ANL/NSE-22/56

Development of Integrated Thermal Fluids Modeling Capability for MSRs

Nuclear Science and Engineering Division

ANL/NSE-22/56

Development of Integrated Thermal Fluids Modeling Capability for MSRs

prepared by
R. Hu, J. Fang, D. Nunez
Nuclear Science and Engineering Division, Argonne National Laboratory

M. Tano, G. Giudicelli
Idaho National Laboratory

R. Salko
Oak Ridge National Laboratory

August 2022

ABSTRACT

The DOE Nuclear Energy Advanced Modeling and Simulation (NEAMS) program supports a full range of computational thermal fluids analysis capabilities and code developments for a broad class of light-water and advanced reactor concepts. The research and development approach under thermal fluids technical area synergistically combines three length and time scales in a hierarchal multi-scale approach. To demonstrate the feasibility and capabilities of a multi-scale thermal fluids capability using these codes, a key joint effort has been pursued to develop an integrated system- and engineering-scale thermal fluids analysis capability with the MOOSE-based codes, through integration of SAM and Pronghorn, both based on the MOOSE framework.

This report summarizes the progress in developing an integrated system- and engineering-scale thermal fluids analysis capability based on SAM and Pronghorn for molten salt reactors (MSRs), which gained significant interest in recently years. Two coupling approaches were studied, i.e. separate domain or domain-segregated coupling approach and the domain-overlapping approach. A series of coupled multi-physics models have been developed for a common reference molten salt fast reactor (MSFR) concept, ranging from standalone SAM system model to integrated SAM-Pronghorn-Griffin models. Both the steady state and the transient simulations are performed to the state-of-the-art simulation capabilities of NEAMS software suite in MSFR system applications.

This report also covers further development and testing of the gas transport model in SAM for MSR modeling support. The presence of noncondensable gas in MSR systems would have strong impacts on fission gas removal and transport of noble metals throughout the system. Fission products removed through the off-gas system can also impact reactivity and can act as an additional point of heat removal. To ensure that the gas transport model is adequately tested, the supported modeling capabilities and features of SAM were identified, and a suite of tests were developed to test the model for each identified feature. Validation and UQ testing were also performed for the model and demonstrated that the buoyancy term, which was originally developed using non-salt/helium gas experimental data, can capture the experimental gas bubble velocity and diameter to within experimental and code uncertainty. An existing MSRE model was also modified in this work to include the gas transport model and to demonstrate the gas model behavior for realistic conditions of interest.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

1 Introduction

1.1 Multiscale thermal fluid modeling

The design and safety analysis of advanced nuclear reactors, including molten salt reactors (MSR), often involves predicting complex thermal-fluid (T-F) phenomena that occur in various reactor components. Accurately and efficiently predicting such phenomena for long transients remains a key challenge for reactor safety analysis. System thermal hydraulics codes have been traditionally the main workhorse to support reactor design and licensing efforts. These fast-running system analysis tools can provide integrated modeling capabilities to simulating whole system behavior under simultaneous occurrence of various phenomena and processes. It can provide a wide range of analyses for iterative reactor design optimizations, prototypical conditions for fuels and materials qualification testing, safety analyses in reactor licensing, and can help identify key areas to designers, code developers, and the regulator for additional R&D work where the most progress is needed.

Higher-fidelity modeling and simulation tools, on the other hand, can take advantage of advances in computing power and modern algorithms for solving more complex systems of equations to enable multi-dimensional, closer-to-first-principles solutions. Computational Fluid Dynamics (CFD) methods have proven to be a powerful tool for nuclear applications to complement traditional systems analyses, with improved accuracy. These capabilities can also facilitate more detailed understanding of the phenomena and reducing the uncertainties in conventional system analysis tools.

The DOE NEAMS program supports a full range of computational thermal fluids analysis capabilities for a broad class of light-water and advanced reactor concepts. The research and development approach under thermal fluids technical area synergistically combines three length and time scales in a hierarchal multi-scale approach. These three overlapping thermal-fluid scales are defined across all reactor concepts as: [1]

- *Lower Length Scale*: The Lower Length Scale focuses upon resolving the high-resolution physics associated with single and multi-phase, highly turbulent conjugate heat transfer (CHT) with highly resolved thermal boundary layers.
- *Engineering Length scale*: The Engineering Length Scale includes coarse mesh approaches for homogenized multi-dimensional conjugate heat transfer (CHT), such as those found in gas-cooled or salt-cooled pebble-bed reactors, or subchannel capabilities tightly coupled nuclear fuels performance.
- *System Scale*: System Scale analysis for nuclear reactors is composed of one-dimensional fluid flow pipe networks and zero-dimensional system components, used for steady-state and transient T/F analysis at the full power plant level. These reduced order systems rely heavily on empirical correlations or models, as many of the flow features and phenomena are no longer resolved.

The tools and the overlapping scales supported by the thermal fluids technical area can be depicted in Figure 1-1. It is a preferred approach to group physics with similar multi-scale requirements into a common algorithm, developing separate software applications to address the

separate scales, and then coupling the applications where appropriate. The connections between different scales and codes [2] can be illustrated in Figure 1-2. At the lower length scale, Nek5000 will be used to investigate flow patterns and heat transfer for specific reactor components and operating conditions. Constitutive models such as turbulence models, heat transfer or friction correlations can be derived based on lower-length scale simulations and used in the engineering-scale or system-scale simulations. At the engineering length scale, Pronghorn can be used to model the reactor core or other key reactor components using a porous media or coarse-grid CFD approach. Finally, at the reactor plant length scale, SAM will be used to investigate the whole-plant behavior during steady-state operation and transients, which can be coupled with Nek5000 or Pronghorn when detailed simulations are needed for any specific components or subsystems.

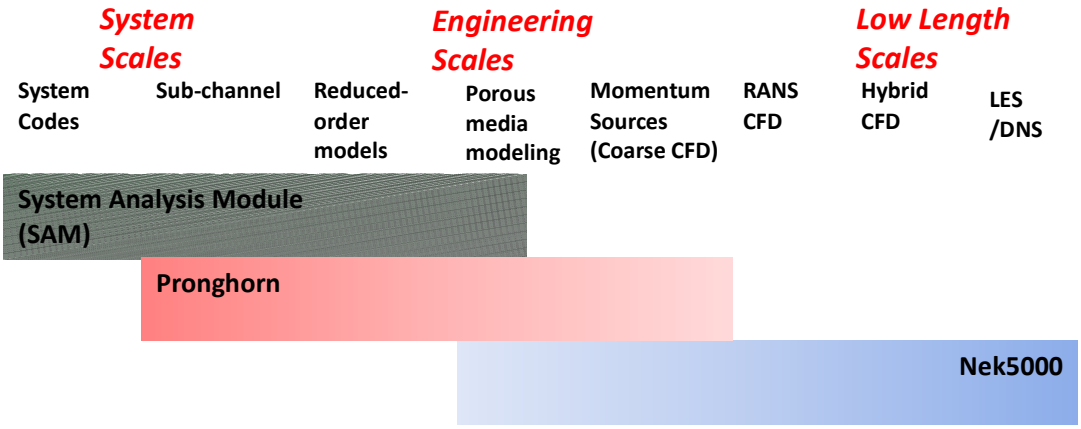

Figure 1-1. Multi-scale thermal fluids capabilities under NEAMS

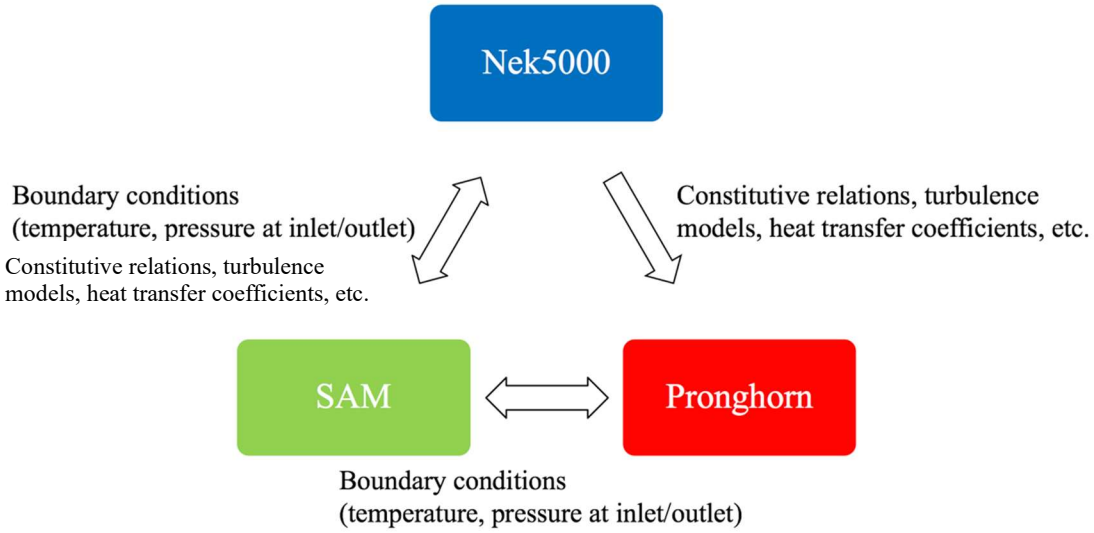

Figure 1-2. Expected integration of Nek5000, Pronghorn and SAM for advanced reactor applications.

The NEAMS thermal fluids technical area therefore embraces a multiscale approach that spans from system scales to micro-scales. Using the INL's Multi-physics Object Oriented Simulation Environment (MOOSE) framework, and its MultiApps and Transfers features in particular,

multiple space and time scales can be solved simultaneously to achieve a tightly coupled converged solution. Being able to couple these codes for multi-scale thermal-fluids modeling and simulation is viewed as a critical objective for reactor vendors in the design optimization process towards commercialization.

In a previous work [3], the integrated system- and engineering-scale thermal fluids analysis capability for advanced non-LWRs was developed using SAM-Pronghorn, both based on the MOOSE framework, and demonstrated for transient simulations of a reference FHR. The integrated capability covers the traditional system thermal hydraulics, subchannel, porous media, and coarse-grid CFD modeling approaches. Three coupling schemes were developed and tested using SAM and Pronghorn, including.

- A basic coupling scheme utilizing the native MOOSE MultiApp system and Picard iteration (with relaxation) methods;
- Advanced coupling schemes with acceleration techniques based on the Secant and Steffensen's methods;
- An integrated model with single-solve approach (developed in SAM).

In recently years, there is a renewed interest in MSR technology in the development of future advanced nuclear reactors. To support the development and utilization of the NEAMS codes for MSR safety analysis and licensing, continuous efforts have been devoted to enhancing integrated code capabilities and updating reference models for the MSR designs. A reference standard problem of a prototypical reactor design is foundational to nuclear industry and regulators to verify the adequacy of computer codes and evaluation models for a specific reactor type.

1.2 SAM Overview

SAM [4][5] is a modern system analysis tool under development at Argonne National Laboratory for advanced non-LWR reactor safety analysis. It aims to provide fast-running, modest-fidelity, whole-plant transient analyses capabilities, which are essential for fast turnaround design scoping and engineering analyses of advanced reactor concepts. While SAM is being developed as a system-level modeling and simulation tool, advanced modeling techniques being implemented include a reduced-order three-dimensional module [6], pseudo 3-D conjugate heat transfer modeling in reactor core [7], flexible and multi-scale modeling of heat transfer between fluid and structures [8], in addition to the advances in software environments and design, and numerical methods.

SAM aims to be a generic system-level safety analysis tool for advanced non-LWRs, including Liquid-Metal-cooled fast Reactors (LMR), Molten Salt Reactors (MSR), Fluoride-salt-cooled High-temperature Reactors (FHR), and High-Temperature Gas-cooled Reactors (HTGR). SAM takes advantage of advances in physical modeling, numerical methods, and software engineering to enhance its user experience and usability. It utilizes an object-oriented computational framework (MOOSE [9]), and its underlying meshing and finite-element library and linear and non-linear solvers, to leverage the modern advanced software environments and numerical methods. Currently, the SAM code provides the following major modeling features:

- One-D pipe networks represent general fluid systems such as the reactor coolant loops

- Flexible integration of fluid and solid components, able to model complex and generic engineering systems. A general liquid flow and solid structure interface model was developed for easier implementation of physics models in the components.
- A pseudo three-dimensional capability by physically coupling the 1-D or 2-D components in a 3-D layout. For example, the 3-D full-core heat-transfer in an SFR reactor core can be modeled. The heat generated in the fuel rod of one fuel assembly can be transferred to the coolant in the core channel, the duct wall, the inter-assembly gap, and then the adjacent fuel assemblies.
- Pool-type reactor specific features such as liquid volume level tracking, cover gas dynamics, heat transfer between 0-D pools, fluid heat conduction, etc. These are important features for accurate safety analyses of SFRs or other advanced reactor concepts.
- A computationally efficient multi-dimensional flow model is under development, mainly for thermal mixing and stratification phenomena in large enclosures and heat transfer in pebble bed for safety analysis.
- Point kinetics, reactivity feedback and decay heat modeling, including reactivity feedbacks due to various feedback mechanisms such as radial core and axial fuel thermal expansions.
- A general species transport capability has been implemented in SAM. The code can track any number of species carried by the fluid flow, mass exchange at the liquid-solid interfaces, and transport inside solids, for various applications.
- Special models for fluid solidification, structure expansion, or reactor component models.
- Plant control and trip system modeling. A set of proportional–integral–derivative (PID) controllers and trip logic units are implemented.
- An infrastructure for coupling with external codes has been developed and demonstrated. The code coupling with STAR-CCM+, SAS4A/SASSYS-1, Nek5000, PORTEUS, TRACE, and MOOSE-based Bison, Pronghorn, Griffin have been demonstrated.

SAM also utilizes the application- and validation-driven code development approach. These demonstration and validations lead up to the continuous assessment of the code capabilities and performance for a wide range of advanced reactor applications. Recent code demonstration activities cover simulations of a molten salt fast reactor, a molten-salt-cooled pebble-bed reactor, a gas-cooled micro reactor, and a stable salt reactor, etc. The demonstration simulations also resulted in reference plant models for these reactor types, which can be further utilized and tested by code users to examine the SAM code capabilities and identify capability gaps for these types of reactors. Recently completed and ongoing code validation activities include using test data from the Fast Flux Test Facility (FFTF), the High Temperature Test Facility (HTTF), the compact integral test facility (CIET), and the molten salt reactor experiment (MSRE).

1.3 Pronghorn Overview

Pronghorn is a multidimensional, coarse-mesh, thermal-hydraulics (TH) code for advanced reactors. It serves the intermediate fidelity realm situated between detailed computational fluid dynamics (CFD) analysis and lumped system models [10]. The goal of Pronghorn is to resolve multi-dimensional effects while maintaining fast execution times on desktop and laptop computers. This allows users to explore behavior during long-duration transients, perform sensitivity analysis, and explore design spaces more thoroughly.

Initially, Pronghorn relied on the Finite Element Method (FEM) discretization supporting both friction dominated and Euler-like formulations for porous compressible flow [11]. Pronghorn has successfully transitioned to the Finite Volume Method, recovering all capabilities developed with FEM [12]. The underlying numerical method is implemented in MOOSE's Navier Stokes module [13], while reactor specifics closures are implemented and maintained in Pronghorn itself. The MOOSE Navier-Stokes module implements an incompressible, a weakly-compressible and a fully compressible FVM formulation. For this work, the weakly-compressible FVM formulation is used. It uses the Boussinesq approximation, uses a collocated grid and Rhie-Chow interpolation of variables on element faces. The Navier-Stokes module implements both a free flow and porous version of the Navier-Stokes equations. In addition to solving the flow equations (mass, momentum, fluid energy), Pronghorn supports conjugate heat transfer with a solid phase collocated within fluid elements. Pronghorn also has native support for scalar quantity advection. This is leveraged for precursor advection in molten salt reactors.

Pronghorn is particularly well suited for modeling the core thermal-hydraulics of both gas and salt-cooled pebble bed reactors. In contrast to explicitly representing the geometry of pebbles, the porous medium capability is used to represent flow through pebble beds. This approach reduces the computational cost by many orders of magnitude with the caveat that accurate closure correlations must be available to retain solution accuracy. Pronghorn implements many published closures for pressure drops, heat transfer coefficients, and effective conductivities of the pebble bed compact [14].

Pronghorn is regularly built and tested with the latest version of MOOSE to ensure its interoperability with the MOOSE ecosystem. It follows INL Software Development Plan 4005 [15] and the software development process is currently streamlined to become compliant with NQA-1 standards. Pronghorn has been coupled with the Griffin [16] neutronics code for modeling gas and salt-cooled pebble bed reactors and pool-type molten salt reactors.

This report summarizes the progress made in FY22 in developing an integrated system- and engineering-scale thermal fluids analysis capability based on SAM and Pronghorn for MSRs. Chapter 2 discussed two coupling approaches used in this work, i.e. separate domain or domain-segregated coupling approach and the domain-overlapping approach. Developmental activities to support the SAM-Pronghorn coupling are described in Chapter 3. In Chapter 4, a series of coupled multi-physics models were developed for a reference molten salt fast reactor (MSFR) concept, ranging from standalone SAM system model to integrated SAM-Pronghorn-Griffin models. Both the steady state and the transient simulations were performed to demonstrate the state-of-the-art simulation capabilities for MSFR system applications.

Additionally, Chapter 5 covers further development and testing of the gas transport model in SAM for MSR modeling support. To ensure that the gas transport model is adequately tested, the supported modeling capabilities and features of SAM were identified, and a suite of tests were developed to test the model for each identified feature. Validation and UQ testing were also performed for the model. It was also applied to the MSRE model to demonstrate the gas transport model for more relevant reactor conditions.

2 SAM-Pronghorn Coupling Schemes

For nuclear reactor system simulations, it is desirable to combine a 3D (or 2D-RZ) model of the core region with a 1D model of the heat transport loops. This multi-scale approach is used to balance computational costs and benefits. The 3D representation of the core can more closely examine potential limiting factors in the core, such as the fuel peak temperature, than a 1D modeling approach of the core. Combined with multi-dimensional neutron kinetics, it enables higher fidelity simulation of power transients. A 1D approach would require high conservatism for estimating failure limits. On the other hand, a 1D model for the rest of the heat transfer loops can coarsely and cheaply model these large regions. While these regions also play a large role in the safety analysis of nuclear reactors, a detailed local multidimensional approach would only be beneficial for a narrow selection of accidental transients.

We explore in this section two coupling approaches for integrated system and engineering nuclear reactor simulations using SAM and Pronghorn respectively. The first approach is the exchange of fluid integral quantities at the domain boundaries, also called domain-segregated or domain decomposition coupling approach. The second is the domain-overlapping approach. In this approach, both SAM and Pronghorn model the entire primary loop and exchange information to remain synchronized.

2.1 Fluid domain boundary coupling or domain-segregated approach

The boundary coupling or domain-segregated approach has been applied in many applications of coupling system codes with CFD codes in nuclear applications, such as the coupling between SAS4A/SASSYS-1 and STAR-CD [17] or STAR-CCM+ [18], RELAP5 and CFX [19], and TRACE and CFX [20]. In all applications, careful control of data exchange flow and the time-synchronization are essential for a numerically stable, and physically valid simulation [21]. Different coupling schemes were used, partially due to the different features and limits within the specific codes.

The major difficulty in this approach is the coupling between the two thermal hydraulics solves. In order to couple the core model to the 1D loops, we exchange pressure, velocity, and temperature at both the inlet and outlet of the core. The 3D core solver receives the core inlet velocity and temperature and the core outlet pressure from the 1D solver, while the 1D solver receives the core outlet velocities and the temperature and core inlet pressure from the 3D core solver. Those quantities are coupled, in that the quantities computed at the core outlet strongly depend on the core inlet, and similarly, the core inlet temperature and velocity/mass flow rate also depend on the core outlet conditions. Because of this coupling, we must iterate between the two solves to converge the core inlet and outlet conditions. This coupling can be quite difficult, as the exchange of pressure boundary conditions causes pressure waves propagating through each domain. Additionally, changes in pressure boundary conditions may cause changes in material properties for compressible flow and prevent the rapid generation of a new numerical solution from the previous iteration solution.

Both SAM and Pronghorn codes are based on the MOOSE framework, which is equipped with a native capability for multiphysics coupling through the MultiApp system. A basic coupling scheme is directly available through exchange of boundary conditions, e.g. pressure, velocity, and temperature, of both codes utilizing the MOOSE MultiApp system, as shown in Figure 2-1. The thermal-hydraulics system is split into a 3D domain (as will be solved by the Pronghorn code) and

1D domain (as will be solved by the SAM code). Exchange of boundary conditions, i.e. velocity and temperature at domain inlet and pressure at domain outlet, are achieved through various kind of postprocessors, which are executed at the end of each time step and transferred to different solvers through the MultiApp system. Boundary conditions exchange are executed in both directions.

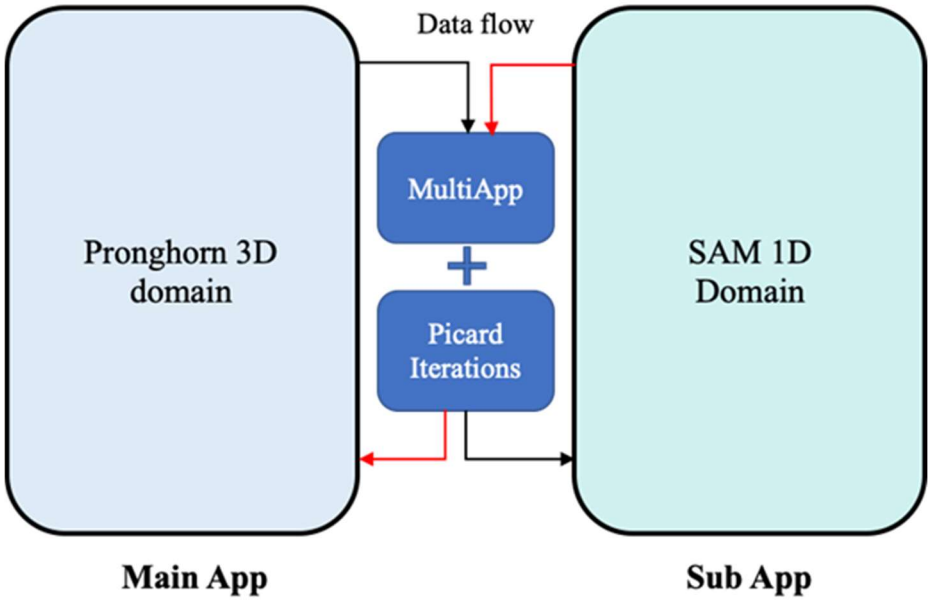

Figure 2-1. Sketch of basic Pronghorn-SAM coupling scheme using MOOSE MultiApp and Picard iterations

The basic boundary coupling scheme utilized for the SAM-Pronghorn integration involves the use of *CoupledPPSTDV* and *CoupledPPSTDJ* SAM components, designed to facilitate the transfer of boundary information to and from other MOOSE applications in MultiApp simulations. The *CoupledPPSTDJ* component is utilized to transfer velocity and temperature information, while *CoupledPPSTDV* transfers pressure and temperature at a given interface. The data transfer to and from the sub-application is carried out by the Transfers System in MOOSE.

The reader is referred to Ref. [3] for the implementation and verification of this approach for a closed FLiBe loop, shown in Figure 2-2, and for a Pebble-Bed Fluoride Salt-cooled High Temperature reactor (FHR). The results from this report may be summarized as follows:

- The simple exchange of boundary conditions without acceleration was severely plagued with numerical issues, despite initialization of both SAM and Pronghorn simulations as standalone simulations.
- Numerical relaxation of boundary conditions, referred here as relaxed Picard fixed point iterations, enabled convergence of the coupling, at the cost of numerous fixed-point iterations, up to 70 at initialization and 30 at the onset of a pump transient.
- Implementation of convergence acceleration techniques, namely replacing relaxed Picard with quasi-Newton (Secant and Steffensen's) methods demonstrated a 3x-6x reduction in the number of fixed-point iterations necessary to obtain convergence on any given timestep.

Later efforts undertaken under the scope of the NEAMS-NRIC Virtual Test Bed achieved numerical convergence of the coupling without relaxation nor acceleration [22]. These also showed that the relaxation to steady state may be achieved with fewer fixed-point iterations in every timestep.

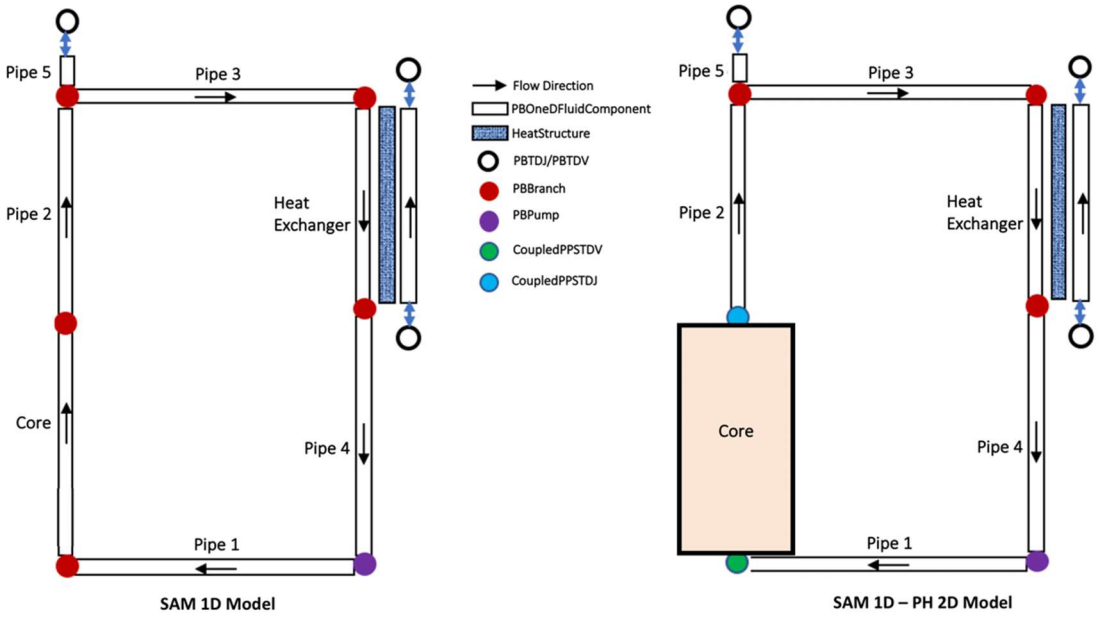

Figure 2-2. SAM 1D and SAM 1D – PH 2D model for the closed FLiBe loop problem, used for the verification of the domain-segregated approach

2.2 Domain overlapping coupling approach

Owe to the numerical challenges associated with the pressure-velocity coupling boundary conditions, a domain overlapping approach for thermal coupling was also studied in this work. It leverages a previous work of the coupled 2D-RZ MSFR primary loop model [23] using Pronghorn and Griffin, also as part of the NEAMS-NRIC Virtual Test Bed exploratory modeling activities in FY21 [24].

This approach is presented in Figure 2-3. In this one, the coupled Griffin-Pronghorn and the SAM models solve simultaneously the fuel (primary) loop. The Griffin-Pronghorn model more accurately more computes the time-resolved power during the transient and pass it to SAM at each time step. Hence, the complete SAM model does not need to rely on point kinetics solves and gets a more accurate power coefficients and reactivity feedback. At the same time, SAM computes the cold-leg, secondary-side temperature of the primary-to-intermediate heat exchanger and pass it to the Griffin-Pronghorn model. Hence, the Griffin-Pronghorn model uses a more realistic cold leg temperature at the primary-to-intermediate heat exchanger.

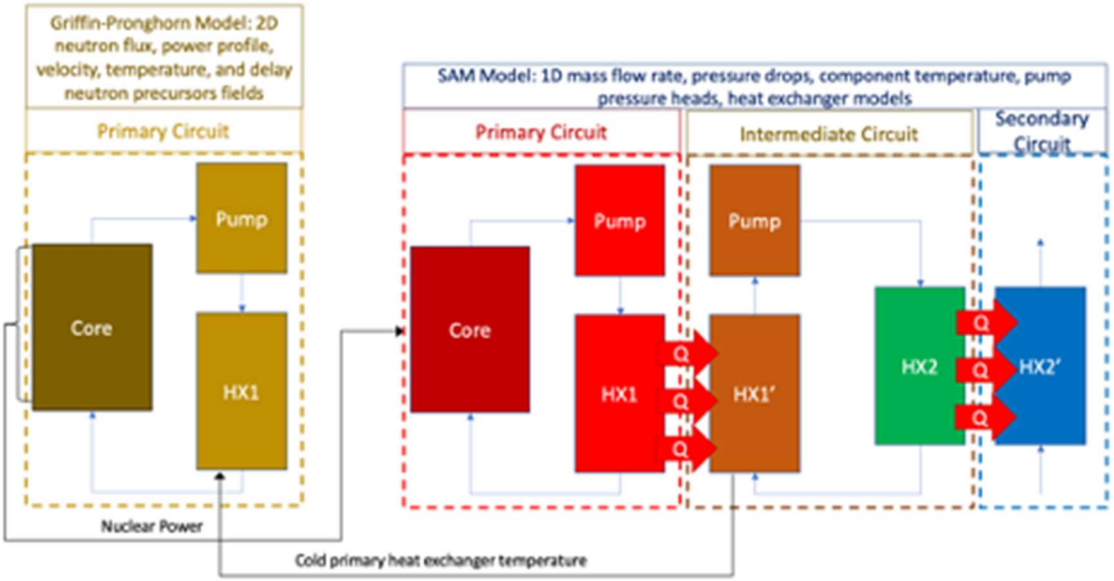

Figure 2-3. Overview of the domain-overlapping thermal coupling approach between Griffin-Pronghorn and SAM [25]

The key advantage of this model is that realistic temperatures are obtained in the SAM fuel loop as no coupling approximations need to be made. The better resolved temperature profiles yield a better prediction of the thermophysical properties in the fuel salt at different points in the loop. Therefore, the pump and heat exchanger performance in the SAM primary loop can be better resolved. Moreover, being able to model the full fuel loop in the Griffin-Pronghorn is beneficial since the model is less dependent on inlet-outlet flow boundary conditions into the liquid fueled core. The disadvantage of this model is that the fuel loop is resolved twice, with Griffin-Pronghorn and SAM simultaneously. However, the SAM model of the core involves only a few components and its computational cost per time step is almost negligible (by a factor of ~300) when compared to the one of Griffin-Pronghorn.

It should be noted that this domain overlapping approach differs from the domain overlapping approach in the literature for system code and CFD code coupling, such as TRACE and STAR-CCM+ [26] or SAM and NekRS [27]. The key aspect of the overlapping-domain coupling is that for a given portion of the system that is modeled by both codes, i.e. overlapped portion, their momentum balances should match. This was achieved by modeling the friction factor in the system code such that the momentum balances of the two codes are equal. Similar to the separate-domain approach discussed in the previous section, the overlapping domain strategy also transfers the inlet velocity from SAM to the inlet of the NekRS portion. However, different from the separate-domain coupling, the overlapping-domain strategy neither transfers pressure data nor the outlet flow rate from the outlet of the CFD model. Rather, it transfers the friction factor computed numerically based on CFD results.

It should be noted that the domain overlapping approaches are being pursued for SAM and NekRS coupling under an NEUP project supported by NEAMS program. Great successes have been demonstrated in several verification and validation test cases. This approach can be pursued as another option for the integrated SAM-Pronghorn capabilities in the future.

3 Development activities to support SAM-Pronghorn 1D-3D coupling

Code updates were needed in the Navier Stokes finite volume module to support the 1D-3D coupling simulations. These developments were meant to both facilitate the creation of coupled input files and enable 3D simulations.

3.1 Extensive and flexible fluid flow boundary conditions

For the domain-segregated or non-overlapping domain decomposition approach to 1D-3D coupling, the two codes exchange pressure, velocity, temperature and scalar concentration information on the boundaries. In SAM, all the boundary quantities are input as postprocessors in the *CoupledPP* class of objects. They form Dirichlet and Neumann boundary conditions depending on the quantities. In Pronghorn, the core inlet information is also received as postprocessors. They may then be processed into either Dirichlet or Neumann boundary conditions. Both options were implemented early in the project in order to maximize flexibility. Both options are implemented in the Action syntax to short input files.

3.1.1 Dirichlet boundary conditions

Dirichlet boundary conditions are easy to set up and native capabilities were already implemented in the MOOSE framework for finite volume discretizations. With those, the user had to perform additional computations to make sure that:

- Boundary velocity values conserved the mass flow rate, using the boundary density and area;
- Boundary temperature values conserved the energy flow rate, using the boundary specific heat and area;
- Boundary scalar concentration values conserved the flow of scalar precursors, using the boundary density and area.

The additional computations would depend on what quantities were transferred between the two codes, as both flow rates and boundary values could be computed in SAM. To simplify this process, all these computations were automated in two objects: *WCNSFVInletVelocityBC* and *WCNSFVInletTemperatureBC*. These two classes support weakly compressible fluid densities as the *WCNSFV* prefix mentions. They support either direct value input or flux/flow rate value input, converted to a value using local boundary information.

A second concern was that Dirichlet boundary conditions were imposed weakly and were subjected to the face interpolation scheme (e.g. upwinding), and thus were only exact in the limit of infinite discretization near the boundary. This was modified as part of other efforts during this project to use the value directly with no regards for the face interpolation scheme.

3.1.2 Neumann or flux boundary conditions

Flux boundary conditions are similarly easy to set up and a general Neumann boundary condition is implemented in the MOOSE framework for finite volume discretizations. The user had to similarly ensure that:

- The integrated mass flux on boundary cells conserved the inlet mass flow rate;
- The integrated momentum flux on boundary cells conserved the inlet momentum flow rate;

- The integrated energy flux on boundary cells conserved the inlet energy flow rate;
- The integrated scalar flux on boundary cells conserved the inlet scalar flow rate.

The additional computations to make sure conservation was achieved would depend on the quantities transferred from SAM, either the flow rates directly or the boundary values. Either are supported in the four new boundary condition objects that implement these computations:

- *WCNSFVMassFluxBC* for the mass conservation equation
- *WCNSFVMomentumFluxBC* for the momentum conservation equation
- *WCNSFVEnergyFluxBC* for the energy conservation equation
- *WCNSFVScalarFluxBC* for each delayed neutron precursor conservation equation

These classes support weakly compressible fluid densities as the *WCNSFV* prefix mentions.

3.2 Side-integration postprocessing of MOOSE functors

MOOSE functors are a new concept introduced for finite volume discretizations. All variables of any type are functors, as are functions and functor material properties. Finite volume discretizations necessitate the evaluation of quantities on cell faces to compute fluxes between two volumes/cells. The sum of the fluxes and the volumetric terms form the balance or conservation equations to solve. The fluxes are summed in the residual of the equation of interest on each cell. While this mechanism is now well established to compute residuals and Jacobians in the finite volume discretization, similar techniques also have to be employed to perform postprocessing. The loops over the mesh to compute side integrals over a boundary of interest are elemental loops, and in finite element discretizations the integrals on element sides (belonging to said boundary) are computed with an inner loop over the quadrature points on those sides.

This approach was not generalized to finite volume. The finite volume shape functions are constant over each element, and their only quadrature point is chosen on the centroid of the element. But there may be multiple faces of the element that are on the boundary of interest. For example, if the neighbor 3D cell on the other side of the boundary of interest has been uniformly refined, there is likely 4 faces that belong to the boundary of interest.

Instead, side integration of functors is performed by a loop over faces within the loop over elements in the mesh. This new API involves the same concept as the previous one, a virtual function to override, *computeFaceInfoIntegral*() instead of *computeQpIntegral*(). The postprocessor objects created by the user will call the new function to tally the integrals when dealing with functors, and will call the old API when dealing with finite element variables or regular material properties. The two objects which implement the new API are:

- *SideFunctorIntegral*(), to integrate any functor on a sideset. It is part of the framework
- *VolumetricFlowRate()* to compute the advective flux of a quantity in fluid flow simulations. It is part of the Navier Stokes module.

3.3 Numerical segregation of precursor solves

The delayed neutron precursors are created by fission in the fuel salt and then advected with the salt through the primary loop. This is a unique characteristic of molten salt reactors. To compute the concentration of delayed neutron precursors, one must solve both the neutronics and fluid flow equations. The fission source distribution directly impacts the neutron precursors distributions as it acts as the source term in those equations. In return, the neutron precursors distributions modify the fission source and thus the solution of the neutron transport equation. This coupling must be modelled but converges rather quickly, and a fixed-point iteration is appropriate to converge it.

The fluid flow velocities advect the precursors and thus directly impact their distributions in the core. However, the precursors only affect the fluid flow velocities through the power distribution. There is no modification of the fluid properties due to the concentration of precursors. This indirect coupling means that effectively there is only a one-way coupling and thus no advantages to solving the fluid flow and the precursors equations together. In fact, solving the equations all together brings the number of unknowns from 4 in the flow equations to 10 in total, which comes at a high memory and computation cost.

Segregating the precursor solves involved two main developments in this FY: enabling the shortened input file syntax and siblings transfer. The latter was funded under another NEAMS scope [28] for a different reactor application initially but deployed here simultaneously.

The short input file syntax or action syntax was initially conceived to solve at least the fluid flow equations, and then if requested the fluid energy equation and the scalar advection equation. This was reworked to allow solving only the mass and momentum equation, or only the fluid energy equation, or only the scalar advection equation. This change necessitated enabling the input of auxiliary variables for the velocities, and pressure for Rhie Chow interpolation, and many adjustments in the parameter checks.

Another feature greatly enabling segregating precursor solves was the implementation of siblings transfers. Siblings transfers are transfers between child applications of a primary application. They avoid passing all quantities through the main application which adds significant memory overhead. This is especially useful for passing velocity (and pressure for Rhie Chow interpolation) fields directly from the fluid flow application to the application solving the precursor equation without saving them on the neutronics main application.

Figure 3-1 shows the coupling scheme enabled on the left, and the difficulty in achieving segregated solves of precursors on the right with two tentative schemes.

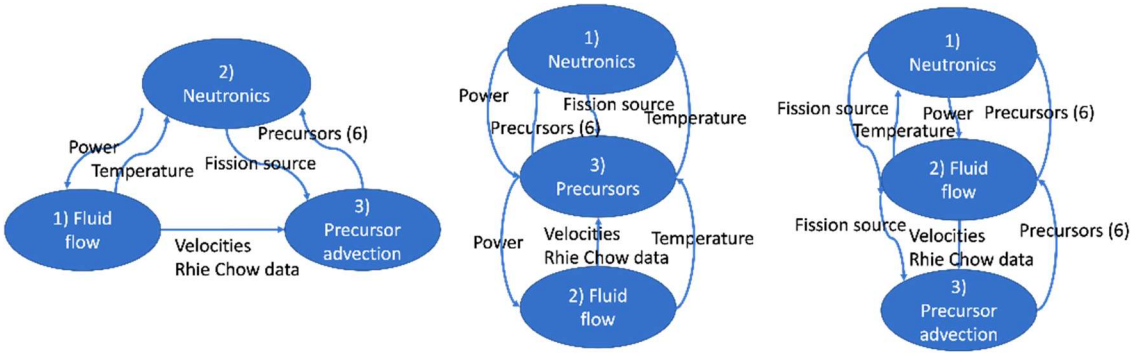

Figure 3-1. MultiApps setup and Transfers for segregated precursors solve [28].

The solve time on a fully spec-ed 2020 Macbook Pro laptop is reduced from 1,200s to 700s for a relaxation to steady state multiphysics transient by segregating the precursors equations from the fluid flow equations. These improvements to the solve are not leveraged by the 1D-2D coupling studies shown in the next chapter. It is expected that siblings transfers of postprocessors, also developed under the mentioned scope, would be leveraged to propagate the boundary conditions to the precursors equations.

4 Demonstration of Integrated SAM-Pronghorn-Griffin Simulation

A series of coupled-physics models have been developed for a common reference MSFR concept, ranging from standalone SAM system model to integrated SAM-Pronghorn-Griffin models. Both the steady state and the transient conditions involving pump head change are modeled. This Chapter discusses the case setups and the corresponding results from various modeling options to showcase the state-of-the-art simulation capabilities of NEAMS software suite in MSFR applications.

4.1 Reference MSFR Design Overview

The specific reactor model considered here is based upon the MSFR design developed under the Euratom EVOL project [23][29]. The reference MSFR is a 3000 MW fast-spectrum reactor with three different circuits: the fuel circuit, the intermediate circuit, and the power conversion circuit. The MSFR is a fast breeder reactor with large negative temperature and void reactivity coefficients, offering unique safety characteristics not found in solid-fuel fast reactors. The core has a height of 1.6 *m* along the centerline, and a height of 2.65 *m* in the peripheral region. The reactor radius ranges from 1.05 to 1.53 *m*. The peripheral wall is a curved surface resulting in the entire core resembling the shape of a stout hourglass. In the current study, the MSFR core region is to be modeled as both a 1-D component in standalone SAM system model and a 2-D axisymmetric domain in SAM-Pronghorn-Griffin simulation. In the fuel circuit, there are 16 sets of pumps and heat exchangers around the core, and the specific fuel salt considered is LiF-ThF4 (0.78-0.22). The corresponding thermophysical properties are listed in Table 4-1, based on which dedicated Equations of States (EOS) have been created in SAM input file to model the fuel salt. The total mass flow rate in the fuel circuit is 18,000 *kg/s*.

Table 4-1. Thermophysical properties of the fuel salt in primary circuit.

		Unit	*LiF-ThF$_4$ (0.78-0.22)*
Density	ρ	kg/m^3	$4983.56 - 0.882 \cdot T$
Dynamic viscosity	μ	mPa·s	$\rho(5.54 \times 10^{-5} \exp(3689/T))$
Thermal conductivity	k	W/(m·K)	$0.928 + 8.397 \times 10^{-5} \cdot T$
Specific heat capacity	c_p	J/(kg·K)	$-1111 + 2.78 \cdot T$

The intermediate circuit is integrated to the primary circuit through the primary heat exchanger. The heat exchanger modelled adopts a shell-and-tube design, which has a height of 2.4 *m*. The heat exchanger parameters, such as the flow areas, hydraulic diameters, heat transfer area density, are tailored to meet the specific heat removal rate (i.e., 3000 *MW* in the current work). The coolant salt selected for the intermediate circuit is LiF-BeF2 (0.66-0.34), of which the related thermophysical properties are provided in Table 4-2. A set of EOS was generated to the model the coolant salt. The heat exchanger is made of Hastelloy N alloy with the thermophysical properties listed in Table 4-3. The intermediate heat exchanger has a height of 3.2 *m*. Meanwhile, the energy conversion circuit is Helium based Joule-Brayton cycle, and the MOOSE built-in Helium EOS is used.

Table 4-2. Thermophysical properties of the coolant salt in intermediate circuit.

		Unit	*LiF-BeF_2 (0.66-0.34)*
Density	ρ	kg/m^3	$2146.3 - 0.4884 \cdot T$
Dynamic viscosity	μ	mPa·s	$0.116\exp(3755/T)$
Thermal conductivity	k	W/(m·K)	1.1
Specific heat capacity	c_p	J/(kg·K)	2390

Table 4-3. Thermophysical properties of Hastelloy® N alloy.

		Unit	*Hastelloy® N alloy*
Density	ρ	kg/m^3	8860
Thermal conductivity	k	W/(m·K)	23.6
Specific heat	c_p	J/(kg·K)	578

4.2 1D SAM Model

Based upon the design specifications of EVOL MSFR, a representative 1-D system model was established for the MSFR concept covering both the fuel/primary and intermediate circuits. The ex-core loops in the primary and intermediate circuits are lumped together, and only one loop is considered in both circuits. The energy conversion system is modeled as the boundary condition to the secondary/cold side of intermediate heat exchanger. The helium enters the secondary side of intermediate heat exchanger at a temperature of 673.15 *K* and at an inflow velocity of 66.65 *m/s*, and at a corresponding pressure of 75 *bar*.

A point kinetic equation (PKE) model is employed to model the neutronics of MSFR system in the standalone SAM models. Six delayed neutron precursor groups are considered. The corresponding decay constants and fractions are listed in Table 4-4 based on Griffin calculations. The total reactivity feedback considered is -4.664 *pcm/K*.

Table 4-4. Delayed neutron precursor groups used in SAM PKE modeling of MSFR.

DNP Group i	β_i	λ_i (s^{-1})
1	8.42817E-05	1.33104E-02
2	6.84616E-04	3.05427E-02
3	4.79796E-04	1.15179E-01
4	1.03883E-03	3.01152E-01
5	5.49185E-04	8.79376E-01
6	1.84087E-04	2.91303E+00

The 1-D MSFR system model is simulated until reaching a steady state, and the steady-state temperature distribution is illustrated in Figure 4-1. As expected, the fuel salt is heated inside the core, and pass on the heat to the coolant in intermediate circuit through the primary heat exchanger. The heat is then transferred to the energy conversion circuit via the intermediate heat exchanger. The mass flow rates are 18,086 *kg/s*, 16,847 *kg/s* and 2,571 *kg/s* for the three circuits, respectively. The core inlet temperature is 951.68 *K* and the outlet temperature is 1050.92 *K*. A fuel temperature rise of 99.23 *K* is obtained along the core, which matches well with the design specification (~ 100 *K*). The temperature rise of the secondary-side coolant salt is about 74.25 *K* after flowing through

the primary heat exchanger, while the temperature rise of helium flow is about 224.52 *K* after flowing out of the intermediate heat exchanger.

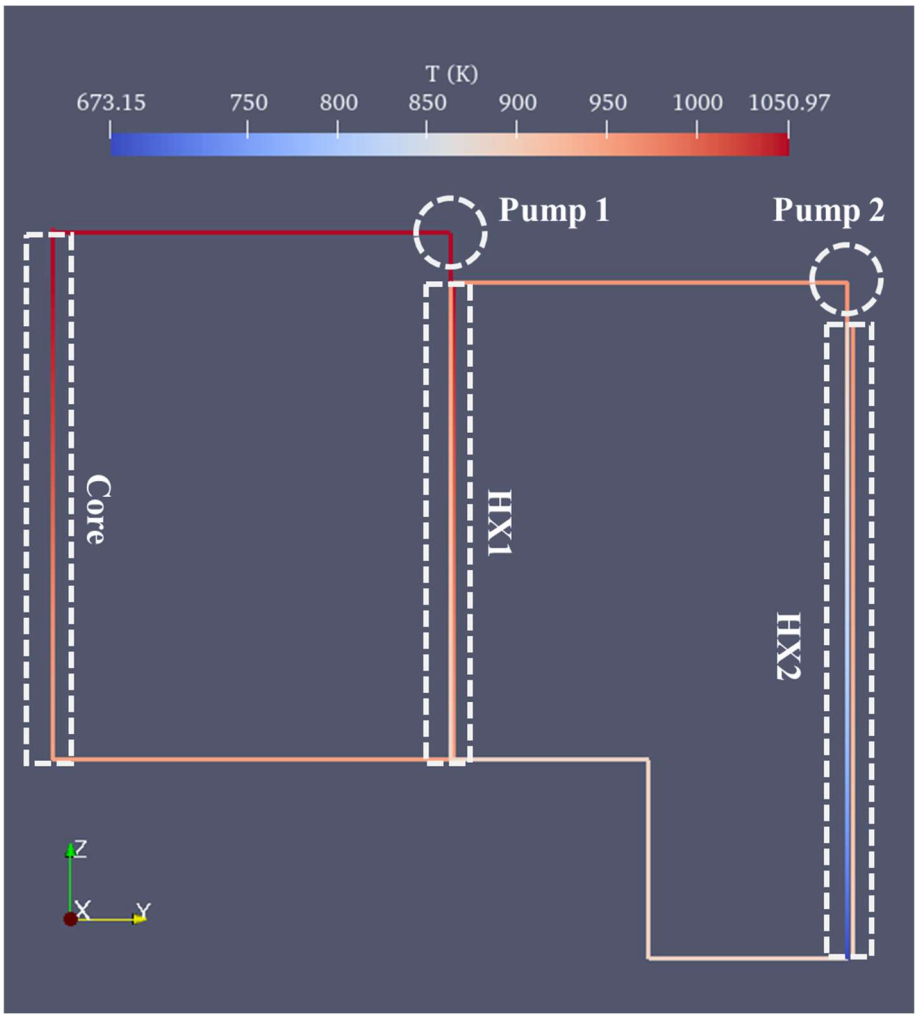

Figure 4-1. Steady-state temperature distribution in 1-D MSFR model.

4.3 2D R-Z Pronghorn-Griffin Model

Leveraging the multi-dimensional modeling capability of Pronghorn and Griffin, a coupled 2D-RZ MSFR core model was developed. It was also part of the NEAMS-NRIC Virtual Test Bed exploratory modeling activities in FY21 [23]. The core model extends past the core to include a homogenized primary loop, which is illustrated in Figure 4-2. The model was restricted to a 2D-RZ configuration to achieve a fast simulation runtime. It was developed in parallel with and leveraged early finite volume capabilities of MOOSE. The related study was one of the first successful couplings of these capabilities with steady-state as well as transient neutronics in Griffin [31]. We reutilized this model with permission in this work.

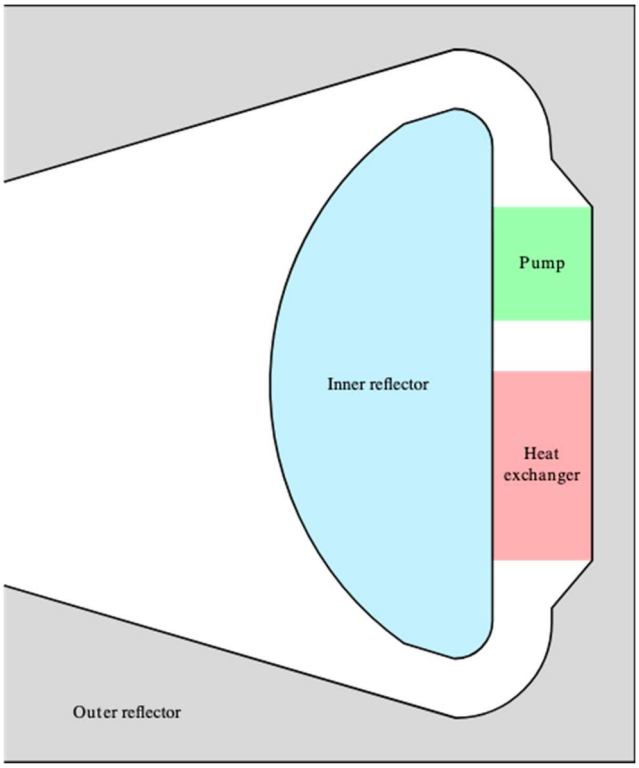

Figure 4-2. Geometry of the 2D-RZ MSFR model

4.3.1 Pronghorn coarse mesh CFD model

The Pronghorn model only covers the fluid domain. The inner and outer reflector regions are deleted from the mesh at the beginning of the simulation. The axis of symmetry of the domain is on the left over the center of the core. The white region in Figure 4-2 is the main flow domain, where power is generated. The salt flows upwards in the center of that region, then enters the loop on the right. The loop contains a homogenized pump and a homogenized heat exchanger. This loop is homogenized to represent the 16 primary loops of the MSFR concept; however, it is not approximated as a porous medium.

The homogenized pump is a volumetric momentum source which accelerates the flow downwards in the homogenized loop. The magnitude of the source was chosen to obtain the desired mass flow rate of 18,000 *kg/s* or an equivalent temperature rise across the core of approximately 100 *K*. The pump head may be controlled using the MOOSE control system to perform pump-related transient. The homogenized heat exchanger is a volumetric heat sink with a convection term proportional to the difference between the local fluid temperature and the heat exchanger cold-leg temperature. The coefficient of proportionality, the volumetric heat exchange coefficient, is chosen to match the heat exchange coefficient in previous studies [32].

The mesh was generated with Cubit. It originally consists of 2,068 QUAD4 elements, and the element count was further increased to over 8,000 after one uniform refinement. It is shown in Figure 4-3. The mesh is well aligned with the flow in the heat-exchanger loop but less so in the main core region. It is unstructured over the core cavity but strictly rectangular in the ex-core region. It was refined closer to the wall, in initial attempts to control the $y+$ near the wall.

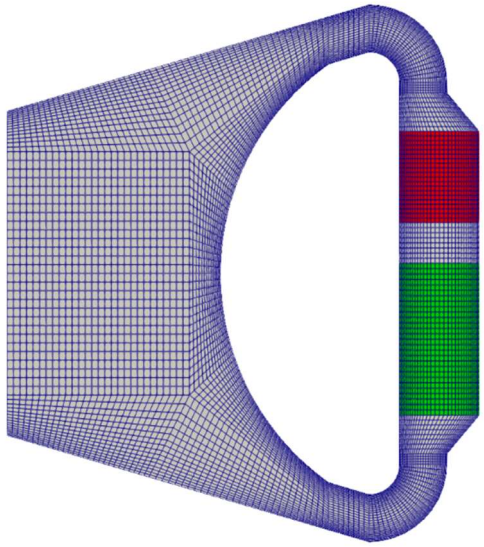

Figure 4-3. Mesh for fluid simulations of the MSFR [23]

Pronghorn solves for the fluid velocities (two components in 2D), the pressure field, the fluid temperature field, and six delayed neutron precursors fields. Decay heat precursor advection is not considered in this model.

The input file leverages the NavierStokesFV action. As such it defines all variables, all equations, all initial conditions, and all boundary conditions in approximately 200 lines of inputs [33]. The simulation is initialized using an isothermal flow simulation lasting for approximating 100s of simulation time. The initial condition is saved in an Exodus solution file and read at the beginning of the coupled multiphysics relaxation-to-steady-state simulation. This considerably accelerates the modeler's workflow, and a similar approach is taken for the transient simulations, which restart from a converged steady-state solutions.

With a Reynolds number close to 1 million in the core; hence, the flow turbulence has to be modeled in the MSFR. The support of turbulence modeling is nascent in Pronghorn. The only model supported currently is a 0-D mixing length turbulent model. While development is underway for a coarse mesh ***k-ε*** model [34], it is not used in this work. The mixing length is capped in most of the domain [31], but unphysical large salt temperatures in excess of 1,300K are still present in the flow domain. The latest turbulence models in development produces more realistic temperatures in these regions [34] so this shortcoming is expected to be resolved within FY23.

4.3.2 Griffin neutronics model

The neutronics model leverages a similar mesh as the fluid flow region, with the addition of the outer and inner reflector regions. The neutron group cross sections are computed by Serpent on an approximate reactor geometry with constant salt and reflector temperature and composition. The group cross sections are tabulated based on temperature from 300K to 1,800K, with 300K intervals.

An eigenvalue criticality calculation determines the steady-state neutronics solution. It may be run uncoupled from the thermal-fluids simulation. It uses the 6-group, CFEM-diffusion solver in Griffin. The delayed neutron precursors are input externally as the delayed neutron source-advection-reaction equation is solved by Pronghorn.

The transient simulation restarts from the steady-state simulation. The fluxes are normalized by the k-effective to ensure the null-transient can be held. The transient solver uses spatial kinetics with a first-order Euler integration. Future refinements may improve on this integration scheme. However, for the present work, we deemed good enough to control accuracy by reducing the time step of the numeric integration.

4.3.3 Pronghorn-Griffin coupling

The coupled Pronghorn-Griffin simulation is achieved by leveraging the MultiApps and Transfers systems in MOOSE. The thermal fluids simulation is placed a child application of the neutronics simulation, and several quantities are transferred after each physics is solved:

- The power density is passed from Griffin to Pronghorn. It serves a heat source and influences the fluid temperature. The buoyancy of the fluid then affects the flow distribution
- The fission source is passed from Griffin to Pronghorn. It serves as a source term in the advection-reaction equations solved by Pronghorn.
- The fluid temperature is passed back from Pronghorn to Griffin. It influences both the density and the Doppler effects in the group cross sections of the salt.
- The delayed neutron precursor concentrations are passed from Pronghorn to Griffin. They influence the neutron source, are summed with the fast neutron source, and therefore modify the neutron flux distributions.

The coupling is iterated to convergence using a Picard fixed point iteration. Only 5 iterations are necessary to achieve convergence on average during the transients.

4.4 Coupled SAM-Pronghorn-Griffin Model and Steady-State Results

4.4.1 Separate domain coupled model execution

The separate domain SAM-Pronghorn-Griffin coupled simulations were executed through a multi-step approach. The two-step approach to obtain steady state solutions is outlined below. In these simulations Griffin is set up as the Parent Application of Pronghorn, and Pronghorn as the Parent application of SAM.

Step 1: Decoupled, steady state solution

In order to avoid divergent solutions in the first time steps of the coupled simulations, the velocity, temperature, pressure, and precursors at the coupling interfaces need to be consistent between SAM and Pronghorn. To ensure consistency, the SAM, and Pronghorn models are first executed independently. The independent execution is carried out by enabling the MOOSE MultiApps system, but without enabling the Transfer System. This setup allows for both simulations to run in parallel without transferring any information at the coupling interfaces. The steady state solution from this step, is then utilized as the initial condition for the coupled simulation in Step 2.

Step 2: Steady state solution with eigenvalue solver

This step is an intermediate step to ensure that a steady state is achieved with the SAM-Pronghorn-Griffin coupled simulation. First, the system is initialized assuming a constant core power. With this initialization, a k-eigenvalue problem is solved in Griffin coupled with SAM and Pronghorn. The simulation is initialized with the solution from Step 1. Here, communication between the SAM-Pronghorn-Griffin is allowed by enabling the MultiApp's Transfer system. The boundary conditions are obtained through multiple post-processors and are exchanged at the end of every time step between the three codes. The exchange of boundary conditions between SAM and Pronghorn occurs in both directions, thus maintaining stability of the coupled system. Furthermore, to ensure the desired convergence in both domains, the Secant acceleration methos was enabled. Note that a MultiApp of type FullSolveMultiApp is used to couple the Griffin k-eigenvalue model to the SAM-Pronghorn model. On the other hand, a MultiApp of type TransientMultiApp is used to couple the SAM and Pronghorn domains.

4.4.2 Domain overlapping model execution

For the steady state calculation of the coupled model using domain overlapping scheme, the coupled Pronghorn-SAM system thermal-hydraulics fields are first initialized assuming a constant power distribution at the core for both the Pronghorn and SAM core models. Then, the k-eigenvalue problem is solved in Griffin coupled to the Pronghorn-SAM system. A *FullSolveMultiApp* couples the Griffin k-eigenvalue model to the Pronghorn-SAM one, whereas Pronghorn and SAM are coupled with a *TransientMultiApp* of matching time step. Piccard iterations are carried out between the steady-state k-eigenvalue problem for neutronics with the transient thermal-hydraulics system until convergence to 1e-8.

4.4.3 Steady-State Results

The results for key fields during steady-state operation are depicted in Figure 4-4 - Figure 4-7. The results from both coupling schemes are the same. Figure 4-4 shows the power density along with the fastest and most-thermal groups neutron flux. An epithermal spectrum is expected for the reactor with the largest power density at the center of the reactor core cavity. Figure 4-5 shows the velocity and the temperature fields. The velocity is higher in the inlet and outlet pipes, where the flow enters and exits the core cavity. No significant recirculation is predicted next to the core cavity with this axisymmetric model. The temperature field is higher at the center of the reactor due to the larger power density in this region and the artificially large viscosity predicted by the mixing-length turbulence model next to the reactor axis. Figure 4-6 depicts the fission source and longest- and shortest-lived neutron precursors families. As expected, the fission source distribution closely follows the power density. The shortest-lived precursors distribution has a distribution

closer to the fission source one, whereas the longest-lived precursors are advected and diffuse in the flow field. Heat transfer in the multi-loop model in SAM goes as expected from the reactor into the secondary circuit as shown in Figure 4-7. In all cases, the inlet temperature to the core is ~950K, the temperature rise over the core is ~1050K, and the fuel mass flow rate is ~18,000 kg/s.

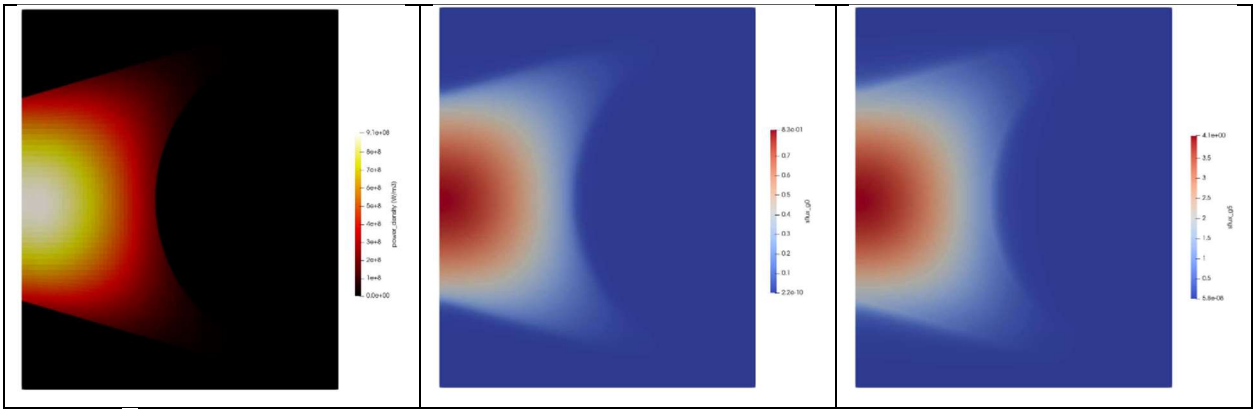

Figure 4-4. Neutronics fields for steady-state operation of the MSFR system including power density (left), fastest group neutron flux (center), and thermal-most group neutron flux (right).

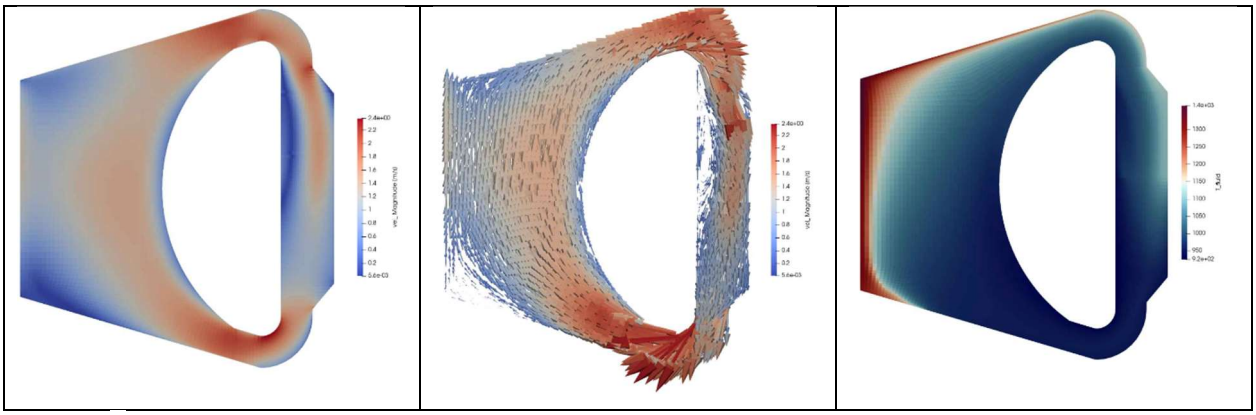

Figure 4-5. Thermal-hydraulics fields for steady-state operation of the MSFR system including velocity magnitude contour plot (left), velocity magnitude vector plot (center), and temperature field (right).

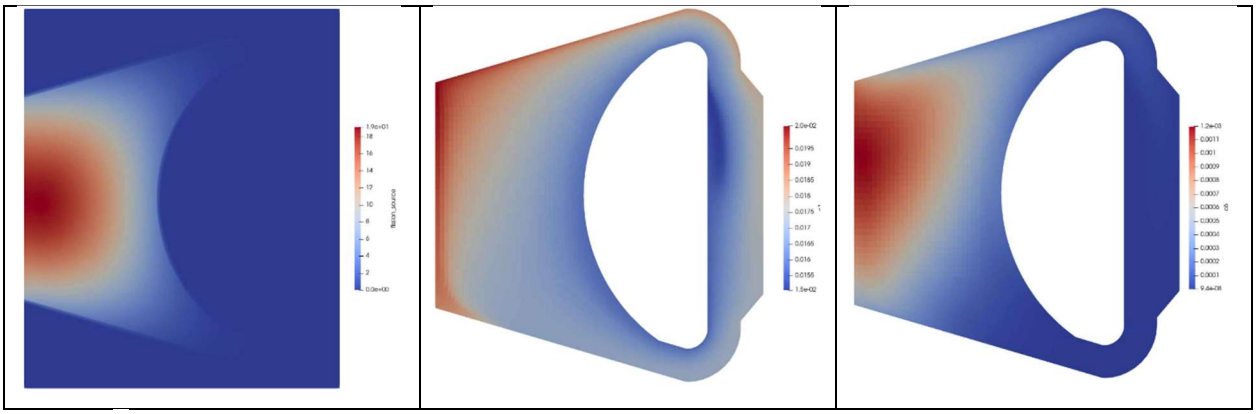

Figure 4-6. Neutron precursors fields for steady-state operation of the MSFR system including fission source (left) and shortest-lived (center) and longest-lived (right) delayed neutron precursors families.

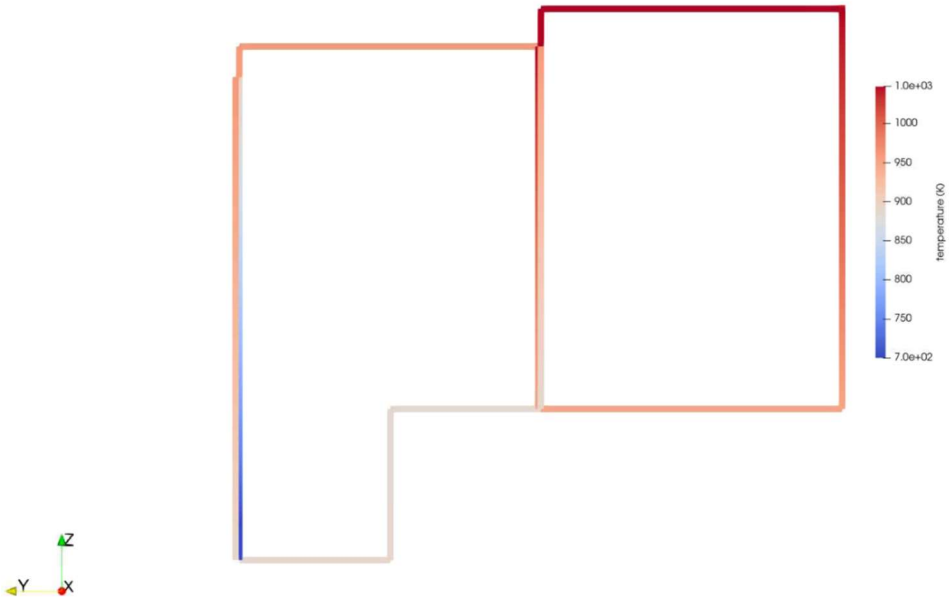

Figure 4-7. Steady-state thermal field in the complete SAM model.

4.5 Transient Demonstrations

4.5.1 Transient simulations of standalone 1-D SAM model

Transient cases are first studied for the standalone 1-D MSFR system model. Two transient scenarios are considered: (a) the 50% pump head loss for the primary pump in 40 seconds, and (b) the total pump trip of the primary pump in 60 seconds. The transient cases were restarted from the steady-state solutions, and the pump head change is initiated at t = 120 *s*. The neutronics response is modeled by the Point Kinetics Equations (PKE). As shown in Figure 4-8, right after the change in primary pump head, the fuel mass flow rate started to decrease. For the transient case of 50% pump head loss, a new steady state was reached at 70% the original mass flow rate. As for the case

of total pump trip, the fuel mass flow rate drops significantly and the new converged mass flow rate is only 12% the original value. Under the new steady-state after the pump trip, natural circulation becomes the dominant mechanism to drive the fuel circulation of primary circuit.

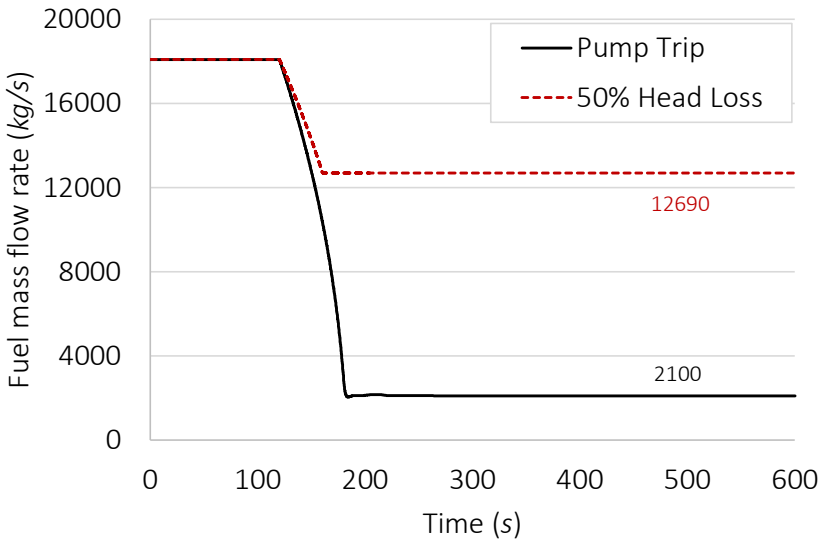

Figure 4-8. Evolution of fuel mass flow rates during the investigated transient scenarios.

Meanwhile, due to slower mass flow rate, the fuel is heated for longer period of time inside the core. As a result, the core outlet temperature started to rise after the pump head drop as shown in Figure 4-9. For the case of 50% pump head loss, the core outlet temperature is observed to be 20 *K* higher than that with normal operating condition. While with the total pump trip, the core outlet temperature can be 163 *K* higher. Due to the negative temperature feedback coefficient, the reactor power decreases as the core temperature rises. As shown in Figure 4-10, the reactor power drops about 3.3% when the primary pump loses half of its capacity, and about 53% when the primary pump stops working. This well demonstrates the passive safety of MSFR design. The corresponding temperature rise along the core is 136 *K* for 50% pump head loss scenario and 424 *K* for the pump trip case.

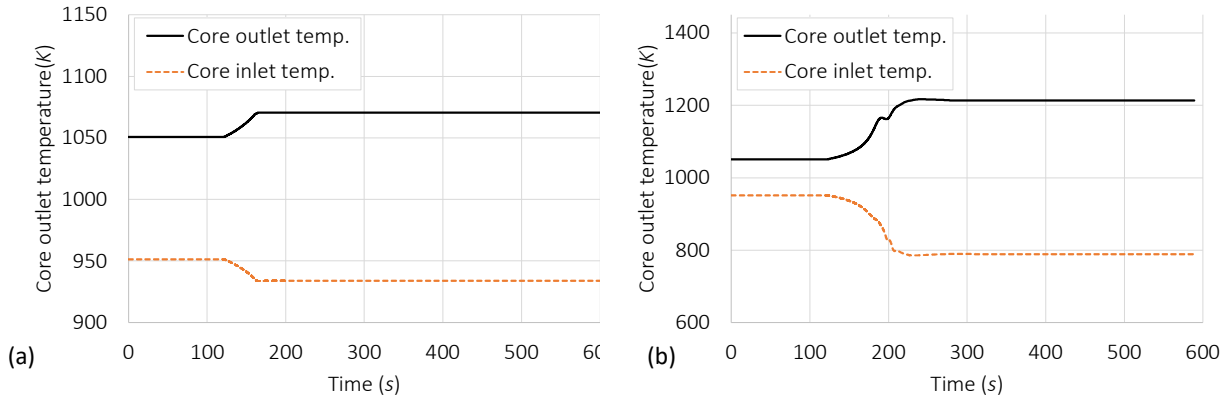

Figure 4-9. Evolution of core inlet and outlet temperatures during the investigated transient scenarios: (a) 50% pump head loss, (b) pump trip.

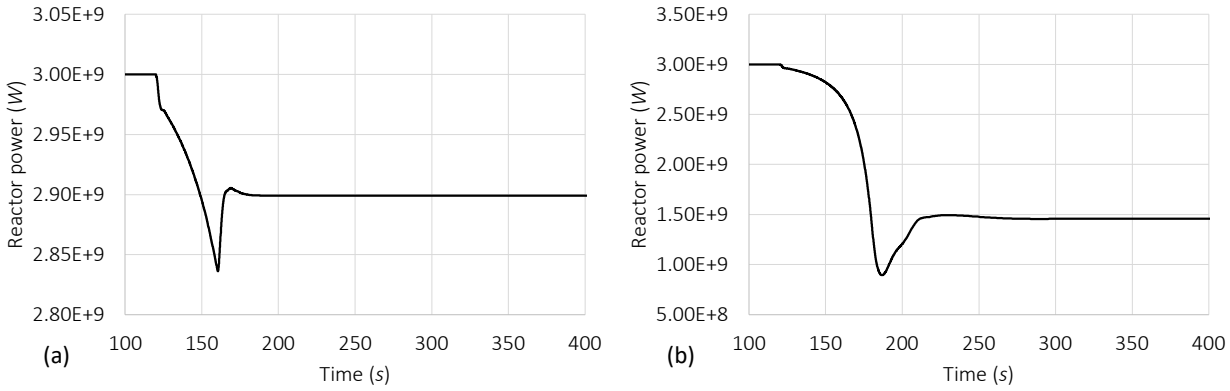

Figure 4-10. Evolution of reactor power during the investigated transient scenarios: (a) 50% pump head loss, (b) pump trip.

4.5.2 Transient simulations of Coupled SAM-Pronghorn-Griffin Model

The same two transients are studied with the domain-segregated and domain-overlapping approaches:

1. A linear primary loop pump cost-down to 50% of the nominal flow in 40 seconds
2. A linear pump trip from the nominal flow with complete pump stop 60 seconds after the beginning of the trip

Results for both approaches are presented separately.

4.5.3 Separate domain approach

Case 1: Pump head pressure coastdown to 50% of nominal value.

The transient cases were restarted from the steady-state solutions, and the pump head change is initiated at t = 1240 s. For the transient case of 50% pump head loss, a new steady state was reached at 68.5% the original mass flow rate, very close to the 70% mass flow from the stand-

along SAM simulations with PKE. Meanwhile, due to slower mass flow rate, the IHX outlet temperature and core inlet temperature decreased since the secondary maintained the high flow rate. The core outlet temperature is observed to increase during the transients. Due to the negative temperature feedback coefficient, the reactor power decreases as the core temperature rises. As shown in Figure 4-11, the reactor power at the new steady-state drops about 4.7% from the normal operating conditions.

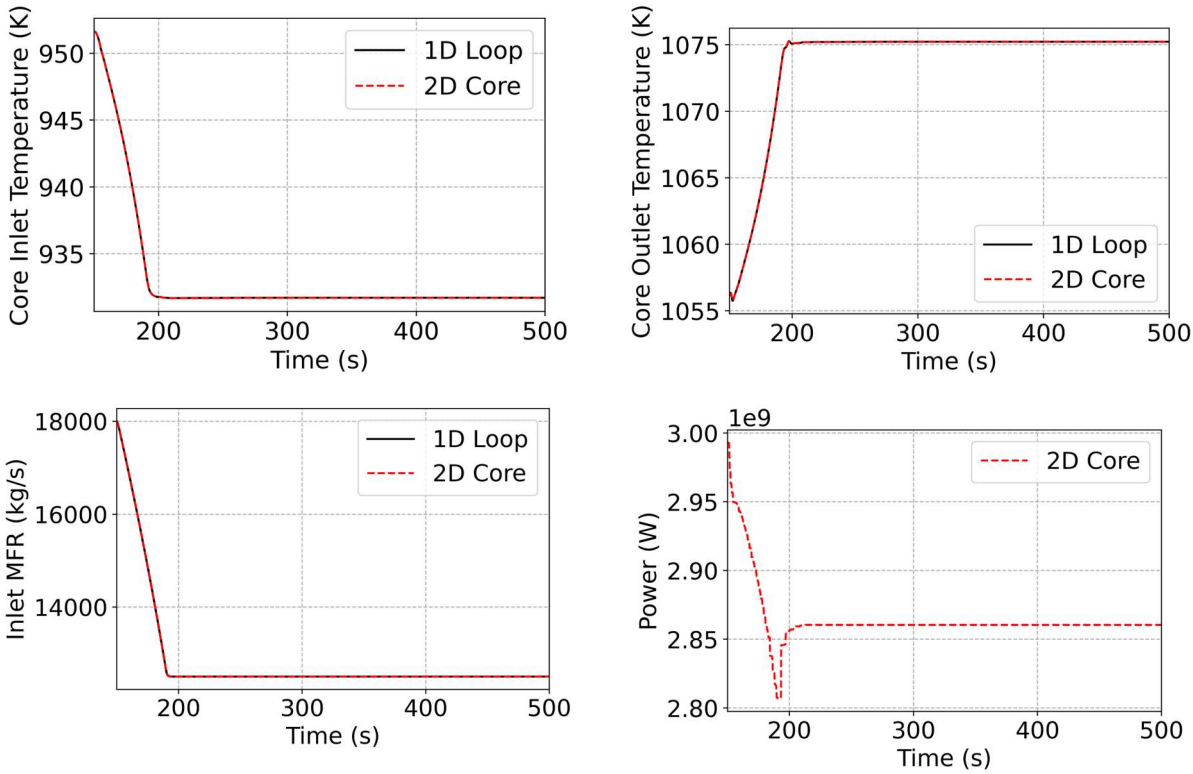

Figure 4-11. Primary loop system response during the pump cost-down to 50% of the nominal pump head transient, separate domain coupling

Case 2: Pump trip case

For the transient case of pump trip, the intermediate loop remains working. Therefore, the core inlet temperature would continuously drop while the primary loop mass flow rate would decrease, and core outlet temperature would increase. As shown in Figure 4-12, the primary loop reached natural circulation at 15.7% of the original mass flow rate, higher than the 12% mass flow from the stand-along SAM simulations with PKE. Meanwhile, the reactor power also reached new steady-state at 60.7% of the normal operating conditions, comparing to the 47% nominal power from the stand-along SAM simulation. The overall transient responses are very similar between the separate domain coupled simulations and the stand-alone SAM simulations.

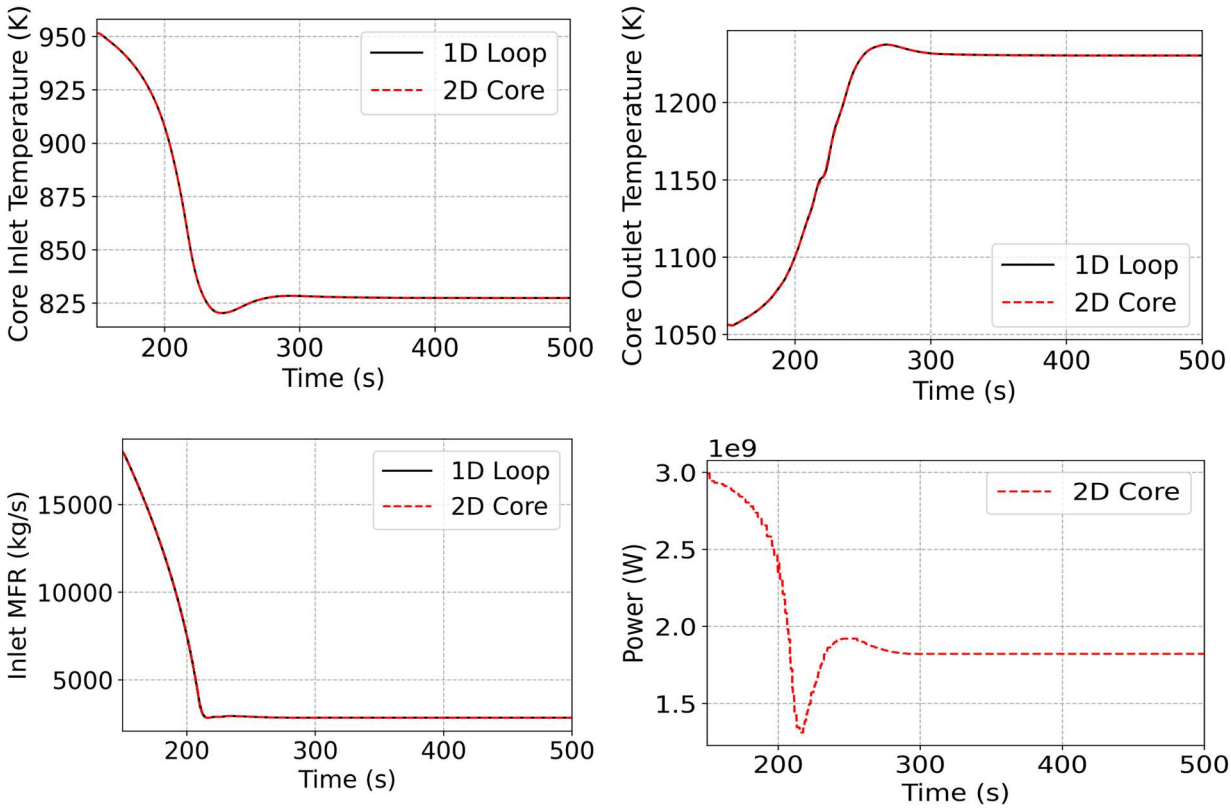

Figure 4-12. Primary loop system response during the pump trip transient, separate domain coupling

4.5.4 Domain-overlapping approach

The evolution of the system fields during the pump coast-down to 50% is depicted in Figure 4-13 and Figure 4-14. Only the Doppler feedback effect has been considered in the domain overlapping approach, i.e., no density feedback effect is considered. The relative pump power with respect to the nominal value is presented with dashed black lines in both figures. As the pump coast-down starts, the temperature in the core rapidly increases, causing the power to rapidly reduces due the inverse neutron prompt jump. Then, after the inverse prompt jump ends, the reactor gets colder than the equilibrium configuration imposed by the current mass flow rate condition, which causes a small power peak. Afterwards, power keeps reducing as the reactor gets hotter while the pump reduces its power, and the density and Doppler effect reduces the effective reactivity of the primary loop. Finally, after the pump coast-down finishes, at 60 seconds since the start of the simulation, there is an influx of colder salt into the core by the over-cooled heat exchangers that still exchange heat with the nominally operating intermediate loop. This causes power to increase back up until the temperature rise in the core stabilizes the inserted reactivity. After this point, the heat exchangers in the primary loop achieve an equilibrium temperature and the power stabilizes to a new equilibrium value that is lower than the original one.

The mass flow rate during the transient reduces following the pump coast down which agrees well with the 1-D standalone SAM model (as shown in Figure 4-8). The flow inertia in the SAM model is negligeable since the mass flow rate closely follows the reduction in the pump power.

The Pronghorn model presents slightly more flow inertia as there is a small time-delay since the reduction in the pump momentum source and the stabilization of the flow rate. For both cases, the flow rate reduces with the square root of the pressure change at the pump, which is expected from the inertia-dominated losses at the reactor's cavity. At the end of the transient the inlet temperature in the core is lower due to the lower flow rate in the primary loop and the nominally operating intermediate loop that overcools the heat exchanger. The temperature rise in the core is higher than the steady-state configuration (~135K) due to the lower mass flow rate and the still relatively high power, which agrees well with the result from standalone SAM transient simulation.

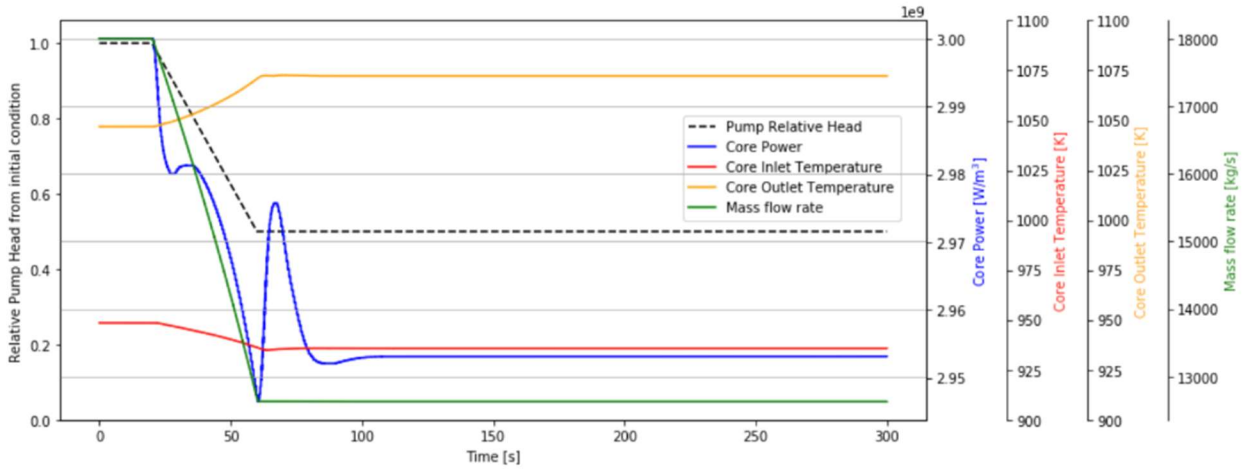

Figure 4-13. Primary loop system variables for the 1D SAM model during the pump cost-down to 50% of the nominal pump head transient.

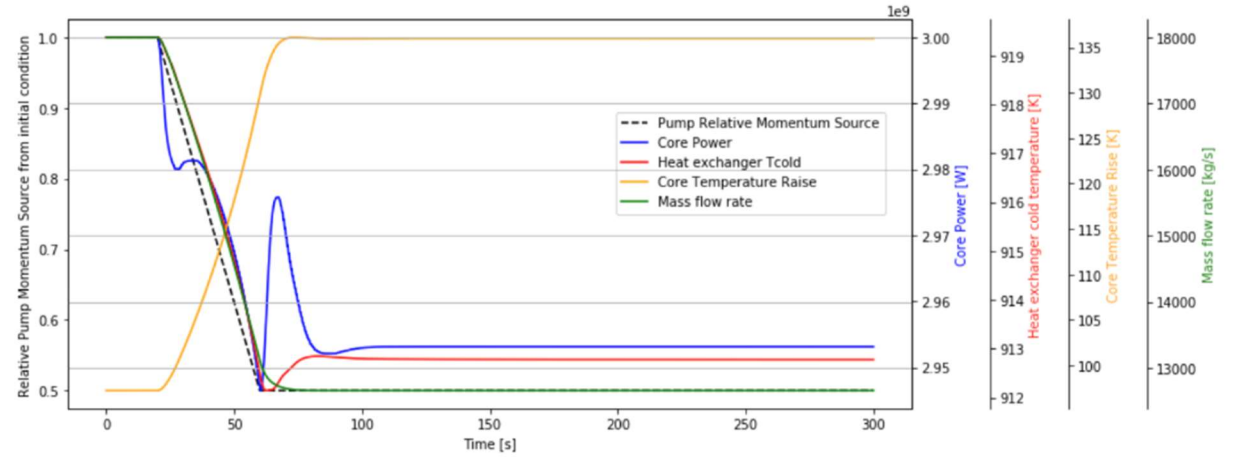

Figure 4-14. Primary loop system variables for the 2D axisymmetric Pronghorn model during the pump cost-down to 50% of the nominal pump head transient.

The evolution of the system fields during the pump trip is depicted in Figure 4-15 and Figure 4-16 for SAM's and Pronghorn core models, respectively. The evolution of the system fields is similar to the one observed for the pump coast down, except that the evolution is accentuated by the more drastic pump power reduction. For instance, the mass flow rate reduces to ~12% of its nominal value after the pump trip, in both SAM and Pronghorn models, and the flow field in the

primary loop is exclusively governed by natural convection after the pump stops. The transition to natural convection flow is much more inertia dominated in the Pronghorn model as seen from the softer evolution of the mass flow rate after the pump stops. In the SAM model, the mass flow rate in the primary loop suddenly goes into the natural convection flow rate almost immediately after the pump stops. However, in the Griffin-Pronghorn model, the mass flow rates retain longer inertia from the pump-driven operation and stablish to the natural convection value after ~70 seconds since the pump stops. The reduction in the core power during the reduction in pump power is now much larger due to the larger increase of the temperature in the core. When the reactor reaches the new equilibrium condition, the power has reduced to ~2.1 GWth and the temperature rise over the core is ~450 K. The subsequent study will focus on understanding the differences from various modeling approaches.

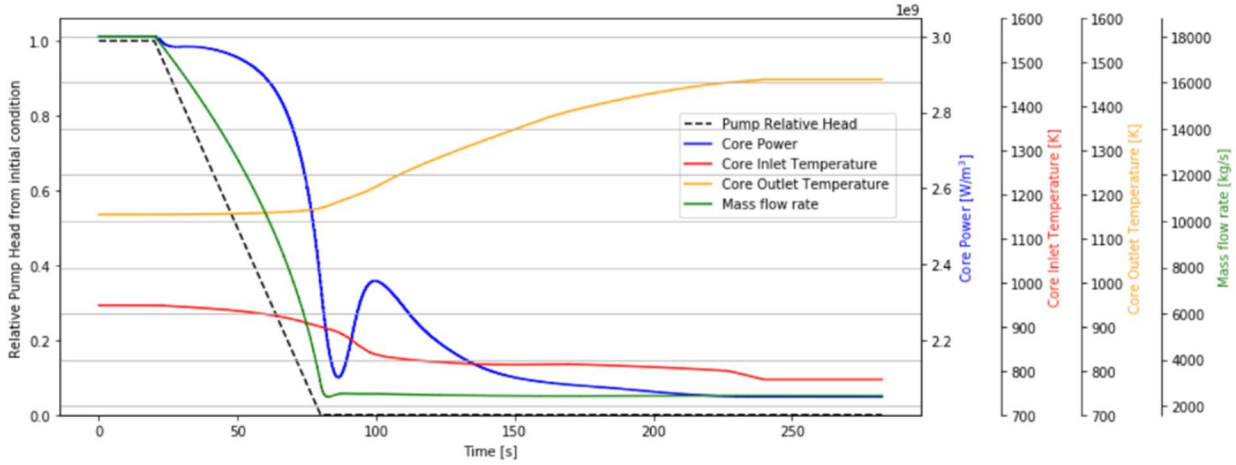

Figure 4-15. Primary loop system variables for the 1D SAM model during the pump trip transient.

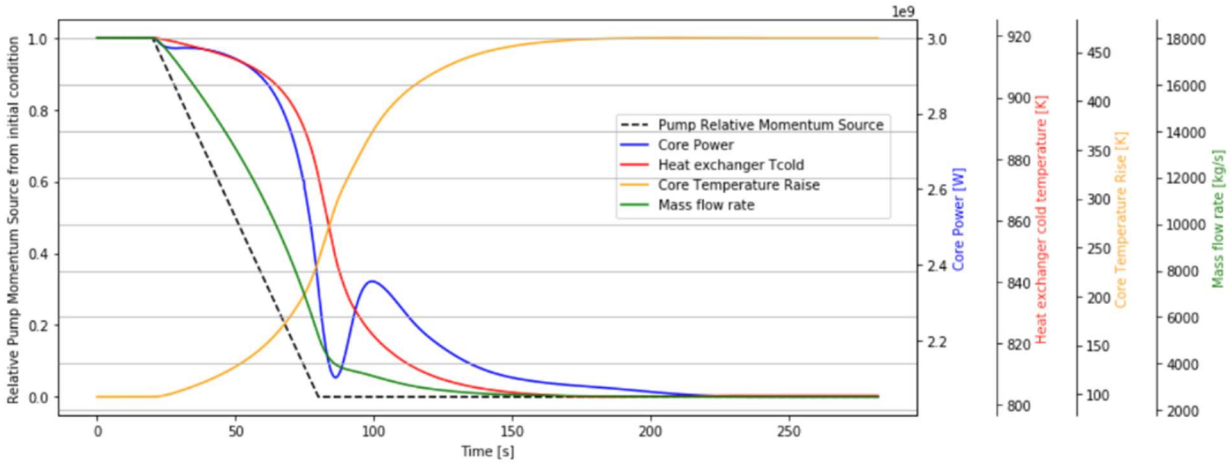

Figure 4-16. Primary loop system variables for the 2D axisymmetric Pronghorn model during the pump trip transient.

Finally, for further understanding of the transients observed, which only integrates Doppler feedback in the modeling, the effective multiplication factor, reactivity, and reactivity-temperature feedback coefficient are presented in Table 4-5. As observed, the Doppler feedback effect saturates at high temperatures, which explains the relatively low reductions in power observed in these simulations on these transients. Work is being currently performed for more appropriately integrating the density effects of the temperature feedback in reactivity. New results will the available for FY23.

Table 4-5. Effective multiplication factor, reactivity, and reactivity-temperature feedback reactivity coefficient as a function of the fuel temperature.

Fuel Temperature (K)	600	900	1000	1100	1200	1300	1500	1800
Effective Multiplication Factor	1.041854	1.026662	1.021501	1.016923	1.012145	1.010790	1.008080	1.004015
Reactivity (pcm)	4017.3	2596.9	2104.9	1664.2	1199.9	1067.5	801.5	399.9
dReactivity/dT (pcm/K)	-4.735	-4.874	-4.664	-4.525	-2.983	-1.326	-1.333	-1.339

4.6 Summary and Future Work

To demonstrate the feasibility and capabilities of a multi-scale thermal fluids capability using NEAMS thermal fluids codes, a key joint effort is pursued for an integrated thermal-hydraulics capability between SAM and Pronghorn for MSRs. Two coupling approaches were explored for SAM and Pronghorn simulations, i.e. domain-segregated and domain-overlapping approaches.

To assess the robustness of the coupling schemes, the integrated simulation of a reference MSFR was performed. A pump coast down transient, where the main salt pump head drops by 50%, and a full pump trip transient were simulated. The steady state and transient SAM-Pronghorn coupled simulations results provide a good basis for the successful demonstration of the utilized coupling approaches. However, future work will need to be carried out, including but not limited to:

- Further verification, demonstration simulations, and performance assessments of complex transients of various coupling schemes.
- Development and testing of a fluid-solid interface coupling approach for thermal coupling (i.e. heat transfer over the heat exchanger).
- Improvements in the robustness of the coupling schemes. The domain overlapping approaches used in SAM and NekRS coupling showed great successes in several verification and validation test cases. This approach can be pursued as another option for the integrated SAM-Pronghorn capabilities.
- An integrated model with single-solve approach has been developed in the SAM code and tested with a simple sodium loop test case. The 3D fluid equations and 1D fluid equations are tightly coupled in the nonlinear residual level and solved simultaneously to overcome potential convergence issues when 3D fluid equations and 1D fluid equations are solved separately. The feasibility of the single solve approach could be also studied for the integrated SAM and Pronghorn capabilities.

5 Updates of SAM Gas Transport Model

In MSR designs, it is common for a sparging gas to be present in the flow loop to aid in the removal of fission products such as xenon and krypton, as was the case of the Molten Salt Reactor Experiment (MSRE) that was performed at Oak Ridge National Laboratory (ORNL) as documented by Engel and Steffy [35]. To support the MSR modeling needs, a gas-transport model was implemented into SAM.

The use of entrained helium gas for fission gas removal requires knowledge of gas interfacial area, which is the surface area available for insoluble gasses to transfer from the liquid to the sparging gas phase. As will be shown in this section, this term will be impacted by local thermal-hydraulic conditions. Gas velocity and its relationship to the bulk salt velocity will impact the volume fraction of the gas as well as the gas bubble radius and both terms impact the interfacial area. Void fraction is also highly dependent on local gas density, which unlike the salt phase, varies strongly with respect to local pressure. Once the interfacial area distribution and gas velocity are known, a species tracking code such as the MOOSE-based chemistry code Mole [36], could take these terms as input to determine the rate of transfer of insoluble gasses into the gas phase.

It is also noteworthy that the presence and behavior of the gas phase does not only impact the behavior of reactivity sinks like xenon and iodine, but can also act as a sink for collecting and redistributing noble metals as well, which can deposit on flow loop surfaces and salt/gas interfaces as described in [37]. Additionally, while volume fraction was low enough that it did not have a significant impact on reactivity feedback in designs like MSRE, it is possible in molten salt fast reactor (MSFR) designs that utilize a more open flow area for localized void to increase to void fractions greater than 1% as shown in [38]. Note that the current model implementation in SAM does not consider the impact of void on the salt governing equations, but this may be an area to research in the future. An additional possible aspect for future work is considering the decay heat of fission products that migrate into the gas, as this information may be necessary for additional heat removal requirements in the off-gas system.

A preliminary implementation of the gas transport model was documented in [39], but it was found through further testing that problems could be encountered with conserving mass of the gas phase in more complex models. To address this issue, the gas transport model was reformulated to improve the stability term and better conserve non-condensable gas mass. The user-interface to the model was also expanded and improved as part of this work. Testing of the model was then expanded substantially by first developing model requirements and implementing a matrix of regression tests to cover each of the requirements. In addition to the regression tests, validation testing and uncertainty quantification (UQ) was also performed using recent experiments of helium bubbles in a molten salt system. To demonstrate the performance of the gas transport model for geometry and conditions of interest, it was used in modeling reactivity insertion and flow change transients in MSRE. A final activity for improving the SAM MSR support was integration of the Saline salt thermophysical property database [40], which provides an interface to the Molten Salt Thermal Database-Thermophysical (MSTDB-TP) [41].

5.1 Improvements of gas transport model in SAM

The initial implementation of the gas transport model, documented in [39], was done by treating the gas phase as a transported species. The general species transport equation in SAM is shown in Eq. (1).

$$\frac{\partial \rho c_i}{\partial t} + \nabla(\rho u c_i) - \nabla(D_i \nabla c_i) + \lambda_i \rho c_i = S_i \tag{1}$$

In the equation, ρ is liquid density, c_i is the species concentration, u is liquid velocity, D_i is the diffusion coefficient, λ_i is the decay coefficient for the species, and S_i is the species volumetric source term. If a species was flagged as a gas by user input, ρ was instead treated as the density of the gas and the decay term was ignored. Furthermore, a drift-flux model was added so that u could be calculated to account for buoyancy and gas distribution effects on the gas velocity. Note that when ρ is swapped with the gas density and u is swapped with the vapor velocity, c_i naturally becomes equivalent to the gas volume fraction (α), as can be seen by observing the gas transport equation.

$$\frac{\partial \alpha \rho_g}{\partial t} + \nabla(\alpha \rho_g u_g) = S. \tag{2}$$

Note that the diffusion and decay terms were removed in this equation. Accounting for this, it was considered that c_i represented α when the species was flagged by the user as a gas. For simple, single-tube geometry tests, this formulation worked to demonstrate the correct functioning of the model; however, once model complexity was increased to flow loops, substantial difficulty was observed in conserving gas mass throughout a transient. To provide background for this issue, it is noted that SAM uses a finite element method (FEM) approach for discretizing the governing equations and it is well-known that advection-dominated problems like the gas transport model are prone to numerical oscillations when solved by FEM. Therefore, SAM employs the Streamline-Upwind/Petrov-Galerkin (SUPG) and Pressure-Stabilizing/Petrov-Galerkin (PSPG) stabilization methods, which are discussed in more details in the SAM theory manual [4]. Briefly put, the stabilization methods add artificial diffusion in the streamline direction to damp out oscillations while still satisfying the weak form of the governing equations exactly. For the species transport equation, the stabilization term is calculated as the stabilization parameter, τ, times the governing equation. Ignoring the diffusion, decay, and source term, this is as shown in Eq. (3).

$$\tau\left(\frac{d(c_i \rho)}{dt} + \frac{d(c_i \rho u)}{dx}\right) \tag{3}$$

The second term in this equation can be split using the chain rule and it can be assumed that $d(\rho u)/dx$ is equal to zero because this will be ensured by the liquid mass governing equation also solved by SAM. However, this assumption is no longer correct when solving a gas transport equation, which has no other mass equation being solved to guarantee this to be true. Therefore, the stabilization term for the gas transport equation shall be as shown here.

$$\tau\left(\frac{d(\alpha\rho_g)}{dt}+u_g\frac{d(\alpha\rho_g)}{dx}+\alpha\rho_g\frac{du_g}{dx}\right) \quad (4)$$

To allow the stabilization to be implemented as shown, the gas transport model was modified so that the solved nonlinear variable is $\alpha\rho_g$ rather than just α. Rather than using the general species transport equation to model gas transport, the implementation was modified to define a new set of kernels: *GasTransport* and *GasTimeDerivative*, to represent the convection/source term and time derivative terms of the governing equation given by Eq. (2). This results in a cleaner implementation because conditional statements are no longer needed in the *PassiveScalarTransport* kernel to determine which density shall be used and which nonlinear variable is being solved. Furthermore, the model was implemented in a way that allows for only a single gas to be modeled in the system, which is consistent with the assumption made by the form of the implemented gas transport model. Because the mass transport model allows for multiple species, additional checks were required to prevent multiple gasses from being modeled by the user.

The AuxKernel used for calculation of volume fraction was also simplified, as it is now only a matter of dividing the nonlinear variable by local gas density to obtain the volume fraction. The drift flux model and interfacial drag calculation were not modified in terms of the physical models being solved and, aside from adjusting the Weber number and salt surface tension property, which will be discussed further in Section 5.3, the gas bubble radius calculation is also unmodified.

With this new implementation, the input options that are now exposed to the user are as follows:

1. gas_model: If set to true, the gas transport equation will be solved, which means additional input will be required for gas mass equation boundary conditions, gas equation of state (EOS), and so on.
2. gas_scaling_factor: This is the scaling factor for the gas equation residual
3. gas_drift_flux: If set to true, the gas will be transported by the drift flux velocity
4. eos_gas: This is a new input for every component, junction, and boundary condition in the model, which is required when gas_model is enabled. It shall be the name of an EOS that was defined in the EOS block of the input.
5. void_bc: This is required in the boundary condition components when the gas model is enabled and it specifies the volume fraction of the gas at that location. Note that the gas mass flow rate at the boundary condition will be calculated using the gas drift flux velocity if that model is enabled.
6. gas_fn: This is an optional input that can be specified for components to point to a function that will be used to specify the gas source term in units of kg m^{-3} s^{-1}.
7. gas_source: This is an optional input that can be specified for components to set a constant, uniform gas source term in units of kg m^{-3} s^{-1}.

Additionally, the user can use the following names to obtain output data:

1. gas_void: Returns gas volume fraction for the postprocessor.
2. gas_aint: Returns gas bubble interfacial area for the postprocessor.

3. gas_bubble_rad: Returns gas bubble radius for the postprocessor.
4. velocity_gas: Returns gas velocity for the postprocessor, which will be equal to bulk velocity if the drift flux model is disabled, or the drift flux velocity if it is enabled.
5. rhog: Returns the gas density for the postprocessor, as calculated by the EOS that the user specified for the gas model.

Finally, a postprocessor called *ComponentNodalGasMassFlowRate*, which provides the gas mass flow rate at selected component locations.

5.2 Regression testing

A critical aspect of implementing a new model into a code is to ensure that automated regression testing is sufficient to guarantee a certain level of quality and robustness of the model. To ensure that the gas transport model is adequately tested, the supported modeling capabilities and features of SAM were identified, and a suite of tests were developed to test the model for each identified feature. The test matrix is shown in Table 5-1.

Table 5-1. Test matrix for the gas transport model implemented into SAM

Model Features			Tests																							
			gas_vertical_pipe	gas_wave	gas_drift_flux_pipe	gas_source_term_drift	gas_point_source_drift	gas_loop_unheated	gas_loop_heated	gas_loop_drift	gas_velocity_fn	gas_branch_2to2	gas_heated_elbow	gas_coupledppstdj_bulk	gas_coupledppstdj_drift	gas_velocity_bulk_fn	gas_source_term_bulk	gas_point_source_bulk	gas_loop_source	gas_hx_source_bulk	gas_pbcore_source	gas_pbcore_point_source	Fully tested
	Initial conditions	constant void	x	x	x	x	x	x	x	x	x	x	x				x	x	x				✓
	Boundary conditions (standalone)	Bulk velocity	x	x								x								x	x	x	✓
		Drift velocity			x								x										✓
		Time-dependent bulk velocity														x							✓
		Time-dependent drift velocity									x												✓
		Constant void	x		x	x	x	x	x	x	x	x	x				x	x	x	x	x	x	✓
		Time-dependent void		x																			✓
	Boundary conditions (coupled)	Bulk velocity												x									✓
		Drift velocity													x								✓
		Time-dependent bulk velocity																					
		Time-dependent drift velocity																					
		Constant void												x	x								✓
		Time-dependent void																					
	Source terms (PBOneDFluidComponent)	Scalar gas source															x						✓
		Function gas source																x	x				✓
	Source terms (PBHeatExchanger primary)	Scalar gas source																		x			✓
		Function gas source																	x				✓
	Source terms (PBHeatExchanger secondary)	Scalar gas source																					
		Function gas source																					
	Source terms (PBCore)	Scalar gas source																			x		✓
		Function gas source																				x	✓
	Components	PBOneDFluidComponent	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x				✓
		PBPump						x	x	x									x				✓
		PBBranch						x	x	x		x	x						x				✓
		PBSingleJunction						x	x	x									x				✓
		PBHeatExchanger							x	x									x	x			✓
		PBCoreChannel																			x	x	✓
		PBTDJ	x	x	x	x	x				x	x	x			x	x	x		x	x	x	✓
		PBTDV	x	x	x	x	x				x	x	x			x	x	x		x	x	x	✓
		PressureOutlet						x	x	x									x				✓
		CoupledPPSTDJ												x	x								✓
		CoupledPPSTDV												x	x								✓
	Orientation	Vertical Upwards (bulk)	x	x				x	x			x		x	x	x			x	x	x	x	✓
		Horizontal (bulk)						x	x			x							x				✓
		Vertical Downwards (bulk)						x	x										x				✓
		Flow loop (bulk)						x	x										x				✓
		Vertical Upwards (drift)			x	x	x			x	x		x				x	x					✓
		Horizontal (drift)								x			x										✓
		Vertical Downwards (drift)								x													✓
		Flow loop (drift)								x													✓
		Heating							x	x											x		✓
	AuxKernels	ScalarBubbleRadiusAux			x					x									x				✓
		ScalarDriftVelocityAux			x					x					x				x	x	x	x	✓
		ScalarInterfacialAreaAux			x					x									x				✓
		ScalarVolumeFractionAux	x	x	x	x		x	x	x	x	x	x	x	x	x	x		x	x	x	x	✓
		DensityAux (rhog)	x	x	x	x				x				x	x		x		x	x	x	x	✓
	PostProcessors	ComponentNodalGasMassFlowRate				x	x	x	x	x	x	x	x	x	x		x	x	x				✓

The primary modeling features of SAM are identified in the left column and includes categories like boundary conditions, initial conditions, and source terms. The criteria for determining a "modeling feature" involved finding any component or kernel that will be affected by the gas transport model. A modeling feature is something that the user will directly or indirectly interact with in setting up the model or in obtaining output data from the model. An additional category, "Orientation" was added to capture specific modeling scenarios that result in a unique way in which the physics is modeled by SAM. For example, in downward flow, the drift flux model will behave differently than upwards flow where gravity is in the opposite direction to the

flow and results in a buoyancy force. Each of these categories is broken down into more specific features that can be modeled in SAM.

The top row of the matrix provides the name of each test that was added to test the modeling features. An "x" is placed in the intersection of the test and the modeling features it activates. Note there are 5 rows (features) that are left untested. The first three are related to coupled boundary conditions. It was unclear that setting time-dependent boundary conditions in the coupled boundary condition block, which is used to obtain boundary condition data from a code that is coupled to SAM, is a feature that requires support because when coupled to another code, SAM should only be accepting the instantaneous velocity or void that was made available in that coupling object rather than overriding those values with Function objects. For the second set of rows, it is not currently intended that modeling of gas transport in the secondary flow loop (on the other side of the heat exchanger) will be needed, so no test was added for this capability.

Each of these tests has a set of acceptance tests which are defined and documented directly in the test input files. Each individual acceptance test results in a boolean, with true meaning the test passes. The regression tests were implemented into the SAM automated testing system.

One of these regression tests, called the "gas_loop_drift" test, was developed to demonstrate the performance of the gas transport model in conditions that are more representative of what will be modeled using sam. The test is a flow loop that includes a heater on one side, a heat exchanger on the other side, a pump to move the fluid, a pressurizer to allow for fluid expansion, and for the gas model, injection and removal points in the loop. This case was created to test the gas transport model and the impact of the drift flux model for flow loop behavior. Gas injection was adjusted to achieve a gas volume fraction that was at a level close to those that were observed in the U-235 fueled MSRE tests. An additional version of this model was created with the drift flux model disabled and the gas injection rate adjusted to achieve a similar mass flow rate as the drift flux case; this was done so that a comparison can be made between the two cases (with and without the drift flux model). Both cases were run for a 60 second transient, which was observed to be long enough for the model to become steady. Heat input was set to achieve a heater temperature rise of approximately 12 K and the pump head was set to achieve a flow velocity of about 0.75 m s^{-1}. Gas was injected using a space- and time-dependent function which injects gas over the first 0.1 m of the heater side and removes gas from the first 0.1 m of the heat exchanger component. A schematic of the test geometry is shown in Figure 5-1.

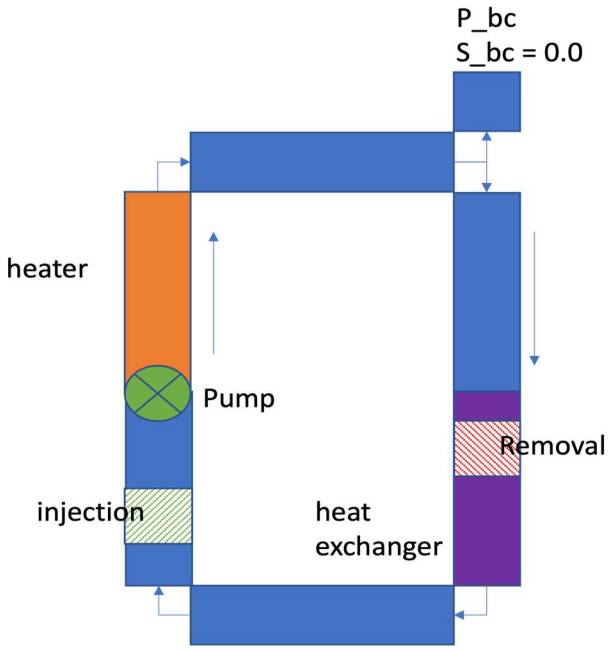

Figure 5-1. Schematic of the flow loop test problem added to the SAM regression tests for testing of the drift flux model.

Note that the model was run by starting the gas injection at the beginning of the transient for both cases. The gas removal was enabled later in the transient. This was necessary to prevent the gas mass flow rate from becoming negative in the beginning of the transient when the injected gas has not yet made its way around the flow loop. Because the gas flows at a different velocity when the drift flux model is enabled, the time of removal needs to be adjusted when drift flux is enabled. For the bulk velocity case, removal was started at 7 s and for the drift velocity case, removal was started at 5.2 s Note also that the removal was started slightly after gas had already flowed passed the removal point, meaning that some gas is always in the system.

The gas density distribution is shown in Figure 5-2 for the drift velocity case, though it was found that both the bulk velocity and drift velocity cases had essentially identical density distributions. The gas density is primarily driven by the pressure distribution. Of the checked locations, pressure will be highest at the heater top because it is directly downstream of the pump, so gas density is also largest at this location. Density drops at the top pipe exit (orange line) and again at the exit of the downward vertical pipe (green line) due to the pressure drop, but then jumps again at the outlet of the heat exchanger due to the temperature drop. It drops to its lowest value in the bottom horizontal pipe exit where pressure and temperature will be lowest amongst checked locations.

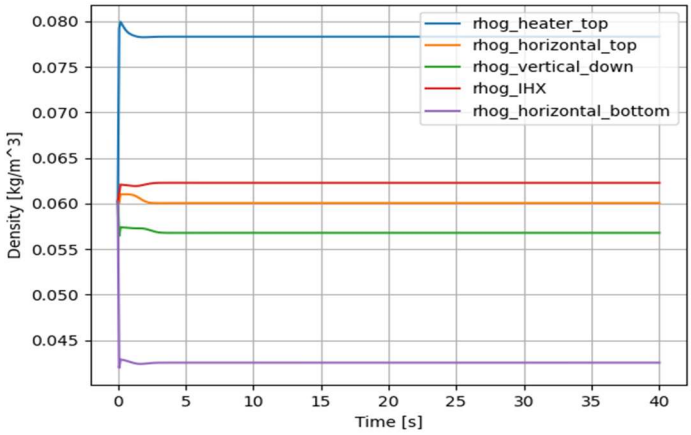

Figure 5-2. Gas density distribution in the system.

The gas velocity distribution for both the bulk-velocity case and drift-velocity case is shown in Figure 5-3 and Figure 5-4, respectively. Velocity is essentially constant throughout the system for the bulk velocity case. For the drift velocity, gas velocity is much higher at the heater exit because the buoyancy effect is applied in the calculation of drift velocity due to the gravity being in opposition to the flow direction. In the horizontal members and the downward vertical members, the buoyancy effect is disabled; however, there is still a distribution effect that accounts for the vapor moving faster in the core region of the component. This leads to velocity still being elevated in the non-heater components when compared to the bulk velocity case.

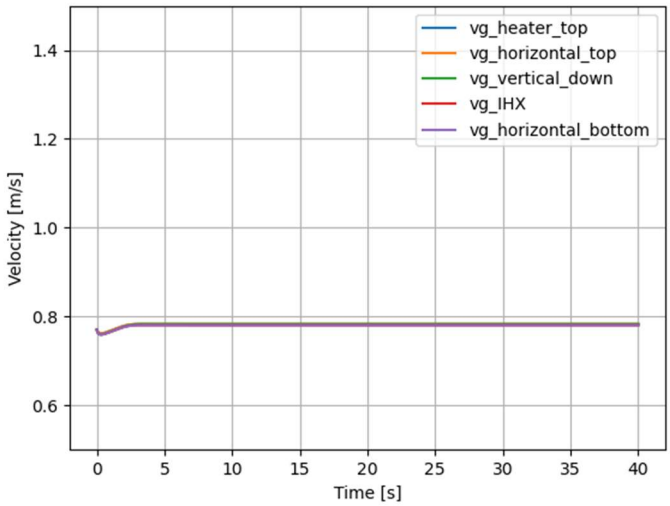

Figure 5-3. Velocity distribution in the bulk velocity case.

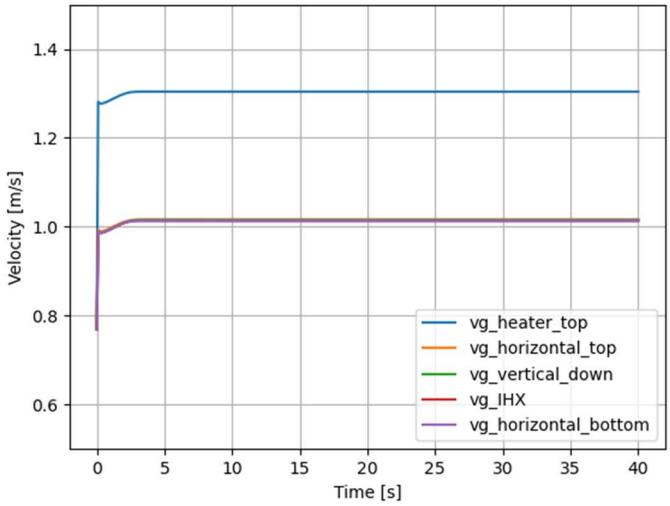

Figure 5-4. Velocity distribution in the drift velocity case.

Figure 5-5 and Figure 5-6 shows the void distribution in the system for both the bulk-velocity and drift-velocity cases, respectively. Here, the overall trends are the same in terms of where void is highest and lowest. Void is lowest in the heat exchanger and bottom horizontal pipe members because gas is removed in the heat exchanger. Conversely, it is higher in the other three members. Beyond that, void is lower where density is higher and higher where density is lower. Overall, the drift velocity case experiences lower void and this difference represents the impact of using the drift flux model; the velocity is higher, which leads to a lower volume fraction. Maximum system void drops from 8e-4 to 6e-4 when drift velocity is enabled.

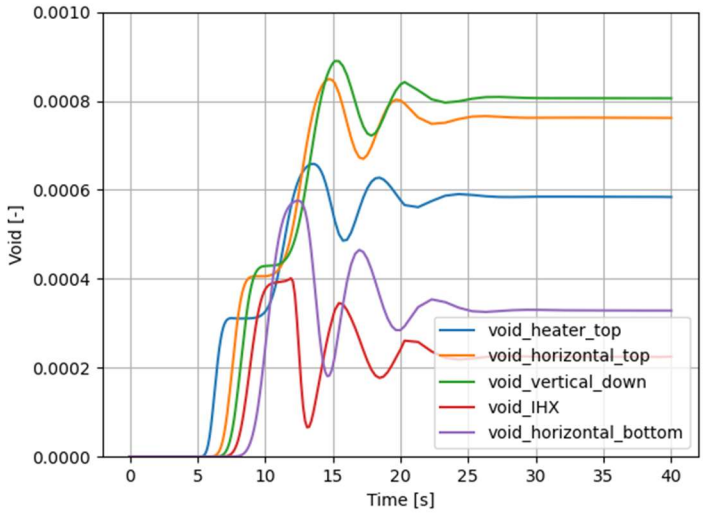

Figure 5-5. Void distribution in the bulk velocity transport case.

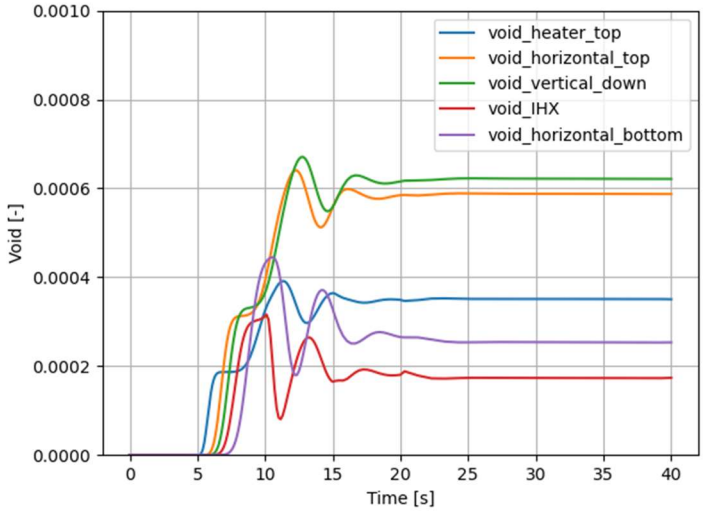

Figure 5-6. Void distribution in the drift velocity transport case.

5.3 Validation of the gas transport model

Recent experiments were performed by Chavez et al. [42] that are relevant for the validation of the SAM gas transport model. In the experiments, helium bubbles were injected, one-by-one, in the bottom of a tube filled with molten, stagnant LiF-NaF-KF (46.5-11.5-42 %). Due to buoyancy, the bubbles rose to the top of the tube, eventually reaching a terminal velocity. Because the salt is not moving, these tests do not serve as a validation of the drift flux distribution parameter (C_0 and C_∞); however, it does allow for validation of the buoyancy portion of the drift flux model, v_{gj}. This validation is important because the buoyancy drift flux model was not originally developed for a liquid salt/helium two-fluid system. A quartz viewing chamber near the top of the tube was used to photograph the bubbles using a high-speed camera, which allowed the bubble shape and size to be determined as well as the bubble velocity. A picture of the experimental facility can be found in Chavez et al. [42].

Based on experimental observations, it was found that bubbles most often exhibited an oblate spheroid shape. The volume of an oblate spheroid is defined by Eq. (5), where w is the bubble width, h is the bubble height, and V_B is the bubble volume. The volume can also be used to calculate an equivalent bubble diameter (d_{eq}), as shown in Eq. (6).

$$V_B = \frac{\pi}{6} w^2 h \tag{5}$$

$$d_{\text{eq}} = \sqrt{\left(\frac{V_B}{\pi/6w}\right)} \tag{6}$$

From the bubble photographs, Chavez measured bubble width and height and used this to calculate the bubble equivalent diameter. Additionally, instantaneous bubble velocity was also calculated from the photographs by considering the change in bubble location with respect to the time between successive photographs. A summary of the measurement data can be found in the paper. Note that a measurement for a single bubble is presented as the mean and standard deviation of data collected from all of the photographs of that bubble as it traverses the viewing window. The data also includes a corrected velocity, calculated using the correlation recommended by [44] (Eq. (7)), to account for wall effects. It was noted that when the ratio of bubble diameter to tube diameter is greater than or equal to 0.2, wall effects must be considered. This ratio for these experiments ranged between 0.15 and 0.33, requiring correction of the data.

$$V_{T,\infty} = V_T(1-\lambda)^{\frac{-3}{2}} \tag{7}$$

The paper also states that the helium injection uncertainty is $\pm$4.2%, bubble height and width measurement uncertainty is $\pm$3%, calculated d_{eq} uncertainty has a RMS of $\pm$9.44%, and fluid properties, used for calculation of dimensionless quantities, (surface tension, density, and viscosity) have an assumed uncertainty of $\pm$2%. The author also calculated the average Weber number (We), which is the ratio of inertial forces to surface tension forces (Eq. (8)), for all bubbles to be 3.62 with a propagated measurement uncertainty of $\pm$13.7%. In Eq. (8), ρ_ℓ is the salt density and σ_ℓ is the surface tension of the salt and helium interface.

$$\text{We} = \frac{\rho_\ell V_T^2 d_{\text{eq}}}{\sigma_\ell} \tag{8}$$

As discussed in [39], the Weber number is used to determine the bubble diameter. The Weber number for a specific two-phase mixture must be determined empirically. In the previous report, this value was set to 3.0 based on a value provided in [43]. For this validation study, this value was updated to the value of 3.62 given in the more recent paper. This parameter was also exposed in the SAM input file to allow the user to modify its value more easily. As observed from Eq. (8), the values used for density and surface tension will also have an impact on the Weber number and, thus, the bubble diameter calculated by SAM. The author provided the equations they used for salt density, viscosity, and surface tension, reproduced here in Eqs. (9), (10), and (11), respectively.

$$\rho_\ell = 2729 - 0.73T \tag{9}$$

$$\mu_\ell = 4 \cdot 10^{-5}\exp\left(\frac{4170}{T}\right) \tag{10}$$

$$\sigma_\ell = 0.2726 - 1.014 \cdot 10^{-4}T \tag{11}$$

Note that the FLiNaK density correlation in SAM is almost identical to Eq. (9), with the first parameter being 2729.3 instead of 2729. SAM did not originally have a correlation for surface tension prior to the gas transport model being added. A correlation for surface tension for FLiBe, obtained from [45] was implemented as a first step. For this study, the SaltEquationOfState module in SAM was modified to include a class procedure that returns the surface tension as a function of temperature. The correlation given in Eq. (11) was implemented when the salt type is set to FLiNaK in SAM, with the Souhal correlation still being used for FLiBe salt. The FLiNaK viscosity in SAM is $2.487 \cdot 10^{-5}\exp(4478.62/T)$, which is different from Eq. (10); however, note that

viscosity is not directly used in the calculation of bubble diameter and, while it does impact the calculation of the gas distribution parameter (C_0), this term has no impact for this case. As a reminder, the gas velocity, v_g, is calculated as shown in Eq. (12).

$$v_g = C_0 j + v_{gj} \tag{12}$$

The mixture superficial velocity, j, is taken to be the bulk salt velocity, which is zero for this experiment and thus eliminates the first term. The gas velocity is then simply calculated from the buoyancy term, v_{gj}, which is calculated for churn-turbulent bubbly flow as described in [46] using Eq. (13).

$$v_{gj} = 1.53 \left[\frac{\sigma_\ell g \Delta\rho}{\rho_\ell^2} \right]^{1/4} \tag{13}$$

In this equation, g is the gravitational acceleration and $\Delta\rho$ is the difference between salt and gas velocity. It is worthy to note that this correlation was originally presented in [47]. Harmathy posed a relationship between the terminal velocity of spherical bubbles ($u_{s,\infty}$), terminal velocity of non-spherical bubbles (u_∞), and the drag coefficient of each (spherical ($C_{s,\infty}$) and non-spherical (C_∞^o)), as shown in Eq. (14).

$$\frac{u_\infty}{u_{s,\infty}} = \sqrt{\frac{C_{s,\infty}}{C_\infty^o}} \tag{14}$$

A spherical bubble rising at terminal velocity in an infinite media will have the following velocity.

$$u_{s,\infty} = K \sqrt{\frac{g \Delta\rho d}{\rho_\ell}} \tag{15}$$

Here, d is the bubble diameter and it is assumed that the bubble is the gas phase and is traveling through a liquid media. The K coefficient is a function of bubble geometry and drag coefficient. Hamrathy states that the drag coefficient ($C_{s,\infty}$) for spherical bubbles in infinite media is know to be 0.44 and K is known to be 1.74. He then derives an expression relating the spherical bubble drag coefficient to non-spherical bubble drag coefficients by examining experimental data from a variety of sources. This relationship was derived from experimental data as a function of the Eötvös number (Eo). The Eötvös dimensionless number is a ratio between gravitational forces and surface tension forces and it is commonly used to characterize the shape of a bubble. Note that the Eötvös number of the Chavez study was between 3.65 and 3.69 with a standard deviation of 0.38, which falls within the experimental data. The equation of the relationship is shown in Eq. (16). Note that this equation is valid for Eo less than 13. This equation, when substituted into Eq. (14) along with the known $C_{s,\infty}$ value of 0.44, leads to the v_{gj} expression given in Eq. (13).

$$\frac{C_\infty^o}{C_{s,\infty}} = 1.29 \mathrm{Eo}^{1/3} \tag{16}$$

The SAM model of the Chavez facility was set up using two, vertically stacked *PBOneDFluidComponent* objects; one for the riser section, and one for the quartz window section above. The discretization included 10 nodes in the 105.5 mm riser section and 5 nodes in the 46.5 mm window section. An inlet boundary condition of zero velocity and 600 °C was applied along with an outlet boundary condition of 101.325 kPa and 600 °C. Flow area was calculated from the specified 19.4 mm inner diameter of the test section. The case was run as a 20 second transient with timestep sizes of 0.1 s. A Function was defined to inject the helium as a uniform volumetric source term in the bottom of the riser tube.

The author claimed to use a helium mass injection rate of 4.8×10^{-6} kg s^{-1} in the facility; however, difficulty was encountered in applying this mass flow rate in the SAM model because volume fraction was found to rise to 100%. Considering that the gas velocity was measured to be 0.23 m s^{-1} on average, that the density of helium is about 0.056 kg m $^{-3}$ at the test conditions, and that the tube flow area is 2.96×10^{-4} m^2, the helium gas flow rate for a tube with only helium flowing through the tube would have a mass flow rate of 3.8×10^{-6} kg s^{-1}. Considering this estimated value is less than the value that was provided by the author, it is clear that there is some discrepancy between how the SAM model is being set up and how the experiments were executed.

Despite this issue, it is noted that the two measured values (bubble velocity due to buoyancy and bubble diameter) can fully be calculated by the closure models that were implemented for drift velocity and bubble diameter with minimal input from the SAM base solution. Because the liquid is not moving, the drift velocity model reduces to the buoyancy term in Eq. (13), which is only a function of local fluid properties. The bubble diameter is a function of fluid properties, the bubble velocity, and the Weber number. While the closure models can essentially be evaluated independent of SAM, it is still worthwhile to model the tests in SAM to ensure the models are implemented correctly and to be used as a regression test on the gas model. Because of this, the helium mass flow rate was set to a reduced value of 4×10^{-7} kg s^{-1}, which results in a volume fraction that is between 8–9%. Postprocessors were added for gas velocity, bubble radius, salt velocity, and gas interfacial area at the top of the riser and top of the window components.

To assess the impact of the experimental uncertainties on SAM predictions, a forward propagation UQ study was done using Dakota 6.16 [48]. Latin hypercube sampling (LHS) was used to perform the analysis with three uncertain parameters, which are summarized in Table 5-2. The output parameters are the gas velocity and the bubble diameter.

Table 5-2. Uncertain input parameters used in uncertainty quantification analysis.

Parameter	Nominal	Standard deviation	Distribution
Surface tension multiplier	1.0	2%	Normal
Density multiplier	1.0	2%	Normal
Weber	3.62	13.7%	Normal

Three analyses were performed using sample sizes of 100 and 500 samples. The SAM simulation of the experimental facility runs in about a second or less, so running a large number of samples is not computationally expensive. The results are well converged for the 500 sample case, with the mean and standard deviations of the output parameters varying by less than 1%. Table 5-3 presents the results of the UQ study. The “Nominal” value is the SAM result using the

nominal input values, while the "Mean" is the average of the sample results. The bubble diameter uncertainty is significantly higher than the bubble velocity uncertainty due to the high uncertainty of the input Weber number.

Table 5-3. SAM output parameter uncertainty.

Parameter	Nominal	Mean	Standard deviation
Bubble velocity [mm/s]	262	262	1.85
Bubble diameter [mm]	4.63	4.63	0.640

The partial rank correlation matrix showing the correlation between input and output parameters is shown in . The relationships are fairly straightforward because of the simplicity of the closure models being used. As expected, surface tension is positively correlated with bubble velocity and diameter while density is negatively correlated. Weber number has the most significant impact on bubble diameter. The impact of density and surface tension on bubble diameter is less than the Weber number because the impact of a change to density or surface tension on the bubble diameter will be offset slightly by an opposite impact to the bubble velocity. Weber number has effectively no impact on the bubble velocity.

Table 5-4. Partial rank correlation matrix between input and output parameters

Parameter	Bubble diameter	Bubble velocity
Surface tension multiplier	0.81	0.95
Density multiplier	-0.809	-0.952
Weber number	0.999	0.007

Figure 5-7 shows the SAM-predicted gas velocity compared to the experimental values. The measured values are shown using the blue dataset. Each measurement value has an associated measurement uncertainty provided by the author that is shown with the blue error bars. Note that 14 bubbles were measured, leading to 14 experimental data points, but only one SAM prediction is made because the test conditions do not vary between the different observed bubbles. The SAM result is shown using the solid red horizontal line and the standard deviation from the UQ analysis is shown using the horizontal dashed lines. This first figure presents the actual measured velocities. Figure 5-8 shows the SAM prediction against the results that have been corrected to account for the wall effects. Figure 5-9 shows the SAM-predicted gas bubble diameter compared to the experimental observations.

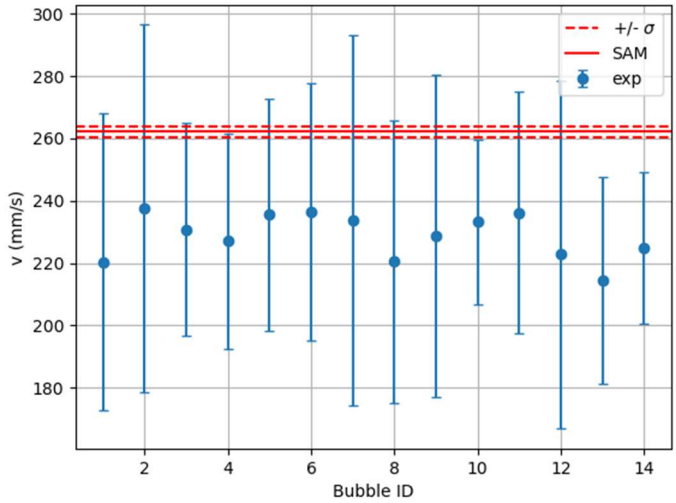

Figure 5-7. SAM prediction (red line) of bubble rise velocity compared to experimental measurements (including wall effects).

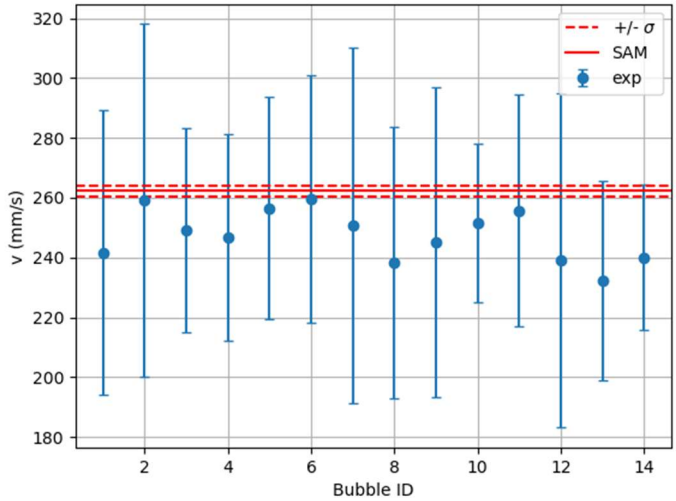

Figure 5-8. SAM prediction (red line) of bubble rise velocity compared to experimental measurements (wall effects corrected).

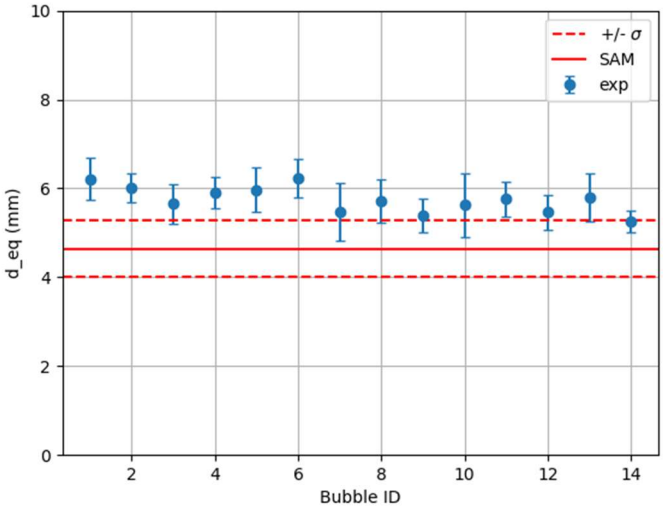

Figure 5-9. SAM prediction (red line) of bubble diameter compared to experimental measurements of equivalent diameter.

A few observations are made related to the comparisons. First, the corrected velocity is the more logical comparison because SAM will not capture wall effects on the bubble velocity; however, it is worthwhile to note that any tube geometries that are modeled by SAM where the tube diameter is close to the bubble diameter will experience a similar modeling error as shown here. Secondly, when looking at the corrected velocities, SAM experiences a slight over-prediction, but all points are captured within the measurement uncertainty. Due to the inverse relationship of relative velocity and bubble diameter, it is expected that SAM will also under-predict the bubble diameter by a similar degree as it over-predicts velocity and this is observed to be the case. Note that the bubble diameter, unlike the velocity, is not corrected for wall effects. It is anticipated that if the experiments were run in a larger tube that is more representative of an infinite media, the under-prediction would be less severe. It is also important to remember that the bubble diameter prediction is highly dependent on the Weber number that is specified in SAM and this value was set from the experimentally observed value, so the diameter cannot be used as a true validation of the SAM model. Finally, as can be gathered from analysis of Hamrathy, the relationship that results in the parameters in the v_{gj} expression certainly has an uncertainty associated with it due to the large scatter of the experimental data, but despite this, the SAM velocity prediction is rather close to the experimental data shown here, which is specific for a salt/helium system.

5.4 Demonstration of gas transport model in MSRE

The MSRE facility was operated at ORNL during the 1960s to test steady state and transient behavior of a liquid salt fueled reactor. A model of the facility was built during a previous NEAMS activity as documented in [49] in order to model one of the reactivity insertion tests that were performed in the facility, which is described in [50]. This existing model was modified for this

work to include the gas transport model and to demonstrate the gas model behavior for more realistic conditions of interest than those that were used for regression and validation testing.

A simplified diagram of the model is shown in Figure 5-10. The core region of the model ("ch2") is modeled using a *PBMoltenSaltChannel* component, which allows for the SAM point-kinetics model to be used for modeling of the core power response to control rod movement, thermal feedback, and delayed neutron effects.

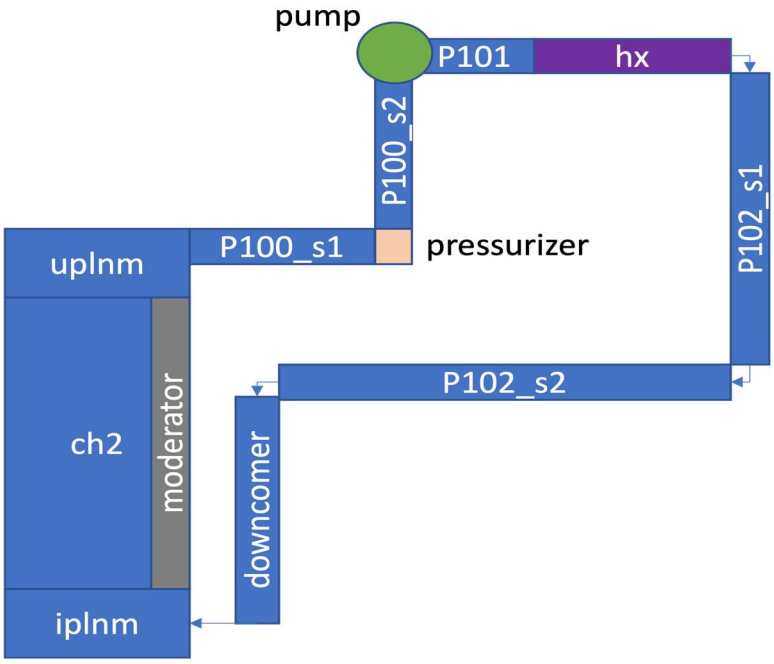

Figure 5-10. Simplified diagram of the SAM MSRE model (not to scale).

Helium injection was made over the first 0.1 m of the "p100_s2" component, which is upstream of the pump. The gas volumetric source term was set to a somewhat arbitrary value of 0.015 kg m^{-3} s^{-1} and the injection time was adjusted so that a void of around 2–5$\cdot 10^{-3}$ was obtained in the majority of the components. Note that the U-233 fueled MSRE runs, which included the reactivity insertion test, had an estimated void of 5–6$\cdot 10^{-3}$ as described in [37]. The MSRE model was originally set up to be run in two steps: the first step was a long transient (2,000 s) that was run to achieve a steady solution and the second step was a reactivity insertion transient that uses the first step solution for its initial conditions. To obtain the desired volume fraction, the gas source was enabled at 5 s and was disabled at 40 s during the first step of the solution, which results in a constant gas mass in the system for the remainder of the transient.

At the completion of the first stage of the transient, the gas model solution was checked and ensured to be steady. The second stage of the transient was then run to perform the reactivity insertion. The power distribution for this case is shown in Figure 5-11.

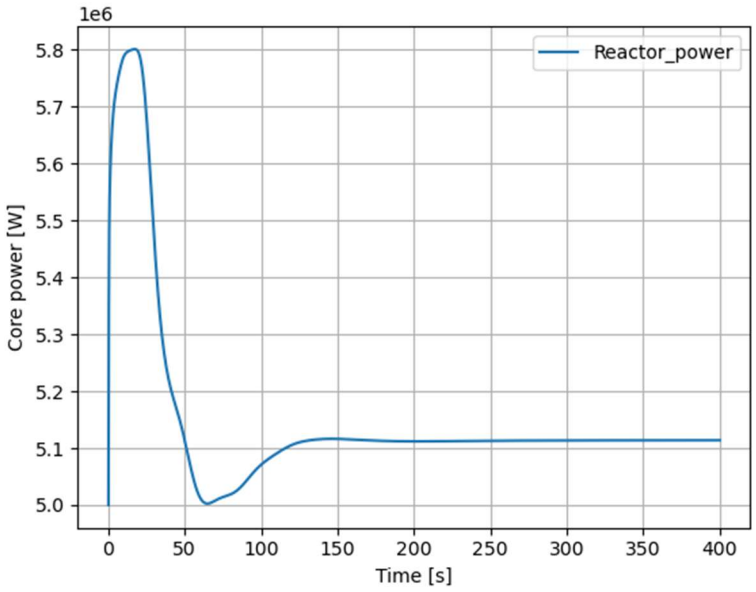

Figure 5-11. Core power response predicted by SAM for the reactivity insertion transient.

During this transient, it was found that the gas volume fraction and interfacial area remained essentially constant at their initial values, as shown in Figure 5-12 and Figure 5-13. The gas model has a minimal response because the resulting change in core temperature is also fairly minimal for this particular transient, as shown in Figure 5-14. This leads to a small change in gas density and, thus, small changes in volume fraction and interfacial area.

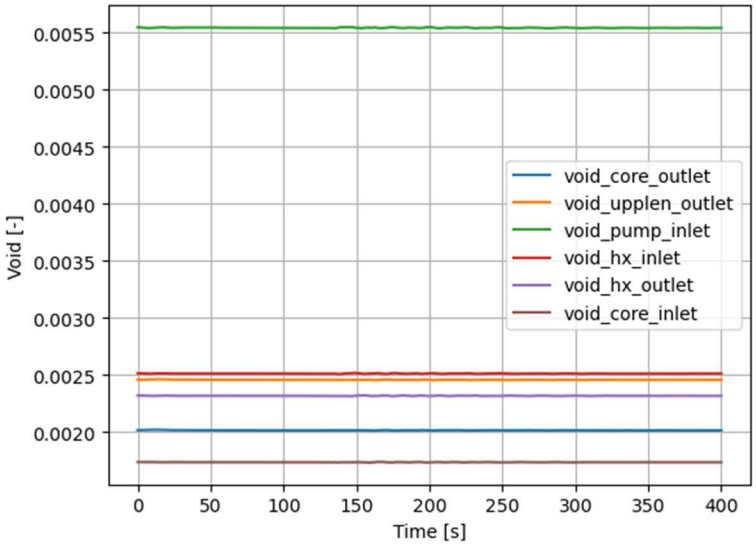

Figure 5-12. Void distribution during the reactivity insertion transient.

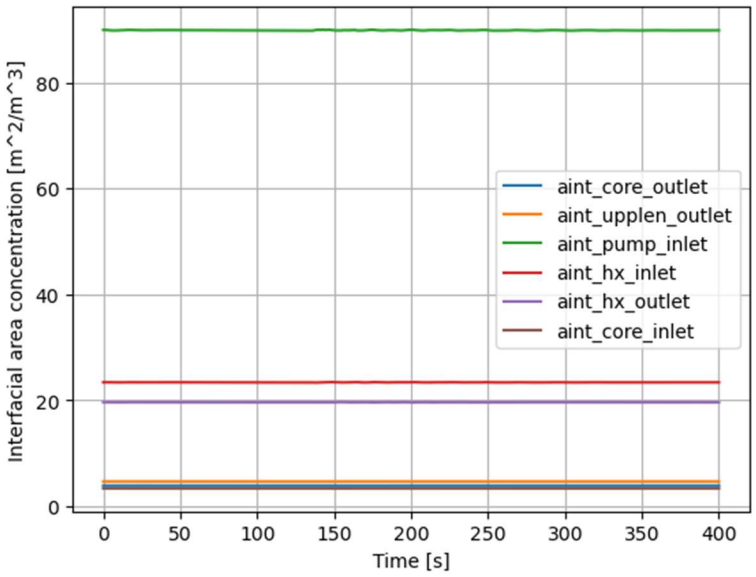

Figure 5-13. Interfacial area distribution during the reactivity insertion transient.

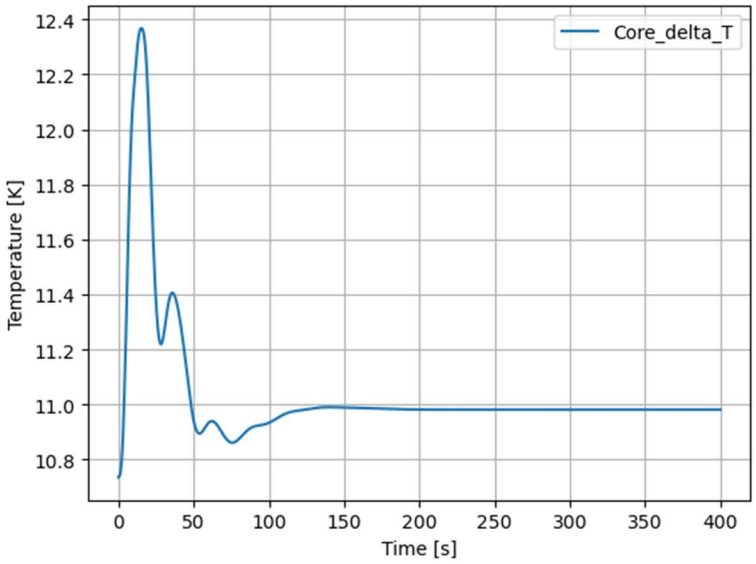

Figure 5-14. Core temperature rise during the reactivity insertion transient.

It is, however, interesting to analyze the steady void distribution, which is quite non-uniform. The behavior is primarily driven by the pressure differences at different points in the model. For example, helium density will be lowest just prior to the pump, leading to a maximum void fraction at this location (green line). Density then increases substantially after the pump, leading to a drop in void at the heat exchanger inlet (red line). The void drops further at the heat exchanger outlet (purple line) due to the drop in temperature and increase in density. As evidenced by the core inlet void, the void fraction is not only influenced by the local pressure. Despite being downstream of the heat exchanger and having a lower pressure and density, the void is actually lower than it was

at the heat exchanger outlet. This is due to the fact that the slip ratio, shown in Figure 5-15, is much higher in the core region due to the addition of buoyancy effects.

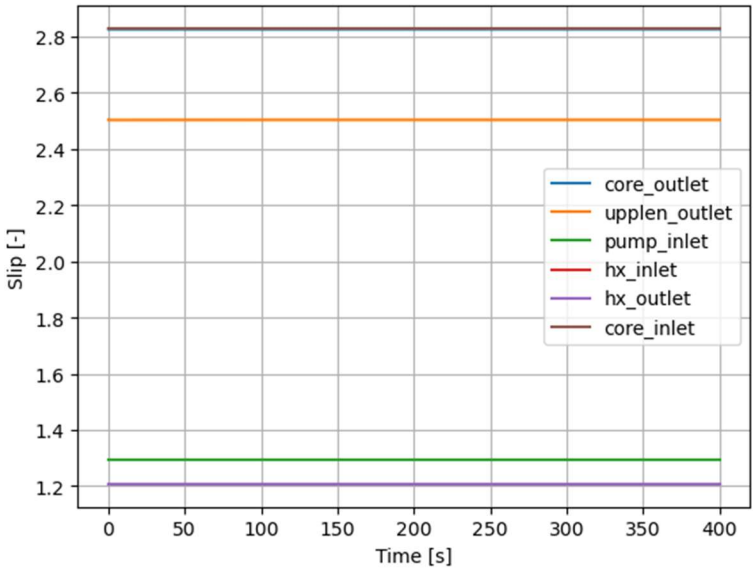

Figure 5-15. Slip ratio at different locations in the MSRE model during the reactivity insertion transient.

To further investigate the gas transport model behavior, an alternative second stage of the transient was developed. Rather than performing a reactivity insertion, a loss-of-flow (LOF) was simulated by applying a forcing function to the pump head. The pump coastdown curve presented in [51] was used to develop this forcing function. The pump coastdown is started at 10 s into the transient. At 15 s, a large negative reactivity insertion is made to shut down the reactor.

The reactor power is shown in Figure 5-16 and the core outlet mass flow rate is shown in Figure 5-17. While core power is rapidly reduced to zero, decay heat continues for the remainder of the transient. This results in a natural circulation flow rate that continues to circulate the salt after the pump coast-down is complete at around 30 s.

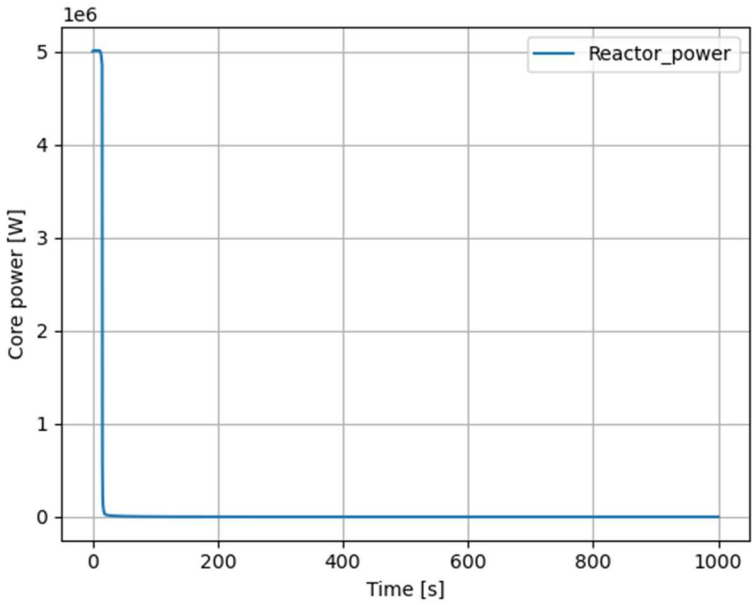

Figure 5-16. Reactor power during the LOF transient.

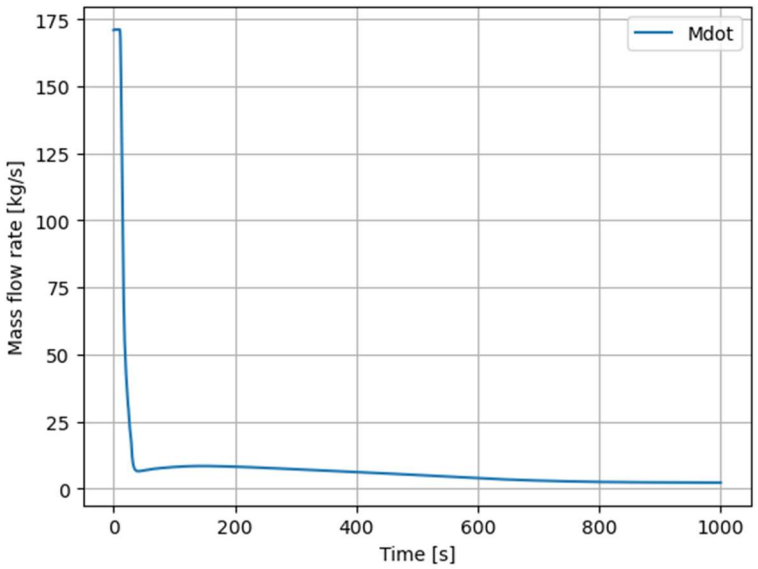

Figure 5-17. Core outlet flow during the LOF transient.

The gas will continue to be transported in the system due to the continual circulation of the salt in the loop, but its behavior is significantly different. The void distribution is shown in Figure 5-18. Void on the core side drops rapidly because the salt velocity slows to being almost stationary, but the gas continues to move at a much faster rate than the salt due to buoyancy, resulting in a high slip. On the other hand, void increases in the other parts of the loop due to pressure and temperature effects on density. This also causes the oscillation in the void in those locations. Also note that the slip will be reduced in the other parts of the loop because there is no buoyancy effect.

As documented in [52], the impact of void on reactivity is essentially insignificant for voids less than 1%, so it is unlikely that this void behavior will impact the core power behavior; however, it does impact the interfacial area, which can impact the fission gas and noble metal transport behavior and fission gas removal after the core is shut down.

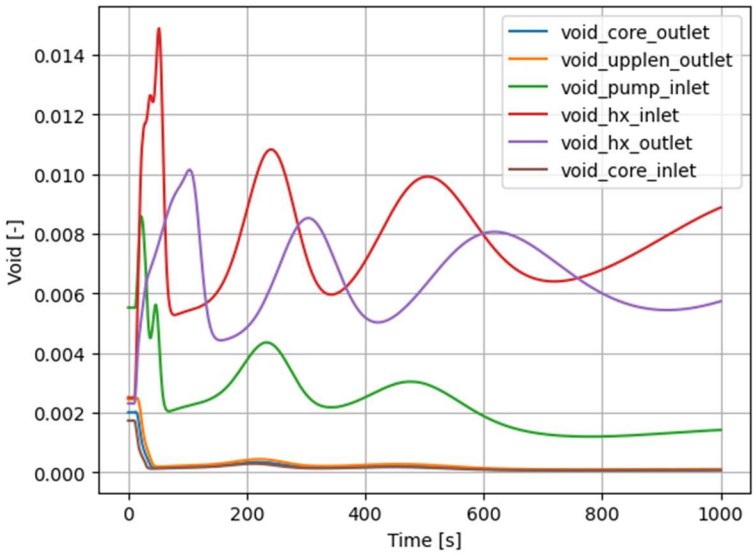

Figure 5-18. Void distribution during the LOF transient.

The interfacial area behavior is shown in Figure 5-19. Note that in this figure, the interfacial area is shown as a ratio of its initial value to indicate how much it changes. In the core region, the interfacial area drops by over 90% and over 70% in the heat exchanger region. The other primary driver of the interfacial area is the calculated bubble radius, which is shown in Figure 5-20. Note that an upper limit of 5 mm is enforced on bubble radius prediction, which is the value predicted in the heat exchanger.

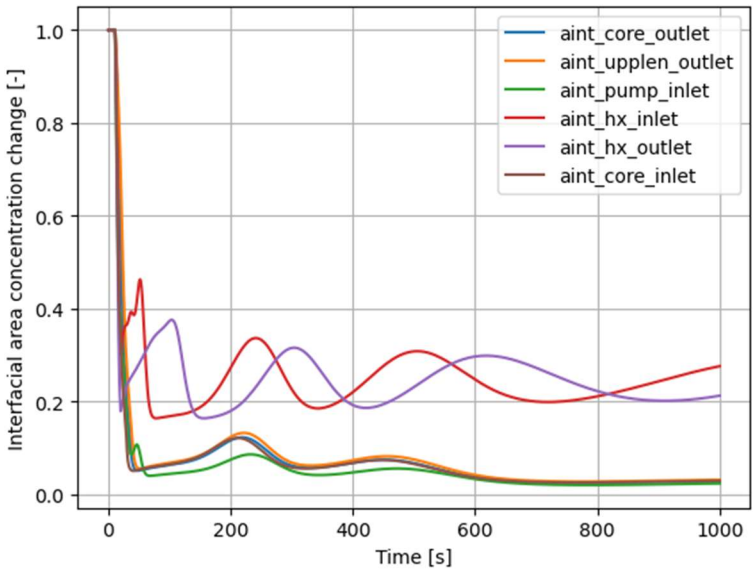

Figure 5-19. Change in interfacial area during the LOF transient.

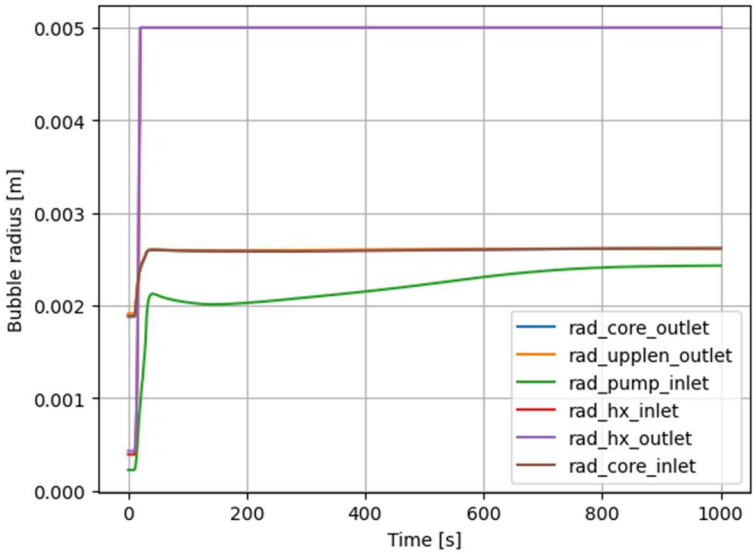

Figure 5-20. Bubble radius distribution during the LOF transient.

The LOF transient is a rather extreme transient for the gas transport because loss of flow means that the gas transport slows substantially and eventually stops. Furthermore, the removal system, which is not modeled in this case, will likely remove much of the gas during the early stage of the transient, resulting even less gas flow. A less extreme case than a total loss of flow is a reduction in flow, which may be performed during operation when reducing plant load. Another SAM simulation was run to simulate this by reducing the flow by about 40% of the nominal value. A negative reactivity insertion was not made for this transient, resulting in an increase in core temperature rise and a slight reduction in core power due to thermal feedback. The core power and temperature rise are shown in Figure 5-21 and Figure 5-22, respectively.

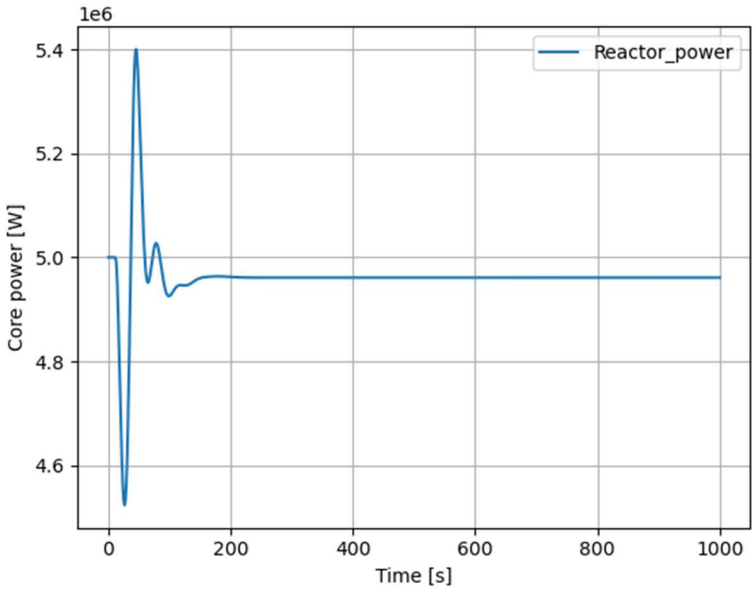

Figure 5-21. Core power when flow is reduced by 40%.

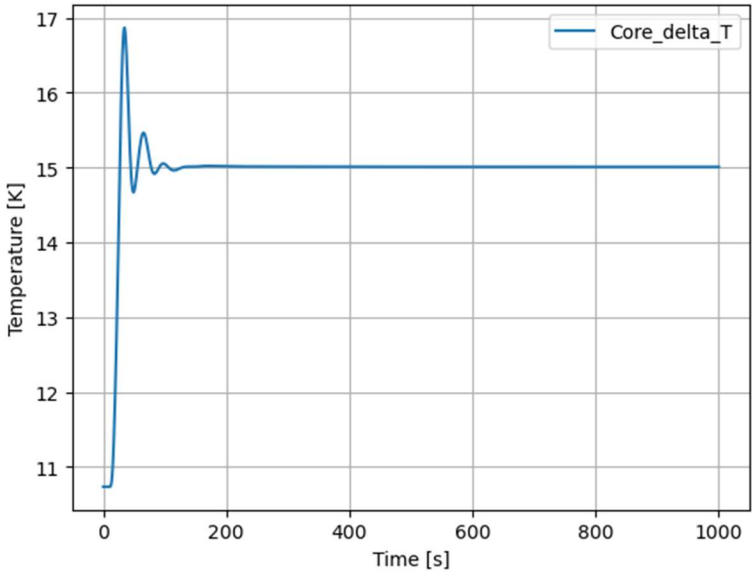

Figure 5-22. Core temperature rise when flow is reduced by 40%.

In this case, the pressure distribution in the loop is not changed as severely, leading to a void distribution that is not as strongly affected, as shown in Figure 5-23. As Figure 5-24 shows, the interfacial area experiences a reduction of about 20% in the core region when flow is reduced. This is less severe than the loss of flow case, but may need to be considered when calculating fission gas and noble metal behavior when plant load changes.

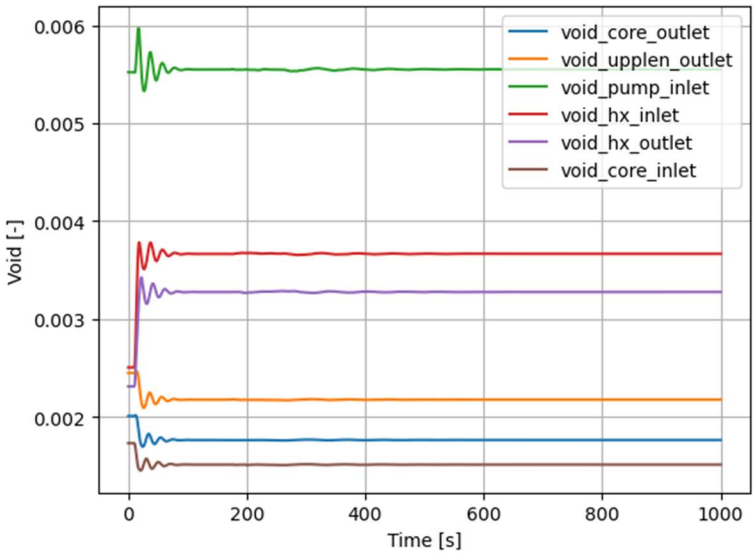

Figure 5-23. Void distribution when flow is reduced by 40%.

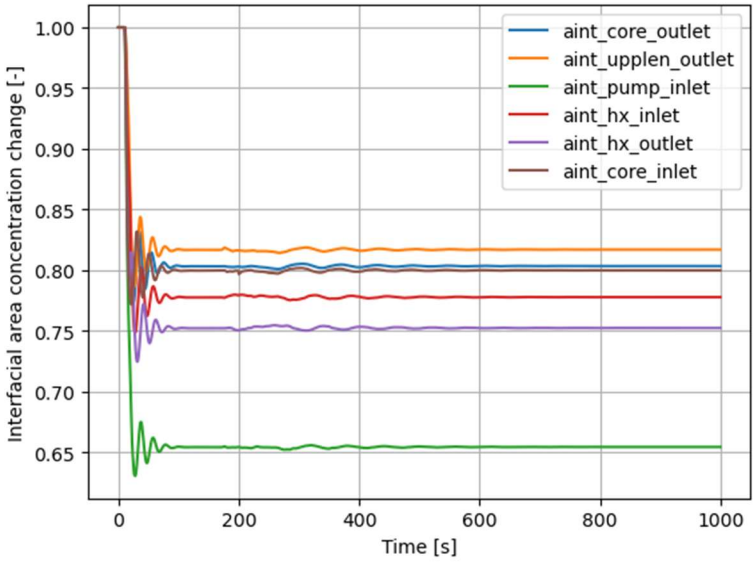

Figure 5-24. Change in interfacial area distribution when flow is reduced by 40%.

5.5 Saline integration

Saline is an application interface (API) to the MSTDB-TP. The application provides access to the database through C/C++, Fortran, and Python interfaces. In general, the user of the API will define a salt, which may be a composition of different salt constituents (e.g., LiF-NaK-KF). Then the API provides procedures that usually take pressure and/or temperature and return properties such as density and viscosity. Saline is a non-MOOSE project, so a few extra steps are needed to use the software in SAM.

Saline is hosted on a public GitLab repository. It was added to the SAM repository as a Git submodule. It uses CMake to manage the build and has a few dependencies of its own. To build it with SAM, the following steps were taken:

1. Saline is configured and built according to its provided documentation.
2. The SAM MakeFile was modified to add the Saline libraries and include files to the MOOSE build system if the Saline subdirectory was found in the SAM directory.
3. A new extension of the PTEquationOfState class called SalineInterface was created, which handles the interface between SAM and Saline.

The SalineInterface class overrides the base class functions for obtaining thermophysical properties, such as rho_from_p_T. If Saline was made available during the build, the appropriate call to Saline will be made and the property will be returned in the correct physical units required by SAM. If Saline was not made available during the build, SAM can still be built; however, if the Saline interface is used, an exception will be raised to the user stating that Saline was not made available. An example of how to use Saline is shown in the code block below.

```
[EOS]
  [./eos]
    type = SalineInterface
    comp_name = "LiF-NaF-KF"
    comp_val = "0.465-0.115-0.42"
    prop_def = "properties.prp"
  [../]
[]
```

An EOS object is entered with the type specified as “SalineInterface”. The three required inputs are the name of the salt composition, the mole percents of the composition, and the database definition. The naming convention requires that salt constituents are separated by a hyphen. The “prop_def” input provides the database that Saline will read. The MSTDB-TP database can be used as an input, but its access is controlled, so it is not provided with the open-source Saline API. The user will have to obtain access to the database separately. Another option is for the user to define their own database of salt properties in the Saline database format. This might be a useful option for users who are working with a proprietary database, which can easily be read by SAM through Saline.

To test the capability, a test that uses a custom database was defined. The test was defined so that it will dump the properties of the salt for two selected temperatures. A separate Python script was developed to evaluate the salt density, viscosity, thermal conductivity, and specific heat at these two selected temperatures, then read the results of the SAM simulation, and compare the expected and actual values of the thermophysical properties.

5.6 Gas transport model conclusions

This section documents the further development and testing of the gas transport model in SAM for support of modeling of MSR designs. The presence of noncondensable gas in MSR systems may be an important modeling consideration due to its impact on fission gas removal and transport

of noble metals throughout the system. Fission products removed through the off-gas system can impact reactivity and can act as an additional point of heat removal. It was shown in this report, through modeling of simple regression problems as well as the MSRE model that gas behavior can be complex.

For steady conditions, the void fraction and interfacial area are found to vary considerably in different locations of the model due to gas velocity changes that arise from void distribution and buoyancy effects. Unlike the fuel salt, local pressure differences also have a significant impact on gas density. While the reactivity insertion transient did not have a large impact on void behavior due to a low impact on gas density, the flow change transients resulted in a more significant change in gas behavior because of the changing pressure distribution and flow field in the system. Ultimately, the gas velocity and interfacial area solution will be used in other models to solve the migration of fission products between the liquid and gas.

In addition to the model demonstration, improvements were made to the model formulation during this project, which were necessary to allow modeling of flow loop systems. The organization of the model in source and the UI was also substantially improved as part of this work. Validation and UQ testing were also performed for the model and demonstrated that the buoyancy term, which was originally developed using non-salt/helium gas experimental data, is able to capture the experimental gas bubble velocity and diameter to within experimental and code uncertainty. Finally, the Saline API to the MSTDB-TP was integrated into SAM, which will allow for better incorporation of new salt thermophysical properties as they become available. Saline also allows for finer control over salt composition and its impact on salt properties.

Future activities to consider include improving on the validation testing, as no validation was performed for the portion of the drift flux model that determines the distribution parameter. Flowing salt experiments with measurement of gas bubbles were not found at the time of this writing, but it is possible to use widely available air/water tests for performing validation. Additionally, simplistic assumptions were made pertaining to the behavior of the drift flux in components where gravity is either in the orthogonal direction or in the flow direction when flow is moving downwards. Other models are available for capturing drift flux in such conditions that can be explored.

The current model targets cases with very low void fraction and thus assumes that the gas phase has no impact on the primary SAM governing equations. If there is a desire to model cases with higher volume fractions, it will be necessary to consider this feedback. While the void fractions of the MSRE were too low to have a significant impact on reactivity, it is possible that other MSR designs may experience higher localized volume fractions, which will require consideration of the void impact on local thermal-hydraulics and neutronics. In this case, it would be prudent to consider implementing the model into Pronghorn as well for detailed core simulations.

Acknowledgement

This work is supported by the U.S. DOE Office of Nuclear Energy's Nuclear Energy Advanced Modeling and Simulation program, under Contract No. DE-AC02-06CH11357. The INL portion of this work is supported by the U.S. Department of Energy, under Contract DE-AC07-05ID14517. The ORNL portion of this work is supported by the U.S. Department of Energy, under Contract DE-AC05-00OR22725.

The development, demonstration, and validation of the integrated MSR modeling capabilities were greatly aided by additional efforts, discussions, and inputs from many colleagues and collaborators, especially Travis Mui, Ling Zou, Tingzhou Fei, and Thanh Hua at Argonne National Laboratory, Elia Merzari at Penn State University, Sebastian Schunert, Alexander Lindsay, and Abdalla Abou-Jaoude at Idaho National Laboratory, Kyoung Lee and Zach Taylor at Oak Ridge National Laboratory.

Reference

[1] E. Merzari and R. Hu, NEAMS Thermal-hydraulics: Overview, Slides presentation at NEAMS Thermal-hydraulics Area Deep Dive, January 2021.

[2] E. Merzari, personal communications, 2020.

[3] R. Hu, D. Nunez, G. Hu, L. Zou, G. Giudicelli, D. Andrs and S. Schunert. "Development of an integrated system- and engineering-scale thermal fluids analysis capability based on SAM and Pronghorn." Argonne National Laboratory, ANL/NSE-21/36, 2021.

[4] R. Hu et al., SAM Theory Manual, Argonne National Laboratory, ANL/NE-17/4 Rev. 1, February 2021.

[5] R. Hu, L. Zou, G. Hu, et al., SAM User's Guide, Argonne National Laboratory, ANL/NSE-19/18 Rev. 1, February 2021.

[6] Hu, R., "Three-Dimensional Flow Model Development for Thermal Mixing and Stratification Modeling in Reactor System Transient Analyses," *Nuclear Engineering and Design*, 345, 209-215 (2019).

[7] Hu R. and Yu Y., "A Computationally Efficient Method for Full-Core Conjugate Heat Transfer Modeling of Sodium Fast Reactors," *Nuclear Engineering and Design*, Vol. 308, 182-193, (2016).

[8] Ling Zou, Guojun Hu and Rui Hu. "Code Enhancement to SAM Multi-scale/multidimensional Heat Transfer Modeling Capabilities." American Nuclear Society Winter Meeting, Chicago, IL US, November 15-19, 2020.

[9] Cody J. Permann, Derek R. Gaston, David Andrs, et al., MOOSE: Enabling massively parallel multiphysics simulation, *SoftwareX*, 11 (2020), 100430.

[10] A. J. Novak, R. W. Carlsen, S. Schunert, P. Balestra, D. Reger, R. N. Slaybaugh & R. C. Martineau (2021) Pronghorn: A Multidimensional Coarse-Mesh Application for Advanced Reactor Thermal Hydraulics, *Nuclear Technology*, 207:7, 1015-1046, DOI: 10.1080/00295450.2020.1825307

[11] P. BALESTRA, D. REGER, and S. SCHUNERT, "Development of PBR Reference Plant for Multiscale, Multiphysics PFLOC/DLOFC Simulations," INL/EXT-20-59507, Idaho National Laboratory (2020).

[12] S. Schunert et al., "Deployment of the Finite Volume Method in Pronghorn for Gas- and Salt-Cooled Pebble-Bed Reactors", INL/EXT-21-63189, Idaho National Laboratory (2021).

[13] John W. Peterson, Alexander D. Lindsay, Fande Kong, Overview of the incompressible Navier–Stokes simulation capabilities in the MOOSE framework, *Advances in Engineering Software*, Volume 119, 2018.

[14] Novak, A.J., Andrs, D., Kelly, J., Slaybaugh, R.N., Martineau, R.C., Schunert, Sebastian, Carlsen, Robert W, and Balestra, Paolo. *Pronghorn Theory Manual*. United States: N. p., 2020. Web. doi:10.2172/1631815.

[15] Software Quality Assurance Plan for MOOSE AND MOOSE-Based Applications, Idaho National Laboratory, PLN-4005.

[16] Wang, Yaqi, Schunert, Sebastian, Ortensi, Javier, Laboure, Vincent, DeHart, Mark, Prince, Zachary, Kong, Fande, Harter, Jackson, Balestra, Paolo, and Gleicher, Frederick. *Rattlesnake: A MOOSE-Based Multiphysics Multischeme Radiation Transport Application*. United States: N. p., 2021.

[17] T.H. Fanning, W.D. Pointer, and J.W. Thomas, "Multi-Resolution Modeling of Subassembly Pin Bundles for Advanced Fast Reactor Safety Simulations," Proceedings of M&C 2009, Saratoga Springs, New York (2009).

[18] J.W. Thomas, T.H. Fanning, R. Vilim, L.L. Briggs, "Validation of the Integration of CFD and SAS4A/SASSYS-1: Analysis of the EBR-II Shutdown Heat Removal Test 17," Proceedings of ICAPP12, Chicao IL, June 24 – 28, (2012).

[19] D.L. Aumiller, E.T. Tomlinson, W.L. Weaver, "An integrated RELAP5-3D and multiphase CFD code system utilizing a semi-implicit coupling technique," Nuclear Engineering and Design, 216, 77–87, (2002).

[20] Davide Bertolotto, Annalisa Manera, Simon Frey, Horst-Michael Prasser, Rakesh Chawl, "Single-phase mixing studies by means of a directly coupled CFD/system-code tool," *Annals of Nuclear Energy*, 36, 310–316, (2009).

[21] R. Hu, J.W. Thomas, and T.H. Fanning, Strategy for Multi-Scale Single-Phase Flow Coupling, Argonne National Laboratory, ANL/NE-13/4, March 2013.

[22] G. Giudicelli, C. Mueller, D. Nunez, T. Hua, J. Fang, A. Abou-Jaoude, Coupled Multiphysics Primary Loop Simulations Of The Mk1-Fhr In The Virtual Test Bed, 2022 American Nuclear Society Winter Meeting, Nov. 2022 (accepted).

[23] Rouch H., Geoffroy O., Rubiolo P., Laureau A., Brovchenko M., Heuer D., Merle-Lucotte, E. (2014). Preliminary thermal–hydraulic core design of the Molten Salt Fast Reactor (MSFR). *Annals of Nuclear Energy*, *64*, 449–456.

[24] A. Abou-Jaoude, G. Giudicelli, D. Gaston, et al., Overview of Advanced Reactor Simulation Capabilities to Support Demonstrations, INL/EXT-21-63162-Rev000, 10.2172/1804658, Idaho National Laboratory and Argonne National Laboratory, 2021

[25] M.Tano, J. Fang, G. Giudicelli, and A. Abou-Jaoude. " Coupled Griffin-Pronghorn-SAM Model of a Molten Salt Fast Reactor System Transient in the Virtual Test Bed", 2022 American Nuclear Society Winter Meeting, Phoenix, AZ, US, November 13-17, 2022.

[26] T.P. Grunloh, A. Manera, A novel multi-scale domain overlapping CFD/STH coupling methodology for multi-dimensional flows relevant to nuclear applications, *Nuclear Engineering and Design,* 318 85–108, (2017).

[27] Huxford, V. Coppo Leite, E. Merzari, L. Zou, V. Petrov, and A. Manera, Development of Innovative Overlapping-Domain Coupling between SAM and NekRS, *Proceedings of NURETH-19*, Brussels, Belgium, March 6 - 11, 2022.

[28] Alexander Lindsay, Logan Harbour, Guillaume Giudicelli, CaseyIcenhour, Fande Kong, Roy Stogner, Brandon Langley, Marco Delchini, Robert Lefebvre, Derek Gaston, and Cody Permann, User-oriented Improvements in the MOOSE Framework in Support of Multiphysics Simulation, Idaho National Laboratory, INL/RPT-22-67144, 2022

[29] Di Ronco A., Cammi A., Lorenzi S. (2020). Preliminary analysis and design of the heat exchangers for the Molten Salt Fast Reactor. *Nuclear Engineering and Technology*, *52(1)*, 51–58.

[30] Pettersen E. E. Coupled multi-physics simulations of the Molten Salt Fast Reactor using coarse-mesh thermal-hydraulics and spatial neutronics. Universit´e Paris-Saclay; 2016.

[31] A. Abou-Jaoude, D. Gaston, G. Giudicelli, B. Feng, and C. Permann. “The Virtual Test Bed Repository: A Library of Multiphysics Reference Reactor Models using NEAMS Tools”, Transactions of the American Nuclear Society, Winter Meeting, 2021.

[32] German, P., Tano, M., Fiorina, C., & Ragusa, J. C. (2022). GeN-ROM—An OpenFOAM®-based multiphysics reduced-order modeling framework for the analysis of Molten Salt Reactors. *Progress in Nuclear Energy*, *146*, 104148.

[33] Abou-Jaoude, A., S. Harper, G. Giudicelli, P. Balestra, S. Schunert, N. Martin, A. Lindsay, M. Tano, and R. Freile. "A workflow leveraging MOOSE transient multiphysics simulations to evaluate the impact of thermophysical property uncertainties on molten-salt reactors." *Annals of Nuclear Energy* 163 (2021): 108546.

[34] M. Tano, R. Freile, J. Fang, A. Abou-Jaoude, D. Shaver, Coupling Coarse-Mesh CFD with Fine-Mesh CFD for Modeling for Molten- Salt Reactors in the Virtual Test Bed (VTB), International Conference of Physics of Reactors (PHYSOR), Pittsburgh, PA, May 15–20, 2022.

[35] J. Engel and R. Steffy. Xenon behavior in the molten salt reactor experiment. Technical Report ORNL-TM-3464, Oak Ridge National Laboratory, 1971.

[36] Kyoung Lee, Benjamin S. Collins, Zack Taylor, and Aaron Graham. Transient convective delayed neutron precursors of u-235 for the molten salt reactor experiment. In Proceedings of Physor 2022, Pittsburgh, PA, USA (May 15–20, 2022), 2022.

[37] R. J. Kedl. The migration of a class of fission products (noble metals) in the molten-salt reactor experiment. Technical Report ORNL-TM-3884, Oak Ridge National Laboratory, 1972.

[38] Federico Caruggi, Antonio Cammi, Eric Cervia, Andrea Di Ronco, and Stefano Lorenzi. Modelling and simulation of the gaseous fission product removal in the molten salt fast

reactor. In The 19th International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-19), 2022.

[39] R. Salko, T. Mui, R. Hu, and E. Merzari. Implementation of a drift-flux model in sam for modeling of passively transported gas in molten salt reactors. Technical Report ORNL/TM-2021/2241, Oak Ridge National Laboratory, 2021.

[40] Shane Henderson, Can Agca, Jake W. McMurray, and Robert Alexander Lefebvre. Saline: An api for thermophysical properties. Technical Report ORNL/TM-2021/2239, Oak Ridge National Laboratory, 2021.

[41] C. Agca, K. E. Johnson, J. W. McMurray, J. A. Yingling, and T. M. Besmann. FY21 status report on the Molten Salt Thermal Properties Database (MSTDB) development. Technical report ORNL/SPR-2021/2102, Oak Ridge National Laboratory, 2021.

[42] Denise E. Chavez, Se Ro Yang, Rodolfo Vaghetto, and Yassin Hassan. Experimental investigation of single helium bubbles rising in flinak molten salt. International Journal of Heat and Fluid Flow, 92, 2021. doi: 10.1016/j.ijheatfluidflow.2021.108875.

[43] D. E. Chavez. Flow visualization of two-phase flow in molten-salt: Helium bubbling in lithium fluoride-sodium fluoride-potassium fluoride (lif-na-kf). Master's thesis, Texas A&M University, 2020. URL http://hdl.handle.net/1969.1/192215.

[44] R. Clift, J.R. Grace, and M.E. Weber. Bubbles, Drops, and Particles. Academic Press, 1978.

[45] M. Sohal, M. Ebner, P. Sabharwall, and P. Sharpe. Engineering database of liquid salt thermophysical and thermochemical properties. Technical Report INL/EXT-10-18297, Idaho National Laboratory, 2010.

[46] N. Zuber and J.A. Findlay. Average volumetric concentration in two-phase flow systems. Journal of Heat Transfer, 87, 1965.

[47] T. Harmathy. Velocity of large drops and bubbles in media of infinite or restricted extent. AIChE Journal, 6, 1960.

[48] Keith R. Dalbey, Michael S. Eldred, Gianluca Geraci, John D. Jakeman, Kathryn A. Maupin, Jason A. Monschke, D. Thomas Seidl, Laura P. Swiler, , Anh Tran, Friedrich Menhorn, and Xiaoshu Zeng. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.16 theory manual. Technical Report SAND2022-6172, Sandia National Laboratories, 2022.

[49] Bo Feng, Shayan Shahbazi, Jun Fang, Ting Fei, and Dillon Shaver. Application of NEAMS codes to capture msr phenomena. Technical Report ANL/NSE-21/48, Argonne National Laboratory, 2021.

[50] Jr. R. C. Steffy. Experimental dynamic analysis of the msre with u-233 fuel. Technical Report ORNL-TM-2997, Oak Ridge National Laboratory, 1970.

[51] B. E. Prince, J. R. Engel, S. J. Ball, P. N. Haubenreich, and T. W. Kerlin. Zero-power physics experiments on the molten-salt reactor experiment. Technical Report ORNL-4233, Oak Ridge National Laboratory, 1968.

[52] Aaron M. Graham, Zack Taylor, Benjamin S. Collins, Robert K. Salko, and Max Poschmann. Multiphysics coupling methods for molten salt reactor modeling and

simulation in vera. Nuclear Science and Engineering, 195(10):1065–1086, 2021. doi: 10.1080/00295639.2021.1901000.

Nuclear Science and Engineering Division
Argonne National Laboratory
9700 South Cass Avenue, Bldg. 208
Argonne, IL 60439

www.anl.gov