A Hybrid Approach to the Neutron Transport k-Eigenvalue Problem using NDA-based Algorithms

Jeffrey A. Willert, C. T. Kelley
North Carolina State University
Raleigh, North Carolina, 27606
jawiller@ncsu.edu, tim_kelley@ncsu.edu

D. A. Knoll, H. Park
Theory Division
Los Alamos National Laboratory
Los Alamos, New Mexico 87545
nol@lanl.gov, hkpark@lanl.gov

ABSTRACT

In order to provide more physically accurate solutions to the neutron transport equation it has become increasingly popular to use Monte Carlo simulation to model nuclear reactor dynamics. These Monte Carlo methods can be extremely expensive, so we turn to a class of methods known as hybrid methods, which combine known deterministic and stochastic techniques to solve the transport equation. In our work, we show that we can simulate the action of a transport sweep using a Monte Carlo simulation in order to solve the k-eigenvalue problem. We'll accelerate the solution using nonlinear diffusion acceleration (NDA) as in [1,2]. Our work extends the results in [1] to use Monte Carlo simulation as the high-order solver.

Key Words: Eigenvalue Calculation, Nonlinear Acceleration, Hybrid Methods

1. INTRODUCTION

We are interested in computing the dominant eigenvalue/eigenvector pair for the neutron transport criticality problem. In general, we write the eigenvalue problem as

$$\mathcal{A}x = \lambda x \tag{1}$$

in which $\mathcal{A}$ is a linear operator whose spectrum is the set of eigenvalues $\{\lambda_i\}_{i=1}^N$ with corresponding eigenvectors $\{x_i\}_{i=1}^N$. We further assume that these eigenvalues have been ordered such that $|\lambda_1| > |\lambda_2| > \cdots |\lambda_N|$ so that (λ_1, x_1) is the dominant eigenpair.

Traditional deterministic methods are modeled around the power method (or power iteration) in which successive applications of $\mathcal{A}$ to some initial vector x_0 converge to the dominant eigenvector. It can easily be shown [1,3,6] that the rate of convergence of the power method depends on the dominance ratio $\rho = |\lambda_2|/|\lambda_1|$. For many reactor problems, ρ can be very close to one and convergence may become prohibitively slow.

In recent years, many algorithmic advances have succeeded in accelerating the convergence of the dominant eigenvalue [1,3,7–9]. These methods for accelerating convergence include moment-based acceleration and rewriting the k-eigenvalue problem as a system of nonlinear equations for which we can apply known mathematical techniques for solving. These moment-based algorithms compute angular moments of the transport equation to reveal a low-order equation which exists in lower dimension. We'll focus primarily on these moment-based acceleration as in [1,2], extending the work of [5] to the k-eigenvalue problem. Other work has been done in [10–12] on the topic of hybrid methods for the k-eigenvalue problem. While within their approach, the Monte Carlo method is used to simulate scattering with a fixed fission source coming from a low-order deterministic method, the Monte Carlo method we employ solves a purely absorbing problem with both the fission source and scattering source coming from a low-order deterministic method.

2. NONLINEAR DIFFUSION ACCELERATION

2.1 The Transport Criticality Problem

As in [1], we'll consider solving the multigroup formulation of the neutron transport eigenvalue problem with isotropic scattering,

$$\hat{\Omega}\cdot\nabla\psi_g(\hat{\Omega},\vec{r})+\Sigma_{t,g}\psi_g(\hat{\Omega},\vec{r})=\frac{1}{4\pi}\left[\sum_{g'=1}^{G}\Sigma_s^{g'\to g}\phi_{g'}(\vec{r})+\frac{\chi_g}{k_{eff}}\sum_{g'=1}^{G}\nu\Sigma_{f,g'}\phi_{g'}(\vec{r})\right], \tag{2}$$

in which ψ_g is the group angular flux and $\phi_g=\int_{4\pi}\psi_g d\Omega$ is the group scalar flux for groups $g=1,\cdots,G$. $\Sigma_{t,g}$, $\Sigma_s^{g'\to g}$, and $\Sigma_{f,g}$ are the total, inscattering and fission cross-sections for group g. χ_g denotes the fission spectrum, ν is the mean number of neutrons emitted per fission event and k_{eff} is the dominant eigenvalue.

We represent Equation 2 in operator notation,

$$\mathcal{L}\Psi=\mathcal{M}\left[\mathcal{S}+\frac{1}{k_{eff}}\mathcal{F}\right]\Phi. \tag{3}$$

in which

$$\begin{aligned}\mathcal{L} &= \hat{\Omega}\cdot\nabla+\Sigma_t\\ \mathcal{M} &= \frac{1}{4\pi}\\ \mathcal{S} &= \Sigma_s\\ \mathcal{F} &= \frac{\chi}{k_{eff}}\nu\Sigma_f.\end{aligned}$$

Here, Ψ and Φ are vectors comprised of the individual group angular and scalar fluxes, respectively:

$$\begin{aligned}\Psi &= [\psi_1,\psi_2,\cdots,\psi_G]\\ \Phi &= [\phi_1,\phi_2,\cdots,\phi_G].\end{aligned}$$

2.2 Acceleration via NDA

As in [1,2,5], we take the zeroth angular moment of Equation 2 and arrive at the neutron balance equation,

$$\nabla \cdot \vec{J}_g(\vec{r}) + (\Sigma_{t,g} - \Sigma_s^{g\rightarrow g})\phi_g(\vec{r}) = \sum_{g' \neq g} \Sigma_s^{g'\rightarrow g}\phi_{g'}(\vec{r}) + \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} \nu\Sigma_{f,g'}\phi_{g'}(\vec{r}) \tag{4}$$

in which the current, $\vec{J}$, is defined by

$$\vec{J}_g = \int_{4\pi} d\Omega\, \hat{\Omega}\psi_g(\hat{\Omega}, \vec{r}). \tag{5}$$

We use the following closure relationship for the current [1,2,5]:

$$\vec{J}_g = -\frac{1}{3\Sigma_{t,g}}\nabla\phi_g + \hat{D}_g\phi_g, \tag{6}$$

which is an approximation to the first angular moment.

Next we define a low order (LO) diffusion-like equation by substituting Equation 8 into Equation 4 in place of the current:

$$\nabla \cdot \left[-\frac{1}{3\Sigma_{t,g}}\nabla\phi_g + \hat{D}_g\phi_g\right] + (\Sigma_{t,g} - \Sigma_s^{g\rightarrow g})\phi_g = \sum_{g' \neq g} \Sigma_s^{g'\rightarrow g}\phi_{g'} + \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} \nu\Sigma_{f,g'}\phi_{g'}. \tag{7}$$

We compute $\hat{D}_g$ via a discretization of Equation 8 using high-order (HO) quantities ϕ_g^{HO} and $\vec{J}_g^{HO}$:

$$\hat{D}_g = \frac{\vec{J}_g^{HO} + \frac{1}{3\Sigma_{t,g}}\nabla\phi_g^{HO}}{\phi_g^{HO}}. \tag{8}$$

High-order quantities are those which are computed directly via integration of the high-order angular flux, Ψ, computed by the inversion of $\mathcal{L}$, which is known as a "transport sweep," in Equation 3 for some approximation to the scalar flux. That is,

$$\begin{aligned}
\Psi^{HO} &= \mathcal{L}^{-1}\mathcal{M}\left[\mathcal{S} + \frac{1}{k_{eff}}\mathcal{F}\right]\Phi, \\
\Phi^{HO} &= \int \Psi^{HO} d\hat{\Omega}, \\
\vec{J}^{HO} &= \int \hat{\Omega}\Psi^{HO} d\hat{\Omega}.
\end{aligned}$$

We will express Equation 7 in operator notation,

$$\mathcal{D}\Phi = (\mathcal{S}_U + \mathcal{S}_L)\Phi + \frac{1}{k_{eff}}\mathcal{F}\Phi, \tag{9}$$

in which $\mathcal{D}$ is the drift-diffusion operator and $\mathcal{S}_U$ and $\mathcal{S}_L$ are the upper and lower scattering operators, respectively. These are given for each group by

$$\begin{aligned} \mathcal{D}_g\Phi &= \nabla\cdot\left[-\frac{1}{3\Sigma_{t,g}}\nabla+\hat{D}_g\right]\phi_g+\left(\Sigma_{t,g}-\Sigma_s^{g\to g}\right)\phi_g \\ \left(\mathcal{S}_{U,g}+\mathcal{S}_{L,g}\right)\Phi &= \sum_{g'\neq g}\Sigma_s^{g'\to g}\phi_{g'}. \end{aligned}$$

2.3 NDA Algorithms

Applying NDA to the transport criticality problem yields a set of algorithms in which we alternate between executing a single HO transport sweep in order to compute group consistency terms ($\hat{D}_g$) and solving LO k-eigenvalue problems. We'll describe these methods in Algorithm 2.3.

Algorithm 2.3: NDA Method

Compute initial iterate $\Phi^{(0)}$ and approximate eigenvalue k^0 using $\hat{D}=0$ and a fixed number of power iterations, set outer iteration counter $m=0$

while eigenvalue not converged **do**

Update counter, $m=m+1$.

Execute transport sweep and compute consistency term:

$$\begin{aligned} \Psi^{(m)} &= \mathcal{L}^{-1}\mathcal{M}\left(S+\frac{1}{k^{(m-1)}}\mathcal{F}\right)\Phi^{(m-1)} \\ \Phi^{HO} &= \int\Psi^{(m)}d\hat{\Omega},\quad \vec{J}^{HO}=\int\hat{\Omega}\Psi^{(m)}d\hat{\Omega} \\ \hat{D}^{(m)} &= \frac{\vec{J}^{HO}+\frac{1}{3\Sigma_t}\nabla\Phi^{HO}}{\Phi^{HO}}. \end{aligned}$$

Solve the LO eigenvalue problem given by

$$\left(\mathcal{D}-\mathcal{S}_U-\mathcal{S}_L\right)\Phi^{(m)}=\frac{1}{k_{eff}^{(m)}}\mathcal{F}\Phi^{(m)}$$

for the next approximation to the eigenvalue/eigenvector pair $(k_{eff}^{(m)},\Phi^{(m)})$.

end while

In this paper, we will not concern ourselves with the method in which the LO eigenvalue problem is solved. However, it is key to note that the eigenvalue iteration is only executed at the level of the low order problem. There are no transport sweeps involved in the inner eigenvalue iteration. In [1], the authors discuss two methods for solving the LO eigenvalue problem. The first of these methods is a basic shifted power iteration and the second method relies on writing the LO eigenvalue problem as a nonlinear system of equations and solving using Newton's method. Both of these methods can converge in very few iterations. The authors of [1] demonstrate that posing the problem as a nonlinear system of equations may be advantageous.

3 HYBRID METHODS FOR NEUTRON TRANSPORT

In Algorithm 2.3 we must invert $\mathcal{L}$ at each iteration in order to update the group consistency term, $\hat{D}_g$, for each group. Previous work using NDA-based acceleration of the k-eigenvalue solution has only considered computing the transport sweep using deterministic methods. Instead, let us consider building these consistency terms using a Monte Carlo transport sweep as in [5].

This transport sweep will be executed as described in Algorithm 3. First, we'll use the current approximation to the scalar flux to build a scattering- and fission-free fixed-source problem. Then, we use a Monte Carlo simulation to compute a HO scalar flux and current. Using these quantities, we'll be able to compute the group consistency terms.

Algorithm 3: Monte Carlo Transport Sweep

Input current approximation to the scalar flux $\Phi^{(m-1)}$

Build a scattering and fission-free fixed source probelm:

$$\mathcal{L}\Psi = \mathcal{Q} \tag{10}$$

where

$$\mathcal{Q} = \mathcal{M}\left(\mathcal{S} + \frac{1}{k_{eff}^{(m-1)}}\mathcal{F}\right)\Phi^{(m-1)}$$

Use Monte Carlo simulation to solve Equation10 and tally Φ^{HO}, $\vec{J}^{HO}$.

We should make sure to note the low-cost nature of this Monte Carlo transport sweep. In general Monte Carlo simulations, we must sample probability distributions at each neutron interaction site to determine whether the interaction resulted in scattering, fission or capture. In these computations, we must keep track of fission sites for future generations of neutrons and continue tracking scattered particles. In our simplified simulation, we need only note the location of the interaction and tally the scalar flux and current along the single flight path from the particles birth location to the interaction site.

4 1-D, 1-GROUP TEST PROBLEMS AND RESULTS

4.1 Implementation Details

As in [5], we understand that at the initial phase of the iteration, it is not necessary to compute the scalar flux and current to extremely high accuracy. For this reason, at the beginning of the iteration, we start with a relatively small number of particles, 10^6 particle flights per transport sweep. At each successive outer iteration we increase the number of particles per simulation by a factor of 2. In practice, this allows us to gain increasingly accurate approximations to the eigenvalue and eigenvector without wasting effort in early sweeps. As we approach the eigenvalue/eigenvector pair, the necessity to compute these values accurately increases.

The factor used to increase the number of particles should be chosen with a couple of considerations in mind. We need to use enough particles at the initial stage of the algorithm so that our approximations to the scalar flux and current are meaningful. Furthermore, by the end of the iteration process, we need only gain enough accuracy to compute the eigenvalue to the desired tolerance. If we expect it will require on order 10^{10} particles to gain the accuracy we desire, we should chose a factor so that 10^{10} particles per simulation is achieved in roughly the same number of iterations that it will take to converge the eigenvalue to these tolerances. In other words, we don't want to execute simulations with far more particles than are needed to achieve the desired level of accuracy.

Furthermore, we chose to use the NDA-NCA [1] algorithm in which the LO eigenvalue problem is solved using Newton's method. In this case, the consistency term $\hat{D}$ is held constant throughout the LO eigenvalue solve, so the issues which arose in [5] can safely be ignored. For this paper, the choice between these two algorithms is arbitrary, as the goal of our work focuses strictly upon the computation of the consistency term within the outer iteration.

4.2 Numerical Results

We will consider two 1-D, 1-Group test problems to demonstrate the effectiveness of this algorithm. Both Test 1 and Test 2 below consist of a single fissile region surrounded by two 5 mean-free-path reflectors [12]. The length of the medium is given by τ, and the length of the fissile region is given by $\tau - 10$.

Consider the parameters for the first test:

$$\begin{aligned} &\Sigma_t = 1 \quad \Sigma_s = .856 \quad \nu\Sigma_f = .144 \quad \tau = 35 \text{ (fissile region)} \\ &\Sigma_t = 1 \quad \Sigma_s = .856 \quad \nu\Sigma_f = 0 \quad\ \ \tau = 35 \text{ (reflector region)} \end{aligned}$$

For this problem, we found the true eigenvalue to be $k_{eff} = .9720469427$, computed using a high accuracy S_n computation. We'll run the simulation 10 times in order to gauge performance. For each simulation we'll discretize using 7000 spatial cells and a total of roughly 3.28×10^{10} particle flights (15 outer iterations). After 10 independent simulations we collect the following results for k_{eff}:

- Mean = **.9720469**297
- Standard deviation = .0000028874
- Max eigenvalue = **.9720**577602 (.001866% error)
- Min eigenvalue = **.97204**20155 (.001850% error)

Consider the parameters for the second test:

$$\begin{aligned} &\Sigma_t = 1 \quad \Sigma_s = .856 \quad \nu\Sigma_f = .144 \quad \tau = 60 \text{ (fissile region)} \\ &\Sigma_t = 1 \quad \Sigma_s = .856 \quad \nu\Sigma_f = 0 \quad\ \ \tau = 60 \text{ (reflector region)} \end{aligned}$$

For this problem, the true eigenvalue is $k_{eff} = .9919893631$. We'll run the simulation 10 times in order to gauge performance. For each simulation we'll discretize using 12000 spatial cells and a total of roughly 3.28×10^{10} particle flights (15 outer iterations). After 10 independent simulations we collect the following results for k_{eff}:

- Mean = **.991989**2935
- Standard deviation = .0000011455
- Max eigenvalue = **.9919**907012 (.000142% error)
- Min eigenvalue = **.99198**75238 (.000178% error)

5. CONCLUSIONS

We have shown that we can compute the dominant eigenvalue for transport criticality problems by replacing the transport sweep in NDA-based algorithms with a Monte Carlo simulation. These Monte Carlo simulations are less computational effort than traditional Monte Carlo simulations. By moving the scattering and fission operators to the LO system, these simulations need only simulate particle flights from particle birth location to the site of their first interaction.

ACKNOWLEDGEMENTS

The work of these authors has been partially 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, National Science Foundation Grant CDI-0941253 and Army Research Office Grants W911NF-11-1-0367, and W911NF-07-1-0112.

REFERENCES

[1] H. Park, D. A. Knoll, and C. K. Newman, "Nonlinear Acceleration of Transport Criticality Problems," *Nuclear Science and Engineering*, **171**, pp. 1-14 (2012).

[2] D. Knoll, H. Park and K. Smith, "Application of the Jacobian-Free Newton-Krylov Method to Nonlinear Accleration of Transport Source Iteration in Slab Geometry," *Nuclear Science and Engineering*, **167**, pp. 122 (2011).

[3] D. A. Knoll, H. Park, and C. Newman, "Acceleration of k-Eigenvalue/Criticality Calculations Using the Jacobian-Free Newton-Krylov Method," *Nuclear Science and Engineering*, **167**, pp. 133-140 (2011).

[4] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philidelphia, (1995).

[5] Jeffrey Willert, C. T. Kelley, D. A. Knoll, and H. Park, "Hybrid Deterministic/Monte Carlo Neutronics", Submitted for publication.

[6] Marvin L. Adams and Edward W. Larsen, "Fast Iterative Methods for Discrete-Ordinates Particle Transport Calculations," *Progress in Nuclear Energy*, **40**, pp. 3 - 159 (2002).

[7] Daniel F. Gill and Yousry Y. Azmy, "Newton's Method for Solving k-Eigenvalue Problems in Neutron Diffusion Theory," *Nuclear Science and Engineering*, **167**, pp. 141-153 (2011).

[8] Daniel F. Gill and Yousry Y. Azmy, "Newton's Method for the Computation of k-Eigenvalues in S_N Transport Applications," *Nuclear Science and Engineering*, **167**, pp. 141-153 (2011).

[9] Matthew T. Calef, Erin D. Fichtl, James S. Warsa, Markus Berndt, and Neil N. Carlson, "A Nonlinear Krylov Accelerator for the Boltzmann k-Eigenvalue Problem," *Submitted to the Journal of Computational Physics for publication*.

[10] Min-Jae Lee, Han Gyu Joo, Deokung Lee and Kord Smith, "Investigation of CMFD Accelerated Monte Carlo Eigenvalue Calculation with Simplified Low Dimensional Multigroup Formulation," *PHSYOR 2010 - Advances in Reactor Physics to Power the Nuclear Renaissance*

[11] Jinan Yang and Edward W. Larsen, "Calculations of k-Eigenvalues and Multi-Group Eigenfunctions Using the Hybrid "Functional Monte Carlo" Method," *PHSYOR 2010 - Advances in Reactor Physics to Power the Nuclear Renaissance*

[12] Edward W. Larsen and Jinan Yang, "A Functional Monte Carlo Method for k-Eigenvalue Problems," *Nuclear Science and Engineering*, **159**, pp. 107-126 (2008).