The Newton-Krylov Method Applied to Negative-Flux Fixup in S_N Transport Calculations

Erin D. Fichtl,* James S. Warsa, and Jeffery D. Densmore

Los Alamos National Laboratory, Computer, Computational and Statistical Sciences Division
Computational Physics Group, Los Alamos, New Mexico 87545

Received July 23, 2009
Accepted December 28, 2009

Abstract—*Under some circumstances, spatial discretizations of the S_N transport equation will lead to negativity in the scalar flux; therefore, negative-flux fixup schemes are often employed to ensure that the flux is positive. The nonlinear nature of these schemes precludes the use of powerful linear iterative solvers such as Krylov methods; thus, solutions are generally computed using so-called source iteration (SI), which is a simple fixed-point iteration. In this paper, we use Newton's method to solve fixed-source S_N transport problems with negative-flux fixup, for which the analytic form of the Jacobian is shown to be nonsingular. It is necessary to invert the Jacobian at each Newton iteration. Generally, an exact inversion is prohibitively expensive and furthermore is not necessary for convergence of Newton's method. In the inexact Newton-Krylov method, the Jacobian is inverted using a Krylov method, which completes at some prescribed tolerance. This tolerance may be quite large in the initial stages of the Newton iteration. In this paper, we compare the use of the exact Jacobian with two approximations of the Jacobian in the inexact Newton-Krylov method. The first approximation is a finite difference approximation. The second is that used in the Jacobian-free Newton-Krylov (JFNK) method, which performs a finite difference approximation without actually generating the Jacobian itself. Numerical results comparing standard SI with the three methods demonstrate that Newton-Krylov can outperform SI, particularly for diffusive materials. The results also show that the additional level of approximation introduced by the JFNK approach does not adversely affect convergence, indicating that JFNK will be robust and efficient in large-scale applications.*

I. INTRODUCTION

The use of an unrefined spatial discretization or inadequate Legendre polynomial expansion of the anisotropic scattering cross section can cause angular or scalar flux solutions of the transport equation to become negative. Without loss of generality, we consider problems with isotropic scattering only, so that the sole cause of negative fluxes is due to the spatial discretization. Although such negativities are mathematically correct for the spatially and angularly discretized transport equation, the angular and scalar fluxes are positive quantities in reality, and negative fluxes are undesirable for a variety of reasons.[1]

In order to restore positivity of the flux, set-to-zero negative-flux fixup (NFF) schemes are standard in transport algorithms.[2] These schemes operate by setting negative angular fluxes to zero as they are produced in the course of a transport sweep. The use of synthetic acceleration schemes in combination with NFF can also drive the spectral radius above unity, causing the source iteration (SI) method, which has historically been used to solve fixed-source transport problems, to diverge. The nonlinear nature of NFF, and of some other approaches to negative-flux mitigation,[3–5] precludes the direct application of more powerful Krylov iterative methods.[6–9] However, there are two ways in which we can indirectly utilize Krylov iterative methods. The first is to "fix the fixup." In Ref. 10, the authors have proposed a linearization scheme that alternates between SI with NFF and Krylov iterations. The system is linearized by applying a single SI with NFF and recording the location in space and angle of the fixups. The same set of fixups is then applied at each transport sweep in the next set of Krylov iterations. This so-called hybrid scheme outperforms standard SI while arriving at the same positive solution. Our

*E-mail: efichtl@lanl.gov

second option is to use Newton's method. There has been recent interest in using Newton's method to solve both k-eigenvalue problems recast as nonlinear problems and inhomogeneous source problems with nonlinear acceleration schemes,[11–15] and the success of these investigations led us to consider using Newton's method to solve fixed-source transport problems with nonlinear set-to-zero NFF schemes.

Newton's method is commonly used to solve nonlinear problems in a variety of fields and can achieve quadratic convergence given a good initial guess.[16] Furthermore, it allows us to capitalize on the efficiency of a Krylov iterative method by using it to perform the inner inversion of the Jacobian, which is a linear operator, at each Newton iteration. This approach to Newton's method is known as the Newton-Krylov (NK) method. One notable and popular variation on the NK method is the Jacobian-free Newton-Krylov (JFNK) method. JFNK introduces another level of approximation in the NK method by using a finite difference to approximate the action of the Jacobian on a vector at every inner Krylov iteration. It has the distinct advantage that formation and storage of the Jacobian are not required; instead, a single nonlinear function evaluation is performed at each inner Krylov iteration.[17] In this paper, we will show that we can apply the JFNK method to solve fixed-source S_N transport problems with NFF. Comparing JFNK to NK and SI indicates that the Jacobian-free approximation does not adversely affect convergence of NK and that JFNK can be far more efficient than SI, particularly in diffusive materials.

This paper is organized as follows. In Sec. II, we describe the two-dimensional (2-D) transport problem that we are solving, discuss spatial and angular discretizations, reformulate the problem as a matrix equation, and detail the NFF scheme we consider. In Sec. III, we describe the NK method and the popular JFNK variation, detail the application of the NK method to the solution of the transport equation, and derive an explicit form of the Jacobian, which is later used to compare standard NK with JFNK. Numerical results are presented in Sec. IV, and in Sec. V, we discuss our conclusions and suggestions for future work.

II. NUMERICAL SOLUTION OF THE TRANSPORT EQUATION

The transport of monoenergetic particles through nonmultiplying materials with isotropic scattering is described by the following form of the Boltzmann transport equation:

$$[\hat{\Omega}\cdot\nabla + \sigma(r)]\psi(r,\hat{\Omega}) = \sigma_s(r)\phi(r,\hat{\Omega}) + q(r,\hat{\Omega}) \ , \tag{1}$$

where

$\psi(r,\hat{\Omega})$ = angular flux

$\phi(r,\hat{\Omega})$ = scalar flux

$q(r,\hat{\Omega})$ = radiation source

$\sigma(r)$ = total cross section

$\sigma_s(r)$ = scattering cross section

r = spatial variable

$\hat{\Omega}$ = angular variable.

II.A. Discretization

Two-dimensional Cartesian geometry is used; therefore, the angular variable is a function of the polar angle θ, which is measured with respect to the x-axis in this implementation, and the azimuthal angle ω, which is measured with respect to the y-axis in the yz-plane. Thus, the x-component of $\hat{\Omega}$ is of length $\mu = \cos(\theta)$ and the y-component of $\hat{\Omega}$ is given by $\eta = \sqrt{1-\mu^2}\cos(\omega)$. A discrete ordinates, or S_N, discretization is applied to the angular variable, and level symmetric quadrature[2] abscissas are used to replace $\hat{\Omega}$ with discrete values of μ and η, which are both defined on the interval $[-1,1]$. In 2-D Cartesian geometry, for a quadrature order of N, there are $N_A = N(N+2)/2$ ordinates, and Eq. (1) can be rewritten as

$$\left[\mu_n\frac{\partial}{\partial x} + \eta_n\frac{\partial}{\partial y} + \sigma(r)\right]\psi_n(r) = \sigma_s(r)\phi(r) + q_n(r) \ , \tag{2}$$

where n denotes a particular angular abscissa. The scalar flux is approximated using the level symmetric quadrature set

$$\phi(r) = \sum_{n=1}^{N_A} w_n\psi_n(r) \ ,$$

where w_n is the n'th quadrature weight. Finally, a linear discontinuous (LD) finite element discretization is applied to the spatial variable. A triangular mesh of N_c triangles, denoted by T_i, is assigned to the system. The fluxes and sources are then written in terms of the linear barycentric basis functions on triangles $B_k^i(r)$ (Ref. 18):

$$\psi^i(r) = \sum_{k=0}^{2}\psi_k^i B_k^i(r) \ , \quad r\in T_i \ . \tag{3}$$

The LD finite element method equations are obtained by substituting Eq. (3) into Eq. (2) and taking the inner product of the resulting equation with each basis function. Finally then, the transport equation has been reduced to a series of $3N_cN_A$ equations for as many unknown values of the fully discretized angular flux. These equations can also be written in terms of the $3N_c$ components of the spatially discretized scalar flux.

II.B. Operator Notation

This system of equations can be written as a matrix equation of the form

$$\mathbf{L}\psi = \mathbf{MSD}\psi + q \ , \tag{4}$$

where ψ and q are vectors containing the $3N_c N_A$ components of the angular flux and radiation source, respectively. $\mathbf{L}$ is the sum of the streaming operator [effecting the action of $\mu_n(\partial/\partial x) + \eta_n(\partial/\partial y)$] and the removal operator [effecting the action of $\sigma(r)$]. $\mathbf{D}$ is the discrete-to-moment operator, which operates on the angular flux to produce the scalar flux, i.e., $\mathbf{D}\psi = \phi$. $\mathbf{S}$ is the scattering-source matrix and $\mathbf{M}$ is the moment-to-discrete operator, which operates on a vector the size of the scalar flux to produce a vector the size of the angular flux. Traditionally, the system is solved using a Richardson (linear fixed-point) iteration, commonly known as source iteration when applied to transport calculations. The iteration is given by

$$\phi^{(z+1)} = \mathbf{DL}^{-1}(\mathbf{MS}\phi^{(z)} + q) \ . \tag{5}$$

II.C. Negative-Flux Fixup

A spatially unrefined LD discretization can cause the transport solution to yield negative values of the flux, which is physically a strictly positive quantity. The standard method for dealing with negativity is to "fix up" the flux by setting negative angular fluxes to zero as they are produced. The remaining fluxes in the cell then have to be recomputed to ensure that particle balance is preserved. This is done by ensuring that the integrals of the left and right sides of the transport equation over the volume of the cell are equal. Furthermore, in 2-D, an additional condition must be imposed if it is necessary to rebalance two of the three fluxes. This condition is given by

$$\frac{\psi_j}{\psi_k} = \frac{\psi_j^R}{\psi_k^R} \ , \tag{6}$$

where ψ_j and ψ_k are the two positive fluxes computed during the transport sweep, and ψ_j^R and ψ_k^R are the rebalanced fluxes.[10]

The matrix $\mathbf{L}$ is inverted by performing a sweep through the spatial mesh for each discrete angular direction. We note that neither $\mathbf{L}$ nor $\mathbf{L}^{-1}$ is formed explicitly; the sweep, which efficiently and directly inverts $\mathbf{L}$, does not require it. The sweep marches across the spatial domain in a specific order so that the fluxes incident on a cell are computed prior to the fluxes in that cell. This ordering varies for each angular ordinate and, for the 2-D triangular mesh, there is at least one and at most two incident angular fluxes per ordinate. $\mathbf{L}$ consists of 3×3 blocks that multiply the three flux unknowns in a particular cell for a particular angle and each step in the sweep considers three rows of the matrix. The off-diagonal blocks operate on the incident fluxes, which have been previously computed in the sweep. These terms are brought to the right side, and the 3×3 matrix on the diagonal of $\mathbf{L}$ is then inverted onto the right side. Thus, the angular flux at ordinate n in cell i is computed by

$$\Psi_{i,n} = \mathbf{L}_{i,n}^{-1}\left(v_{i,n} - \sum_{i'=1}^{N_c^{inc}} \mathbf{L}_{i'\to i,n}\psi_{i',n}\right) = \mathbf{L}_{i,n}^{-1}y_{i,n} \ , \tag{7a}$$

where i' denotes spatial cells whose angular flux at ordinate n is incident on cell i, of which there are N_c^{inc};

$$v_{i,n} = \mathbf{S}_i\phi_i + Q_{i,n}$$

contains the scattering and radiation sources; and $y_{i,n}$ is the right side of the sweep equation. The fixup scheme can then be written as a two-step process:

$$\Psi_{i,n}^F = 0.5(|\Psi_{i,n}| + \Psi_{i,n}) = H(\Psi_{i,n})\bullet\Psi_{i,n} = R(\Psi_{i,n}) \tag{7b}$$

and

$$\Psi_{i,n}^R = \Psi_{i,n}^F \frac{p\left(v_{i,n} - \sum_{i'=1}^{N_c^{inc}} \mathbf{L}_{i'\to i,n}\psi_{i',n}\right)}{p\mathbf{L}_{i,n}\Psi_{i,n}^F} = \Psi_{i,n}^F \cdot r_{i,n} \ , \tag{7c}$$

where $|x|$ denotes the absolute value applied to each element of x and $\bullet$ denotes the Schur product, in which the terms in the same position of two matrices (vectors) of equal size are scalar multiplied and placed in a third matrix (vector) of equal size. $H(x)$ denotes the Heaviside step function applied to each element of x, where we specifically define $H(0) = 1$ so that NFF is not applied if the flux is already zero, and $R(x)$ denotes the ramp function applied to each element of x. The row vector $p = [1 \ \ 1 \ \ 1]$ sums the terms in a column vector, and the "rebalance factor" in Eq. (7c), which is a scalar quantity, is denoted by $r_{i,n}$. If $\psi_{i,n} = \Psi_{i,n}$, then no fixup is conducted and the system is linear. The scheme in which $\psi_{i,n} = \Psi_{i,n}^R$ is termed set-to-zero fixup with rebalance.

This NFF scheme is clearly nonlinear, but it is still possible to solve the system using fixed-point iteration, which for nonlinear problems is generally known as nonlinear Richardson or Picard iteration[19]:

$$\phi^{(z+1)} = \mathbf{D}\tilde{\mathbf{L}}^{-1}(\phi^{(z)})(\mathbf{MS}\phi^{(z)} + q) \ . \tag{8}$$

The fixup action is conducted as $\mathbf{L}$ is being inverted; therefore, the operator being applied to Ψ in Eqs. (7) is nonlinear, is dependent on ϕ, and is denoted by $\tilde{\mathbf{L}}^{-1}(\phi)$

to differentiate between the linear and nonlinear operators. We note that in Eq. (8), the flux fixup is a function of the current scalar flux vector $\phi^{(z)}$; therefore, the equation is in fact a Picard linearization of the transport equation with NFF.

III. SOLVING THE TRANSPORT EQUATION USING JACOBIAN-FREE NEWTON-KRYLOV

III.A. Jacobian-Free Newton-Krylov

Newton-Krylov methods combine Newton's method, which is commonly used to solve nonlinear equations and can achieve super-linear convergence, with Krylov subspace methods for solving the Newton correction equations. Newton iteration solves the equation $F(u) = 0$ for u, where F is generally a nonlinear operator. First, a Taylor expansion is taken about the current iterate u_z:

$$F(u_{z+1}) = F(u_z) + F'(u_z)(u_{z+1} - u_z) + \dots \ , \qquad (9)$$

where z is the iteration index. Setting $F(u_{z+1}) = 0$ and neglecting higher-order terms yields an iteration of the form

$$\mathbf{J}(u_z)\cdot\delta u_z = -F(u_z) \qquad (10a)$$

and

$$u_{z+1} = u_z + \delta u_z \ , \qquad (10b)$$

where $\mathbf{J}(u_z) \equiv F'(u_z)$ is the Jacobian with individual elements given by

$$J_{ij} = \frac{\partial F_i(u)}{\partial u_j} \ . \qquad (11)$$

The iteration terminates when $\|F(u_{z+1})\|$ is sufficiently small, where $\|\cdot\|$ denotes the L_2-norm. The first step in the iteration is to solve Eq. (10a) for the Newton correction, δu_z. Direct inversion of the Jacobian is generally prohibitively expensive; therefore, in practice $\mathbf{J}^{-1}F$ is often approximated using a Krylov iterative method—in this case, the generalized minimum residual method (GMRES). We note that if the system is linear and therefore of the form $\mathbf{A}u = b$, then $F(u) = \mathbf{A}u - b$ and the Jacobian is simply $\mathbf{A}$. At the first inner Krylov iteration,

$$\delta u_0 \approx \mathbf{A}^{-1}(-\mathbf{A}u_0 + b) = -u_0 + \mathbf{A}^{-1}b$$

and

$$u_1 \approx u_0 - u_0 + \mathbf{A}^{-1}b = \mathbf{A}^{-1}b \ .$$

The solution is approximate because $\mathbf{A}$ is not inverted exactly, however, if the tolerance for the Krylov iteration is small enough with respect to the tolerance for the Newton iteration, the Newton iteration will converge in a single iteration. This is equivalent to solving the system using the Krylov method employed on the inner iteration; therefore, as the mesh is refined and the number of fixups drops, NK limits to a single linear Krylov solve.

Krylov methods require the application of the operator $\mathbf{J}(u_z)$ on the vectors returned by the solver; however, it is not always possible to form the Jacobian. Furthermore, it may be very computationally demanding to form and store the Jacobian even if it is possible to do so, so it is desirable to perform the action of the operator without actually generating it. In either case, the Jacobian-free approach can be used to approximate the action of the Jacobian on the vector v by[20,21]

$$\mathbf{J}(u)\cdot v \approx \frac{F(u + \epsilon v) - F(u)}{\epsilon} \ . \qquad (12)$$

The error in $\mathbf{J}(u)\cdot v$ is proportional to ϵ; thus, if ϵ is too large, the error in the approximation will be large. If ϵ is too small, however, the approximation will be contaminated by round-off error. In order to stay away from these extremes, ϵ can be chosen as a function of the norms of u and v at each evaluation of $\mathbf{J}$ using[17]

$$\epsilon = \frac{\sqrt{\epsilon_{mach}}(1 + \|u\|)}{\|v\|} \ , \qquad (13)$$

where ϵ_{mach} indicates machine epsilon, which is generally on the order of 10^{-16}. We also note that since $F(u)$ was computed in the previous iteration in order to check convergence, a single function evaluation, $F(u + \epsilon v)$, is needed to compute the action of the Jacobian at each GMRES iteration.

At each Newton iteration, it is not necessary to solve Eq. (10a) to low tolerance. In fact, "over-solving" the correction equation in the early Newton iterations may actually be detrimental to performance.[17] Inexact Newton methods capitalize on this fact, converging the residual in the inner Krylov iteration for Newton iteration $(z + 1)$ to a tolerance of $\eta_z\|F(u_z)\|$ where the forcing term η_z is some value less than unity. Thus, the inner Krylov iteration completes when

$$\|\mathbf{J}(u_z)\delta u_z + F(u_z)\| \le \eta_z\|F(u_z)\| \ .$$

When η_z is large, the inner Krylov iteration will converge quickly, but more Newton iterations will most likely be required. On the other hand, if η_z is small, the number of inner Krylov iterations at each Newton iteration increases, although there may be fewer Newton iterations. Thus, there is a trade-off between the effort required to perform the inner Krylov iterations and the number of Newton iterations.

III.B. Application to the Transport Equation

For the transport equation, F is given by

$$\begin{aligned} F(\phi) &= \phi - \mathbf{D}\tilde{\mathbf{L}}^{-1}(\phi)(\mathbf{MS}\phi + q) \\ &= \phi - \mathbf{D}\tilde{\mathbf{L}}^{-1}(v(\phi))v(\phi) = 0 \ , \end{aligned} \qquad (14)$$

where F is rewritten in terms of $v(\phi) = (\mathbf{MS}\phi + q)$, as this is the vector that $\tilde{\mathbf{L}}^{-1}$ operates on. The Jacobian is found by taking the derivative of F with respect to the unknown, in this case ϕ. Because $\tilde{\mathbf{L}}^{-1}$ is applied to v, it is more straightforward to take the derivative with respect to v and then multiply by $\partial v/\partial\phi = \mathbf{MS}$. The Jacobian is then

$$\mathbf{J} = \frac{\partial F}{\partial\phi} = \frac{\partial F}{\partial v}\cdot\frac{\partial v}{\partial\phi} = \mathbf{I} - \mathbf{D}\,\frac{\partial(\tilde{\mathbf{L}}^{-1}v)}{\partial v}\,\mathbf{MS}$$

$$= \mathbf{I} - \mathbf{D}\left(\tilde{\mathbf{L}}^{-1} + \frac{\partial\tilde{\mathbf{L}}^{-1}}{\partial v}\otimes v\right)\mathbf{MS}\ , \tag{15}$$

where $\otimes$ denotes the Kronecker product. Clearly, if no negative fluxes are produced (i.e., the mesh is sufficiently fine) or fixup is off, $\partial\tilde{\mathbf{L}}^{-1}/\partial v = 0$, and the Jacobian is simply

$$\mathbf{J} = \mathbf{I} - \mathbf{DL}^{-1}\mathbf{MS}\ .$$

The form of the Jacobian is of great importance, as a Jacobian that is singular at one of the Newton iterates or the solution itself may result in nonconvergence of the inner Krylov iteration. As we will show below, the Jacobian does contain discontinuities, but it is finite at every point in the domain and it cannot be singular.

In Eq. (7c) $\tilde{\mathbf{L}}^{-1}v$ is equivalent to Ψ^R; therefore, the Jacobian can be formed explicitly if we can take the derivative of Ψ^R with respect to v. This is best done by taking the derivatives of Eqs. (7) with respect to $v_{i,n}$ and $v_{i'',n}$, where i'' denotes not only cells with fluxes incident on cell i (cells i') but also those cells with fluxes incident on all cells that have come before cells i' in the sweep pattern for angle n. Writing down the derivative of $\tilde{\mathbf{L}}^{-1}v$ is involved, but its action is easily summarized. We begin by taking the derivative of $\Psi^R_{i,n}$ [Eq. (7c)] with respect to $v_{i,n}$,

$$\frac{\partial\Psi^R_{i,n}}{\partial v_{i,n}} = r_{i,n}\,\frac{\partial\Psi^F_{i,n}}{\partial v_{i,n}} + \Psi^F_{i,n}\,\frac{\partial r_{i,n}}{\partial v_{i,n}}$$

$$= r_{i,n}H(\Psi_{i,n})\bullet\mathbf{L}^{-1}_{i,n} + \frac{\Psi^F_{i,n}p}{p\mathbf{L}_{i,n}\Psi^F_{i,n}}$$

$$\times\,(\mathbf{I} - \mathbf{L}_{i,n}(r_{i,n}H(\Psi_{i,n})\bullet\mathbf{L}^{-1}_{i,n}))$$

$$= \tilde{\mathbf{L}}^{-1}_{i,n} + \frac{\Psi^F_{i,n}p}{p\mathbf{L}_{i,n}\Psi^F_{i,n}}\,(\mathbf{I} - \mathbf{L}_{i,n}\tilde{\mathbf{L}}^{-1}_{i,n})\ , \tag{16}$$

where $\tilde{\mathbf{L}}^{-1}_{i,n} = r_{i,n}H(\Psi_{i,n})\bullet\mathbf{L}^{-1}_{i,n}$ is the operator that is applied to $y_{i,n} = (v_{i,n} - \sum_{i'=1}^{N_c^{inc}}\mathbf{L}_{i'\to i,n}\Psi_{i',n})$ [see Eq. (7a)] in order to obtain $\Psi^R_{i,n}$. This operator can be applied to any vector but will perform the fixups as though it is being applied to $y_{i,n}$. Numerically, this is accomplished by storing the multipliers used in the computation of $\tilde{\mathbf{L}}^{-1}v$ and then applying them in place of Eqs. (7b) and (7c) in the sweep. The next step is to take the derivative with respect to $v_{j,n}$, $j \neq i$,

$$\frac{\partial\Psi^R_{i,n}}{\partial v_{j,n}} = r_{i,n}\,\frac{\partial\Psi^F_{i,n}}{\partial v_{j,n}} + \Psi^F_{i,n}\,\frac{\partial r_{i,n}}{\partial v_{j,n}}$$

$$= -\tilde{\mathbf{L}}^{-1}_{i,n}\sum_{i'=1}^{N_c^{inc}}\mathbf{L}_{i'\to i,n}\,\frac{\partial\Psi^R_{i',n}}{\partial v_{j,n}} - \frac{\Psi^F_{i,n}p}{p\mathbf{L}_{i,n}\Psi^F_{i,n}}$$

$$\times\,(\mathbf{I} - \mathbf{L}_{i,n}\tilde{\mathbf{L}}^{-1}_{i,n})\sum_{i'=1}^{N_c^{inc}}\mathbf{L}_{i'\to i,n}\,\frac{\partial\Psi^R_{i',n}}{\partial v_{j,n}}$$

$$= -\left(\tilde{\mathbf{L}}^{-1}_{i,n} + \frac{\Psi^F_{i,n}p}{p\mathbf{L}_{i,n}\Psi^F_{i,n}}\,(\mathbf{I} - \mathbf{L}_{i,n}\tilde{\mathbf{L}}^{-1}_{i,n})\right)$$

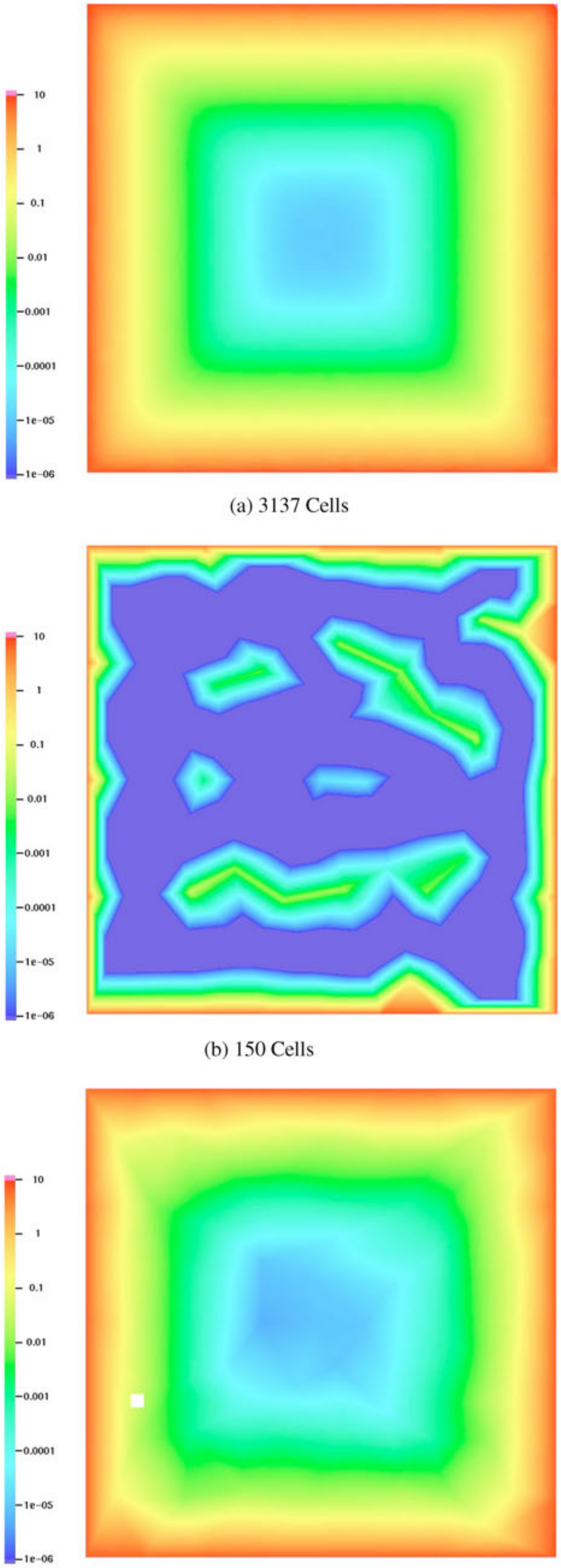
$$\times\sum_{i'=1}^{N_c^{inc}}\mathbf{L}_{i'\to i,n}\,\frac{\partial\Psi^R_{i',n}}{\partial v_{j,n}}$$

$$= -\frac{\partial\Psi^R_{i,n}}{\partial v_{i,n}}\sum_{i'=1}^{N_c^{inc}}\mathbf{L}_{i'\to i,n}\,\frac{\partial\Psi^R_{i',n}}{\partial v_{j,n}}\ , \tag{17}$$

where all $\partial\Psi^R_{i',n}/\partial v_{j,n}$ are known. If cell j precedes cell i, then $\partial\Psi^R_{i',n}/\partial v_{j,n}$ has already been computed in the course of the sweep. If there is no connection between cells i and j in the sweep, or cell j does not precede cell i, then $\partial\Psi^R_{i',n}/\partial v_{j,n} = 0$. Clearly, the mechanism necessary to perform the action of the Jacobian is already available in the form of a transport sweep algorithm. It is only necessary to store Ψ^F and the fixups applied when F is computed at the end of each Newton iteration in order to apply the action of the exact Jacobian. As stated previously, the Jacobian is not singular, but it does contain discontinuities from $\tilde{\mathbf{L}}^{-1}_{i,n}$, which contains a Heaviside step function.

IV. NUMERICAL RESULTS

The test problem is a 1-cm square with σ_t = 40.0 cm^{-1}, unit isotropic incident angular fluxes at each boundary, and a comparatively small uniform volume source of magnitude 10^{-5} in the domain to prevent small negativities that may occur when the flux solution is essentially zero. In all cases, S_4 quadrature is used and Newton's iteration is initialized with the uncollided flux, which is computed by doing a single transport sweep. NFF is used during this initial sweep only if it is to be used in the remainder of the computation. Because the scattering source is isotropic, it cannot be negative provided the scalar flux is nonnegative. The only source of negativity then derives from a spatial mesh that does not adequately resolve the solution. As can be seen in Fig. 1, for 150 spatial cells without fixup on, the solution is negative in a good portion of the domain, while with fixup on, the solution is very close to the converged solution. For the unresolved discretization when fixup is off, ~50% of the angular fluxes produced in the sweep [i.e., the result of Eq. (7a)] are negative, and there is at least one negative flux in every cell. When fixup is on, there are still negative angular fluxes produced in every cell over the course of the sweep, but only 23 to 25% of the unknowns are negative. As the grid is refined, the solution converges and fewer negative fluxes are produced. For the 3137 cell grid, a small number of negative fluxes do appear in the initial iterations, but the negativities disappear as the iterate approaches the solution, so the solutions with and without fixup are very close.



(a) 3137 Cells

(b) 150 Cells

(c) 150 Cells with Fixup

Fig. 1. Scalar flux ($\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$).

Table I shows the mean of the absolute value of the difference between various fluxes for several mesh refinements and $c = 0.7$, where $c = \sigma_s/\sigma$ is the scattering ratio. The difference flux ϕ_{diff} is computed using scalar flux values at the nodes, which are computed using the outflow angular fluxes from all of the triangles that share a vertex on the node. Potential outflow directions from a triangle at a particular node are illustrated in Fig. 2. For a particular cell, only those angles that are not incident on either of its edges adjacent to a node can contribute to the outflow at that node. Clearly, at each node only one triangle can contribute an angular flux for a particular angular abscissa, and there is exactly one angular flux per angle when contributions from all of the triangles with that node in common are combined. The first row of Table I shows the mean value of the difference between fluxes computed with (ϕ_{NFF}) and without (ϕ) fixup. The second and third rows show the mean value of the difference between a fine-mesh solution ϕ_f computed on a 7889-cell mesh, and the flux computed on the coarser meshes with and without fixup, respectively. The mesh is refined by keeping the nodes from the coarser mesh and placing additional nodes in the system; therefore, a more refined mesh will have nodes at all of the nodes in a less refined mesh. In order to compare the coarse-mesh solutions with the refined-mesh solution, we therefore look at the difference between the fluxes at the nodes of the coarse mesh. As can be seen, the discrepancy between the fine- and coarse-mesh solutions decreases as the mesh is refined and the solution converges. As the mesh is refined, the difference between ϕ and ϕ_{NFF} for a given mesh refinement also decreases as fixups become increasingly less necessary, and when the errors in ϕ_{NFF} or ϕ are not equal, ϕ_{NFF} is always more accurate than ϕ.

Table II shows a comparison of the norm of F, iteration counts, and number of transport sweeps without fixup for several different scattering ratios. For JFNK, inner iteration count refers to the total number of linear GMRES iterations conducted over the course of the outer Newton iteration, and the number of transport sweeps is equivalent to the number of times F was computed. We also note that we used the MATLAB intrinsic "gmres" to perform the inner Krylov iterations. This algorithm, which

TABLE I

Mean of the Absolute Value of the Difference Between Fluxes Computed With and Without Fixup, ϕ_{NFF} and ϕ, and Between the Solution on a Fine Mesh ϕ_f and Those on Various Coarser Meshes*

	Cells (nodes)				
ϕ_{diff}	150 (88)	312 (173)	790 (418)	1585 (823)	3137 (1613)
$\phi_{NFF} - \phi$	2.0780e−01[a]	7.5145e−02	1.3053e−02	1.0751e−03	3.9892e−11
$\phi_f - \phi_{NFF}$	3.1615e−01	2.1339e−01	9.6002e−02	4.1571e−02	2.1494e−02
$\phi_f - \phi$	4.1632e−01	2.5386e−01	1.0135e−01	4.1809e−02	2.1494e−02

*ϕ_f was computed on a mesh of 7889 triangles and 4004 nodes and $c = 0.7$.

[a] Read as 2.0780×10^{-1}.

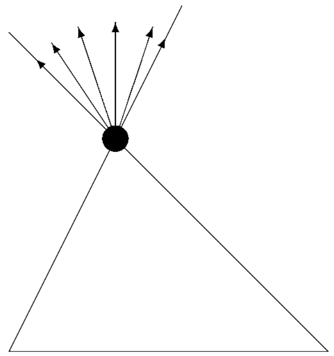


Fig. 2. Potential outflow angles for a triangle at one of its nodes.

TABLE II

Comparison of SI, GMRES, and JFNK Without Flux Fixup for $\eta = 10^{-3}$

			Iteration Count		
c	Iterative Method	$\|F\|$	Outer	Inner	Transport Sweeps
0.05	SI	3.623e−09[a]	7	—	7
	GMRES	1.284e−08	3	—	5
	JFNK	6.859e−08	2	2	9
0.4	SI	1.618e−07	18	—	18
	GMRES	5.995e−08	7	—	9
	JFNK	2.265e−07	2	6	13
0.7	SI	5.799e−07	41	—	41
	GMRES	2.788e−07	11	—	13
	JFNK	7.503e−07	2	10	17
0.9	SI	9.759e−07	127	—	127
	GMRES	6.142e−07	19	—	21
	JFNK	2.996e−09	3	25	35
0.99	SI	9.759e−07	988	—	988
	GMRES	7.606e−07	39	—	41
	JFNK	9.090e−09	3	58	68

[a] Read as 3.623×10^{-9}.

is set up to perform restarted GMRES, performs two matrix multiplies in addition to those performed at each GMRES iteration even when restart is off. As can be seen, JFNK always requires more transport sweeps than GMRES even though for small to moderate c, it requires fewer GMRES iterations overall. This is due to the additional transport sweeps required by both GMRES (two per Newton iteration, as stated previously) and Newton's iteration, for which F must be computed at the initial guess and then each time the iterate is updated (once per Newton iteration). JFNK is more efficient than SI for moderate to large c, but its relative efficiency declines with decreasing c. Table III shows a similar comparison when fixup is on. GMRES is excluded because the NFF operator is nonlinear and GMRES will not converge. JFNK still requires fewer transport sweeps than SI for large to moderate c, but by smaller margins than when fixup is off.

In order to illustrate the advantage of using an inexact Newton's method over standard Newton, Fig. 3 shows the number of sweeps (i.e., function evaluations), GMRES iterations, and Newton iterations plotted as a function of forcing parameter η for the physical system shown in Fig. 1c. At each Newton iteration, the number of sweeps is given by the number of Krylov iterations plus three. The additional three sweeps come from a single evaluation of the residual at the end of the Newton step and two sweeps in the GMRES algorithm, as noted earlier. Decreasing the number of Newton iterations can therefore be advantageous provided that this decrease is not offset by a large increase in the number of inner Krylov iterations. Lowering the forcing parameter will almost certainly result in an increase in the number of Krylov iterations at each Newton iteration and will therefore result in greater efficiency only if there is a corresponding decrease in the number of Newton iterations. As can be seen, in this case the number of Newton iterations

TABLE III

Comparison of SI and JFNK with Flux Fixup for $\eta = 10^{-3}$

c	Iterative Method	$\lvert F \rvert$	Iteration Count		Transport Sweeps
			Outer	Inner	
0.05	SI	4.876e−09[a]	7	—	7
	JFNK	3.368e−08	2	2	9
0.4	SI	1.520e−07	18	—	18
	JFNK	2.954e−09	3	9	19
0.7	SI	4.879e−07	39	—	39
	JFNK	8.684e−08	4	17	30
0.9	SI	7.951e−07	112	—	112
	JFNK	4.737e−08	5	40	56
0.99	SI	9.821e−07	984	—	984
	JFNK	4.167e−07	5	81	101

[a] Read as 4.876×10^{-9}.

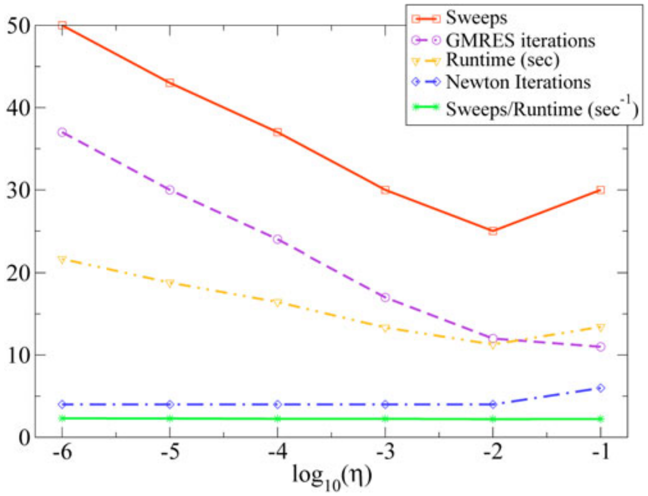


Fig. 3. Number of sweeps, number of GMRES iterations, run time, number of Newton iterations, and ratio of number of sweeps to run time given as a function of the forcing parameter η.

decreases by two when η decreases from 10^{-1} to 10^{-2}, and there is a corresponding decrease in the number of sweeps despite the fact that the number of GMRES iterations increases by one. As η decreases further, however, there is no corresponding decrease in the number of Newton iterations, but the numbers of GMRES iterations and sweeps continue to increase. Also shown are the run time and the ratio of the number of sweeps to the run time. As can be seen, the run time closely follows the number of sweeps and the ratio is essentially constant, indicating that sweep count is an excellent metric with which to measure computational effort. The reason for this is well known: A sweep is the most computationally intensive portion of a transport algorithm. Sweep count is therefore used as the sole measure of computational effort henceforth.

Because the analytic Jacobian is available to us, we can compare the convergence of NK when the exact Jacobian, a finite difference Jacobian, and JFNK are employed. The exact Jacobian was applied at each Newton iteration by storing Ψ^F and the multipliers used in the application of $\tilde{\mathbf{L}}^{-1}$ as $F(u_z)$ was being computed, where u_z is the current Newton iterate, and then applying those same multipliers whenever $\tilde{\mathbf{L}}^{-1}$ was applied in the Jacobian. The finite difference Jacobian was generated at each Newton step using a finite difference approximation where the j'th column of the matrix was computed using

$$\mathbf{J}_{:,j}(u_z) = \frac{F(u_z + \epsilon p_j) - F(u_z)}{\epsilon} \ , \qquad (18)$$

where p_j is a unit vector that is zero everywhere except row j and $\epsilon = 10 \cdot \sqrt{\epsilon_{mach}}$. And finally, the action of the Jacobian was approximated using JFNK. Table IV shows the number of Newton iterations, inner GMRES iterations, Jacobian-vector multiplies, and transport sweeps as well as the norm of F at the final solution for five different scattering ratios and $\eta = 10^{-3}$. The finite difference Jacobian differs from the other two in that there is no way to apply it without first generating it—a computationally intensive procedure requiring the computation of F (which involves a transport sweep) to generate each column. Applying the Jacobian then involves a full matrix-vector multiply. We therefore present two measures of computational effort: the number of transport sweeps and the number of times the Jacobian is applied. As can be seen, the number of GMRES iterations performed in the inversion of the Jacobian is small, so restarting is not required. In comparing the three types of Newton iteration, they are seen to be virtually indistinguishable. The number of Jacobian-vector multiplies is the same in all three cases, but using the finite difference Jacobian requires a much larger number of transport sweeps. The only other difference between the methods is in the final value of $\|F\|$, which varies slightly for NK with the finite difference Jacobian. JFNK, however, performs identically to NK with the exact Jacobian and furthermore does not require any additional function evaluations or storage of fixups. Given that its cost and performance are the same, its storage requirements smaller, and its implementation simpler than using NK with the exact Jacobian, JFNK would seem to be the method of choice.

As can also be seen, the number of Newton and inner GMRES iterations increases with increasing scattering ratio. The rate of convergence of the inner Krylov iteration is indicated by the eigenvalue spectrum of the

TABLE IV

Comparison of Newton Iterations for the Exact Jacobian, Finite Difference Jacobian, and JFNK for Set-to-Zero Fixup with Rebalance and $\eta = 10^{-3}$

c	Jacobian Type	Newton Iterations	GMRES Iterations	Jacobian Multiplies	Transport Sweeps	$\|F\|$
0.05	Exact, JFNK	2	(1,1)	6	9	3.368e−08[a]
	Finite difference	2	(1,1)	6	903	3.368e−08
0.4	Exact, JFNK	3	(3,3,3)	15	19	2.954e−09
	Finite difference	3	(3,3,3)	15	1354	2.956e−09
0.7	Exact, JFNK	4	(4,5,4,4)	25	30	8.684e−08
	Finite difference	4	(4,5,4,4)	25	1805	8.685e−08
0.9	Exact, JFNK	5	(7,8,8,8,9)	50	56	4.737e−08
	Finite difference	5	(7,8,8,8,9)	50	2256	4.737e−08
0.99	Exact, JFNK	5	(12,14,17,19,23)	95	101	4.167e−07
	Finite difference	5	(12,14,17,19,23)	95	2256	4.164e−07

[a] Read as 3.368×10^{-8}.

Jacobian—a tightly clustered spectrum close to one indicates more rapid convergence than a spread-out spectrum with eigenvalues close to zero. The Jacobian is different for each Newton iterate, hence the difference in inner iteration count over the course of the Newton iteration. Table V shows the real part of the smallest eigenvalue of J at various iterates ϕ_z for the same scattering ratios as in Table IV. As can be seen, overall the minimum eigenvalue of the Jacobian decreases as the Newton iterate approaches the solution, and a corresponding increase can be seen in the number of inner Krylov iterations in Table IV. Although not shown, the maximum eigenvalue is always less than or equal to one. The spectrum of the Jacobian indicates that it is nonsingular and that GMRES should be able to invert it efficiently.

For each Newton iteration Table VI shows the L_2-norm of F ($\|F\|$) at the end of the iteration, the number of inner Krylov iterations, and the number of times F was applied over the course of those iterations, which is equivalent to the number of transport sweeps conducted, for $c = 0.7$. It also provides information on the number of negative terms encountered during the course of an iteration. For Newton iteration $(z+1)$, negative terms present in ϕ_z and $\phi_z + \epsilon v$ are counted as $F(\phi_z)$, and for each inner Krylov iteration, $F(\phi_z + \epsilon v)$ are computed, where v is the vector returned by the Krylov method. At the same time, the number of negative terms encountered (and fixed up if NFF is on) in Ψ over the course of the sweep [see Eq. (7a)] is also tabulated. These counts are given as a percentage of the number of cells with at least one negative value over the course of the Newton iteration and an average percentage of all unknowns that are negative. It is our experience that the fixups conducted as $F(\phi_z)$ and $F(\phi_z + \epsilon v)$ are computed is fairly consistent, particularly in the later Newton iterations, which is expected because ϵv should be small. Results are shown with and without NFF. As can be seen, there are negative elements of Ψ in all of the cells over the course of each

TABLE V

Real Part of the Minimum Eigenvalue of $\mathbf{J}(\phi_z)$

c	min($\lambda(\mathbf{J})$)					
	$\mathbf{J}(\phi_0)$	$\mathbf{J}(\phi_1)$	$\mathbf{J}(\phi_2)$	$\mathbf{J}(\phi_3)$	$\mathbf{J}(\phi_4)$	$\mathbf{J}(\phi_5)$
0.05	0.95159	0.95159	0.95159	—	—	—
0.4	0.61336	0.61424	0.61433	0.61432	—	—
0.7	0.32389	0.32639	0.32519	0.32520	0.32519	—
0.9	0.13080	0.12794	0.11962	0.11677	0.11622	0.11622
0.99	0.044028	0.030242	0.014695	0.013806	0.013806	0.013806

TABLE VI

Iteration Profile for JFNK Without and With NFF for $\eta = 10^{-3}$ and $c = 0.7$

Newton Iteration	$\|F\|$	Krylov Iterations	Applications of F	$\Psi < 0$ (%)		$\phi < 0$ (%)	
				Cells	Unknowns	Cells	Unknowns
Without NFF							
0	1.137e+01[a]		1				
1	2.829e−03	5	8	100.0	49.3	96.7	51.6
2	7.503e−07	5	8	100.0	46.4	98.7	47.1
With NFF							
0	9.409e+00		1				
1	5.504e−01	4	7	100.0	24.7	0.0	0.0
2	1.267e−02	5	8	100.0	24.1	21.3	6.6
3	9.141e−05	4	7	100.0	23.1	0.0	0.0
4	8.684e−08	4	7	100.0	23.1	0.0	0.0

[a] Read as 1.137×10^1.

iteration. When NFF is off, there are also negative elements of ϕ_z and $\phi_z + \epsilon v$ in the majority of the cells, but with NFF on, the percentage is much smaller and goes to zero as ϕ_z approaches the solution, which should be positive everywhere. When NFF is on, in the first Newton iteration ϕ_0 and $\phi_0 + \epsilon v$ are also positive everywhere because the current iterate ϕ_0 is the uncollided flux, which is the result of a single transport sweep with NFF. With NFF off, the percentage of negative angular fluxes is larger than that with NFF on. Also, we note that the average percentage of negative angular fluxes (hence, the number of fixups conducted when NFF is on) is similar to that observed in SI.

V. CONCLUSION

We applied the NK method to the spatially discretized S_N transport problem with NFF on triangular meshes.[10] The nonlinear nature of flux fixup schemes makes Newton's method a natural, though largely overlooked, choice of iterative solver. NK methods can be viewed as a means by which Krylov iterative methods, which have been shown to efficiently solve linear transport problems, can be applied to nonlinear transport applications. In the absence of negative fluxes, the two methods are equivalent provided the tolerance on the inner Krylov iteration is sufficiently small. The Jacobian can have a strong impact on the efficiency of the inner Krylov iteration and, by extension, on Newton's method. We have shown that the Jacobian is nonsingular and is well-suited for inversion using Krylov methods such as GMRES. Using the JFNK approach, in which only the evaluation of the nonlinear function is required and explicit knowledge of the form of the Jacobian is not, did not adversely affect the convergence of NK, making JFNK an efficient and robust solution method for S_N transport with NFF.

Further investigations will include extending the approach to anisotropic scattering and to multigroup k-eigenvalue calculations,[15] wherein the set-to-zero NFF is included directly in the nonlinear formulation of the k-eigenvalue problem. NFF in the transport operator makes any linear diffusion synthetic acceleration (DSA) method for SI acceleration inconsistent.[22] This inconsistency could cause convergence of SI to degrade or cause it to not converge at all. We plan to explore using JFNK with a linear DSA preconditioner applied to the S_N transport equation with NFF. The convergence of the Newton iteration should be insensitive to the resulting inconsistency of the DSA preconditioner. This approach can be compared to using nonlinear DSA to precondition the inner Krylov iterations, which has previously been used successfully in transport problems without NFF (Ref. 11).

ACKNOWLEDGMENTS

The authors would like to thank S. Hamilton of Emory University for the use of his transport code to produce numerical results and J. Morel of Texas A&M University for insightful discussions on the use of JFNK for transport applications. This information has been authored by an employee or employees of the Los Alamos National Security LLC (LANS), operator of the Los Alamos National Laboratory under contract DE-AC52-06NA25396 with the U.S. Department of Energy.

REFERENCES

1. K. D. LATHROP, "Spatial Differencing of the Transport Equation: Positivity vs. Accuracy," *J. Comput. Phys.*, **4**, 475 (1969).

2. E. LEWIS and W. MILLER, *Computational Methods of Neutron Transport*, American Nuclear Society, La Grange Park, Illinois (1993).

3. B. L. BIHARI and P. N. BROWN, "High Order Finite Volume Nonlinear Schemes for the Boltzmann Transport Equation," *Computational Methods in Transport*, pp. 401–422, T. J. BARTH, M. GRIEBEL, D. E. KEYES, R. M. NIEMINEN, D. ROOSE, and T. SCHLICK, Eds., Vol. 48 of *Lecture Notes in Computational Science and Engineering*, Springer, Berlin-Heidelberg, (2006).

4. R. P. SMEDLEY-STEVENSON, "Single-Ray Streaming Behavior for Discontinuous Finite Element Spatial Discretizations," *Nucl. Sci. Eng.*, **142**, 75 (2002).

5. S. R. MERTON, R. P. SMEDLEY-STEVENSON, C. C. PAIN, A. G. BUCHAN, and M. D. EATON, "A Non-Linear Optimal Discontinuous Petrov-Galerkin Method for Stabilizing the Solution of the Transport Equation," *Proc. Int. Conf. Mathematics, Computational Methods, and Reactor Physics*, Saratoga Springs, New York, May 3–7, 2009, American Nuclear Society (2009) (CD-ROM).

6. S. F. ASHBY, P. N. BROWN, M. R. DORR, and A. C. HINDMARSH, "Preconditioned Iterative Methods for Discretized Transport Equations," *Proc. Int. Topl. Mtg. Advances in Mathematics, Computations, Reactor Physics*, Pittsburgh, Pennsylvania, April 28–May 2, 1992, Vol. 2, pp. 6.1 2–1 (1991).

7. B. W. PATTON and J. P. HOLLOWAY, "Application of Preconditioned GMRES to the Numerical Solution of the Neutron Transport Equation," *Ann. Nucl. Energy*, **29**, 109 (2002).

8. B. GUTHRIE, J. P. HOLLOWAY, and B. W. PATTON, "GMRES as a Multi-Step Transport Sweep Accelerator," *Transp. Theory Stat. Phys.*, **28**, 83 (1999).

9. J. S. WARSA, T. A. WAREING, and J. E. MOREL, "Krylov Iterative Methods and the Degraded Effectiveness of Diffusion Synthetic Acceleration for Multidimensional S_N Calculations in Problems with Material Discontinuities," *Nucl. Sci. Eng.*, **147**, 218 (2004).

10. S. HAMILTON, M. BENZI, and J. WARSA, "Negative Flux Fixups in Discontinuous Finite Element SN Transport," *Proc. Int. Conf. Mathematics, Computational Methods, and Reactor Physics*, Saratoga Springs, New York, May 3–7, 2009, American Nuclear Society (2009) (CD-ROM).

11. D. A. KNOLL, H. PARK, and K. SMITH, "Application of the Jacobian-Free Newton-Krylov Method in Computational Reactor Physics," *Proc. Int. Conf. Mathematics, Computational Methods, and Reactor Physics*, Saratoga Springs, New York, May 3–7, 2009, American Nuclear Society (2009) (CD-ROM).

12. V. MAHADEVANA and J. RAGUSA, "Novel Hybrid Scheme to Compute Several Dominant Eigenmodes for Reactor Analysis Problems," *Proc. Int. Conf. Physics of Reactors* (PHYSOR'08) "Nuclear Power: A Sustainable Resource," Interlaken, Switzerland, September 14–19, 2008 (2008).

13. D. F. GILL and Y. Y. AZMY, "A Jacobian-Free Newton-Krylov Iterative Scheme for Criticality Calculations Based on the Neutron Diffusion Equation," *Proc. Int. Conf. Mathematics, Computational Methods, and Reactor Physics*, Saratoga Springs, New York, May 3–7, 2009, American Nuclear Society (2009) (CD-ROM).

14. D. F. GILL and Y. Y. AZMY, "Jacobian-Free Newton-Krylov as an Alternative to Power Iterations for the *k*-Eigenvalue Transport Problem," *Trans Am. Nucl. Soc.,* **100**, 291 (2009).

15. D. F. GILL, Y. Y. AZMY, J. D. DENSMORE, and J. S. WARSA, "Newton's Method for the Computation of Eigenvalues in Transport Applications," *Nucl. Sci. Eng.* (submitted for publication).

16. J. E. DENNIS and R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM, Philadelphia (1996).

17. D. A. KNOLL and D. R. KEYES, "Jacobian-Free Newton-Krylov Methods: A Survey of Approaches and Applications," *J. Comput. Phys.*, **193**, 357 (2004).

18. O. C. ZIENKIEWICZ and R. L. TAYLOR, *The Finite Element Method*, Vol. 1, 5th ed., Butterworth-Heinemann, Oxford (2000).

19. C. KELLEY, *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philadelphia (1995).

20. T. F. CHAN and K. R. JACKSON, "Nonlinearly Preconditioned Krylov Subspace Methods for Discrete Newton Algorithms," *SIAM J. Sci. Stat. Comput.,* **5**, 533 (1984).

21. P. N. BROWN and Y. SAAD, "Hybrid Krylov Methods for Nonlinear Systems of Equations," *SIAM J. Sci. Stat. Comput.*, **11**, 450 (1990).

22. R. E. ALCOUFFE, "Diffusion Synthetic Acceleration Methods for the Diamond-Differenced Discrete-Ordinates Equations," *Nucl. Sci. Eng.*, **64**, 344 (1977).