

A JACOBIAN-FREE NEWTON-KRYLOV ITERATIVE SCHEME FOR CRITICALITY CALCULATIONS BASED ON THE NEUTRON DIFFUSION EQUATION

D. F. Gill

The Pennsylvania State University
138 Reber Building, University Park, PA 16802
dfg121@psu.edu

Y. Y. Azmy

Department of Nuclear Engineering
North Carolina State University
Campus Box 7909, 1110 Burlington Laboratories, Raleigh, NC, 27695
yyazmy@ncsu.edu

ABSTRACT

Newton-Krylov methods, primarily using the Jacobian-Free Newton-Krylov (JFNK) approximation, are examined as an alternative to the traditional power iteration method for the calculation of the fundamental eigenmode in reactor analysis applications based on diffusion theory. One JFNK approach can be considered an acceleration technique for the standard power iteration as it is “wrapped around” the power method, allowing for simplified implementation in preexisting codes. Since the Jacobian is not formed the only extra storage required is associated with the workspace of the Krylov solver used at every Newton step. Another Newton-based method is developed which solves the generalized eigenvalue problem formulation of the k-eigenvalue problem, both using the JFNK approximation and utilizing the action of the true Jacobian matrix. These Newton-based methods were compared to the unaccelerated and Chebyshev-accelerated power method for a number of reactor models. Results show calculation of the fundamental mode using JFNK acceleration of the power method generally results in fewer iterations and shorter run times than when using the unaccelerated and Chebyshev-accelerated power methods. The Newton-based approaches implemented for the solution of the generalized eigenvalue problem formulation of the k-eigenvalue problem are shown to exhibit poor convergence properties for the iterations associated with the linearized Newton step, highlighting the need for an effective preconditioner.

Key Words: Newton-Krylov, JFNK, Power method, Eigenvalue diffusion problem

1. INTRODUCTION

Calculation of the fundamental eigenmode in criticality problems has traditionally utilized the classical power iteration method, which though robust, converges slowly for dominance ratios [1] near one. In practical situations the dominance ratio is often near unity, resulting in slow convergence and a need for acceleration techniques to improve the convergence of the power iterations. Two common approaches in diffusion theory are Chebyshev iteration [2] and Wielandt shift [3]. Alternative approaches to power iteration have been researched in an attempt to improve upon the performance of accelerated power iteration methods. Subspace iterations [4] and Krylov subspace methods, such as the Implicit Restarted Arnoldi Method (IRAM) have been successfully

applied to transport [5] and diffusion criticality problems [6]. These methods can also be used to find multiple eigenmodes, not just the fundamental mode. Very recently the use of JFNK methods has also been investigated in conjunction with IRAM as a tool for use in BWR modal analysis [7]. In the current work, similar JFNK methods are explored as an alternative to the power iteration for seeking the fundamental mode. One such method focuses on creating an algorithm which relies on the kernel calculation of the classical power iteration at its core. By building the method around the existing power iteration it can be implemented within the framework of existing codes without requiring substantial alteration. Another method considers posing the k-eigenvalue problem as a generalized eigenvalue problem, resulting in a method that does not require inversion of the diffusion operator.

This paper will briefly present the basics of Newton-Krylov methods and the JFNK approximation, the application of those methods to fixed-point iterations and the generalized eigenvalue problem and then apply the ideas developed to the multigroup diffusion problem. Details regarding the implementation of the Newton-based methods for the diffusion eigenvalue problem will then be presented and finally numerical results for a number of reactor models will be presented and discussed, to demonstrate the computational efficiency and potential drawbacks of the new approaches.

2. JACOBIAN-FREE NEWTON-KRYLOV METHODS

JFNK methods are inexact Newton methods designed to solve nonlinear systems of equations and often used as a means to couple together stand-alone codes modeling multiple physical phenomena; an excellent survey of the technique and its applications has been published by Knoll and Keyes [8]. The term Jacobian-Free is used because the Jacobian of the nonlinear system is neither formed nor stored, rather only accessed through Jacobian vector products. For the nonlinear system $\Gamma(u) = 0$, the traditional Newton iteration is obtained via a truncated Taylor expansion about the m^{th} iterate, u^m , resulting in the iterative sequence

$$J(u^m)\delta u^m = -\Gamma(u^m), \quad u^{m+1} = u^m + \delta u^m, \quad m = 0, 1, \dots \quad (1)$$

where $J(u^m)$ is the Jacobian at u^m , $\Gamma'(u^n)$, and it is assumed the initial guess, u^0 , is given. Thus each step in the sequence requires the solution of the linear system defined above and continues until suitable convergence is achieved. Rather than directly invert $J(u^m)$ an inexact Newton method iteratively solves the linearized system and when an iterative Krylov solver is used this is called a Newton-Krylov method. In the case of JFNK, since the iterative solver is a Krylov subspace method it only requires the action of the Jacobian on some vector, not explicit construction of the matrix. This product can be approximated by

$$Jv \approx \frac{\Gamma(u + \epsilon v) - \Gamma(u)}{\epsilon} \quad (2)$$

where ϵ is a small finite-difference perturbation parameter. Using the JFNK method there is no need for the formation of the Jacobian, one only needs to be able to evaluate the nonlinear function, Γ , at an arbitrary state. The JFNK method is therefore summarized by solving Eq. (1), where

$$\|\Gamma(u^m) + J(u^m)\delta u^m\| \leq \gamma^m \|\Gamma(u^m)\| \quad (3)$$

which basically requires a reduction in some residual norm by a factor of γ^m before the linear system is considered converged for some Newton step m . All Jacobian-vector products required by the Krylov method used to iteratively solve the linear Newton step are approximated using Eq. (2). Before describing the application of the JFNK methods to the k-eigenvalue problem in neutron diffusion theory the traditional power iteration approach will be presented.

2.1. Neutron Diffusion Equations

Using the nomenclature of Duderstadt and Hamilton [9] the multigroup diffusion equations can be given by

$$(-\nabla \cdot D_g \nabla + \Sigma_{R_g}) \phi_g - \sum_{g' \neq g} \Sigma_{sg'g} \phi_{g'} = \lambda \chi_g \sum_{g'=1}^G \nu_{g'} \Sigma_{fg'} \phi_{g'}, \quad g = 1, \dots, G \quad (4)$$

Upon spatial discretization Eq. (4) can be written as

$$M\phi = \lambda F\phi \quad (5)$$

where M_g denotes the discrete form of the operator $-\nabla \cdot D_g \nabla + \Sigma_{R_g}$ and λ is the reciprocal of the multiplication factor, k_{eff} . Here M and F are block matrices given by

$$M = \begin{bmatrix} M_1 & -\Sigma_{s_{21}} & -\Sigma_{s_{31}} & \dots & -\Sigma_{s_{G1}} \\ -\Sigma_{s_{12}} & M_2 & -\Sigma_{s_{32}} & \dots & -\Sigma_{s_{G2}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\Sigma_{s_{1G}} & -\Sigma_{s_{2G}} & -\Sigma_{s_{3G}} & \dots & M_G \end{bmatrix} \quad (6)$$

$$F = \begin{bmatrix} \chi_1 \nu_1 \Sigma_{f_1} & \chi_1 \nu_2 \Sigma_{f_2} & \dots & \chi_1 \nu_G \Sigma_{f_G} \\ \chi_2 \nu_1 \Sigma_{f_1} & \chi_2 \nu_2 \Sigma_{f_2} & \dots & \chi_2 \nu_G \Sigma_{f_G} \\ \vdots & \vdots & \ddots & \vdots \\ \chi_G \nu_1 \Sigma_{f_1} & \chi_G \nu_2 \Sigma_{f_2} & \dots & \chi_G \nu_G \Sigma_{f_G} \end{bmatrix} \quad (7)$$

Both M and F have dimension $N \times N$ where $N = n \times G$ with n being the total number of spatial cells and G the number of energy groups. The block components of these matrices are of size $n \times n$, with the matrices $\Sigma_{s_{gg}}$ and $\chi_{g'} \nu_{g'} \Sigma_{f_g}$ both being simple diagonal matrices that map cross sections to spatial cells. The diffusion operator matrices, M_g , are dependent upon the spatial discretization, but for the finite-difference technique are symmetric banded matrices with three, five or seven diagonals for one-, two-, or three-dimensional problems, respectively. The scalar neutron flux is given by

$$\phi = [\phi_1^T \quad \phi_2^T \quad \dots \quad \phi_G^T]^T \quad (8)$$

so that ϕ is a vector of length N , composed of vectors ϕ_g , $g = 1, \dots, G$, each of length n . The quantity f is a vector of length n which represents the spatial distribution of the fission source and is defined by

$$f(\phi) \equiv \sum_{g'=1}^G \nu_{g'} \Sigma_{fg'} \phi_{g'} \quad (9)$$

Since the shape of the fission source is the same for all groups, the fission source for group g is simply given by, $f_g = \chi_g f$, where χ_g is just the scalar value of the fission spectrum for that group.

Writing Eq. (5) in the traditional power iteration formulation of the eigenvalue problem yields

$$\phi^{(l+1)} = \lambda^{(l)} (M^{-1}F) \phi^{(l)} \quad (10)$$

where $\lambda^{(l+1)}$ is generally given by some weighted update formula $\bar{\lambda}(\phi^{(l+1)}, \phi^{(l)}, \lambda^{(l)})$. The iterative process of updating $\phi^{(l)}$ and $\lambda^{(l)}$ is known as the outer iteration (or power iteration). Evaluating $M^{-1}F$ is usually done on a group-wise basis which requires the solution of a linear system of equations for each group. The within-group problem when upscattering is present is given by

$$\begin{aligned} M_g \phi_g^{(l+1)} &= \sum_{g'=1}^{g-1} \Sigma_{s_{g'g}} \phi_{g'}^{(l+1)} + \sum_{g'=g+1}^G \Sigma_{s_{g'g}} \phi_{g'}^{(l)} + \lambda^{(l)} f_g^{(l)} \\ &= \left(s_g^{(l+1)\text{down}} + s_g^{(l)\text{up}} + \lambda^{(l)} f_g^{(l)} \right) \end{aligned} \quad (11)$$

Alternatively the updated flux and scattering sources could be simultaneously computed through a nested iterative process over groups with upscattering. The iterations associated with solving the within-group problem are referred to as inner iterations. When there is no upscattering, M , becomes a lower triangular matrix and the system can easily be solved by considering each within-group equation in succession as described by Duderstadt and Hamilton [9]. The presence of upscattering can be treated by either using the scalar fluxes from the previous iterate when calculating the upscattering source (in which case the index k can be ignored) or by simultaneously solving for the group fluxes in groups with non-negligible upscattering as indicated in Eq. (11), which involves another level of iteration over these energy groups. This iteration over energy groups would be completely contained within a single outer iteration.

2.2. JFNK as Acceleration of Power Iteration

To generalize the use of JFNK methods for coupling physical subsystems Xu and Downar [10] described an approach to apply the JFNK method to nested iteration schemes. These nested schemes, referred to as fixed point iterations, are defined by the function Φ , with the solution given by u^* , such that $u^* = \Phi(u^*)$. Using the fixed point iteration function one can define an associated nonlinear system

$$\Gamma_{\text{acc}}(u) = u - \Phi(u) \quad (12)$$

which can then be directly substituted for Γ in the JFNK algorithm defined at the beginning of Sec. 2. Brown and Saad [11] point out this approach can be considered an acceleration of the fixed point iteration or a nonlinearly preconditioned form of the original system. If the power iteration, given by Eq. (10), is expressed as a fixed point iteration such that $u^{(l+1)} = \Phi(u^{(l)})$ then Φ and u are given by

$$\Phi(u) = \begin{bmatrix} \lambda M^{-1} F \phi \\ \bar{\lambda}(u) \end{bmatrix}, \quad u = \begin{bmatrix} \phi \\ \lambda \end{bmatrix} \quad (13)$$

where $\bar{\lambda}(u)$ represents the update formula for λ . Generally, the power iteration, denoted by the function Φ , is performed as described in the previous section, however as seen by Eq. (12) only the difference between successive power iterations is necessary to evaluate Γ_{acc} . This means that the kind of implementation specific details about how to normalize the flux, treat upscattering, or update λ do not need to be treated explicitly outside of Φ . Since Φ represents a single outer

iteration any code which is capable of performing power iterations can be modified to use the JFNK method as a means of acceleration.

This results in a system very similar to that proposed by Mahadevan and Ragusa [7]. Symbolically the only difference arises from the final equation in the nonlinear system. The last equation in their system determines how close the current fluxes are to satisfying an imposed normalization condition, a result of interpreting the eigenvalue problem as an optimization problem. In Eq. (13) the last equation simply gives the difference between successive eigenvalue estimates calculated via power iteration. Though both systems can be written identically in operator notation, aside from the last equation, Eq. (13) is intended to represent a single outer (power) iteration as traditionally used to solve the multigroup diffusion equations.

2.3. Using JFNK to Solve the Generalized Eigenvalue Problem

Rather than solving the eigenvalue problem treated by power iterations, Eq. (10), that requires inversion (iterative) of M suppose one considers applying JFNK to the generalized eigenvalue problem given by Eq. (5). This problem can be posed as a nonlinear function by

$$\Gamma_{\text{gep}}(u) = \begin{bmatrix} M\phi - \lambda F\phi \\ -\frac{1}{2}\phi^T\phi + \frac{1}{2} \end{bmatrix}, \quad u = \begin{bmatrix} \phi \\ \lambda \end{bmatrix} \quad (14)$$

where the final equation is now used to impose a normalization condition on the scalar flux vector. Any normalization condition could be used in place of the one used here, but Stewart [12] points out that choosing this specific normalization condition results in a Newton correction that is optimal in the sense that it is orthogonal to the previous correction. Using the JFNK algorithm described in Sec. 2 with the nonlinear function Γ_{gep} results in a Newton-based approach which avoids the solution of the within-group problem defined by Eq. (11).

The full matrices M and F are generally not available so an algorithm designed to evaluate Γ_{gep} is now examined. Consider separating the vector-valued function $\Gamma_{\text{gep}}(u)$, of length $N + 1$, into G sub-vectors each of length n , $\Gamma_g(u)$ for $g = 1, \dots, G$, and a scalar component, $\Gamma_\lambda(u)$, such that

$$\Gamma_{\text{gep}} = [\Gamma_1^T \quad \Gamma_2^T \quad \dots \quad \Gamma_G^T \quad \Gamma_\lambda]^T \quad (15)$$

In block form the first G blocks (N elements) of $\Gamma_{\text{gep}}(u)$ are given by

$$\begin{bmatrix} \Gamma_1 \\ \Gamma_2 \\ \vdots \\ \Gamma_G \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} M_1 & -\Sigma_{s_{21}} & \dots & -\Sigma_{s_{G1}} \\ -\Sigma_{s_{12}} & M_2 & \dots & -\Sigma_{s_{G2}} \\ \vdots & \vdots & \ddots & \vdots \\ -\Sigma_{s_{1G}} & -\Sigma_{s_{2G}} & \dots & M_G \end{bmatrix} & -\lambda \begin{bmatrix} \chi_1\nu_1\Sigma_{f_1} & \dots & \chi_1\nu_G\Sigma_{f_G} \\ \chi_2\nu_1\Sigma_{f_1} & \dots & \chi_2\nu_G\Sigma_{f_G} \\ \vdots & \ddots & \vdots \\ \chi_G\nu_1\Sigma_{f_1} & \dots & \chi_G\nu_G\Sigma_{f_G} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_G \end{bmatrix} \quad (16)$$

so that the operations associated with evaluating Γ_{gep} can be broken down on a group-by-group basis. For a general group g ,

$$\Gamma_g(u) = M_g\phi_g - \sum_{g'=g+1}^G \Sigma_{s_{g'g}}\phi_{g'} - \sum_{g'=1}^{g-1} \Sigma_{s_{g'g}}\phi_{g'} - \lambda f_g \quad (17)$$

$$= M_g\phi_g - (s_g^{\text{down}} + s_g^{\text{up}} + \lambda f_g) \quad (18)$$

The final component of $\Gamma_{\text{gep}}(u)$, $-\frac{1}{2}\phi^T\phi + \frac{1}{2}$, can be written

$$\Gamma_{\lambda}(u) = -\frac{1}{2} (\phi_1^T\phi_1 + \phi_2^T\phi_2 + \dots + \phi_G^T\phi_G) + \frac{1}{2} \quad (19)$$

resulting in a method to evaluate $\Gamma_{\text{gep}}(u)$ that emulates the group-wise procedure of the outer-inner iteration scheme. The capability to construct each of the components of $\Gamma_{\text{gep}}(u)$ is presumed to exist in any diffusion code capable of performing power iterations. Each evaluation of Γ_{gep} requires less computation than an evaluation of Γ_{acc} but this is not sufficient to guarantee improved performance because the convergence properties of the two methods are unknown.

Aside from avoiding the inner iterations associated with $M^{-1}F\phi$ this method has the added benefit of treating upscattering and downscattering equally. There are no additional iterations required over energy groups which have non-negligible upscattering. Looping through energy groups in a given sequence is an artifact of the way data is stored not because of any requirement that this be so. In this case the coupling of energy groups is treated at the level of the Newton iteration.

2.4. Newton-Krylov Approach to the Generalized Eigenvalue Problem

In this variation of the previous approach, the JFNK approximation is abandoned in favor of utilizing the true Jacobian in conjunction with a Newton-Krylov method. Given the nonlinear, quadratic in u , function $\Gamma_{\text{gep}}(u)$ defined by Eq. (14), the Jacobian is given by

$$J(u) = \begin{bmatrix} M - \lambda F & -F\phi \\ -\phi^T & 0 \end{bmatrix} \quad (20)$$

However, in the context of a Newton-Krylov method it is unnecessary to explicitly calculate the Jacobian in this manner, instead the construction of the Jacobian-vector product $J(u)v$ will be investigated. Suppose v is partitioned as Γ_{gep} was in the previous section, i.e., $v = [v_1^T \ v_2^T \ \dots \ v_G^T \ v_{\lambda}]^T$, and this is shortened even further by $v = [v_{\phi}^T \ v_{\lambda}]^T$, where $v_{\phi} = [v_1^T \ \dots \ v_G^T]^T$. Using this notation the Jacobian-vector product can be written

$$Jv = \begin{bmatrix} (Jv)_1 \\ \vdots \\ (Jv)_g \\ \vdots \\ (Jv)_G \\ (Jv)_{\lambda} \end{bmatrix} = \begin{bmatrix} M_1 v_1 - \sum_{g'=g+1}^G \Sigma_{s_{g'g}} v_{g'} - \lambda f_1(v_{\phi}) - v_{\lambda} f_1(\phi) \\ \vdots \\ M_g v_g - \sum_{g'=1}^{g-1} \Sigma_{s_{g'g}} v_{g'} - \sum_{g'=g+1}^G \Sigma_{s_{g'g}} v_{g'} - \lambda f_g(v_{\phi}) - v_{\lambda} f_g(\phi) \\ \vdots \\ M_G v_G - \sum_{g'=1}^{G-1} \Sigma_{s_{g'g}} v_{g'} - \lambda f_G(v_{\phi}) - v_{\lambda} f_G(\phi) \\ - (\phi_1^T v_1 + \phi_2^T v_2 + \dots + \phi_g^T v_g + \dots + \phi_G^T v_G) \end{bmatrix} \quad (21)$$

which can again be evaluated on a group-wise basis. The vectors $(Jv)_g$ contain a matrix-vector product of M_g and v_g , the “scattering sources” which are calculated using v_{ϕ} rather than ϕ , the product of λ and the group “fission source” due to v_{ϕ} rather than ϕ , and finally the product of the actual group fission source and the last component of v , v_{λ} . Thus, evaluating $J(u)v$ is nearly

identical to the evaluation of $\Gamma_{\text{gep}}(u)$, yielding an algorithm which does not require an approximation of the action of the Jacobian.

3. NUMERICAL RESULTS

Numerical results were generated for a number of sample problems using a test code which implements the JFNK method as power method acceleration, referred to as JFNK(PI) when inner iterations are not preconditioned and JFNK(IC-PI) when inner iterations are preconditioned using Incomplete Cholesky factorization. The JFNK method developed for the generalized eigenvalue problem is implemented and denoted by JFNK(GEP) when discussing numerical results while the Newton-Krylov (true Jacobian) approach for the generalized eigenvalue problem has also been implemented and is simply referred to as Newton-Krylov. The test code is also capable of performing standard power iterations and Chebyshev accelerated power iterations. This code will be used to examine the performance of the Newton-based methods compared to that of the unaccelerated and accelerated power method and the numerical parameters involved in the Newton-based approaches will be studied to determine their influence on performance.

The test code used to implement the various eigenproblem solution algorithms for diffusion theory is written in Fortran 90/95 and is intended to solve the two-dimensional multigroup diffusion equations. The spatial discretization is a mesh-centered finite-difference approach in Cartesian geometry that supports vacuum (Marshak) and reflective boundary conditions. Sparse storage is utilized for the storage of the M_g matrices and the Conjugate Gradient algorithm was used to solve the within-group problem during the power iteration process, with the option to precondition these equations using the Incomplete Cholesky Factorization of M_g as the preconditioner; the DLAP [13] Fortran implementations of CG and preconditioned CG were used. The power iteration was implemented as described by Duderstadt and Hamilton [9], with λ updated via

$$\lambda^{(l+1)} = \lambda^{(l)} \frac{(f^{(l+1)}, f^{(l)})}{(f^{(l+1)}, f^{(l+1)})} \quad (22)$$

and the Chebyshev acceleration implemented as described by Ferguson and Derstine [14]. The L_2 norm of the new scalar flux iterate, $\phi^{(l+1)}$, was set to unity each outer iteration.

The module associated with the Newton methods was designed to interact with the diffusion code primarily through function calls. The JFNK acceleration of the power method requires only a functional call to the subroutine that performs outer iterations. GMRES was used to solve the linear Newton step in all of the Newton-based processes, again utilizing the DLAP subroutines. Each method tested was stopped when all of the following conditions were met:

$$|k^{(l+1)} - k^{(l)}| < \epsilon_k \quad (23a)$$

$$\frac{\|f^{(l+1)} - f^{(l)}\|_2}{(f^{(l+1)}, f^{(l)})^{1/2}} < \epsilon_2 \quad (23b)$$

$$\max_i \left| \frac{f_i^{(l+1)} - f_i^{(l)}}{f_i^{(l+1)}} \right| < \epsilon_\infty \quad (23c)$$

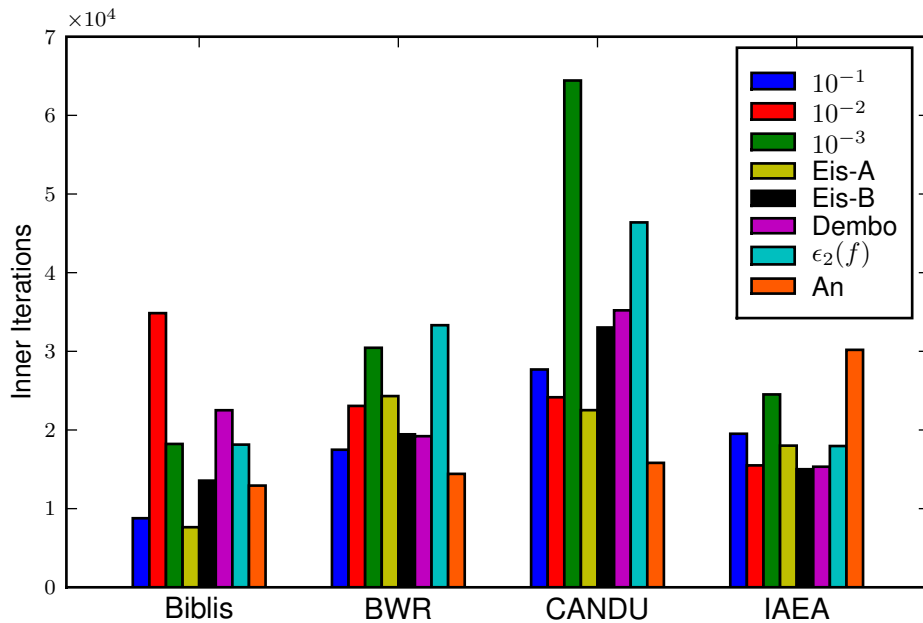


Figure 1. Impact of γ on Inner Iteration Count for JFNK(PI)

For the power method these conditions were tested between outer iterations, while for JFNK(PI) they were checked when calculating the Newton residual at each Newton iteration. For the generalized eigenvalue problem techniques these conditions were checked using successive Newton iterates. One of two convergence sets was used; these are termed regular or tight convergence, given by $\{5 \times 10^{-6}, 5 \times 10^{-5}, 5 \times 10^{-4}\}$ and $\{5 \times 10^{-14}, 5 \times 10^{-14}, 5 \times 10^{-14}\}$, respectively for the criteria $\{\epsilon_k, \epsilon_2, \epsilon_\infty\}$.

Four reactor benchmarks were used to generate input models for the eigenvalue methods implemented in the test code: The IAEA 2-D benchmark (identification 11-A2 [15] Supp. 2), the Biblis Test Problem [16], and static versions of a BWR transient (identification 14-A1 [15] Supp. 2) and a CANDU kinetics problem (identification 17-A1 [15] Supp. 3). Each case uses quarter-core symmetry, i.e. reflective boundary conditions on inner surfaces and vacuum on outer surfaces, and the reflector was extended into the void to create a square domain. Each problem was modeled using a mesh with a uniform cell size per dimension, Δ of 1 cm, 1.05 cm, 1 cm, and 2 cm for the IAEA, Biblis, BWR, and CANDU problems, respectively, unless otherwise specified. An extensive number of numerical experiments were performed to explore the effects of various computational parameters associated with the Newton-based methods. These parameters were the Jacobian perturbation ϵ as seen in Eq. (2), the inexact Newton forcing factor γ , as used in Eq. (3), the number of standard power iterations used to initialize the calculation and the control parameters for the inner iterations (maximum number of inner iterations per group and inner iteration convergence tolerance). The values of ϵ considered were all empirically based except for that of Xu [17]. Results for JFNK(GEP) and JFNK(PI) show sensitivity to ϵ but in an unpredictable manner while JFNK(IC-PI) is generally quite insensitive to the perturbation value.

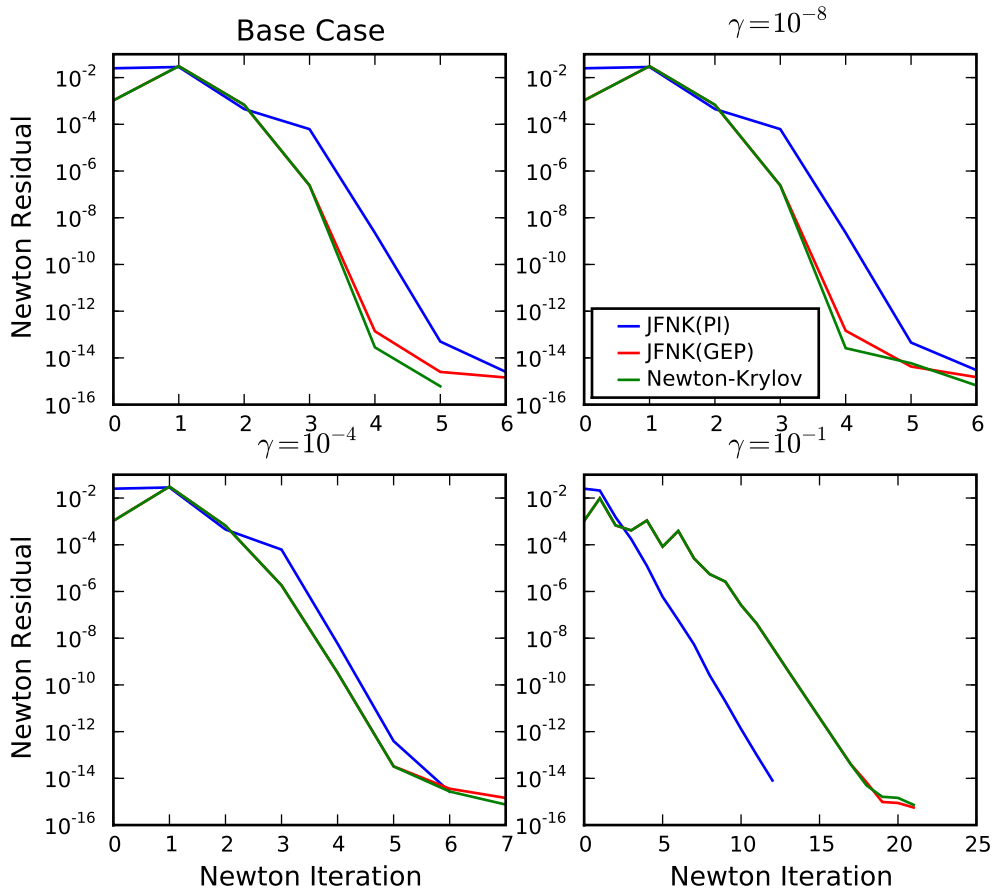


Figure 2. Convergence of Newton-Based Methods for IAEA Benchmark as a Function of γ

The choice of the forcing factor sequence, γ^m , is an important criterion and many possible values were considered. Constant γ values of 10^{-1} , 10^{-2} , and 10^{-3} were used as were two methods developed by Eisenstat and Homer [18], termed Eis-A and Eis-B, which dynamically vary with the Newton iteration based on the iterative progress. A method proposed by Dembo where $\gamma^m = \min\{\frac{1}{m+2}, \|F(u^m)\|\}$ is utilized along with a method which bases the forcing factor on the global error of the scalar flux, i.e. γ^m is set to the fission source error from the previous iteration as defined by the left side of equation Eq. (23b). The final sequence of forcing factors tested was based on the method recently developed by An [19]. Results of these numerical experiments for JFNK(PI) are given in Fig. 1.

The methods of Eisenstat and An, which are the most sophisticated, appear to result in consistently good performance as does a constant γ of 10^{-1} . The initial guess used to begin each problem was a flat flux such that $\|\phi^0\|_2 = 1$ and with $k^0 = 1$. This starting condition was either used directly in all of the methods tested or was used to perform a set number of initial power iterations ranging in number from 0 to 25. It was found that in general the Newton-methods are

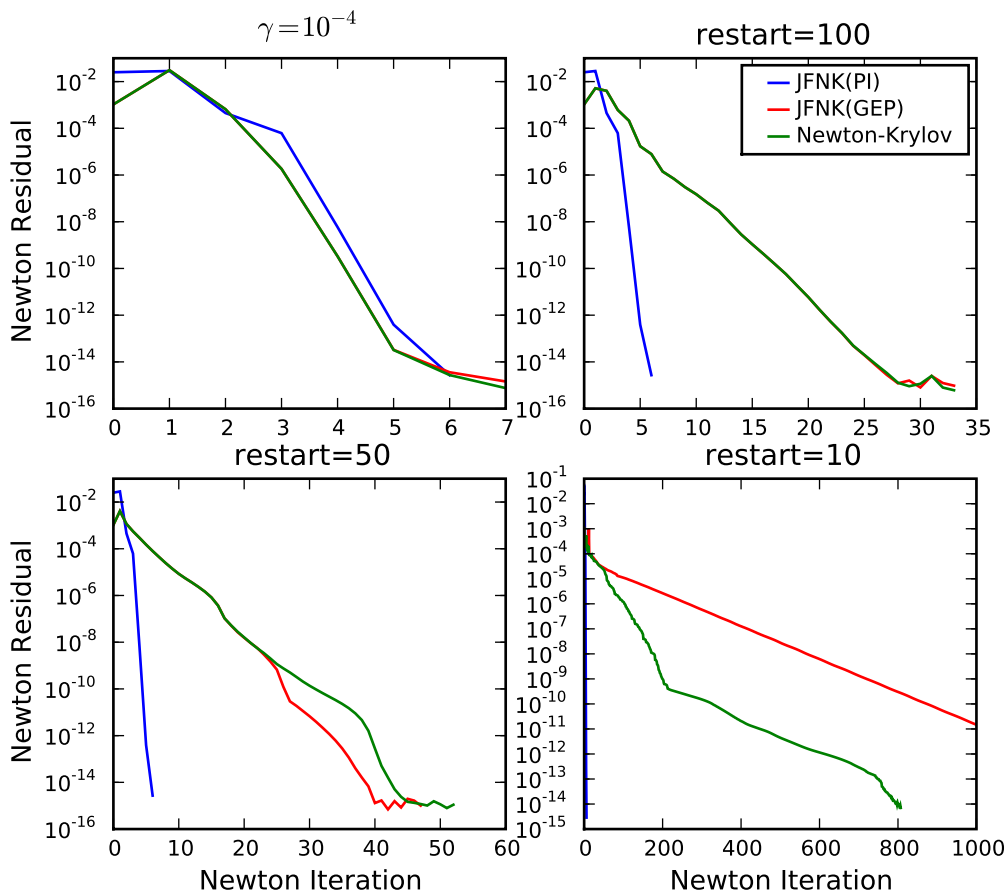


Figure 3. Convergence of Newton-Based Methods for IAEA Benchmark as a Function of GMRES Iterative Properties, 100 Iteration Maximum

quite sensitive to the choice of initial guess, although again in a rather unpredictable manner such that no best choice for this parameter was identified. Testing of the inner iteration parameters for the power-based methods (power, Chebyshev accelerated power, and JFNK(PI)) considered terminating the within-group iterations by either reaching a maximum or by attaining some desired tolerance. In general these results indicate that for the preconditioned case a lower tolerance is a good choice as is a relatively small number of maximum inner iterations (3-10) per group. However in the non-preconditioned case the possibility exists that setting the tolerance too low for JFNK(PI) can significantly increase the total number of iterations required.

Before directly comparing results of the Newton-based approaches to power iterations it is worthwhile to examine the convergence rate of the Newton-based approaches and what factors play a role in determining this convergence. The base-case input parameters in this set of tests will tend towards converging any iterative process almost as tightly as possible since execution time is not a concern for this part of the study. The eigenpair is converged using the tight convergence

Table I. Power-Based Results for Diffusion Test Problems, Tight Convergence

Problem	Method	k_{eff}	Newton (GMRES)	Outer	Inner	Time (s)
IAEA	Power	1.029588	—	701	49070	96.67
	Chebyshev	1.029588	—	369	25798	50.20
	JFNK(PI)	1.029588	15(168)	204	13793	26.24
	Reference	—	—	—	—	—
Biblis	Power	1.025124	—	947	42728	102.40
	Chebyshev	1.025124	—	282	12787	33.28
	JFNK(PI)	1.025124	17(150)	190	8850	21.94
	Reference	—	—	—	—	—
BWR	Power	0.996120	—	1146	60625	104.10
	Chebyshev	0.996120	—	324	17403	30.56
	JFNK(PI)	0.996120	12(188)	218	12098	21.42
	Reference	—	—	—	—	—
CANDU	Power	0.981990	—	407	28490	74.66
	Chebyshev	0.981990	—	367	22834	63.52
	JFNK(PI)	0.981990	21(145)	193	13022	34.26
	Reference	—	—	—	—	—

criteria and the GMRES iterations (the linearized Newton step) use a maximum dimension size of 1000 with 50 restarts with a constant γ of 10^{-12} . The inner iterations are terminated after 1000 iterations in a group or when the error has been reduced by a factor of 10^{-12} .

This set of experiments was only run for the IAEA benchmark problem and the results are plotted in Fig. 2. This plot shows the convergence of the Newton-based methods for the base case described and also shows the effects of γ on the convergence rate. This effect is most clear for $\gamma = 10^{-1}$ where it can be seen that the number of Newton iterations required to converge has more than doubled. The plot for the base case illustrates the effects of the Jacobian-Free approximation as seen by the differences between JFNK(GEP) and Newton-Krylov, which are really quite minor. Fig. 3 shows the effects of varying the GMRES iteration properties on convergence. The base case is the same as before but with $\gamma = 10^{-4}$, while in the other plots the number of iterations allowed between restarts for GMRES is varied such that the total number of iterations allowed is always 100, e.g. restart=50 means 50 iterations were performed between restart and with 1 restart was performed yielding 100 allowable iterations. This change seemingly has very little effect on the JFNK(PI) method while the convergence behavior of the generalized eigenvalue problem approaches is impacted significantly. To avoid this type of breakdown when using these methods it is believed the GMRES iterations must be properly preconditioned, whereas no preconditioners were used with these methods in any of the experiments described here.

Comparisons done with power iteration using the tight convergence criteria set and

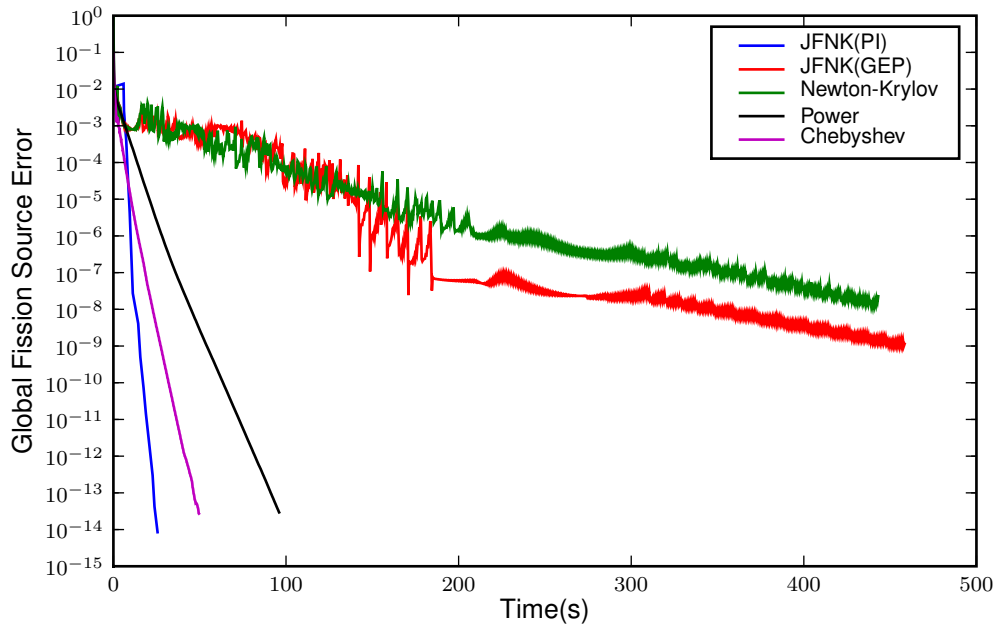


Figure 4. Convergence of All Methods for IAEA Problem, Tight Convergence

Table II. IC Preconditioned Power-Based Results for IAEA Benchmark

	$\Delta(\text{cm})$	k_{eff}	Newton (GMRES)	Outer	Inner	Time (s)
Power	10	1.029523	—	136	412	0.392
Chebyshev	10	1.029531	—	73	222	0.220
JFNK(IC-PI)	10	1.029524	5(25)	36	125	0.112
Power	5	1.029190	—	136	686	4.688
Chebyshev	5	1.029184	—	77	390	2.704
JFNK(IC-PI)	5	1.029191	4(22)	31	173	1.084
Power	2.5	1.029420	—	137	1234	67.800
Chebyshev	2.5	1.029428	—	72	646	36.070
JFNK(IC-PI)	2.5	1.029421	4(23)	32	300	16.360
Power	1.25	1.029560	—	139	1390	1053.000
Chebyshev	1.25	1.029555	—	87	870	659.500
JFNK(IC-PI)	1.25	1.029561	4(30)	39	390	296.900
Reference	—	1.029585	—	—	—	—

non-preconditioned inner iterations are given in Table I. In this set of runs $\epsilon = \sqrt{(1 + \|u\|) \epsilon_{\text{mach}}}$ (ϵ_{mach} is machine precision), Eis-B was used to determine the forcing factor, and inner iterations were terminated after either 35 iterations in a group or when the error was reduced by a factor of 10^{-2} . From the table it can be seen that for these input parameters JFNK(PI) offers an improvement in convergence of the power iteration and the Chebyshev accelerated implementation. In some cases the reduction in total iteration count (and execution time) is almost 50%. Next only the IAEA problem is considered and the JFNK(GEP) and Newton-Krylov approaches are included. Since the iteration counts cannot be directly compared, the global flux error, ϵ_2 in Eq. (23), is plotted in Fig. 4 as a function of execution time. In these runs GMRES was permitted 100 total iterations, with 10 iterations between restarts. The need for preconditioning of the linearized Newton step for the generalized eigenvalue problem approaches is underscored in this case as these methods fail to converge within 1000 Newton iterations due to the inability of the GMRES iterations to adequately solve the linear Newton step.

Another comparison was performed using preconditioned inner iterations for the IAEA problem, with mesh refinement. While the results in Table II show a drastic reduction in the overall number of inner iterations when compared to Table I, due to preconditioning the execution times are much higher, which is likely due to an inefficient implementation in the software. Examining these results it can be seen the JFNK(IC-PI) method bests both the accelerated and unaccelerated power method for each level of mesh refinement. The number of outer and inner iterations required are substantially lower for the JFNK method with the run time of the method also reflecting this fact. In each case we see the JFNK method converges to the fundamental mode, which is not guaranteed. Although successive applications of the power iteration will invariably converge to the fundamental mode, the nonlinear equations Γ_{acc} and Γ_{gep} are satisfied by any mode. Though it is possible to converge to any eigenpair, this has only been witnessed when the initial guess is artificially constructed to very closely approximate another eigenpair.

4. CONCLUSIONS

In this work three Newton-based methods have been developed which are capable of solving the k-eigenvalue problem associated with the multigroup diffusion equations. Recognizing that the traditional power iteration can be written as a nonlinear system of equations it is possible to cast an inexact Newton method employing the Jacobian-Free approximation as an accelerated power iteration. This allows for an accelerated power iteration scheme which retains the outer-inner solution procedure of the multigroup diffusion equations. Two related methods are also developed which utilize Newton's method to compute the fundamental mode. One of these methods again uses the JFNK structure but abandons the outer-inner iteration structure such that the only linear system to be solved is the Newton step (and not within-group equations). The second of these methods does not rely on the JFNK approximation but rather constructs the true Jacobian-vector product. Both of these methods are shown to be composed almost entirely of operations that are already performed during the standard power iteration process. These two methods also have the benefit of not requiring any special treatment of upscattering as there are no longer any within-group iterations: each group is given equal treatment. These methods are presently being examined in the context of transport theory rather than the diffusion approximation. Preliminary results obtained using the JFNK method as an acceleration technique for power iterations when applied to the 3-D transport problem are reported in [20].

Numerical results were generated using a test code which implements the Newton-based methods and the traditional power method (unaccelerated and with Chebyshev acceleration). It has been shown that given sufficient convergence of iterative processes each of the Newton-based methods will converge to an eigenvalue / eigenvector pair. It has been observed that in practice the only time these methods do not converge to the fundamental mode is if an initial guess is supplied that is artificially close to a higher mode. The implementation of the JFNK method as an acceleration technique for the standard power iteration is the most robust and competitive of the Newton-based methods as evidenced by numerical results. Though in a tightly converged situation the convergence rate of this approach is generally less than the other Newton approaches, it is much less susceptible to breaking down due to non-convergence of the GMRES iteration. It has been shown in certain situations that this method can outperform the Chebyshev accelerated power iteration in terms of iteration count and execution time; the advantages are especially clear when a high precision estimate of the eigenpair is sought.

The Newton-based approaches developed for the generalized eigenvalue problem are not power iteration dependent, instead these methods rely entirely on matrix vector products in the evaluation of the nonlinear function, meaning there is no intermediate inverse operator (inner iterations). Also, it is not necessary to perform upscattering iterations if upscattering is present. It has been shown that the operators used in these methods can be constructed on a group-wise basis, reducing the amount of modification necessary to current solution structures. Numerical results for these methods show that given tight convergence of the GMRES iteration the convergence rates are superior to the JFNK acceleration of the power method. The approach which utilizes the action of the true Jacobian performs slightly better than the approach which approximates the Jacobian action via JFNK. Trouble arises with these methods, however, when the GMRES iterations are not sufficiently converged, which happens frequently unless a large number of iterations are permitted and a large number of basis vectors are accumulated. Setting the total number of iterations and the maximum GMRES subspace size to practical values destroys the utility of these methods. Using a preconditioner for the linearized Newton problem may allow for a reasonable number of GMRES iterations to be used such that the other benefits of these approaches are fully realized.

ACKNOWLEDGEMENTS

This research was performed under appointment of the first author to the Rickover Graduate Fellowship Program sponsored by Naval Reactors Division of the U.S. Department of Energy.

REFERENCES

- [1] E.L. Wachspress, *Iterative Solution of Elliptic Systems*, Prentice-Hall, Englewood Cliffs, NJ, USA (1966).
- [2] D.A. Hageman and D.M. Young, *Applied Iterative Methods*, Dover, Mineola, New York, USA (1981).
- [3] T. Downar, D. Lee, Y. Xu, T. Kozlowski, J. Staudenmeier, "PARCS v2.6 US NRC Core Neutronics Simulator- Theory Manual," Purdue University/NRC, 2004.

- [4] V. Vidal, G. Verdú, D. Ginestar, and J. L. Muñozcobo, "Variational Acceleration for Subspace Iteration Method, Application to Nuclear Power Reactors," *International Journal for Numerical Methods in Engineering*, **41**, pp. 391-407 (1998).
- [5] J. Warsa, T. A. Wareing, J. E. Morel, and J. M. McGhee, "Krylov Subspace Iterations for Deterministic k-Eigenvalue Calculations," *Nuclear Science and Engineering*, **147**, pp. 26-42 (2004).
- [6] G. Verdú, R. Miró, D. Ginestar, and V. Vidal, "The Implicit Restarted Arnoldi Method, An Efficient Alternative to Solve the Neutron Diffusion Equation," *Annals of Nuclear Energy*, **26**, pp. 579-93 (2002).
- [7] V. Mahadevan and J. Ragusa, "Novel Hybrid Scheme to Compute Several Dominant Eigenmodes for Reactor Analysis Problems," *Proceeding of International Conference on the Physics of Reactors*, Interlaken, Switzerland, September 14-19 (2008).
- [8] D. A. Knoll and D. E. Keyes, "Jacobian-Free Newton-Krylov Methods: A Survey of Approaches and Applications," *Journal of Computational Physics*, **193**, pp. 357-397 (2004).
- [9] J.J. Duderstadt and L.J. Hamilton, *Nuclear Reactor Analysis*, John Wiley and Sons (1976).
- [10] Y. Xu and T. Downar, "The Implementation of Matrix Free Newton/Krylov Methods Based on a Fixed Point Iteration," *Proceeding of Mathematics and Computation, Supercomputing, Reactor Physics, and Nuclear and Biological Applications*, Avignon, France, September 12-15 (2005).
- [11] P. Brown and Y. Saad, "Convergence of Nonlinear Newton-Krylov Algorithms," *SIAM Journal of Optimization*, **4**, pp. 297-330 (1994).
- [12] G. W. Stewart, *Matrix Algorithms, Volume II: Eigensystems*, SIAM (2001).
- [13] Mark K. Seager, "A SLAP for the Masses," *Lawrence Livermore National Laboratory: UCRL-100267*, December (1998).
- [14] D. R. Ferguson and K. L. Derstine, "Optimized Iteration Strategies and Data Management Considerations for Fast Reactor Finite Difference Diffusion Theory Codes," *Nuclear Science and Engineering*, **64**, pp. 593-604 (1977).
- [15] "Argonne Code Center: Benchmark Problem Book," ANL-7416, Argonne National Laboratory (1977).
- [16] A. Hébert, "Application of the Hermite Method for Finite Element Reactor Calculations," *Nuclear Science and Engineering*, **91**, pp. 34-58 (1985).
- [17] Y. Xu and T. J. Downar, "Optimal Perturbation Size of Matrix Free Newton/Krylov Methods," *Proceedings of Joint Int. Topical Meeting on Math. & Comp. and Supercomputing in Nucl. Apps.*, Monterey, California, April (2007).
- [18] S. Eisenstat and H. Walker, "Choosing the Forcing Terms in an Inexact Newton Method," *SIAM Journal of Scientific Computing*, **17** pp.16-32 (1996).
- [19] H. An, et. al, "A Choice of Forcing Terms in Inexact Newton Method," *Journal of Computational and Applied Mathematics*, **200** pp. 47-60 (2007).
- [20] D. F. Gill and Y. Y. Azmy, "Jacobian-Free Newton-Krylov as an Alternative to Power Iterations for the k-Eigenvalue Transport Problem," *Transactions of the American Nuclear Society*, Atlanta, GA, USA June 14-18 (2009).