PHYSOR 2004 -The Physics of Fuel Cycles and Advanced Nuclear Systems: Global Developments
Chicago, Illinois, April 25-29, 2004, on CD-ROM, American Nuclear Society, Lagrange Park, IL. (2004)

Numerical Convergence Analysis of the CMFD Method for the Three-Dimensional Two-Group Diffusion Problem

Deokjung Lee[1], Thomas J. Downar[1], and Yonghee Kim[2]
[1]*Purdue University, West Lafayette, IN 47907-1290 USA*
[2]*Korea Atomic Energy Research Institute, Yusung, Daejeon 305-353, Korea*

The convergence of the CMFD method was examined for three-dimensional, two-group diffusion problems. Two test problems were investigated: a homogeneous 3-D 2-G model eigenvalue problem and the NEACRP LWR transient benchmark problem. A One-Node (1-N) kernel was implemented into PARCS for both steady state and transient problems which complemented the existing Two-Node (2-N) kernel. The convergence rates of both the one and two node kernels were measured numerically with emphasis on the convergence of current correction factors (CCFs). This paper presents the numerical test results and analyzes them in the context of the Fourier analysis with simple model problems presented in previous publications by the authors.

KEYWORDS: CMFD, convergence analysis, PARCS, CCF, convergence rate, under-relaxation

1. Introduction

For several years the nonlinear Coarse Mesh Finite Difference (CMFD) algorithm [1] has been widely used for the acceleration of both diffusion and transport calculations [2-5]. However, there has not been a methodical convergence analysis of the method for practical LWR analysis. Recently, Lee et al. [6,7] and Cho et al. [8] reported an analytic study of the CMFD convergence characteristics by linearizing the nonlinear algorithm and applying Fourier analysis for the diffusion and transport calculations, respectively. However, their analyses were limited to simple model problems, i.e., one-dimensional one-group fixed source problem, primarily because of the complexity of the analytic derivation of the spectral radius for multi-dimensional multi-group problems. The convergence study of the CMFD method here was performed numerically for the practical three-dimensional two-group steady state and transient problems.

2. Methodology

2.1 Three-Dimensional Two-Group Neutron Diffusion Equation

The three-dimensional steady state two-group neutron diffusion equation can be written in Cartesian coordinates as:

$$\begin{aligned} -D_1\left(\frac{d^2}{dx^2}+\frac{d^2}{dy^2}+\frac{d^2}{dz^2}\right)\phi_1(x,y,z)+\Sigma_{r1}\phi_1(x,y,z) &= \frac{1}{k_{eff}}\left(\nu\Sigma_{f1}\phi_1(x,y,z)+\nu\Sigma_{f2}\phi_2(x,y,z)\right) \\ -D_2\left(\frac{d^2}{dx^2}+\frac{d^2}{dy^2}+\frac{d^2}{dz^2}\right)\phi_2(x,y,z)+\Sigma_{r2}\phi_2(x,y,z) &= \Sigma_{12}\phi_1(x,y,z) \end{aligned} \quad (1)$$

where all the notations are standards [2]. The transverse integration of Eq. (1) results in the following one-dimensional diffusion equation:

$$-\mathbf{D}\frac{d^2}{du^2}\boldsymbol{\varphi}_u(u)+\mathbf{A}\boldsymbol{\varphi}_u(u)=-\mathbf{L}_u(u) \tag{2}$$

where $\lambda \equiv 1/k_{eff}$, $\boldsymbol{\varphi}_u(u)=\left[\phi_{1u}(u) \quad \phi_{2u}(u)\right]^T$, $\mathbf{L}_u=\left[L_{1u}(u) \quad L_{2u}(u)\right]^T$, and $\phi_{gu}(u)$ is the transverse integrated one-dimensional flux, $L_{gu}(u)$ is a u-directional transverse leakage, g=1,2. and $\mathbf{D}$, $\mathbf{A}$ are 2x2 matrices.

The derivation of the two-node kernel can be found elsewhere [2], which was already implemented in the PARCS code. With the solutions to the two-node problems, the CCFs can be updated by

$$\hat{D}_g=-\frac{J_g^{nodal}+\tilde{D}_g\left(\bar{\phi}_g^{\,r}-\bar{\phi}_g^{\,l}\right)}{\bar{\phi}_g^{\,r}+\bar{\phi}_g^{\,l}} \tag{3}$$

where J_g^{nodal} is the interface current from the two-node solution, $\tilde{D}_g$ is a standard finite difference coupling coefficients, and $\hat{D}_g$ is the CCF.

2.2 One-Node ANM Kernel

The general solution of Eq. (2) can be written for the one-node kernel derivation [9]:

$$\boldsymbol{\varphi}_u(u)=\begin{bmatrix} r & s \\ 1 & 1 \end{bmatrix}\begin{bmatrix} a_{u1}sn(\kappa u)+a_{u2}cn(\kappa u) \\ a_{u3}\sinh(\mu u)+a_{u4}\cosh(\mu u) \end{bmatrix}+\begin{bmatrix} a_{01}^u \\ a_{02}^u \end{bmatrix}-\mathbf{A}^{-1}\left(\mathbf{b}_1^u f_1(\xi)+\mathbf{b}_2^u f_2(\xi)\right) \tag{4}$$

where $a_{up}; p=1,2,3,4$ and $a_{0g}^u; g=1,2$ are 6 unknown coefficients to be determined per direction. The constraints necessary to determine a total of 18 unknown coefficients for each node are the 4 consistency conditions of the node average flux, 2 nodal balance equations, and 12 partial incoming currents $j_{ug,s}^{in}$; $u=x,y,z$, $g=1,2$ and $s=l/r$

- 4 consistency conditions of the node average flux:

$$\bar{\boldsymbol{\varphi}}_x=\bar{\boldsymbol{\varphi}}_y \text{ and } \bar{\boldsymbol{\varphi}}_y=\bar{\boldsymbol{\varphi}}_z \tag{5}$$

- 2 nodal balance equations:

$$\frac{\mathbf{J}_x^r-\mathbf{J}_x^l}{h_x}+\frac{\mathbf{J}_y^r-\mathbf{J}_y^l}{h_y}+\frac{\mathbf{J}_z^r-\mathbf{J}_z^l}{h_z}+\mathbf{A}\bar{\boldsymbol{\varphi}}_x=0 \tag{6}$$

- 12 partial incoming currents:

$$\mathbf{j}_{ul,r}^{in}=\frac{1}{4}\boldsymbol{\varphi}_u(\mp\frac{h_u}{2})\pm\frac{1}{2}\mathbf{J}_u(\mp\frac{h_u}{2});\ u=x,y,z \tag{7}$$

The node average flux consistency conditions and nodal balance equations can be rewritten more

compactly as:

$$\begin{bmatrix} \bar{\phi}_{1x} - \bar{\phi}_{1y} \\ \bar{\phi}_{2x} - \bar{\phi}_{2y} \\ \bar{\phi}_{1y} - \bar{\phi}_{1z} \\ \bar{\phi}_{2y} - \bar{\phi}_{2z} \\ NB(g=1) \\ NB(g=2) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{8}$$

where "NB" means "Nodal Balance". This can be expressed using the coefficients of the general solution as:

$$\mathbf{T}\,\mathbf{a} + \mathbf{W}\,\mathbf{a}_e + \mathbf{V}\,\mathbf{b}_2 = \mathbf{0} \tag{9}$$

where $\mathbf{a} = \begin{bmatrix} a_{x01} & a_{x02} & a_{y01} & a_{y02} & a_{z01} & a_{z02} \end{bmatrix}^T$, $\mathbf{a}_e = \begin{bmatrix} a_{x2} & a_{x4} & a_{y2} & a_{y4} & a_{z2} & a_{z4} \end{bmatrix}^T$, $\mathbf{b}_2 = \begin{bmatrix} b_{x12} & b_{x22} & b_{y12} & b_{y22} & b_{z12} & b_{z12} \end{bmatrix}^T$, and $\mathbf{T}$ and $\mathbf{W}$ are 6x6 matrices.

The partial incoming current conditions also can be rewritten as:

$$\begin{bmatrix} j^{in}_{ur1} - j^{in}_{ul1} \\ j^{in}_{ur2} - j^{in}_{ul2} \end{bmatrix} = \mathbf{M}_u \begin{bmatrix} a_{u1} \\ a_{u3} \end{bmatrix} + \mathbf{P}_u \begin{bmatrix} b^u_{11} \\ b^u_{21} \end{bmatrix} \tag{10}$$

where $\mathbf{M}_u$ and $\mathbf{P}_u$ are 2x2 matrices and

$$2\,\mathbf{j}^{in}_{sum} = \mathbf{a} + \mathbf{B}\,\mathbf{a}_e + \mathbf{P}\,\mathbf{b}_2 \tag{11}$$

where $\left[\mathbf{j}^{in}_{sum}\right]^T = \left[\left(j^{in}_{xr1} + j^{in}_{xl1}\right) \quad \left(j^{in}_{xr2} + j^{in}_{xl2}\right) \quad \left(j^{in}_{yr1} + j^{in}_{yl1}\right) \quad \left(j^{in}_{yr2} + j^{in}_{yl2}\right) \quad \left(j^{in}_{zr1} + j^{in}_{zl1}\right) \quad \left(j^{in}_{zr2} + j^{in}_{zl2}\right)\right]$,
$\mathbf{B} = diag(\mathbf{B}_x, \mathbf{B}_y, \mathbf{B}_z)$, $\mathbf{P} = 2 \times diag(\mathbf{P}_x, \mathbf{P}_y, \mathbf{P}_z)$, and $\mathbf{B}_u$ is a 2x2 matrix

Rearranging Eqs. (9) and (11) leads to

$$(\mathbf{W} - \mathbf{T}\,\mathbf{B})\,\mathbf{a}_e = \mathbf{T}\,(\mathbf{P}\,\mathbf{b}_2 - 2\mathbf{j}^{in}_{sum}) - \mathbf{V}\,\mathbf{b}_2 \tag{12}$$

$$\mathbf{a} = 2\mathbf{j}^{in}_{sum} - \mathbf{B}\,\mathbf{a}_e - \mathbf{P}\,\mathbf{b}_2 \tag{13}$$

Eq. (12) is solved for the even coefficients by Gaussian elimination of the 6x6 linear system of equations. The coefficients **a** are obtained from the even coefficients using Eq. (13) and the odd coefficients are obtained by Eq. (10) for each direction.

In the one-node CMFD algorithm, two correction factors for each node interface are required and two forms of the CCFs were considered in the numerical tests:

- Shin's form [3]

  Two net currents at left and right interfaces of a node are corrected by the correction factors as:

$$J_r = -\tilde{D}\left(\phi_r - \bar{\phi}\right) - \hat{D}_r\left(\bar{\phi} + \phi_r\right)$$
$$J_l = -\tilde{D}\left(\bar{\phi} - \phi_r\right) - \hat{D}_l\left(\bar{\phi} + \phi_l\right) \tag{14}$$

where $\tilde{D} = 2D/h$ and the CCFs are defined by:

$$\hat{D}_r = \frac{-J_r - \tilde{D}\left(\phi_r - \bar{\phi}\right)}{\left(\bar{\phi} + \phi_r\right)} \quad \text{and} \quad \hat{D}_l = \frac{-J_l - \tilde{D}\left(\bar{\phi} - \phi_l\right)}{\left(\bar{\phi} + \phi_l\right)} \tag{15}$$

- Joo's form [9]

  Net current and surface flux are corrected by two separate correction factors:

$$J = -\tilde{\tilde{D}}\left(\bar{\phi}_r - \bar{\phi}_l\right) - \hat{D}\left(\bar{\phi}_r + \bar{\phi}_l\right)$$
$$\phi_s = \alpha\bar{\phi}_r + (1-\alpha)\bar{\phi}_l + \beta\left(\bar{\phi}_r + \bar{\phi}_l\right) \tag{16}$$

where $\tilde{\tilde{D}} = D/h$ and $\alpha = \frac{\theta_r}{\theta_r + \theta_l}$; $\theta = \frac{D}{h}$. Two correction factors are defined as

$$\hat{D} = \frac{-J - \tilde{\tilde{D}}\left(\bar{\phi}_r - \bar{\phi}_l\right)}{\left(\bar{\phi}_r + \bar{\phi}_l\right)} \quad \text{and} \quad \beta = \frac{\phi_s - \alpha\bar{\phi}_r - (1-\alpha)\bar{\phi}_l}{\left(\bar{\phi}_r + \bar{\phi}_l\right)} \tag{17}$$

and the under-relaxation is defined by

$$\hat{D}^{(k+1)} = \alpha\,\hat{D}^{(k+1/2)} + (1-\alpha)\,\hat{D}^{(k)} \quad \text{and} \quad \beta^{(k+1)} = \alpha\,\beta^{(k+1/2)} + (1-\alpha)\,\beta^{(k)} \tag{18}$$

where $\hat{D}^{(k+1/2)}$ and $\beta^{(k+1/2)}$ are the CCFs obtained using Eq. (17), α is the relaxation parameter, and k is the iteration index. The two forms of the CCFs are compatible in the sense that one form can be obtained from the other form, i.e., using Shin's CCFs, Eq. (15), the net current and surface flux of Joo's form can be obtained as [10]:

$$J = -\tilde{\tilde{\tilde{D}}}(\bar{\phi}_r - \bar{\phi}_l) - \hat{\hat{D}}(\bar{\phi}_r + \bar{\phi}_l)$$
$$\phi_s = \alpha_r\bar{\phi}_r + \alpha_l\bar{\phi}_l \tag{19}$$

where

$$\tilde{\tilde{\tilde{D}}} = \frac{4\theta_r\theta_l + \hat{D}_r\hat{D}_l}{2(\theta_r + \theta_l) - \hat{D}_r + \hat{D}_l}, \qquad \hat{\hat{D}} = \frac{2(\theta_l\hat{D}_r + \theta_r\hat{D}_l)}{2(\theta_r + \theta_l) - \hat{D}_r + \hat{D}_l},$$
$$\alpha_r = \frac{2\theta_r + \hat{D}_r}{2(\theta_r + \theta_l) - \hat{D}_r + \hat{D}_l}, \qquad \alpha_l = \frac{2\theta_l - \hat{D}_l}{2(\theta_r + \theta_l) - \hat{D}_r + \hat{D}_l}.$$

The current correction factor of Joo's form is the same as the conventional two-node current correction factor which was already implemented in PARCS [2]. Therefore, Joo's CCF was implemented in PARCS

using the already existing CCF with an additional surface flux correction factor (SFCF) β. Shin's CCFs were implemented using the conversion of CCFs given by Eq. (19).

2.3 Numerical Estimate of Convergence Rate

The CMFD algorithm is a nonlinear algorithm of global / local iterations which achieves high-order accurate solutions using the CCFs. The iterations can be written as below using operator notation:

$$\mathbf{M}^{(k)}(\hat{D}^{(k)})\mathbf{\Phi}^{(k+1)} = \frac{1}{k_{eff}^{(k+1)}}\mathbf{F}^{(k)}\mathbf{\Phi}^{(k+1)} \tag{20}$$

$$\hat{D}^{(k+1)} = K_n(\mathbf{\Phi}^{(k+1)}, k_{eff}^{(k+1)});\ n = 1 \text{ or } 2 \tag{21}$$

where $\mathbf{M}(\hat{D})$ and $\mathbf{F}$ represents neutron removal and generation, respectively. Superscript (k) represents iteration index and K_n is either one-node kernel (n=1) or two-node kernel (n=2). The convergence of CCFs can be measured using the formula:

$$\text{Convergence Rate} = \frac{\left\| \hat{D}^{(k)} - \hat{D}^{(k-1)} \right\|_2}{\left\| \hat{D}^{(k-1)} - \hat{D}^{(k-2)} \right\|_2} \tag{22}$$

In order to obtain a consistent numerical measurement of the CCF convergence rate, the iterations between each CCF updates are fully converged. However, the usual practice is to employ only three or four outer iterations between CCF updates in order to reduce the overall computational time. This should be noted when assessing the numerical tests. It is worthwhile to note that the convergence rate of Eq. (22) is not like a dominance ratio of the eigenvalue problem which is essentially dependent on the core size measured in diffusion length, but literally a convergence rate of CCFs which will be shown later to be a function of the problem mesh size.

3. Numerical Convergence Analysis

3.1 Simple Three-Dimensional Two-Group Problem

The first three-dimensional two-group model problem is an eigenvalue problem in which the geometry is a cube with uniform mesh in all directions. The material is homogeneous and the cross section data set (Table 1) was taken from the composition #4 base cross section of the NEACRP LWR transient benchmark [11]. The boundary conditions are 'incoming current zero' at all boundaries. As indicated in Fig. 1, the convergence rate of the CCFs becomes asymptotic for the core size greater than 100 cm. Test cases used in the numerical tests are listed in Table 2. The core sizes for each test were greater than 100 cm and the number of total mesh in each direction were chosen to be greater or equal to 10.

Table 1 Cross Section Data for Two-Group Problem

Group g	D_g	Σ_{rg}	$\nu\Sigma_{fg}$	Σ_{12}
1	1.500711	0.026968	0.00498277	0.0182498
2	0.415038	0.065255	0.0839026	

Table 2 List of Test Cases (3-D Cubic Core with Uniform Meshes)

# of Meshes	Mesh Size (cm)	Core Size (cm)
50	2	100
30	3.5	105
20	5	100
14	7.5	105
10	10	100
10	20	200
10	40	400
10	80	800

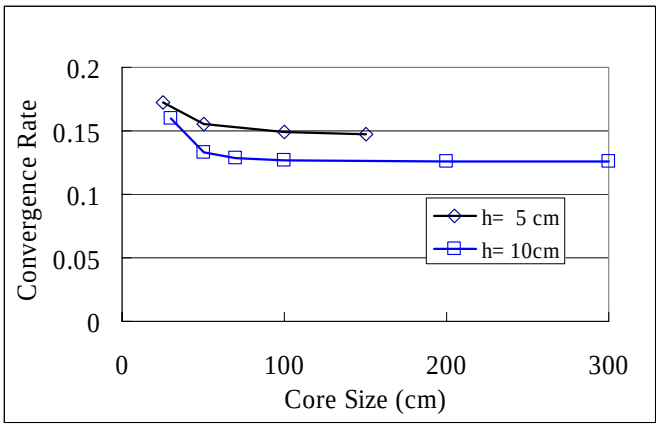


Fig. 1 Sensitivity of Convergence Rate to Core Size
(2-N CMFD; 3-D 2-G Model Eigenvalue Problem)

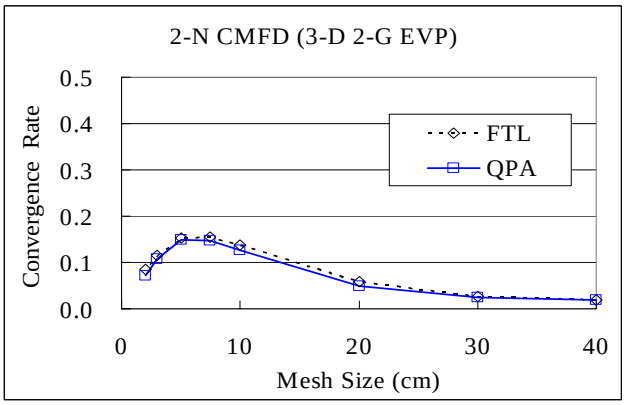


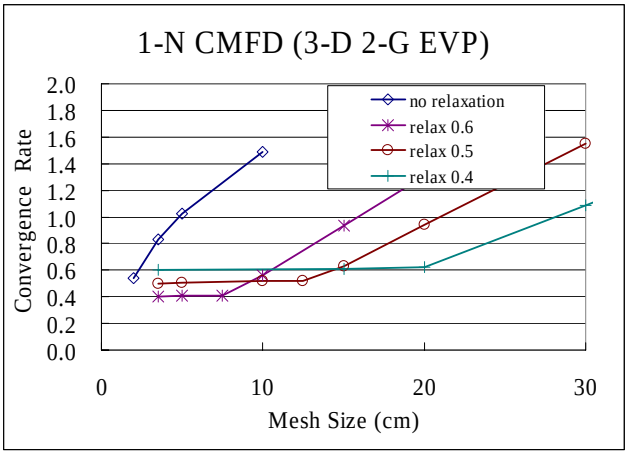
Fig. 2 Convergence Rate vs. Mesh Size
(2-N CMFD; 3-D 2-G Model Problem)

In Fig. 2, the convergence rate of the CCFs vs. mesh size is plotted for 2-N CMFD algorithm. FTL and QPA represent 'Flat Transverse Leakage' and 'Quadratic Polynomial Approximation', respectively. The overall trend is the same as that of the 1-D, 1-G FSP [6,7]

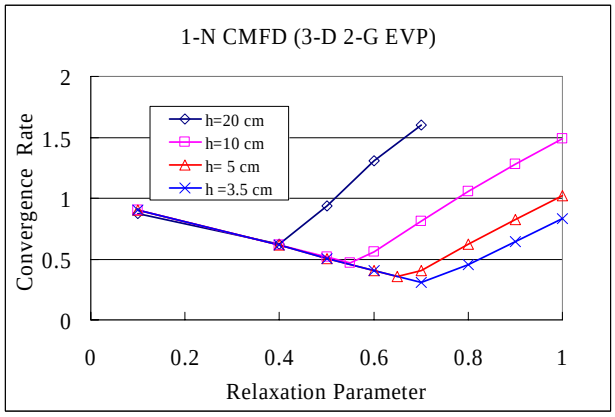
As for the one-node CMFD algorithm, there are several possible variants depending on the particular strategy used to implement it as in Table 3. In the table, 'Jacobi Jin update' denotes the partial current update strategy such that all the incoming currents of one-node problem are provided from CMFD, and the 'Gauss-Seidel (G-S) Jin update' means if the outgoing partial currents of the adjacent node have already been updated, then they will be used as incoming currents of the node instead of the ones from CMFD. Therefore, in the natural ordering 'G-S Jin update' scheme, half of the incoming currents of a node are from the adjacent nodes and the remaining half are from CMFD

Table 3 Parameters for One-Node CMFD Strategy

CCF Form	Shin's form / Joo's form
Jin Update	Jacobi / Gauss-Seidel
# of Sweeps	1 ~ 7
Under-relaxation	$0 < \alpha < 1$
Sweep Ordering	Natural / Red-Black



(a) **Convergence Rate vs. Mesh Size**



(b) **Convergence Rate vs. Relaxation Parameter**

Fig. 3 Sensitivity of 1-N CMFD Convergence Rate to Mesh Size and Relaxation Parameter with 3-D 2-G Model EVP (Shin's CCF; Jacobi-style Jin Update, Single Sweep)

Shin's CCF with a single one-node sweep (Jacobi-style Jin Update) was used for a sensitivity study of the 1-N CMFD convergence rate to the mesh size and relaxation parameter. The results shown in Fig. 3 reveal the following trends:

- As the mesh size increases, the convergence becomes slower and eventually the algorithm diverges (convergence rate > 1.0) for large mesh size.
- Under-relaxation can expand the stability domain of the algorithm. The greater the under-relaxation, the larger the stability domain.
- When an under-relaxation of α is applied, the convergence rate approaches (1-α) as the mesh size approaches zero.

The observations above and the overall trends of the convergence behavior shown in Fig. 3 are the same as those for the 1-D 1-G FSP model problem [6,7]

The tests shown in Fig. 4 are with Shin's form of the CCF and the 1-N CMFD for the 3-D 2-G model eigenvalue problem. 'N-O' and 'R-B' represent 'natural ordering' and 'red-black ordering', respectively. Based on this figure, the following trends are evident:

- The 'G-S Jin update' converges faster than the 'Jacobi Jin update'.
- The natural ordering and the red-black ordering show comparable performance.
- The more sweeps of the one-node local problems, the faster the convergence.

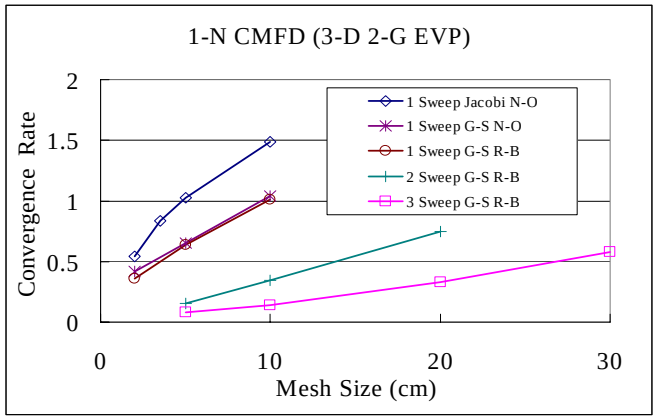


Fig. 4 Sensitivity of 1-N CMFD Convergence Rate to Sweep Strategy for the 3-D 2-G Model EVP (Shin's CCF)

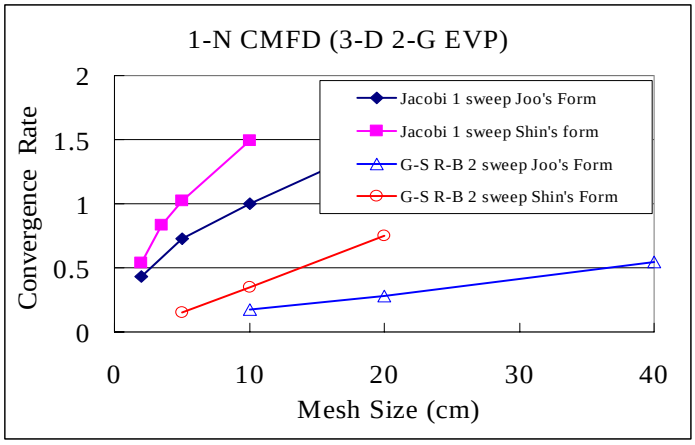


Fig. 5 Sensitivity of 1-N CMFD Convergence Rate to CCF Form
(Jacobi / 2 G-S-style Sweeps, R-B Ordering)

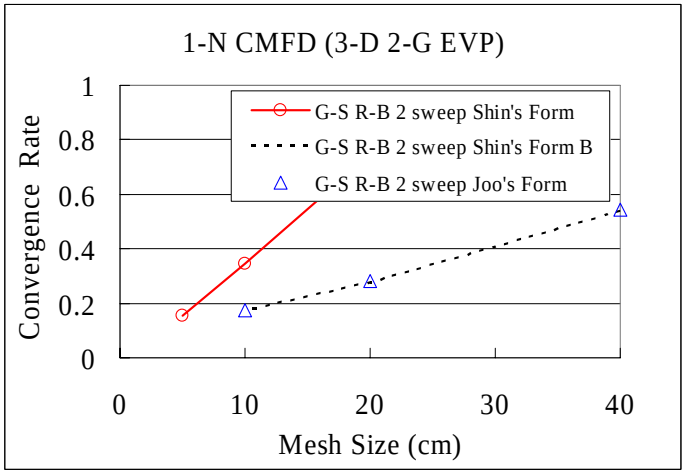


Fig. 6 Sensitivity of 1-N CMFD Convergence Rate to CCF Form
(Shin's Form B : CCF Update after All the Local One-Node Sweeps)

As shown in Fig. 5, Joo's form of the CCF converges faster than Shin's CCF. However, in the earlier discussion based on Eq. (19), the two CCF forms were noted to be equivalent. The performance difference observed in Fig. 5 was caused by the way in which the CCFs are updated. Joo's CCF is updated after all the local one-nodes are swept since it is dependent on the two nodes adjacent to the interface. Conversely, Shin's CCF is updated simultaneously with the one-node solution and which is more consistent with the one-node CMFD concept. If Shin's CCFs are also updated in the same way as Joo's CCF, then the convergence rates of two CCFs are same as shown in Fig. 6.

3.2 NEACRP LWR Transient Problem

The NEACRP LWR transient benchmark problem [11] was solved using the PARCS code with the existing two-node kernel and the newly implemented one-node kernel. The solutions from the one-node kernel are consistent with those from the two-node one regardless of the number of one-node local sweeps, except when a single local sweep was used in which case the problem did not converge.

In Fig. 7, the 'Jacobi Jin Update' and the 'G-S Jin Update' are compared with each other in terms of

the global fission source error reduction for the NEACRP A1 steady state problem. The global fission source error is defined as:

$$\text{Global Fission Source Error} = \frac{\left\| \mathbf{e}^{(k+1)} \right\|_2}{\gamma} \tag{27}$$

where

$$\mathbf{e}^{(k+1)} = \boldsymbol{\psi}^{(k+1)} - \boldsymbol{\psi}^{(k)}, \quad \boldsymbol{\psi}^{(k)} = \text{vector of } \psi^{m,(k)}; \text{m} \in \text{Core}, \quad \gamma = \sqrt{\sum_{m \in \text{Core}} \left(\psi^{m,(k)} * \psi^{m,(k+1)} \right)}$$

and where $\psi^m = \frac{1}{k_{eff}} \sum_{g=1}^{2} \nu \Sigma_{fg}^m \phi_g^m$ is a fission source at node m, , k is an outer iteration index.

Joo's CCF and triple sweep were used with the 1-N kernel. As indicated, the 'Jacobi Jin Update' diverges but the 'G-S Jin Update' converges. Also, oscillations are observed for the global fission source of the 1-N CMFD algorithm but not for the 2-N CMFD algorithm.

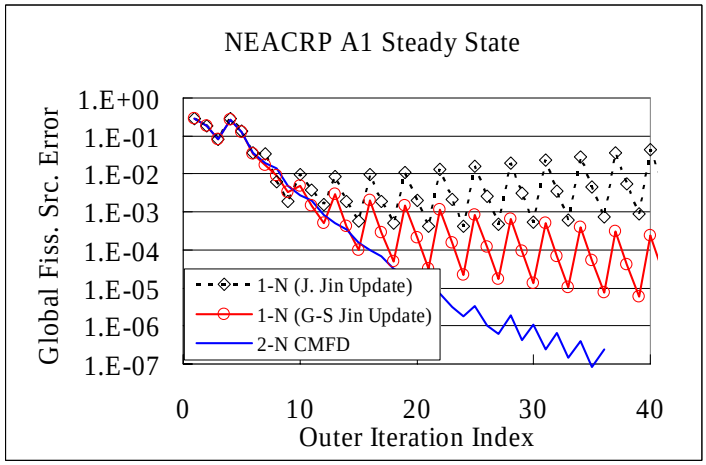


Fig. 7 Jacobi vs. Gauss-Seidel Jin Update (Joo's CCF, Triple Sweep)

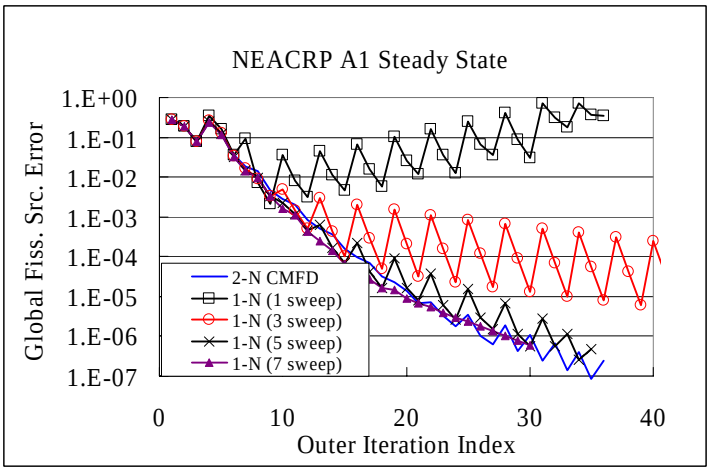


Fig. 8 Convergence Rate of 1-N/2-N CMFD (Joo's CCF , G-S Jin Update)

As indicated in Fig. 8, a single-sweep 1-N CMFD algorithm diverges because of the oscillations of the fission source error. As the number of one-node sweeps increases, the oscillations of the global fission source errors are damped out, and with 7 sweeps, the oscillations almost disappear and the error reduction of the 1-N CMFD becomes comparable to the 2-N CMFD.

In Table 4, the number of nodal updates and the execution time of PARCS for the NEACRP A1 problem are summarized for the 2-N CMFD and the 1-N CMFD kernels. 'Nodal Update' represents the number of CCF updates using the solutions to the local one-node / two-node problems and 'CMFD update' represents the number of outer iterations to solve the eigenvalue problem with CCFs in the steady state and to solve the fixed source problem with CCFs in the transient. As indicated in the table, a 5-sweep 1-N CMFD is comparable to a 2-N CMFD in terms of the number of updates, for both the steady state and transient calculations. However, a 5-sweep 1-N CMFD algorithm takes longer than a 2-N CMFD because of the multiple sweeps. The CMFD execution time of 2-N CMFD and a 5-sweep 1-N CMFD are similar but the nodal update time of the 5-sweep 1-N CMFD is much longer.

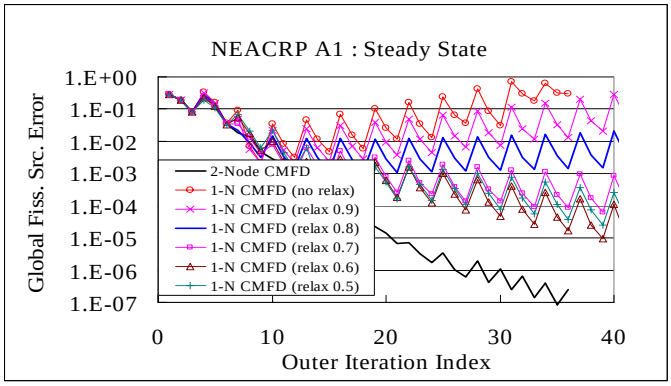
Table 4 Summary of NEACRP A1 Calculation (Joo's CCF, G-S Jin Update, R-B Ordering)

(a) Steady State Calculation

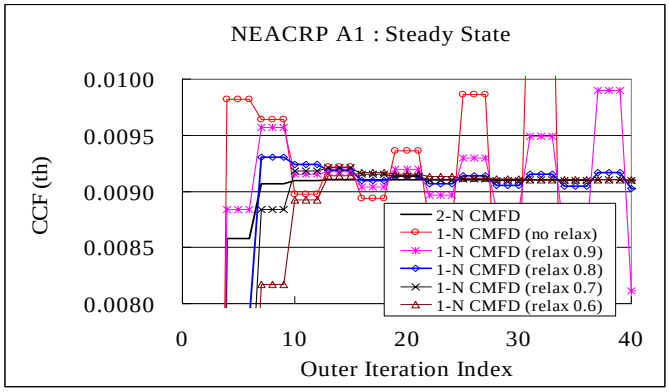
	2-Node	1-Node (# of sweep)			
		1	3	5	7
CBC (ppm)	561.256	Diverge	561.256		
Nodal Update	13		35	12	10
CMFD Update	36		99	35	30
Nodal Time (s)	0.7		5.4	2.9	3.3
CMFD Time (s)	1.9		4.3	1.9	1.8
Total Time (s)	2.6		9.8	4.9	5.2
Time/Nodal Sweep (s)	0.053		0.052	0.049	0.047
percentage to 2-N (%)			97.3	91.7	88.0

(b) Transient Calculation

	2-Node	1-Node (# of sweep)		
		3	5	7
Peak Power (%)	125.87	125.86	125.84	125.99
Nodal Update	31	49	28	24
CMFD Update	274	292	278	270
Nodal Time (s)	1.7	7.7	6.7	7.9
CMFD Time (s)	19.9	20.5	19.6	19.4
Total Time (s)	27.9	34.6	32.9	33.6
Time/Nodal Sweep (s)	0.054	0.052	0.048	0.047
percentage to 2-N (%)		96.4	87.9	87.2



(a) Error Reduction vs. Relaxation Parameter



(b) **CCF Convergence of Node** (k=9, l=40)

Fig. 9 Convergence of 1-Sweep 1-N CMFD for NEACRP A1 Steady State
(Joo's CCF, G-S Jin Update, R-B Ordering)

In Fig. 9, various under-relaxation parameters were tested for the 1-sweep 1-N CMFD algorithm using the NEACRP A1 steady state problem. As indicated in Fig. 9 (a), the optimum relaxation appears to be about 0.6. As shown in Fig. 9 (b), the CCF oscillations are damped by under-relaxation. The CCF in this figure is from the thermal energy group at the west boundary of the node $k = 9$, $l = 40$. Since the total number of axial plane is 18, the plane number of $k = 9$ represents almost a core center and the radial location of $l = 40$ is shown in Fig. 10. In Table 5, the optimum relaxation also seems to be 0.6 in terms of the number of nodal updates and total computation time. As indicated, even the optimized 1-N CMFD algorithm is much slower than the 2-N CMFD.

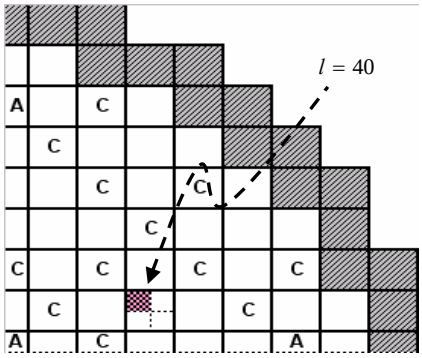


Fig. 10 PARCS Quarter-Core Model for NEACRP A1 Case (A :100 steps, C:200 steps)

Table 5 Summary of NEACRP A1 Steady State with 1-Sweep 1-N CMFD
(Joo's CCF, G-S Jin Update, R-B Ordering)

	2-Node	1-Node (Relaxation Parameter)			
		1.0 – 0.8	0.7	0.6	0.5
CBC (ppm)	561.256	Diverge	561.256		
Nodal Update	13		89	28	33
CMFD Update	36		251	79	91
Nodal Time (s)	0.7		6.42	2.0	2.3
CMFD Time (s)	1.9		10.23	3.8	4.1
Total Time (s)	2.6		16.7	5.9	6.5
Time/Nodal Sweep (s)	0.053	–	0.072	0.072	0.070
percentage to 2-N (%)		–	135.9	135.9	131.3

Table 6 Summary of NEACRP A1 Steady State with 2- Sweep 1-N CMFD
(Joo's CCF, G-S Jin Update, R-B Ordering)

	2-Node	1-Node (Relaxation Parameter)				
		1.0	0.9	0.8	0.7	0.6
CBC (ppm)	561.256	Diverge	561.256			
Nodal Update	13		80	23	14	16
CMFD Update	36		226	65	40	46
Nodal Time (s)	0.7		9.19	2.7	1.6	1.9
CMFD Time (s)	1.9		10.58	3.3	2.4	2.5
Total Time (s)	2.6		19.83	6.1	4.1	4.5
Time/Nodal Sweep (s)	0.053	–	0.057	0.058	0.056	0.060
percentage to 2-N (%)		–	108.2	108.9	106.3	112.5

Table 7 Summary of NEACRP A1 Steady State with 3-Sweep 1-N CMFD
(Joo's CCF, G-S Jin Update, R-B Ordering)

	2-Node	1-Node (Relaxation Parameter)			
		1.0	0.9	0.8	0.7
CBC (ppm)	561.256	561.256			
Nodal Update	13	35	17	11	12
CMFD Update	36	99	49	33	35
Nodal Time (s)	0.7	5.4	2.6	1.7	1.9
CMFD Time (s)	1.9	4.3	2.3	1.8	1.9
Total Time (s)	2.6	9.8	4.9	3.6	3.8
Time/Nodal Sweep (s)	0.053	0.052	0.050	0.052	0.052
percentage to 2-N (%)		97.3	94.6	97.1	97.3

The same tests were performed with double one-node sweeps. As indicated in Table 6, the optimum value of relaxation is about 0.7, and the 1-N CMFD algorithm is comparable to the 2-N CMFD algorithm in terms of the number of nodal updates, but not in terms of the computation time. Again, the same tests were performed for the triple local one-node sweeps. The performance of the 1-N CMFD algorithm with 3 one-node sweeps and with a relaxation parameter of 0.8 is comparable to the 2-N CMFD in terms of the number of updates as indicated in Table 7.

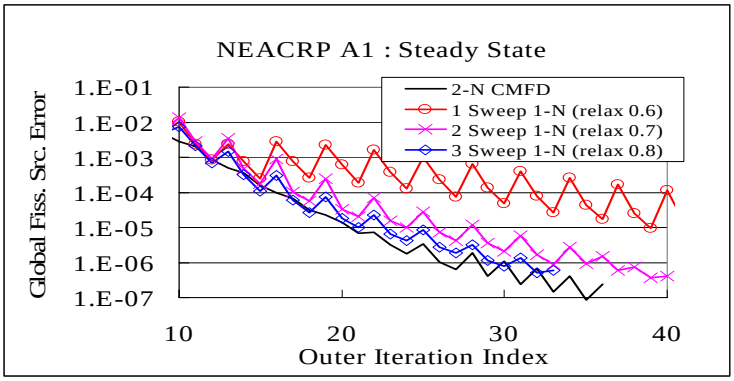


Fig. 11 Convergence of 1-N CMFD for NEACRP A1 Steady State
(Joo's CCF, G-S Jin Update, R-B Ordering)

Table 8 Summary of NEACRP Steady State Calculations
(Joo's CCF; triple sweeps (1-N CMFD), G-S Jin Update, R-B Ordering, Under-relaxation 0.8)

Parameters	A1		B1		C1		A2	
	HZP, QC		HZP, QC		HZP, HC		HFP, QC	
	2N	1N	2N	1N	2N	1N	2N	1N
Nodal Update	13	11	11	11	11	19	13	11
CMFD Update	36	33	31	32	31	58	37	33
Nodal Time (s)	0.7	1.9	0.6	1.7	1.3	5.8	0.8	1.7
Total Time (s)	2.8	3.8	2.4	3.8	4.9	12.1	3.5	4.1
Time/Nodal Sweep (s)	0.055	0.056	0.055	0.052	0.115	0.102	0.058	0.052
percentage to 2-N (%)		102.9		93.2		88.8		89.5

Table 9 Summary of NEACRP Transient Calculations
(Joo's CCF; triple sweeps (1-N CMFD), G-S Jin Update, R-B Ordering, Under-relaxation 0.8)

Parameters	A1		B1		C1		A2	
	HZP, QC		HZP, QC		HZP, HC		HFP, QC	
	2N	1N	2N	1N	2N	1N	2N	1N
Peak Power (%)	125.9	126.0	125.9	126.0	482.2	482.4	108.6	108.6
Nodal Update	31	29	30	28	19	15	10	10
CMFD Update	274	277	262	261	327	322	182	208
Nodal Time (s)	1.7	4.3	1.8	4.3	2.3	4.5	0.6	1.5
Total Time (s)	27.9	29.7	27.3	29.5	81.3	83.3	19.1	20.1
Time/Nodal Sweep (s)	0.054	0.050	0.060	0.051	0.120	0.101	0.056	0.050
percentage to 2-N (%)		92.2		84.3		84.2		88.1

In Fig. 11, the 1-N CMFD algorithms with single, double, and triple one-node sweeps are plotted with optimum relaxation parameter of each. In terms of convergence rate (global fission source error reduction), the 3-sweep 1-N CMFD algorithm with a relaxation parameter of 0.8 shows the best performance.

As shown in Tables 8 and 9, the triple-sweep 1-N CMFD algorithm with a relaxation parameter of 0.8

was tested for NEACRP steady state and transient calculations. In terms of the CCF update count, the 1-N CMFD algorithm is comparable to the 2-N CMFD algorithm for both the steady state and transient problems. However, the computation time takes longer for the 1-N CMFD algorithm because of the multiple one-node sweeps. This can be mitigated by further optimizations of the 1-N kernel coding. As indicated in the tables, the one-node kernel has more potential for optimization.

4. Conclusions

Convergence of the CCF was analyzed using numerical estimates of the convergence rate with two test problems: a 3-D 2-G eigenvalue problem and the NEACRP LWR transient benchmark problem. It was observed that the overall trend of the CCF convergence is similiar to that of the 1-D 1-G fixed source model problem which was previously analyzed in the references 6 and 7. Specifically,

- The CCF convergence rates of the 1-N / 2-N CMFD algorithms depend on the mesh size and the relaxation parameters.
- The 2-N CMFD algorithm is always stable. However, the 1-N CMFD algorithm is stable only for small mesh sizes and diverges for large mesh sizes.
- With under-relaxation of the CCFs, the stability domain of 1-N CMFD algorithm can be substantially expanded.
- The larger the mesh size, the more under-relaxation is required.

Additional insights about the algorithms were obtained from the numerical tests, which might be useful for the practical implementation of the 1-N CMFD algorithm:

- As the number of local one-node sweeps increases, CCF converges faster.
- The 'Gauss-Seidel Jin Update' is more effective than the 'Jacobi Jin Update'
- The CCF update after sweeping all the nodes is more efficient than using an instantaneous CCF update right after solving each one-node problem.

It is noted that the insights obtained from the analytic study with simple model problems can be applied to predict the convergence behavior of the algorithms applied to the more realistic problem.

References

1) K. S. Smith, "Nodal Method Storage Reduction by Non-linear Iteration," *Trans. Am. Nucl. Soc.*, **44**, 265 (1983).
2) H. G. Joo, D. A. Barber, G. Jiang, and T. J. Downar, *PARCS: A Multi-Dimensional Two-Group Reactor Kinetics Code Based on the Nonlinear Analytic Nodal Method*, PU/NE-98-26, Purdue University, 1998.
3) H. C. Shin, Y. H. Kim, and Y. B. Kim, "One-Node Coarse-Mesh Finite Difference Algorithm for Fine-Mesh Finite Difference Operator," *Trans. Am. Nucl. Soc.*, **81**, p.150 (1999).
4) K. S. Smith and J. D. Rhodes, III, "Full-Core, 2-D, LWR Core Calculations with CASMO-4E," *Proc. ANS Reactor Physics Topical Meeting, PHYSOR 2002,* Seoul, Korea, October 7-10 (2002).
5) H. G. Joo, J. Y. Cho, H. Y. Kim, S. Q. Zee, and M. H. Chang, "Dynamic Implementation of the Equivalence Theory in the Heterogeneous Whole Core Transport Calculation," *Proc. ANS Reactor Physics Topical Meeting, PHYSOR 2002,* Seoul, Korea, October 7-10 (2002).
6) Deokjung Lee, Thomas J. Downar, and Yonghee Kim, "Convergence Analysis of the Nonlinear Coarse Mesh Finite Difference Method," *Nuclear Mathematical and Computational Sciences: A Century in Review*, Gatlinburg, Tennessee, April 6-11, American Nuclear Society (2003).
7) Deokjung Lee, Thomas J. Downar, and Yonghee Kim, "Convergence Analysis of the Nonlinear

Coarse Mesh Finite Difference Method for One-Dimensional Fixed Source Neutron Diffusion Problem," *Nucl. Sci. Eng.*, to appear.

8) Nam Zin Cho and Chang Je Park, "A Comparison of Coarse Mesh Rebalance and Coarse Mesh Finite Difference Accelerations for the Neutron Transport Calculations," *Nuclear Mathematical and Computational Sciences: A Century in Review*, Gatlinburg, Tennessee, April 6-11, American Nuclear Society (2003).
9) H. G. Joo, et. al., "One-Node Solution Based Nonlinear Analytic Nodal Method," *Proc. KNS Spr. Mtg*, (2000).
10) Chang-Ho Lee, *A Multi-level Acceleration Method for the Multi-group SP3 Approximation to the Neutron Transport Equation*, Ph.D. Thesis, School of Nuclear Engineering, Purdue University, West Lafayette, IN. (2001).
11) H. Finnemann and A. GALATI, "NEACRP 3-D LWR Core Transient Benchmark," NEACRP-L-335(Rev. 1), Nuclear Energy Agency, (Jan. 1992).