# COMPARATIVE ANALYSIS OF STATIC AND KINETIC CALCULATIONAL FEATURES IN THE ANALYTIC FUNCTION EXPANSION NODAL METHOD

**Do Sam Kim and Nam Zin Cho**

Korea Advanced Institute of Science and Technology
Department of Nuclear and Quantum Engineering
373-1 Kusong-dong, Yusong-gu
Taejon, Korea 305-701
Email : kds@nurapt.kaist.ac.kr ; nzcho@mail.kaist.ac.kr

## ABSTRACT

We present the characteristics and comparative analysis of several features used in the AFEN method and implemented in the AFENX code. The two acceleration methods of coarse mesh finite difference (CMFD) and coarse group rebalance (CGR) are described. They are applied both to the original AFEN formulation and to the formulation that uses interface flux moments instead of edge fluxes. The results show that the CGR acceleration is more efficient than the CMFD acceleration. Also, we present the results of adaptive time step size control and space-time discretization stability analysis in the kinetics calculation.

## 1. INTRODUCTION

Recently, the AFEN method[1-3] has been developed to overcome the limitations[4] caused by the transverse integration in modern nodal methods[5]. All of the methods are now implemented in the AFENX code[6] which can treat the multigroup problems with albedo boundary conditions in three-dimensional rectangular, hexagonal and cylindrical R-Z geometries. In this paper, we compared and analyzed the CMFD[7,8] and CGR[9] acceleration methods applied to the different formulations of AFEN method. An adaptive time step size control method is introduced using an improved second-derivative-based error approximation in the kinetics calculation and derived the stability criteria of the space-time discretization.

## 2. ACCELERATION METHODS

### 2.1 Coarse Mesh Finite Difference (CMFD) Nonlinear Acceleration

In AFEN, two correction factors were adopted to make the interface currents in the CMFD method preserve the values of the higher order nodal method as follows[7]:

$$J_{g,r}^{m,u} = -\frac{2D_g^m}{h_u^m}\left(\tilde{\phi}_{g,r}^{m,u} - \phi_g^{m,av}\right) - \frac{2D_{g,r}^{m,u}}{h_u^m}\left(\tilde{\phi}_{g,r}^{m,u} + \phi_g^{m,av}\right), \tag{1a}$$

$$J_{g,l}^{m+1,u} = -\frac{2D_g^{m+1}}{h_u^{m+1}}\left(\tilde{\phi}_{g,l}^{m+1,u} - \phi_g^{m+1,av}\right) + \frac{2D_{g,l}^{m+1,u}}{h_u^m}\left(\tilde{\phi}_{g,l}^{m+1,u} + \phi_g^{m+1,av}\right), \tag{1b}$$

where

$u$ : direction index,

$\tilde{\phi}_{g,r}^{m,u}$ : interface flux of node $m$ at right,

$\tilde{\phi}_{g,l}^{m+1,u}$ : interface flux of node $m$ +1 at left,

$D_{g,r}^{m,u}$ : correction factor for node $m$ at right,

$D_{g,l}^{m+1,u}$ : correction factor for node $m$+1 at left,

$\phi_g^{m,av}$, $\phi_g^{m+1,av}$ : node average fluxes of node $m$ and $m$+1, respectively.

Note that Eq. (1) uses two direction-dependent corrective "diffusion coefficients" per axis for a node, which were adopted and generalized in Ref. 10 to propose a whole-core nodal/CMFD scheme. To obtain the correction factors, we need a relation between interface currents and node average flux. Because there are 9 nodal unknowns for each group in two-dimensional geometry, the interface current of each group is expressed in terms of 2 node average fluxes, 7 interface currents and 8 corner-point fluxes. In three-dimensional geometry, it is expressed in terms of 2 node average fluxes, 11 interface currents and 24 edge fluxes. We applied two different CMFD schemes to the developed codes. The two-node CMFD scheme solves the two-node problem in which the interface currents between two nodes expressed in terms of node-average fluxes, adjacent interface currents and edge fluxes as the boundary conditions. In the whole-core AFEN/CMFD scheme, the correction factors are updated by using the interface net currents obtained during the AFEN iteration, not by solving the two-node problem. Due to the nonlinearity and the larger number of correction factors which are directly connected to the edge fluxes, the strategy for the CMFD iteration should be carefully designed to obtain stable convergence.

2.2 Coarse Group Rebalance (CGR) Acceleration

Although the CMFD method solves mainly the FDM equation and the correction factors are updated, the CGR rebalance method uses the rebalance equation only for the acceleration of the original AFEN iteration. The rebalance equation is obtained from the group summation of the two-group neutron diffusion equations which can be written as follows:

$$\left[ R^m + \sum_{u=x,y,z} \frac{1}{h_u^m} \left( \sum_{s=l,r} \sum_{g=f,t} J_{g,s}^{m,u,out} \right) - \frac{1}{k_{eff}} P^m \right] f^m = \sum_{u=x,y,z} \frac{1}{h_u^m} \sum_{g=f,t} \left( J_{g,l}^{mr,u,in} + J_{g,r}^{ml,u,in} \right), \qquad (2)$$

where

$$P^m = \nu\Sigma_{f1}^m \phi_1^{m,av} + \nu\Sigma_{f2}^m \phi_2^{m,av},$$
$$R^m = \Sigma_{a1}^m \phi_1^{m,av} + \nu\Sigma_{a2}^m \phi_2^{m,av}.$$

In this method, it is more convenient to use the partial current formulation than the net current formulation. Eq. (2) has the same structure of the FDM equations and can be solved as an eigenvalue ($k_{eff}$) problem with the eigenvector ($f^m$). The calculated value of $f^m$ is multiplied to all the nodal unknowns within node *m* and is used as the initial conditions for next AFEN outer iteration.

## 3. EDGE FLUX VERSUS INTERFACE FLUX MOMENT

In this section, we compare the acceleration methods implemented in two different AFEN formulations. Formulation I is the original AFEN method where the nodal unknowns are node average fluxes, interface fluxes and edge fluxes. Recently, the method was refined by addition of the interface flux moments as nodal unknowns. Formulation II is obtained from this formulation by omitting the edge fluxes. Following is the number of nodal unknowns per group per node in the two formulations in three-dimensional geometry:

- Formulation I: 1 node average flux, 6 interface fluxes and 12 edge fluxes.
- Formulation II: 1 node average flux, 6 interface fluxes and 12 interface flux moments.

The detailed comparisons of the accuracy and basis functions are given in Ref. 11.

3.1 CMFD scheme

Although the two-node scheme was successful in two-dimensional geometry, the scheme has some instability in three-dimensional geometry originating from its nonlinearity. Fig. 1 shows the calculational flowchart of the scheme in the formulation II. In the case of the two-node CMFD scheme, initialization using whole-core AFEN iteration and flux moments update (shown as grey box) after FDM iteration are essential for its convergence. This means that the correction factors obtained from the two-node calculation should be consistently updated as the flux moments are updated. In the case of the whole-core AFEN/CMFD scheme, we need additional iteration steps (shown as grey boxes) in Case (b) of Fig. 1 to make the iteration stable. Without this additional iteration steps, the whole-core AFEN/CMFD scheme does not converge. In formulation I, the two-node CMFD scheme works but the whole-core AFEN/CMFD scheme does not converge in spite of the additional iteration.

### 3.2 CGR acceleration scheme

The calculational procedure of the CGR acceleration scheme is shown in Fig. 2. Original AFEN iteration in the initial phase is performed to reduce computing time by providing good initial guess of fluxes. This scheme does not suffer from the stability problem occurring in the CMFD scheme in both of the formulations.

## 4. VARIBLE TIME STEP AND STABILITY

### 4.1 Adaptive time step size control in kinetics calculation

In addition to the CGR method, automatic time step size control using an approach of the second-derivative-based error approximation[12] is developed for the AFEN kinetics calculation. The temporal truncation error in the limit of small time step size is proportional to the second derivative of the fluxes and can be approximated as follows:

$$\phi_g^m(t_n) = \phi_{g,\Delta t}^m + \frac{1}{2}\left.\frac{d^2\phi_g^m}{dt^2}\right|_{t_n} \Delta t^2 + O(\Delta t^3), \tag{3a}$$

$$\left.\frac{d^2\phi_g^m}{dt^2}\right|_{t_n} \approx \phi_{g,2\Delta t}^m - \phi_{g,\Delta t}^m \equiv \Delta\phi_{g,\Delta t}^m, \tag{3b}$$

where

$$\Delta t = t_n - t_{n-1} = t_n - t_{n-2},$$

$\phi_g^m(t_n)$ : exact flux at $t_n$,

$\phi_{g,2\Delta t}^m$ : flux at $t_n$ obtained by advancing from time $t_{n-2}$ to $t_n$ in a single time step,

$\phi_{g,\Delta t}^m$ : flux at $t_n$ obtained by advancing from time $t_{n-2}$ to $t_n$ in two equal time steps.

Although the authors of Ref. 12 used approximations to obtain the truncation error, we calculate $\phi_{g,2\Delta t}^m$ explicitly by using $\phi_{g,\Delta t}^m$ as the initial guess. If we denote by $\Delta\tilde{t}_n$ the time-step size that produces the desired absolute accuracy $\varepsilon$, where $\varepsilon$ is the specified accuracy, we find that $\Delta\tilde{t}_n$ can be estimated by

$$\Delta\tilde{t}_n = \Delta t_n \left( \frac{\varepsilon}{|\Delta\phi_{g,\Delta t}^m|} \right). \tag{4}$$

Using this information, we control the time-step size by the following algorithm in our work:

1. If $\Delta\tilde{t}_n > \Delta t_n$, then the current time step was successful, and for the next time step, we use $\Delta t_{n+1} = \min\{\Delta\tilde{t}_n, \alpha\Delta t_n\}$ with $\alpha = 1.5$.
2. If $\Delta\tilde{t}_n < \beta\Delta t_n$, where $\beta$ (<1.0) is a constant, then the current time step failed, and we have to go back two steps (i.e., n-2) and repeat them using a reduced time-step size given by $\Delta t_{n-1} = \Delta t_n = \Delta\tilde{t}_n$, referred to as time-step recovery.
3. Otherwise, we keep employing the same time-step size.

The coefficients $\alpha$ and $\beta$ were introduced to cause a smooth transition in the time-step sizes and were selected based on experience. The results show that increase of $\beta$ means increase of the number of time-step recoveries.

4.2 Space-time discretization stability analysis of the AFEN kinetics calculation

In some cases of the FDM analysis, the solutions of the difference and the differential equations can differ in the limit as $\Delta x$ and $\Delta t$ approach zero. It is worthwhile checking this potential in the AFEN kinetics calculation. To simplify the analysis, we use the one-group, one-dimensional time-dependent neutron diffusion equation in a homogeneous slab with an equally spaced mesh system as follows:

$$\frac{1}{V}\frac{\partial\phi^m(x,t)}{\partial t} = W\phi^m(x,t) + Q(t), \tag{5}$$

where

$V$ : neutron velocity,

$$W = D\frac{d^2}{dx^2} - (\Sigma_a - \nu\Sigma_f),$$

$Q(t)$ : time-dependent neutron source.

The final form of Eq.(5) with $Q(t) = 0$ after space-time discretization (number of spatial nodes =$N$ ) can be written as follows:

$$\left(\frac{[I]}{V\Delta t} - \theta[W]\right)\vec{\phi}(t_{n+1}) = \left(\frac{[I]}{V\Delta t} - (1-\theta)[W]\right)\vec{\phi}(t_n), \tag{6}$$

where

$[I]$ : identity matrix,

$\vec{\phi}(t) = \left(\phi^1(t), \phi^2(t), \cdots, \phi^N(t)\right)^T$,

$\theta$ = arbitrary parameter in the range $0 \le \theta \le 1$.

In AFEN kinetics calculation, the flux is written as a sum of analytic part $\phi_a^m(x,t)$ and polynomial part $\phi_p^m(x,t)$. By using analytic basis functions for $\phi_a^m(x,t)$ and linear order of polynomials for $\phi_p^m(x,t)$, the term $[W]\vec{\phi}$ can be written as follows:

$$[W]\vec{\phi} = \begin{pmatrix} -\frac{2D'}{\Delta x^2} - (\Sigma_a - \nu\Sigma_f) & \frac{D'}{\Delta x^2} & \cdots & 0 \\ \frac{D'}{\Delta x^2} & -\frac{2D'}{\Delta x^2} - (\Sigma_a - \nu\Sigma_f) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{2D'}{\Delta x^2} - (\Sigma_a - \nu\Sigma_f) \end{pmatrix} \begin{pmatrix} \phi_a^1 \\ \phi_a^2 \\ \vdots \\ \phi_a^N \end{pmatrix}$$

$$+ \begin{pmatrix} -\frac{2D}{\Delta x^2} - (\Sigma_a - \nu\Sigma_f) & \frac{D}{\Delta x^2} & \cdots & 0 \\ \frac{D}{\Delta x^2} & -\frac{2D}{\Delta x^2} - (\Sigma_a - \nu\Sigma_f) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{2D}{\Delta x^2} - (\Sigma_a - \nu\Sigma_f) \end{pmatrix} \begin{pmatrix} \phi_p^1 \\ \phi_p^2 \\ \vdots \\ \phi_p^N \end{pmatrix}, \tag{7}$$

where

$$D' = \frac{\Delta x^2 \kappa^2 D}{4 Sinh^2(\frac{\Delta x \kappa}{2})},$$

$$\kappa = \sqrt{\frac{\Sigma_a - \nu\Sigma_f}{D}}.$$

In AFEN, the polynomial part is obtained by applying a Galerkin scheme to Eq.(6). Due to the nonlinear characteristics, we introduce two ratio parameters between the total average flux and the average fluxes of the two components as follows:

$$\eta^m = \phi_a^m / \phi^m, \quad \mu^m = \phi_p^m / \phi^m, \tag{8}$$

where

$$\eta^m + \mu^m = 1.$$

Then, Eq.(7) can be rewritten as follows:

$$[W]\vec{\phi} = \begin{pmatrix} -\frac{2D^{1,''}}{\Delta x^2} - (\Sigma_a - \nu\Sigma_f) & \frac{D^{2,''}}{\Delta x^2} & \cdots & 0 \\ \frac{D^{1,''}}{\Delta x^2} & -\frac{2D^{2,''}}{\Delta x^2} - (\Sigma_a - \nu\Sigma_f) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{2D^{N,''}}{\Delta x^2} - (\Sigma_a - \nu\Sigma_f) \end{pmatrix} \begin{pmatrix} \phi^1 \\ \phi^2 \\ \vdots \\ \phi^N \end{pmatrix} \tag{9}$$

where

$$D^{m,''} = D'\eta^m + D\mu^m. \tag{10a}$$

This quantity can be rewritten as follows:

$$D^{m,''} = \frac{\Delta x^2 \kappa^2 D \eta^m}{4 Sinh(\frac{\Delta x \kappa}{2})^2} + D(1-\eta^m) = D\left[1 - \delta\eta^m\right], \tag{10b}$$

where

$$\delta = \left(1 - \frac{\Delta x^2 \kappa^2}{4 Sinh^2(\frac{\Delta x \kappa}{2})}\right).$$

The structure of Eq.(9) is the same to that of FDM except that the values of $D^{m,''}$ are

solution dependent. To simplify the analysis, we use the same $\eta^m$ for all the nodes. In an implicit scheme ($\theta = 1$) in time of the FDM space discretization, the instability condition is known as follows:

$$\frac{1}{1-\Delta t\omega_k} < -1 . \tag{11}$$

where $\omega_k$ are the eigenvalues of the coefficient matrix in Eq.(9). Eq.(11) can written as follows:

$$\Delta t > \frac{\Delta x^2}{4D(\delta\eta^m - 1)} . \tag{12}$$

Because the iteration between polynomial part and analytic part is nonlinear, it is not easy to analyze the stability exactly. But Eq. (12) can be used as a useful criterion for the stability analysis of space–time discretization. The value of $\delta$ varies between 0 and 1. In the worst case of $\delta$=1, occurring at a very large value of $\Delta x$, the instability condition becomes $\eta^m > 1.0 + \Delta x^2/(4D\Delta t)$. In practical cases, the value of $\eta^m$ usually lies between 0.95 and 1.05. Thus, the AFEN space-time discretization used in kinetics calculation should be stable in real problems that we usually encounter.

## 5. RESULTS AND DISCUSSION

Table I shows the computing time of the three-dimensional AFEN method with the acceleration methods applied to the well known IAEA-3D and LMW benchmark problems. It shows that the CMFD scheme is not so much effective as the CGR scheme and it does not show stable convergence in some cases of the problem. The major difference between the two results is the number of outer iterations. In formulation I, the two-node scheme reduces the number of outer iterations by a factor of about 5, while the CGR scheme reduces it by a factor of about 30-60. Although the number of inner iterations per outer iteration is larger in the CGR scheme, the time increase in the inner iteration is small compared with the time savings obtained from the reduced number of outer iterations. The whole-core AFEN/CMFD scheme does not converge in formulation I. In formulation II, the computing time of the whole-core AFEN/CMFD calculation is shorter than that of the two-node CMFD case because it does not require to solve two-node problems. Again, the CGR scheme is best in this formulation.

Table II shows the results of the adaptive time step size control applied to the 2-D TWIGL

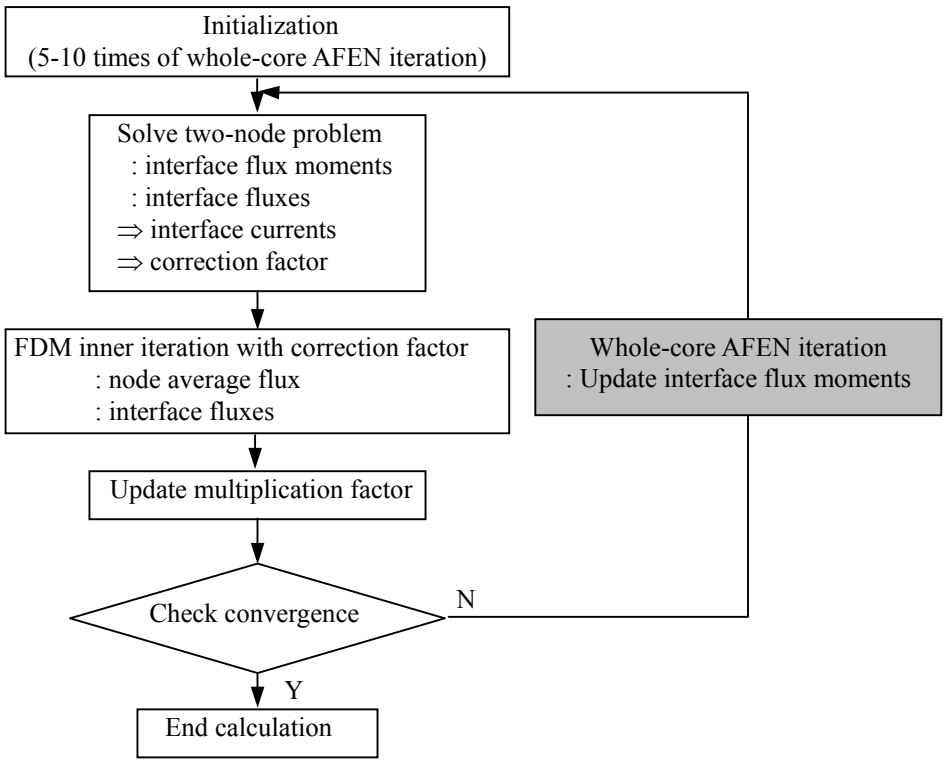
benchmark problem. If we set the value of $\beta$=1.0, the time step recovery is made at every instance when $\Delta\tilde{t}_n < \Delta t_n$. Cases 1, 4 and 7 show that this is not an efficient way in the aspect of computing time. However, if we decrease $\beta$ as in cases 3, 6, and 9, accuracy loss can occur in some cases. Thus, the value of $\beta$ should be selected carefully not to allow large errors. In the calculation of $\phi^m_{g,2\Delta t}$, $\phi^m_{g,\Delta t}$ can be used as the initial guess, which reduces the number of outer iterations by about 50 %.

## 6. CONCLUSIONS

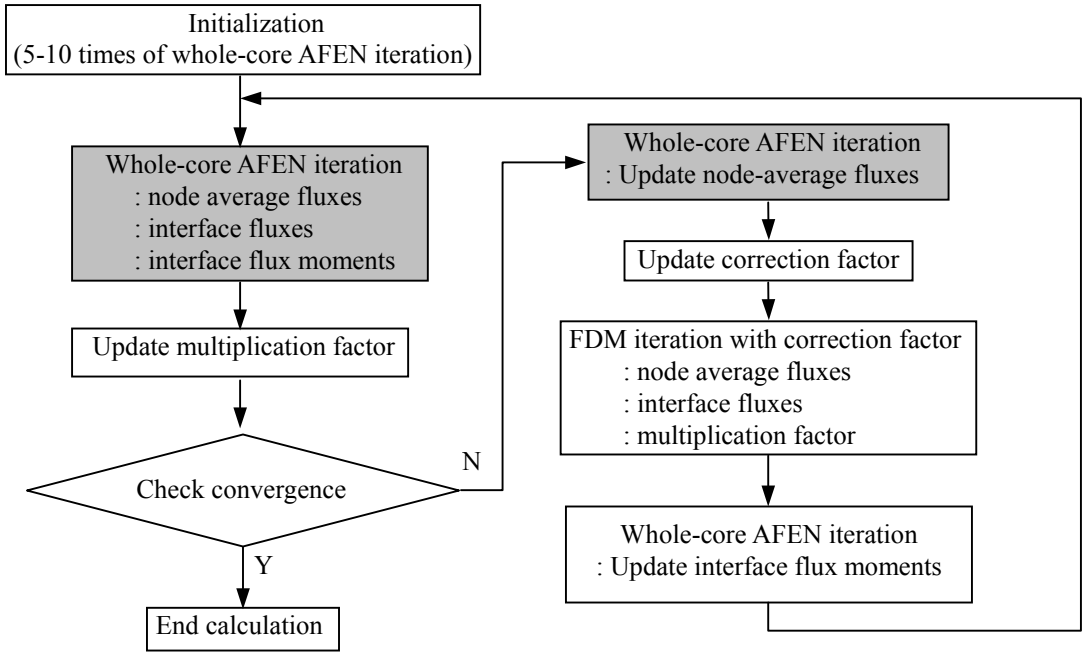
In this paper, we presented and compared the several features in the AFEN calculation currently implemented in the AFENX code. The results show that the CGR acceleration method is computationally more efficient than the CMFD acceleration method in three-dimensional geometry. The CMFD method is not efficient due to the edge fluxes or interface flux moments, which are directly connected to the correction factors and must be updated consistently. The CMFD method exhibits the stability problem in convergence in some cases, which is a general characteristic of the method. The CGR method uses only one correction factor which rebalances the reaction rates and shows stable convergence. In kinetics calculation, an adaptive time step size control method was used using an improved second-derivative-based error approximation. Also, we derived the stability criteria of the space-time discretization.

## REFERENCES


1. J. M. Noh and N. Z. Cho, "A New Approach of Analytic Basis Function Expansion to Neutron Diffusion Calculation," *Nucl. Sci. Eng*., **116**, 165 (1994).
2. N. Z. Cho and J. M. Noh, "Analytic Function Expansion Nodal method for Hexagonal Geometry*," Nucl. Sci. Eng*., **121**, 245 (1995).
3. D. S. Kim and N. Z. Cho, "Kinetics Calculation under Space-Dependent Feedbacks in Analytic Function Expansion Nodal Method via Solution Decomposition and Galerkin Scheme," *Nucl. Sci. Eng.,* **140**, 267 (2002).
4. Y. A. Chao, "Comments on the Treatment of Transverse Leakage in Advanced Nodal Codes for Hexagonal Nodes," *Nucl. Sci. Eng.*, **109**, 423 (1991).
5. R. D. Lawrence, "Progress in Nodal Methods for the Solution of the Neutron Diffusion and Transport Equations," *Progress in Nuclear Energy.*, **17**, 271 (1986).
6. N. Z. Cho and D. S. Kim, "AFENX: A Reactor Core Analysis Code Based on Analytic Function Expansion Nodal Method," in this Proceedings.
7. K. S. Moon, N. Z. Cho, J. M. Noh and S. G. Hong., "Acceleration of the Analytic Function Expansion Nodal Method by Two-Factor Two-node Nonlinear Iteration," *Nucl. Sci. Eng.*, **132**, 194 (1999).
8. D. S. Kim and N. Z. Cho, "Two-Node Nonlinear Iterative AFEN Method in Three-Dimensional Geometry," *Proceedings of the Korean Nuclear Society Autumn Meeting*, Seoul, Korea, October (1998).
9. D. S. Kim and N. Z. Cho, "Acceleration of Three-Dimensional AFEN Nodal Codes via Coarse Group Rebalance and Direct Matrix Inverse," *Proc. M&C, Reactor Physics and Environmental Analysis in Nuclear Applications*, Madrid, Spain, Vol.1, 168-174 (1999), and references therein.
10. Y. A. Chao, "Coarse Mesh Finite Difference Methods and Applications," *Proc. PHYSOR 2000, Advances in Reactor Physics and Mathematics and Computation into the Next Millennium,* Pittsburgh, U.S.A., May 2000, (CD-ROM).
11. S. W. Woo, N. Z. Cho and J. M. Noh, "The Analytic Function Expansion Nodal Method Refined with Transverse Gradient Basis Functions and Interface Flux Moments," *Nucl. Sci. Eng.*, **139**, 156 (2001).
12. N. Crouzet and P. J. Turinsky, "A Second-Derivative-Based Adaptive Time-Step Method for Spatial Kinetics Calculations," *Nucl. Sci. Eng.*, **123**, 206 (2001).




(a) Two-node CMFD scheme




(b) Whole-core AFEN/CMFD scheme

Fig. 1. CMFD schemes in AFENX with flux moments nodal unknowns

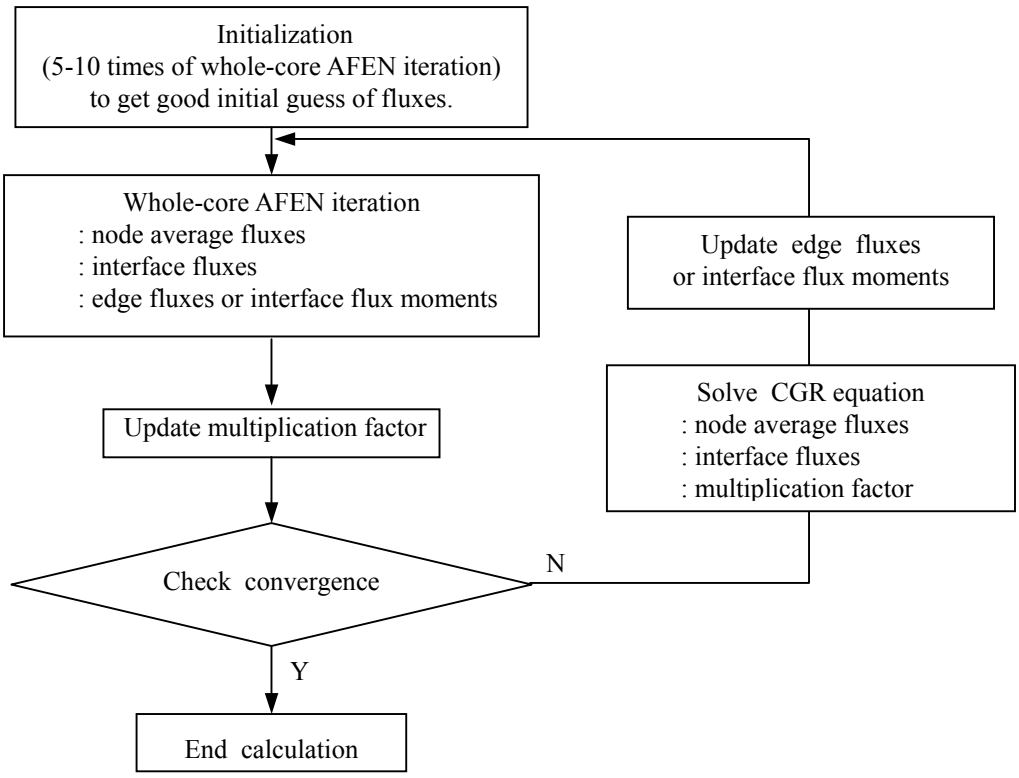



Fig. 2. Calculational flowchart of CGR acceleration scheme in AFENX.

Table I. Comparisons of the AFEN acceleration methods

(1/4 core, Relative convergence criteria : flux=1.E-5, HP C180)

| Formulation I* | | | |
|---|---|---|---|
| | | IAEA-3D problem (9x9x19 nodes) | LMW problem (6x6x10 nodes) |
| No acceleration scheme | | | |
| Net current version | Computing time | 126.09 s | 8.53 s |
| | Number of iterations (inner/outer) | 1/2086 | 1/577 |
| Partial current version | Computing time | 56.78 s | 3.63 s |
| | Number of iterations (inner/outer) | 1/2091 | 1/601 |
| Two-node CMFD acceleration scheme (net current version) | | | |
| Computing time | | 37.22 s | 2.42 s |
| Number of FDM iterations (inner/outer) | | 1/398 | 1/92 |
| Number of initial AFEN iterations | | 10 | 10 |
| CGR acceleration scheme (partial current version) | | | |
| Computing time | | 1.77 s | 0.24 s |
| Number of outer iterations | | 34 | 22 |
| Number of CGR rebalances | | 29 | 17 |
| Number of FDM iterations (inner/outer) in CGR rebalance | | 1/12 | 1/7 |
| Formulation II** | | | |
| | | IAEA-3D problem (9x9x19 nodes) | LMW problem (6x6x10 nodes) |
| No acceleration scheme | | | |
| | | | |
| Net current version | Computing time | 43.03 s | 2.73 s |
| | Number of iterations (inner/outer) | 1/1581 | 1/452 |
| Partial current version | Computing time | 19.93 s | 1.90 s |
| | Number of iterations (inner/outer) | 1/1712 | 1/487 |
| Two-node acceleration scheme (net current version) | | | |
| Computing time | | 28.03 | 2.14 s |
| Number of FDM iterations (inner/outer) | | 1/500 | 1/161 |
| Number of initial AFEN iterations | | 20 | 5 |
| Whole-core AFEN/CMFD acceleration scheme (net current version) | | | |
| Computing time | | 7.32 s | 0.56 s |
| Number of FDM iterations (inner/outer) | | 10/67 | 5/26 |
| Number of whole-core AFEN iterations | | 143 | 61 |
| CGR acceleration scheme (partial current version) | | | |
| Computing time | | 1.42 s | 0.21 s |
| Number of outer iterations | | 31 | 24 |
| Number of CGR rebalances | | 31 | 24 |
| Number of FDM iterations (inner/outer) in CGR rebalance | | 1/12 | 1/7 |

* Nodal unknowns in each group: 1 node average flux, 6 interface average fluxes and 12 edge fluxes

** Nodal unknowns in each group: 1 node average flux, 6 interface average fluxes and 12 interface flux moments

Table II. Adaptive time step size control in AFEN kinetics calculation

(2-D TWIGL problem, $0 \le t \le 0.5\,\text{sec}$ , flux convergence criteria: 1.E-5, Zion-1000)

| Calculation with fixed time step size | | | | |
|---|---|---|---|---|
| Case | $\Delta t$ | Number of time steps | $P^*_{acc} / P(0)$ (% error) | Computing time (s) |
| Ref | 0.0001 | 5000 | 0.897161 | 6.68 |
| 1 | 0.001 | 500 | 0.897129 (-0.004) | 1.46 |
| 2 | 0.005 | 100 | 0.897007 (-0.017) | 0.70 |
| 3 | 0.01 | 50 | 0.896900 (-0.029) | 0.55 |
| 4 | 0.02 | 25 | 0.896786 (-0.042) | 0.40 |
| 5 | 0.05 | 10 | 0.897597 (0.049) | 0.24 |
| Calculation with adaptive time step size | | | | |
| Case | $\varepsilon$ | $\beta$ | $N_1^{**} / N_2^{***}$ | $P^*_{acc} / P(0)$ (% error) | Computing time (s) |
| 1 | 0.0001 | 1.0 | 114 (436) | 0.897158 (0.000) | 1.67 |
| 2 | | 0.8 | 108 (179) | 0.897155 (-0.001) | 0.89 |
| 3 | | 0.3 | 82 (128) | 0.897123 (-0.004) | 0.81 |
| 4 | 0.0005 | 1.0 | 48 (206) | 0.897215 (0.006) | 1.92 |
| 5 | | 0.8 | 47 (91) | 0.897249 (0.010) | 0.86 |
| 6 | | 0.3 | 38 (65) | 0.897326 (0.018) | 0.53 |
| 7 | 0.001 | 1.0 | 33 (143) | 0.897336 (0.019) | 1.92 |
| 8 | | 0.8 | 32 (67) | 0.897421 (0.029) | 0.86 |
| 9 | | 0.3 | 16 (26) | 0.894804 (-0.263) | 0.53 |

* $P_{acc} = \int_0^{0.5} P(t)\, dt$

** Number of effective time steps on which the final results are based

*** Number of time steps calculated