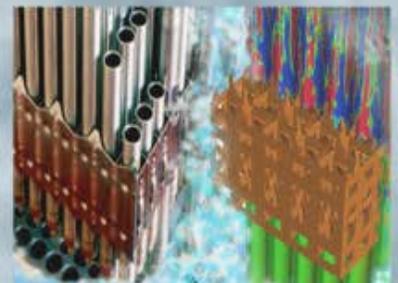
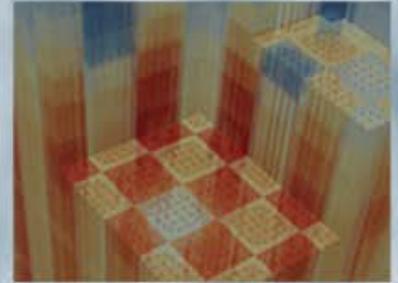


Analysis of Anderson Acceleration on a Simplified Neutronics/Thermal Hydraulics System

Alexander Toth and C. T. Kelley
North Carolina State University

Stuart Slattery, Steven Hamilton, and
Kevin Clarno
Oak Ridge National Laboratory

April 19, 2015



ANALYSIS OF ANDERSON ACCELERATION ON A SIMPLIFIED NEUTRONICS/THERMAL HYDRAULICS SYSTEM

A. Toth¹, C.T Kelley¹, S. Slattery², S. Hamilton², K. Clarno², and R. Pawlowski³

¹Department of Mathematics , North Carolina State University , Box 8205 , Raleigh, NC 27695 ,
artoth@ncsu.edu; tim_kelley@ncsu.edu

² Oak Ridge National Laboratory* , One Bethel Valley Rd. , Oak Ridge, TN 37831 ,
slatterysr@ornl.gov; hamiltonsp@ornl.gov; clarnokt@ornl.gov

³ Sandia National Laboratories[†] , MS 0316, P.O. Box 5800 , Albuquerque, NM 87185 ,
rppawlo@sandia.gov

ABSTRACT

A standard method for solving coupled multiphysics problems in light water reactors is Picard iteration, which sequentially alternates between solving single physics applications. This solution approach is appealing due to simplicity of implementation and the ability to leverage existing software packages to accurately solve single physics applications. However, there are several drawbacks in the convergence behavior of this method; namely slow convergence and the necessity of heuristically chosen damping factors to achieve convergence in many cases. Anderson acceleration is a method that has been seen to be more robust and fast converging than Picard iteration for many problems, without significantly higher cost per iteration or complexity of implementation, though its effectiveness in the context of multiphysics coupling is not well explored. In this work, we develop a one-dimensional model simulating the coupling between the neutron distribution and fuel and coolant properties in a single fuel pin. We show that this model generally captures the convergence issues noted in Picard iterations which couple high-fidelity physics codes. We then use this model to gauge potential improvements with regard to rate of convergence and robustness from utilizing Anderson acceleration as an alternative to Picard iteration.

Key Words: Multiphysics coupling, Picard iteration, Anderson acceleration

1 INTRODUCTION

Accurate simulation of nuclear reactors requires the simultaneous solution of several interdependent physical systems. In particular, we are concerned with the interdependence of the distribution of neutrons throughout the core and the transfer of heat in the fuel and coolant regions. Currently, many couplings between physics codes for analysis of light water reactors are implemented as Picard (or fixed point) iterations [1–4]. In this sort of iteration, single sets of physics are sequentially

*This manuscript has been authored by the Oak Ridge National Laboratory, managed by UT-Battelle, LLC, under Contract No. DE-AC05-00OR22725 with the US Department of Energy. The US Government retains and the publisher, by accepting the article for publication, acknowledges that the US Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US Government purposes.

[†]Sandia National Laboratories is a multiprogram laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94-AL85000.

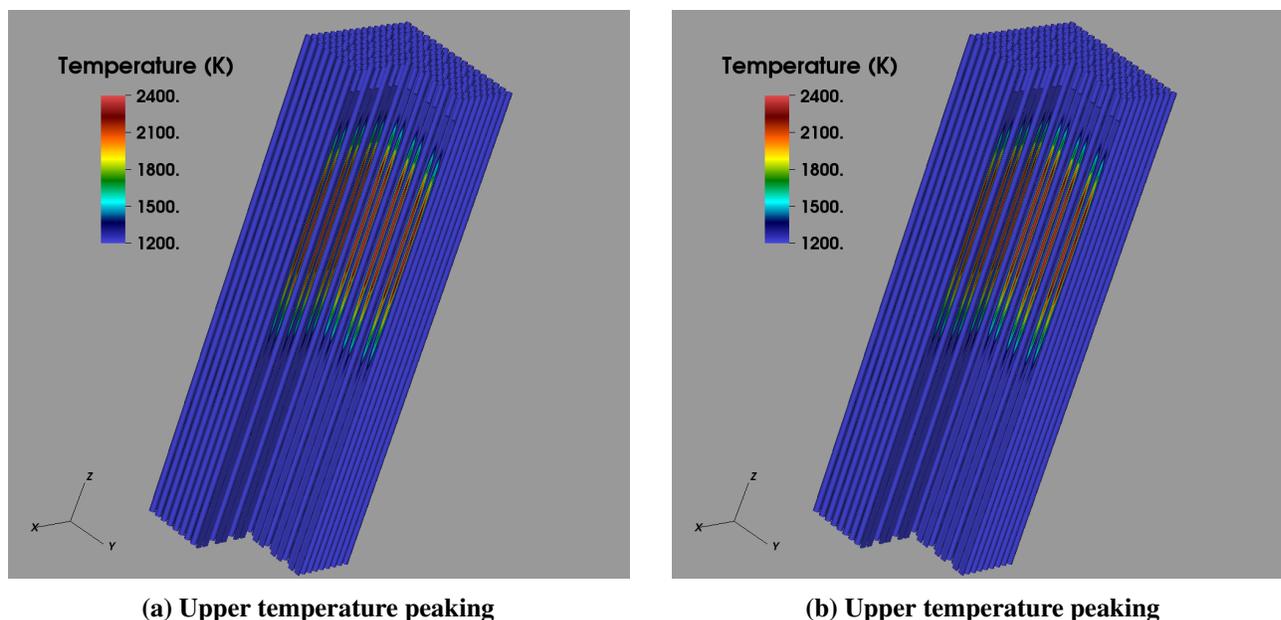


Figure 1. Oscillatory temperature shift in Insilico/AMP coupling

solved with the updated solution transferred to the other sets until some measure of convergence or failure is reached. The primary advantages of this sort of iteration are ease of implementation (no derivative information required) and the ability to utilize existing software for accurate single physics solves. However, Picard iteration suffers drawbacks due to slow convergence (usually at best q-linear), and poor robustness [5]. It has been observed that at high enough power, oscillatory behavior arises in the iteration which seems to contribute to the poor convergence behavior. This behavior is illustrated in Fig. 1. This figure shows the temperature profile for consecutive iterations of a single assembly calculation using Insilico for neutronics and AMP for fuel performance with integrated subchannel flow [4]. As the iteration proceeds, there is an oscillatory shift between the lower temperature peaking on the left and the upper temperature peaking on the right, and the iteration fails to converge. This oscillation is equally observed in various code couplings for 3×3 “mini-assembly” calculations [3] as well as for larger single assembly to full core problems [1, 4], so it does not seem that this behavior depends strongly on the codes being coupled or the number of pins considered. This issue is typically addressed by employing a numerical damping scheme. However, the behavior of the iteration is strongly dependent on the level of damping, and the level which leads to fastest convergence is dependent on the operating power level. The oscillatory behavior seems primarily to be an axial phenomenon, so we attempt to utilize a one-dimensional model in order to recreate and analyze this behavior. We additionally use this model to evaluate the potential of Anderson acceleration as an alternative solution method to Picard iteration. Anderson acceleration is a method which attempts to accelerate the convergence rate for fixed point iterations by utilizing secant information stored from previous iterations, and has been seen to be more robust and fast converging than Picard in several fields [6–9]. Its effectiveness in the context of coupled multiphysics problems has been largely untested to this point.

The rest of the paper is organized as follows: Section 2 describes the physics models governing

the simplified coupled system. Section 3 describes the implementation of Picard iteration and Anderson acceleration to solve this system. Section 4 displays numerical results illustrating how the simplified model captures convergence behavior observed in couplings between high-fidelity codes, and compares the convergence behavior of Picard and Anderson on this problem. Lastly, Section 5 presents conclusions and areas for future investigation.

2 PHYSICS MODELS

In this study, we model the interdependence between the neutron distribution and the temperature distributions within the fuel and coolant regions in a single fuel pin cell of height L . We denote the cross sectional area at a given axial height as $A(z)$. This area comprises of a circular fuel region with radius R_f inscribed within a square coolant region. We attempt to capture the behavior of the physical system by a one-dimensional model, so we consider equations which describe the axial behavior of the neutron distribution and fuel and coolant temperatures, which we treat as constant values at a given axial height.

For reactor analysis, the distribution of neutrons is governed by the Boltzmann transport equation, and codes for this purpose solve some approximation of this equation. As the purpose of this study concerns convergence behavior rather than simulation accuracy, we consider the one-group diffusion equation

$$-\nabla \cdot D(\vec{r}, T) \nabla \phi(\vec{r}) + (\Sigma_t(\vec{r}, T) - \Sigma_s(\vec{r}, T)) \phi = \frac{1}{k} \nu \Sigma_f(\vec{r}, T) \phi(\vec{r}), \quad (1)$$

where ϕ is the neutron scalar flux, Σ are the material cross sections, D is the diffusion coefficient, ν is the mean number of neutrons per fission, and k is the dominant eigenvalue. The cross section dependence on \vec{r} and T indicates that these quantities depend on both the material and its temperature. At the radial boundaries, we assume reflective boundary conditions, and axially we assume Marshak vacuum boundary conditions. To obtain a one-dimensional equation, we integrate (1) over the radial area $A(z)$. Introducing radially homogenized cross sections $\bar{\Sigma} = \frac{\int_{A(z)} \Sigma(\vec{r}, T) \phi(\vec{r}) dA}{\int_{A(z)} \phi(\vec{r}) dA}$ and diffusion coefficient $\bar{D} = 1/(3\bar{\Sigma}_t)$, and the radially integrated scalar flux $\bar{\phi} = \int_{A(z)} \phi(\vec{r}) dA$, this reduces to the following

$$-\frac{d}{dz} \bar{D} \frac{d\bar{\phi}}{dz} + (\bar{\Sigma}_t - \bar{\Sigma}_s) \bar{\phi} = \frac{1}{k} \nu \bar{\Sigma}_f \bar{\phi}. \quad (2)$$

The neutron distribution affects the other physical systems primarily through heat generated from fission. The linear heat generation rate is given by

$$q'(z) = \int_{A(z)} E_f \Sigma_f(\vec{r}) \phi(\vec{r}) dA = E_f \bar{\Sigma}_f(z) \bar{\phi}(z), \quad (3)$$

where E_f is the energy released per fission. As Equation (2) represents an eigenvalue problem, the eigenfunction has no explicit magnitude, and we choose to set the average linear power to a prescribed value P^*

$$\frac{1}{L} \int_0^L E_f \bar{\Sigma}_f \bar{\phi} dz = P^*. \quad (4)$$

Table I. Homogenized cross sections at reference fuel and coolant temperatures

| T_f | T_w | $\bar{\Sigma}_t$ | $\bar{\Sigma}_s$ | $\nu\bar{\Sigma}_f$ |
|-------|-------|------------------|------------------|---------------------|
| 565 | 565 | 0.655302 | 0.632765 | 0.0283063 |
| 1565 | 565 | 0.653976 | 0.631252 | 0.0277754 |
| 565 | 605 | 0.61046 | 0.589171 | 0.0265561 |

The radial homogenization has the effect of reducing the cross section values at a given axial height to a function of the fuel temperature T_f and the coolant temperature T_w at that height. We compute cross sections at a given fuel and coolant temperature by a linear interpolation using data precomputed at various reference fuel and coolant temperatures. The computed cross section values at the considered reference temperatures, which have been computed by the SCALE module XSPROC [10], are shown in Table I.

Next, we let the fuel temperature be governed by a simple relation derived from Newton's Law of Cooling [11], which states

$$q'' = h(T_f - T_w), \quad (5)$$

where q'' is the heat flux and h is the heat transfer coefficient. To relate q'' and q' we consider the differential length of the rod between axial heights z and $z + dz$. This corresponds to a heat transfer surface area of $2\pi R_f dz$. In this model, we assume all the power generated in the fuel is deposited radially into the coolant, so the power transferred through this area is given by $q' dz$. Hence, the heat flux is $q'' = \frac{q'}{2\pi R_f}$. Making this substitution gives the relation

$$2\pi R_f h(T_f - T_w) = q'. \quad (6)$$

The last set of physics that we consider is coolant flow. In this model, we assume that flow is only in the axial direction with a constant mass flow rate \dot{m} . We again consider a differential length dz about z . We let the change in temperature from z to $z + dz$, dT_w , be governed by the simplified steady-flow thermal energy equation [11], which states

$$\dot{m}c_p(T_w)dT_w = q, \quad (7)$$

where q is the power generation in this interval. The power transferred to the coolant is again $q = q' dz$, so we let the axial profile of the coolant temperature be governed by the differential equation

$$\dot{m}c_p(T_w)\frac{dT_w}{dz} = q'. \quad (8)$$

Given some inflow temperature T_{IN} , this can be solved to give the axial coolant temperature profile. We let specific heat values be given by a piecewise linear interpolation using computed data at reference temperatures and pressures given in [12].

3 SOLUTION METHODS

We seek $\bar{\phi}$, k , T_f , and T_w such that Equations (2), (4), (6), and (8) are simultaneously satisfied. In this section, we describe how Picard iteration and Anderson acceleration can be utilized to solve

this problem.

3.1 Picard Iteration

A standard method to solve this type of problem is Picard iteration, in which individual physics components are sequentially solved and the updated solution is transferred to the other physics. A straightforward example of this is shown in Algorithm 1, in which the neutronics, fuel temperature, and coolant temperature systems are alternately solved. A significant drawback of this sort of iteration scheme lies in its potentially poor convergence behavior. It has been observed that for problems of this type that oscillatory instability induced by certain error modes may arise leading to slow convergence or possibly divergence. In practice, this is addressed by introducing a numerical damping on the temperature iterates, as seen in Algorithm 2. The convergence behavior of this method depends strongly on the value of the damping factor ω , and it has been reported that the value which results in fastest convergence is typically in the range 0.3-0.6 [3].

Algorithm 1 Picard Iteration

Given initial temperature iterates $T_{f,0}, T_{w,0}$

for $j = 0, 1, \dots$ **do**

$$\text{Solve } -\frac{d}{dz}\bar{D}(T_{f,j}, T_{w,j})\frac{d\phi}{dz} + (\bar{\Sigma}_t(T_{f,j}, T_{w,j}) - \bar{\Sigma}_s(T_{f,j}, T_{w,j}))\phi = \frac{1}{k}\nu\bar{\Sigma}_f(T_{f,j}, T_{w,j})\phi$$

$$\text{Set } \bar{\phi}_{j+1} = \frac{P^*\phi}{\frac{1}{L}\int_0^L E_f \bar{\Sigma}_f(T_{f,j}, T_{w,j})\phi dz} \text{ and } q' = E_f \bar{\Sigma}_f(T_{f,j}, T_{w,j})\bar{\phi}_{j+1}$$

$$\text{Solve } \dot{m}c_p(T_{w,j+1})\frac{dT_{w,j+1}}{dz} = q'$$

$$\text{Set } T_{f,j+1} = T_{w,j+1} + \frac{q'}{2\pi R_f L}$$

end for

Algorithm 2 Damped Picard Iteration

Given initial temperature iterates $T_{f,0}, T_{w,0}$ and damping parameter ω

for $j = 0, 1, \dots$ **do**

$$\text{Solve } -\frac{d}{dz}\bar{D}(T_{f,j}, T_{w,j})\frac{d\phi}{dz} + (\bar{\Sigma}_t(T_{f,j}, T_{w,j}) - \bar{\Sigma}_s(T_{f,j}, T_{w,j}))\phi = \frac{1}{k}\nu\bar{\Sigma}_f(T_{f,j}, T_{w,j})\phi$$

$$\text{Set } \bar{\phi}_{j+1} = \frac{P^*\phi}{\frac{1}{L}\int_0^L E_f \bar{\Sigma}_f(T_{f,j}, T_{w,j})\phi dz} \text{ and } q' = E_f \bar{\Sigma}_f(T_{f,j}, T_{w,j})\bar{\phi}_{j+1}$$

$$\text{Solve } \dot{m}c_p(\hat{T}_w)\frac{d\hat{T}_w}{dz} = q'$$

$$\text{Set } \hat{T}_f = \hat{T}_w + \frac{q'}{2\pi R_f L}$$

$$\text{Set } T_{f,j+1} = (1 - \omega)T_{f,j} + \omega\hat{T}_f \text{ and } T_{w,j+1} = (1 - \omega)T_{w,j} + \omega\hat{T}_w$$

end for

The ordering of single physics solves in Algorithm 1 is not the only way to implement Picard iteration for this problem, and we may additionally consider alternating mappings. For instance, we may switch the order in which the temperature updates are computed, in which case the updated fuel temperature would be computed using the coolant temperature from the previous iteration. The ordering in Algorithm 1 in effect solves (6) and (8) for T_f and T_w as a coupled system given some q' , whereas this alternate ordering in a sense decouples these equations and treats them as separate physical systems. We have observed that this alternate map ordering displays modestly worse convergence behavior, so we will only consider the damped iteration given in Algorithm 2.

Algorithm 3 Anderson Acceleration

Given initial iterate u_0 , storage depth parameter $m \in \mathbb{N}$, and mixing parameter β
 Set $u_1 = G(u_0)$; $F_0 = G(u_0) - u_0$
for $k = 1, 2, \dots$ **do**
 Set $m_k = \min\{m, k\}$
 Set $F_k = G(u_k) - u_k$
 Determine $\alpha^{(k)} = (\alpha_0^{(k)}, \dots, \alpha_{m_k}^{(k)})$ which solves $\min_{\alpha = (\alpha_0, \dots, \alpha_{m_k})} \|\sum_{i=0}^{m_k} \alpha_i F_{k-m_k+i}\|$
 subject to the constraint $\sum_{i=0}^{m_k} \alpha_i = 1$
 Set $u_{k+1} = (1 - \beta) \sum_{i=0}^{m_k} \alpha_i^{(k)} u_{k-m_k+i} + \beta \sum_{i=0}^{m_k} \alpha_i^{(k)} G(u_{k-m_k+i})$
end for

3.2 Anderson Acceleration

Due to the limitations of Picard iteration, it is of interest to utilize a more robust and faster converging solution method. Anderson acceleration is an alternative which has been gaining attention in this regard. This method was first proposed in [13], and it (or the closely related Pulay mixing) has since become widely used in the context of self-consistent field calculations [6–8]. This method has also been used to some success in accelerating power iteration to solve the k-eigenvalue formulation of the neutron transport equation [9]. The Anderson acceleration algorithm for solving the fixed point problem $u = G(u)$ proceeds as shown in Algorithm 3. This method attempts to accelerate the convergence of Picard iteration by utilizing secant information from previous iterations in order to compute a better approximation to the solution. The storage depth parameter m determines the maximum depth of previous iterates which will be stored, and we refer to the algorithm with any particular m as Anderson- m . The mixing parameter β plays a similar role to the damping factor for Picard. This method is attractive in part because it requires no more information to implement than Picard iteration. The only requirement is the ability to evaluate the fixed point map G . This provides an advantage over Newton-like methods, as these require derivative information to compute the Newton step [5]. For many realistic problems of interest, obtaining analytic Jacobian information is infeasible. Jacobian-free Newton-Krylov (JFNK) methods remain an option, but as they require one residual evaluation per linear iteration, this has been seen to not be competitive without exploiting potentially problem specific features [4]. However, the only additional cost per Anderson iteration over Picard is the solution of the least-squares problem, and in most cases m is kept fairly small (less than 10). Supposing that the evaluation of G is fairly expensive, the cost of the least-squares solve is essentially negligible. In past work on this method, it has been shown that Anderson acceleration may be viewed as a sort of quasi-Newton method, which the authors in [14] refer to as a Type-II Broyden method. Additionally, it has been shown that when applied to linear functions, Anderson acceleration is “essentially equivalent” to GMRES iteration in the sense that the iterates of one method can be easily computed from those of the other [15]. More recently, local convergence results have been shown for the case where the (potentially nonlinear) fixed point map is a contraction in a neighborhood of the solution [16].

In order to implement Anderson acceleration for this problem, all that is required is to define the fixed point map G , and there is some flexibility in how to do this. We define a fixed point map such that we can write Algorithm 1 in the form $u_{j+1} = G(u_j)$, where u is some set of the state

variables. For this, we first define the following operators: $g_\phi(T_f, T_w)$ is the resulting scalar flux from solving (2) and (4) given cross sections evaluated with fuel and coolant temperature profiles T_f and T_w , $p(T_f, T_w) = E_f \Sigma_f(T_f, T_w) g_\phi(T_f, T_w)$ is the corresponding linear power distribution, $g_f(T_w, q') = T_w + \frac{q'}{2\pi R_f h}$ is the solution fuel temperature profile from (6) given coolant temperature T_w and linear power q' , and $g_w(q')$ is the resulting coolant temperature profile from solving (8) given linear power q' . Then, letting u be comprised of $\bar{\phi}$, T_f , and T_w , we define

$$G_1 \begin{pmatrix} \bar{\phi} \\ T_f \\ T_w \end{pmatrix} = \begin{pmatrix} g_\phi(T_f, T_w) \\ g_f(g_w(p(T_f, T_w)), p(T_f, T_w)) \\ g_w(p(T_f, T_w)) \end{pmatrix}. \quad (9)$$

Note that the evaluation of G_1 only depends on T_f and T_w , so we can additionally define another map with $\bar{\phi}$ eliminated from the primary state variables

$$G_2 \begin{pmatrix} T_f \\ T_w \end{pmatrix} = \begin{pmatrix} g_f(g_w(p(T_f, T_w)), p(T_f, T_w)) \\ g_w(p(T_f, T_w)) \end{pmatrix}. \quad (10)$$

In this map, the scalar flux computation is embedded in the computation of the linear power operator $p(T_f, T_w)$. These two fixed point maps produce the same result given the same $T_{f,0}$ and $T_{w,0}$ for Picard iteration, but may differ for Anderson. As storage may be a concern for realistic problems of interest, in forthcoming numerical tests we will consider Anderson applied to the fixed point map G_2 , eliminating the need to store scalar flux profiles from previous iterates.

4 RESULTS

We now present numerical results for solving the simplified model problem with Picard iteration and Anderson acceleration. For these tests, we select physical parameters to be approximately realistic. We let the dimensions of the fuel rod be given by $L = 360\text{cm}$ and $R_f = 0.5\text{cm}$. We set the pressure of the system to 15.5MPa , set a 100% power baseline at $P^* = 200 \frac{\text{W}}{\text{cm}}$, let the energy released per fission be $E_f = 191.4\text{MeV}$, and let the heat transfer coefficient be $h = 0.2 \frac{\text{W}}{\text{m}^2 \text{K}}$. For the flow model, we assume an incoming coolant temperature of $T_{IN} = 565\text{K}$ and a mass flow rate of $\dot{m} = 0.3 \frac{\text{kg}}{\text{s}}$. We discretize each of the systems on the same evenly spaced mesh consisting of $N = 201$ axial nodes, and we discretize Equations (2) and (8) by finite differences. In general, each of the systems need not be solved on the same mesh, but differing meshes only increases the complexity of data transfers between physical systems and should not significantly affect the convergence behavior of the coupled system. We solve the generalized eigenproblem resulting from discretization of (2) by the Generalized Davidson solver in the Trilinos package Anasazi, and we solve the nonlinear system from (8) by JFNK using the Trilinos package NOX [17]. To terminate each iteration, we require each of $\bar{\phi}$, k , T_f , and T_w to be sufficiently converged by requiring

$$\begin{aligned} \frac{\|\bar{\phi}_{j+1} - \bar{\phi}_j\|}{\|\bar{\phi}_0\|} &< \tau, & \frac{|k_{j+1} - k_j|}{k_0} &< \tau, \\ \frac{\|T_{f,j+1} - T_{f,j}\|}{\|T_{f,0}\|} &< \tau, & \frac{\|T_{w,j+1} - T_{w,j}\|}{\|T_{w,0}\|} &< \tau. \end{aligned} \quad (11)$$

For Anderson, $\bar{\phi}$ and k are the values computed internally in the evaluation of the fixed-point map. In the following tests, we let $\tau = 10^{-4}$. We begin with initial fuel and coolant profiles identically equal to T_{IN} .

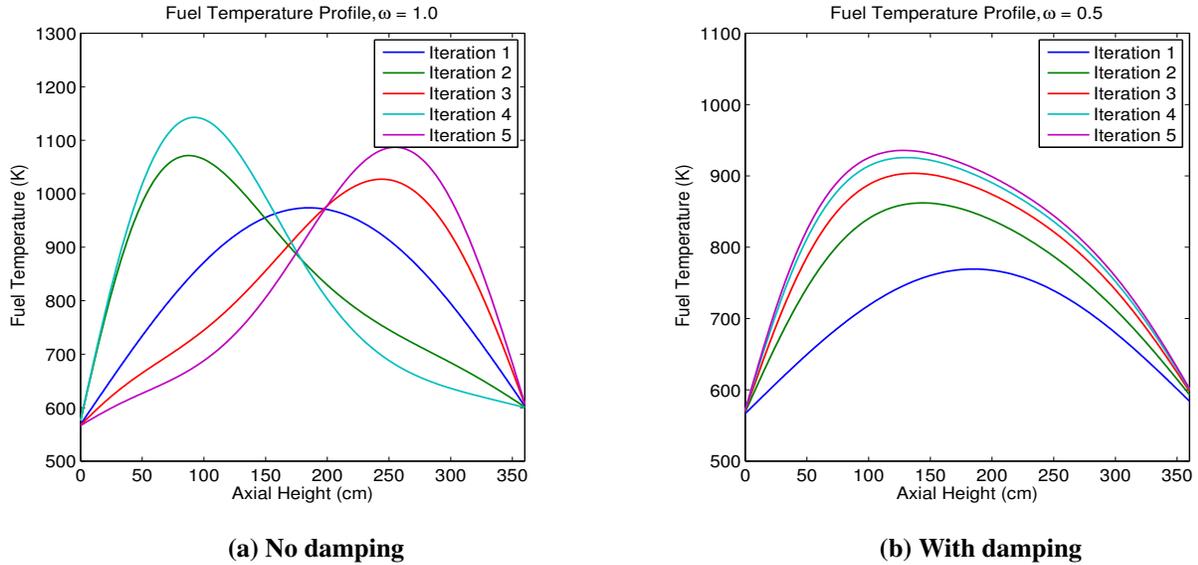


Figure 2. Fuel temperature behavior for Picard iteration, without and with damping

We first consider the Picard coupling to verify that the one dimensional model in fact recreates the convergence behavior previously noted to occur in high-fidelity couplings. Fig. 2a shows the temperature behavior for the one dimensional model without temperature damping. As in Fig. 1, we observe an oscillatory shift in the temperature distribution between a lower-peaked mode and an upper-peaked mode. Hence, it seems that this model seems to retain sufficient physics to recreate the desired oscillatory behavior. This behavior seems to occur due to the tight coupling between the temperature profiles and the linear power. We first note that as computed in this model the cross sections are decreasing functions of both coolant and fuel temperature. Initially, the rise in the coolant temperature over the height of the rod causes the lower region to be relatively optically thick. Hence there are more interactions and fissions in this region, so when the next linear power is computed, it displays a sharp peak in this region. The fuel temperature follows the power shape very closely, so this in turn leads to the lower peak in the fuel temperature. However, this spike in the fuel temperature then leads the lower region to become relatively optically thin, so the next linear power peak shifts to the upper region. The fuel temperature again follows, and the iteration oscillates in this manner. The oscillations push the peaks in power and fuel temperature profiles in the correct direction, but they repeatedly overshoot the solution and never find the correct shape. Fig. 2b shows the effect of damping on the convergence behavior of the fuel temperature profile. This shows how damping limits the change in the temperature profile from iteration to iteration, allowing the solution to approximate the proper shape while converging in magnitude. In Figure 3, we see the effect varying the damping parameter has on the number of Picard iterations to convergence. As expected, the iteration counts are very sensitive to the damping parameter. The optimal damping parameters we observe fall in the range 0.4 – 0.6 which agrees well with what has been reported previously. Additionally, the optimal damping levels are sensitive to the power level, and the number of iterations to convergence at the optimum increases with power. In general, this model seems to recreate the relevant convergence behavior observed in high-fidelity couplings.

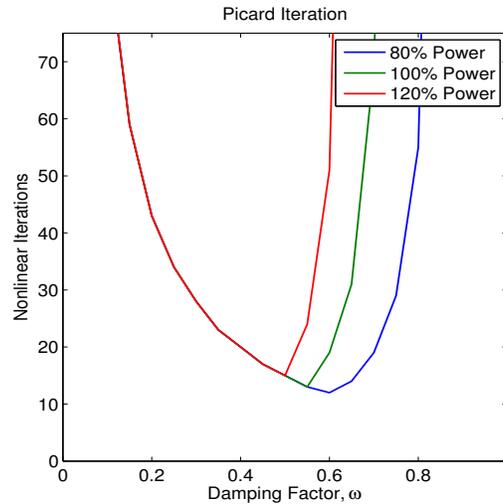
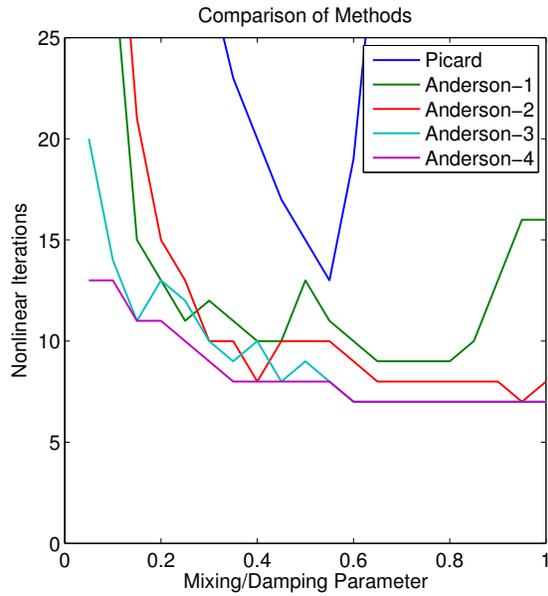
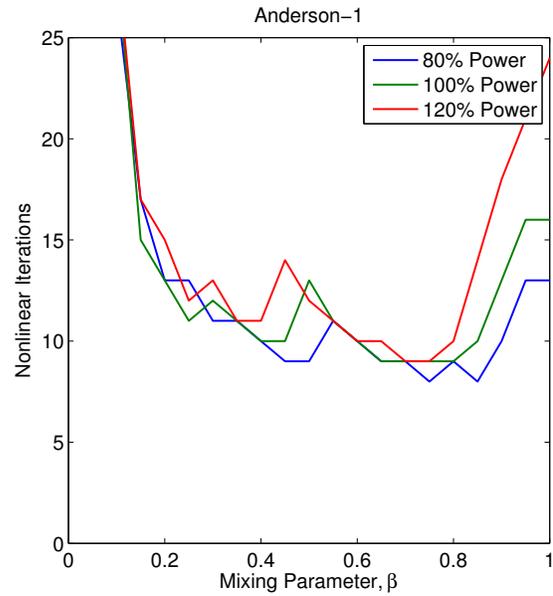


Figure 3. Picard iterations to convergence, varying damping factor

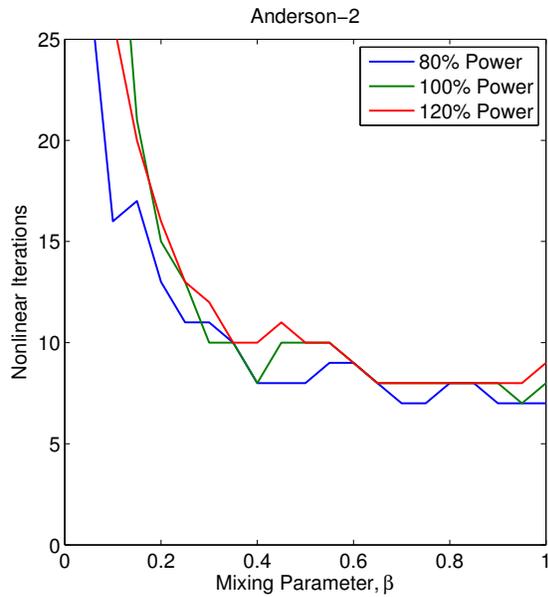
We next consider Anderson acceleration to solve the same problem. Fig. 4 illustrates the convergence behavior for Anderson varying the power levels, mixing parameter and storage depth. We see that Anderson converges in as few, generally fewer, iterations than the optimally damped Picard iterations over a fairly wide range of mixing parameters. In addition to at worst a modest improvement in terms of number of iterations to convergence, we observe a significant improvement in robustness. Whereas each Picard iteration required at least some level of damping to achieve convergence, each of the Anderson iterations was convergent for each mixing parameter, and the number of iterations to convergence is generally insensitive to the choice of mixing parameter. The iterations stagnate as the mixing parameter approaches zero, but away from zero, there is generally no obvious relation between the iteration counts and the mixing parameter. Anderson-1 also stagnates a bit for $\beta > 0.8$, but the iteration counts are fairly flat for mixing parameters between 0.3 and 0.8. For larger storage depths though, there is no stagnation for larger mixing parameters, and the interval between 0.3 and 1.0 represents a range of near optimal mixing parameters. As a result, the mixing parameter does not need to be tuned to obtain good performance. Next, we have noted that the optimal damping parameter for the Picard iterations depends on the power level. In addition to this, the number of iterations at the optimal damping level increases with the power. However, the number of Anderson iterations to convergence for a given mixing parameter does not depend as strongly on the power level. The range of mixing parameters over which Anderson works well does not depend strongly on the power level, and increasing power seems to increase iteration counts only marginally, if at all. With regard of the choice of storage depth, the results agree well with past experience, in that increasing storage depth improves performance to a point, after which additional storage depth provides little to no improvement [16]. We observe that Anderson-2 generally converges faster than Anderson-1, and Anderson-3 is faster than Anderson-2, but less noticeably. Anderson-3 seems moderately more insensitive to variations in the mixing parameter than Anderson-2, and improvements seem to stagnate beyond this point. Anderson-3 or Anderson-4 then seem optimal for this problem, as increasing the storage depth beyond this point provides negligible benefit at cost of additional storage.



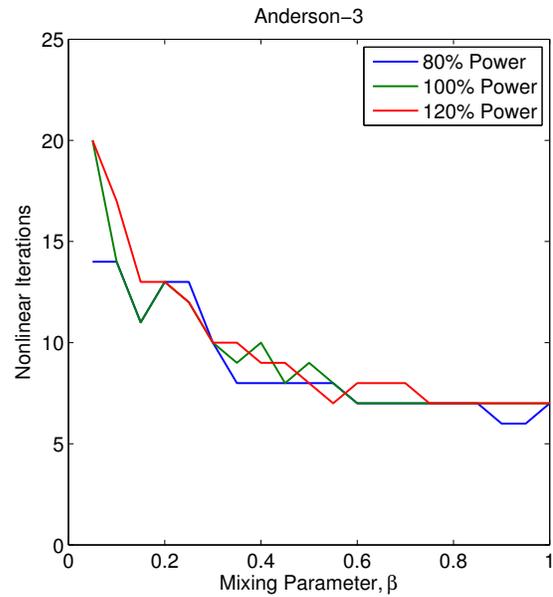
(a) Comparison of methods, 100% power



(b) Anderson-1



(c) Anderson-2



(d) Anderson-3

Figure 4. Nonlinear iterations to convergence for Anderson acceleration

5 CONCLUSIONS

In this study we have developed a simplified one-dimensional model for simulating the coupling between neutronics and thermal hydraulics in a single fuel rod, and showed that when solved by Picard iteration this model retains sufficient physics to display the convergence issues observed in couplings between high-fidelity codes. In particular, the iteration displays oscillatory instability that necessitates numerical damping in order to obtain a convergent iteration, and we observe the expected dependence iterations to convergence has on the level of damping. The observed iteration counts and range of damping factors which lead to convergence are similar to what has been reported in high-fidelity couplings at comparable power levels. This suggests that this simplified model should serve as a good surrogate for the purpose of analysis of the convergence behavior of Anderson acceleration in the context of multiphysics coupling in a LWR.

When utilizing Anderson acceleration for this problem, several of the disadvantages of Picard iteration are addressed. First, we generally see Anderson converge as fast or faster than optimally damped Picard over a fairly wide range of mixing parameters, and the iteration counts are relatively insensitive to changes in the mixing parameter. Second, the number of iterations to convergence and the optimal range of mixing parameters is not as strongly dependent on the power level as what is observed for Picard. Hence, a priori knowledge of an ideal mixing parameter is not necessary to obtain good performance. These results for Anderson acceleration are promising and suggest that utilizing this method in couplings between high-fidelity physics codes merits further study. A preliminary investigation with implementing Anderson acceleration for a coupling between Insilico and AMP has in fact been performed on a single fuel assembly in [4]. In this, it was observed that Anderson only converged over a limited range of power, with the iteration failing due to violating bounds in material property evaluations. It is worth noting that this study did not vary mixing parameters or storage depth for Anderson. When convergent, though, the iterations converged at a comparable rate to optimally damped Picard iteration, which agrees well with what was observed in the current study. It may then be of interest to determine if it is possible to address these material bound failures, or whether this simplified model may be modified in order to replicate and comprehend these failures.

6 ACKNOWLEDGMENTS

This research was supported by the Consortium for Advanced Simulation of Light Water Reactors (<http://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

7 REFERENCES

- [1] L. Monti and T. Schulenberg, "Coupled ERANOS / TRACE System for HPLWR 3 Pass Core Analyses," *International Conference on Mathematics, Computational Methods and Reactor Physics*, pp. 1–14, Saratoga Springs, NY, 2009.

- [2] J. Yan et al., “Coupled Computational Fluid Dynamics and MOC Neutronic Simulations of Westinghouse PWR Fuel Assemblies with Grid Spacers,” *14th International Topical Meeting on Nuclear Reactor Thermalhydraulics*, Toronto, Ontario, 2011.
- [3] S. Hamilton et al., “Multiphysics simulations for LWR analysis,” *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering*, Sun Valley, ID, 2013.
- [4] M. Berrill, K. Clarno, S. Hamilton, and R. Pawlowski, “Evaluation of Coupling Approaches,” Tech. Rep. L3:PHI.CMD.P8.01, Consortium for Advanced Simulation of LWRs (2014).
- [5] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, Society for Industrial and Applied Mathematics, Philadelphia, PA, frontiers edition, (1995).
- [6] P. Pulay, “Convergence Acceleration of Iterative Sequences. The Case of SCF Iteration,” *Chemical Physics Letters*, **73**, 2, pp. 393–398 (1980).
- [7] P. Pulay, “Improved SCF convergence acceleration,” *Journal of Computational Chemistry*, **3**, 4, pp. 556–560 (1982).
- [8] T. Rohwedder and R. Schneider, “An analysis for the DIIS acceleration method used in quantum chemistry calculations,” *Journal of Mathematical Chemistry*, **49**, 9, pp. 1889–1914 (2011).
- [9] M. T. Calef, E. D. Fichtl, J. S. Warsa, M. Berndt, and N. N. Carlson, “Nonlinear Krylov acceleration applied to a discrete ordinates formulation of the k-eigenvalue problem,” *Journal of Computational Physics*, **238**, pp. 188–209 (2013).
- [10] “SCALE:A Comprehensive Modeling and Simulation Suite for Nuclear Safety Analysis and Design,” ORNL/TM-2005/39, Version 6.1, Oak Ridge National Laboratory, Oak Ridge, TN (2011).
- [11] T. Bergman, A. Lavine, F. Incropera, and D. Dewitt, *Fundamentals of Heat and Mass Transfer*, Wiley, 7th edition, (2011).
- [12] M. El-Wakil, *Nuclear Heat Transport*, American Nuclear Society, La Grange Park, IL (1993).
- [13] D. G. Anderson, “Iterative Procedures for Nonlinear Integral Equations,” *Journal of the ACM*, **12**, 4, pp. 547–560 (1965).
- [14] H.-R. Fang and Y. Saad, “Two classes of multiseccant methods for nonlinear acceleration,” *Numerical Linear Algebra with Applications*, **16**, 3, pp. 197–221 (2009).
- [15] H. F. Walker and P. Ni, “Anderson Acceleration for Fixed-Point Iterations,” *SIAM Journal on Numerical Analysis*, **49**, 4, pp. 1715–1735 (2011).
- [16] A. Toth and C. T. Kelley, “Convergence analysis for Anderson acceleration,” *SIAM Journal on Numerical Analysis*, to appear (2013).
- [17] M. Heroux et al., “An Overview of Trilinos,” Tech. Rep. SAND2003-2927, Sandia National Laboratories (2003).