Transactions of the Korean Nuclear Society Spring Meeting
Jeju, Korea, May 23-24, 2019

A Response Matrix Formulation of the Refined AFEN Method with Interface Flux Moments in the Two-dimensional Hexagonal Geometry

Jae Man Noh
Korea Atomic Energy Research Institute
jmnoh@kaeri.re.kr

1. Introduction

A nodal method based on the AFEN method [1-6] in the hexagonal geometry has been developed to improve the computational efficiency of the higher order finite element method (FEM) installed in CAPP.[7] As an effort to achieve this purpose, an AFEN equivalent method based on the single-node nonlinear finite difference Method (FDM) Response Matrix has been tried but it cannot provide a numerically stable solution as noted in Ref. 8. Several attempts, including recommended in Ref. 8, to improve the numerical stability have been made but they were in vain. The FDM response matrix methods have not been successful as long as they are fundamentally based on the single node AFEN solution.

As noted in Ref. 8, the one-node based nonlinear FDM is more suitable for the response matrix method than the two-node based one. However, it failed to provide a numerically stable solution. To ensure numerical stability, the direct response matrixing of the refined AFEN with interface flux moments[4] has been adopted in this paper rather than detouring via a two-node nonlinear FDM response matrix equivalent to the AFEN. In general, a response matrix method has the advantage that in the calculation of the unknowns and their matrix coefficients of a node, the domain can be confined within the node independently of its neighboring nodes. It calculates the six outgoing interface partial currents of each node at each inner iteration step by solving the node with the boundary conditions of six incoming interface partial currents. Considering that one interface is shared by two nodes, it can be noted that three interface unknowns are equivalently calculated per each single node calculation. On the other hand, the original AFEN method[4] determines only one interface unknown by solving a two-node problem by imposing flux and current continuity conditions across the interface between the two nodes. Therefore, assuming the same number of outer iterations is involved to achieve the same accuracy, the response matrix method becomes much more efficient than the original AFEN method.

The refined form of AFEN which uses the interface flux moments instead of the corner points as nodal unknowns fits more to the concept of the response matrix which can be defined as a response (i.e., outgoing partial currents) of a single node to the input (i.e., incoming partial currents). It is difficult to define (not a flux but) a partial current at a corner point of a node.

This paper presents the results of a two-dimensional reactor core analysis using the proposed method. The main numerical performance will remain the same even if it is expanded to three dimensions.

2. Methodology

Deriving the AFEN Response Matrix which expresses the outgoing interface partial currents into the incoming interface partial currents has two steps for convenience. In the first step, the AFEN single node is solved to obtain the relationship between the interface fluxes and the interface currents. In the second step, the response matrix is derived by replacing the interface fluxes and interface currents in this relationship with the incoming and outgoing interface partial currents.

The first step is identical to that described in Ref. 8, but it is repeated almost as it is in the reference for the sake of readability of this paper.

2.1 AFEN Solution of Single-Node Problem [4,8]

2.1.1 Intranodal Flux Expansion

Solving the single node problem with interface current and current moment boundary conditions by the AFEN methods starts from expending the intranodal flux distribution into the analytic basis functions with and without transverse-direction linear functions.

$$\boldsymbol{\phi}(x,y) = \boldsymbol{\varphi}(x,y) + \boldsymbol{\varphi}(u,v) + \boldsymbol{\varphi}(\rho,q) \tag{1}$$

where

$$\boldsymbol{\varphi}(x,y) = \sinh\left(\sqrt{\boldsymbol{\Lambda}}x\right)\left(\mathbf{A}_{x0} + \mathbf{A}_{x1}\,y\right) + \cosh\left(\sqrt{\boldsymbol{\Lambda}}x\right)\left(\mathbf{B}_{x0} + \mathbf{B}_{x1}\,y\right) \tag{2}$$

$$\boldsymbol{\Lambda} = \mathbf{D}^{-1}\boldsymbol{\Sigma} \tag{3}$$

and $\mathbf{D}$ and $\boldsymbol{\Sigma}$ are the diffusion coefficient and crosssection matrixes, respectively and **A**'s and **B**'s are expansion coefficients. Here, (x, y), (u,v) and (p,q) are the three coordinates in Fig. 1 introduced for convenience.

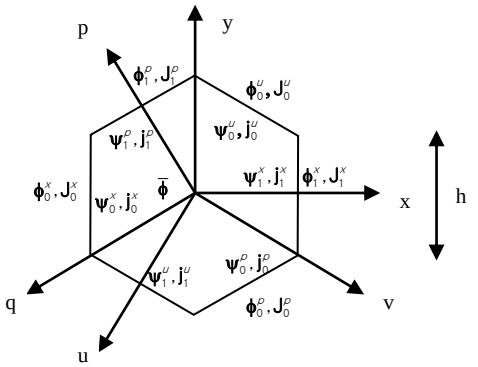

Fig. 1. Coordinate systems and nodal unknowns

Note that this flux expansion has twelve terms with one coefficient each and all of them completely satisfy the diffusion equation for the node. Of course, both the basis functions and the coefficients of this expansion are square matrices with the number of energy groups as its order. However, thanks to the matrix function theory, they can be treated like scalar as long as they are functions of Λ.[9,10]

The average flux of the node is defined from this flux expansion as follows.

$$\bar{\phi} = \frac{8\sqrt{3}}{9h^2}\int_0^{\frac{\sqrt{3}}{2}h}\int_0^{-\frac{\sqrt{3}}{3}x+h} \phi(x,y)\,dy\,dx \quad . \tag{4}$$

The interface fluxes and the flux moments e.g., at the x1 interface are respectively defined by

$$\phi_{x1} = \frac{1}{h}\int_{-\frac{h}{2}}^{\frac{h}{2}} \phi(\frac{\sqrt{3}}{2}h, y)dy \tag{5}$$

$$\psi_{x1} = \frac{1}{h}\int_{-\frac{h}{2}}^{\frac{h}{2}} w(y)\,\phi(\frac{\sqrt{3}}{2}h, y)dy \tag{6}$$

Here, w(y) is the weighting function. Two types of w(y), i.e., the step function and the linear function of y are used as the weighting function.

$$w(y) = \begin{cases} -1 \ when \ y<0 \\ 1 \ \ when \ y\ge 0 \end{cases} \quad and \quad w(y)=y\,. \tag{7}$$

If the step function is used, it is equivalent to the case where an interface is cut in half and the continuous condition of flux and current is applied for each half of the interface.

Strictly speaking, when applying equivalence theory, the interface fluxes and moments in Eqs. (5) and (6) are homogeneous ones. They are multiplied by the discontinuity factors to yield the heterogeneous ones. However, for simplicity of derivation, we ignore the discontinuity factors at this moment. In implementing, of course, the discontinuity factors are involved.

Further, the interface current and the current moment at the example interface are consistently defined by

$$J^{in}_{x1} = \frac{D}{h}\int_{-\frac{h}{2}}^{\frac{h}{2}} \frac{\partial}{\partial x}\phi(x,y)dy\Big|_{x=\frac{\sqrt{3}}{2}h} \tag{8}$$

$$j^{in}_{x1} = \frac{D}{h}\int_{-\frac{h}{2}}^{\frac{h}{2}} w(y)\frac{\partial}{\partial x}\phi(x,y)dy\Big|_{x=\frac{\sqrt{3}}{2}h} \tag{9}$$

2.1.2 AFEN Solution of Single-Node Problem

Solving the single node problem in Fig. 1 to obtain the intranodal flux distribution means expressing 12 coefficients of the flux expansion Eq. (1) in terms of 6 interface currents and 6 interface current moments. This problem seems to involve inversing a 12x12 matrix. However, the decoupling transformation of Ref. [4] simplifies it to a problem of inversing several smaller matrixes.

This transformation transforms both the flux expansion coefficients and the nodal unknowns. First, the parity transformation transforms the nodal unknowns into their even and odd forms e.g., in the x-direction:

$$\omega_{xe} = \frac{\phi_{x1}+\phi_{x0}}{2} - \bar{\phi}, \qquad \omega_{xo} = \frac{\phi_{x1}-\phi_{x0}}{2} \tag{10}$$

$$\sigma_{xe} = \frac{\psi_{x1}+\psi_{x0}}{2}, \qquad \sigma_{xo} = \frac{\psi_{x1}-\psi_{x0}}{2} \tag{11}$$

Applying symmetric thinking, the even and the odd forms of the interface currents and current moments are defined as follows:

$$\eta^{in}_{xe} = \frac{J^{in}_{x1}+J^{in}_{x0}}{2}, \qquad \eta^{in}_{xo} = \frac{J^{in}_{x1}-J^{in}_{x0}}{2} \tag{12}$$

$$\varsigma^{in}_{xe} = \frac{j^{in}_{x1}+j^{in}_{x0}}{2}, \qquad \varsigma^{in}_{xo} = \frac{j^{in}_{x1}-j^{in}_{x0}}{2} \tag{13}$$

Now, the direction transformation transforms the coefficients of the expansion flux and the nodal unknowns further as shown below.

$$C_{k\theta} = \frac{C_{xk}+C_{uk}+C_{pk}}{3},\ C_{k\varepsilon} = \frac{2C_{xk}-C_{uk}-C_{pk}}{3},\ C_{k\chi} = \frac{C_{uk}-C_{pk}}{3} \tag{14}$$

$$\omega_{s\theta} = \frac{\omega_{xs}+\omega_{us}+\omega_{ps}}{3},\omega_{s\varepsilon} = \frac{2\omega_{xs}-\omega_{us}-\omega_{ps}}{3},\omega_{s\chi} = \frac{\omega_{us}-\omega_{ps}}{3} \tag{15}$$

$$\sigma_{s\theta} = \frac{\sigma_{xs}+\sigma_{us}+\sigma_{ps}}{3},\ \sigma_{s\varepsilon} = \frac{2\sigma_{xs}-\sigma_{us}-\sigma_{ps}}{3},\sigma_{s\chi} = \frac{\sigma_{us}-\sigma_{ps}}{3} \tag{16}$$

where coefficient letter C is A or B, interface index *k* is 0 or 1 and even-odd index *s* is *e* or *o*.

Expressing the transformed unknowns in terms of the transformed expansion coefficients, we can realize that the original 12x12 matrix equation is decoupled with four 2x2 matrix equations and four scalar equations. In particular, two of the four 2x2 matrixes are in the relationship of similarity transformation to the other two. Therefore, solving a single-node problem simply involves finding the inverses of two 2x2 matrixes and four scalars, for example:

$$\begin{bmatrix}\omega_{o\chi}\\ \sigma_{e\varepsilon}\end{bmatrix} = T\begin{bmatrix}A_{0\chi}\\ B_{1\varepsilon}\end{bmatrix} \text{ and } \begin{bmatrix}\eta^{in}_{o\chi}\\ \varsigma^{in}_{e\varepsilon}\end{bmatrix} = DM\begin{bmatrix}A_{0\chi}\\ B_{1\varepsilon}\end{bmatrix} \tag{17}$$

where M and T are 2x2 matrixes with elements of Λ matrix functions.

By eliminating the coefficient vector, the interface fluxes and flux moments can be expressed in terms of the interface currents and current moments, which are boundary conditions e.g., as follows:

$$\begin{bmatrix}\omega_{o\chi}\\ \sigma_{e\varepsilon}\end{bmatrix} = TM^{-1}D^{-1}\begin{bmatrix}\eta^{in}_{o\chi}\\ \varsigma^{in}_{e\varepsilon}\end{bmatrix} \tag{18}$$

where TM^{-1} is also a matrix function system of Λ. It is noted that any matrix functions of a matrix can be evaluated easily, e.g., by diagonalizing the matrix using its eigensystem. However, it will be soon realized that this kind of luck is not maintained in the final response matrix equation.

2.2 AFEN Response Matrix

The response matrix that computes the output, which are the outgoing interface partial currents and moments out of a node, from the input, which are the incoming interface partial currents and moments into the node, is derived by noting that the interface partial currents and moments at the interface *s* in the direction *d* are expressed in terms of the interface fluxes, flux moments, currents and current moments given the equations from (6) to (9).

$$P^{f}_{dk} = \frac{J^{f}_{dk}}{2} + \frac{\phi_{dk}}{4} \tag{19}$$

$$p^{f}_{dk} = \frac{j^{f}_{dk}}{2} + \frac{\psi_{dk}}{4} \tag{20}$$

where flow direction index *f* is *in* or *out*, direction index *d* is *x*, *u*, or *p*, and *k* is the interface index annotated just

after Eq. 16. Then, the interface fluxes and currents are equivalently given by

$$J^{in}_{dk} = P^{in}_{dk} - P^{out}_{dk} \qquad \phi_{dk} = 2\left(P^{in}_{dk} + P^{out}_{dk}\right) \tag{21}$$

$$j^{in}_{dk} = p^{in}_{dk} - p^{out}_{dk} \qquad \psi_{dk} = 2\left(p^{in}_{dk} + p^{out}_{dk}\right) \tag{22}$$

Since the relationships (19) and (20) are linear and the parity and direction transformations explained in the previous section are also linear, the partial currents and moments shall have their transformed forms with respect to the both transformations and these forms shall have the relationships corresponding to those of Eq. (20) and (21) as follows,

$$\eta^{in}_{o\chi} = Z^{in}_{o\chi} - Z^{out}_{o\chi} \qquad \omega_{o\chi} = 2\left(Z^{in}_{o\chi} + Z^{out}_{o\chi}\right) \tag{23}$$

$$\varsigma^{in}_{e\varepsilon} = z^{in}_{e\varepsilon} - z^{out}_{e\varepsilon} \qquad \sigma_{e\varepsilon} = 2\left(z^{in}_{e\varepsilon} + z^{out}_{e\varepsilon}\right) \tag{24}$$

Substituting these relationships into Eq. (18) and solving for the transformed outgoing partial currents, we finally obtain the response matrix in the transformed system as follows,

$$\begin{bmatrix} Z^{out}_{o\chi} \\ z^{out}_{e\varepsilon} \end{bmatrix} = Q \begin{bmatrix} Z^{in}_{o\chi} \\ z^{in}_{e\varepsilon} \end{bmatrix} \tag{25}$$

where $\mathbf{Q} = -(2\mathbf{I} + \mathbf{TM}^{-1}\mathbf{D}^{-1})^{-1}(2\mathbf{I} - \mathbf{TM}^{-1}\mathbf{D}^{-1})$. Unfortunately, the matrix, $2\mathbf{I} + \mathbf{TM}^{-1}\mathbf{D}^{-1}$ is not a matrix function of Λ anymore because it contains the diffusion coefficient matrix D. Therefore, the inversion of this matrix becomes a full inversion of a 2G X 2G matrix.

Note that the interface partial currents and moments can be easily transformed into their linearly transformed partners and vice versa. Once the interface incoming partial currents and are given for a node, the interface outgoing partial currents and moments can be calculated by, e.g., Eq. (18). Then, these outgoing partial currents become the partial currents incoming into its neighboring nodes. This provides an iterative process to solve the whole entire core eigenvalue problem through the well-known inner-outer iteration.

As mentioned in Ref. [4], the inverse of M is singular when one of the eigenvalues of the crosssection matrix is very small. This singularity is removed in the manner described in the reference.

2.3 Assistive Techniques

2.3.1. RGB Sweeping Scheme[1][8]

As described above, the response matrix calculations are performed only within a single node regardless of neighboring nodes. Therefore, these calculations are carried out by sequentially moving from one node to another. In this case, it is advantageous to sweep the nodes by dividing them into red (R), green (G), and blue (B) nodes as shown in Fig. 2, like checkerboard sweeping for a rectangular node core.

This kind of iteration schemes is known to be good in convergence and stability due to geometrical balance. It further enhances the advantage in parallel-computing that the response matrix method has already.

In addition, the memory required is saved by storing inputs of the response calculation, i.e., incoming partial currents and outputs, i.e., outgoing partial currents in the same storage. This is because the outgoing partial currents resulting from previous two other color types of node calculations automatically become incoming partial currents for the third kind of node calculations.

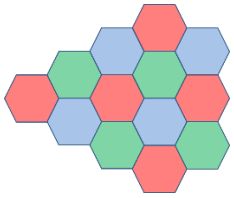

Fig. 2. RGB sweeping scheme

2.3.2. Nonlinear FDM Acceleration

As described in Ref. [8], if we define the two nonlinear correction factors per interface, the FDM response matrix corresponding to the AFEN one in the equation (25) is given at the interface *s* of a node by

$$P^{out}_{sg} = \frac{\sqrt{3}h - 4(D_g + \widetilde{D}_{sg})}{\sqrt{3}h + 4(D_g + \widetilde{D}_{sg})} P^{in}_{sg} + \frac{2(D_g - \widetilde{D}_{sg})}{\sqrt{3}h + 4(D_g + \widetilde{D}_{sg})} \bar{\phi}_g \tag{26}$$

for the partial currents and by

$$p^{out}_{sg} = \frac{\sqrt{3}h - 4(D_g + \widetilde{d}_{sg})}{\sqrt{3}h + 4(D_g + \widetilde{d}_{sg})} p^{in}_{sg} + \frac{2(D_g - \widetilde{d}_{sg})}{\sqrt{3}h + 4(D_g + \widetilde{d}_{sg})} \bar{\psi}_g \tag{27}$$

for the partial current moments. Here, all the notations have the same meanings as in Ref. [8].

In order to accelerate the overall computational time, we may consider an iteration scheme that repeats sequentially a certain number of the FDM outer iterations with Eq. (26) and (27) and the other number of the AFEN iterations with Eq. (25). Of course, the nonlinear correction factors $\widetilde{D}_{sg}, \widetilde{d}_{sg}$ are calculated at the every exit of AFEN iterations to make two types of iterations equivalent to each other as follows,

$$\widetilde{D}_{sg} = -\frac{\sqrt{3}hJ^{out}_{sg} + 2D_g(\phi_{sg} - \bar{\phi}_g)}{2(\phi_{sg} + \bar{\phi}_g)} \tag{28}$$

$$\widetilde{d}_{sg} = -\frac{\sqrt{3}hj^{out}_{sg} + 2D_g(\psi_{sg} - \bar{\psi}_g)}{2(\psi_{sg} + \bar{\psi}_g)} \tag{29}$$

This iteration is different from the nonlinear FDM iteration described in Ref. [8]. The former updates partial currents consistently in both the FDM and AFEN iterations but the latter updates them only in the FDM iteration while AFEN is used only to evaluate the nonlinear correction factors.

3. Numerical Results and Discussion

In order to verify the AFEN response matrix method developed in this study, we solved the VVER440 benchmark problem described in Ref. [4]. The results are compared in Table I with those of the conventional form of the refined AFEN method in Ref. [4] and those of the FDM response matrix method without any nonlinear correction factors. The boundary condition is the zero incoming partial current for the two response matrix methods and the zero flux for the conventional form of the AFEN method. Therefore, the k-effective difference between two AFEN methods is purely due to the different boundary conditions. This difference is

[1] This part repeats Ref. [8] with some modifications

slightly larger because a small node size makes this problem sensitive to the boundary condition.

The refined AFEN response method significantly outperforms the conventional AFEN method numerically. The number of outer iterations required to obtain the same computational accuracy is less than half that required for the conventional AFEN method. Because a single outer iteration step for the response matrix method is more efficient than that for the original AFEN method as explained in Introduction, the whole computing time consumption is significantly low in the response matrix method.

The acceleration scheme presented in the Section 2.3.2 was also tested against the VVER440 problem. Many combinations of the number of the FDM iteration and that of the AFEN iteration were tried but none of them could accelerate the AFEN response matrix although some resulted in a numerically stable scheme. It is recalled that the acceleration scheme proposed in Ref. [8] even fails to give a stable scheme. As seen in Table I, the original FDM requires more number of outer iterations to achieve the same accuracy compared with the AFEN response method. This means that the diagonal dominance of the FDM response matrix is smaller than that of the AFEN one. In general, the nonlinear FDM with correction factors makes the diagonal dominance even worse compared to the FDM without correction factors. Awkward idea to accelerate the system with higher diagonal dominance using the system with lower diagonal dominance may be the reason why the nonlinear FDM acceleration is unsuccessful. If this is true, any nonlinear FDM schemes including ones based on even the two-node problem will fail.

Table I. Results of the VVER440 problem (1/6 core)

	K-eff(1)	# of iterations(2)	Computing Time (msec)(3)
AFEN RM	1.009646	460	70
Original AFEN	1.008632	1,111	192
FDM	1.018224	660	30

(1) The k-effective value calculated by a FDM fine mesh calculation with the zero-flux boundary condition is 1.008636. This boundary condition is consistent with the original AFEN (but not with the AFEN response matrix (RM)).
(2) # of outer iterations required to achieve a less than 10^{-7} accuracy in both k-effective and node-wise sources
(3) Measured in a PC with Intel® Core™ i7-4930K CPU

4. Conclusions

The nonlinear FDM-based AFEN response matrix method in the hexagonal geometry proposed in Ref. [8] to improve efficiency of the CAPP code failed to give a numerically stable scheme. Therefore, giving up any nonlinear FDM response matrix methods equivalent to the AFEN, the direct response matrix formulation of the refined AFEN with interface flux moments has been developed in this paper. In this method, the partial current moments are used as nodal unknowns instead of the corner point fluxes.

This method was tested against the VVER440 benchmark problem and verified that it could provide much faster computing speed compared to the original refined AFEN method. In addition, it was found that the diagonal dominance of this method was larger than those of the FDM and the original refined AFEN formulation by judging by a reduced number of outer iterations to reach the same accuracy. This was presented as a cause of poor performance of the nonlinear FDM schemes in accelerating the AFEN response matrix.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (No. 2017M2A8A1014757).

References

1. J. M. NOH and N. Z. CHO, "A New Approach of Analytic Basis Function Expansion to Neutron Diffusion Nodal 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. N. Z. CHO, Y. H. KIM, and K. W. PARK, "Extension of Analytic Function Expansion Nodal Method to Multigroup Problems in Hexagonal-Z Geometry," Nucl. Sci. Eng., 126, 35 (1997).
4. 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).
5. N. Z. CHO and J. J. LEE, "Analytic Function Expansion Nodal (AFEN) Method in Hexagonal-z Three Dimensional Geometry for Neutron Diffusion Calculation," J. Nucl. Sci. and Tech., 43, 1320 (2006).
6. K. S. MOON, N. Z. CHO and J. M. NOH, "Acceleration of the Analytic Function Expansion Nodal Method by Two-Factor Two-Node Nonlinear Iteration," Nucl. Sci. Eng., 132, 193 (1999).
7. H. C. LEE, T. Y. HAN, C. K. JO and J. M. NOH, "Development of the HELIOS/CAPP code system for the analysis of pebble type VHTR cores," Ann. Nucl. Eng., 71, 120 (2014).
8. J. M. NOH, " An AFEN Equivalent Hexagonal Nodal Method based on the Single-Node Nonlinear FDM Response Matrix," Trans. KNS Spring Mtg Jeju, Korea, October 25-26, 2018.
9. J. M. NOH, et al., "A General Approach to Multigroup Extension of the Analytic Function Expansion Nodal Method Based on Matrix Function Theory," Proc. 1996 Joint Intl. Conf. Mathematical Methods and Super Computing for Nuclear Applications, Saratoga Springs, New York, Octorber 6-10, 1997, Vol. 1, p. 144, American Nuclear Society (1997).
10. N. DUNFORD and J. T. SCHWARTZ, Linear Operators Part I: General Theory, Interscience Publishers, New York, 1971.