Nuclear Science and Engineering

ISSN: 0029-5639 (Print) 1943-748X (Online) Journal homepage: https://www.tandfonline.com/loi/unse20

Jacobian-Free Newton-Krylov Nodal Expansion Methods with Physics-Based Preconditioner and Local Elimination for Three-Dimensional and Multigroup *k*-Eigenvalue Problems

Xiafeng Zhou & Fu Li

To cite this article: Xiafeng Zhou & Fu Li (2018) Jacobian-Free Newton-Krylov Nodal Expansion Methods with Physics-Based Preconditioner and Local Elimination for Three-Dimensional and Multigroup *k*-Eigenvalue Problems, Nuclear Science and Engineering, 190:3, 238-257, DOI: 10.1080/00295639.2018.1435136

To link to this article: https://doi.org/10.1080/00295639.2018.1435136

NUCLEAR SCIENCE AND ENGINEERING · VOLUME 190 · 238–257 · JUNE 2018

DOI: https://doi.org/10.1080/00295639.2018.1435136

Jacobian-Free Newton-Krylov Nodal Expansion Methods with Physics-Based Preconditioner and Local Elimination for Three-Dimensional and Multigroup *k*-Eigenvalue Problems

Xiafeng Zhou[a]* and Fu Li[b]

[a]*Huazhong University of Science and Technology, School of Energy and Power Engineering, Department of Nuclear Engineering and Technology, Wuhan 430074, China*
[b]*Tsinghua University, Institute of Nuclear and New Energy Technology, Beijing 100084, China*

Received November 15, 2017
Accepted for Publication January 29, 2018

Abstract — *Motivated by the high accuracy and efficiency of nodal methods on the coarse meshes and the superlinear convergence and high efficiency of Jacobian-free Newton-Krylov (JFNK) methods for large-scale nonlinear problems, a new JFNK nodal expansion method (NEM) with the physics-based preconditioner and local elimination NEM_JFNK is successfully developed to solve three-dimensional (3D) and multigroup k-eigenvalue problems by combining and integrating the NEM discrete systems into the framework of JFNK methods. A local elimination technique of NEM_JFNK is developed to eliminate some intermediate variables, expansion coefficients, and transverse leakage terms through equivalent transformation as much as possible in order to reduce the computational cost and the number of final-solving variables and residual equations constructed in NEM_JFNK. Then efficient physics-based preconditioners are successfully developed by approximating the matrices of the diffusion and removal terms, transverse leakage terms using the three-adjacent-node quadratic fitting methods, and scatter source terms, which make full use of the traditional power iteration. In addition, the Eisenstat-Walker forcing terms are used in the developed NEM_JFNK method to adaptively choose the convergence criterion of linear Krylov iteration within each Newton iteration based on the Newton residuals and to improve computational efficiency further. Finally, the NEM_JFNK code is developed for 3D and multigroup k-eigenvalue problems in neutron diffusion calculations and the detailed study of convergence, computational cost, and efficiency is carried out for several 3D problems. Numerical results show that the developed NEM_JFNK methods have faster convergence speed and are more efficient than the traditional NEM using power iteration, and the speedup ratio is greater for the higher convergence criterion.*

Keywords — *Jacobian-free Newton-Krylov, nodal expansion method, physics-based preconditioner.*

Note — *Some figures may be in color only in the electronic version.*

I. INTRODUCTION

In recent years, the Jacobian-free Newton-Krylov (JFNK) method[1] has been gradually developed to solve neutron transport or diffusion problems,[2–5] thermal-hydraulic models,[6–8] and multiphysics coupled problems[9–11] in nuclear reactor simulations due to its superlinear convergence, high efficiency, and Jacobian-free technique that does not require one to explicitly construct and store the Jacobian matrix, instead only needing to evaluate the system residuals. These advantages of JFNK are especially attractive to solve large-scale complicated coupled nonlinear systems because ensuring the convergence and forming the Jacobian matrix are very difficult and time-consuming tasks.

The transverse-integrated coarse-mesh nodal methods can yield higher numerical accuracy and efficiency compared with the finite volume method and can greatly reduce the discrete mesh number, computer storage, and

*E-mail: zhouxiafeng@hust.edu.cn

computational cost for the large-scale problems in the desired precision.[12,13] Therefore, when adopting the JFNK method to simulate the large-scale complicated multiphysics coupled problems in nuclear reactor analysis, we still hope to choose the efficient coarse-mesh nodal methods to discretize the neutron transport or diffusion equations on the coarse mesh to improve the computational accuracy and efficiency as much as possible.

To achieve the above-mentioned aim and make full use of the respective advantages of the two methods, we consider combining nodal methods with the JFNK method to improve further the efficiency of reactor physics analysis. However, most research on JFNK is mainly based on finite volume[2,5] or finite element discrete schemes,[14–16] and JFNK based on nodal methods is rarely involved and studied in public literature; thus, some key critical techniques and special treatments need to be explored.

Therefore, motivated by the high efficiency and accuracy of nodal methods on the coarse meshes and the superlinear convergence and high efficiency of JFNK methods for large-scale complicated coupled nonlinear problems, in this work nodal expansion method (NEM) discrete systems are integrated into the general framework of the JFNK method. Then a new Jacobian-free Newton-Krylov nodal expansion method (NEM_JFNK) is successfully developed to solve three-dimensional (3D) and multigroup k-eigenvalue problems in neutron diffusion calculation. To improve the computational accuracy and efficiency as much as possible, a local elimination technique of NEM_JFNK is developed to eliminate some variables through equivalent transformation in order to reduce the number of final-solving variables and the computational cost of residual equations. Then efficient physics-based preconditioners are also successfully developed by approximating the Jacobian matrix of the diffusion and removal terms, transverse leakage terms, and scatter source terms, which make full use of the traditional power iteration. In addition, the developed NEM_JFNK methods employ the Eisenstat-Walker forcing terms[17] to adaptively choose the convergence criterion of the linear Krylov iteration within each Newton iteration based on the Newton residuals. The paper focuses on the theory and analysis about how to combine NEM and the JFNK method and the core ideas of the above-mentioned key methods.

The paper is organized as follows. Section II presents the NEM_JFNK formulation and some special treatments including computational framework, local elimination techniques, the construction of the physics-based preconditioner, and the flowchart and computation cost analysis in detail. Section III analyzes and discusses the numerical convergence properties and computational cost of NEM_JFNK and its comparison with the traditional and effective NEM using power iteration in terms of 3D and multigroup k-eigenvalue problems. Finally, a brief summary is given in Sec. IV.

II. NEM_JFNK FORMULATION AND SPECIAL TREATMENTS

The 3D and multigroup neutron diffusion equations in Cartesian coordinate[18] can be given by

$$-\frac{\partial}{\partial x}\left(D_g\frac{\partial\phi_g}{\partial x}\right)-\frac{\partial}{\partial y}\left(D_g\frac{\partial\phi_g}{\partial y}\right)-\frac{\partial}{\partial z}\left(D_g\frac{\partial\phi_g}{\partial z}\right)+\Sigma_{R,g}\phi_g=Q_g=\sum_{\substack{g'=1\\g'\neq g}}^{G}\Sigma_{s,\,g'\to g}\phi_{g'}+\frac{\chi_g}{k_{eff}}\sum_{g'=1}^{G}\upsilon\Sigma_{f,\,g'}\phi_{g'}\,, \tag{1}$$

where

ϕ_g = scalar flux

D_g = diffusion coefficient

$\Sigma_{R,g}$ = removal cross section

$\Sigma_{s,\,g'\to g}$ = scattering cross section from group g' to g

χ_g = fission spectrum

k_{eff} = effective multiplication factor

$\upsilon\Sigma_{f,\,g'}$ = average neutron number times fission cross section

G = total number of energy groups.

NEM is first adopted to discretize the neutron diffusion equations. Specifically, the transverse integration process is applied to convert a multidimensional partial differential equation into multiple one-dimensional (1D) transverse-integrated equations in form and these 1D equations are coupled with each other by the transverse leakage terms and then use a three-adjacent-node quadratic fitting to construct the shape of transverse leakage terms from transverse directions. The computing domain is divided into $I \times J \times K$ nodes and the width of each node is $\left[-h_x^{i,j,k}, h_x^{i,j,k}\right] \times \left[-h_y^{i,j,k}, h_y^{i,j,k}\right] \times \left[-h_z^{i,j,k}, h_z^{i,j,k}\right]$ with $i = 1, \ldots, I$, $j = 1, \ldots, J$, and $k = 1, \ldots, K$. Figure 1 shows the schematic of discretization and the detailed formulation and some special treatments of NEM_JFNK are presented in Secs. II.A through II.D.

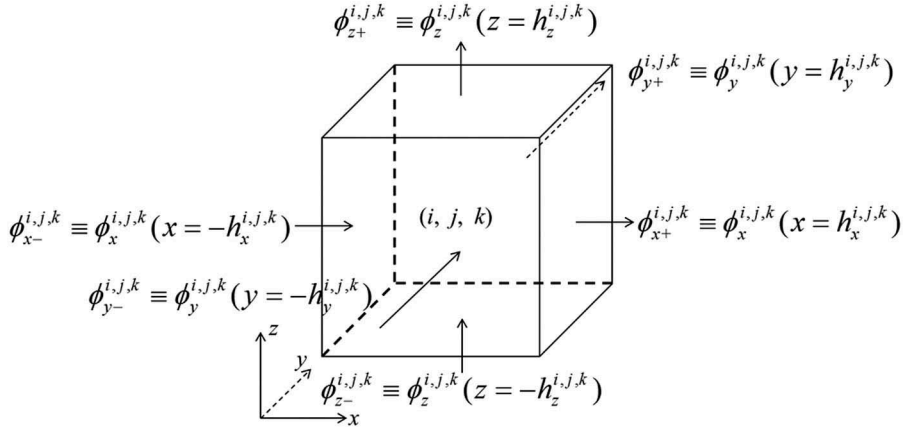


Fig. 1. Schematic of discretization for NEM.

II.A. NEM_JFNK Basic Framework

First, the nonlinear residual functions need to be obtained from the discrete systems of NEM. By applying the transverse integration strategy for Eq. (1) over the node (*i, j, k*), the transverse-integrated equations in the different coordinate directions are yielded by

$$\frac{dJ_{gr}(r)}{dr} + \bar{\Sigma}_{R,g} \cdot \phi_{gr}(r) = S_{gr}(r) = Q_{gr}(r) - L_{gr}(r) \ , \tag{2}$$

$$\phi_{gr}(r) = \int_{-h_\xi}^{h_\xi} \int_{h_\eta}^{h_\eta} \phi_g(r, \xi, \eta) d\xi d\eta \ , \tag{3}$$

$$J_{gr}(r) = -\bar{D}_g \frac{d\phi_{gr}(r)}{dr} \ , \tag{4}$$

$$Q_{gr}(r) = \frac{1}{4h_\xi h_\eta} \int_{-h_\xi}^{h_\xi} \int_{h_\eta}^{h_\eta} Q_g(r, \xi, \eta) d\xi d\eta \ , \tag{5}$$

and

$$L_{gr}(r) = \frac{1}{4h_\xi h_\eta} \int_{-h_\xi}^{h_\xi} \int_{h_\eta}^{h_\eta} -\frac{\partial}{\partial y}\left(D_g \frac{\partial \phi_g}{\partial y}\right) - \frac{\partial}{\partial z}\left(D_g \frac{\partial \phi_g}{\partial z}\right) d\xi d\eta \ , \tag{6}$$

where

$$r = x, y, z \neq \xi \neq \eta$$
$$\xi = y, z, x$$
$$\eta = z, x, y \ .$$

The indices (*i, j, k*) are omitted for simplicity of notation and the overbar notations indicate the average values in the node (*i, j, k*). $\phi_{gr}(r)$ is the r-dependent transverse-integrated variable and the pseudo-source terms $S_{gr}(r)$ are divided into two terms: transverse-integrated true source term $Q_{gr}(r)$ and the transverse leakage term $L_{gr}(r)$. To solve Eq. (2), $Q_{gr}(r)$, $L_{gr}(r)$, and $\phi_{gr}(r)$ are approximated by an expansion of Legendre polynomials $f_{r,n}(r)$:

$$\phi_{gr}(r) = \sum_{n=0}^{N_\phi=4} a_{gr,n} \cdot f_{r,n}(r) \ , \tag{7}$$

$$Q_{gr}(r) = \sum_{n=0}^{N_Q=2} q_{gr,n} \cdot f_{r,n}(r) \ , \tag{8}$$

and

$$L_{gr}(r) = \sum_{n=0}^{N_L=2} l_{gr,n} \cdot f_{r,n}(r) \ , \tag{9}$$

where $a_{gr,n}$, $q_{gr,n}$, and $l_{gr,n}$ are the expansion coefficients of $\phi_{gr}(r)$, $Q_{gr}(r)$, and $L_{gr}(r)$, respectively. $a_{gr,n}$ can be determined by the node-average variables $\bar{\phi}_g$, the currents J_{gr-}, J_{gr+}, and $N_\phi - 2$ the weighted residual equations:

$$\bar{\phi}_g = \int_{h_r}^{h_r} \phi_{gr}(r) dr \ , \tag{10}$$

$$J_{gr-} \equiv J_{gr}(r = -h_r) \ , \tag{11}$$

$$J_{gr+} \equiv J_{gr}(r = h_r) \ , \tag{12}$$

and

$$\int_{h_r}^{h_r} f_{r,n}(r) \cdot Eq.(2) dr \ \left(n = 1, \ldots, N_\phi - 2\right). \tag{13}$$

After that, by imposing the continuity conditions of $J_{gr\pm}$ and $\phi_{gr\pm}$ at the interface between adjacent nodes, the discrete equations in terms of $J_{gr\pm}$ can be obtained for $r = x$:

$$\begin{aligned} &Aw \cdot J_{gx+}^{i-1,j,k} + Ap \cdot J_{gx+}^{i,j,k} + Ap \cdot J_{gx+}^{i+1,j,k} \\ &\quad = \bar{\phi}_g^{i,j,k} - \bar{\phi}_g^{i+1,j,k} + Bp \cdot \left(q_{gx,1}^{i,j,k} - l_{gx,1}^{i,j,k}\right) \\ &\qquad + Be \cdot \left(q_{gx,1}^{i+1,j,k} - l_{gx,1}^{i+1,j,k}\right) + Cp \\ &\qquad (q_{(gx,2)}^{(i,j,k)} - l_{(gx,2)}^{(i,j,k)}) + Ce \cdot (q_{(gx,2)}^{(i+1,j,k)} - l_{(gx,2)}^{(i+1,j,k)}), \end{aligned} \tag{14}$$

The detailed expressions of the coefficients in Eq. (14) are not presented for simplicity. Analogously, the discrete equations in the y and z coordinate direction can be also obtained in a similar way and the node-average variables $\bar{\phi}_g$ can be computed by nodal balance equation:

$$\sum_{r=x,y,x} \frac{J_{gr+}^{i,j,k} - J_{gr-}^{i,j,k}}{2h_r^{i,j,k}} + \bar{\Sigma}_{R,g} \cdot \bar{\phi}_g^{i,j,k} = \bar{Q}_g^{i,j,k}. \tag{15}$$

The coefficients $q_{gr,n}$ and $\bar{Q}_g^{i,j,k}$ can be expressed as

$$\bar{Q}_g = \sum_{\substack{g'=1 \\ g' \neq g}}^{G} \bar{\Sigma}_{s,g' \to g} \bar{\phi}_{g'} + \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} \overline{\upsilon\Sigma}_{f,g'} \bar{\phi}_{g'}, \tag{16}$$

$$\begin{aligned} q_{gr,1} = &\sum_{\substack{g'=1 \\ g' \neq g}}^{G} \bar{\Sigma}_{s,g' \to g} \left(a_{gr,1} - \frac{a_{gr,3}}{10}\right) \\ &+ \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} \overline{\upsilon\Sigma}_{f,g'} \left(a_{gr,1} - \frac{a_{gr,3}}{10}\right), \end{aligned} \tag{17}$$

and

$$\begin{aligned} q_{gr,2} = &\sum_{\substack{g'=1 \\ g' \neq g}}^{G} \bar{\Sigma}_{s,g' \to g} \left(a_{gr,2} - \frac{a_{gr,4}}{35}\right) \\ &+ \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} \overline{\upsilon\Sigma}_{f,g'} \left(a_{gr,2} - \frac{a_{gr,4}}{35}\right). \end{aligned} \tag{18}$$

The standard treatment of transverse leakage terms is adopted to construct the shape of transverse leakage terms from transverse directions by a three-adjacent-node quadratic fitting. For $r = x$:

$$l_{gr,0}^{i,j,k} = \frac{J_{g\xi+}^{i,j,k} - J_{g\xi-}^{i,j,k}}{2h_\xi^{i,j,k}} + \frac{J_{g\eta+}^{i,j,k} - J_{g\eta-}^{i,j,k}}{2h_\eta^{i,j,k}}, \tag{19}$$

$$l_{gr,1} = lw1 \cdot l_{gr,0}^{i-1,j,k} + lp1 \cdot l_{gr,0}^{i,j,k} + le1 \cdot l_{gr,0}^{i+1,j,k}, \tag{20}$$

and

$$l_{gr,2} = lw2 \cdot l_{gr,0}^{i-1,j,k} + lp2 \cdot l_{gr,0}^{i,j,k} + le2 \cdot l_{gr,0}^{i+1,j,k}, \tag{21}$$

where $lw1$, $lp1$, $le1$, $lw2$, $lp2$, and $le2$ can be expressed as the functions of the width of three adjacent nodes. These coefficients $l_{gr,n}$ in the y and z coordinate direction are also treated in a similar way and the power iteration is adopted to compute the effective multiplication factor k_{eff}. After that, the discrete systems of NEM can be rewritten as a general formulation:

$$(\mathbf{M} - \mathbf{N} - \mathbf{L}) \cdot \vec{\Phi} = \frac{1}{k_{eff}} \cdot \mathbf{F} \cdot \vec{\Phi}, \tag{22}$$

$$k_{eff}^{l} = k_{eff}^{l-1} \cdot \frac{\sum_{g'=1}^{G} \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} \overline{\upsilon\Sigma}_{f,g'} \left(\bar{\phi}_{g'}^{i,j,k}\right)^{l}}{\sum_{g'=1}^{G} \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} \overline{\upsilon\Sigma}_{f,g'} \left(\bar{\phi}_{g'}^{i,j,k}\right)^{l-1}}, \tag{23}$$

and

$$\vec{\Phi} = \left[\vec{\Phi}_1;\ \vec{\Phi}_2;\ \vec{\Phi}_3;\ \ldots;\ \vec{\Phi}_G\right], \tag{24}$$

where

$$\vec{\Phi}_g = \left[\bar{\phi}_g,\ J_{gr+},\ \phi_{gr+},\ a_{gr,n},\ q_{gr,n},\ l_{gr,n}\right]^T$$

$r = x,\ y,\ z$

$\mathbf{M}$ = square matrix representing discretization of diffusion and removal terms in the r-dependent direction

$\mathbf{N}$ = square matrix representing scatter terms from group g' to g

$\mathbf{L}$ = square matrix representing transverse leakage terms

$\mathbf{F}$ = square matrix representing fission terms

l = step of power iteration.

To solve the discrete systems of NEM using JFNK simultaneously, it is straightforward to compute the nonlinear residual functions for all the parameters and variables $\vec{X} = \left[\vec{\Phi}; k_{eff}\right]$ with $g = 1, 2, \ldots, G$ by subtracting the terms on the right side of Eq. (22) from the left side, that is

$$\vec{R}(\vec{X}) = \begin{bmatrix} (\mathbf{M} - \mathbf{N} - \mathbf{L}) \cdot \vec{\Phi} - \dfrac{1}{k_{eff}} \cdot \mathbf{F} \cdot \vec{\Phi} \\ k_{eff}^{l} - k_{eff}^{l-1} \cdot \dfrac{\sum_{g'=1}^{G} \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} \overline{\upsilon\Sigma}_{f,g'} \left(\bar{\phi}_{g'}^{i,j,k}\right)^{l}}{\sum_{g'=1}^{G} \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} \overline{\upsilon\Sigma}_{f,g'} \left(\bar{\phi}_{g'}^{i,j,k}\right)^{l-1}} \end{bmatrix} . \tag{25}$$

The Newton iteration is used for the nonlinear k-eigenvalue problems, which are defined as

$$\mathbf{Ja}\left(\vec{X}^{m}\right) \cdot \delta\vec{X}^{m} = -\vec{R}\left(\vec{X}^{m}\right) \tag{26}$$

and

$$\vec{X}^{m+1} = \vec{X}^{m} + \delta\vec{X}^{m} , \tag{27}$$

where $\mathbf{Ja}\left(\vec{X}^{m}\right)$ is the Jacobian matrix of the m'th Newton step. At each step of Newton iteration, the linear equation [Eq. (26)] is solved using a Krylov method [for example, the Generalized Minimum Residual Method (GMRES) and Biconjugate Gradient Stabilized Method (BICGSTAB)] and then the solutions are updated by Eq. (27). Because Krylov methods need only the Jacobian matrix-vector product and do not require one to explicitly construct the Jacobian matrix, the Jacobian matrix-vector product can be approximated by a finite difference of the system residuals $\vec{R}(\vec{X})$ for the JFNK methods:

$$\mathbf{Ja}\left(\vec{X}^{m}\right) \cdot \vec{\mathrm{v}} = \frac{\vec{R}(\vec{X}^{m} + \varepsilon\vec{\mathrm{v}}) - \vec{R}\left(\vec{X}^{m}\right)}{\varepsilon} , \tag{28}$$

where $\vec{\mathrm{v}}$ is a basis vector of Krylov iteration and ε is the perturbation parameter. Many choices of ε are studied in the references.[19] In this work the following expression for ε is adopted[20]:

$$\varepsilon = \frac{\sqrt{\varepsilon_{mach}}}{\vec{\mathrm{v}}_2} \max\left\{ \left(\vec{X}^{m}\right)^{T} \cdot \vec{\mathrm{v}},\ 1.0 \right\} \cdot \mathrm{sign}\left[\left(\vec{X}^{m}\right)^{T} \cdot \vec{\mathrm{v}} \right] , \tag{29}$$

where ε_{mach} is the machine precision. In addition, the convergence criterion of the linear Krylov iteration within each Newton iteration has a great influence on the convergence rate of the Newton iteration and total computational cost.[17] Therefore, an adaptive convergence criterion of linear Krylov iterations based on the Newton residuals is used in the developed NEM_JFNK method to improve convergent rate and computational efficiency by adopting the Eisenstat-Walker forcing term[17] that is expressed as

$$\| \vec{R}(\vec{X}^{m}) + \mathbf{J}a(\vec{X}^{m}) \cdot \delta\vec{X}^{m} \| \leq \eta_m \| \vec{R}(\vec{X}^{m}) \| , \tag{30}$$

$$\eta_m = \gamma \left(\frac{\| \vec{R}(\vec{X}^{m}) \|}{\| \vec{R}(\vec{X}^{m-1}) \|} \right)^{\alpha} , \tag{31}$$

$$\eta_m = \max\left\{ \eta_m,\ \gamma\eta_{m-1}^{\alpha} \right\} \text{ whenever } \gamma\eta_{m-1}^{\alpha} > 0.1 , \tag{32}$$

and

$$\eta_m = \min\{ \eta_m,\ \eta_{\max} \} , \tag{33}$$

where

$$\gamma = 0.9$$
$$\alpha = 2.0$$
$$\eta_{\max} = 0.9.$$

Based on the derivation in Sec. II.A, the basic framework of NEM_JFNK can be summarized in Table I.

II.B. Local Elimination

To make the computational efficiency of NEM_JFNK as high as possible, we hope there are as few final-solving variables $\vec{X}$ and nonlinear residual functions $\vec{R}(\vec{X})$ constructed in NEM_JFNK as possible. Therefore, through the equivalent transformation approach in this work, a local elimination technique is developed to eliminate some intermediate variables and expansion coefficients of transverse-integrated variables $\phi_{gr}(r)$, true source terms $Q_{gr}(r)$, and transverse leakage terms $L_{gr}(r)$ as much as possible in order to reduce further the size of nonlinear equations and computational cost.

To explain the key idea more clearly, we assume that the variables $\vec{X}$ can be expressed as $\vec{X} = \left[\vec{X}_1; \vec{X}_2\right]$ and $\vec{X}_2$ is a function of $\vec{X}_1$ $\left[\text{that is, } \vec{X}_2 = \vec{G}(\vec{X}_1)\right]$. Based on this

TABLE I

Basic Framework of the NEM_JFNK Method

Basic framework of NEM_JFNK method

$\vec{X}^0$ = initial guess
for m = 0, 1, . . . , do
Compute the nonlinear residual functions $\vec{R}(\vec{X}^m)$ based on the discrete systems from NEM
Solve $\mathbf{Ja}(\vec{X}^m)\cdot\delta\vec{X}^m = -\vec{R}(\vec{X}^m)$ using Krylov iteration and Eisenstat-Walker forcing terms where
$\|\vec{R}(\vec{X}^m) + \mathbf{Ja}(\vec{X}^m)\cdot\delta\vec{X}^m\| \le \eta_m \|\vec{R}(\vec{X}^m)\|$ with Eqs. (31), (32), and (33)
and $\mathbf{Ja}(\vec{X}^m)\cdot\vec{v} = \dfrac{\vec{R}(\vec{X}^m+\varepsilon\vec{v}) - \vec{R}(\vec{X}^m)}{\varepsilon}$ with Eq. (29)
$\vec{X}^{m+1} = \vec{X}^m + \delta\vec{X}^m$
end for

assumption, $\vec{X}_2$ can be eliminated through the equation $\vec{X}_2 = \vec{G}\left(\vec{X}_1\right)$ and the nonlinear residual function $\vec{R}(\vec{X})$ can be simplified as $\vec{R}_{\vec{X}_1}\left(\vec{X}_1\right)$ by the following derivation:

$$\vec{R}(\vec{X}) = \begin{bmatrix} \vec{R}_{\vec{X}_1}\left(\vec{X}_1;\vec{X}_2\right) \\ \vec{X}_2 - \vec{G}\left(\vec{X}_1\right) \end{bmatrix} = \begin{bmatrix} \vec{R}_{\vec{X}_1}\left(\vec{X}_1;\vec{G}\left(\vec{X}_1\right)\right) \\ 0 \end{bmatrix} \Rightarrow \vec{R}_{\vec{X}_1}\left(\vec{X}_1\right), \tag{34}$$

where $\vec{R}_{\vec{X}_1}\left(\vec{X}_1\right)$ means that only the nonlinear residual functions of $\vec{X}_1$ are constructed based on the discrete systems and the final-solving variables only include $\vec{X}_1$ so that the computational size of the nonlinear problems is reduced. The detailed treatments of local elimination techniques for NEM_JFNK methods are presented below.

It is observed from Eqs. (10) through (13) that $a_{r,n}$ can be determined by the node-average variables $\bar{\phi}_g$ and the currents $J_{gr\pm}$, $q_{gr,n}$, and $l_{gr,n}$:

$$\begin{bmatrix} a_{gr,0} & a_{gr,1} & a_{gr,2} & a_{gr,3} & a_{gr,4} \end{bmatrix}^T = \mathbf{A}\cdot\begin{bmatrix} \bar{\phi}_g & J_{gr+} & J_{gr-} & q_{gr,1}-l_{gr,1} & q_{gr,2}-l_{gr,2} \end{bmatrix}^T \tag{35}$$

and

$$\mathbf{A} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & \dfrac{-\left(60De_{gr}+\Gamma_{gr}\right)}{12De_{gr}\left(10De_{gr}+\Gamma_{gr}\right)} & \dfrac{-\left(60De_{gr}+\Gamma_{gr}\right)}{12De_{gr}\left(10De_{gr}+\Gamma_{gr}\right)} & \dfrac{5h_r}{3\left(10De_{gr}+\Gamma_{gr}\right)} & 0.0 \\ 0.0 & \dfrac{-\left(140De_{gr}+\Gamma_{gr}\right)}{20De_{gr}\left(42De_{gr}+\Gamma_{gr}\right)} & \dfrac{\left(140De_{gr}+\Gamma_{gr}\right)}{20De_{gr}\left(42De_{gr}+\Gamma_{gr}\right)} & 0.0 & \dfrac{7h_r}{5\left(42De_{gr}+\Gamma_{gr}\right)} \\ 0.0 & \dfrac{-5\Gamma_{gr}}{6De_{gr}\left(10De_{gr}+\Gamma_{gr}\right)} & \dfrac{-5\Gamma_{gr}}{6De_{gr}\left(10De_{gr}+\Gamma_{gr}\right)} & \dfrac{-10h_r}{3\left(10De_{gr}+\Gamma_{gr}\right)} & 0.0 \\ 0.0 & \dfrac{-7\Gamma_{gr}}{4De_{gr}\left(42De_{gr}+\Gamma_{gr}\right)} & \dfrac{7\Gamma_{gr}}{4De_{gr}\left(42De_{gr}+\Gamma_{gr}\right)} & 0.0 & \dfrac{-21h_r}{42De_{gr}+\Gamma_{gr}} \end{bmatrix}, \tag{36}$$

where $De_{gr} = \bar{D}_g/2/h_r$ and $De_{gr} = 2h_r\bar{\Sigma}_{R,g}$. By combining Eq. (17) and Eqs. (35) and (36), $a_{gr,1-3} = a_{gr,1} - a_{gr,3}/10$ and $a_{gr,2-4} = a_{gr,2} - a_{gr,4}/35$ can be given by, respectively

$$a_{gr,1-3} = \frac{2h_r}{10De_{gr} + \Gamma_{gr}} \left(\sum_{\substack{g'=1 \\ g' \neq g}}^{G} \bar{\Sigma}_{s,g' \to g} a_{g'r,1-3} + \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} \overline{\upsilon\Sigma}_{f,g'} a_{g'r,1-3} - l_{gr,1} \right) - \frac{5}{10De_{gr} + \Gamma_{gr}} \left(J_{gr+} + J_{gr-} \right) \tag{37}$$

and

$$a_{gr,2-4} = \frac{2h_r}{42De_{gr} + \Gamma_{gr}} \left(\sum_{\substack{g'=1 \\ g' \neq g}}^{G} \bar{\Sigma}_{s,g' \to g} a_{g'r,2-4} + \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} \overline{\upsilon\Sigma}_{f,g'} a_{g'r,2-4} - l_{gr,2} \right) - \frac{7}{42 + \Gamma_{gr}} \left(J_{gr+} - J_{gr-} \right). \tag{38}$$

Equations (37) and (38) can be rewritten as a general formulation:

$$\mathbf{A13} \cdot \begin{bmatrix} a_{1r,1-3} \\ a_{2r,1-3} \\ \vdots \\ a_{Gr,1-3} \end{bmatrix} = - \begin{bmatrix} \dfrac{2h_r l_{1r,1} + 5(J_{1r+} + J_{1r-})}{10De_{1r} + \Gamma_{1r}} \\ \dfrac{2h_r l_{2r,1} + 5(J_{2r+} + J_{2r-})}{10De_{2r} + \Gamma_{2r}} \\ \vdots \\ \dfrac{2h_r l_{Gr,1} + 5(J_{Gr+} + J_{Gr-})}{10De_{Gr} + \Gamma_{Gr}} \end{bmatrix} \tag{39}$$

and

$$\mathbf{A24} \cdot \begin{bmatrix} a_{1r,2-4} \\ a_{2r,2-4} \\ \vdots \\ a_{Gr,2-4} \end{bmatrix} = - \begin{bmatrix} \dfrac{2h_r l_{1r,2} + 7(J_{1r+} - J_{1r-})}{42De_{1r} + \Gamma_{1r}} \\ \dfrac{2h_r l_{2r,2} + 7(J_{2r+} - J_{2r-})}{42De_{2r} + \Gamma_{2r}} \\ \vdots \\ \dfrac{2h_r l_{Gr,2} + 7(J_{Gr+} - J_{Gr-})}{42De_{Gr} + \Gamma_{Gr}} \end{bmatrix}, \tag{40}$$

where **A13** and **A24** are $G \times G$ matrices. $l_{gr,n}$ can be also expressed in terms of $J_{gr\pm}$ based on Eqs. (19), (20), and (21). Analogously, a general formulation for $\bar{\phi}_g$ can be rewritten by combining Eqs. (15) and (16):

$$\mathbf{A0} \cdot \begin{bmatrix} \bar{\phi}_1 \\ \bar{\phi}_2 \\ \vdots \\ \bar{\phi}_G \end{bmatrix} = - \begin{bmatrix} \sum\limits_{r=x,y,x} \dfrac{J_{1r+} - J_{1r-}}{2h_r} \\ \sum\limits_{r=x,y,x} \dfrac{J_{2r+} - J_{2r-}}{2h_r} \\ \vdots \\ \sum\limits_{r=x,y,x} \dfrac{J_{Gr+} - J_{Gr-}}{2h_r} \end{bmatrix}, \tag{41}$$

where **A0** is also a $G \times G$ matrix. Based on Eqs. (39), (40), and (41), $\bar{\phi}_g$, $a_{gr,1-3}$, and $a_{gr,2-4}$ can determined by computing the inverse matrices of **A0**, **A13**, and **A24**. After that, all the variables $\bar{\phi}_g$, ϕ_{gr+}, $a_{gr,1-3}$, $a_{gr,2-4}$, $l_{gr,n}$, $q_{gr,n}$, and $a_{gr,n}$ can be expressed as functions of J_{gr+}. So J_{gr+} and k_{eff} can be chosen only as the final-solving variables and the nonlinear residual functions of J_{gr+} and k_{eff} are only constructed from Eqs. (14) and (23).

However, the computational cost of the inverse matrixes of **A0**, **A13**, and **A24** increases with the total number of groups G. To make NEM_JFNK applicable for various values of G, we finally decide to choose J_{gr+}, k_{eff}, $\bar{\phi}_g$, $a_{gr,1-3}$, and $a_{gr,2-4}$ as the final-solving variables for the developed NEM_JFNK method and all the other variables can be eliminated through Eqs. (16), (17), (18), (19), (20), (21), and (35). The corresponding nonlinear residual functions of the final-solving variables can be determined from Eqs. (14), (15), (23), (32), and (33), that is

$$\vec{R}\left(\vec{\tilde{X}}\right) = \begin{bmatrix} \left(\tilde{\mathbf{M}} - \tilde{\mathbf{N}} - \tilde{\mathbf{L}}\right) \cdot \vec{\tilde{\Phi}} - \frac{1}{k_{eff}} \cdot \tilde{\mathbf{F}} \cdot \vec{\tilde{\Phi}} \\ k_{eff}^{l} - k_{eff}^{l-1} \cdot \dfrac{\sum_{g'=1}^{G}\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K} \overline{\upsilon\Sigma}_{f,g'} \left(\bar{\phi}_{g'}^{i,j,k}\right)^{l}}{\sum_{g'=1}^{G}\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K} \overline{\upsilon\Sigma}_{f,g'} \left(\bar{\phi}_{g'}^{i,j,k}\right)^{l-1}} \end{bmatrix}, \tag{42}$$

where

$$\vec{\tilde{\Phi}} = \left[\vec{\tilde{\Phi}}_1; \vec{\tilde{\Phi}}_2; \vec{\tilde{\Phi}}_3; \cdots; \vec{\tilde{\Phi}}_G\right]$$

$$\vec{\tilde{X}} = \left[\vec{\tilde{\Phi}}; k_{eff}\right]$$

$$\vec{\tilde{\Phi}}_g = \begin{bmatrix} J_{gr+} & \bar{\phi}_g & a_{gr,1-3} & a_{gr,2-4} \end{bmatrix}^T$$

$$\text{the tilde} = \text{discretization corresponding to } \vec{\tilde{X}},$$

and where the square matrices $\tilde{\mathbf{M}}$, $\tilde{\mathbf{N}}$, $\tilde{\mathbf{L}}$, and $\tilde{\mathbf{F}}$ have similar meanings to those in Eq. (22).

II.C. Physics-Based Preconditioner

Since most of computational time for the JFNK methods is spent on the Krylov iteration for local linear equations [Eq. (26)], an efficient preconditioner of the Krylov methods is one of the most important key techniques to improve the convergence and ensure the high efficiency of the NEM_JFNK methods:

$$\mathbf{P}^{-1} \cdot \mathbf{J}\boldsymbol{a}\left(\vec{\tilde{X}}^{m}\right) \cdot \vec{\mathrm{v}} = \frac{\mathbf{P}^{-1} \cdot \vec{R}\left(\vec{\tilde{X}}^{m} + \varepsilon\vec{\mathrm{v}}\right) - \mathbf{P}^{-1} \cdot \vec{R}\left(\vec{\tilde{X}}^{m}\right)}{\varepsilon}, \tag{43}$$

where **P** is the preconditioning matrix. Physics-based preconditioners for **P** are selected and constructed by making full use of the standard framework of NEM. Based on the discrete systems of NEM, $\tilde{\mathbf{M}}$, $\tilde{\mathbf{N}}$, and $\tilde{\mathbf{L}}$ can be expressed as the following formulation in detail:

$$\tilde{\mathbf{M}} = \begin{bmatrix} \tilde{\mathbf{M}}_1 & & & \\ & \tilde{\mathbf{M}}_2 & & \\ & & \ddots & \\ & & & \tilde{\mathbf{M}}_G \end{bmatrix},$$

$$\tilde{\mathbf{N}} = \begin{bmatrix} \mathbf{0} & \tilde{\mathbf{N}}_{2\to1} & \cdots & \tilde{\mathbf{N}}_{G\to1} \\ \tilde{\mathbf{N}}_{1\to2} & \mathbf{0} & \cdots & \tilde{\mathbf{N}}_{G\to2} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\mathbf{N}}_{1\to G} & \tilde{\mathbf{N}}_{2\to G} & \cdots & \mathbf{0} \end{bmatrix},$$

and

$$\tilde{\mathbf{L}} = \begin{bmatrix} \tilde{\mathbf{L}}_1 & & & \\ & \tilde{\mathbf{L}}_2 & & \\ & & \ddots & \\ & & & \tilde{\mathbf{L}}_G \end{bmatrix}; \tag{44}$$

$$\tilde{\mathbf{M}}_{\mathbf{g}} = \begin{bmatrix} \tilde{\mathbf{M}}_{\mathbf{J}_{gr+},\,\mathbf{J}_{gr+}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \tilde{\mathbf{M}}_{\bar{\Phi}_g,\,\mathbf{J}_{gr+}} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \tilde{\mathbf{M}}_{\mathbf{a}_{gr,1-3},\,\mathbf{J}_{gr+}} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \tilde{\mathbf{M}}_{\mathbf{a}_{gr,2-4},\,\mathbf{J}_{gr+}} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}$$

and

$$\tilde{\mathbf{L}}_{\mathbf{g}} = \begin{bmatrix} \tilde{\mathbf{L}}_{\mathbf{J}_{gr+},\,\mathbf{J}_{gr+}} & & & \\ & \mathbf{0} & & \\ & & \mathbf{0} & \\ & & & \mathbf{0} \end{bmatrix}; \tag{45}$$

and

$$\tilde{\mathbf{M}}_{\mathbf{J_{gr+}},\,\mathbf{J_{gr+}}} = \begin{bmatrix} \tilde{\mathbf{M}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gx+}}} & \tilde{\mathbf{M}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gy+}}} & \tilde{\mathbf{M}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gz+}}} \\ \tilde{\mathbf{M}}_{\mathbf{J_{gy+}},\,\mathbf{J_{gx+}}} & \tilde{\mathbf{M}}_{\mathbf{J_{gy+}},\,\mathbf{J_{gy+}}} & \tilde{\mathbf{M}}_{\mathbf{J_{gy+}},\,\mathbf{J_{gz+}}} \\ \tilde{\mathbf{M}}_{\mathbf{J_{gz+}},\,\mathbf{J_{gx+}}} & \tilde{\mathbf{M}}_{\mathbf{J_{gz+}},\,\mathbf{J_{gy+}}} & \tilde{M}_{\mathbf{J_{gz+}},\,\mathbf{J_{gz+}}} \end{bmatrix}$$

and

$$\tilde{\mathbf{L}}_{\mathbf{J}_{gr+},\,\mathbf{J}_{gr+}} = \begin{bmatrix} \mathbf{0} & \tilde{\mathbf{L}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gy+}}} & \tilde{\mathbf{L}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gz+}}} \\ \tilde{\mathbf{L}}_{\mathbf{J_{gy+}},\,\mathbf{J_{gx+}}} & \mathbf{0} & \tilde{\mathbf{L}}_{\mathbf{J_{gy+}},\,\mathbf{J_{gz+}}} \\ \tilde{\mathbf{L}}_{\mathbf{J_{gz+}},\,\mathbf{J_{gx+}}} & \tilde{\mathbf{L}}_{\mathbf{J_{gz+}},\,\mathbf{J_{gy+}}} & \mathbf{0} \end{bmatrix}, \tag{46}$$

where

$g = 1,\ 2,\ \ldots, G$

$\tilde{\mathbf{M}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gy+}}}$ = the coefficient matrices of J_{gy+} for diffusion and removal terms in the corresponding discrete equations of J_{gx+}

$\tilde{\mathbf{L}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gy+}}}$ = the coefficient matrices of J_{gy+} for transverse leakage terms in the corresponding discrete equations of J_{gx+}

$\mathbf{I}$ = unit diagonal matrix

scattering matrices $\widetilde{\mathbf{N}}_{\mathbf{g'\to g}}$ = diagonal matrices that map scattering cross sections to spatial cells,

and where other elements in Eqs. (44), (45), and (46) have similar meanings with $\widetilde{\mathbf{M}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gy+}}}$ and $\widetilde{\mathbf{L}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gy+}}}$.

The diagonal block matrices $\widetilde{\mathbf{M}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gx+}}}$, $\widetilde{\mathbf{M}}_{\mathbf{J_{gy+}},\,\mathbf{J_{gy+}}}$, and $\widetilde{\mathbf{M}}_{\mathbf{J_{gz+}},\,\mathbf{J_{gz+}}}$ are tridiagonal matrices constructed from the transverse-integrated equations, which are solved using the TriDiagonal Matrix Algorithm (TDMA) method. Then an alternating direction iterative method (ADI) is usually adopted to solve the coupled discrete systems in the *x-, y-*, and *z*-directions for the traditional NEM. Therefore, the following matrix is formed and chosen as the physics-based preconditioner $\mathbf{P}$ for NEM_JFNK:

$$\mathbf{P} = \begin{bmatrix} \tilde{\mathbf{P}} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}$$

and

$$\tilde{\mathbf{P}} = \begin{bmatrix} \tilde{\mathbf{M}}_{\mathbf{P1}} - \tilde{\mathbf{L}}_{\mathbf{P1}} & \mathbf{0} & \cdots & \mathbf{0} \\ -\tilde{\mathbf{N}}_{\mathbf{1}\to 2} & \tilde{\mathbf{M}}_{\mathbf{P2}} - \tilde{\mathbf{L}}_{\mathbf{P2}} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ -\tilde{\mathbf{N}}_{\mathbf{1}\to G} & -\tilde{\mathbf{N}}_{2\to G} & \cdots & \tilde{\mathbf{M}}_{\mathbf{P}G} - \tilde{\mathbf{L}}_{\mathbf{P}G} \end{bmatrix}, \tag{47}$$

where the final diagonal element 1 is considered for the preconditioning of the nonlinear residuals of k_{eff}. The expression of $\widetilde{\mathbf{M}}_{\mathbf{Pg}}$ and $\widetilde{\mathbf{L}}_{\mathbf{Pg}}$ for $g = 1, 2, \ldots, G$ can be given by

$$\tilde{\mathbf{M}}_{\mathbf{Pg}} = \begin{bmatrix} \begin{bmatrix} \tilde{\mathbf{M}}_{\mathbf{J_{gx+}},\,\mathbf{J_{gx+}}} & \mathbf{0} & \mathbf{0} \\ \tilde{\mathbf{M}}_{\mathbf{J_{gy+}},\,\mathbf{J_{gx+}}} & \tilde{\mathbf{M}}_{\mathbf{J_{gy+}},\,\mathbf{J_{gy+}}} & \mathbf{0} \\ \tilde{\mathbf{M}}_{\mathbf{J_{gz+}},\,\mathbf{J_{gx+}}} & \tilde{\mathbf{M}}_{\mathbf{J_{gz+}},\,\mathbf{J_{gy+}}} & \tilde{\mathbf{M}}_{\mathbf{J_{gz+}},\,\mathbf{J_{gz+}}} \end{bmatrix} & \begin{matrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \\ \\ \end{matrix} \\ \begin{matrix} \tilde{\mathbf{M}}_{\bar{\varphi}_{\mathbf{g}},\,\mathbf{J_{gr+}}} \\ \tilde{\mathbf{M}}_{\mathbf{a_{gr,1-3}},\,\mathbf{J_{gr+}}} \\ \tilde{\mathbf{M}}_{\mathbf{a_{gr,2-4}},\,\mathbf{J_{gr+}}} \end{matrix} & \begin{matrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{matrix} \end{bmatrix} \tag{48}$$

and

$$\widetilde{\mathbf{L}}_{\mathbf{Pg}} = \begin{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \widetilde{\mathbf{L}}_{\mathbf{J_{gy+},\,J_{gx+}}} & \mathbf{0} & \mathbf{0} \\ \widetilde{\mathbf{L}}_{\mathbf{J_{gz+},\,J_{gx+}}} & \widetilde{\mathbf{L}}_{\mathbf{J_{gz+},\,J_{gy+}}} & \mathbf{0} \end{bmatrix} & & & \\ & \mathbf{0} & & \\ & & \mathbf{0} & \\ & & & \mathbf{0} \end{bmatrix}. \tag{49}$$

For actual NEM_JFNK codes, the preconditioner **P** does not need to be explicitly constructed and the inverse matrix of the preconditioner **P** is easily obtained by taking advantage of the original framework of NEM using power iteration. It is also straightforward and efficient to use TDMA and ADI methods in the process of solving the inverse matrix of the preconditioner **P**. In addition, the standard framework of NEM using power iteration can be expressed as

$$\widetilde{\mathbf{P}} \cdot \overrightarrow{\widetilde{\Phi}}^{l} = \left[\widetilde{\mathbf{P}} - \left(\widetilde{\mathbf{M}} - \widetilde{\mathbf{N}} - \widetilde{\mathbf{L}}\right)\right] \cdot \overrightarrow{\widetilde{\Phi}}^{l-1} + \frac{1}{k_{eff}} \cdot \widetilde{\mathbf{F}} \cdot \overrightarrow{\widetilde{\Phi}}^{l-1}. \tag{50}$$

Equation (50) can be rewritten as

$$\widetilde{\mathbf{P}}^{-1}\left[\left(\widetilde{\mathbf{M}} - \widetilde{\mathbf{N}} - \widetilde{\mathbf{L}}\right) \cdot \overrightarrow{\widetilde{\Phi}}^{l-1} - \frac{1}{k_{eff}} \cdot \widetilde{\mathbf{F}} \cdot \overrightarrow{\widetilde{\Phi}}^{l-1}\right] = \overrightarrow{\widetilde{\Phi}}^{l-1} - \overrightarrow{\widetilde{\Phi}}^{l}, \tag{51}$$

where l is the iterative step of the power iteration. Based on Eqs. (51) and (23), the terms $\mathbf{P}^{-1} \cdot \vec{R}\left(\overrightarrow{\tilde{X}}^{m}\right)$ and $\mathbf{P}^{-1} \cdot \vec{R}\left(\overrightarrow{\tilde{X}}^{m} + \varepsilon\vec{v}\right)$ in Eq. (43) can be simplified as

$$\mathbf{P}^{-1} \cdot \vec{R}\left(\overrightarrow{\tilde{X}}^{m}\right) = \overrightarrow{\tilde{X}}^{m} - \overrightarrow{\tilde{X}}^{m,\,out\mathrm{NEM}} \tag{52}$$

and

$$\mathbf{P}^{-1} \cdot \vec{R}\left(\overrightarrow{\tilde{X}}^{m} + \varepsilon\vec{v}\right) = \left(\overrightarrow{\tilde{X}}^{m} + \varepsilon\vec{v}\right) - \overrightarrow{\tilde{X}}^{m\varepsilon,out\mathrm{NEM}}, \tag{53}$$

where $\overrightarrow{\tilde{X}}^{m,out\mathrm{NEM}}$ and $\overrightarrow{\tilde{X}}^{m\varepsilon,out\mathrm{NEM}}$ are the corresponding output solutions at the new iterative step using NEM with power iteration when the solutions at the old iterative step are $\overrightarrow{\tilde{X}}^{m}$ and $\overrightarrow{\tilde{X}}^{m} + \varepsilon\vec{v}$, respectively. Therefore, the nonlinear residual functions [Eq. (43)] after preconditioning can be computed more easily by Eq. (54):

$$\mathbf{P}^{-1} \cdot \boldsymbol{Ja}\left(\overrightarrow{\tilde{X}}^{m}\right) \cdot \vec{v} = \vec{v} - \frac{\overrightarrow{\tilde{X}}^{m\varepsilon,out\mathrm{NEM}} - \overrightarrow{\tilde{X}}^{m,out\mathrm{NEM}}}{\varepsilon}. \tag{54}$$

II.D. Flowchart and Computation Cost Analysis

The flowchart of the NEM_JFNK method is shown in Fig. 2. $\| \vec{R}\left(\vec{\tilde{X}}^{0}\right) \|$ is the two-norm of the residual functions of the initial values $\overrightarrow{\tilde{X}}^{0}$ and the two parameters *rtol* and *atol* are the convergence criterion of the Newton iteration. These final-solving variables $\overrightarrow{\tilde{X}}$ are updated at each Newton step and then all the other variables that were eliminated through

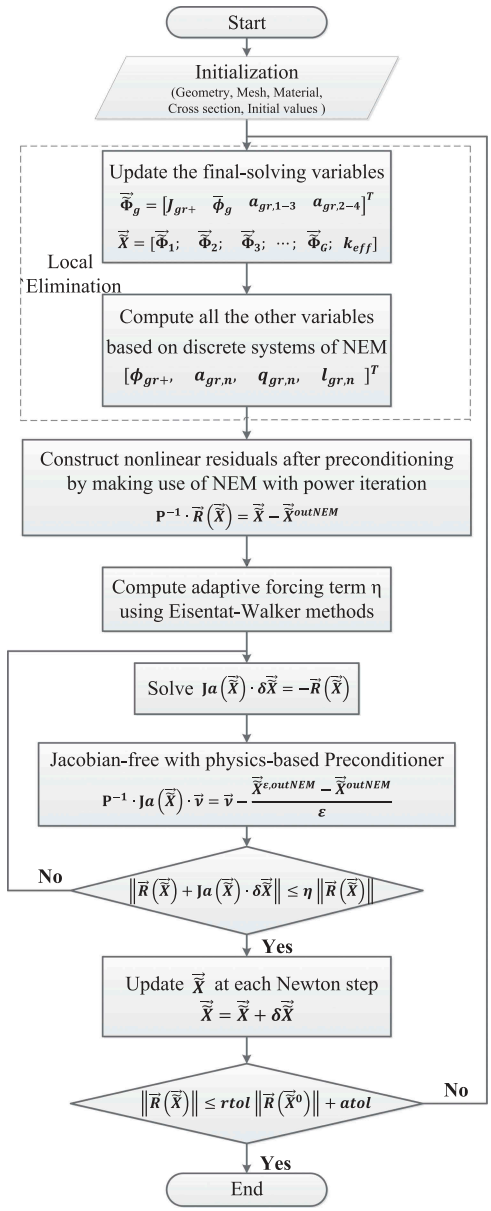


Fig. 2. Flowchart of NEM_JFNK methods.

the equivalent transformation approach can be computed in terms of $\vec{\tilde{X}}$. The nonlinear residual functions after preconditioning and the product of the Jacobian matrix and vector with physics-based preconditioner can be easily obtained by make full use of the original framework of NEM with power iteration. In this paper, the BICGSTAB method and the restarted GMRES method are chosen as the representative Krylov methods to solve the linear equations, and the restarted number of GMRES is 30.

Based on the flowchart in Fig. 2, the total computational time for NEM_JFNK methods can be approximately expressed as

$$t \approx NSteps \cdot t_{\eta} + \left[NSteps + \begin{pmatrix} 1 & \text{for GMRES} \\ 2 & \text{for BICGSTAB} \end{pmatrix} \cdot \sum_{n=1}^{NSteps} KSteps(n) \right] \cdot t_R \ , \quad (55)$$

where

t = total computational time for the NEM_JFNK method

t_{η} = time for computing forcing terms using the Eisenstat-Walker method within each Newton step

t_R = time for evaluation of the preconditioning nonlinear residual functions once

$NSteps$ = total number of Newton iteration

$KSteps(n)$ = number of Krylov iteration within each Newton step.

The value 1 or 2 means that preconditioning nonlinear residuals are evaluated once for GMRES methods and twice for BICGSTAB methods within each Krylov iterative step. Since the preconditioning nonlinear residual functions are based on NEM discrete equations using power iteration, the computational time t_R of calling the nonlinear residual functions once approximately equals the time $t_{\text{NEM\_}PI\_one}$ of NEM using power iteration in only one inner iteration:

$$t_R \approx t_{\text{NEM\_}PI\_one} \ . \quad (56)$$

It can be observed from Eqs. (55) and (56) that NEM_JFNK methods have higher efficiency or more advantages than traditional NEM using power iteration only when the following condition is satisfied:

$$\left[NSteps + \begin{pmatrix} 1 & \text{for GMRES} \\ 2 & \text{for BICGSTAB} \end{pmatrix} \cdot \sum_{n=1}^{NSteps} KSteps(n) \right] << \text{NEM\_}PI\_Steps \ , \quad (57)$$

where NEM_*PI*_*Steps* is the total inner iterative number of traditional NEM using power iteration.

III. NUMERICAL ANALYSIS AND DISCUSSION

Based on the formulation and algorithm presented in Sec. II, the code NEM_JFNK is developed in the MATLAB language by hand for the 3D and multigroup k-eigenvalue problems in neutron diffusion equations and two test problems are solved to study the numerical properties in this paper. The first case is a 3D and two-group pressurized water reactor (PWR) model named International Atomic Energy Agency (IAEA) 3D PWR problem[21] to analyze the convergence behavior, computational cost and efficiency, and so on. As a further hint, the second case is a complicated 3D and four-group pebble-bed core model. There are 180 kinds of compositions arranged in the pebble-bed core and all the cross sections and parameters are chosen from the realistic modular pebble-bed high-temperature gas-cooled reactor (HTR-PM) in order to analyze the computational properties of NEM_JFNK methods for HTR-PM as much as possible.

III.A. IAEA 3D PWR Problem

The IAEA 3D PWR problem has been a very important standard benchmark problem to measure and analyze the numerical performance of calculation methods. The schematic of one-quarter–core geometry, boundary conditions, and two-group cross sections for each composition are shown in Fig. 3 and Table II. To deal with the physical boundary easily in this work, the external boundary region (dashed part in Fig. 3) is filled with the material in region 4. $J_{\text{in}} = 0$ and $J = 0$ represent the vacuum boundary and symmetric boundary, respectively.

The numerical solutions of neutron fluxes, effective multiplication factor k_{eff}, and these errors using the NEM_JFNK methods on the uniform mesh 17 × 17 × 38 are presented in Fig. 4 and Table III, respectively. The reference solutions are obtained by using NEM with power iteration for a high convergence criterion on a very fine mesh and the root-mean-square (RMS) error is defined as follows:

$$\text{RMS} = \sqrt{\left(\vec{\tilde{X}}^{\,computation} - \vec{\tilde{X}}^{\,reference} \right) / length\left(\vec{\tilde{X}} \right)} \ , \quad (58)$$

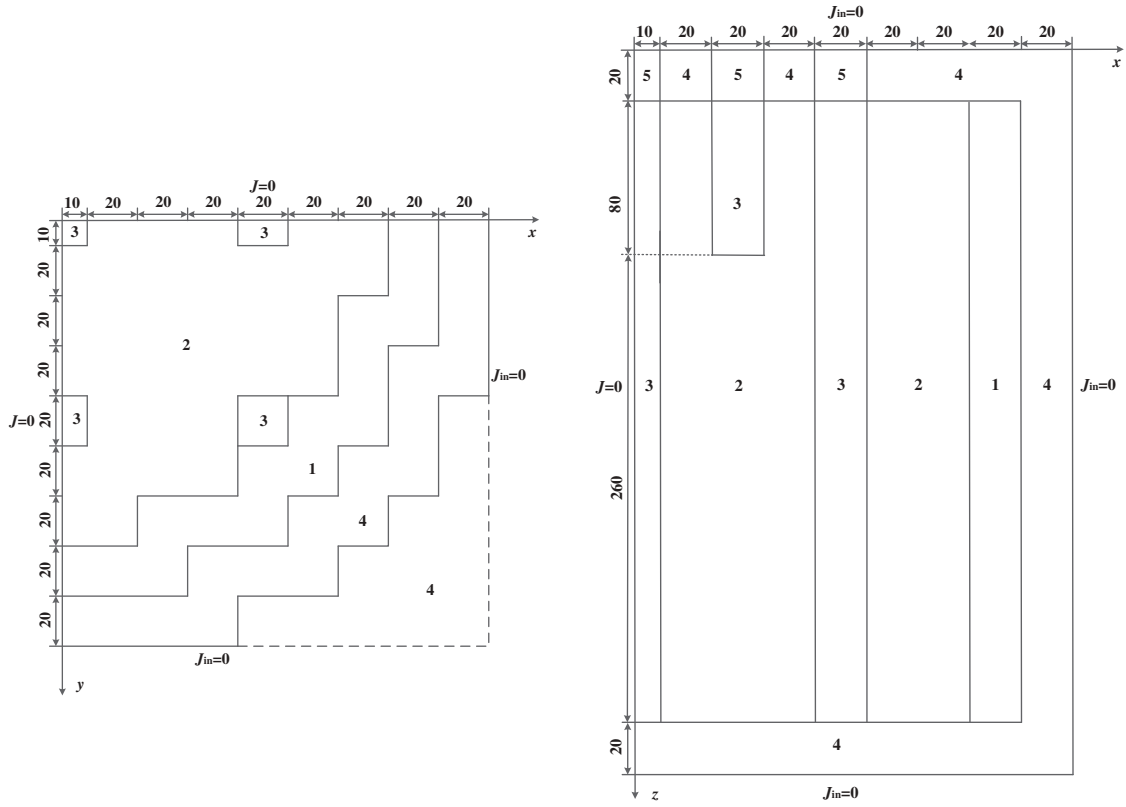


Fig. 3. Schematic of one-quarter–core geometry for IAEA 3D problem.

TABLE II

Two-Group Cross Sections of Each Composition for the IAEA 3D Problem

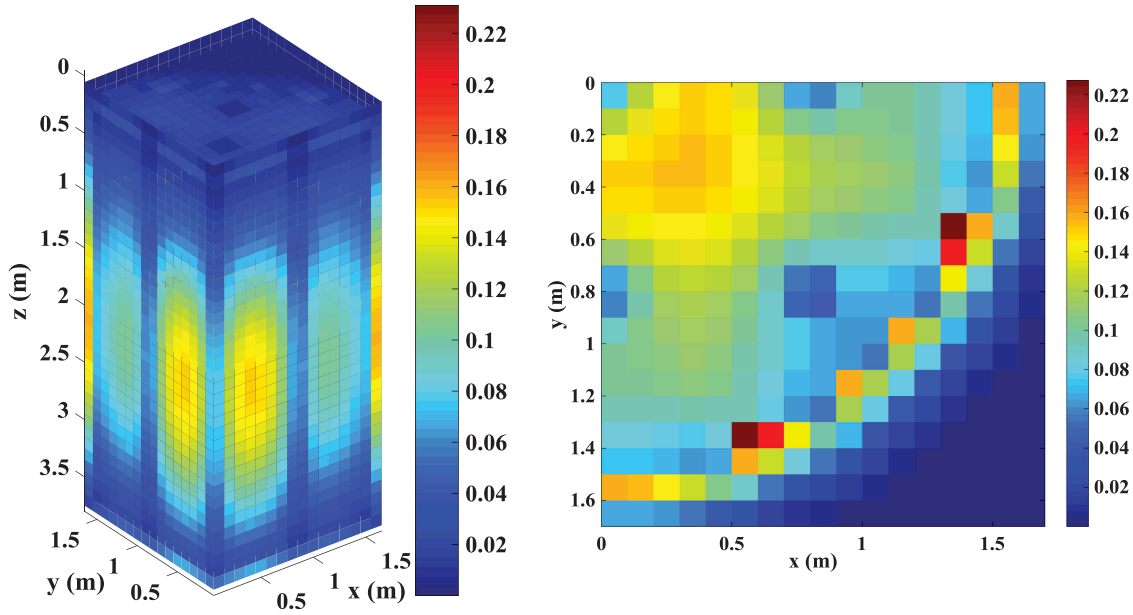
Region	D_1 (cm)	D_2 (cm)	$\Sigma_{1\to2}$ (cm^{-1})	$\Sigma_{R,1}$ (cm^{-1})	$\Sigma_{R,2}$ (cm^{-1})	$\upsilon\Sigma_{f,2}$ (cm^{-1})	Material
1	1.5	0.4	0.02	0.01	0.08	0.135	Fuel 1
2	1.5	0.4	0.02	0.01	0.085	0.135	Fuel 2
3	1.5	0.4	0.02	0.01	0.13	0.135	Fuel 2 + rod
4	2.0	0.3	0.04	0	0.01	0	Reflector
5	2.0	0.3	0.04	0	0.055	0	Reflector + rod

where $\vec{\tilde{X}}^{computation}$, $\vec{\tilde{X}}^{reference}$, and $length\left(\vec{\tilde{X}}\right)$ are the numerical solutions, reference solutions, and the length of the vector $\vec{\tilde{X}}$, respectively. Numerical solutions show that the NEM_JFNK methods have good numerical accuracy.

To analyze the convergence behavior, Fig. 5a gives the curves about the normalized residual error varied with the nonlinear iterative number for NEM_JFNK methods and NEM with power iteration, and Fig. 5b shows the Krylov iterative number with each Newton step. It is observed from Fig. 5 that the Newton and Krylov iterative numbers of the NEM_JFNK methods are far less than the iterative number of traditional NEM with power iteration. The total Krylov iterative number for NEM_JFNK (GMRES) is 57, which is about 2 times as large as that (the value is 29) for NEM_JFNK (BICGSTAB). So Eq. (57) is satisfied as shown in Eq. (59):

$$NSteps + [1(\text{GMRES})\ or\ 2(\text{BICGSTAB})] \cdot \sum_{n=1}^{NSteps} KSteps(n) = 62(\text{GMRES})\ or\ 63(\text{BICGSTAB}) < \text{NEM\_}PI\text{\_}Steps = 532\ . \quad (59)$$

In addition, to analyze the advantage of local elimination, the computational cost for different selection methods of



(a) Three-dimensional profile (b) Profile at z=1.85 m

Fig. 4. Numerical solutions of fluxes (g = 2) using NEM_JFNK (GMRES) for IAEA 3D problem.

TABLE III

Result and Error of IAEA 3D Problem for Different Numerical Methods

Method	k_{eff}	Relative Error of k_{eff}	RMS of Neutron Fluxes
NEM_JFNK (BICGSTAB)	1.0291375	0.00335%	2.865165E−04
NEM_JFNK (GMRES)	1.0291377	0.00337%	2.870565E−04
NEM	1.0291375	0.00335%	2.884303E−04
Reference	1.0291030	–	–

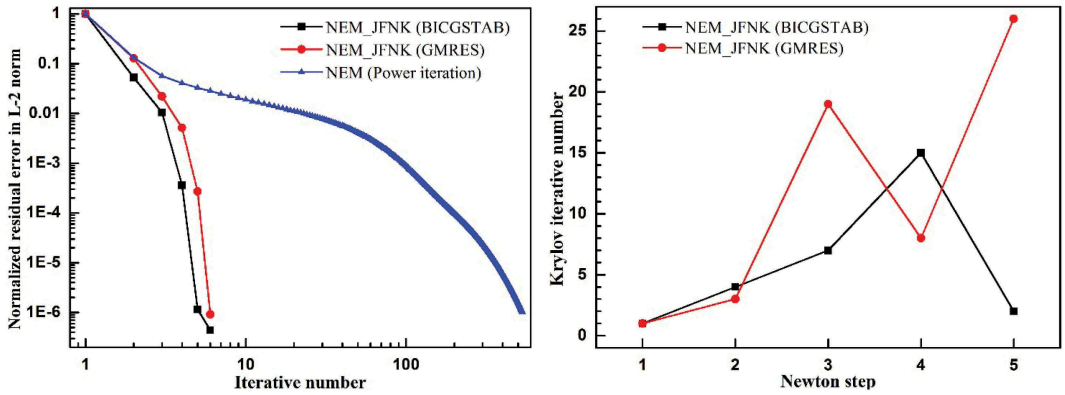
final-solving variables is shown in Table IV and the convergent criterion is (*rtol, atol*) = (10^{-6}, 10^{-12}). The final-solving variables $[J_{gr+}\ \bar{\phi}_g\ a_{gr,1-3}$ and $a_{gr,2-4}]$ are the selection methods for local elimination in this paper, and it is observed that the local elimination technique improves the efficiency.

The computational cost for different mesh sizes, convergence criterions, and numerical methods is summarized in detail in Tables V, VI, and VII. It is shown that NEM_JFNK (BICGSTAB) spends the least time for IAEA 3D problems and has the highest efficiency. The efficiency of NEM_JFNK (BICGSTAB) is about 4 to 5 times for (*rtol, atol*) = (10^{-6}, 10^{-12}) and 7 to 8 times for (*rtol, atol*) = (10^{-10}, 10^{-16}) as much as that of traditional NEM with power iteration, which indicates the NEM_JFNK (BICGSTAB) methods take greater advantage over traditional NEM with the improvement of the required convergence precision. We also can know the efficiency of NEM_JFNK (GMRES) is about 3 to 4 times for (*rtol, atol*) = (10^{-6}, 10^{-12}) and 5 to 6 times for (*rtol, atol*) = (10^{-10}, 10^{-16}) as much as that of traditional NEM with power iteration. In addition, the advantage of the computational cost for the adaptive forcing terms η using the Eisenstat-Walker method is analyzed and compared with that of the constant value of η, as shown in Table VIII.

III.B. Three-Dimensional and Four-Group Pebble-Bed Core Models

To analyze the numerical performance of the NEM_JFNK methods for pebble-bed core problems further, a 3D and four-group pebble-bed core model is constructed and solved in this work. All the cross sections and parameters are selected from the realistic HTR-PM and 180 kinds of compositions are arranged in the pebble-bed core. The computational domain is [0, 2.5 m] × [0, 2.5 m] × [0, 16.8 m] and the boundary conditions are given by

$$J = 0 \text{ for } x = 0.0 \text{ m} ,$$



(a) Normalized residual error vs iterative number (b) Krylov number with each Newton step

Fig. 5. Convergence behavior of IAEA 3D problem for different methods.

TABLE IV

Cost Comparison of IAEA 3D Problem for Different Selection Methods of Local Elimination

Method	Final Variables for NEM_JFNK	Number of Final Variables for Group g Within Each Mesh	Time (s)
NEM	–	–	14.70
NEM_JFNK (BICGSTAB)	$[J_{gr+} \quad \bar{\phi}_g \quad a_{gr,1-3} \quad a_{gr,2-4}]$	10	2.62
	$[J_{gr+} \quad \bar{\phi}_g \quad a_{gr,1} \quad a_{gr,2} \quad a_{gr,3} \quad a_{gr,4}]$	16	6.98
NEM_JFNK (GMRES)	$[J_{gr+} \quad \bar{\phi}_g \quad a_{gr,1-3} \quad a_{gr,2-4}]$	10	3.67
	$[J_{gr+} \quad \bar{\phi}_g \quad a_{gr,1} \quad a_{gr,2} \quad a_{gr,3} \quad a_{gr,4}]$	16	10.45

$$J_{\text{in}} = 0 \text{ for } x = 2.5 \text{ m} ,$$

$$J = 0 \text{ for } y = 0.0 \text{ m} ,$$

$$J_{\text{in}} = 0 \text{ for } y = 2.5 \text{ m} ,$$

$$J_{\text{in}} = 0 \text{ for } z = 0.0 \text{ m} ,$$

and

$$J_{\text{in}} = 0 \text{ for } z = 16.8 \text{ m} . \tag{60}$$

The nonuniform mesh is used because of nonuniform material distribution and NEM_JFNK (BICGSTAB) are only chosen as a representative of NEM_JFNK methods to solve the case due to its high accuracy for IAEA 3D problems. The numerical solutions using the NEM_JFNK (BICGSTAB) method and the errors of k_{eff} for different methods on the coarse material mesh 16 × 16 × 92 are shown in Fig. 6 and Table IX. Numerical solutions show that the NEM_JFNK method still has good numerical accuracy.

Figure 7 gives the curves about the normalized residual error varied with the nonlinear iterative number for different methods and the Krylov iterative number with each Newton step for NEM_JFNK (BICGSTAB). The Newton and Krylov iterative numbers of NEM_JFNK (BICGSTAB) are also far less than the iterative number of traditional NEM with power iteration for the 3D pebble-bed problem. Equation (57) can be expressed as

$$NSteps + 2 \cdot \sum_{n=1}^{NSteps} KSteps(n) = 61 < \text{NEM\_}PI\_Steps = 269 . \tag{61}$$

Tables X and XI show the computational cost of NEM_JFNK (BICGSTAB) compared with that of NEM with power iteration for different mesh sizes and convergence criterions. It is shown that the efficiency of NEM_JFNK (BICGSTAB) is about 3 to 4 times for (*rtol, atol*) = (10^{-6}, 10^{-12}) and 9 times for (*rtol, atol*) = (10^{-10}, 10^{-16}) as much as that of traditional NEM with power iteration. The advantage of the adaptive

TABLE V

Computational Performance Comparison of the IAEA 3D Problem Using the Mesh 9 × 9 × 19

Criterion *rtol, atol*	Method	Normalized Residual Error	Nonlinear Iterative Number	Total Krylov Number	Total Number of Residual Function Evaluations	Time (s)	Speedup Ratio
1.0E−06 1.0E−12	NEM	9.93E−07	532	–	–	2.22	–
	NEM_JFNK (BICGSTAB)	4.16E−07	5	32	71	0.49	4.53
	NEM_JFNK (GMRES)	4.37E−07	5	45	62	0.60	3.70
1.0E−10 1.0E−16	NEM	9.97E−11	1209	–	–	4.78	–
	NEM_JFNK (BICGSTAB)	1.88E−11	5	52	111	0.65	7.35
	NEM_JFNK (GMRES)	4.79E−11	5	91	99	0.76	6.23

TABLE VI

Computational Performance Comparison of the IAEA 3D Problem Using the Mesh 17 × 17 × 38

Criterion *rtol, atol*	Method	Normalized Residual Error	Nonlinear Iterative Number	Total Krylov Number	Total Number of Residual Function Evaluations	Time (s)	Speedup Ratio
1.0E−06 1.0E−12	NEM	9.96E−07	534	–	–	14.70	–
	NEM_JFNK (BICGSTAB)	4.43E−07	5	29	65	2.62	5.61
	NEM_JFNK (GMRES)	9.15E−07	5	57	64	3.67	4.01
1.0E−10 1.0E−16	NEM	9.90E−11	1211	–	–	33.61	–
	NEM_JFNK (BICGSTAB)	4.78E−11	5	48	103	4.05	8.30
	NEM_JFNK (GMRES)	3.95E−11	5	99	107	6.56	5.12

TABLE VII

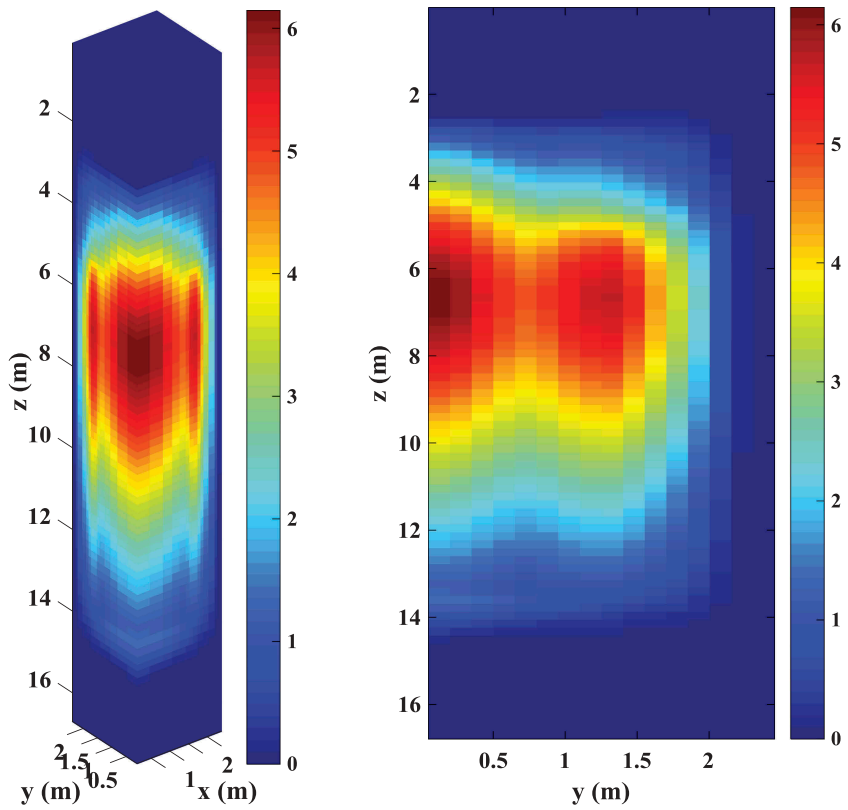
Computational Performance Comparison of the IAEA 3D Problem Using the Mesh 34 × 34 × 76

Criterion *rtol, atol*	Method	Normalized Residual Error	Nonlinear Iterative Number	Total Krylov Number	Total Number of Residual Function Evaluations	Time (s)	Speedup Ratio
1.0E−06 1.0E−12	NEM	9.93E−07	536	–	–	137.00	–
	NEM_JFNK (BICGSTAB)	4.32E−07	5	32	71	24.5	5.59
	NEM_JFNK (GMRES)	4.62E−07	5	62	69	39.52	3.47
1.0E−10 1.0E−16	NEM	9.99E−11	1212	–	–	309.48	–
	NEM_JFNK (BICGSTAB)	3.16E−11	6	55	118	40.80	7.59
	NEM_JFNK (GMRES)	4.07E−11	6	107	115	67.62	4.58

TABLE VIII

Performance Analysis of Adaptive η Using NEM_JFNK (BICGSTAB) for IAEA 3D Problem

Method of Choosing Forcing Terms η		Number of Newton Iterations	Total Krylov Number	Total Number of Residual Function Evaluations	Time (s)
Adaptive (Eisenstat-Walker)		5	29	65	2.62
Constant value	0.5	13	28	71	2.79
	0.1	7	22	31	3.07
	0.05	6	38	84	3.42
	0.01	8	615	1241	52.30
	0.005	10	946	1923	78.11



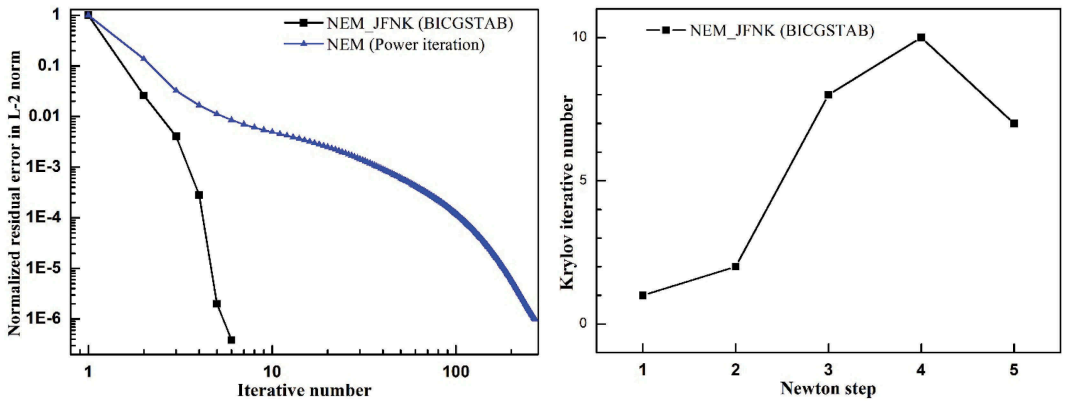
(a) Profile of 3D (b) Profile at x=0.125 m

Fig. 6. Numerical solutions of fluxes (g = 4) using NEM_JFNK (BICGSTAB) for the pebble-bed model.

TABLE IX

Result and Error of Pebble-Bed Model for Different Numerical Methods

Method	k_{eff}	Relative Error of k_{eff}
NEM_JFNK(BICGSTAB)	1.0153724	0.00581%
NEM	1.0153720	0.00577%
Reference	1.0153134	–



(a) Normalized residual error vs iterative number (b) Krylov number with each Newton step

Fig. 7. Convergence behavior of pebble-bed model for different methods.

TABLE X

Computational Performance Comparison of Pebble-Bed Model Using the Mesh 16 × 16 × 92

Criterion *rtol, atol*	Method	Normalized Residual Error	Nonlinear Iterative Number	Total Krylov Number	Total Number of Residual Function Evaluations	Time (s)	Speedup Ratio
1.0E−06	NEM	9.95E−07	269	–	–	36.52	–
1.0E−12	NEM_JFNK (BICGSTAB)	3.86E−07	5	28	63	10.89	3.35
1.0E−10	NEM	9.99E−11	3262	–	–	441.50	–
1.0E−16	NEM_JFNK (BICGSTAB)	8.98E−11	5	118	243	45.44	9.72

TABLE XI

Computational Performance Comparison of Pebble-Bed Model Using the Mesh 24 × 24 × 157

Criterion *rtol, atol*	Method	Normalized Residual Error	Nonlinear Iterative Number	Total Krylov Number	Total Number of Residual Function Evaluations	Time (s)	Speedup Ratio
1.0E−06	NEM	9.78E−07	267	–	–	152.15	–
1.0E−12	NEM_JFNK (BICGSTAB)	4.70E−07	4	22	50	36.72	4.14
1.0E−10	NEM	9.99E−11	3001	–	–	1693.7	–
1.0E−16	NEM_JFNK (BICGSTAB)	9.61E−11	4	113	232	174.27	9.71

TABLE XII

Performance Analysis of Adaptive η Using NEM_JFNK (BICGSTAB) for Pebble-Bed Problem

Method of Choosing Forcing Terms η		Number of Newton Iterations	Total Krylov Number	Total Number of Residual Function Evaluations	Time (s)
Adaptive (Eisenstat-Walker)		4	22	50	36.72
Constant value	5E−01	11	21	55	40.77
	1E−01	5	22	51	37.83
	5E−02	5	27	61	45.55
	1E−02	4	44	94	70.12
	5E−03	3	27	59	43.83
	1E−03	3	40	85	63.39
	5E−04	4	83	172	129.23

forcing terms η for the pebble-bed 3D problem is summarized in Table XII.

IV. CONCLUSION

This paper presents a new NEM_JFNK method with a physics-based preconditioner and local elimination to solve 3D and multigroup *k*-eigenvalue problems in neutron diffusion equations by combining and integrating the NEM discrete systems into the framework of the JFNK methods. The developed NEM_JFNK methods make full use of the respective advantages of the NEM and JFNK methods. Then we develop a local elimination technique, a physics-based preconditioner based on the iterative framework of the traditional NEM, and an adaptive forcing term for NEM_JFNK methods to improve the efficiency and accuracy on the coarse mesh as much as possible.

It is observed from these numerical solutions that the NEM_JFNK methods can greatly improve the convergence rate and the computational efficiency. For the cases in this paper, the computational efficiency of the NEM_JFNK methods is about 3 to 5 times for (*rtol, atol*) = (10^{-6}, 10^{-12}) and 7 to 9 times for (*rtol, atol*) = (10^{-10}, 10^{-16}) as much as that of traditional NEM with power iteration. With the improvement of the required convergence precision, the NEM_JFNK (BICGSTAB) methods take greater advantage over traditional NEM with power iteration. Further studies are needed to integrate the acceleration methods such as Chebyshev and Wielandt shift methods into the NEM_JFNK methods in order to improve the efficiency further. Then the developed NEM_JFNK methods will be implemented into the transient-coupled engineering design code TINTE to solve the more realistic HTR problems and improve the computational efficiency for large-scale complicated multiphysics coupled problems in nuclear reactor analysis.

Acknowledgment

This research is supported by the Chinese National S&T Major Project ZX06901 and the Chinese National Natural Science Foundation Project 11375099.

References

1. D. A. KNOLL and D. E. KEYES, "Jacobian-Free Newton-Krylov Methods: A Survey of Approaches and Applications," *J. Comput. Phys.*, **193**, *2*, 357 (2004); https://doi.org/10.1016/j.jcp.2003.08.010.
2. D. F. GILL and Y. Y. AZMY, "Newton's Method for Solving *k*-Eigenvalue Problems in Neutron Diffusion Theory," *Nucl. Sci. Eng.*, **167**, *2*, 141 (2011); https://doi.org/10.13182/NSE09-98.
3. V. S. MAHADEVAN and J. RAGUSA, "Novel Hybrid Scheme to Compute Several Dominant Eigenmodes for Reactor Analysis Problems," *Int. Conf. Physics of Reactors (PHYSOR2008)*, Interlaken, Switzerland, September 14–19, 2008.
4. D. A. KNOLL, H. PARK, and C. NEWMAN, "Acceleration of *k*-Eigenvalue/Criticality Calculations Using the Jacobian-Free Newton-Krylov Method," *Nucl. Sci. Eng.*, **167**, *2*, 133 (2011); https://doi.org/10.13182/NSE09-89.
5. D. F. GILL et al., "Newton's Method for the Computation of *k*-Eigenvalues in S_N Transport Applications," *Nucl. Sci. Eng.*, **168**, *1*, 37 (2011); https://doi.org/10.13182/NSE10-01.
6. L. ZOU, H. H. ZHAO, and H. B. ZHANG, "Solving Phase Appearance/Disappearance Two-Phase Flow Problems with

High Resolution Staggered Grid and Fully Implicit Schemes by the Jacobian-Free Newton-Krylov Method," *Comput. Fluids*, **129**, 179 (2016); https://doi.org/10.1016/j.compfluid.2016.02.008.

7. H. PARK et al., "Jacobian-Free Newton Krylov Discontinuous Galerkin Method and Physics-Based Preconditioning for Nuclear Reactor Simulations," *Int. Conf. Physics of Reactors (PHYSOR2008)*, Interlaken, Switzerland, September 14–19, 2008.

8. P. LUCAS, A. H. V. ZUIJLEN, and H. BIJL, "Fast Unsteady Flow Computations with a Jacobian-Free Newton–Krylov Algorithm," *J. Comput. Phys.*, **229**, *24*, 9201 (2010); https://doi.org/10.1016/j.jcp.2010.08.033.

9. J. GAN, Y. L. XU, and T. DOWNAR, *A Matrix-Free Newton Method for Coupled Neutronics Thermal-Hydraulics Reactor Analysis*, Nuclear Mathematical and Computational Sciences, Gatlinburg, Tennessee (2003).

10. V. S. MAHADEVAN, "High-Resolution Numerical Methods for Coupled Non-Linear Multi-Physics Simulations with Applications in Reactor Analysis," PhD Thesis, Texas A&M University (2010).

11. D. R. GASTON et al., "Physics-Based Multiscale Coupling for Full Core Nuclear Reactor Simulation," *Ann. Nucl. Energy*, **84**, 45 (2015); https://doi.org/10.1016/j.anucene.2014.09.060.

12. R. D. LAWRENCE, "Progress in Nodal Methods for the Solution of the Neutron Diffusion and Transport Equations," *Prog. Nucl. Energy*, **17**, 271 (1986); https://doi.org/10.1016/0149-1970(86)90034-X.

13. K. SMITH. "An Analytical Nodal Method for Solving the Two-Group, Multi-Dimensional, Static and Transient Neutron Diffusion Equations," PhD Thesis, Massachusetts Institute of Technology (1979).

14. Y. Q. WANG, "Nonlinear Diffusion Acceleration for the Multigroup Transport Equation Discretized with S_N and Continuous FEM with RattleSnake," *Proc. Int. Conf. Mathematics and Computational Methods Applied to Nuclear Science & Engineering (M&C2013)*, Sun Valley, Idaho, May 5–9, 2013.

15. J. D. HALES et al., "Verification of the BISON Fuel Performance Code," *Ann. Nucl. Energy*, **71**, 81 (2014); https://doi.org/10.1016/j.anucene.2014.03.027.

16. R. A. BERRY et al., "RELAP-7 Theory Manual," INL/EXT–14-31366, Idaho National Laboratory (2016).

17. S. C. EISENSTAT and H. F. WALKER, "Choosing the Forcing Terms in an Inexact Newton Method," *SIAM J. Scientific Comput.*, **17**, *1*, 16 (1996); https://doi.org/10.1137/0917003.

18. J. J. DUDERSTADT and L. J. HAMILTON, *Nuclear Reactor Analysis*, Wiley, New York (1976).

19. Y. L. XU, "A Matrix Free Newton/Krylov Method for Coupling Complex Multi-Physics Subsystems," PhD Thesis, Purdue University (2004).

20. P. N. BROWN and Y. SAAD, "Hybrid Krylov Methods for Nonlinear Systems of Equations," *SIAM J. Sci. Stat. Comput.*, **11**, *3*, 450 (1990); https://doi.org/10.1137/0911026.

21. "Benchmark Problem Book," ANL-7416, Suppl. 2, Argonne National Laboratory (1977).