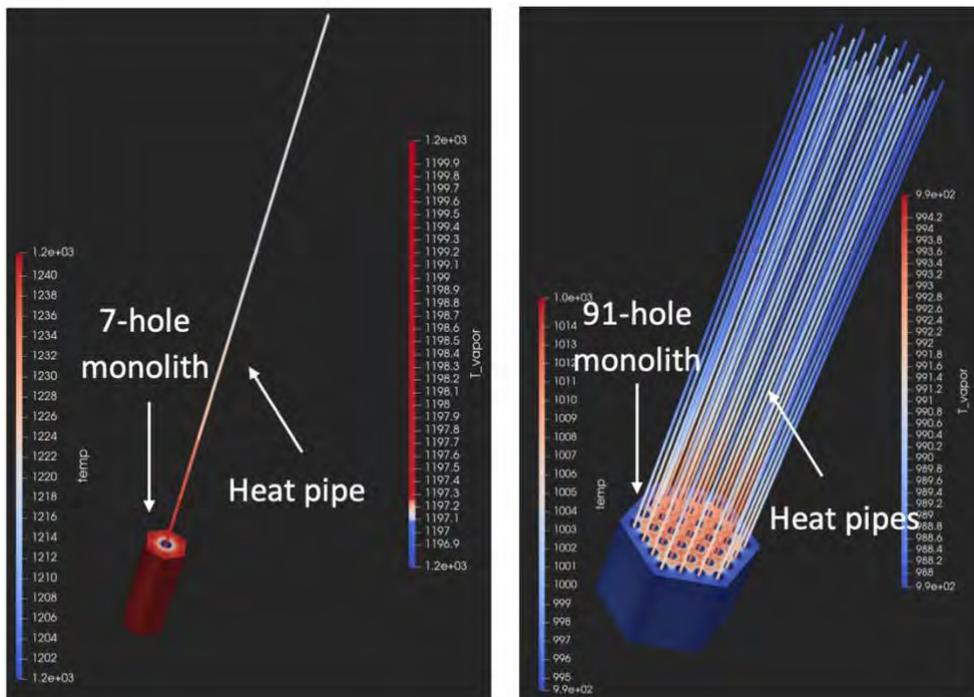


Figure 10. MOOSE multi-app simulation solution of heat transfer and heat pipe thermal transfer for single heat pipe (left) and 37-heat-pipe (right) monoliths.

Due to time limitations, the coupled heat transfer and heat pipe performance simulations were the largest coupled simulations to be performed. Initial steps were taken to generate the mesh and input files for reactor physics simulations, but they were not completed in time.

Throughout the standalone and coupled simulation modeling, significant developer support was needed. There was a range of user documentation for different parts of the workflow. Some parts of the workflow were documented with excellent user documentation. While other parts of the workflow were not documented for users yet and the information had to be gathered from the developers. This is not uncommon for actively developed codes, especially for the newly developed capability.

From a user perspective, streamlining inputs and the workflow would facilitate usage. The streamlining process is currently in progress for parts of the code suite. The NEAMS program is actively developing a new capability called GRIFFIN. GRIFFIN is intended to replace MAMMOTH and incorporate additional capabilities and streamlining much of the workflow for reactor analysis. INL is partnering with Argonne National Laboratory to incorporate into GRIFFIN the capabilities of Argonne's MC2-3 code, for generating multigroup cross sections, and PROTEUS code, for deterministic Sn, method of characteristics, and nodal methods. The ongoing development of GRIFFIN should significantly increase the usability of DireWolf and streamline the workflow for coupled reactor physics simulations.

It is recommended that the NEAMS program utilize their App Drivers technical area for code testing. The problems identified in this report would be useful problems to demonstrate capability. The microreactor program can work with NEAMS to define additional test problems as well as providing information on experiments being performed by the microreactor program to support validation.

4. Summary

DireWolf is a multiphysics MOOSE-based application for microreactor design and analysis. It has been used to exercise its capability for structural heat transfer and heat pipe performance. Two models have been generated and simulations performed with them to verify solution convergence for the coupled multi-app. DireWolf has great potential to be an integrated microreactor analysis tool. However, the tool is under significant development to validate the physics and reduce external dependencies for mesh and cross-section generation. Currently, the workflow for the use of DireWolf requires multiple specialists to generate the required inputs and meshes for a complete multiphysics analysis. Throughout this research, this has been a process that has a steep learning curve and needs extensive documentation. Development is in progress to validate the codes, streamline the workflow, and reduce external dependencies over the coming years. DireWolf advances multiphysics modeling, simulation, and analysis of microreactor designs by leveraging the internal coupling mechanisms for data transfer between physics packages in MOOSE and combining everything into a single executable. The use of independent external codes would require a similar number of input files and meshes with the added burden of manual data transfer to achieve even a loose coupling. Accomplishing a coupled multiscale neutronics, fuel performance, flow, and radiation damage simulation for a novel geometry is an incredibly complex task that will necessitate a lot of input in any scenario. DireWolf is a promising multiphysics code that is being developed to address these incredibly complex and challenging microreactor applications and is constantly being improved.

5. REFERENCES

1. Sabharwall, P., Lin, C.-S., Hansel, J., Laboure, V., Andrs, D., Hoffman, W., . . . Martineau, R. (2019). Application of Integrated Modeling and Simulation Capabilities for Full Scale Multiphysics Simulation of Microreactor Concepts. Idaho National Laboratory.
2. Kennedy, J. C., Sabharwall, P., Frick, K., McKellar, M., Bragg-Sitton, S., Rao, D. V., & McClure, P. (2019). Special Purpose Application Reactors: Systems Integration Decision Support. Idaho National Laboratory.

3. Martineau, R. C. (2019). DireWolf: MOOSE-based Multiphysics Simulator for Megawatt Scale Micro-Reactor. GAIN-EPRI-NEI-US NIC Micro-Reactor Workshop.
4. Gaston, D. R., Permann, C. J., Peterson, J. W., Slaughter, A. E., Andrs, D., Wang, Y., . . . Martineau, R. C. (2015). Physics-based multiscale coupling for full core nuclear reactor simulation. *Annals of Nuclear Energy*, 45-54.
5. Gaston, D., Newman, C., Hansen, G., & Lebrun-Grandie, D. (2009). MOOSE: A Parallel Computational Framework for Coupled Systems of Nonlinear Equations. *Nuclear Engineering and Design*, 239, 1768-1778.
6. Permann, C. J., Gaston, D. R., Andrs, D., Carlsen, R. W., Kong, F., Lindsay, A. D., . . . Martineau, R. C. (2020). MOOSE: Enabling Massively Parallel Multiphysics Simulation. *SoftwareX*, 11, 100430.
7. Martineau, R., Andrs, D., Carlsen, R., Gaston, D., Hansel, J., Kong, F., . . . Slaybaugh, R. (2019). Multiphysics for nuclear energy applications using a cohesive computational framework. 18th International Topical Meeting on Nuclear Reactor Thermal Hydraulics, NURETH 20, (pp. 3092-3106). Portland, OR.
8. Wang, Y., Schunert, S., Baker, B. A., & Laboure, V. (2020). Rattlesnake Theory Manual. Idaho National Laboratory.
9. Wang, Y., Schunert, S., Baker, B. A., & Laboure, V. (2020). Rattlesnake User Manual. Idaho National Laboratory.
10. DeHart, M., Gleicher, F. N., Laboure, V., Ortensi, J., Prince, Z., Schunert, S., & Wang, Y. (2019). MAMMOTH Theory Manual. Idaho National Laboratory.
11. Leppanen, J. (2015). Serpent a Continuous-energy Monte Carlo Reactor Physics Burnup Calculation Code. VTT Technical Research Centre of Finland.
12. DeHart, M., Gleicher, F. N., Laboure, V., Ortensi, J., Prince, Z., Schunert, S., & Wang, Y. (2020). MAMMOTH User Manual. Idaho National Laboratory.
13. Laboure, V., Ortensi, J., & Hummel, A. (2018). HTTR 3-D Cross-Section Generation with Serpent and MAMMOTH. Idaho National Laboratory.
14. Hales, J. D., Williamson, R. L., Novascone, S. R., Pastore, G., Spencer, B. W., Stafford, D. S., . . . Carlson, N. (2020). BISON Theory Manual: The Equations Behind Nuclear Fuel Analysis. Idaho National Laboratory.
15. Hales, J. D., Gamble, K. A., Spencer, B. W., Novascone, S. R., Pastore, G., Liu, W., . . . Carlson, N. (2020). BISON User's Manual. Idaho National Laboratory.
16. Hansel, J. E., Berry, R. A., Andrs, D., & Martineau, R. C. (2019). Sockeye Theory Manual. Idaho National Laboratory.
17. Franklin, A., Hansel, J., Andrs, D., Kunick, M., Berry, R., Johns, R., & Martineau, R. (2019). Initial Implementation of Discretized Flow Equations in Sockeye. Idaho National Laboratory.
18. Schumert, S., Sessim, M., & Schwen, D. (2018). Radiation Damage from Neutron Scattering.
19. Schumert, S., Schwen, D., & Vanwassenhova, D. J. (2019). Efficient Surrogate for Low Energy Damage Effects in Polyatomic Materials.
20. Sandia National Laboratories. (2020). CUBIT. Retrieved 01 02, 2020, from <https://cubit.sandia.gov/>
21. Sandia National Laboratories. (2019). CUBIT 15.1 User Documentation. Retrieved 01 02, 2020, from https://cubit.sandia.gov/public/15.1/help_manual/WebHelp/cubit_users_manual.html