INL/RPT-22-69391
ANL/NSE-22/82

Improvements to the Griffin Transport Solvers

September 2022

Zachary M. Prince[1], Yeon Sang Jung[2], Shikhar Kumar[2], Yaqi Wang[1], Joshua T. Hanophy[1], Vincent M. Laboure[1], and Changho Lee[2]

[1]*Nuclear Science and Technology Division, Idaho National Laboratory*
[2]*Nuclear Science and Engineering Division, Argonne National Laboratory*

DISCLAIMER

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

INL/RPT-22-69391
ANL/NSE-22/82

Improvements to the Griffin Transport Solvers

Zachary M. Prince[1], Yeon Sang Jung[2], Shikhar Kumar[2], Yaqi Wang[1], Joshua T. Hanophy[1], Vincent M. Laboure[1], and Changho Lee[2]

[1]Nuclear Science and Technology Division, Idaho National Laboratory
[2]Nuclear Science and Engineering Division, Argonne National Laboratory

September 2022

Idaho National Laboratory
Argonne National Laboratory

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under UChicago Argonne, LLC
Contract DE-AC02-06CH11357
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

Page intentionally left blank

ABSTRACT

Griffin is a Multiphysics Object-Oriented Simulation Environment (MOOSE) based reactor multiphysics analysis application jointly developed by Idaho National Laboratory and Argonne National Laboratory. The code includes a variety of steady-state solvers for fixed-source, k-eigenvalue, adjoint, and subcritical multiplication, as well as transient solvers for point-kinetics, improved quasi-static, and spatial dynamics. This document summarizes the transport solver development efforts pursued during Fiscal Year 2022. We added the multiphysics transient capability for the coarse-mesh finite difference accelerated Richardson iteration for discontinuous finite element method discrete ordinates (DFEM-SN) scheme to support high-order heterogeneous transport simulations. HFEM (hybrid finite element method) - PN (spherical harmonics expansion) was completed and red-black iteration was added for solving the HFEM-PN system with both preconditioned Jacobian-free Newton Krylov and Richardson iteration solvers. The HFEM-PN scheme, as one of the low-order transport schemes, is expected for supporting routine design simulations. Pin power reconstruction capability was also designed and implemented with the Griffin ISOXML module to enhance all the low-order transport solvers for more accurate multiphysics simulations. Numerical results are presented for demonstrating the capabilities and verifying their performance, and future works are discussed.

ACKNOWLEDGEMENTS

This work was funded under the Nuclear Energy Advanced Modeling and Simulation program managed by the Department of Energy Office of Nuclear Energy and was also supported by the United States Department of Energy, Office of Nuclear Energy, under contract DE-AC07-05ID14517 for Department of Energy Idaho Operations Office and under contract DE-AC02-06CH11357 for UChicago Argonne, LLC.
The authors are thankful to the following MOOSE/libMesh developers: Derek Gaston, Logan H. Harbour, Alexander Lindsay, Fande Kong, Guillaume Giudicelli, Roy Stogner, and Jason Miller. Their continuing support in constructive design discussions, supportive framework development, and the Griffin git repository, testing, documentation, distribution, etc. proved to be indispensable.

Page intentionally left blank

CONTENTS

FIGURES

TABLES

ACRONYMS

ABTR	advanced burner test reactor
AMG	algebraic multigrid
ANL	Argonne National Laboratory
CFEM	continuous finite element method
CMFD	coarse-mesh finite difference
CMM	cumulative migration method
DF	discontinuity factor
DFEM	discontinuous finite element method
DNP	delayed neutron precursor
DoF	degree of freedom
FEM	finite element method
FY	Fiscal Year
HFEM	hybrid finite element method
ID	implicit discretization
INL	Idaho National Laboratory
IQS	improved quasi-static method
LWR	light water reactor
MOOSE	Multiphysics Object-Oriented Simulation Environment
MPI	message passing interface
PJFNK	preconditioned Jacobian-free Newton-Krylov
PKE	point kinetics equation
PN	spherical harmonics expansion
RB	red-black
RMS	root-mean-squared
SN	discrete ordinates
SPH	super homogenization
TREAT	transient reactor test facility

Page intentionally left blank

1. INTRODUCTION

Griffin is a Multiphysics Object-Oriented Simulation Environment (MOOSE) based reactor multiphysics analysis application jointly developed by Idaho National Laboratory (INL) and Argonne National Laboratory (ANL). Griffin is made flexible to support the design and analysis of various types of advanced reactors for which multiphysics often dictates the predictability of modeling and simulation efforts. We refer readers to the previous reports we have published [1–3] for more details of the Griffin design and previous developments. The radiation transport module of Griffin includes a variety of steady-state solvers for fixed-source, k-eigenvalue, adjoint, and subcritical multiplication, as well as transient solvers for point-kinetics, improved quasi-static, and spatial dynamics. All these transport solvers are developed in the multiphysics environment. This document summarizes the improvements we made to the transport solvers during Fiscal Year (FY) 2022.

The efforts can be catalogued into two parts: one is in the high-order transport solver with discontinuous finite element method (DFEM) with discrete ordinates (SN) to support high-resolution heterogeneous simulations and the other is in the low-order transport solver with hybrid finite element method (HFEM) with spherical harmonics expansion (PN) to support routine calculations for problems with significant spatial homogenization. Both solvers are made to support multiphysics simulations with the MOOSE framework. During FY 2022, we extended the coarse-mesh finite difference (CMFD) acceleration for the high-order transport solver to support multiphysics transient analysis. The improved quasi-static method (IQS) method is enhanced to work seamlessly with CMFD-accelerated transport calculations for transients. Along with these developments, we made various improvements to the transport sweeper and CMFD solver to further boost the performance of heterogeneous transport calculations. INL led the conceptualization and implementation of these improvements to the DFEM-SN transport solver. ANL exercised the newly developed capabilities in microreactor simulation efforts [3].

In FY 2022, we made HFEM-PN fully functional as well as HFEM with diffusion approximation. We improved the performance of the HFEM-PN solver by adding the red-black (RB) iteration and established the application of the RB iteration to both the Richardson iteration and the preconditioning technique for the preconditioned Jacobian-free Newton-Krylov (PJFNK) solver. To broaden the application of this low-order transport solver, it is crucial to have all low-order solvers support homogenization equivalence with super homogenization (SPH) or discontinuity factors (DFs). Efforts were made to make HFEM-PN work with DFs. INL led the conceptualization and ANL led the conceptualization and implementation of the RB iteration for the HFEM-PN solver.

Another effort was to make low-order solvers generate pin powers, which are crucial for better resolution on the power distribution within the reactor core and for accurate multiphysics simulations. We extended the ISOXML module to support assembly form functions. These form functions are similar to homogenization factors in that they are generated by lattice physics calculations and corrected by Griffin low-order calculations. The workflow has been carefully designed so that the utilization of form functions is transparent to users as homogenization equivalence factors. The implementation allows us to extend for multiphysics simulations with unstructured geometry for advanced reactor designs. INL led the conceptualization and ANL led the conceptualization and implementation of pin power reconstruction.

In Section 2, we give more details on multiphysics transient implementation with DFEM-SN scheme, particularly about the multiphysics iteration and IQS, and how they are incorporated into Richardson iteration accelerated with CMFD. We include a separate section (Section 2.3) to summarize various small, but crucial improvements to the overall performance of the high-order transport solver. In Section 3, we present the equations for HFEM-PN, the RB iteration, and the homogenization equivalence using DFs with HFEM-Diffusion. In Section 4, we present the design and implementation of the pin power reconstruction capability in Griffin. We conducted numerical verification in each of these three sections. We conclude this report and propose further development and improvements for Griffin transport solvers in Section 5.

2. DFEM-SN TRANSPORT SOLVER IMPROVEMENTS

We have extended the capabilities of the Griffin high-order transport solver—namely DFEM with SN—to support a wider range of reactor physics applications. The following subsections detail these, including the extension of the iterative solver to support multiphysics coupling, derivation of CMFD acceleration for transient simulation, and inclusion of a multi-time-scale approach using IQS.

2.1 Multiphysics

The neutronics-only steady-state DFEM-SN solver is described in detail in Reference [3], which includes the sweeper and CMFD implementation. To account for multiphysics evaluation and feedback, an inner iteration was implemented, which couples CMFD and multiphysics. For the purposes of this exposition, we consider the material temperature as a feedback mechanism, governed by fission heat-up and heat conduction.

A tight-coupling approach is used to resolve the coupling between neutronics and heat conduction, whereby separating the physics into different systems and evaluating them individually in an iterative procedure. During this iteration, relevant values are transferred between the systems for the evaluation. For this example, the power density is evaluated in the neutronics systems and transferred to the heat conduction system, and the resulting temperature is transferred to the neutronics systems to recalculate cross sections. Generally, the evaluation of feedback quantities is much less intensive than the neutronics. Therefore, these quantities are evaluated during the DFEM-SN iteration, usually after the CMFD prolongation. This process is shown in Fig. 1. It is possible to specify an inner loop of iteration between CMFD and feedback quantity evaluation; however, a divergence of the outer iteration has been observed when performing this inner iteration [4]. This issue can be resolved by setting the maximum inner loop number to one or two. This is an acceptable solution because the number of CMFD iterations is typically governed by the effectiveness of the diffusion acceleration, and the thermal feedback does not affect this number significantly.

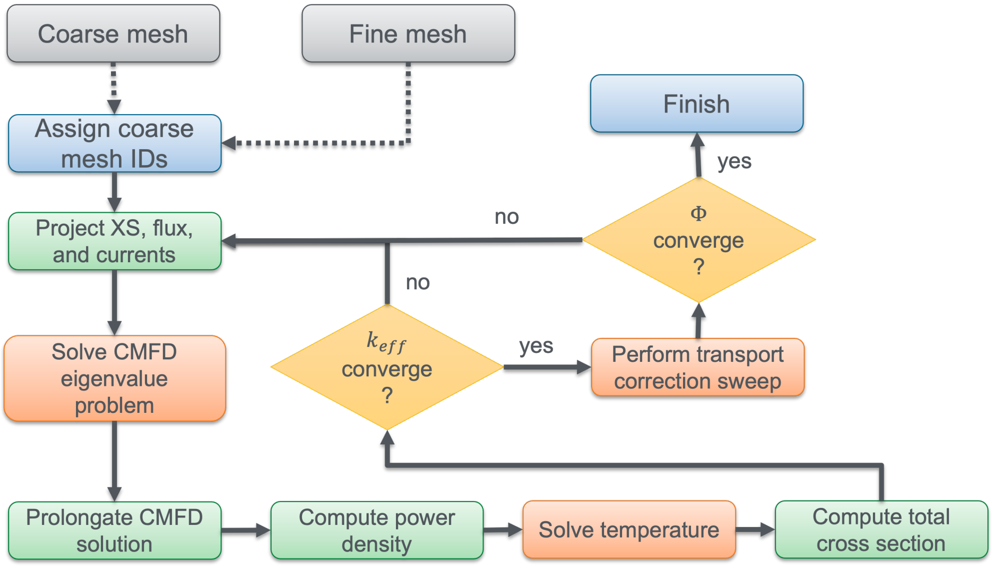


Figure 1: DFEM-SN solution process with CMFD and thermal feedback.

The tightly-coupled approach permits simpler solving algorithms for each of composing physics without worrying about advanced preconditioning techniques that account for the couplings among physics. It also permits some flexibility in how the physics are discretized. For instance, DFEM is preferred for the neutronics, which utilizes the solver algorithms described in Reference [3], and continuous finite element method

(CFEM) is preferred for the parabolic heat-conduction system. Furthermore, the meshes, domain, or even the dimensionality of the systems do not need to match; an example would be coupling one-dimensional thermal fluids. Resolving the discrepancy in function spaces is typically the responsibility of transfers, using techniques like projection or interpolation.

Performing coupled simulations in Griffin utilize the MOOSE `MultiApps-Transfers` infrastructure. The idea behind this infrastructure is that multiple instances of an application can be programmatically instantiated and quantities can be communicated between them. With this, a hierarchy of applications is created with a main application driving the sub-applications and transfer of data. For the neutronics-heat conduction example, two inputs are created: one for the neutronics physics and the other for heat conduction. The neutronics is typically the main application and executes the heat conduction solve and transfers each iteration. For more information regarding `MultiApps` and `Transfers`, see Reference [5].

2.2 Transient

In addition to including multiphysics coupling, we also extended the DFEM-SN solver to support transient simulations. This extension involved changes to sweeper and CMFD, as well as the inclusion of IQS to the iteration method.

2.2.1 Sweeper and Coarse-Mesh Finite Different Transient Modifications

The time-dependent version of the neutron transport equation involves some extra terms involving time integration and delayed neutron precursors. For the sweeper, these terms are handled inherently with the source constructed from the residual evaluation. However, to improve the convergence of the iteration by having the sweeper invert more of the transport operator, the implicit term from the time integration has been added to form a modified collision term. For implicit-Euler time integration, the effective collision term is

$$\Sigma_{\mathrm{t},g}^{\mathrm{eff}} = \Sigma_{\mathrm{t},g} + \frac{1}{v_g \Delta t}. \tag{1}$$

Although, this modification has not been observed to affect convergence significantly for multiphysics transients so far, the modification does not affect sweeper grind time and could prove important for applications with faster transients.

A more significant extension into transient simulation is the modifications in the CMFD implementation. Since the transient neutron transport equation is effectively a fixed-source problem at each time step after time integration, the CMFD acceleration is as well. As such, instead of using the SLEPc eigenvalue solver with the generalized eigenproblem matrices, we build a single matrix with a source term and use the PETSc linear system solver (KSP). Furthermore, additional quantities needed to be projected to formulate a consistent matrix and source, namely the implicit and explicit terms in the time integration and delayed neutron precursor (DNP) treatment. These projections are performed by adding the implicit time term to form an effective total cross section and comparatively an effective fission spectrum for the implicit DNP term. All the explicit terms are summed together to form an effective source term. These effective properties are projected identically to the their eigenvalue counterpart, and the matrix and source are formed naturally.

2.2.2 Improved Quasi-Static Method

IQS [6, 7] is a spatial kinetics time-integration method that involves factorizing the solution into a time-dependent scalar amplitude and a shape defined over the full phase space, including time. It is believed that the time dependence of the shape is weaker than that of the flux itself, thus we can use a much smaller time-step size for amplitude and a relatively large step size for shape to achieve a comparable accuracy in solving directly with the smaller time-step size. We refer to the smaller time step as the micro-step and to the time step for shape evaluations as the macro-step. Because the cost of solving the amplitude equation or point kinetics equation (PKE)—a small ordinary differential equation system—is trivial compared to solving for the shape, the overall computing time is expected to drop significantly. The two-time-scale solution process, a micro-scale for the amplitude and a macro-scale for the shape, is illustrated in Fig. 2. It is noted that this process is quite common in multiphysics simulations where multiple physics could have significantly different time scales.

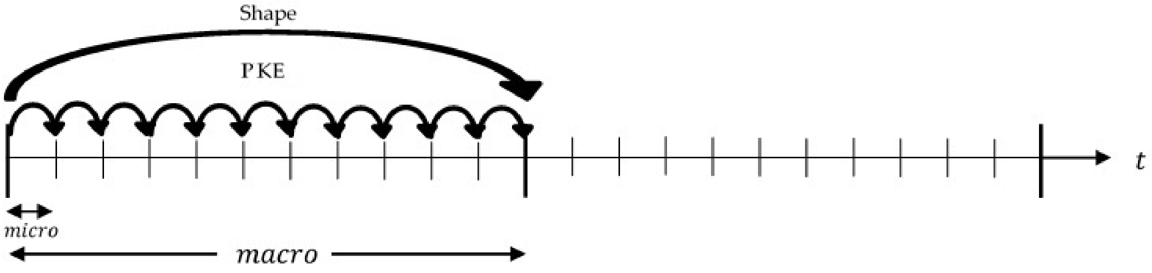
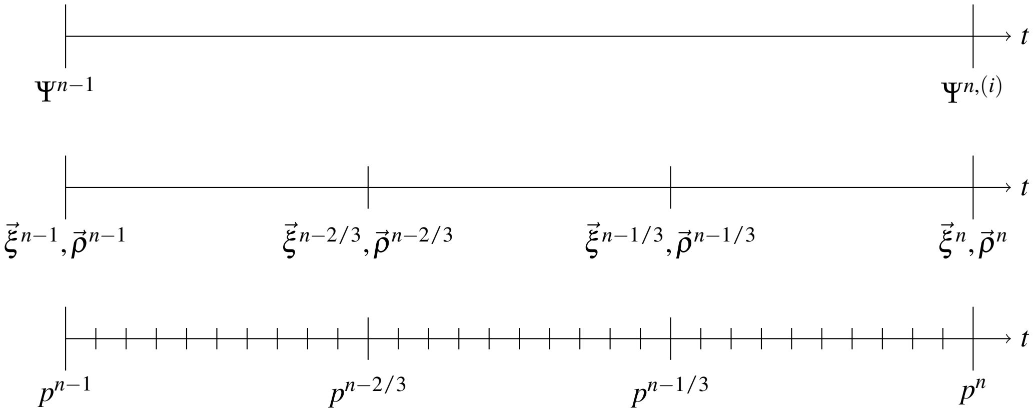


Figure 2: IQS method visualization.

Applying the shape-flux factorization to the neutron transport equation results in two systems of coupled equations. First is the equation for shape, which is largely identical to the original transport equation, except with an extra term dependent on amplitude. The other is the zero-dimensional PKE with coefficients dependent on the spatial-kinetics flux. Since the PKE is a time-only-dependent equation, it is very computationally inexpensive to evaluate relative to any of the other space-dependent operations. So we can solve this equation with extremely small micro-time steps and high-order time integration. By default, Griffin uses a fourth-order Runge-Kutta time integration and step-doubling time adaption with almost machine-precision tolerance. The coupling between the equations for shape and amplitude is resolved within the sweep-CMFD iteration (i.e. by performing one PKE solve per sweep-CMFD iteration).

Along with the different time scales for amplitude and shape, feedback quantities will have a different time scale as well. From experience, feedback mechanisms like heat conduction have a time scale between amplitude and shape [8]. Therefore, in order to minimize the number of transport sweeps, another time scale for multiphysics is introduced, shown in Fig. 3. The middle time scale in this figure represents when feedback quantities and PKE coefficients are updated; it is important to note that the number of updates in this figure is arbitrary.



t
Ψ^{n-1}
$\Psi^{n,(i)}$
t
$\vec{\xi}^{n-1}, \vec{\rho}^{n-1}$
$\vec{\xi}^{n-2/3}, \vec{\rho}^{n-2/3}$
$\vec{\xi}^{n-1/3}, \vec{\rho}^{n-1/3}$
$\vec{\xi}^{n}, \vec{\rho}^{n}$
t
p^{n-1}
$p^{n-2/3}$
$p^{n-1/3}$
p^{n}

Figure 3: Time scales for IQS with multiphysics feedback, where Ψ^n represents the spatial-kinetics angular flux at time step n, $\vec{\xi}$ represents feedback quantities like fuel temperature evaluated through multiphysics coupling, $\vec{\rho}$ represents the PKE coefficients used to evaluate amplitude, and p represents the flux amplitude.

2.3 Other Improvements

2.3.1 Sweeping Efficiency Improvements

Continued improvement of the sweep solver is an ongoing effort, and progress was made towards increasing the sweeper efficiency since results were last reported. Because heterogeneous calculations with transport can commonly contain billions of unknowns, efficiency improvements made to the transport sweeper can have significant impacts on the overall solve time. The scaling test reported in Reference [3] is repeated here to show the impact of code improvements. The specific test we repeated is the one using a cubic domain with a structured mesh of cubic elements. The problem is partitioned among the processors into equal cubic domains. Compared to the result shown previously, the solve time per unknown, or the grind time as it is generally referred to, has improved. This is due to small optimizations, such as a newly implemented custom dense linear solver that does not, for example, waste operations checking if rows need to be swapped since coefficients on diagonals should always be of similar magnitude. Loops were restructured to improve vectorization and data locality.

2.3.2 Coarse-Mesh Finite Difference Improvements and Extensions

The solver performing CMFD acceleration has also been improved over the past year, specifically in its scalability. Previously, all the data from projected material properties and prolongation factors were held on every processor, regardless of fine-mesh or coarse-mesh partitioning. This implementation allowed for faster development and simpler code; however, the memory consumption by the solver and full message passing interface (MPI) reduction communications became significant with relatively larger problems. Consequently, we distributed this data such that only the processors that need a particular piece of the data were given it, all based on the domain decomposition of the coarse mesh. This improved projection and prolongation performance in the solver and reduced memory consumption. Furthermore, in the case where the coarse mesh is identical to the fine mesh, we ensure that the partitionings are identical, further minimizing MPI communication.

In addition to performance improvements through code optimization, we introduced a new CMFD solver option. We have found that algebraic multigrid (AMG) has been a resilient and efficient option for the CMFD solver. However, it was brought to the developers' attention that some problems result in a poorly conditioned CMFD system that AMG is unable to solve. Consequently, we introduced a direct LU solver option that is able to perform in these corner cases. However, this direct solver is not as scalable, memory efficient, or as fast as AMG and should not be utilized ubiquitously. As such, the AMG option is left as the default.

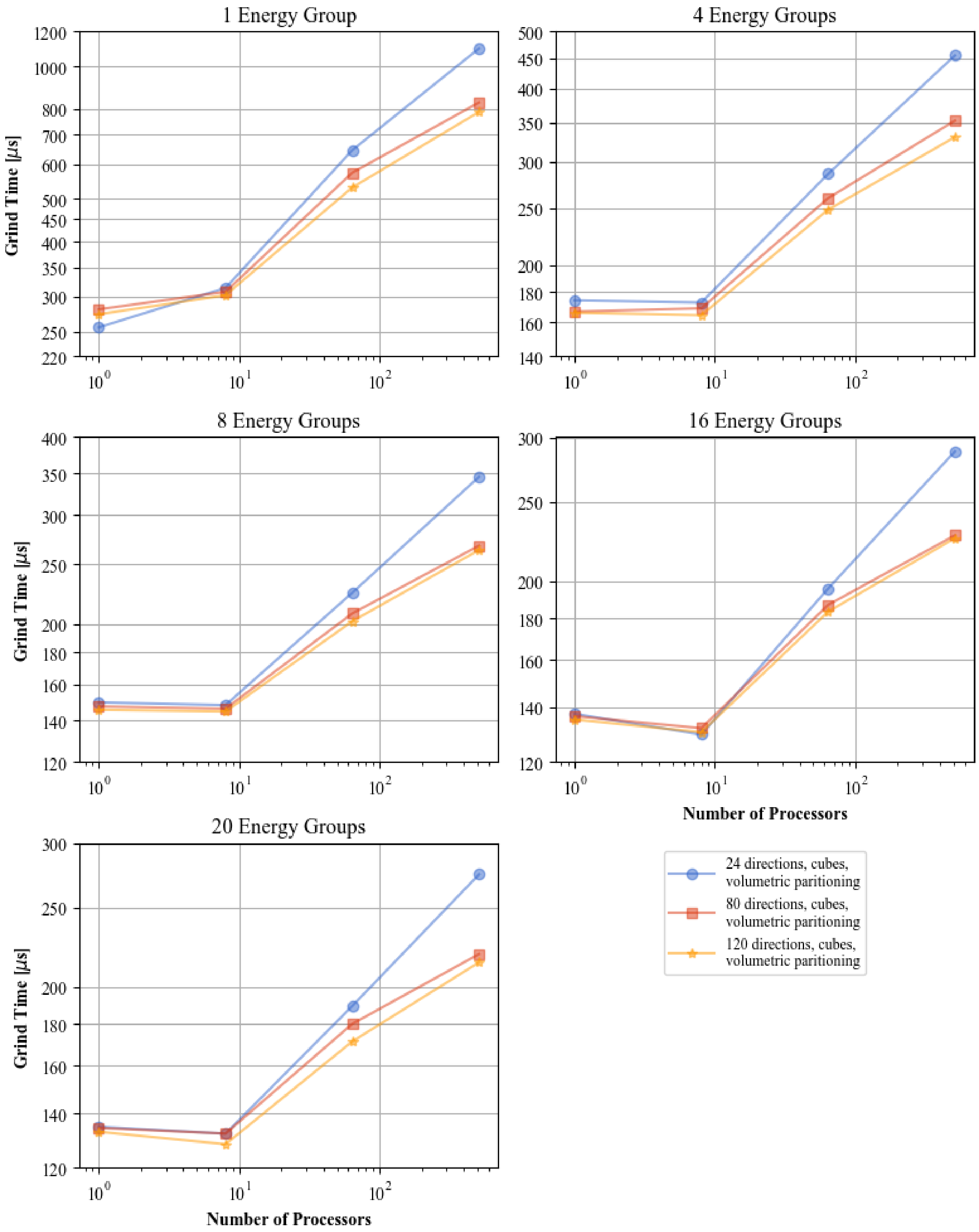


Figure 4: Weak scaling for a structured mesh with structured partitioning, where grind time is the solve time per unknown.

2.4 Results

2.4.1 Modified C5G7 with Multiphysics

This example demonstrates the application of Griffin's DFEM-SN CMFD solver to effectively and efficiently perform heterogeneous, steady-state neutronics calculations with multiphysics feedback. The example utilizes the two-dimensional C5G7 benchmark geometry and neutronics model detailed in Reference [9] and adds thermal feedback. We demonstrated the same multiphysics problem in Reference [10] using the second-order transport formulation.

For reference, Fig. 5 shows the geometry of the problem. The neutronics for this problem is identical to the benchmark except the total cross section is dependent on temperature in the fuel regions,

$$\Sigma_{\mathrm{t},g}(\vec{x},T) = \Sigma_{\mathrm{t},g}^{\mathrm{ref}}(\vec{x}) + \delta\Sigma_{\mathrm{t}}(\vec{x},T), \tag{2}$$

where $\Sigma_{\mathrm{t},g}^{\mathrm{ref}}$ is the cross section from the benchmark and

$$\delta\Sigma_{\mathrm{t},g}(\vec{x},T) = \begin{cases} \Sigma_{\mathrm{a},g}(\vec{x})\gamma_g\left(\sqrt{T}-\sqrt{T^{\mathrm{ref}}}\right), & \vec{x}\in[\text{U02, MOX}] \\ 0, & \text{otherwise} \end{cases}. \tag{3}$$

The temperature is calculated using a heat conduction model,

$$\vec{\nabla}\cdot k\vec{\nabla}T(\vec{x}) = p(\vec{x},\Phi), \quad \vec{x}\in[\text{U02, MOX}], \tag{4}$$

where p is the power density computed with

$$p = \frac{P}{\int_{\mathscr{D}^{\mathrm{fuel}}}\sum_{g=1}^{G}\kappa_g\Sigma_{f,g}\Phi_g\,dx}\sum_{g=1}^{G}\kappa_g\Sigma_{f,g}\Phi_g, \tag{5}$$

where P is the rated power of the core. The boundary condition is a convective heat flux with the water surrounding each fuel pin,

$$-k\vec{\nabla}T(\vec{x})\cdot\vec{n} = h\left(T(\vec{x})-T^{\mathrm{fluid}}\right), \quad \vec{x}\in\Gamma^{\mathrm{pin}}. \tag{6}$$

The thermal feedback properties are quantified below:

$$
\begin{aligned}
P &\equiv 186.868\,[\mathrm{kW/cm}] \\
T^{\mathrm{ref}} &\equiv 300\,[\mathrm{K}] \\
[\gamma_1,...,\gamma_G] &\equiv [0,0,0,0.0005,0.005,0.01,0.0025]\;[\mathrm{K}^{-1/2}] \\
k &\equiv 0.025\,[\mathrm{Wcm}^{-1}\mathrm{K}^{-1}] \\
h &\equiv 2.994\,[\mathrm{Wcm}^{-2}\mathrm{K}^{-1}] \\
T^{\mathrm{fluid}} &\equiv 570[\mathrm{K}]
\end{aligned}
$$

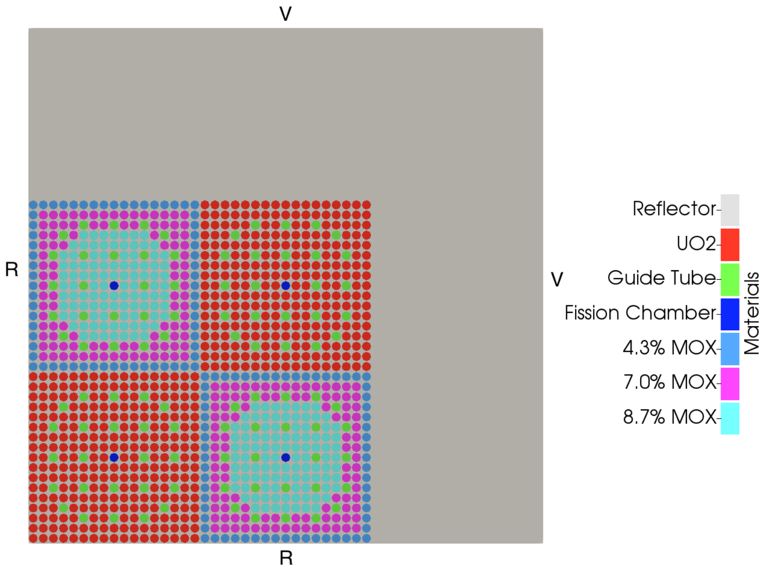
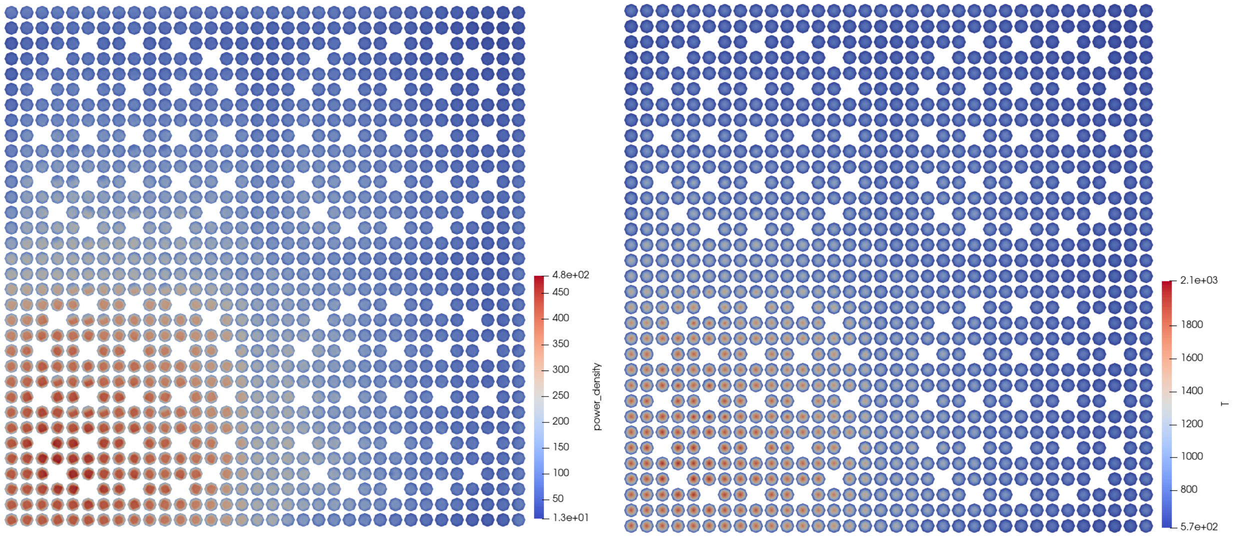


Figure 5: Geometry of the two-dimensional C5G7 benchmark problem. Boundary conditions are vacuum (V) or reflective (R).

Fig. 6 shows the resulting power and temperature profile from a converged feedback simulation, respectively. There are two parameters that we can modify to see what results in the most optimal convergence: `fixed_point_max_its` and `max_inner_its`, which represent the outer and inner iteration loops in Fig. 1, respectively. Table 1 shows the results of different values of these parameters. What these results indicate is that performing more heat conduction–CMFD fixed point iterations between sweeps can destabilize the solve procedure, resulting in divergent iteration, basically the CMFD system becomes ill-conditioned. While increasing the number of inner sweeps within each Richardson iteration can improve performance and stabilize the procedure, but there are diminishing return at larger values. Note that performing four or more iterations between sweeps caused the solve to diverge, no matter the number of inner sweeps performed. Therefore, it is generally recommended to utilize the default `fixed_point_max_its` value of 1.



(a) Power Density (W/cm^3)

(b) Temperature (K)

Figure 6: Power and temperature profiles for C5G7 problem with thermal feedback.

Table 1: Convergence and run-time results for various iteration specifications, run with 96 cores on INL's Sawtooth cluster. When converged, calculations with different parameters result in the same solution within tolerances, as indicated by k_{eff} and $\bar{T}$.

`fixed_point_max_its`	`max_inner_its`	Richardson Iterations	k_{eff}	$\bar{T}$ (K)	Run-time (s)
No feedback	1	17	1.18651	N/A	53.606
No feedback	2	6	1.18650	N/A	48.889
No feedback	4	5	1.18650	N/A	51.982
No feedback	8	4	1.18650	N/A	58.948
1	1	18	1.12941	840.512	63.895
1	2	6	1.12940	840.513	53.159
1	4	5	1.12940	840.512	56.192
1	8	5	1.12940	840.512	76.646
2	1	67	1.12940	840.513	260.852
2	2	15	1.12940	840.512	128.044
2	4	15	1.12940	840.513	158.791
2	8	14	1.12940	840.513	205.47
3	1	Did Not Converge			
3	2	100	1.12821	840.821	875.504
3	4	26	1.12940	840.513	278.614
3	8	100	1.12564	841.971	1524.826

2.4.2 Multiphysics Transient Microreactor

We demonstrate the benefits of our improved methods on the prototypical microreactor presented in Reference [11]. The design is based on the empire reactor [12] with modifications made to ensure a negative overall temperature reactivity coefficient, necessary to simulate realistic transients. This 2-MWth core consists of 18 hexagonal assemblies arranged in two rings and surrounded by 12 control drums contained within a radial reflector. Each assembly is composed of 96 fuel pins (in red in Fig. 7, pictures courtesy of [11]), 60 moderator pins (in blue) and 61 heat pipes (in yellow). Detailed dimensions and material composition can be found in [11]. The cross sections were tabulated with respect to fuel temperature at 800, 1000, 1200 and 1400 K, to moderator/monolith/heat pipe temperature at 800, 1000 and 1200 K, and to reflector temperature at 600 and 1000 K. Leveraging the symmetry of the problem, the DFEM-SN, CMFD, and thermal meshes are limited to 1/12th of the 3D, full-core geometry, shown in Fig. 8, and are generated using the MOOSE reactor module [13].

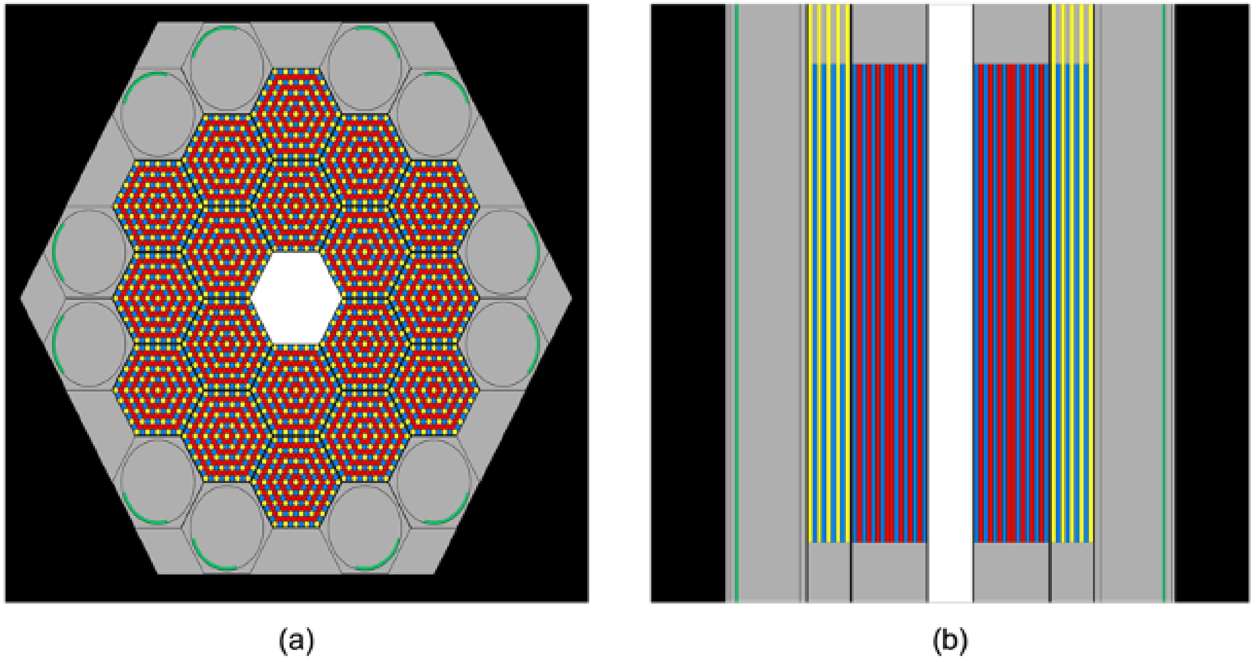
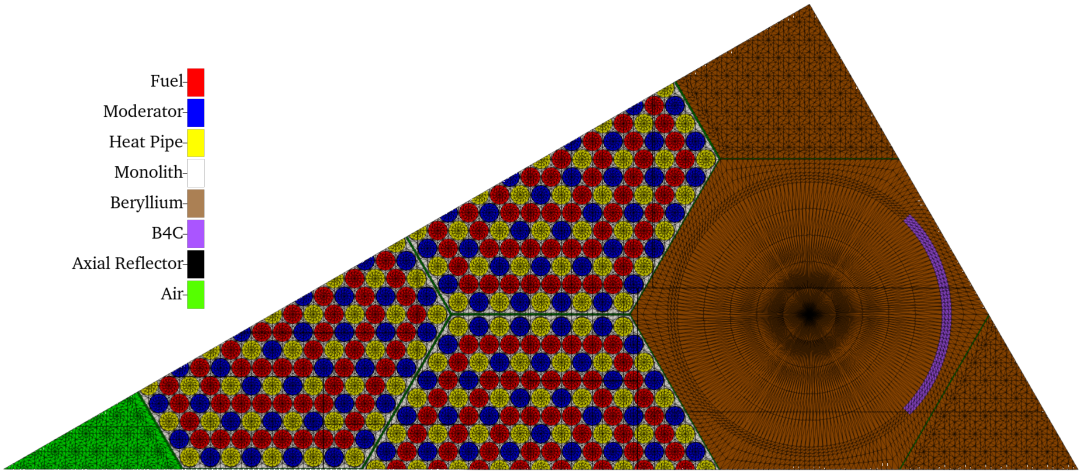
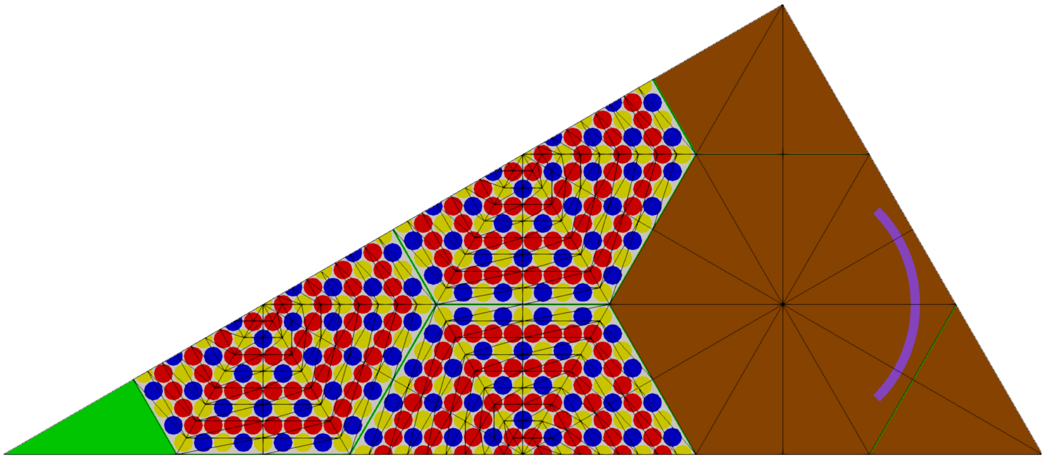


Figure 7: (a) Radial view and (b) axial view of the SiMBA problem Serpent model.

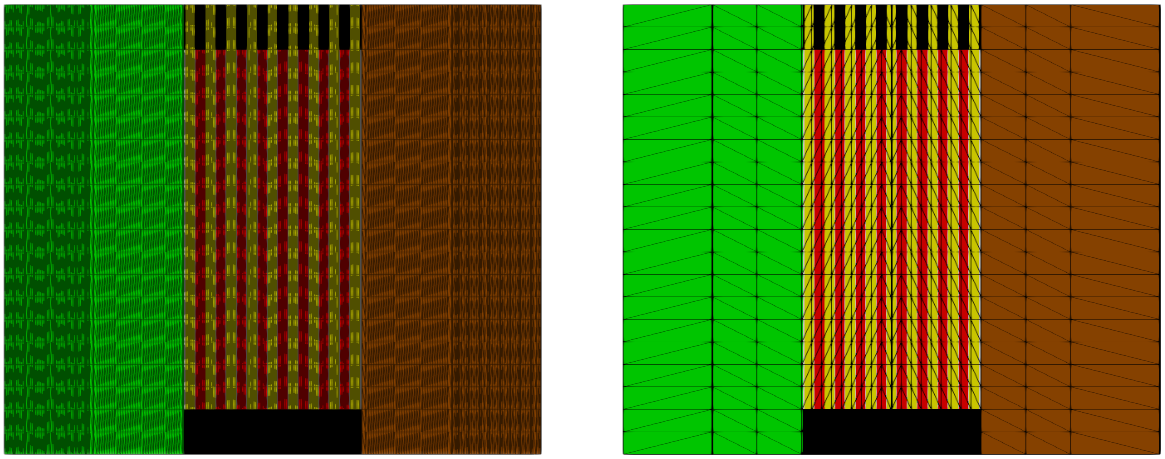
In terms of multiphysics coupling, the same physics as in Reference [11] are solved for except that no hydrogen redistribution is considered and the heat pipes are modeled through boundary conditions using a single representative equivalent heat transfer coefficient—computed in Reference [11] wherein all the heat pipes were individually simulated using the Sockeye flow model—of 300 W/K/m^2.



(a) Radial Fine Mesh



(b) Radial Coarse Mesh



(c) Axial Fine Mesh

(d) Axial Coarse Mesh

Figure 8: 1/12th geometry and mesh for Griffin microreactor model.

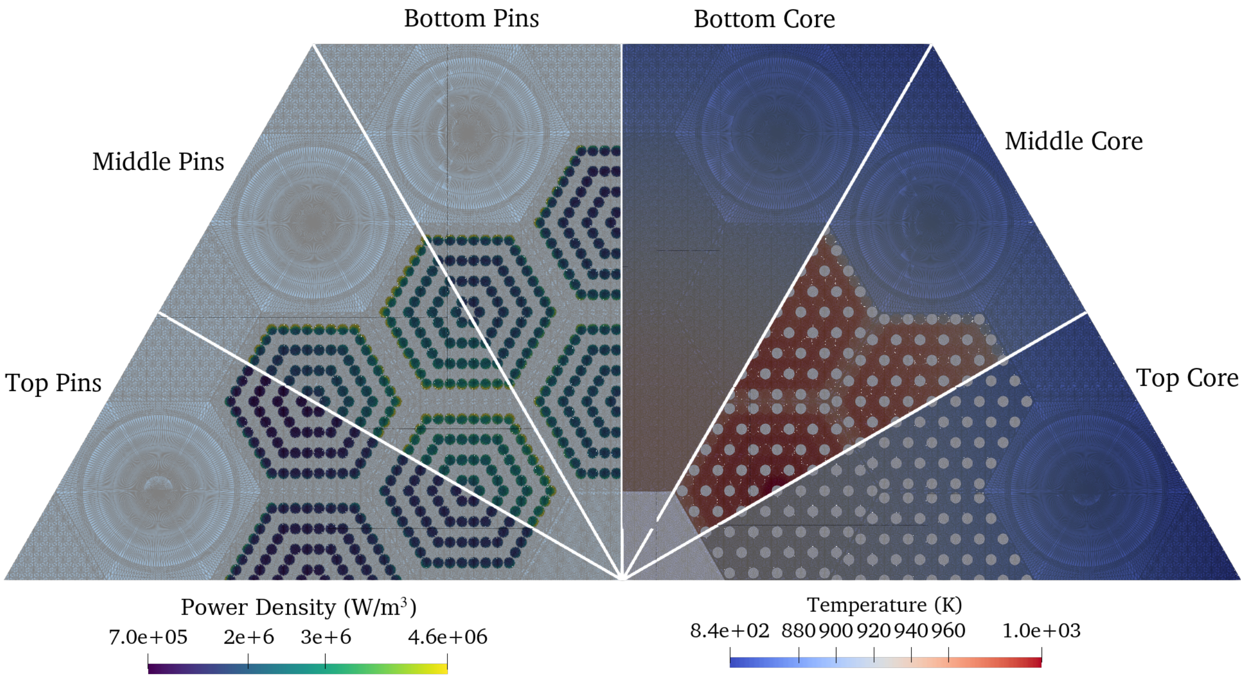


Figure 9: Initial power and temperature profile of microreactor at various axial positions.

To initiate a transient for this microreactor model, we emulated a single heat pipe failure in the 1/12th core geometry (six heat pipes in the full core) by selecting the one at the center of the hottest assembly and setting its heat transfer coefficient to zero. The transient was run for 1,000 seconds. The resulting change in power and temperature distributions is shown in Fig. 10. Five quantities of interest were chosen for the proceeding study: power, maximum fuel temperature, maximum moderator temperature, maximum monolith temperature, and maximum reflector temperature. The change in each of these quantities over the transient is shown in Fig. 11.

For this transient study, four different types of simulations with various time step sizes were performed:

1. ID —non-IQS method, which we denote as the implicit discretization (ID) of the neutron transport equation.
2. IQS—IQS without multiphysics time scaling (i.e., the temperature is evaluated at the same time step as the shape).
3. IQS*N*—IQS with N temperature evaluations between shape evaluations, so the heat conduction time step is $\Delta t(T) = \Delta t(\Psi)/N$. An N of two and five were chosen for this study

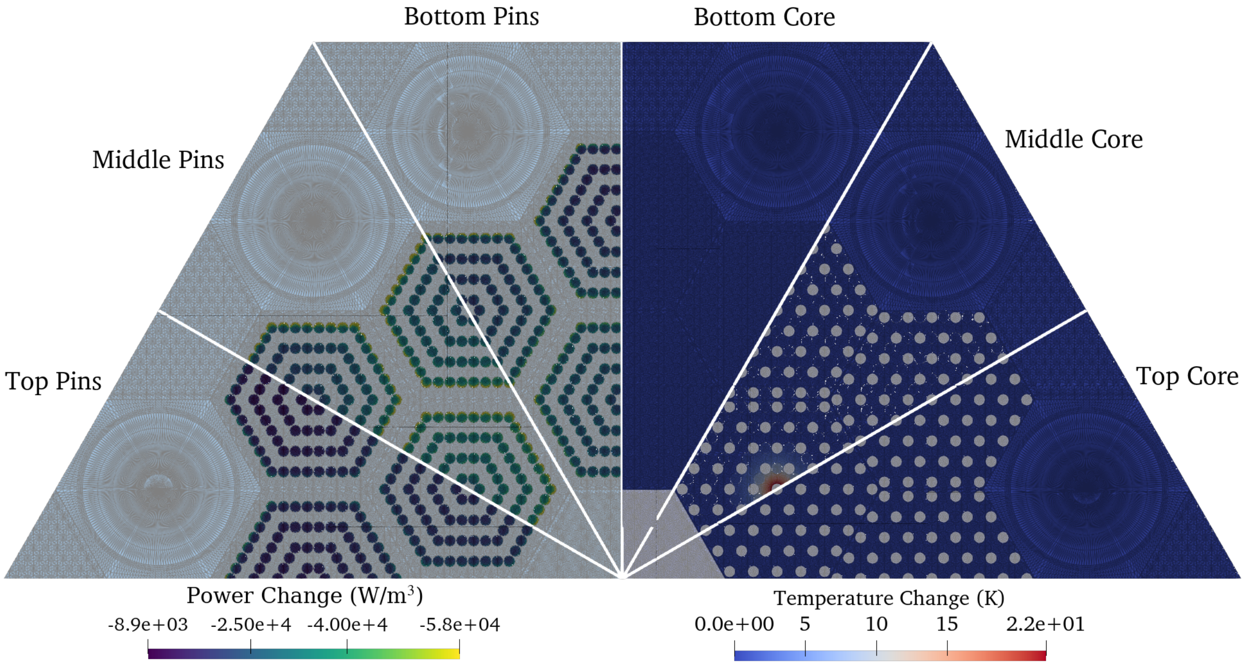


Figure 10: Profile of the change in power and temperature for single heat pipe failure at various axial positions.

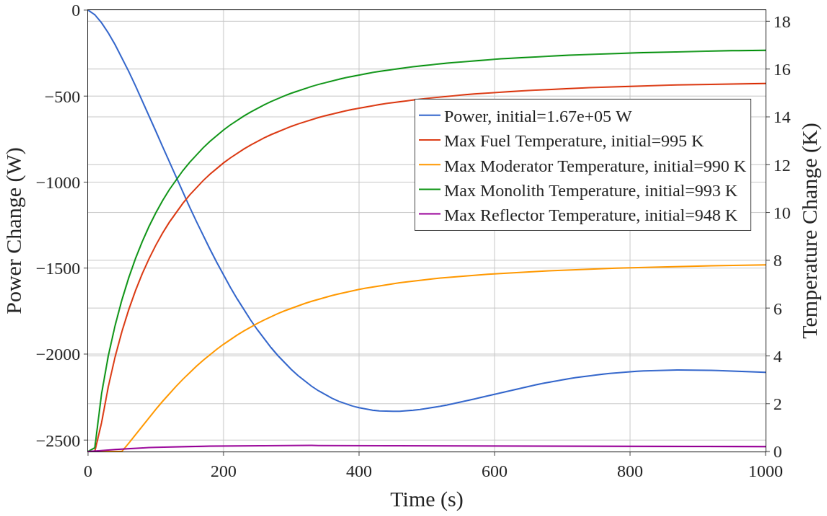


Figure 11: Change in selected quantities over simulated transient for single heat pipe failure.

Simulations with 5, 10, 20, and 50 uniform time steps were performed to show the convergence of the selected quantities. A reference calculation was also performed using ID with a 5-second time step. We used the reference calculation to evaluate a relative error at every time step:

$$\mathrm{Error}(t) \equiv \left| \frac{q(t) - q_{\mathrm{ref}}(t)}{q_{\mathrm{ref}}(t)} \right|, \tag{7}$$

where q represents each quantity and q_{ref} is the quantity evaluated from the reference calculation. A maximum and average error were then calculated over all time steps, which are listed in Table 2.

We observe from these results that the basic IQS implementation, without intermediate temperature evaluation, offers little benefit when compared to ID. This is possibly due to the slow, temperature-driven nature of the transient. IQS has shown more impressive performance with faster pulse transients, that are kinetics driven [8]. Although, it is worth noting that ID with only five time steps had wildly inaccurate results, while IQS was able to capture the transient. IQS with the intermediate temperature evaluations proved to be much more effective, consistently decreasing the error as more evaluations were performed. This result further shows the importance of implementing a multi-time-scale approach for these reactor dynamics models. However, performing more temperature evaluations between shape evaluations has diminishing returns as they are performed at every iteration within the shape time step, making the heat conduction solve the bottleneck of the performance.

Table 2: Error comparison between ID and IQS in selected quantities for single heat pipe failure.

No.	Maximum Relative Error				Average Relative Error			
Steps	ID	IQS	IQS2	IQS5	ID	IQS	IQS2	IQS5
Power								
5	1.617	0.003357	0.003747	0.001672	1.033	0.001142	0.001362	0.0007104
10	0.002069	0.002036	0.001118	0.0005088	0.00082	0.0008139	0.0004126	0.0002145
20	0.001076	0.001015	0.000482	0.0002358	0.0004241	0.0003999	0.000182	9.257e-05
50	0.000378	0.0003485	0.0001188	0.000153	0.0001469	0.000137	4.593e-05	5.909e-05
Max Fuel Temperature								
5	0.09825	0.002896	0.001392	0.0004061	0.08275	0.001097	0.0005054	0.0001769
10	0.002139	0.002118	0.001084	0.0003609	0.0006532	0.0006534	0.0003113	0.0001026
20	0.00132	0.001315	0.0006244	0.0001566	0.0003514	0.0003516	0.0001579	3.912e-05
50	0.0005334	0.0005329	0.000197	4.19e-05	0.0001281	0.0001281	4.32e-05	8.845e-06
Max Moderator Temperature								
5	0.08693	0.001541	0.0008649	0.0003054	0.07047	0.0007936	0.0003998	0.0001539
10	0.0008903	0.0008715	0.0004692	0.0001547	0.0004073	0.0004074	0.0002014	6.922e-05
20	0.0004842	0.000476	0.0002285	5.251e-05	0.0002015	0.000202	9.224e-05	2.365e-05
50	0.0001775	0.0001753	6.031e-05	1.548e-05	6.929e-05	6.941e-05	2.352e-05	4.62e-06
Max Monolith Temperature								
5	0.09246	0.002932	0.001406	0.0004113	0.07796	0.001107	0.0005094	0.0001782
10	0.002189	0.002169	0.001099	0.0003655	0.0006609	0.000661	0.0003141	0.0001034
20	0.001394	0.001389	0.0006438	0.0001601	0.0003572	0.0003574	0.0001598	3.954e-05
50	0.0007223	0.0007218	0.0002491	5.039e-05	0.0001331	0.0001332	4.464e-05	9.059e-06
Max Reflector Temperature								
5	0.03301	8.987e-05	1.438e-05	2.469e-05	0.02245	3.116e-05	4.982e-06	1.202e-05
10	4.437e-05	3.897e-05	1.678e-05	5.341e-06	1.438e-05	1.404e-05	5.347e-06	2.57e-06
20	2.419e-05	2.19e-05	9.756e-06	2.03e-06	7.557e-06	7.48e-06	3.057e-06	1.042e-06
50	9.389e-06	8.951e-06	3.01e-06	1.69e-06	2.629e-06	2.616e-06	8.335e-07	5.932e-07

3. HFEM-PN TRANSPORT SOLVER IMPROVEMENTS

We extended the HFEM solver option in Griffin from the diffusion approximation to PN in order to properly account for the dominant transport effect of fast reactors originating from pronounced anisotropic scattering and long neutron mean-free-path. The RB iteration scheme [14] was also implemented for the HFEM solver to make it a practical solver option for spatially homogenized problems with significantly reduced computational times. In parallel to the HFEM extension work, we also implemented the homogenization equivalence capability in the HFEM-Diffusion solver by employing DFs on the element interfaces. Finally, we verified these new capabilities using the advanced burner test reactor (ABTR) and transient reactor test facility (TREAT) benchmark cases [15, 16].

3.1 Extension to PN Equation

The weak form of the even-parity transport equation for HFEM is:

$$\begin{gathered}(\vec{\Omega}\cdot\vec{\nabla}\Psi^{+*}(\vec{\Omega}),\boldsymbol{G}[\vec{\Omega}\cdot\vec{\nabla}\Psi^{+}(\vec{\Omega})])_{\mathscr{D}}+(\Psi^{+*}(\vec{\Omega}),\boldsymbol{H}[\Psi^{+}(\vec{\Omega})])_{\mathscr{D}}\\ -(\Psi^{-*}(\vec{\Omega}),\vec{\Omega}\cdot\vec{n}[\Psi^{+}(\vec{\Omega})])_{\Gamma}-([\Psi^{+*}(\Omega)],\vec{\Omega}\cdot\vec{n}\Psi^{-}(\vec{\Omega}))_{\Gamma}+(\Psi^{+*},|\vec{\Omega}\cdot\vec{n}|\Psi^{+})_{\partial\mathscr{D}}\\ =(\Psi^{+*}(\vec{\Omega}),S^{+}(\vec{\Omega}))_{\mathscr{D}}+(\vec{\Omega}\cdot\nabla\Psi^{+*}(\vec{\Omega}),\boldsymbol{G}[S^{-}(\vec{\Omega})])_{\mathscr{D}}.\end{gathered} \tag{8}$$

where Ψ^{+} and Ψ^{+*} are the even angular fluxes and its test functions defined on the solution domain $\mathscr{D}$ and the boundary $\partial\mathscr{D}$, Ψ^{-} and Ψ^{-*} are the odd angular fluxes and its test functions defined on element interfaces Γ and $\boldsymbol{G}$, $\boldsymbol{H}$ operators are corresponding to the streaming and reactions terms of transport equation, and S^{+} and S^{-} are external even and odd parity sources, respectively. After the PN expansions for even and odd parity angular fluxes are plugged in, the HFEM weak-form for the PN equation can be derived. Additionally, the *Lagrange multiplier* terms acting on the element interfaces are derived in such a way to yield the currents preserving the neutron balance of each even-moment term. The details of the HFEM-PN weak form can be found in Reference [17].

Based on the MOOSE framework, the individual components of the weak form for the PN equations were implemented as kernels and boundary conditions that evaluate the residual and Jacobian terms accordingly. The volumetric and surface variables used in the HFEM kernels are defined as elemental variable types that represent solutions over individual mesh elements but not on mesh vertices and support higher-order representations of intra-nodal solutions. The polynomial expansion order for surface elements should be lower by at least two degrees than the one for volumetric elements, because of the HFEM formulation rank condition [18]. The developed kernels were assembled in the Griffin transport system framework to properly coordinate the calculation flows of both diffusion and PN. In order to explicitly account for cross-group anisotopric scattering with multiple energy groups, the volumetric odd-parity flux moments are required to determine the odd-parity source term. An additional process was implemented in the PN option to evaluate the volumetric odd-parity flux moments from the even-parity flux moment solutions based on the parity differential equations. Thus, the explicit treatment of anisotropic scattering is available in the HFEM-PN solver, which is essential for advanced reactor analysis.

The linear system is a saddle point problem due to the presence of *Lagrange multiplier* coupling across the element interfaces. As a temporary solution, the system was directly solved by the SuperLU-DIST package with LU decomposition linked through PETSc [19]. The same approach was used in the diffusion solver option developed in FY-21 [3]. We later extended it with the PJFNK solver by having the preconditioning matrix only contain the diagonal parts of the even parity and have an identity matrix for the odd parity in order to avoid using a direct solver. In this FY, we implemented the classic RB iteration algorithm [14] for solving the response matrix-based linear system yielded from the HFEM-PN and HFEM-Diffusion weak forms, which is addressed in Section 3.2.

3.2 Red-Black Iteration Scheme

The discretized weak form of HFEM-PN equations can be represented in a matrix-vector operation form as:

$$\begin{bmatrix} \mathbf{A} & \mathbf{E} \\ \mathbf{E}^T & 0 \end{bmatrix} \begin{bmatrix} \psi_\mathrm{e} \\ \psi_\mathrm{o} \end{bmatrix} = \begin{bmatrix} \mathbf{s_v} \\ 0 \end{bmatrix}, \tag{9}$$

where $\mathbf{A}$ corresponds to the volumetric kernels (streaming and reaction) and the boundary condition acting on the even-parity moments, $\mathbf{E}$ corresponds to the *Lagrange multiplier* kernels applied on both the interfaces to the even- and odd-parity moments, and $\mathbf{s_v}$ is the volumetric source term, including fixed sources and scattering sources from the even- and odd-parity moments. The even-parity moment variable appearing in Eq. (9) can be eliminated. Then, it can be recast into the following form having the odd-parity moment variable as the only unknown to be determined:

$$\mathbf{E}^T\mathbf{A}^{-1}\mathbf{E}\psi_{\text{o}} = \mathbf{E}^T\mathbf{A}^{-1}\mathbf{s_v}. \tag{10}$$

Once the odd-parity moment is determined, the even-parity moment is automatically obtained by using:

$$\psi_{\text{e}} = \mathbf{A}^{-1}(\mathbf{s}_{\text{v}} - \mathbf{E}\psi_{\text{o}}). \tag{11}$$

This suggests that the HFEM-PN equation solutions can be effectively obtained by only solving the equations for odd-parity moment variable. In order to solve Eq. (10) in an iterative manner, the partial currents are defined by combining the even- and odd-parity moments, and the equations are rewritten as the response matrix form in terms of partial current variables as:

$$\mathbf{j} = \mathbf{R}\mathbf{j} + \mathbf{B}\mathbf{s}_{\text{v}}, \tag{12}$$

where **R** and **B** are the response matrices applied to the partial current **j** and volumetric source $\mathbf{S}_{\text{v}}$, which are the input of response matrix operation. Eq. (12) provides a way to update the partial currents. The RB iteration scheme is employed in order to efficiently resolve the partial current couplings appearing in Eq. (12) through the partial current iterations. The details of derivation and implementation of the RB iteration scheme can be found in Reference [17, 20].

The RB iteration scheme is integrated into Griffin as:

- a preconditioner of the existing PJFNK solver option
- an independent solver option, which solves the linear system by sweeping the domain.

In the preconditioner approach, the underlining preconditioning matrix is essentially the same as the system matrix. The iterative RB sweeping can be considered an acceleration because it possibly damps modes of different frequencies with different speeds. Alternatively, the independent solver option is implemented in the framework as the Richardson iteration executioner. In each Richardson iteration, the RB iteration solver solves a fixed-source problem by setting the residual vector as the fixed source term. For practically solving the multigroup problems, the group sweeping is also introduced which allows it to update scattering source terms during the Richardson iterations. Both solve options support multiphysics simulations as seamlessly in Griffin as other transport solvers.

3.3 Discontinuity Factors

The weak form of the HFEM-Diffusion equation for a one-group diffusion equation on internal elements is given as:

$$(\vec{\nabla}\phi^*, D\vec{\nabla}\phi)_{\mathscr{D}} + (\phi^*, \Sigma_\mathrm{a}\phi)_{\mathscr{D}} - (\chi^*, [\phi])_\Gamma - ([\phi^*], \chi)_\Gamma = (\phi^*, q)_{\mathscr{D}}, \tag{13}$$

where ϕ is the scalar flux and χ is the *Lagrange multiplier* that is equivalent to the surface net current. Here, the third term to weakly imposed on the interface flux continuity. Thus, we can introduce a DF in this term as:

$$(\chi^*, [f\phi])_\Gamma = (\chi^*, f^+\phi^+ - f^-\phi^-)_\Gamma, \tag{14}$$

where f is the DF defined on each interface of individual element. By doing so, the flux discontinuity on the interface can be properly incorporated in the HFEM-Diffusion equation while the single current value preserving the neutron balance of neighboring elements can be defined. For a vacuum boundary, the additional variable for the boundary interface is defined by projecting the volumetric flux as:

$$(\chi^*, f\hat{\phi} - \phi)_{\partial\mathscr{D}}, \tag{15}$$

where $\hat{\phi}$ is the interface flux and f is the DF. Then, the interface flux is explicitly used in the vacuum boundary condition treatment. Note that we do not need to introduce the DFs on the reflective boundaries.

When the reference partial current values on the interface is known, the DF can be explicitly obtained by using the following relations:

$$f_\Gamma^\pm = \frac{J_{ref}^\pm \mp J/2}{\phi/4}. \tag{16}$$

For the vacuum boundaries, the above relation can be further simplified as:

$$f_{\partial\mathscr{D}}^+ = \frac{2J_{ref}^+}{\hat{\phi}}. \tag{17}$$

It is noted that here an interface can be a collection of element interfaces, between a pair of equivalence zones.

In order to illustrate the DF implementation on the HFEM-Diffusion solver options, the two-element problems was solved for the given DFs applied on the element interface. As shown in Fig. 12, the scalar fluxes are discontinuous on the interface as holding the DF relation given in Eq. (14) while the scalar flux is continuous on the interface without the DFs. We also confirmed that solutions with and without DFs are strictly satisfied the local neutron balance equation.

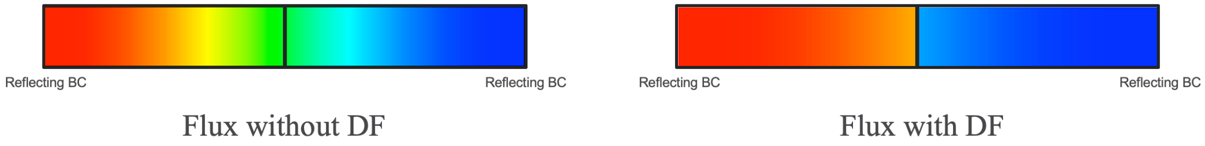


Figure 12: Illustration of DF on HFEM-Diffusion solution.

We implemented a similar DF implementation as the DFEM-Diffusion scheme [16]. DF are evaluated in the same PJFNK approach presented in [16]. Since it is not currently incorporated with the RB iteration scheme, the DF scheme is only compatible with the direct inversion mode of the HFEM-Diffusion solver option. The RB iteration scheme will be updated to allow the usage of DF in the next FY.

3.4 Verification Tests

3.4.1 HFEM-PN Extension

Using the HFEM-PN solver option, the 2D ABTR problem was solved using various orders of PN (P1–P5) and anisotropic scattering (P0–P5). In the tests, the 2D core model derived from the 3D ABTR model was used, which is composed of homogenized hexagonal assemblies in a nine-ring lattice configuration. Each assembly is divided into two quadrilateral elements since libMesh [21], the underlining mesh framework used by Griffin, does not support the hexagonal element type yet. The reference calculations were performed using the CFEM-PN solver of Griffin using the refined mesh structure where each hexagonal assembly was further divided into 128 quadrilateral elements to obtain the converged eigenvalues. The results are gathered in Table 3.

For all test cases varying PN and anisotropic scattering orders, the obtained eigenvalues converged to the corresponding reference values with only a few pcm differences as the basis function orders of local elements increases. When the lowest order of basis function was used, the eigenvalue errors increased, with the higher order of PN and anisotropic scattering, respectively, while the errors were almost constant in other cases. This is because a lower order basis function could not capture the drastic intra-nodal variations of even- and odd-parity flux moments near material interfaces. To verify the spatial distributions of HFEM-PN solutions, the intra-nodal distributions of even-parity higher order moments were compared with the reference CFEM-PN distributions. As shown in Fig. 13, the higher-order moment distributions obtained with the HFEM and CFEM options matched. These results indicate that the higher-order moments having complex shapes could be well captured using the higher-order representations with a coarse mesh structure. From the 2D ABTR calculation, we confirmed that the HFEM-PN solver was correctly implemented in Griffin, showing a good convergence characteristic on a spatially homogenized model as refining the PN and basis function orders.

Table 3: HFEM-PN eigenvalue solutions with a change of PN, scattering and expansion orders for 2D ABTR problem

PN Order	Scattering Order	Eigenvalue (Δk, pcm)			Ref. Eigenvalue
		Basis Function Order (Volume / Surface)			
		2 / 0	3 / 1	4 / 2	
1	0	1.29335 (153)	1.29209 (26)	1.29181 (-1)	1.29182
	1	1.20800 (119)	1.20701 (20)	1.20679 (-2)	1.20681
3	0	1.29584 (263)	1.29352 (32)	1.29319 (-1)	1.29320
	1	1.21183 (305)	1.20904 (26)	1.20877 (-1)	1.20878
	2	1.21389 (388)	1.21030 (28)	1.21002 (0)	1.21002
	3	1.21320 (342)	1.21004 (26)	1.20978 (0)	1.20978
5	1	1.21223 (338)	1.20912 (27)	1.20884 (-1)	1.20885
	3	1.21406 (412)	1.21022 (28)	1.20993 (-1)	1.20994

3.4.2 Red-Black Iterations

The performance of the HFEM-PN solver with the RB iteration scheme was examined by solving the ABTR benchmark problems with various PN orders and basis function settings. In the PN extension verification tests, the cross-section set and geometry were limited to the two-group structure and 2D mesh, respectively, due to the direct inversion of the HFEM-PN linear system used. Once the RB iteration scheme was in place, the verification tests of HFEM-PN solver could be performed for the 2D and 3D meshes with the cross-section set with different group structures.

First, we examined the performance improvement from the RB iteration scheme for the 2D 2G ABTR

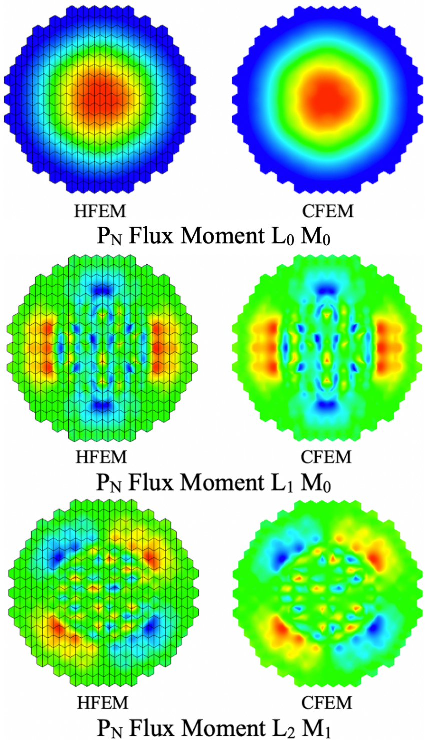


Figure 13: Comparisons of PN flux moment distributions for 2D ABTR problems obtained using HFEM-PN and CFEM-PN solver options.

benchmark problem. The eigenvalue results are not discussed here since the RB iteration scheme does not alter the converged solution, and the eigenvalues from the RB iteration scheme are identical with the corresponding values given in Table 3. Table 4 compares the computing times of the direct inversion based on LU decomposition and the RB iteration schemes. As the degrees of freedom (DoFs) increase, the computing times of direct solver cases are drastically increased and quickly reached a practically unfeasible level for reactor analysis. The computing times are significantly reduced by introducing the RB iteration, as shown in Table 4. Thus, the RB iteration scheme allows researchers to effectively solve HFEM-PN equations.

The HFEM-PN solver performance was further investigated for the 2D ABTR problem with the various energy group structures: 2G, 9G, and 33G. The HFEM-PN solver computing time for each energy group structure used are summarized in Table 5 with the different PN and anisotropic scattering orders. For the P3 and P5 options with P3 anisotropic scattering, the HFEM-PN solver obtains the converged solution in about 1 and 2 minutes, respectively, with the PJFNK mode that uses the RB iteration scheme as a preconditioner. The Richardson iteration scheme takes more time versus the PJFNK mode, and the computing time gaps between two modes increase with an increasing DoFs, which can be also observed in Table 4. The accuracy

Table 4: Performance improvements of HFEM-PN using RB iterations scheme for a 2D ABTR problem (eight processors).

PN Order	Scattering Order	Basis Function Order (Volume / Surface)	# of DoFs	Computing Time, seconds		
				Direct Solver	RB Iteration	
					PJFNK	Richardson
1	0	2 / 0	8,438	0.9	1.39	1.04
		3 / 1	14,852	3.1	2.08	1.80
		4 / 2	22,278	10.1	2.78	2.05
3	3	2 / 0	33,752	44.2	2.14	1.59
		3 / 1	59,408	361.7	4.05	4.14
		4 / 2	89,112	1,728.7	4.94	4.63
5	3	2 / 0	75,942	958.0	2.85	2.47
		3 / 1	133,668	N/A	7.56	9.42
		4 / 2	200,502	N/A	11.00	13.73

of obtained eigenvalue was assessed against the reference eigenvalue results from the DFEM-SN transport calculations with P3 anisotropic scattering options, where each hexagonal assembly is represented using 32 quadrilateral elements and 32 directions are used to the angular discretization. As shown in Table 6, the eigenvalue results from the HFEM-PN solver option converge to the reference eigenvalues using the higher order PN option. Additionally, the accuracy of assembly power distributions obtained from the HFEM-PN solver option was verified against the reference DFEM-SN results. Fig. 14 compares the reference power distribution with the P3 and P5 results for the 2D ABTR problem using the 33G cross-section set. The power difference is a maximum of about 0.13% in the P3 result and further reduced to less than 1% with the P5 option, which indicates good agreement with the reference transport solution.

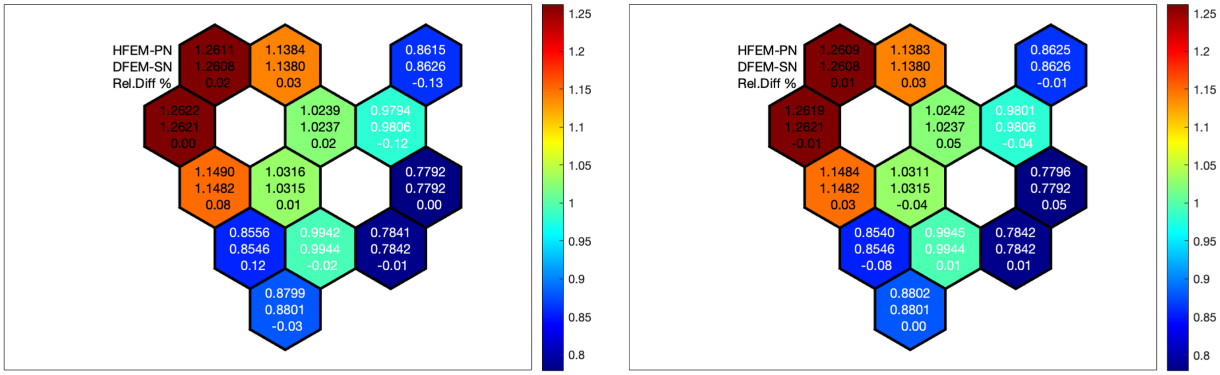
Table 5: Performance of HFEM-PN for the 2D ABTR problem with various energy groups and PN orders (volume / surface basis function orders: 3 / 1, eight processors).

PN Order	Scattering Order	Group	# of DoFs / Group	Computing Time, seconds	
				PJFNK	Richardson
1	0	2	7,426	2.1	1.8
		9		5.2	5.2
		33		18.3	17.8
3	3	2	29,704	4.1	4.1
		9		13.4	15.7
		33		58.9	70.9
5	3	2	66,834	7.6	9.4
		9		27.6	43.3
		33		105.4	191.3

The parallel performance of RB iteration scheme was examined by solving the 2D 33G ABTR problems

Table 6: Comparisons of HFEM-PN eigenvalue solutions with reference DFEM-SN results for the 2D ABTR problem (volume / surface basis function orders: 3 / 1).

Group	PN Order	Scattering Order	Eigenvalue	
			HFEM-PN (Δk)	Reference (DFEM-SN)
2	3	3	1.21022 (-24)	1.20997
	5	3	1.21004 (-7)	
9	3	3	1.23128 (60)	1.23189
	5	3	1.23182 (7)	
33	3	3	1.22825 (62)	1.22887
	5	3	1.22881 (7)	



P_N Order: 3

P_N Order: 5

Figure 14: Comparisons of assembly-wise power distribution with HFEM-PN and reference DFEM-SN results for 2D 33G ABTR problem (volume / surface basis function orders: 3 / 1, anisotropic scattering order: 3).

with various numbers of processors, as summarized in Table 7 and Fig. 15. The computational time of the computing time progressively decreased as increasing the number of processors for the PJFNK and Richardson iteration modes, which shows good parallel scalability of the solver. The parallel performance will be further improved in the next FY by optimizing the implementation of RB iteration scheme.

Table 7: Parallel performance of HFEM-PN with the RB iteration scheme for the 2D 33G ABTR problem (PN order: 3, scattering order: 3, volume / surface basis function orders: 3 / 1).

# Proc.	# Elem / Proc.	# of DoFs / Proc.	Computing Time, seconds	
			PJFNK	Richardson
1	1,575	980,232	350.79	390.96
4	391	241,560	99.27	118.15
8	190	119,064	58.91	70.91
12	127	81,048	42.28	56.33
16	105	63,888	36.15	45.44
24	69	43,296	27.74	39.61
32	50	31,152	22.47	33.04
48	33	20,328	20.80	33.67
64	24	14,784	20.55	35.05

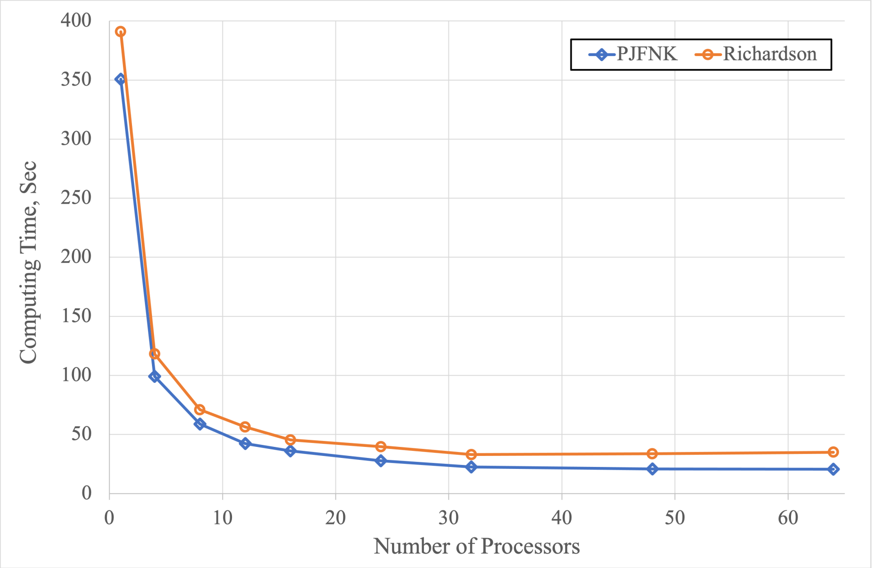


Figure 15: Parallel performance comparisons of the HFEM-PN solver with the RB iteration scheme for 2D 33G ABTR problem.

Finally, we preliminarily performed the 3D ABTR problems with the multigroup cross-section sets, and the detailed comparisons of accuracy and performance against the DFEM-SN solver options and other codes such as INSTANT [22] and VARIANT [23] are underway. The preliminary results are given in Table 8 and Fig. 16. The reference eigenvalue result is obtained using the DFEM-SN solver option with a

P3 anisotropic scattering treatment. In the DFEM-SN calculations, the 3D mesh file is uniformly refined compared to the HFEM-PN one (128 quadrilateral elements per assembly), and 32 directions are used to the angular discretization. As expected, the eigenvalue results converged to the reference value as the PN order increases. Fig. 16 illustrates that the HFEM-PN solver can represent the variation of intra-nodal flux and moment distributions properly with the higher-order basis functions.

In the verification tests, we consistently observed that the PJFNK option performs better than the Richardson iteration option, especially for a problem with a large number of DoFs. It seems that the PJFNK option has the advantage in that:

- The algorithm that is used in the PJFNK executioner requires small number of Newton iterations for the eigenvalue system.
- The RB iterations don't need to fully converge in the PJFNK executioner since it was used as the preconditioner.

The Richardson iteration option has potential performance improvement since it can be further accelerated by introducing the CMFD scheme that is adopted in the DFEM-SN option, and we plan to integrate the CMFD acceleration scheme.

Table 8: HFEM-PN eigenvalue solutions and computing times with various PN for 3D 9G ABTR problem (volume / surface basis function orders: 3 / 1, 64 processors)

PN Order	Scattering Order	Eigenvalue		Computing Time, seconds	
		HFEM-PN (Δk, pcm)	Reference (DFEM-SN)	PJFNK	Richardson
1	1	1.03200 (-1302)	1.04322	95.5	84.6
3	3	1.04197 (-125)		228.2	387.2
5	3	1.04316 (-6)		653.9	1,106.1

3.4.3 Discontinuity Factor

The 3 × 3 2D TREAT mockup benchmark problem [16], shown in Fig. 17, is used as a test problem to verify that discontinuity factors were correctly computed using the HFEM-Diffusion solver. Vacuum boundaries are applied to the right and top boundaries, and reflecting boundaries are applied to the left and bottom of the 3 × 3 mesh. Once the DFs are computed and applied to the benchmark problem, the assembly averaged powers are compared to the heterogeneous Serpent model used to compute the homogenized cross

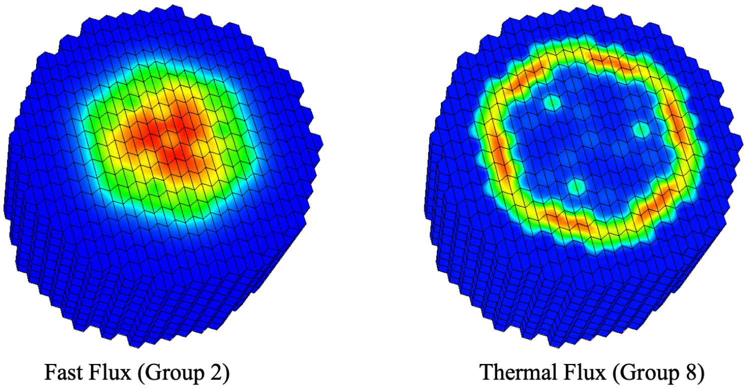


Figure 16: Thermal and fast flux distribution obtained from the HFEM-PN solver for the 3D 9G ABTR problem.

sections and reference parameters for DF computation in Griffin. The percent error levels by assembly are shown in Table 9, Which are supposed to be all zeros but these power differences are within the uncertainty range of assembly powers tallied in Serpent-2. These assembly power results are also in line with when the DFEM-Diffusion solver is used to compute DFs and homogenized assembly powers.

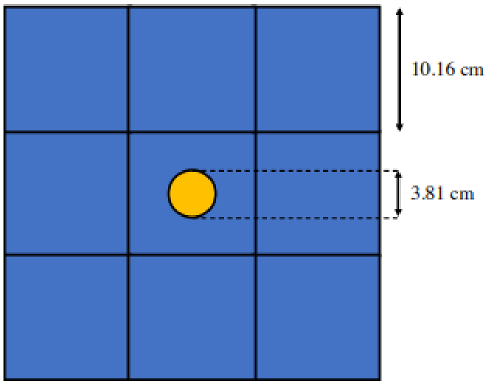


Figure 17: 3 × 3 2D mockup benchmark problem based on the TREAT fuel (blue) and control rod (yellow) compositions.

Table 9: Relative errors in assembly powers for the 3 × 3 2D TREAT mockup benchmark problem between Serpent heterogeneous and Griffin homogeneous models. In Griffin, DFs are applied for all surfaces in the problem using the HFEM-Diffusion solver. Assemblies on bottom and left outer boundaries correspond to reflecting boundaries, while assemblies on top and right outer boundaries correspond to vacuum boundaries.

Relative errors (%) in Assembly Powers		
0.02%	0.11%	-0.08%
-0.04%	0.01%	0.11%
0.01%	-0.04%	0.02%

4. PIN POWER RECONSTRUCTION

For problems that leverage spatial homogenization to compute the resultant flux distribution, pin power reconstruction is a necessary feature to evaluate pin-by-pin power distributions within fuel assemblies of interest. In order to reconstruct heterogeneous pin powers from homogeneous flux solutions, smoothly varying homogeneous flux solutions calculated from Griffin are used in conjunction with precomputed form functions to recover heterogeneous pin powers [24, 25]. Pin power reconstruction in Griffin is designed to work in a flexible manner so that it can support calculations for advanced reactor designs where the fuel assembly design can significantly deviate from traditional light water reactor (LWR) fuel assembly designs. Currently, the Griffin implementation allows users to define the pin mesh on a regular Cartesian or hexagonal pin lattice, with the eventual target of being able to define the pin mesh on a completely unstructured grid as well. Moreover, the pin power reconstruction methodology is not tied to a particular type of transport solver within Griffin but is implemented as a postprocessing step designed to operate on the output flux solution from any of the CFEM-, DFEM-, or HFEM-based solvers available in Griffin. Section 4.1 outlines the theory and methodology used for pin power reconstruction, while Section 4.3 details the specifics of how users can invoke the pin power reconstruction user object in Griffin. In order to test this newly developed functionality, Section 4.3 outlines two benchmark tests to compare the performance of Griffin-calculated pin powers to reference pin powers computed with the Serpent-2 Monte Carlo code [26].

4.1 Theory and Methodology

A new Griffin user object—named `PinPowerReconstructUserObject`—was created to handle pin power reconstruction. Here, a distinction must be made between the *solution mesh*, which represents the mesh over which the homogeneous flux solution is computed, and the *reconstruction mesh*, which is the pin-level mesh over which pin powers will be reconstructed. The solution mesh can be accounted for as the mesh discretization used to represent the homogenized assemblies in the problems, while the reconstruction mesh represents the pin elements that are overlaid onto the solution mesh. `PinPowerReconstructUserObject` requires that the solution mesh be tagged with an ”assembly ID” extra element ID to designate assembly regions in the mesh, where solution mesh elements that share the same extra element integer can be regarded as a single assembly region. Using the volume-averaged centroid of each unique assembly ID, the reconstruction mesh is overlaid onto the solution mesh in a Cartesian or hexagonal lattice configuration. Fig. 18 represents the distinction between the solution and reconstruction meshes, where the black-colored mesh is used to compute the homogenized flux solution in Griffin, while the green-colored mesh establishes the elements that will be integrated over to compute the integral powers within the pin. In this figure, the four solution mesh elements each have unique assembly IDs, and the 3 × 3 pin lattice is overlaid onto each of these four homogenized assembly regions. The reconstruction mesh is only defined within the scope of the pin power reconstruction postprocessing routines, not throughout the transport solver calculations.

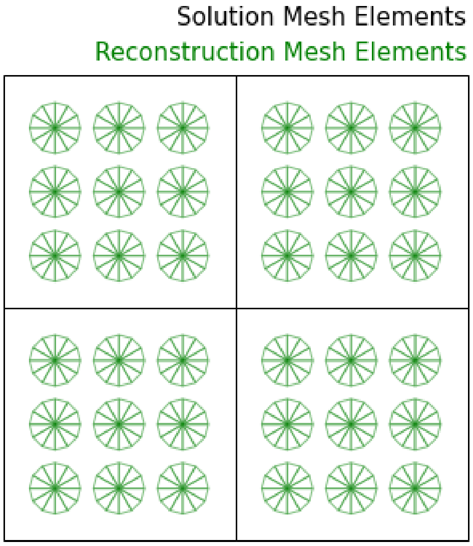
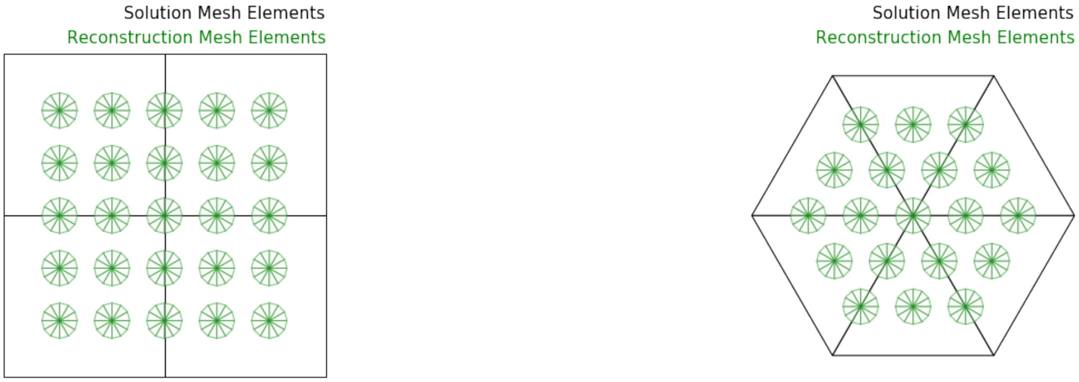


Figure 18: Illustration of solution mesh elements (black squares) vs. reconstruction mesh elements (green triangles).

In order to provide more flexibility to the user, all reconstruction mesh elements for a particular assembly ID do not need to be completely contained within a single solution mesh element. In fact, the solution mesh elements can be discretized in any way the user chooses. The only requirement is that each reconstruction mesh element of the overlaid pins must be contained within a single solution mesh element (i.e., a reconstruction mesh element cannot be contained within multiple solution mesh elements). Fig. 19b and Fig. 19b illustrate this feature for a Cartesian and hexagonal pin lattice, respectively. In both of these figures, multiple solution mesh elements are tagged with the same assembly ID to designate a single homogenized assembly region, and the reconstruction mesh elements are overlaid onto the solution mesh using the volume-averaged centroid of the entire assembly region. In this manner, each solution mesh element is responsible for integrating the κ-fission reaction rate of the pin mesh elements that are contained within that specific solution mesh element.



(a) Pin power reconstruction mesh overlay for Cartesian lattice

(b) Pin power reconstruction mesh overlay for hexagonal lattice

Figure 19: Examples of pin power reconstruction mesh overlay for cases where multiple solution mesh elements share a single “assembly ID” extra element integer value, denoting a single “assembly” region.

Another aspect worth noting regarding the reconstruction pin mesh overlay is that regular Cartesian and hexagonal pin lattices in reactor geometries are typically modeled as circular pins. However, since meshes used by MOOSE-based codes must be defined using either triangular or quadrilateral elements, these circular pins are discretized into a number of triangular subelements of equal volume with a side length made to match the volume of the original circular pin. The flux solution and quadrature points corresponding to the nodes that comprise of each triangular subelement in the pin are re-evaluated, and quadrature integration is used to compute the integral fission production rate within each triangular element. The pin power for pin i, then, is computed as:

$$P_i = \sum_{g=1}^{G} FF_{g,i} \sum_{t=1}^{T} \sum_{q=1}^{Q} (\kappa\Sigma_f)_{q,t,g,hom} \Phi_{q,t,g,hom}, \tag{18}$$

where g represents the group index for the G number of energy groups in the problem, t represents the index used to identify the T triangles that sub-discretize the circular pin i, q represents the quadrature point index for the Q quadrature points within triangle t, and $FF_{g,i}$ is the form function for group g and pin i. In this manner, $(\kappa\Sigma_f)_{q,t,g,hom}$ denotes the homogeneous κ-fission (recoverable energy release per fission) cross section evaluated at quadrature point q within triangle element t and group g, while the scalar flux $\Phi_{q,t,g,hom}$ represents the re-evaluated homogeneous flux solution at the quadrature point q, triangle element t, and group g. Since the flux shape within an assembly is readily available in the Griffin solution with finite element method (FEM), the flux reconstruction process required by conventional transverse integration-based nodal methods does not need to be performed in Griffin.

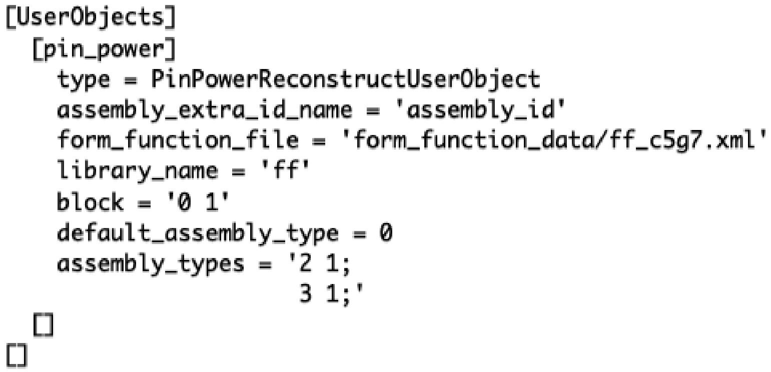
In Griffin, each pin is discretized into $T = 12$ triangles, and this number was chosen to balance the trade-off between improved numerical integration accuracy over a circular volume (larger value for T) and faster integration over the circular domain (smaller value for T). Moreover, this pin discretization ensures that the reconstruction mesh elements line up exactly with the solution mesh elements of Cartesian, x-type hexagonal, and y-type hexagonal assemblies.

In Eq. (18), the form function is a single multiplicative factor used to superimpose heterogeneous pin effects onto the homogeneous fission production rate distribution. This data is stored in a separate library in ISOXML and read by the `PinPowerReconstructUserObject` in Griffin, and details of this library can be found in Reference [27]. In addition to storing the form function data, ISOXML's `Form_Function_Library` also stores the the pin geometry and lattice layout of the pin for a particular assembly.

Regarding the algorithm specifics for computing pin powers, the `PinPowerReconstructUserObject` iterates through each solution mesh element, determines the reconstruction mesh elements contained within the specific solution mesh element, and computes the integral κ-fission rate of each reconstruction mesh element according to Eq. (18) through the quadrature integration method. These integral reaction rates are aggregated to a results array that stores the total pin power based on "assembly ID" and "pin ID", based on which assembly and pin the reconstruction mesh element belongs to, and this happens in a completely distributed fashion. Therefore, if multiple MPI processes are responsible for separate solution mesh elements that split a single pin into multiple regions, an `MPI.AllGather` call is a final step to ensure that integral reaction rate results that reside on separate MPI processes are summed together for reconstruction mesh elements that belong to the same pin region. These integral pin powers are finally outputted for each unique assembly ID and pin location in a human-readable manner, as well as in a CSV format.

4.2 User Input for PinPowerReconstructUserObject

As mentioned in Section 4.1, since ISOXML's `Form_Function_Library` stores pin-wise and group-wise form function data as well as pin geometry and lattice layout information, none of this information needs to be defined directly within the `PinPowerReconstructUserObject` input block. Consequently, the parameters to the `PinPowerReconstructUserObject` can be defined relatively simply, and a typical input block for this user object is shown in Fig. 20. The extra element ID name that designates the assembly location over which to compute pin powers is specified using the parameter `assembly_extra_id_name`. The location of the form function library is specified in `form_function_file`, and the library name containing relevant form function and pin geometry information is specified in `library_name`. Finally, in order to map assembly IDs in the problem geometry to the pin structure and form function data to be used within the assembly, the parameter `assembly_types` is used to map each unique assembly ID in the solution mesh to the corresponding ID in the form function library. For scenarios where multiple assemblies need to be mapped to the same form function ID, the parameter `default_assembly_type` can be used to create a default mapping between all assembly ID's and a specific form function ID. For assembly IDs that require a separate form function ID mapping, the `assembly_types` map should be used.



```
[UserObjects]
  [pin_power]
    type = PinPowerReconstructUserObject
    assembly_extra_id_name = 'assembly_id'
    form_function_file = 'form_function_data/ff_c5g7.xml'
    library_name = 'ff'
    block = '0 1'
    default_assembly_type = 0
    assembly_types = '2 1;
                      3 1;'
  □
□
```

Figure 20: Typical input block definition of `PinPowerReconstructUserObject`.

4.3 Verification Tests

In order to test the pin power reconstruction capabilities within Griffin, three verification tests are introduced in this section: the 2D ABTR benchmark [15], 2D Empire benchmark [28], and the C5 benchmark [9]. Among them, the first one is a fast spectrum problem based on hexagonal assemblies, and the other two are thermal spectrum problems based on hexagonal and Cartesian assemblies, respectively.

For all verification problems tested here, assembly-homogenized core calculations were conducted using DFEM-Diffusion with DF data that were derived from whole-core or colorset calculations, and the form function data for the pin power reconstruction were generated from a single fuel assembly for the Empire benchmark and whole-core or colorset calculations for the C5 benchmark. The use of SPH instead of DF as an equivalence parameter was attempted as well for the Empire benchmark, however its discussion is deferred to next fiscal year since a further study is needed to accurately reconstruct a homogeneous flux solution within an assembly for out-core fuel assemblies when DF data are generated for fuel assemblies only.

4.3.1 ABTR Benchmark

The ABTR is a sodium-cooled fast spectrum reactor design with 199 assemblies arranged in a hexagonal grid with nine rings and an assembly pitch of 14.685 cm [15]. Within each fuel assembly, 217 fuel pins are arranged in a hexagonal lattice of nine rings, with a pin pitch of 0.9134 cm and a pin radius of 0.3501 cm. For this benchmark problem, a 2D configuration of this core design was modeled. First, a heterogeneous core model was created with the Serpent-2 code, where homogenized cross sections, reference group-wise scalar fluxes and group-wise outgoing partial surface currents for all assemblies within the core, and reference pin powers in all fissile assemblies are all calculated within the Serpent-2 model. This model used four energy groups and applied vacuum boundary conditions on the radial boundary. A schematic of the heterogeneous model is shown in Fig. 21.

With the homogenized cross sections as well as the reference flux and outgoing current data, a homogenized Griffin model is created in order to compute DFs for each of the six surfaces in each assembly in the ABTR core [16]. The homogeneous model is run once again with DF's computed, and this DF-based equivalence procedure is applied to ensure consistency in the groupwise scalar fluxes and groupwise outgoing surface currents between the heterogeneous Serpent-2 model and the homogenized Griffin model. Once the DFs are applied, the resulting homogeneous flux distribution is postprocessed through the `PinPowerReconstructUserObject` to compute reconstructed pin powers for each of the fuel assemblies in the core. In Griffin, the HFEM-Diffusion transport solver of PN order 3 is used, and each homogenized hexagonal assembly is discretized into six equilateral triangular solution mesh elements.

Within `PinPowerReconstructUserObject`, no form functions were applied to the integral homoge-

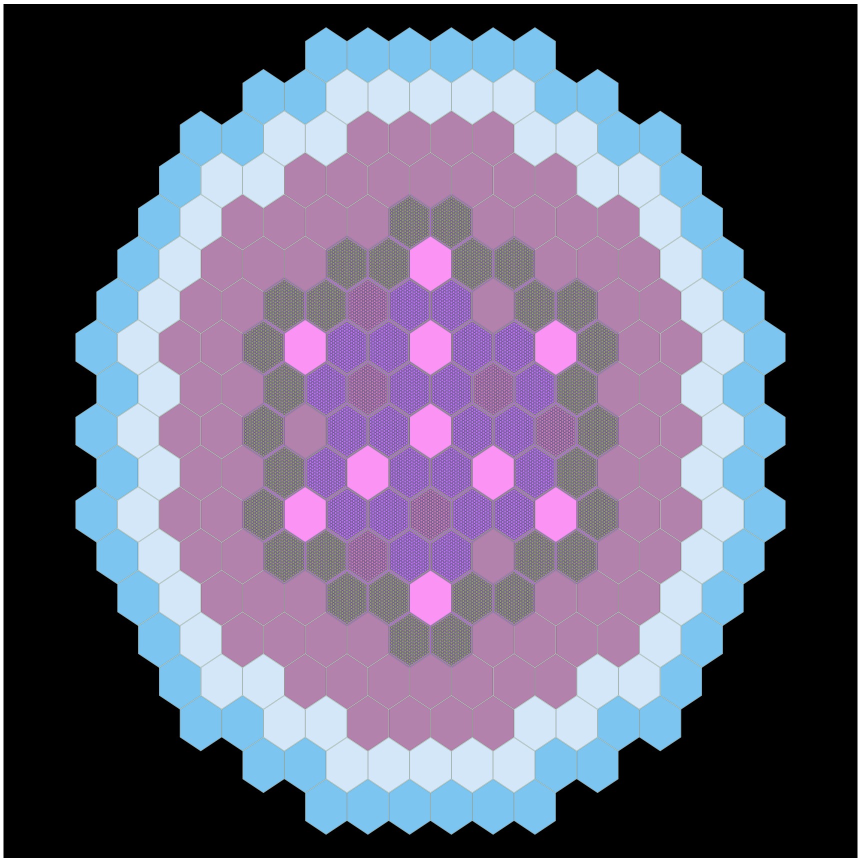


Figure 21: Heterogeneous Serpent-2 model used to compute homogenized cross-section data, reference equivalence parameters for DF calculation, and reference pin powers for all fuel assemblies in the ABTR model.

neous pin powers, as strong heterogeneous effects were not expected due to the fast spectrum of this reactor type. The resulting pin powers were compared to the reference pin powers tallied with Serpent-2. Fig. 22 plots the assembly-wise relative errors in pin power between Serpent-2 and Griffin for each fuel assembly for the top right one-third core, where the top number in each assembly corresponds to the assembly ID, the middle number is the root-mean-squared (RMS) error of powers for the pins within the assembly, and the bottom number corresponds to the maximum absolute relative error in pin powers within the assembly. The bottom-left corner of the figure corresponds to the center of the ABTR core. The color map is scaled by a RMS error, where blue is used to denote low RMS error assemblies and red is used to denote high RMS error assemblies. For this problem, the core-wide RMS error in pin powers is 0.9%, and the maximum absolute relative error within the core is 7.7%.

To show a more fine-grained spatial distribution of pin power errors, Fig. 23 plots the relative pin power error for assemblies with ID's 47, 63, 80, 97, 130, and 163, which all exhibit relatively high pin power error

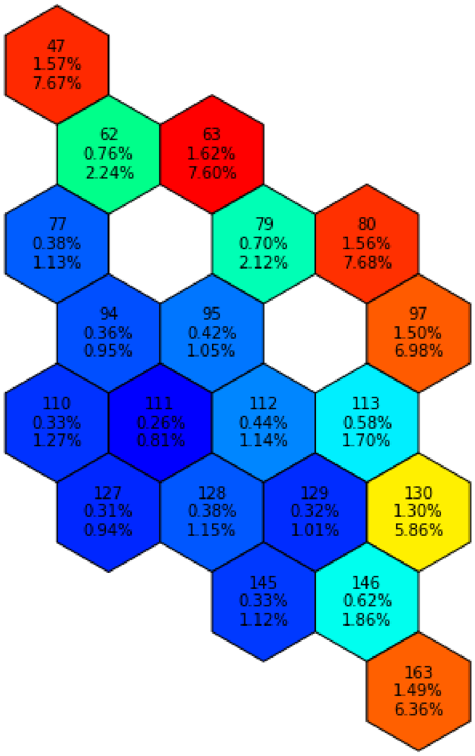


Figure 22: Assemblywise error map for pin powers within all fuel assemblies of the one-third ABTR core (Assembly IDs 110 and 127 are the closest to the core center).

levels. Most pin errors within these assemblies exhibit relatively low magnitudes, as evidenced by the low overall RMS error levels in each pin. However, the maximum pin power error seems to be concentrated in a single corner region of the assembly typically furthest away from the center of the core. Implementing corner DFs to the representation of the corner fluxes within the assembly should help reduce this maximum error [24], which is future work. Moreover, these assemblies where maximum pin power error levels are highest are also furthest away from the center of the core, where on an absolute scale, power levels are also lower in magnitude.

4.3.2 Empire Benchmark

The Empire benchmark is a thermal microreactor problem with four rings of assemblies also arranged in a hexagonal grid with an assembly pitch of 32.353 cm. Each fuel assembly contains nine rings of pin

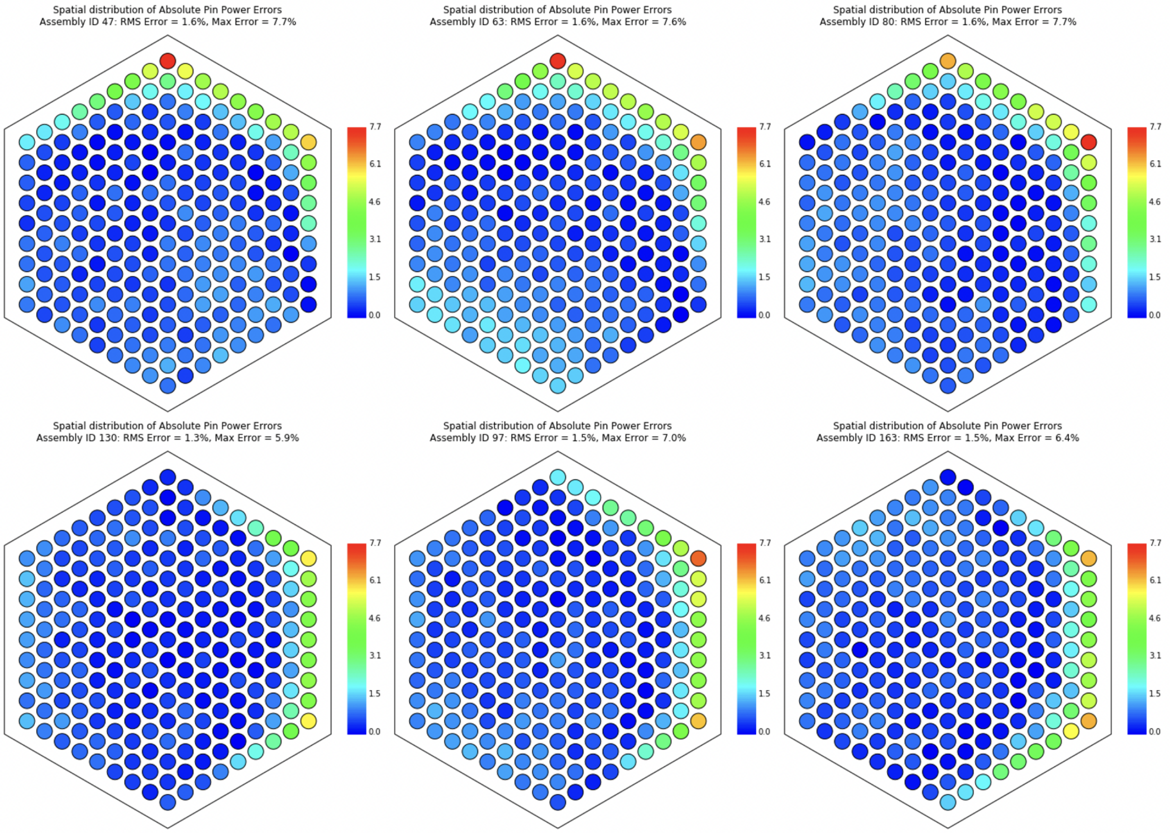


Figure 23: Spatial distribution of pin-by-pin absolute errors between Griffin and Serpent-2 for candidate assemblies in the ABTR core.

regions with a pitch of 2.15 cm, of 61 sodium and stainless-steel heat pipes, 96 moderator pins, and 60 0.925-cm radius fuel pins [28]. Once again, a 2D configuration of this design is modeled for this verification problem. Similar steps are taken as with Section 4.3.1, where the homogenized cross sections, reference parameters for equivalence parameter generation (flux and outgoing currents), and reference pin powers are computed with the Serpent-2 code. Fig. 24 plots the geometry of this heterogeneous serpent-2 model. Here, nine energy groups are used and vacuum boundary conditions are applied to the outer boundary. The Empire core model is a central void region, surrounded by two rings of fuel assemblies and an outer ring of reflector assemblies. Some of these reflector assemblies also contain a rotating control drum. For this model, the control drums are assumed to be in a fixed position in a radially outward facing orientation.

This time, however, the DFEM-Diffusion solver is used to generate discontinuity factors and calculate pin powers for all fuel assemblies in Griffin, in order to show that the current pin power reconstruction methodology is not tied to a particular transport solver. Moreover, a separate single assembly Serpent-

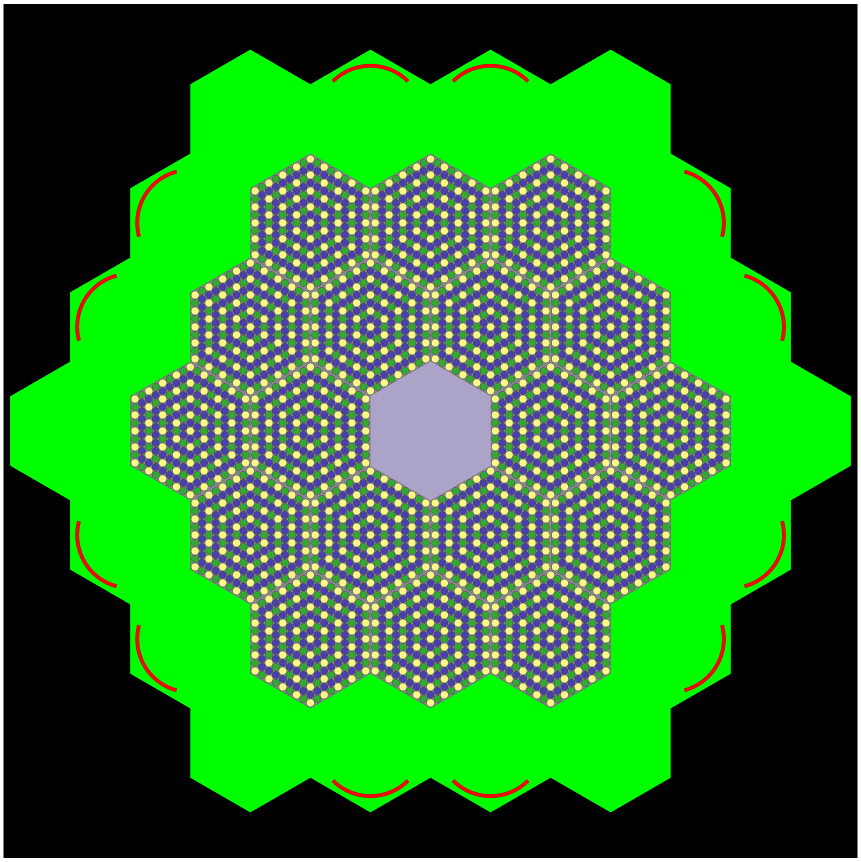


Figure 24: Heterogeneous Serpent-2 model used to compute homogenized cross-section data, reference equivalence parameters for DF calculation, and reference pin powers for all fuel assemblies in the Empire model. The green pins within the second and third rings of assemblies correspond to the fuel pin locations.

2 calculation is conducted to compute pin-by-pin heterogeneous κ-fission rates for a single Empire fuel assembly, which can be used to compute form functions. Since reflective boundary conditions are applied on this single assembly problem, flat homogeneous flux distributions are assumed during the form function calculation. These form functions are applied during the pin power reconstruction algorithm to superimpose pin-level heterogeneities onto the homogeneous flux solution calculated by Griffin, according to Eq. (18).

Fig. 25 plots the relative pin power error between Griffin and Serpent-2 for the Empire core. Due to the sixth-core symmetry of the Empire model, pin power errors are only shown for the three fuel assemblies to the bottom right of the central air region. A corewide RMS pin power error of 1.8% is observed, and the maximum absolute value of relative pin error within the core is 7.4%. In order to reduce these sources of error, a more sophisticated way of computing form functions such as a full-core procedure, can be pursued in the future instead of the single-assembly procedure.

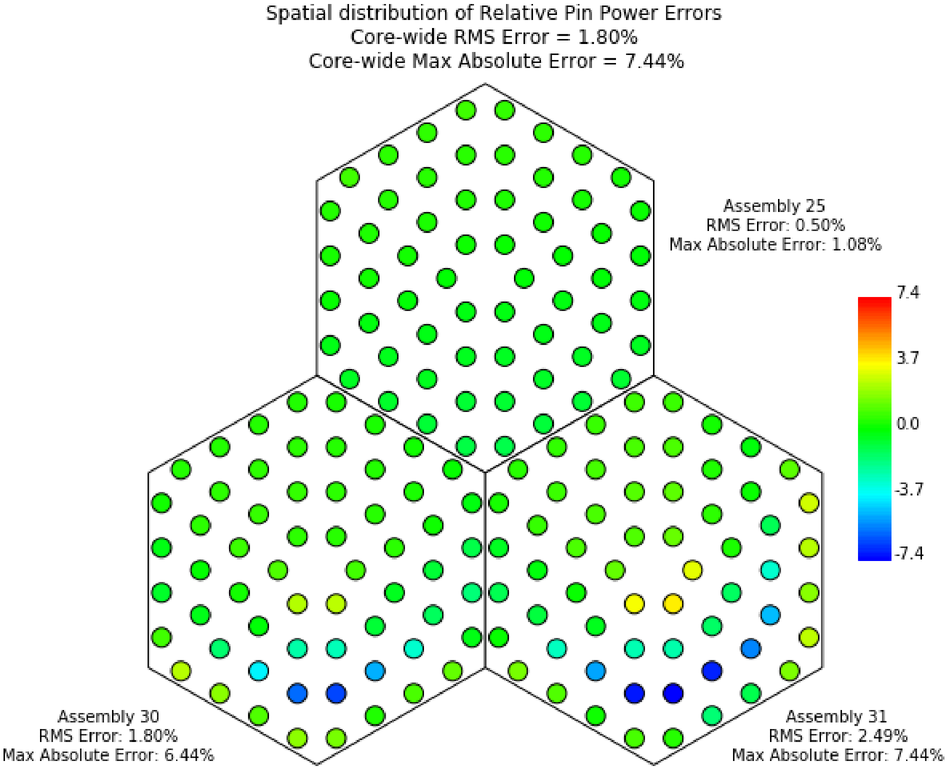


Figure 25: Spatial distribution of relative pin power errors between Serpent-2 heterogeneous model and Griffin homogeneous model for the Empire benchmark. The central void region is located at the top left of the top assembly shown in this diagram.

4.3.3 C5 Benchmark

As an initial verification problem, the 2D C5 benchmark problem is used to test the pin power reconstruction functionality described in the previous section. The 2D C5 benchmark is from the C5G7 benchmark problem [9], keeping its geometry but using 11-group cross sections in this study instead. Griffin's DFEM-Diffusion transport solver is used to solve for the whole-core flux solution, and the cumulative migration method (CMM) is used to generate assemblywise diffusion coefficients [29]. Moreover, each of the nine assembly-sized nodes in the problem is further discretized into four quadrilateral elements to solve for the whole-core homogeneous flux in the problem. Since this problem is solved using the diffusion equation, DFs are applied at each element surface (excluding reflecting boundary interfaces where DFs are set to 1.0) to make net current solutions between the heterogeneous and homogeneous systems equivalent. In this manner, three sets of data need to be defined as inputs to solve the C5 benchmark problem with Griffin's

DFEM-Diffusion formulation: the homogenized assembly cross sections, the pinwise form functions for each assembly, and the discontinuity factors for each surface in the problem. These sets of data are all managed using Griffin's ISOXML module [27]. The DFEM-Diffusion solver was used to compute reconstructed pin powers.

Two approaches for generating this input data are presented in this section, where eigenvalue differences and pin-by-pin errors will be compared against a reference Monte Carlo solution. The first approach uses a *full-core procedure* to generate homogenized cross sections, discontinuity factors, and pin-by-pin form functions. In this approach, it is assumed that a full-core heterogeneous Monte Carlo simulation is available to generate all input data to the Griffin DFEM-Diffusion solver. The second approach is based on the *colorset procedure*, which runs a series of single assembly and colorset calculations to generate input data for a subset of assemblies. It is then assumed that the conditions from these colorset calculations can be applied directly to the full-core case. Consequently, the full-core procedure is able to reproduce the full-core homogeneous flux and pin powers very accurately from the full-core heterogeneous Monte Carlo solution, but these parameters can be quite costly to calculate, especially as the size of the core increases. On the other hand, the colorset procedure generates input data more quickly but introduces larger approximations compared to the full-core diffusion solution by assuming that the conditions from the colorset calculations would be similar to those of the full-core case.

Full-core procedure In the full-core procedure to generate input cross sections, discontinuity factors, and form function data, the heterogeneous 2D full-core C5 problem is solved with the Serpent-2 code. For cross-section generation, Serpent-2 computes assembly-homogenized cross sections for each of the nine fuel assembly–sized nodal regions in the problem. In addition, reference fluxes for each nodal region and reference partial outgoing currents for each of the four surfaces of each nodal region are tallied in Serpent-2, which are then inputted to Griffin to generate DFs for each of the nodal surfaces in the problem using four elements per assembly node [16]. Finally, form function data are generated using Eq. (18), where the heterogeneous pinwise fission production rate is computed in Serpent-2, while the homogeneous pinwise fission production rate is computed in Griffin by setting up the equivalent homogeneous 2D full-core problem. In this manner, accurate homogeneous flux distributions can be calculated from leveraging equivalence and cross-section data from full-core Serpent-2 calculations, while form function data from full-core calculations will correctly superimpose heterogeneous pin-level effects onto the reconstructed pin power distribution in Griffin.

Colorset procedure As mentioned previously, the colorset procedure uses a subset of assemblies to generate input cross section, DF, and form function data, assuming that these conditions can be approximately extended to the full-core case. The input cross sections are generated from a 1 × 1 UO_2 single assembly calculation, a 2 × 1 MOX-Reflector calculation, and a 2 × 2 MOX-Reflector calculation. DFs are computed using a 2 × 1 UO_2-MOX colorset calculation to compute DFs for the UO_2-MOX interface, a 2 × 1 MOX-Reflector colorset calculation is performed to compute the DFs for the MOX-Reflector interface, and a 2 × 2 MOX-Reflector colorset calculation is conducted to generate DFs for the interface between the reflector and outer UO_2 assembly. In each of these cases, four elements per assembly node are used to compute discontinuity factors. Finally, single assembly MOX and single assembly UO_2 calculations with reflective boundary conditions are used to generate form function data for each pin in the fuel assemblies. Fig. 26 illustrates the colorsets and boundary conditions used for generating discontinuity factors for the full-core Griffin calculation.

Eigenvalue and pin power results The eigenvalue and pin power results for the 2D full-core C5 problem for the two procedures outlined in the previous section are summarized in Table 10. The reference solution is generated from Serpent-2 by computing the pin powers directly from the heterogeneous Monte Carlo solution for each pin in the benchmark problem. Using the full-core procedure, the Griffin DFEM-Diffusion eigenvalue can be made to match the Serpent-2 eigenvalue exactly by adjusting the DFs so that homogenized assembly powers and outgoing partial currents converge to those computed by reference Serpent-2 quantities. Moreover, form functions are computed from a heterogeneous pin power from Serpent-2 as well, so superimposing these form functions onto the homogeneous fission production rate distribution yields an exact agreement in reconstructed pin powers with Serpent-2 pin powers. However, as explained in the previous section, generating input parameters from a full-core procedure can be computationally costly and quite a cumbersome process to define all relevant input data as the core size increases.

Instead, the colorset procedure is examined to alleviate this computational burden by introducing additional assumptions to the full-core DF, cross section, and form function generation procedure to extrapolate colorset conditions to the full-core case. In this case, this procedure yields a 12-pcm difference in eigenvalue and a 2.2% RMS and maximum 4.6% error in pin powers when compared to the reference Serpent-2

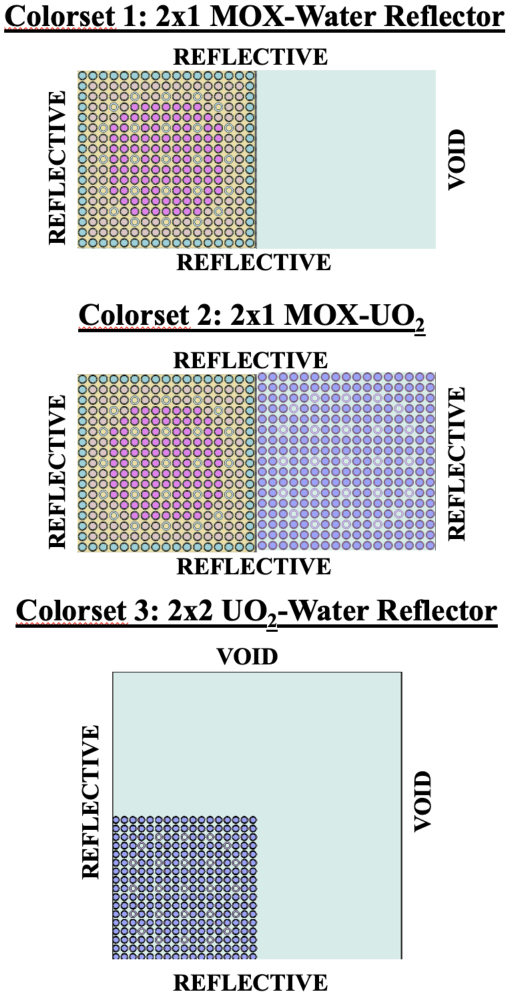


Figure 26: Colorsets used to generate DFs for the C5 problem.

solutions. These results are consistent with previous studies for the C5G7 problem [24]. A pin-by-pin error map between Serpent-2 and Griffin using the colorset procedure is shown in Fig. 27.

5. CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

For the DFEM-SN transport solver improvements of this report, we discussed the extension of developed capabilities that are necessary for heterogeneous reactor dynamics simulations. First, we discussed modifications to the sweep-CMFD iteration algorithm to include the evaluation of feedback quantities. Although, this evaluation occurs within an inner loop with CMFD acceleration, we found with the C5G7 multiphysics example that performing this inner iteration can result in divergent behaviour. Second, we discussed the en-

Table 10: Eigenvalue and pin power results for Griffin DFEM-Diffusion solution compared to Serpent-2 reference solution.

	Eigenvalue	Eigenvalue Difference (pcm)	Pin Power RMS Error (%)	Pin Power Max Error (%)
Serpent-2 Monte Carlo (Reference)	1.20442	—	—	—
Griffin DFEM-Diffusion (Full-core procedure)	1.20442	0	0.0	0.0
Griffin DFEM-Diffusion (Colorset procedure)	1.20431	-11	2.2	4.6

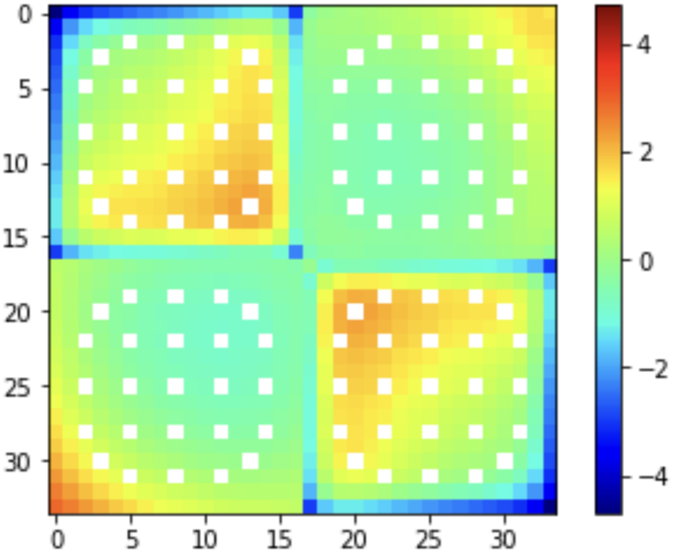


Figure 27: Relative errors (%) in pin powers for Griffin in the colorset procedure. The bottom left corner is the core center, and the top right quadrant represents the UO_2 fuel assembly that borders the water reflector region.

hancements to the DFEM-SN solver for transient simulations. These enhancements included the inclusion of additional implicit and explicit terms from time integration and DNPs to the sweeper and CMFD solver. IQS was also introduced with a multi-time-scale approach for multiphysics problems. The micro reactor model exemplifies this capability with the ability to tractably solve the seven-dimensional neutron transport equation with feedback in a heterogeneous geometry. The model also showcases the importance of a multi-time-scale approach provided by IQS. Finally, we discussed more subtle improvements to the sweeper and CMFD solver for better run-time performance.

As efforts to implement the effective low-order solver option in Griffin for assembly-homogenized models in FY-22, we extended the HFEM solver to the PN equations in order to properly account for the pronounced transport effects in advanced reactor designs. In the initial implementation done in the previous fiscal year, the solution scheme for HFEM directly inverts the linear system with LU decomposition, which is not efficient for cases with a large number of DoFs. The RB iteration scheme for the HFEM-PN solver was implemented and integrated in Griffin, which can effectively solve the large HFEM linear system by sweeping the domain. The implementation was verified with several benchmark problems, including ABTR cases by comparing with reference solutions obtained with different types of transport solvers. Expected spatial and angular convergences were observed. The RB iteration scheme can effectively alleviate the substantial computing burden of solving the HFEM linear system, which is essential for accomplishing a routine design calculation that requires a fast-running option with proper accuracy. Along with the HFEM solver extension work, we developed and implemented the DF method dedicated to the HFEM-Diffusion solver for homogenization equivalence.

Finally, pin power reconstruction capabilities were added to the Griffin solver. This capability is generalized to work with all existing Griffin neutron transport solvers, where a structured Cartesian or hexagonal pin lattice mesh is overlaid onto a homogenized solution mesh. This new functionality was tested onto three benchmark problems—2D ABTR, 2D Empire, and 2D C5. These benchmark problems exhibited core-wide RMS errors of 0.9–2.2% when combining Griffin HFEM-Diffusion and DFEM-Diffusion solvers with DF-based equivalence methods. Future work includes looking at extending this analysis to SPH-based equivalence approaches, as well as testing the functionality on the HFEM-PN solver. Eventually, the pin power reconstruction capability will be extended to a general reconstructed pin mesh provided by the user to handle cases where the pin mesh does not follow a structured lattice. The more user-friendly workflow incorporating the pin power reconstruction for more accurate multiphysics simulations will also be considered.

5.2 Future Work

- Removal of angular flux solution — For the DFEM-SN solver, it may not be necessary to store the entire angular flux solution. In transient cases, the time derivative and IQS PKE coefficients can be approximated using flux moments. This would allow only storing flux moments, which would improve cache performance in the sweeper, projection and prolongation operations in CMFD, and PKE parameter evaluation for IQS.

- Time adaptivity — From an analyst's perspective, a significant improvement would be implementing a time adaptation strategy for these simulations. Selecting an appropriate time step for a certain error is usually an inefficient trial and error process and having a non-constant time step throughout the transient—changing time steps as time scales change—would improve computational efficiency substantially.

- HFEM solver option — The performance of RB iteration will be improved so that the grind times of the red-black iteration and of the response matrix assembly are comparable with stand-alone previous HFEM-PN implementations in the VARIANT and INSTANT code. Feasibility of CMFD acceleration method in the HFEM-PN calculations will be investigated to accomplish desirable performance for a routine design analysis because it has a potential to reduce computational times of Richardson iteration mode by providing quick convergence of global flux solutions. Further verification works will be performed for various reactor types.

- Pin power reconstruction — The pin power reconstruction will be tested for the SPH approach with a study on accurate pin power representation, and its application and test will be extended to multi-physics transient calculations as well.

REFERENCES

[1] C. H. Lee, Y. Jung, H. Park, E. Shemon, J. Ortensi, Y. Wang, V. Labouré, and Z. Prince, "Griffin Software Development Plan," Research Report INL/EXT-21-63185, ANL/NSE-21/23, Idaho National Laboratory, Argonne National Laboratory, June 2021.

[2] C. H. Lee, Y. S. Jung, Z. Zhong, J. Ortensi, V. Laboure, Y. Wang, and M. DeHart, "Assessment of the Griffin reactor multiphysics application using the Empire micro reactor design concept," Tech. Rep. INL/LTD-20-59263, ANL/NSE-20/23, Argonne National Laboratory and Idaho National Laboratory, July 2020.

[3] Y. Wang, C. H. Lee, Y. S. Jung, Z. Prince, J. Hanophy, L. Harbour, H. Park, and J. Ortensi, "Performance improvements to the Griffin transport solvers," Tech. Rep. INL/EXT-21-64272, ANL/NSE-21/51, Idaho National Laboratory and Argonne National Laboratory, September 2021.

[4] Q. Shen and B. Kochunas, “A robust, relaxation–free multiphysics iteration scheme for cmfd–accelerated neutron transport k–eigenvalue calculations–i: Theory,” *Nuclear Science and Engineering*, vol. 195, no. 11, pp. 1176–1201, 2021.

[5] D. R. Gaston, C. J. Permann, J. W. Peterson, A. E. Slaughter, D. Andrs, Y. Wang, M. P. Short, D. M. Perez, M. R. Tonks, J. Ortensi, L. Zou, and R. C. Martineau, “Physics-based multiscale coupling for full core nuclear reactor simulation,” *Annals of Nuclear Energy*, vol. 84, pp. 45–54, October 2015.

[6] K. Ott, “Quasistatic treatment of spatial phenomena in reactor dynamics,” *Nuclear Science and Engineering*, vol. 26, no. 4, pp. 563–565, 1966.

[7] S. Dulla, E. H. Mund, and P. Ravetto, “The quasi-static method revisited,” *Progress in Nuclear Energy*, vol. 50, pp. 908–920, November 2008.

[8] Z. M. Prince and J. C. Ragusa, “Multiphysics reactor-core simulations using the improved quasi-static method,” *Annals of Nuclear Energy*, vol. 125, pp. 186–200, mar 2019.

[9] E. E. Lewis, M. A. Smith, G. Palmiotti, T. A. Taiwo, and N. Tsoul-Fanidis, “Benchmark specification for deterministic 2-D/3-D MOX fuel assembly transport calculations without spatial homogenisation (C5G7 MOX),” Tech. Rep. NEA/NSC/DOC(2001)4, OECD/NEA Expert Group on 3-D Radiation Transport Benchmarks, 2001.

[10] Y. Wang, S. Schunert, J. Ortensi, V. Laboure, M. DeHart, Z. Prince, F. Kong, J. Harter, P. Balestra, and F. Gleicher, “Rattlesnake: A moose-based multiphysics multischeme radiation transport application,” *Nuclear Technology*, vol. 207, no. 7, pp. 1047–1072, 2021.

[11] S. Terlizzi and V. Labouré, “Asymptotic hydrogen redistribution analysis in yttrium hydride moderated heat-pipe-cooled microreactors using direwolf,” *submitted to Annals of Nuclear Energy*, 2022.

[12] C. Matthews et al., “Coupled multiphysics simulations of heat pipe microreactors using direwolf,” *Nuclear Technology*, vol. 207, no. 7, pp. 1142–1162, 2021.

[13] E. Shemon, Y. S. Jung, S. Kumar, Y. Miao, K. Mo, A. Oaks, and S. Richards, “Moose framework meshing enhancements to support reactor analysis,” tech. rep., Argonne National Laboratory, ANL/NSE-21/43, 9 2021.

[14] E. E. Lewis and G. Palmiotti, "Red-black response matrix acceleration by transformation of interface variables," *Nuclear Science and Engineering*, vol. 130, no. 2, pp. 181–193, 1998.

[15] Y. Chang *et al.*, "Advanced burner test reactor preconceptual design report," Tech. Rep. ANL-ABR-1 (ANL-AFCI-173), Argonne National Laboratory, September 2006.

[16] V. Laboure, J. Ortensi, Y. Wang, S. Schunert, and M. DeHart, "Iterated discontinuity factors based on reference outgoing currents for discontinuous finite element diffusion," in *PHYSOR 2018: Reactor Physics paving the way towards more efficient systems*, (Cancun, Mexico), April 2018.

[17] Y. Wang, S. Schunert, V. Laboure, and Z. Prince, "Rattlesnake theory manual," Tech. Rep. INL/EXT-17-42103, Idaho National Laboratory, 2018.

[18] P. A. Raviart and J. M. Thomas, "Primal hybrid finite element methods for 2nd order elliptic equations," *Mathematics of Computation*, vol. 31, no. 138, pp. 391–413, 1977.

[19] X. S. Li and J. W. Demmel, "SuperLU_DIST," *ACM Transactions on Mathematical Software*, vol. 29, pp. 110–140, June 2003.

[20] Y. Wang, "INSTANT theory manual - part I. PN-hybrid-FEM for the multigroup transport equation," tech. rep., Idaho National Laboratory, February 2012.

[21] B. S. Kirk, J. W. Peterson, R. H. Stogner, and G. F. Carey, "libMesh : a C++ library for parallel adaptive mesh refinement/coarsening simulations," *Engineering with Computers*, vol. 22, pp. 237–254, December 2006.

[22] Y. Wang, C. Rabiti, and G. Palmiotti, "Krylov solvers preconditioned with the low-order red-black algorithm for the PN hybrid FEM for the INSTANT code," in *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2011)*, (Rio de Janeiro, RJ, Brazil), Latin American Section (LAS) / American Nuclear Society (ANS), May 2011.

[23] G. Palmiotti, E. E. Lewis, and C. B. Carrico, "VARIANT: Variational anisotropic nodal transport for multidimensional cartesian and hexagonal geometry calculation," Technical Report ANL-95/40, ARGONNE National Laboratory, 1995.

[24] H. G. Joo, J. I. Yoon, and S. G. Baek, “Multigroup pin power reconstruction with two-dimensional source expansion and corner flux discontinuity,” *Annals of Nuclear Energy*, vol. 36, no. 1, pp. 85–97, 2009.

[25] K. Rempe, K. S. Smith, and A. Henry, “Simulate-3 pin power reconstruction: methodology and benchmarking,” *Nuclear Science and Engineering*, vol. 103, no. 4, pp. 334–342, 1989.

[26] J. Leppänen, “Serpent – a continuous-energy Monte Carlo reactor physics burnup calculation code.,” tech. rep., VTT Technical Research Centre of Finland, 2015.

[27] H. Park, C. Lee, O. W. Calvin, S. Kumar, Y. Wang, and J. Ortensi, “Improved cross section generation capability of griffin in fy22,” Tech. Rep. INL/RPT-22-69178, ANL/NSE-22/60, Idaho National Laboratory and Argonne National Laboratory, September 2022.

[28] M. DeHart, J. Ortensi, and V. Labouré, “NEAMS reactor physics assessment problem,” Tech. Rep. INL/LTD-20-59184, Idaho National Laboratory, 2020.

[29] Z. Liu, K. Smith, B. Forget, and J. Ortensi, “A cumulative migration method for computing rigorous transport cross sections and diffusion coefficients for lwr lattices with monte carlo,” tech. rep., Idaho National Lab.(INL), Idaho Falls, ID (United States), 2016.