

An Assessment of Coupling Algorithms for Nuclear Reactor Core Physics Simulations[☆]

Steven Hamilton^{a,*}, Mark Berrill^a, Kevin Clarno^a, Roger Pawlowski^b,
Alex Toth^c, C. T. Kelley^c, Thomas Evans^a, Bobby Philip^a

^a*Oak Ridge National Laboratory, 1 Bethel Valley Rd., Oak Ridge, TN 37831 U.S.A.*

^b*Sandia National Laboratories, MS 0316, P.O. Box 5800, Albuquerque, NM 87185 U.S.A.*

^c*North Carolina State University, Department of Mathematics, Box 8205, Raleigh, NC 27695 U.S.A*

Abstract

This paper evaluates the performance of multiphysics coupling algorithms applied to a light water nuclear reactor core simulation. The simulation couples the k -eigenvalue form of the neutron transport equation with heat conduction and subchannel flow equations. We compare Picard iteration (block Gauss-Seidel) to Anderson acceleration and multiple variants of preconditioned Jacobian-free Newton-Krylov (JFNK). The performance of the methods are evaluated over a range of energy group structures and core power levels. A novel physics-based approximation to a Jacobian-vector product has been developed to mitigate the impact of expensive on-line cross section processing steps. Numerical simulations demonstrating the efficiency

[☆]Notice: This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energys National Nuclear Security Administration under contract DE-AC04-94AL85000.

*Corresponding Author

Email addresses: `hamiltonsp@ornl.gov` (Steven Hamilton), `berrillma@ornl.gov` (Mark Berrill), `clarnokt@ornl.gov` (Kevin Clarno), `rppawlo@sandia.gov` (Roger Pawlowski), `artoth@ncsu.edu` (Alex Toth), `tim_kelley@ncsu.edu` (C. T. Kelley), `evanstm@ornl.gov` (Thomas Evans), `philipb@ornl.gov` (Bobby Philip)

of JFNK and Anderson acceleration relative to standard Picard iteration are performed on a 3D model of a nuclear fuel assembly. Both criticality (k -eigenvalue) and critical boron search problems are considered.

Keywords: multiphysics, Jacobian-free Newton-Krylov, Anderson acceleration, nuclear reactor analysis

1. Introduction

Determining the steady-state power and temperature distributions within an operating nuclear reactor is an important component of reactor design and analysis. This task requires simultaneously solving equations describing the distribution of neutrons throughout the reactor as well as the transfer of heat through the fuel and structural materials into fluid coolant regions. Current core analysis methods rely on the use of a Picard iteration [1, 2, 3, 4, 5, 6, 7], alternating between solving individual physics components. Although this approach offers a simple path to coupling different physics codes due to the minimal code interaction required, there are also significant drawbacks. Picard iteration lacks a global convergence result and, at best, achieves a q-linear convergence rate [8]. Additionally, user-defined relaxation schemes are usually required to achieve convergence. Newton-based methods, however, are shown to be globally convergent with q-quadratic convergence rates. The downside to Newton-based methods is that the need for residual and sensitivity information requires more invasive access to application codes. While access to analytical Jacobian matrices is commonly infeasible, Jacobian-free Newton-Krylov (JFNK) methods [9] can be used to realize many of the benefits of Newton-based methods while only requiring evaluation of nonlinear functions. While JFNK methods have been successfully applied in many areas, to date, the application to multiphysics reactor simulations has been limited to few-group diffusion approximations to the transport equation with analytic temperature feedback models not suitable for accurate reactor analyses.

In this study, we investigate the performance of Anderson acceleration and JFNK solvers compared to Picard iteration on multiphysics problems that couple 3D discretizations of the radiation transport and heat transfer equations along with a simple subchannel flow model for modeling of pressurized water reactors (PWRs). Although Newton and JFNK methods have been used previously for neutronics-only problems [10, 11, 12] and even for

multiphysics problems [13, 14, 15, 16, 17], previous studies have used only few-group nuclear cross sections with analytic temperature variation. In this study we consider the impact of utilizing many energy groups with on-line generation of cross section data for use by the neutronics solver. One of the dominant costs associated with the current model is the on-line generation of cross sections; a significant contribution of this paper is development of a low-cost approximate function evaluation for the JFNK approach that mitigates this cross section processing cost. Another notable contribution is the design of a JFNK boron search formulation, allowing direct computation of the critical boron concentration as an alternative to standard indirect searches. Coupling algorithms are evaluated on both criticality (k -eigenvalue) and boron search problems.

The remainder of the paper is organized as follows: Section 2 describes the physics models, Section 3 describes various coupling algorithms, Section 4 contains numerical results for a single PWR fuel assembly, and Section 5 presents conclusions and proposals for future areas of investigation.

2. Physics Models

In this paper, we consider the solution of multiphysics problems involving coupling between neutron transport and heat transfer. In particular, we focus on the solution of problems involving light water reactors (LWRs). Most of the fundamental ideas described here are applicable to a wide range of reactor types, but certain aspects of the problem, such as geometric features, are particular to LWRs (and possibly PWRs in particular). For nuclear reactor problems, the standard formulation of the neutron transport equation is the k -eigenvalue problem

$$\begin{aligned} \hat{\Omega} \cdot \nabla \psi(\vec{r}, E, \hat{\Omega}) + \sigma(\vec{r}, E, T) \psi(\vec{r}, E, \hat{\Omega}) = \\ \frac{1}{4\pi} \int_0^\infty dE' \int_{4\pi} d\hat{\Omega}' \sigma_s(\vec{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, T) \psi(\vec{r}, E', \hat{\Omega}') + \\ \frac{1}{4\pi k} \chi(\vec{r}, E) \int_0^\infty dE' \int_{4\pi} d\hat{\Omega}' \nu \sigma_f(\vec{r}, E', T) \psi(\vec{r}, E', \hat{\Omega}') , \quad (1) \end{aligned}$$

where $\vec{r}$ is the coordinate vector, $\hat{\Omega}$ is the direction of particle travel, E is the particle energy, T is the temperature of the background material, σ is the total cross section, σ_s is the scattering cross section, $\nu \sigma_f$ is the neutron production cross section, and χ is the fission spectrum. The goal for solving

this equation is to find the largest value of the eigenvalue k and the corresponding eigenvector ψ . Because Eq. (1) represents an eigenvalue problem, the vector ψ has no explicit magnitude. We choose a natural normalization by setting the global heat generation rate (due to nuclear fission occurring in the fuel) to a pre-defined value, i.e.,

$$\frac{1}{4\pi} \int dV \int_0^\infty dE \int_{4\pi} d\hat{\Omega} \kappa \sigma_f \psi = P^* , \quad (2)$$

where κ is the heat generated per fission event, and nonlocal energy deposition (e.g., gamma heating) effects have been ignored.

As noted in Eq. (1), the cross sections are dependent on the temperature of the media, T . Thus, for a reactor not operating at a constant temperature, it is also necessary to solve a heat conduction equation within the solid fuel and clad regions, with fission providing the thermal source, i.e.,

$$-\nabla \cdot K(T) \nabla T = \frac{1}{4\pi} \int_0^\infty dE \int_{4\pi} d\hat{\Omega} \kappa \sigma_f(E) \psi(E, \hat{\Omega}) , \quad (3)$$

where K is the material thermal conductivity. Because no fission occurs in the clad regions, the source in those locations is zero. The exterior surface of the clad is then coupled to the coolant through the subchannel model that solves equations describing the conservation of mass, momentum, and energy,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0 \quad (4)$$

$$\frac{\partial \rho v_i}{\partial t} = -\nabla \cdot (\rho v_i \vec{v}) + (-\nabla p + \nabla \cdot \vec{\tau}) - \vec{g} \quad (5)$$

$$\frac{\partial U}{\partial t} + \nabla \cdot (U \vec{v}) = -p \nabla \cdot \vec{v} + \Phi + \nabla K(T) \nabla T + \dot{q} , \quad (6)$$

where ρ is the mass density, $\vec{v}$ is the velocity, p is the pressure, $\vec{g}$ is the force exerted by gravity, $\vec{\tau}$ is the viscosity tensor, U is the internal energy density, Φ is the dissipation function, and $\dot{q}$ is the thermal source. Note that the internal energy density is related to the enthalpy density through $U = h - p$. We solve these equations using a two-equation approximation in which we assume that the coolant flow is only in the axial direction and neglect thermal

diffusion between the channels [18]. Assuming steady-state, this reduces to

$$\frac{\partial \rho v_z^2}{\partial z} + \frac{\partial p}{\partial z} - \frac{\partial \tau}{\partial z} = -g(-\nabla p + \nabla \cdot \tau) - g, \quad (7)$$

$$\frac{\partial h v_z}{\partial z} = -v_z \frac{\partial p}{\partial z} - \Phi + \frac{\partial}{\partial z} \nabla K(T) \frac{\partial T}{\partial z} + \dot{q}. \quad (8)$$

The thermal source consists primarily of convective heat transfer from the clad and is the primary coupling mechanism with the temperature of the clad and fuel pellets. The degrees of freedom for the subchannel model are the enthalpy and pressure.

For notational simplicity, in the remainder of this document we will allow T to refer to not only the solution of the thermal diffusion equation in the pellets and clad, but also the solution to the subchannel equations. Our goal is therefore to find distributions ψ , T and a value k such that Eqs. (1)–(3) and (7)–(8) are simultaneously satisfied. For ease of notation, we introduce an operator notation for discretized forms of the preceding equations:

$$\mathbf{A}(T)\phi = \lambda \mathbf{B}_\nu(T)\phi \quad (9)$$

$$\mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T)\phi = P^* \quad (10)$$

$$\mathbf{L}(T)T = \mathbf{R}_E \mathbf{B}_\kappa(T)\phi, \quad (11)$$

where $\mathbf{A}$ represents the streaming, collision, and scattering terms from Eq. (1), $\mathbf{B}_{\nu/\kappa}$ represents the fission integral weighted with ν (as in Eq. (1)) or κ (as in Eqs. (2)–(3)), respectively, $\mathbf{R}_S$ and $\mathbf{R}_E$ are restriction operators in space and energy, respectively, and $\mathbf{L}(T)$ refers not only to the thermal diffusion equation (3), but also the subchannel equations of (7)–(8). Here, Eq. (9) has taken advantage of the fact that only the angle-integrated variable ϕ ($\phi \equiv \int \psi d\hat{\Omega}$) must generally be stored rather than the corresponding angle-dependent ψ . In addition, we have written the eigenvalue as $\lambda \equiv \frac{1}{k}$.

2.1. Cross Section Processing

Problem-independent, evaluated nuclear cross section data is prepared for use with deterministic transport by collapsing hundreds of thousands of data points for every potential isotope to create multi-group cross section libraries applicable to a class of problems, such as thermal-spectrum nuclear reactors [19]. For each potential reaction (hundreds) of a neutron with every potential isotope (hundreds) in the multi-group library, the probability of that interaction occurring (a cross section) has been averaged over many (tens to

hundreds) neutron-energy groups at room temperature [20]. In addition, the multi-group libraries contain additional data that enables the extension of the data to approximate materials at other temperatures using an intermediate resonance theory approach [21]. In stand-alone neutronics calculations, such as radiation shielding or criticality safety, this data is often used with a deterministic transport solver to predict the spatial distribution of neutrons in a system [22].

For multiphysics applications, such as nuclear reactor analysis, the computational cost of modeling hundreds of energy groups is overly burdensome, and the accuracy when directly using tens of energy groups is not sufficient. Therefore, an additional cross section processing step is utilized to collapse the multi-group cross section library generated for the class of problem (e.g., thermal reactor) into a smaller, problem-specific collapsed multi-group library, which can contain from two to tens of neutron energy groups. In order for this collapsed multi-group data to accurately represent the energy spectrum in the problem, it is necessary to compute the averages using a weighting function that matches the true solution as closely as possible. This weighting function is usually generated by performing several local multi-group 1-D (pincell) or 2-D (lattice) transport calculations to capture local nearest-neighbor contributions to the solution. Such a calculation must be performed for every unique type of fuel pin in the problem. In addition, the variation in material temperature on the collapsed data must be taken into account [21].

The traditional approach is to precompute collapsed multi-group data at several different temperatures and then interpolate the values to the temperature of interest. A more accurate approach, however, is to perform a separate cross section processing calculation at every different temperature under consideration [4]. The latter approach of “on-line” generation of cross sections is the method considered in this paper.

2.2. Critical Boron Search Formulation

A nuclear reactor operating under normal conditions is always maintained in a critical state: $k \equiv 1$. Because of this, rather than compute the value of k for a given reactor configuration it is sometimes preferred to compute the value of some parameter which results in a critical configuration. In a PWR, this parameter is often the concentration of soluble boron in the coolant. In this case, a second formulation is defined where the eigenvalue is no longer an unknown, and the matrix $\mathbf{A}$ becomes a function of the boron concentration,

C. Thus Eq. (9) becomes

$$\mathbf{A}(T, C)\phi = \lambda_{\text{target}}\mathbf{B}_\nu(T)\phi, \quad (12)$$

where λ_{target} is the target eigenvalue for the search, which will typically be unity for full core calculations but may be another value for calculations involving only a portion of a reactor. Together with Eqs. (10) and (11), Eq. (12) represents the problem statement for a critical boron search.

We briefly note that other single-parameter critical searches (e.g., control rod height) can be written in a form similar to (12), with the matrices $\mathbf{A}$ and possibly $\mathbf{B}$ depending on the given parameter. Additional critical searches will not be considered further in this paper.

3. Solution Approaches

The radiation transport and heat transfer systems for the k -eigenvalue formulation described in Eqs. (9)–(11) can be written as a single system of coupled equations:

$$f_\lambda \begin{pmatrix} \phi \\ \lambda \\ T \end{pmatrix} = \begin{bmatrix} f_\phi(\phi, \lambda, T) \\ f_\lambda(\phi, T) \\ f_T(\phi, T) \end{bmatrix} = \begin{bmatrix} \mathbf{A}(T)\phi - \lambda\mathbf{B}_\nu(T)\phi \\ \mathbf{R}_S\mathbf{R}_E\mathbf{B}_\kappa(T)\phi - P^* \\ \mathbf{L}(T)T - \mathbf{R}_E\mathbf{B}_\kappa(T)\phi \end{bmatrix} = \mathbf{0}. \quad (13)$$

Alternatively, to solve the boron search problem described in Section 2.2, the system can be written as

$$f_C \begin{pmatrix} \phi \\ C \\ T \end{pmatrix} = \begin{bmatrix} f_\phi(\phi, C, T) \\ f_C(\phi, T) \\ f_T(\phi, T) \end{bmatrix} = \begin{bmatrix} \mathbf{A}(T, C)\phi - \lambda_{\text{target}}\mathbf{B}_\nu(T)\phi \\ \mathbf{R}_S\mathbf{R}_E\mathbf{B}_\kappa(T)\phi - P^* \\ \mathbf{L}(T)T - \mathbf{R}_E\mathbf{B}_\kappa(T)\phi \end{bmatrix} = \mathbf{0}. \quad (14)$$

In this section we explore strategies for solving these systems of equations.

3.1. Picard Iteration

One straightforward approach to solving this system of equations is to alternate between solves of the individual physics in a block Gauss-Seidel approach. This Picard iteration can be written as shown in Alg. 1. It has been observed in several studies with various physics approximations and/or discretizations that this simple iteration scheme applied to LWR problems is prone to poor convergence and possibly divergence due to oscillations induced by certain error modes. The standard remedy for this issue is to introduce a

Algorithm 1 Picard Iteration (Gauss-Seidel) for k -eigenvalue Formulation

Given T_0
for $m = 0, 1, \dots$ until converged **do**
 Solve $\mathbf{A}(T^m)\hat{\phi} = \lambda\mathbf{B}_\nu(T^m)\hat{\phi}$ for $\hat{\phi}, \lambda$
 Set $\phi^{m+1} = \frac{P^*}{\mathbf{R}_S\mathbf{R}_E\mathbf{B}_\kappa(T^m)\hat{\phi}}\hat{\phi}$
 Solve $\mathbf{L}(T^{m+1})T^{m+1} = \mathbf{R}_E\mathbf{B}_\kappa(T^m)\phi^{m+1}$ for T^{m+1}
end for

Algorithm 2 Damped Picard Iteration for k -eigenvalue Formulation

Given T_0
for $m = 0, 1, \dots$ until converged **do**
 Solve $\mathbf{A}(T^m)\hat{\phi} = \lambda\mathbf{B}_\nu(T^m)\hat{\phi}$ for $\hat{\phi}, \lambda$
 Set $\phi^{m+1} = \frac{P^*}{\mathbf{R}_S\mathbf{R}_E\mathbf{B}_\kappa(T^m)\hat{\phi}}\hat{\phi}$
 Solve $\mathbf{L}(\hat{T})\hat{T} = \mathbf{R}_E\mathbf{B}_\kappa(T^m)\phi^{m+1}$ for $\hat{T}$
 Update $T^{m+1} = \omega\hat{T} + (1 - \omega)T^m$
end for

damping parameter, ω , such that Alg. 1 is replaced by the modified version given in Alg. 2. Optimal values for the damping parameter are typically between 0.3 and 0.6 [1, 2, 3]. Note that it is also possible to perform damping on the scalar flux (or power) instead of temperature.

For the boron search formulation, since Eq. (12) does not represent a linear eigenvalue problem the way that Eq. (9) does, solving the coupled system of equations using Picard iteration is slightly more complicated. One approach is to use a traditional solution approach (e.g., power iteration) to solve the original eigenvalue problem, but to modify the algorithm so that the boron concentration is updated instead of the eigenvalue [23, 24]. This approach, however, presents two problems. First, subspace eigenvalue solvers (e.g., Arnoldi's method, or a generalized Davidson method) that are commonly far more efficient than power iteration can no longer be used. Second, every time the boron concentration is modified, the problem cross sections must be updated; this can be a significant computational expense if cross sections are being processed on-line. A more common approach is to solve the boron search problem indirectly by solving Eqs. (9)–(11), but adding an extra outer iteration to update the boron concentration such that the eigenvalue at convergence is unity, resulting in Alg. 3. In this approach,

Algorithm 3 Damped Picard Iteration for Boron Search Formulation

Given $T^0, C^0, \frac{\partial \rho}{\partial C}$
for $m = 0, 1, \dots$ until converged **do**
 Solve $\mathbf{A}(T^m, C^m)\hat{\phi} = \lambda \mathbf{B}_\nu(T^m)\hat{\phi}$ for $\hat{\phi}, \lambda$
 Set $\phi^{m+1} = \frac{P^*}{\mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T^m)\hat{\phi}} \hat{\phi}$
 Set $C^{m+1} = C^m - \frac{1}{\frac{\partial \rho}{\partial C}} (\lambda_{\text{target}} - \lambda)$
 Solve $\mathbf{L}(\hat{T})\hat{T} = \mathbf{R}_E \mathbf{B}_\kappa(T^m)\phi^{m+1}$ for $\hat{T}$
 Update $T^{m+1} = \omega \hat{T} + (1 - \omega)T^m$
end for

Algorithm 4 Anderson Acceleration

Given x_0, β and $m \geq 1$.
 Set $x_1 = \mathbf{G}(x_0); F_0 = \mathbf{G}(x_0) - x_0$.
for $k = 1, 2, \dots$ (until converged) **do**
 Set $m_k = \min\{m, k\}$.
 Set $F_k = \mathbf{G}(x_k) - x_k$.
 Minimize $\|\sum_{j=0}^{m_k} \alpha_j^k F_{k-m_k+j}\|$ subject to
 $\sum_{j=0}^{m_k} \alpha_j^k = 1$.
 Set $x_{k+1} = (1 - \beta) \sum_{j=0}^{m_k} \alpha_j^k x_{k-m_k+j} + \beta \sum_{j=0}^{m_k} \alpha_j^k \mathbf{G}(x_{k-m_k+j})$.
end for

an estimate for the derivative of the reactivity with respect to the boron concentration, $\frac{\partial \rho}{\partial C}$, must be provided.

3.2. Anderson Acceleration

The Picard algorithms described above are an example of fixed-point iteration

$$x = \mathbf{G}(x). \quad (15)$$

Recently, a fixed-point acceleration scheme called Anderson acceleration [25] has garnered much attention and has been shown to be effective on certain classes of problems [26, 27, 28]. The Anderson acceleration scheme is shown in Alg. 4. The minimization problem is solved in the ℓ^2 norm as described in Ref. [29].

Anderson acceleration is attractive for multiphysics coupling since it can be applied with minimal requirements beyond what typical black-box Picard coupling requires (i.e., it is non-invasive). Anderson acceleration builds

on Picard iteration by storing solutions from previous fixed-point iterations to accelerate convergence. Fang and Saad [26] have shown that Anderson is related to quasi-Newton methods (known as a type II Broyden or “bad” Broyden), while Walker and Ni [29] have shown that for linear systems of equations, Anderson is “essentially equivalent” to GMRES. Toth and Kelley [30] have further characterized the convergence behavior of Anderson acceleration. Lott et. al. [27] have shown, for instance, that Anderson not only improved the speed of convergence, but also significantly improved the robustness.

Although it is possible to apply Anderson acceleration directly to the monolithic residual set of Eq. (13) (e.g., $F_k = f_\lambda$), preliminary studies indicate that this approach exhibits very poor convergence behavior for the problem of interest and is unlikely to be competitive without significant work to develop appropriate preconditioning techniques ($F_k = M^{-1}f_\lambda$, where M^{-1} is the preconditioning operator). Note that an approach similar to this was used for the stand-alone neutronics k -eigenvalue problem in Ref. [28] with some success, though the absence of multiphysics feedback and the differing solution algorithms used for comparison make it difficult to draw substantive comparisons to our current problem. A more effective approach is to use Anderson acceleration to accelerate and/or stabilize Picard iteration. Based on Alg. 1, we can define two fixed-point maps. First we define a mapping, $P = \mathbf{g}(T)$, that computes a power distribution from a known temperature distribution, i.e.,

$$P = P^* \frac{\mathbf{R}_E \mathbf{B}_\kappa(T) \phi}{\mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T) \phi}, \quad (16)$$

where ϕ is the solution to the eigenvalue problem

$$\mathbf{A}(T) \phi = \lambda \mathbf{B}_\nu(T) \phi. \quad (17)$$

Next we define a second mapping, $T = \mathbf{h}(P)$, that is given by solving the nonlinear equation

$$\mathbf{L}(T) T = P. \quad (18)$$

With this notation, the Picard iteration scheme described by Alg. 1 can be written as the fixed-point mapping

$$T^{j+1} = \mathbf{h}(\mathbf{g}(T^j)), \quad (19)$$

which can be solved using Anderson acceleration. Alternative mappings are also possible, such as the reverse mapping that acts on the power unknowns,

Algorithm 5 Inexact Newton Method [31]

Given x_0 and $\eta \in [0, 1)$
for $m = 0, 1, \dots$ until converged **do**
 Solve (approximately) $\mathbf{J}(x^m)\delta x = -f(x^m)$ for δx such that
 $\|f(x^m) + J(x^m)\delta x\| \leq \eta\|f(x^m)\|$.
 Set $x^{m+1} = \delta x + x^m$.
end for

i.e.

$$P^{j+1} = \mathbf{g}(\mathbf{h}(P^j)), \quad (20)$$

or even a mapping that allows acceleration of all unknowns, i.e.

$$x^{j+1} = \begin{pmatrix} T^{j+1} \\ P^{j+1} \end{pmatrix} = \begin{pmatrix} \mathbf{g}(T^j) \\ \mathbf{h}(P^j) \end{pmatrix}. \quad (21)$$

In this work, we apply Anderson acceleration to the fixed-point mapping in Eq. (19); the mappings of Eqs. (20) and (21) were tested and produced similar results.

3.3. Jacobian-Free Newton-Krylov

An alternative approach for solving Eq. (13) is to use an Inexact Newton method [31] (Alg. 5). Computation of the Jacobian matrix, $\mathbf{J} = \frac{\partial f}{\partial x}$, is generally not possible because the temperature dependence of nuclear data is not available in closed form, but is rather the result of solving numerous local transport problems as described in Section 2.1. The availability of a nonlinear function but difficulty in formulating the corresponding Jacobian matrix suggests that the use of Jacobian-free Newton Krylov (JFNK) methods might be appropriate [9, 32]. JFNK methods are based on two primary ideas. First, if the Newton correction equation is solved using a Krylov subspace method, then access to the full Jacobian matrix is not necessary; only the action of the Jacobian applied to a vector is required. The second idea is that the product of the Jacobian and a given vector can be approximated using a finite difference approach, e.g.,

$$\mathbf{J}(u^m)v \approx \frac{f(u^m + \epsilon v) - f(u^m)}{\epsilon}. \quad (22)$$

Note that second (or higher) order approximations are possible (as is the use of automatic differentiation to approximate Jv), but the form in Eq. (22) is

far more common because it requires only a single function evaluation per Jacobian-vector product. This is because the value of $f(u^m)$ can be computed and stored once per Newton iteration so that each subsequent Jacobian-vector product requires only the evaluation of $f(u^m + \epsilon v)$. The JFNK method therefore allows an approximate Newton method to be performed using only evaluations of the nonlinear function. Discussions on the selection of ϵ can be found in Ref. [9].

JFNK methods have been used previously to solve the k -eigenvalue problem by itself [10, 11, 12], as well as time-dependent multiphysics problems involving radiation transport coupled with thermal-hydraulics [14, 15]. Additionally, steady-state coupling between neutron diffusion and thermal hydraulics using Newton’s method (not JFNK) was studied in Refs. [13] and [16]. A significant distinguishing feature between these studies and the current work is that all of the previous studies used nuclear cross sections that were either fixed or allowed to vary in a prescribed, analytic manner. In this study, we consider “on-line” cross section processing in which the methods described in Section 2.1 are applied during the course of the calculation. Furthermore, the cited studies all use a very small number of energy groups (no more than seven groups were used in any of these studies, and two-group cross sections were used in all cases considering multiphysics problems), while the current study considers the use of as many as 23 energy groups with cross section processing using up to 252 groups. This approach is necessary to achieve the accuracy required for typical reactor physics analyses [4, 5, 33]. Finally, we are aware of no study that provides a direct comparison of the computational efficiency of Picard iteration and any Newton-based method for coupled neutronics problems. As we will see in Section 4, a straightforward application of JFNK does not result in a viable solution approach when on-line cross section processing is performed.

3.3.1. JFNK Preconditioning

In order for JFNK to be competitive with other solution approaches, it is necessary to efficiently solve the Newton correction equation. Because a Krylov method is used to solve the linear system involving the Jacobian matrix, the development of effective preconditioning strategies is vital. Although the use of JFNK eliminates the need to explicitly form or store the Jacobian matrix, some knowledge about the Jacobian is still beneficial in the construction of a preconditioner. The true Jacobian corresponding to the

k -eigenvalue formulation defined by (13) can be written as

$$\mathbf{J}_\lambda \begin{pmatrix} \phi \\ \lambda \\ T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T) - \lambda \mathbf{B}_\nu(T) & -\mathbf{B}_\nu(T)\phi & \frac{\partial(\mathbf{A}(T)\phi - \lambda \mathbf{B}_\nu(T)\phi)}{\partial T} \\ \mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T) & 0 & \frac{\partial(\mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T)\phi)}{\partial T} \\ -\mathbf{R}_E \mathbf{B}_\kappa(T) & 0 & \frac{\partial(\mathbf{L}(T)T - \mathbf{R}_E \mathbf{B}_\kappa(T)\phi)}{\partial T} \end{bmatrix}, \quad (23)$$

where the entries in the last column of this block matrix are left as partial derivatives with respect to temperature to indicate that these terms are generally not available in closed form due to the dependence on material properties. Similarly, the Jacobian matrix corresponding to the boron search formulation in (14) can be written as

$$\mathbf{J}_C \begin{pmatrix} \phi \\ C \\ T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T) - \lambda_{\text{target}} \mathbf{B}_\nu(T) & \frac{\partial(\mathbf{A}(T)\phi)}{\partial C} & \frac{\partial(\mathbf{A}(T)\phi - \lambda \mathbf{B}_\nu(T)\phi)}{\partial T} \\ \mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T) & 0 & \frac{\partial(\mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T)\phi)}{\partial T} \\ -\mathbf{R}_E \mathbf{B}_\kappa(T) & 0 & \frac{\partial(\mathbf{L}(T)T - \mathbf{R}_E \mathbf{B}_\kappa(T)\phi)}{\partial T} \end{bmatrix}. \quad (24)$$

A simple approach to preconditioning a linear system involving (23) or (24) is a block diagonal approach in which preconditioners for each physics are applied independently, i.e.,

$$\mathbf{M} = \begin{bmatrix} \widehat{\mathbf{A}} - \lambda \widehat{\mathbf{B}}_\nu & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \widehat{\mathbf{L}} \end{bmatrix}, \quad (25)$$

where $\widehat{(\cdot)}$ indicates that some approximation of the operator is used in the preconditioner. Possible choices for this approximation include incomplete factorizations, algebraic multigrid methods, or physics-based approximations. This preconditioner selection has the advantages of being relatively simple to construct and inexpensive to apply (assuming the approximations to the individual physics operators are inexpensive). Additionally, the block diagonal structure of this preconditioner means that the different physics components can be applied independently and simultaneously, allowing for the possibility

to treat the physics domains simultaneously in parallel. Neglecting all terms in the Jacobian that correspond to coupling between different physics components may lead to a reduction in the effectiveness of the preconditioner. Therefore, it may be beneficial to capture some of the off-diagonal terms from the true Jacobian. Possible approaches for moving beyond block-Jacobi preconditioning include Schur complement and operator splitting based techniques, such as those considered in Ref. [34] for coupled radiation diffusion problems, **but are not considered further in this work**. Development of preconditioning strategies that account for coupling behavior between physics components has the potential to significantly improve the convergence behavior of the JFNK approach and the development of such techniques is an attractive area of research for future investigations.

3.3.2. Approximate Jacobian-Vector Products

As noted in Section 2, the cross section data in Eq. (1) is dependent on the material temperature, resulting in the temperature dependence of the matrices $\mathbf{A}$ and $\mathbf{B}$. Generating data suitable for use in a deterministic radiation transport solver generally involves performing a large number of small (1-D or 2-D) radiation transport calculations involving many energy groups, as described in Section 2.1. Because the tabulated data used in these small calculations is temperature dependent, a new calculation must be performed for every region in the reactor for which a distinct temperature is defined. Furthermore, every time the temperature of a given region is modified, a new calculation must be performed. Although each individual calculation represents a relatively small computational burden, the large number of calculations that may be required in a given simulation may result in a large portion of the overall computational effort being spent in this cross section processing. Using the JFNK method, evaluation of the function in Eq. (13) at a perturbed state point appears as

$$f_{\lambda} \begin{pmatrix} \phi + \Delta\phi \\ \lambda + \Delta\lambda \\ T + \Delta T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T + \Delta T)(\phi + \Delta\phi) - (\lambda + \Delta\lambda)\mathbf{B}_{\nu}(T + \Delta T)(\phi + \Delta\phi) \\ \mathbf{R}_S\mathbf{R}_E\mathbf{B}_{\kappa}(T + \Delta T)(\phi + \Delta\phi) - P^* \\ \mathbf{L}(T + \Delta T)(T + \Delta T) - \mathbf{R}_E\mathbf{B}_{\kappa}(T + \Delta T)(\phi + \Delta\phi) \end{bmatrix}, \quad (26)$$

and, similarly, evaluation of Eq. (14) results in

$$f_C \begin{pmatrix} \phi + \Delta\phi \\ C + \Delta C \\ T + \Delta T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T + \Delta T, C + \Delta C)(\phi + \Delta\phi) - \lambda_{\text{target}} \mathbf{B}_\nu(T + \Delta T)(\phi + \Delta\phi) \\ \mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T + \Delta T)(\phi + \Delta\phi) - P^* \\ \mathbf{L}(T + \Delta T)(T + \Delta T) - \mathbf{R}_E \mathbf{B}_\kappa(T + \Delta T)(\phi + \Delta\phi) \end{bmatrix}, \quad (27)$$

where the appearance of $\mathbf{A}(T + \Delta T)$ and $\mathbf{B}_\nu(T + \Delta T)$ indicate the need to recompute cross sections based on the perturbed temperature distribution and boron concentration *at every linear (Krylov) iteration*. This is in contrast to Picard iteration, where cross sections only need updating once every fixed-point iteration. With the expectation that several Newton iterations will be required to converge a given problem and potentially dozens of linear iterations will be necessary for every Newton iteration, the time spent simply updating cross sections is likely to be prohibitive with a straightforward application of JFNK.

In order to circumvent this potential bottleneck, we propose using an approximate Newton update equation,

$$\hat{\mathbf{J}}(u^m) \delta^m = -f(u^m), \quad (28)$$

where $\hat{\mathbf{J}}$ indicates that an approximation to the Jacobian is used. In contrast to inexact Newton methods which involve solving the true Newton correction equation in an approximate manner [31], the current approach is more accurately described as a preconditioned **Picard** iteration [8]. In the interest of maintaining the attractive matrix-free nature of JFNK, we can determine a corresponding approximate function evaluation, $\hat{f}$, such that a finite difference operation approximates a product with $\hat{\mathbf{J}}$ rather than the full Jacobian, i.e.,

$$\hat{\mathbf{J}}(u^m)_v \approx \frac{\hat{f}(u^m + \epsilon v) - f(u^m)}{\epsilon}, \quad (29)$$

where $\hat{f}$ is an approximation to Eq. (13) that does not require recalculation of cross section data. This approach is consistent with strategies studied in Ref. [35]; utilizing inexactness in the function evaluations within JFNK was also studied in Ref. [36], although in that study the model used to perform approximate function evaluations was periodically updated to capture additional physics information, whereas the approximation in the current study

remains static. It should be noted that the full nonlinear function, including processing of all cross sections, must still be evaluated once for every nonlinear iteration and is used to evaluate convergence of the nonlinear iterations.

We explore two approximations to the Jacobian-vector product in this paper. The first approximation is to simply neglect the temperature variation of cross sections during an approximate Jacobian-vector product, corresponding to an approximate Jacobian of

$$\widehat{\mathbf{J}}_\lambda \begin{pmatrix} \phi \\ \lambda \\ T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T) - \lambda \mathbf{B}_\nu(T) & -\mathbf{B}_\nu(T)\phi & 0 \\ \mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T) & 0 & 0 \\ -\mathbf{R}_E \mathbf{B}_\kappa(T) & 0 & \frac{\partial(\mathbf{L}(T)T)}{\partial T} \end{bmatrix}, \quad (30)$$

or, equivalently, an approximate function evaluation of

$$\widehat{f}_\lambda \begin{pmatrix} \phi + \Delta\phi \\ \lambda + \Delta\lambda \\ T + \Delta T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T)(\phi + \Delta\phi) - (\lambda + \Delta\lambda)\mathbf{B}_\nu(T)(\phi + \Delta\phi) \\ \mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T)(\phi + \Delta\phi) - P^* \\ \mathbf{L}(T + \Delta T)(T + \Delta T) - \mathbf{R}_E \mathbf{B}_\kappa(T)(\phi + \Delta\phi) \end{bmatrix}. \quad (31)$$

This selection, however, results in no updated temperature (or boron) information exchange being communicated from the heat transfer solver to the neutronics domain during a given nonlinear iteration. This lack of information is expected to have a detrimental effect on the convergence behavior of the nonlinear solver. We denote this modified JFNK method as MJFNK1 in the results section. For the boron search formulation, the equivalent approximate Jacobian (**neglecting the dependence of the cross sections on both temperature and boron concentration**) is given by

$$\widehat{\mathbf{J}}_C \begin{pmatrix} \phi \\ C \\ T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T, C) - \lambda_{\text{target}} \mathbf{B}_\nu(T, C) & 0 & 0 \\ \mathbf{R}_S \mathbf{R}_E \mathbf{B}_\kappa(T, C) & 0 & 0 \\ -\mathbf{R}_E \mathbf{B}_\kappa(T, C) & 0 & \frac{\partial(\mathbf{L}(T)T)}{\partial T} \end{bmatrix}. \quad (32)$$

Note that this matrix is block lower triangular and has a zero **diagonal** block, so it is singular regardless of ϕ , C , or T and should not be expected to produce a viable method.

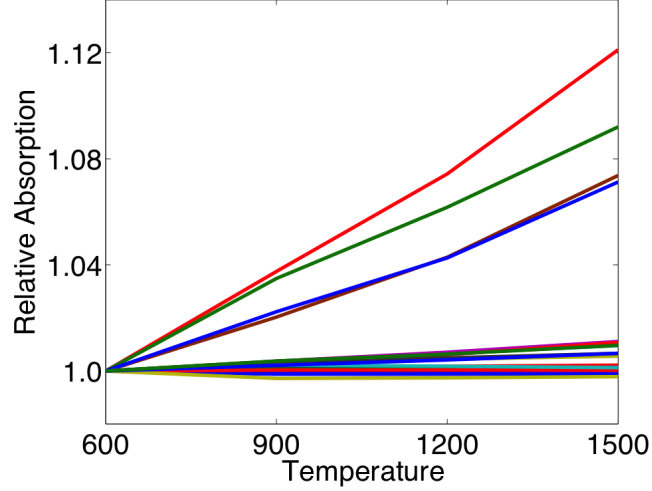


Figure 1: Temperature dependence of 23 group homogenized absorption cross sections for a 3.1% enriched PWR fuel pin. Each curve represents the absorption cross section in a different energy group relative to the group value at $T = 600K$.

It is possible to include some information about the temperature feedback effect on cross sections without performing a full cross section processing step at each function evaluation. One such approach can be accomplished by noting that the strongest temperature feedback effect is due to an increase in absorption with increasing temperature (largely due to Doppler broadening in ^{238}U). As shown in Fig. 1, the temperature dependence of absorption cross sections, σ_a , is approximately linear over a wide range of temperatures (this dependence is often stated as being proportional to $\sqrt{T}$ [37], however, we feel that a linear fit more accurately represents our data). This suggests a modification to the Jacobian approximation of Eq. (30) which uses a linear approximation to the temperature dependence of the absorption cross section and neglects the temperature dependence of all other cross sections:

$$\widehat{\mathbf{J}}_{\lambda} \begin{pmatrix} \phi \\ \lambda \\ T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T) - \lambda \mathbf{B}_{\nu}(T) & -\mathbf{B}_{\nu}(T)\phi & \frac{\partial \sigma_a}{\partial T} \phi \\ \mathbf{R}_S \mathbf{R}_E \mathbf{B}_{\kappa}(T) & 0 & 0 \\ -\mathbf{R}_E \mathbf{B}_{\kappa}(T) & 0 & \frac{\partial (\mathbf{L}(T)T)}{\partial T} \end{bmatrix}. \quad (33)$$

The function evaluation corresponding to Eq. (33) is

$$\widehat{f}_\lambda \begin{pmatrix} \phi + \Delta\phi \\ \lambda + \Delta\lambda \\ T + \Delta T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T)(\phi + \Delta\phi) - (\lambda + \Delta\lambda)\mathbf{B}_\nu(T)(\phi + \Delta\phi) + \frac{\partial\sigma_a}{\partial T}\phi\Delta T \\ \mathbf{R}_S\mathbf{R}_E\mathbf{B}_\kappa(T)(\phi + \Delta\phi) - P^* \\ \mathbf{L}(T + \Delta T)(T + \Delta T) - \mathbf{R}_E\mathbf{B}_\kappa(T)(\phi + \Delta\phi) \end{bmatrix}. \quad (34)$$

Because the cross sections, and the absorption cross sections in particular, are strongly dependent on the boron concentration, an additional correction term to account for the change in absorption due to changes in the boron concentration must be added. With this term, the analog of Eq. (33) is

$$\widehat{\mathbf{J}}_C \begin{pmatrix} \phi \\ C \\ T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T, C) - \lambda_{\text{target}}\mathbf{B}_\nu(T, C) & \frac{\partial\sigma_a}{\partial C}\phi & \frac{\partial\sigma_a}{\partial T}\phi \\ \mathbf{R}_S\mathbf{R}_E\mathbf{B}_\kappa(T, C) & 0 & 0 \\ -\mathbf{R}_E\mathbf{B}_\kappa(T, C) & 0 & \frac{\partial(\mathbf{L}(T)T)}{\partial T} \end{bmatrix} \quad (35)$$

with a corresponding function evaluation of

$$\widehat{f}_C \begin{pmatrix} \phi + \Delta\phi \\ C + \Delta C \\ T + \Delta T \end{pmatrix} = \begin{bmatrix} \mathbf{A}(T, C)(\phi + \Delta\phi) - \lambda_{\text{target}}\mathbf{B}_\nu(T, C)(\phi + \Delta\phi) + \frac{\partial\sigma_a}{\partial T}\phi\Delta T + \frac{\partial\sigma_a}{\partial C}\phi\Delta C \\ \mathbf{R}_S\mathbf{R}_E\mathbf{B}_\kappa(T, C)(\phi + \Delta\phi) - P^* \\ \mathbf{L}(T + \Delta T)(T + \Delta T) - \mathbf{R}_E\mathbf{B}_\kappa(T, C)(\phi + \Delta\phi) \end{bmatrix}. \quad (36)$$

In writing the two derivatives separately, we are assuming that the temperature and boron derivatives are independent of each other. Although there will certainly be some slight dependence, because the temperature derivative is isolated to the fuel and the boron derivative is isolated to the moderator, it is expected that the dependence will be slight. The presence of the derivative terms in Eq. (35) is expected to remedy the singular Jacobian of Eq. (32). Adding the feedback of absorption sensitivities to the approximated residuals is described as MJFNK2 in Section 4.

4. Results

For the numerical experiments in this paper, the radiation transport equation is approximated using the simplified P_N (SP_N) angular approximation

[38, 39, 40]. These equations are discretized spatially using a finite volume approach as implemented in the Denovo package [41]. The simple nature of the SP_N equations offers significant advantages in the current study: the operators $\mathbf{A}$ and $\mathbf{B}$ from Eq. (9) can be explicitly constructed as sparse matrices and, therefore, algebraic preconditioners (algebraic multigrid, for instance) can be easily applied. Aside from the development of appropriate preconditioners, it is expected that similar behavior would be observed for other transport formulations (such as discrete ordinates). The cross sections used by the SP_N equations are generated by the XSPROC module of the SCALE package [21]. The heat transfer and subchannel equations are solved using the Advanced Multiphysics (AMP) package [42, 43]. The heat transfer equation is discretized using standard trilinear continuous Galerkin finite elements and the subchannel equations employ a finite difference approximation [18]. Validation of the pin heat transfer models in AMP has been performed through comparison to data from several experiments [44, 45]. Preliminary investigation into multiphysics coupling involving AMP and Denovo was performed in Ref. [46].

To test the behavior of different nonlinear solvers on a realistic problem, we consider the solution of CASL AMA Progression Problem 6 [4]. This problem consists of a single 17×17 PWR fuel assembly with 264 fuel pins containing 3.1% enriched UO_2 , 24 guide tubes, and a single central instrumentation tube. Eight spacer grids are located along the axial length of the assembly, as well as upper and lower assembly nozzles. A full description of the problem, including detailed material and geometric specifications, is contained in Ref. [4].

For the base configuration, we model the assembly at a power of 17.67 MW and 1300 ppm dissolved boron; the effect of power level on solver convergence will be studied later in this section. The base configuration uses a 252 energy group cross section library and the XSPROC module of the SCALE package to collapse these cross sections to 23 groups for Denovo SP_N calculations. Distinct cross sections are used for each fuel pin and each of 49 axial levels. An SP_3 angular order (containing two angular moments) is used for all calculations, along with P_1 scattering. A 2×2 spatial mesh per pin cell is used in the x-y plane, with a maximum axial mesh size of 2 cm, resulting in 290,156 mesh cells for the full assembly. The AMP heat transfer problem contains 15,504 mesh cells per fuel pin (over both the fuel and clad meshes), resulting in approximately 4.1 million total cells. A linear continuous finite element discretization of the heat transfer problem is used. All problems in

this study are executed in parallel on 289 processing cores, resulting in a decomposition of one fuel pin per core. All calculations were performed on the Oak Ridge Leadership Computing Facility Eos cluster, a 744-node Cray XC30 cluster.

Five different solver approaches are considered. First is a damped Picard iteration with the damping applied to the temperature component of the solution as described in Section 3. Except where otherwise noted, all calculations use a damping factor of $\omega = 0.5$ which appears to produce nearly optimal convergence behavior for a wide range of problems. Within Picard iteration, the k -eigenvalue problem is solved with a generalized Davidson eigensolver, which has been shown to be highly efficient for solving the k -eigenvalue problem [47, 48], and the heat transfer and subchannel flow equations are solved simultaneously using a JFNK approach. The second solver consists of Anderson acceleration applied to the fixed point map of Eq. (19). Inner iterations are performed exactly the same as in Picard iteration. Except when specifically stated, a mixing parameter of $\beta = 0.75$ is used. A storage depth (the number of previous residuals used by the Anderson acceleration algorithm) of $m = 2$ is used in all cases; varying this parameter yielded no improvements to either the rate of convergence or the robustness of the solver.

The next solution approach consists of a full JFNK solver using the nonlinear function evaluation described by Eq. (13). Recall that this selection requires performing full cross section processing at every *linear* iteration of the Krylov solver. The final two solvers are modified JFNK approaches based on the function evaluations in Section 3.3.2. The approach denoted by MJFNK1 corresponds to the function evaluation of Eq. (31), in which the temperature dependence of the cross sections is entirely neglected during a nonlinear iteration. The approach denoted by MJFNK2 corresponds to Eq. (34), in which a linear approximation to the temperature dependence of the absorption cross section is used during each nonlinear iteration. This linear approximation is determined by performing stand-alone XSPROC calculations at 600K and 1500K to compute pin-homogenized absorption cross sections (the clad and coolant are kept at the nominal value of 565K for these calculations). In both modified JFNK approaches, only a single cross section processing step is performed per nonlinear iteration. In all of the JFNK-based methods, a scale factor of 10^{-3} is applied to the component of the residual corresponding to Eq. (10), effectively converting the units of the power from AMP's native W/kg to W/g. This scaling brings the magnitude of the residual more in line with the other residual components and provides

a slight improvement in convergence behavior. All other components are left unscaled.

Preconditioning of the radiation transport problem is accomplished using a physics-based multigrid in energy preconditioner [49], and the heat transfer problem is preconditioned using an algebraic multigrid approach from the Trilinos ML package [50]. Within Picard and Anderson, the transport preconditioner is used directly for subspace expansion within the generalized Davidson solver, and the heat transfer preconditioner is applied to the linear solver within the JFNK solver. In the JFNK approaches, a block diagonal preconditioner is formed using the individual physics preconditioners. Because the preconditioner selection is the same in all cases, we believe this presents the fairest possible comparison of the solvers. The nonlinear stopping tolerance for all cases is $\tau = 10^{-4}$, although the stopping criterion is applied slightly differently for each solver. For Picard iteration and Anderson acceleration, the stopping criterion is when the relative L_2 norm of the update for each variable (temperature and power) separately is less than the prescribed tolerance, i.e.,

$$\frac{\|T^{k+1} - T^k\|_2}{\|T^{k+1}\|_2} < \tau, \text{ and} \quad (37)$$

$$\frac{\|P^{k+1} - P^k\|_2}{\|P^{k+1}\|_2} < \tau. \quad (38)$$

For the JFNK-based methods, the absolute length-scaled norm of the function defined by Eq. (13) is used. A brief numerical justification of the equivalence of these differing stopping criteria will be provided later in this section. For Picard and Anderson iteration, the convergence tolerance is fixed at 10^{-5} for the heat transfer problem and at 10^{-7} for the transport eigenvalue problem. For the JFNK methods, a fixed linear solver tolerance of 10^{-5} is used. More sophisticated inner stopping criteria have been developed to reduce the time to solution [51] and in some cases even improve the robustness of the nonlinear solver [52]. Optimization of inner stopping criteria remains an area for future investigation.

Figure 2 shows the convergence behavior for each of the solvers considered. Picard iteration demonstrates the expected linear convergence rate. Anderson acceleration displays a more erratic convergence curve, although the average rate of convergence is quite comparable to Picard. Convergence for the full JFNK method is quadratic, as expected for a Newton method.

MJFNK1 follows the convergence behavior of the full JFNK for the first few iterations before deviating and ultimately converging linearly with a rate similar to that of Picard iteration. MJFNK2 follows the convergence rate of full JFNK even more closely, and although it eventually deviates from quadratic convergence at small residual norms, it still converges at a very rapid rate. The deviation from quadratic convergence is expected because the Newton correction equation has been altered through the use of the approximate function evaluations; the departure from quadratic convergence occurs when the residual norm is approximately the same magnitude as the error between the true and approximate functional evaluations. **The reason for the erratic behavior of Picard iteration and Anderson acceleration at residual norms below 10^{-6} is not clear.** Although the convergence tolerances used to generate this plot are several orders of magnitude smaller than what is often applied to such multiphysics problems, there are certain applications (e.g., sensitivity and uncertainty quantification) that may require the ability to converge problems very accurately.

Table 1 shows the effect of the stopping criteria on the time to solution. The coarsest tolerance is representative of a typical value for an engineering-scale calculation, and the finer tolerances are indicative of possible values used in sensitivity studies. This study clearly displays the issue with a “pure” JFNK implementation that was described in Section 3.3.2: although the convergence is very favorable with respect to the number of nonlinear iterations, each iteration is very costly due to the cross section processing that must be done at every linear iteration. The modified JFNK approaches avoid most of this cross section processing expense, resulting in much lower time to solution. MJFNK2, in particular, performs exceptionally well, resulting in a smaller time to solution than any other approach at all convergence tolerances. The fast convergence rate is particularly noticeable at tighter tolerances.

To clearly demonstrate the reason for the poor timing behavior of the full JFNK approach, and also to compare the merits of the different solvers, we now look at varying the energy resolution of the problem, both in the cross section processing and in the construction of the SP_N matrices. This allows us to vary the cost of cross section processing from (relatively) cheap to very expensive. Figure 3 shows the overall timing results for each solver, and Tables 2–4 provide a breakdown of the timing by individual physics component. The most obvious trend is the sharp increase in the JFNK run time as the number of XSPROC energy groups is increased. With only eight energy groups, JFNK is reasonably competitive with (albeit still slower

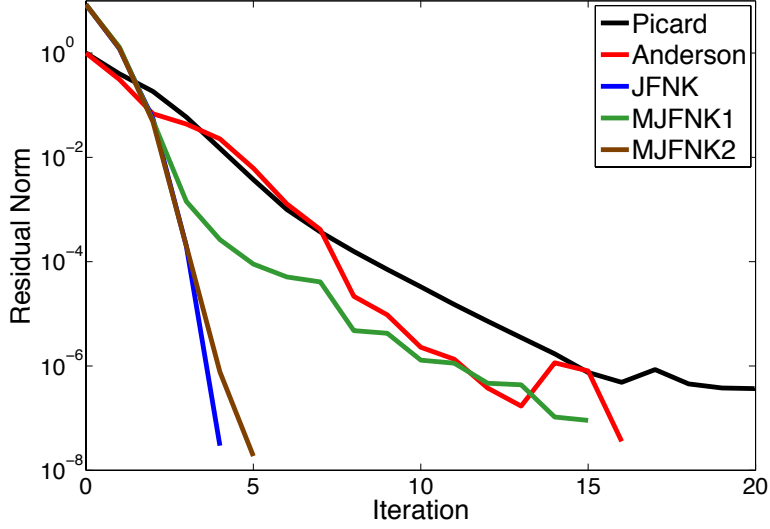


Figure 2: Nonlinear convergence behavior for different solvers.

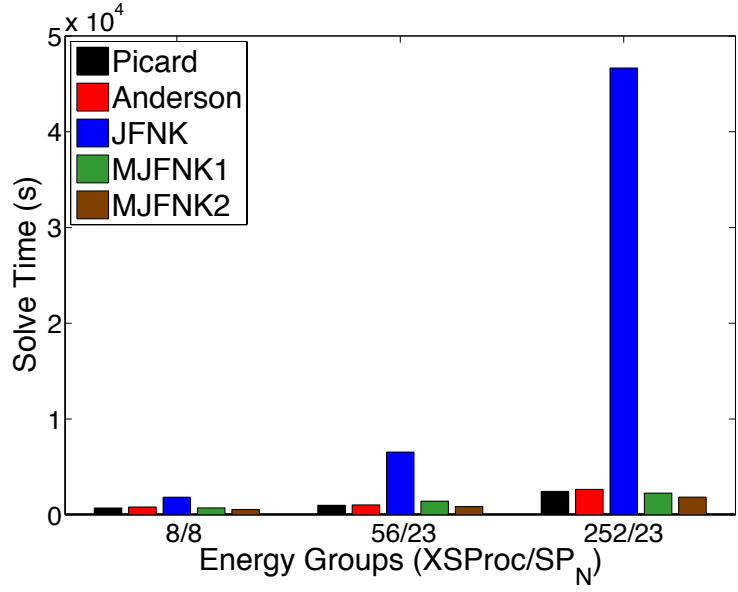
Method	Convergence Tolerance			
	10^{-4}	10^{-5}	10^{-6}	10^{-7}
Picard	2437	3197	3981	DNC
Anderson	2647	3204	3490	4603
JFNK	46653	46653	46653	46653
MJFNK1	2263	3684	6097	7930
MJFNK2	1846	1846	1846	2283

Table 1: Solver timing in seconds as function of convergence tolerance. DNC indicates that the method failed to achieve the prescribed tolerance after 50 iterations.

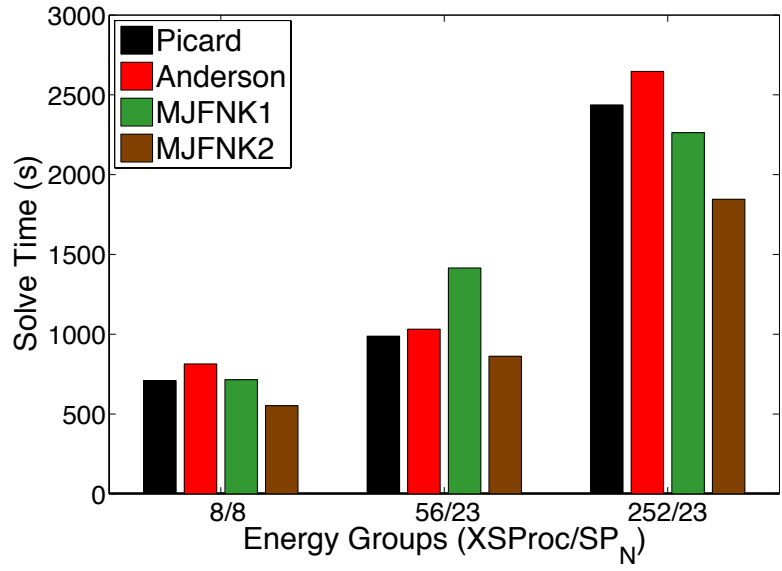
than) the other methods, but, with 252 XSProc groups, JFNK is more than 25 times slower than any other solver. Because of the poor performance with the 252 group library, which is the preferred approach for engineering analyses [4, 33], the JFNK approach is excluded from consideration in the remaining studies in this paper.

The remaining solvers all result in somewhat similar behavior over the range of energy group structures. In all cases, MJFNK2 is the most efficient solver, typically 25–30% faster than Picard iteration. In general, the JFNK-based approaches result in less time being spent in cross section processing and the thermal diffusion components of the calculation, but modest increases in the time spent in the SP_N portion of the calculation. It should be noted that the generalized Davidson eigenvalue solver used in Picard and Anderson was shown in Refs. [47, 48] to be highly efficient for solving the S_N k -eigenvalue problem, and the same conclusion was made for the SP_N version of the problem considered here in Ref. [49]. For stand-alone neutronics solvers not utilizing such an efficient approach, it is very possible that a reduction in computational effort for the transport portion of the problem could also be reduced relative to Picard. Some caution should be taken when interpreting the individual timings in Tables 2–4, because physics operators (and particularly preconditioning) are applied differently in Picard/Anderson versus the JFNK-based methods. Therefore, a perfect comparison is generally not possible. We have made an effort to separate the different components consistently, but some cases (particularly the SP_N component) result in significantly different times per application of the operator between the methods. Additionally, operations such as mesh generation, reading of input files, writing of output files, and mesh transfer operations are included in the total solution time but not in any of the individual physics components. Thus the sum of the components may be significantly less than the total run time. **In particular, mesh transfer operations are performed more frequently in the JFNK-based methods (once per linear iteration rather than once per nonlinear iteration with Picard), resulting in more time not accounted for in the individual physics components.**

The convergence behavior of Picard iteration and Anderson acceleration as a function of damping/mixing parameter at several different power levels is shown in Fig. 4. The Picard curve shows the same general shape as the corresponding plot from Ref. [46] and is consistent with the behavior observed in Refs. [1, 2, 3], all of which reported using damping factors between 0.3 and 0.6. An interesting feature of this curve is that the convergence be-



(a) All Solvers



(b) JFNK Excluded

Figure 3: Convergence behavior with different energy resolution.

Method	Total	SP _N	Thermal	XSProc
Picard	709 (9)	18 (267)	615 (283)	43 (9)
Anderson	814 (10)	21 (319)	704 (321)	52 (11)
JFNK	1836 (4)	54 (249)	395 (249)	1244 (249)
MJFNK1	715 (5)	64 (296)	457 (296)	27 (6)
MJFNK2	552 (4)	46 (219)	339 (219)	23 (5)

Table 2: Timing in seconds (operator applies) by component for 8 group XSProc, 8 group SP_N.

Method	Total	SP _N	Thermal	XSProc
Picard	988 (9)	91 (329)	638 (295)	206 (9)
Anderson	1032 (8)	96 (347)	676 (308)	205 (9)
JFNK	6545 (4)	165 (234)	393 (234)	5705 (234)
MJFNK1	1415 (6)	284 (429)	669 (429)	160 (7)
MJFNK2	862 (4)	154 (236)	366 (236)	114 (5)

Table 3: Timing in seconds (operator applies) by component for 56 group XSProc, 23 group SP_N.

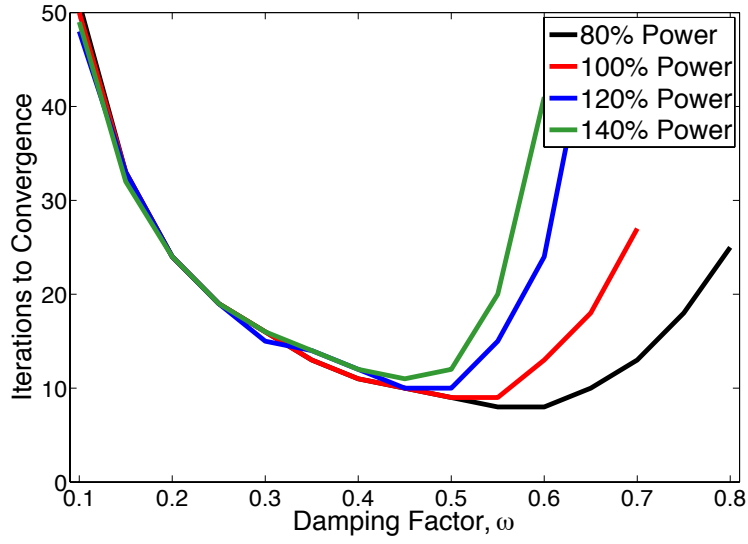
Method	Total	SP _N	Thermal	XSProc
Picard	2437 (9)	81 (296)	470 (296)	1507 (9)
Anderson	2647 (9)	92 (332)	483 (305)	1673 (10)
JFNK	46653 (4)	173 (252)	417 (252)	45472 (252)
MJFNK1	2263 (5)	210 (323)	499 (323)	1003 (6)
MJFNK2	1846 (4)	145 (224)	346 (224)	835 (5)

Table 4: Timing in seconds (operator applies) by component for 252 group XSProc, 23 group SP_N.

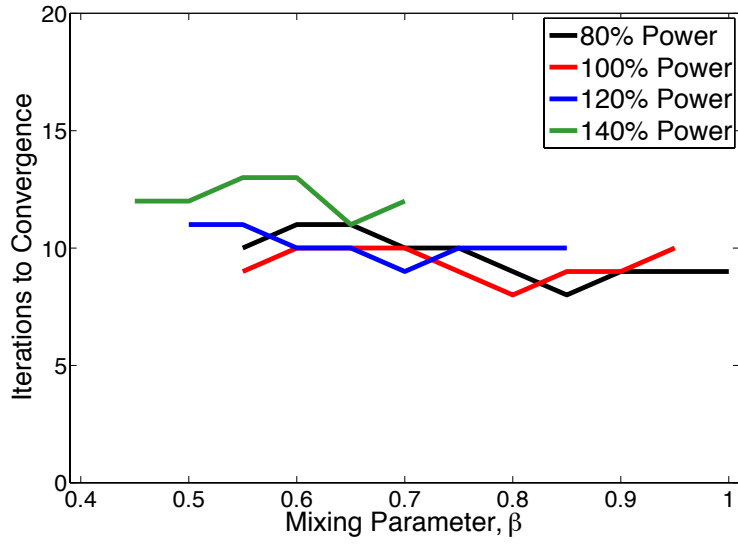
havior depends on the power level, with high power levels requiring smaller damping factors for optimal convergence. Anderson acceleration exhibits a less dramatic dependence on the mixing parameter, β . For a given power level, the number of iterations to convergence varies by only 2 or 3 iterations within a given range of mixing parameters, and convergence is not achieved outside of this window. The locations where Anderson acceleration does not converge do not represent true divergence of the method, but rather, the failure to converge occurs due to the finite bounds on material models being violated by the estimate of the solution at some iteration. It is very possible that either extending the material models or modifying the solution vector to remain within the bounds of the models would allow Anderson acceleration to converge. The reason for these sharp bounds on the window of convergence for the Anderson mixing parameter is not well understood. As with Picard iteration, the region of convergence shifts slightly as the power level increases, and the number of iterations required to converge also increases slightly.

Table 5 provides the time required to achieve convergence for each solution strategy at four different power levels. The time required for Picard iteration and Anderson acceleration to converge is approximately constant for most power levels, showing a slight upward trend at higher power levels, indicating that stronger coupling is present at high power. MJFNK1 performs very well at low power levels, reducing runtimes relative to Picard or Anderson. At high power levels, however, the time required for convergence greatly increases, reaching nearly twice the runtime of Picard at 140% power. This behavior is easily understood by noting that the low heat generation rate at low power results in relatively small changes in temperature, and therefore the effect of neglecting the temperature dependence of cross sections is not too large. High power levels, however, produce large temperature variations and a corresponding degradation in convergence due to not capturing the effects of changes in cross sections. The second modified JFNK approach performs very well across all power levels, resulting in the fastest time to solution for all cases and displaying very little variation with power level, even out to 140% power.

Table 6 shows the behavior of the solvers for the solution of the critical boron search problem described in Section 2.2 compared to the standard k -eigenvalue form of the problem. The target eigenvalue for the boron search is $\lambda_{\text{target}} = \frac{1}{1.2}$. Picard iteration uses the approach defined by Eq. (14), and Anderson acceleration uses a fixed point map based on that equation. A



(a) Picard Iteration



(b) Anderson Acceleration

Figure 4: Convergence behavior at different power levels.

Method	Power Level			
	80%	100%	120%	140%
Picard	2155	2437	2728	3020
Anderson	2499	2508	2732	3297
MJFNK1	1890	2263	4012	5417
MJFNK2	1466	1846	1839	1863

Table 5: Solver timing in seconds as function of power level.

Method	<i>k</i> -eigenvalue		Boron Search	
	Iterations	Time (s)	Iterations	Time (s)
Picard	9	2437	12	3056
Anderson	9	2647	9	2770
MJFNK2	4	1846	4	1945

Table 6: Solver comparison for boron search problem.

fixed value of $\frac{\partial \rho}{\partial C} = 8 \times 10^{-5}$ (as defined in Section 3.1) is used in these calculations. As discussed in Section 3.3.2, the MJFNK1 approach results in a singular Jacobian for the boron search problem and therefore was excluded from consideration. No attempt was made to modify the function evaluation to produce a nonsingular Jacobian, although it may be possible to do so. MJFNK2 requires not only an approximation of the temperature derivative of the absorption cross section, but also an approximation of the corresponding derivative with respect to boron concentration as seen in Eq. (36). This value was computed by performing cross section evaluations at 0 ppm and 2000 ppm boron and using the resulting slope as the derivative. Relative to the standard *k*-eigenvalue form of the coupled problem, Picard requires a few extra iterations to converge, indicating a slightly stronger coupling between physics in this problem. Anderson acceleration and MJFNK2 show much more robustness on this problem, displaying almost no variation in convergence behavior when moving from the standard eigenvalue search to the boron search.

It was noted earlier in this section that a different stopping criteria must be applied for JFNK-style methods as compared to the more standard criteria that are applied within Picard iteration and Anderson acceleration. To

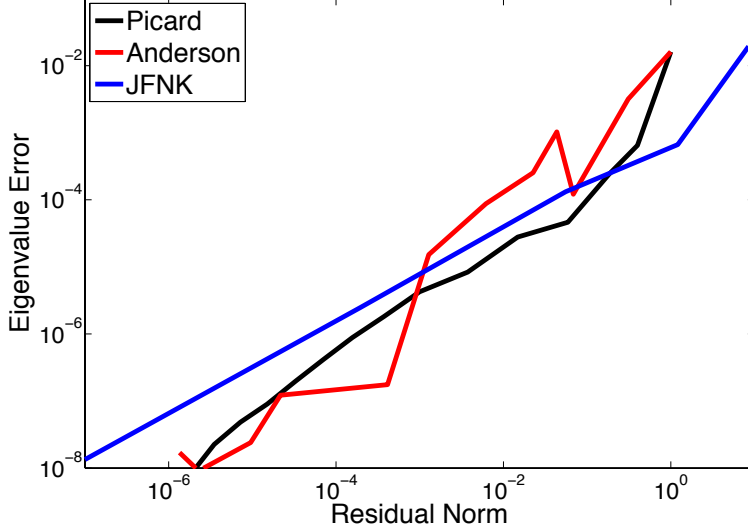


Figure 5: Relationship between residual norm and eigenvalue accuracy.

evaluate whether these differing stopping criteria are appropriate, we plot the error in the computed eigenvalue in Fig. 5 as a function of the residual norm for the different solver types. Because all of the JFNK-style methods evaluate convergence based on identical criteria, only a single curve is used to represent their collective behavior. The reference eigenvalue is the mean of the values computed by these three methods at a convergence tolerance of $\tau = 10^{-7}$, which were in agreement to nearly eight digits. These curves indicate that while perfect agreement is clearly not possible, comparable accuracy is achieved for all methods. At most residual norm values, the discrepancy in the eigenvalue accuracy amongst the methods is almost always less than an order of magnitude.

A brief mention should be made of the memory usage of the various solution approaches. It should be noted that all vectors used were stored as double precision arrays. Picard iteration uses virtually no memory on its own (only a single extra copy of the solution vector to determine convergence), and therefore the memory usage is determined by the individual solvers. In this study, the JFNK solver used by the heat transfer solver internally uses a GMRES linear solver, which requires a subspace with up to 20 vectors containing the thermal solution. The generalized Davidson solver used in the SP_N calculations requires multiple subspaces, which combined requires the

storage of up to 75 vectors of the length of the SP_N solution vector. Memory requirements for Anderson acceleration are equivalent to Picard, except that a small number of additional vectors (in this study, 2) are used by the solver. For the JFNK-based methods, the primary memory requirement is through the GMRES linear solver used. In this study, no restarting was employed. This resulted in a subspace size of as many as 80 vectors, each having the combined length of the thermal and SP_N solutions. Note that all of the GMRES linear solvers, as well as the generalized Davidson solver, offer the possibility of more aggressive restarting capabilities to limit the size of the subspaces that are used, typically at a cost of performing a small number of additional iterations to reach the same convergence criteria. In addition to memory associated with solver subspaces, there is an additional memory cost associated with forming the problem operators and corresponding preconditioners. These costs, however, are consistent across every solver option because consistent parameters and preconditioning options were used.

5. Conclusions

In this study we have provided an assessment of several different non-linear solvers for use in problems involving coupled neutronics and thermal hydraulics. In particular, comparisons of damped Picard iteration, Anderson acceleration, and Jacobian-free Newton-Krylov have been performed. Because a naïve implementation of JFNK results in performing a very large number of cross section processing steps, two modified variants of JFNK have been introduced that only require processing cross sections at each nonlinear iteration. Numerical results on CASL AMA Problem 6 indicate that if on-line generation of cross sections represents a large portion of the runtime of a calculation, then a direct JFNK implementation results in a prohibitively large number of cross section processing steps. It has been shown that approximations to the nonlinear operator that avoid cross section processing can largely preserve the fast convergence rate of JFNK without the overhead. Although the computational gains for Newton methods relative to Picard iteration were modest in this study, it is evident that the convergence behavior is more robust than that of Picard iteration and is not subject to the selection and optimization of a damping parameter.

Although Anderson acceleration resulted in similar behavior to Picard iteration with respect to both time to solution and robustness, further investigation is warranted in this area based on the success of the solver in other

multiphysics areas, as well as the ease of extending existing Picard-style coupling schemes to take advantage of Anderson acceleration. The failures of Anderson acceleration in this study were invariably due to limits on the extents of the material models in the various physics operators. Therefore, it may be possible to either extend the bounds of these models or enable a safeguarded line search within Anderson acceleration to limit the solution to the physical bounds of the models. This has the potential to greatly improve the robustness of the Anderson acceleration approach. Future work is necessary to develop more sophisticated preconditioners that may further increase the computational efficiency of Newton-like methods. Application of the MJNFK2 approach to boiling water reactors or other nuclear reactor designs would require re-evaluating the particular physics approximations that are relevant and appropriate for a given design.

Acknowledgments

This research was supported by the Consortium for Advanced Simulation of Light Water Reactors (www.casl.gov), an Energy Innovation Hub (<http://www.energy.gov/hubs>) for Modeling and Simulation of Nuclear Reactors under U.S. Department of Energy Contract No. DE-AC05-00OR22725.

This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

This research was supported by NSF Grant DMS-1406349.

References

- [1] X. Liu, X. Cheng, Thermal-hydraulic and neutron-physical characteristics of a new SCWR fuel assembly, *Annals of Nuclear Energy* 36 (2009) 28–36.
- [2] L. Monti, T. Schulenberg, Coupled ERANOS/TRACE system for HP-WLR 3 pass core analyses, in: *International Conference on Mathematics, Computational Methods & Reactor Physics*, Saratoga Springs, NY, 2009.

- [3] J. Yan, et al., Coupled computational fluid dynamics and MOC neutronic simulations of westinghouse PWR fuel assemblies with grid spacers, in: 14th International Topical Meeting on Nuclear Reactor Thermal Hydraulics, Toronto, ON, 2011.
- [4] S. Palmtag, Coupled single assembly solution with VERA (Problem 6), Tech. Rep. CASL-U-2013-0150-000, Consortium for Advanced Simulation of LWRs (2013).
- [5] S. Palmtag, Demonstration of neutronics coupled to thermal-hydraulics for a full-core problem using VERA, Tech. Rep. CASL-U-2013-0196-000, Consortium for Advanced Simulation of LWRs (2013).
- [6] B. Kochunas, D. Jabaay, B. Collins, T. Downar, Demonstration of neutronics coupled to thermal-hydraulics for a full-core problem using COBRA-TF/MPACT, Tech. Rep. CASL-U-2014-0051-000, Consortium for Advanced Simulation of LWRs (2014).
- [7] C. Chauliac, et al., NURESIM – a European simulation platform for nuclear reactor safety: Multi-scale and multi-physics calculations, sensitivity and uncertainty analysis, Nuclear Engineering and Design 241 (9) (2011) 3416–3426.
- [8] C. T. Kelley, Iterative Methods for Linear and Nonlinear Equations, Vol. 16 of Frontiers in Applied Mathematics, SIAM, Philadelphia, PA, 1995.
- [9] D. Knoll, D. Keyes, Jacobian-free Newton-Krylov methods: a survey of approaches and applications, Journal of Computational Physics 193 (2004) 357–397.
- [10] V. S. Mahadevan, High resolution numerical methods for coupled non-linear multi-physics simulations with applications in reactor analysis, Ph.D. thesis, Texas A&M University (2010).
- [11] J. Knoll, H. Park, C. Newman, Acceleration of k -eigenvalue/criticality calculations using the Jacobian-free Newton-Krylov method, Nuclear Science and Engineering 167 (2) (2011) 133–140.

- [12] D. Gill, Y. Azmy, Newton's method for solving k -eigenvalue problems in neutron diffusion theory, Nuclear Science and Engineering 167 (2) (2011) 141–153.
- [13] D. Y. F. Kastanya, P. J. Turinsky, Development and implementation of a Newton-BICGSTAB iterative solver in the FORMOSA-B BWR core simulator code, Nuclear Science and Engineering 150 (2005) 56–71.
- [14] J. C. Ragusa, V. S. Mahadevan, Consistent and accurate schemes for coupled neutronics thermal-hydraulics reactor analysis, Nuclear Engineering and Design 239 (3) (2009) 566–579.
- [15] H. Park, D. A. Knoll, D. R. Gaston, R. C. Martineau, Tightly coupled multiphysics algorithms for pebble bed reactors, Nuclear Science and Engineering 166 (2) (2010) 118–133.
- [16] J. M. Pounders, R. Ingram, Natural equilibria in steady-state neutron diffusion with temperature feedback, in: International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, Sun Valley, ID, 2013.
- [17] D. R. Gaston, et al., Physics-based multiscale coupling for full core nuclear reactor simulation, Annals of Nuclear Energy 84 (2015) 45–54.
- [18] J. E. Hansel, J. C. Ragusa, S. Allu, M. A. Berrill, K. T. Clarno, Analysis of physics-based preconditioning for single-phase subchannel equations, in: International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2013), 2013.
- [19] M. Chadwick, M. Herman, P. Oblozinsky, M. Dunn, Y. Danon, A. Kahler, D. Smith, B. Pritychenko, G. A. adn R. Arcilla, R. Brewer, D. Brown, R. Capote, A. Carlson, ENDF/B-VII.1 nuclear data for science and technology: Cross sections, covariances, fission product yields and decay data, Nuclear Data Sheets 112 (12) (2011) 2887–2996.
- [20] E. E. Lewis, W. F. Miller, Jr., Computational Methods of Neutron Transport, American Nuclear Society, Inc., La Grange Park, Illinois, USA, 1993.
- [21] B. Rearden, R. Lefebvre, J. Lefebvre, K. Clarno, M. Williams, L. Petrie, U. Mertuyurek, Modernization enhancements in SCALE 6.2,

in: PHYSOR 2014 - The Role of Reactor Physics Toward a Sustainable Future, Kyoto, Japan, 2014.

- [22] SCALE: A comprehensive modeling and simulation suite for nuclear safety analysis and design, Tech. Rep. ORNL/TM-2005/39, Version 6.1, Oak Ridge National Laboratory, Oak Ridge, TN (2011).
- [23] D. R. Vondy, T. B. Fowler, G. W. Cunningham, VENTURE: A code block for solving multigroup neutronics problems applying the finite-difference diffusion-theory approximation to neutron transport, version II, Tech. Rep. ORNL-5062/R1, Oak Ridge National Laboratory (1977).
- [24] A. Dall’Osso, A neutron balance approach in critical parameter determination, *Annals of Nuclear Energy* 35 (9) (2008) 1686–1694.
- [25] D. G. Anderson, Iterative procedures for nonlinear integral equations, *J. Assoc. Comput. Machinery* 12 (1965) 547–560.
- [26] H. Fang, Y. Saad, Two classes of multisecant methods for nonlinear acceleration, *Numerical Linear Algebra with Applications* 16 (2009) 197–221.
- [27] P. A. Lott, H. F. Walker, C. S. Woodward, U. M. Yang, An accelerated picard method for nonlinear systems related to variably saturated flow, *Advances in Water Resources* 38 (2012) 92–101.
- [28] M. T. Calef, E. D. Fichtl, J. S. Warsa, M. Berndt, N. N. Carlson, Non-linear krylov acceleration applied to a discrete ordinates formulation of the k-eigenvalue problem, *Journal of Computational Physics* 238 (2013) 188–209.
- [29] H. F. Walker, P. Ni, Anderson acceleration for fixed-point iterations, *SIAM J. Numer. Anal.* 49 (4) (2011) 1715–1735.
- [30] A. Toth, C. T. Kelley, Convergence analysis for Anderson acceleration, submitted for publication.
- [31] R. S. Dembo, S. C. Eisenstat, T. Steihaug, Inexact Newton methods, *SIAM Journal on Numerical Analysis* 19 (2) (1982) 400–408.

- [32] P. N. Brown, Y. Saad, Hybrid Krylov methods for nonlinear systems of equations, *SIAM Journal on Scientific and Statistical Computing* 11 (1990) 450–481.
- [33] F. Franceschini, A. Godfrey, J. Kulesza, R. Oelrich, Westinghouse VERA test stand - zero power physics test simulations for the AP1000 PWR, Tech. Rep. CASL-U-2014-0012-001, Consortium for Advanced Simulation of LWRs (2014).
- [34] P. N. Brown, C. S. Woodward, Preconditioning strategies for fully implicit radiation diffusion with material-energy transfer, *SIAM Journal on Scientific Computing* (2001) 499–516.
- [35] P. N. Brown, H. F. Walker, R. Wasyk, C. S. Woodward, On using approximate finite differences in matrix-free Newton-Krylov methods, *SIAM Journal on Numerical Analysis* 46 (4) (2008) 1892–1922.
- [36] M. Tonks, D. Gaston, C. Permann, P. Millett, G. Hansen, D. Wolf, A coupling methodology for mesoscale-informed nuclear fuel performance codes, *Nuclear Engineering and Design* 240 (10) (2010) 2877–2883.
- [37] J. J. Duderstadt, L. J. Hamilton, *Nuclear Reactor Analysis*, John Wiley & Sons, 1976.
- [38] E. W. Larsen, J. E. Morel, J. M. McGhee, Asymptotic derivation of the multigroup P1 and simplified PN equations with anisotropic scattering, *Nuclear Science and Engineering* 123 (1996) 328–342.
- [39] P. S. Brantley, E. W. Larsen, The simplified P3 approximation, *Nuclear Science and Engineering* 134 (2000) 1–21.
- [40] E. W. Larsen, G. Thömmes, A. Klar, T. Götz, Simplified PN approximations to the equations of radiative heat transfer and applications, *Journal of Computational Physics* 183 (2) (2002) 652–675.
- [41] T. Evans, A. Stafford, R. Slaybaugh, K. Clarno, DENOVO: A new three-dimensional parallel discrete ordinates code in SCALE, *Nuclear Technology* 171 (2010) 171–200.
- [42] K. T. Clarno, et al., The AMP (Advanced MultiPhysics) nuclear fuel performance code, *Nuclear Engineering and Design* 252 (1) (2012) 108–120.

- [43] B. Philip, M. A. Berrill, S. Allu, S. P. Hamilton, R. S. Sampath, K. T. Clarno, G. Dilts, A parallel multi-domain solution methodology applied to nonlinear thermal transport problems in nuclear fuel pins, *Journal of Computational Physics* 286 (2015) 143–171.
- [44] A. M. Phillippe, K. T. Clarno, J. E. Banfield, L. J. Ott, B. Philip, M. A. Berrill, R. S. Sampath, S. Allu, S. P. Hamilton, A validation study of pin heat transfer for UO₂ fuel based on the IFA-432 experiments, *Nuclear Science and Engineering* 177 (3) (2014) 275–290.
- [45] A. M. Phillippe, K. T. Clarno, J. E. Banfield, L. J. Ott, B. Philip, M. A. Berrill, R. S. Sampath, S. Allu, S. P. Hamilton, A validation study of pin heat transfer for MOX fuel based on the IFA-597 experiments, *Nuclear Science and Engineering* 178 (2014) 172–185.
- [46] S. Hamilton, K. Clarno, M. Berrill, T. Evans, G. Davidson, R. Lefebvre, R. Sampath, Multiphysics simulations for LWR analysis, in: *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering*, Sun Valley, ID, 2013.
- [47] S. Hamilton, Numerical solution of the k -eigenvalue problem, Ph.D. thesis, Emory University, Atlanta, GA (2011).
- [48] S. Hamilton, M. Benzi, A Davidson method for the k -eigenvalue problem, *Transactions of the American Nuclear Society* 105 (2011) 432–434.
- [49] S. P. Hamilton, T. M. Evans, Efficient solution of the simplified P_N equations, *Journal of Computational Physics* 285 (2015) 155–170.
- [50] M. Gee, C. Siefert, J. Hu, R. Tuminaro, M. Sala, ML 5.0 smoothed aggregation user’s guide, Tech. Rep. SAND2006-2649, Sandia National Laboratories (2006).
- [51] S. C. Eisenstat, H. F. Walker, Globally convergent inexact newton methods, *SIAM Journal on Optimization* 4 (2) (1994) 393–422.
- [52] R. P. Pawlowski, J. N. Shadid, J. P. Simonis, H. F. Walker, Globalization techniques for Newton-Krylov methods and applications to the fully coupled solution of the Navier-Stokes equations, *SIAM Review* 48 (4) (2006) 700–721.