

Parallelization of Nonoverlapping Local/Global Iterative Method with 2-D/1-D Fusion Transport Kernel and p-CMFD Wrapper for Transient Reactor Analysis

Bumhee Cho and Nam Zin Cho*

Korea Advanced Institute of Science and Technology (KAIST)
291 Daehak-ro, Yuseong-gu, Daejeon, Korea 34141

*Corresponding author: nzcho@kaist.ac.kr

1. Introduction

As modern computing power grows, direct whole-core transport calculations become more viable. The 2-D/1-D fusion method [1, 2] has been developed as a candidate for a 3-D transport solver. However, the computing power growth is limited by CPU clock speed [3], and huge memory requirement still remains as a problem in the whole-core transport calculation [4, 5]. To lessen those issues, the nonoverlapping local/global iterative (NLG) method with the 2-D/1-D fusion kernel and the global p-CMFD wrapper has been developed [6], and extended to transient calculations [7]. In the NLG iteration, local problems are independent, so the parallelization is quite straightforward. By adopting the parallel computing, computing time can be reduced, and computing memory can be distributed in the parallel computing nodes.

In this paper, the NLG iteration has been parallelized in the local problems under MPI protocol.

2. Transient 2-D/1-D Fusion Method

The local kernel in the NLG iteration is the 2-D/1-D fusion method. The basic idea is that a 3-D transport equation is decomposed into a radial 2-D equation and an axial 1-D “transport” equation, and they are coupled through their leakage source terms. The 2-D equation is solved by the MOC method, and the 1-D equations is solved by the S_N method.

2.1 Time-Dependent Neutron Transport Equation

To formulate a time-discretized equation, the following time-dependent neutron transport equation in a given discretized angle Ω_j is considered:

$$\begin{aligned} \frac{1}{v_g} \frac{\partial \psi_{g,j}(\vec{r}, \bar{\Omega}_j, t)}{\partial t} + \bar{\Omega}_j \cdot \nabla \psi_{g,j} + \sigma_{t,g} \psi_{g,j} = \\ \frac{1}{4\pi} \left[\frac{(1-\beta)}{k_{eff}} \chi_{p,g} \sum_{g'} v \sigma_{f,g'} \phi_{g'} + \sum_{g'} \sigma_{s0,g' \rightarrow g} \phi_{g'} \right. \\ \left. + \sum_d \chi_{d,g} \lambda_d C_d(\vec{r}, t) \right], \end{aligned} \quad (2.1)$$

$$\frac{\partial C_d(\vec{r}, t)}{\partial t} = -\lambda_d C_d + \frac{\beta_d}{k_{eff}} \sum_{g'=1} v \sigma_{f,g'} \phi_{g'}. \quad (2.2)$$

where all notations are standard in reactor physics.

The delayed neutron precursor densities are integrated analytically by the assumption of fission source as a second-order polynomial in time. The fully implicit method is applied to the time derivative term, and isotropic assumption is applied to the time derivative term to avoid huge memory requirement [8, 9]. With these assumptions, the following time-discretized equation is obtained:

$$\sin \theta_j \frac{\partial \psi_{g,j}(p, z)}{\partial p} + \xi_j \frac{\partial \psi_{g,j}}{\partial z} + \sigma_{t,g} \psi_{g,j} = q_g. \quad (2.3)$$

where q_g includes fission, delayed neutron precursor, and time difference source terms, and the time step index is omitted for the sake of brevity.

2.2 2-D Transport Equation (MOC)

Eq. (2.3) is integrated over the axial direction for a computational plane k , then the following 2-D transport equation in a flat source region (FSR) m is obtained:

$$\begin{aligned} \sin \theta_j \frac{\partial \psi_{g,j,k,m}(p)}{\partial p} + \sigma_{t,g,k,m} \psi_{g,j,k,m} = \\ q_{g,k,m} - L_{g,j,k,m}^z, \end{aligned} \quad (2.4)$$

where

$$L_{g,j,k,m}^z = \xi_j \frac{\psi_{g,j,k+1/2,m} - \psi_{g,j,k-1/2,m}}{\Delta_k}.$$

Note that the axial leakage source term is given in the FSR for each discretized angle, so the solution of Eq. (2.4) is “transport” in the FSR (or fine-mesh) level for the given axial leakage. The axial leakage is updated during the 1-D transport calculation described in the next section. Eq. (2.4) is solved by a conventional 2-D MOC solver described in the literature [2].

2.3 1-D Transport Equation (S_N)

Eq. (2.3) is integrated over the radial direction for FSR m , then the following 1-D transport equation is obtained:

$$\xi_j \frac{\partial \psi_{g,j,m}(z)}{\partial z} + \sigma_{t,g,k,m} \psi_{g,j,m} = q_{g,m}(z) - L'_{g,j,k,m}, \quad (2.5)$$

where

$$L'_{g,j,k,m} = \frac{1}{A_m} \int_{A_m} \sin \theta_j \frac{\partial \psi_{g,j,k}(p)}{\partial p} dA.$$

The radial leakage source term is given in the FSR level, and updated during the 2-D MOC calculation. Eq. (2.5) is solved by the linear characteristics (LC) scheme, which gives accurate results in the slab geometry [10]. Eqs. (2.4) and (2.5) are solved by an iterative approach, and result in transport solution.

2.4 Rod Cusping Correction by the NSI Method

Rod movement scenario is usually involved in 3-D transient problems. Therefore, a computational mesh in which a rod is partially inserted should adopt a proper homogenization technique, otherwise it gives the rod cusping phenomenon [10]. In the 2-D/1-D calculation, the neighboring spectral index (NSI) method is used to treat a partially rodded node (PRN) [7]. The NSI method uses the flux spectrum of the neighbor node of the PRN, and the spectrum is used as weighting functions for the homogenization of the PRN. The flux spectrum is updated nonlinearly during the 2-D/1-D calculation, and it lessens much of the rod cusping phenomenon [7].

3. NLG Iteration Framework

In the NLG iteration, local problems are solved by the 2-D/1-D fusion method for given incoming angular fluxes as boundary conditions, and the solution of each local problem is used to construct the global equation via the p-CMFD equation [12, 13]. The local kernel and the global wrapper are linked by the local interface boundary conditions. The p-CMFD equation plays a role as the global wrapper, since it gives a modulated (updated) source in local problems and local incoming angular fluxes. The schematic illustration of the NLG iteration is shown in Fig. 1.

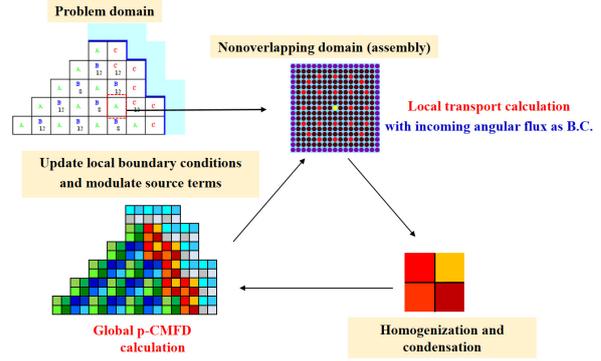


Fig. 1. Schematic illustration of the NLG iteration

3.1 Transient p-CMFD Methodology

After homogenization, condensation and integration of Eq. (2.3) in coarse group G and coarse mesh I , the following equation is obtained:

$$\sum_{u=x,y,z} \sum_{s=0,1} \frac{J_{G,u,s}^{+,I}(t_{n+1}) - J_{G,u,s}^{-,I}(t_{n+1})}{H_u^I} + \sigma_{r,G}^I \phi_G^I(t_{n+1}) = Q_G^I + \sum_{G' \neq G} \sigma_{s0,G' \rightarrow G}^I \phi_{G'}^I(t_{n+1}), \quad (3.1)$$

where

$$Q_G^I = (\alpha_G^I + (1 - \beta^I) \chi_G^I) F^I(t_{n+1}) + S_{d,G}^I(t_n) - \frac{1}{v_G^I} \frac{\phi_G^I(t_{n+1}) - \phi_G^I(t_n)}{\Delta t_{n+1}},$$

and other homogenized quantities are defined elsewhere [7].

In the p-CMFD equation, the partial currents are given as follows:

$$J_{G,u,s}^{+,I} = \frac{-\tilde{D}_{G,u,s}^+ (\phi_G^{I*} - \phi_G^I) + 2\hat{D}_{G,u,s}^+ \phi_G^I}{2}, \quad (3.2)$$

$$J_{G,u,s}^{-,I} = \frac{\tilde{D}_{G,u,s}^- (\phi_G^{I*} - \phi_G^I) + 2\hat{D}_{G,u,s}^- \phi_G^{I*}}{2}. \quad (3.3)$$

After substituting, Eqs. (3.2) and (3.3) into, and rearranging Eq. (3.1), the following transient fixed-source problem (TFSP) matrix is obtained:

$$A\phi = C, \quad (3.4)$$

where

ϕ is scalar flux vector,

C consists of $S_{d,G}^I(t_n) + (v_G^I \Delta t_{n+1})^{-1} \phi_G^I(t_n)$,

A consists of leakage, scattering, fission, delayed neutron operators, and $(v_G^I \Delta t_{n+1})^{-1} \phi_G^I(t_{n+1})$.

Eq. (3.4) can be solved by any linear equation solver. BiCGStab(2) [13] is used in this study.

3.2 Update of the Local Information

After solving Eq. (3.4), the local boundary conditions, the fission source, and other source terms are updated by the following equations:

$$\psi_{g,j,\partial D}^{new} = \frac{J_{G,s}^{-,l}}{J_{G,s}^{-,l,local}} \psi_{g,j,\partial D}^{old}, \quad (3.5)$$

$$\phi_{g,k,m}^{new} = \frac{\phi_G^l}{\phi_G^{l,local}} \phi_{g,k,m}^{old}, \quad (3.6)$$

where $g \in G$ and ∂D lies on local boundaries.

Note that the partial currents are directly obtained as the results of the p-CMFD equation.

3.3 Parallelization of Local Problems

In the NLG iteration, every local problem can be solved independently if the incoming angular fluxes are given as the boundary conditions. By using MPI protocol, the local problems are solved in distributed computing nodes simultaneously (domain parallelization). Therefore, huge computing memory and heavy calculational processing required in the transport calculation are distributed over the independent computing nodes. Once the local calculations are finished, a master node copies the local average quantities to be used in the global wrapper of the p-CMFD equation. After solving the p-CMFD equation, all local problems update their outgoing angular fluxes by Eq. (3.5), and they are passed to the neighboring local problems as their incoming boundary conditions. Therefore, communication time is required at this step, and the communication takes longer time as the local boundaries have more angular fluxes.

Since the 2-D/1-D fusion kernel is used as the local solver, it would be a good strategy to choose the single assembly as the local problem size. As shown in Eqs. (2.4) and (2.5), the angle-dependent leakage source terms must be saved in every FSR, and the single assembly in a PWR requires a few giga-byte level computing memory to obtain an accurate solution (0.02 ray spacing, 8 azimuthal angles, 3 polar angles, 50 FSRs in a fuel cell, and 20 axial meshes). Therefore, it can be said that the whole-core transport calculation is feasible by the parallel NLG iteration method without the cell homogenization technique [2, 6].

Not only domain parallelization but also angle parallelization is possible in the NLG iteration. The angle parallelization can be efficiently achieved by OpenMP protocol which uses the advantage of shared memory in each independent computing node. The parallelization schemes are described in the literature [15, 16].

4. Numerical Results

Numerical problems consist of three problems; 1) a three-dimensional homogeneous rodged-assembly ejection problem, 2) a three-dimensional heterogeneous single assembly rod ejection problem, and 3) a modification of the C5G7 benchmark [5] problem with rod ejection. The NLG iteration has been implemented in CRX-2K [7]. All numerical problems have the same discretized angle condition; 3 polar angles and 8 azimuthal angles per octant are used. TY quadrature set [17] is used as the polar angle quadrature set. Other calculational conditions such as the size of FSR and ray spacing are set to guarantee an accurate solution, and it is not included in this paper for the sake of brevity. Intel Xeon X5670 @ 2.93 GHz was used for the calculations.

4.1 Mini-Core 3D Problem [18]

The geometry of this problem is shown in Fig. 2, and the rodged assembly (UOXR) is ejected in 0.1 sec. The two-group cross sections and the information of six delayed neutron precursors are given in the literature [7]. The local problem size is chosen as 10.71 x 10.71 cm as shown in Fig. 2. To see the parallel efficiency, the NLG iteration is solved in three cases; 1) a single computing node, 2) three computing nodes, and 3) nine computing nodes. The p-CMFD acceleration is not parallelized in this study. The time step size is set to 2 ms.

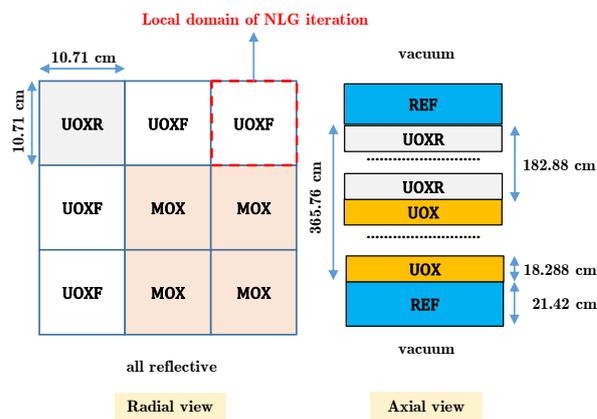


Fig. 2. Geometry (Mini-Core 3D)

Fig. 3 shows the relative power change over the time, and Table I shows the summary of the results. As shown in Fig. 3, the p-CMFD acceleration and the NLG iteration are identical within the numerical error criteria. The results by the NLG iteration with parallel computing nodes are exactly the same with the results by a single computing node, so they are omitted in Fig. 3. As shown in Table I, the NLG iteration takes longer computing time than the p-CMFD acceleration if the parallel computing is not applied due to weak couplings

between the local problems in the early stages of the local/global iterations. However, the computing speed becomes faster as more computing nodes are involved in the parallel computing as shown in Table I. Obviously, the memory requirement per node is reduced with the parallel computing.

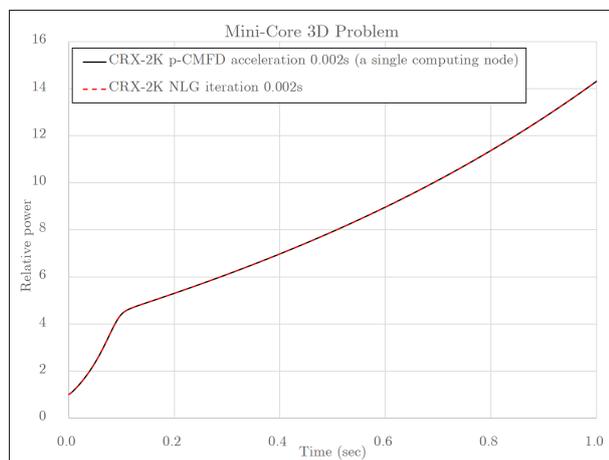


Fig. 3. Relative power vs time (Mini-Core 3D)

Table I: Result summary (Mini-Core 3D)

		p-CMFD acceleration	NLG iteration		
number of computing nodes		1	1	3	9
k-eff		1.08019	1.08019		
Computing time / (speedup)	steady-state (hr)	1.07	1.47 / (0.7)	0.48 / (2.2)	0.17 / (6.4)
	transient calculations (hr)	64.9	65.0 / (1.0)	23.0 / (2.8)	8.9 / (7.3)
Memory requirement (GB) per node		5.3	6.0	2.1	0.7

4.2 Single UO_2 Assembly Problem

The second problem consists of a single UO_2 assembly, and the fuel rods are described with heterogeneity as shown in Fig. 4. Seven-group cross sections are sourced from the C5G7 benchmark problem [5], and the information pertaining to delayed neutron precursors is sourced from the literature [19]. All control rods are initially inserted 7.14 cm in an active core, and the perturbation is originated from all rod ejection in 0.1 sec. Since this problem consists of a single UO_2 assembly, local problem size is smaller than the assembly size as shown in Fig. 4. The NLG iteration is solved in three cases; 1) a single computing node, 2) three computing nodes, and 3) nine computing nodes. The time step size is set to 2 ms.

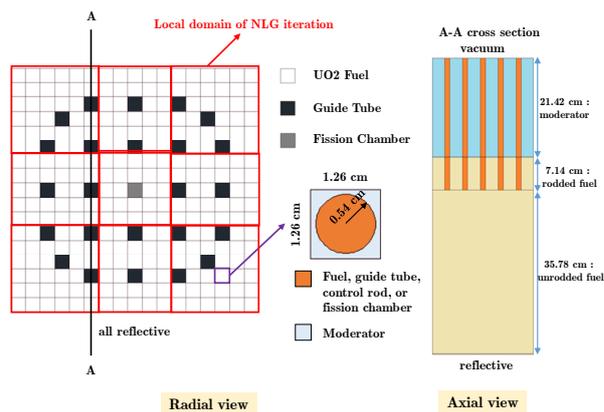


Fig. 4. Geometry (single UO_2 assembly)

The results are shown in Fig. 5 and Table II. As shown in Fig. 5, both the p-CMFD acceleration and the NLG iteration have identical results within the numerical error criteria. As shown in Table II, the computing time is reduced in the NLG iteration with parallel computing nodes. However, the speedup is rather poor in this problem. There are two specific main reasons for the poor scale-up of computing speed; 1) the local problem sizes are different (from 5 by 5 to 6 by 6), so there is an idle time in some computing nodes. However, this is not a critical issue compared to the second reason, 2) the local problem size is relatively small, so the communication time has a relatively big portion. Note that the ratio of the local interface angular flux memory to the local interior angular flux memory becomes larger as the local size becomes smaller (surface to volume ratio). In this problem, the communication time portion in every five 2-D/1-D sweeps is around 25 % in the three nodes case and 39 % in the nine nodes case.

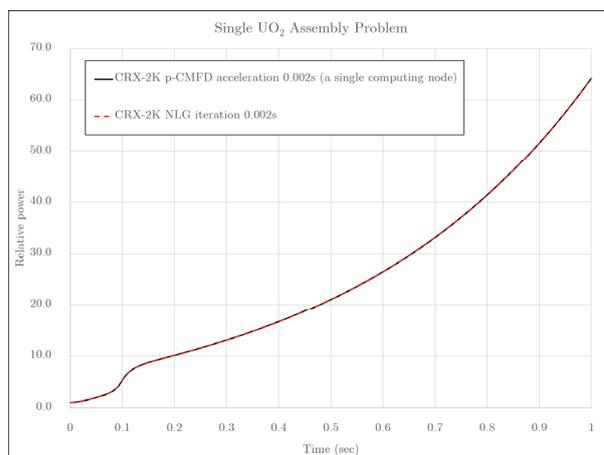


Fig. 5. Relative power vs time (single UO_2 assembly)

Table II: Result summary (single UO₂ assembly)

		p-CMFD acceleration	NLG iteration		
number of computing nodes		1	1	3	9
k-eff		1.27186	1.27186		
Computing time / (speedup)	steady-state (hr)	0.87	0.96 / (0.9)	0.48 / (1.8)	0.21 / (4.1)
	transient calculations (hr)	18.7	21.4 / (0.9)	13.61 / (1.4)	8.42 / (2.2)
Memory requirement (GB) per node		4.7	5.1	1.9	0.7

4.3 Modified C5G7 Benchmark Problem

The C5G7 benchmark problem [5] has been modified. The initial state of this problem is “Rodded A” state, and eight rods (red colored in Fig. 6) are ejected in 0.05 sec. Cross sections and delayed neutron precursors are the same with the previous numerical problem. The time step size is 2ms. This problem cannot be solved by the p-CMFD acceleration or the NLG iteration by a single computing node due to the lack of computing memory.

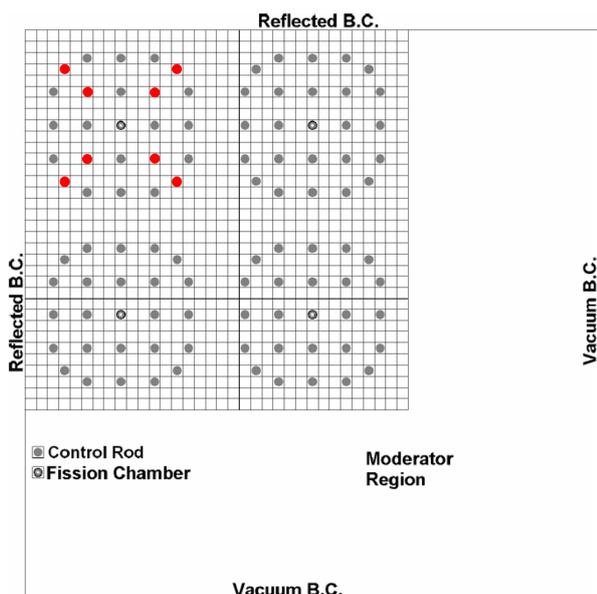


Fig. 6. Geometry (modified C5G7)

The results are shown in Fig. 7 and Table III. As shown in Fig. 7, the power increases rather slowly compared to the previous two problems due to the small reactivity change (~ 0.5 \$). As shown in Table III, the memory requirement in the p-CMFD acceleration is around 37.1 GB, but the computing node in this study has 16 GB per node (note that Titan Cray XK7 [20] has 32GB per node). Therefore, the p-CMFD acceleration is not possible to solve this problem, while the NLG iteration can solve it with nine parallel computing nodes thanks to the distributed memory.

The whole-core transport calculation without any cell homogenization techniques looks feasible with the parallelized NLG iteration on a proper parallel computing system, and a single assembly size would be an appropriate choice for the size of local problem, considering the communication time ratio and the computing memory per computing node.

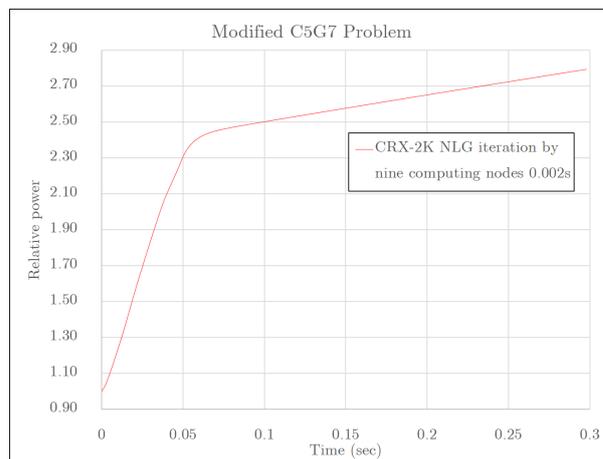


Fig. 7. Relative power vs time (modified C5G7)

Table III: Result summary (modified C5G7)

		p-CMFD acceleration	NLG iteration	
Number of computing nodes		1	1	9
k-eff		N/A	1.12833	
Computing time	Steady-state (hr)	N/A	N/A	2.27
	Transient calculations (hr)	N/A	N/A	23.9
Memory requirement (GB) per node		34.2	37.1	4.5

5. Conclusions

The NLG iteration method for transient transport calculations has been developed and implemented in CRX-2K. The main advantage of the NLG iteration is its natural parallelization, so the NLG iteration is parallelized in domain (local problems) by using MPI protocol. Each local problem is solved by an independent computing node, so the heavy transport calculations and the heavy memory requirement are distributed over the computing nodes.

Numerical results show that; 1) the NLG iteration speeds up the computing time by parallelization, and 2) the NLG iteration distributes the computing memory, so the “transport” calculation may be feasible without any cell homogenization techniques.

As future works, the following areas are identified; 1) code optimization to reduce the communication time, 2) a study to increase the time step size to reduce the transient calculation time.

References

- [1] G. S. Lee and N. Z. Cho, "2D/1D fusion method solutions of the three-dimensional transport OECD benchmark problem C5G7 MOX," *Prog. Nucl. Energy* **48**, 410 (2006).
- [2] G. S. Lee, "Development of 2-D/1-D fusion method for three-dimensional whole-core heterogeneous neutron transport calculations," Ph.D. Thesis, KAIST (2006).
- [3] H. Sutter, "The free lunch is over: a fundamental turn toward concurrency in software," *Dr. Dobbs's J.* **30**, 3 (2005)
- [4] J. E. Hoogenboom, W. R. Martin, and B. Petrovic, "Monte Carlo performance benchmark for detailed power density calculation in a full size reactor core," Benchmark specifications revision 1.1, June (2010)
<<http://www.nea.fr/dbprog/MonteCarloPerformanceBenchmark.htm>>.
- [5] M. A. Smith, E. E. Lewis, and B-C. Na, "Benchmark on deterministic transport calculations without spatial homogenization: A 2-D/3-D MOX fuel assembly 3-D benchmark," NEA/NSC/DOC(2003) 16. Organization for Economic Co-operation and Development, Nuclear Energy Agency (2003).
- [6] S. Yuk, N. Z. Cho, "Whole-core transport solutions with 2-D/1-D fusion kernel via p-CMFD acceleration and p-CMFD embedding of nonoverlapping local/global iterations," *Nucl. Sci. Eng.*, **181**, 1 (2015).
- [7] B. Cho and N. Z. Cho, "A Nonoverlapping local/global iterative method with 2-D/1-D fusion transport kernel and p-CMFD wrapper for transient reactor analysis," *Ann. Nucl. Energy* **85**, 937 (2015).
- [8] J. Y. Cho et al., "Transient capability of the DeCART code," KAERI, KAERI/TR-2930 (2005).
- [9] A. Talamo, "Numerical solution of the time-dependent neutron transport equation by the method of the characteristics," *J. Comput. Phys.*, **240**, 248 (2013).
- [10] R. E. Alcouffe et al., "Computational efficiency of numerical methods for the multigroup, discrete-ordinates neutron transport equations: the slab geometry case," *Nucl. Sci. Eng.* **71**, 111 (1977).
- [11] H. S. Joo, "Resolution of the control rod cusping problem for nodal methods," Ph.D. thesis, MIT (1984).
- [12] N. Z. Cho, "The partial current-based CMFD (p-CMFD) method revisited," *Trans. Kor. Nucl. Soc.*, Gyeongju, Korea, October 25 – 26, 2012
<http://www.kns.org/kns_files/kns/file/229%C1%B6%B3%B2%C1%F8.pdf>.
- [13] N. Z. Cho, G. S. Lee, and C. J. Park, "Partial current-based CMFD acceleration of the 2D/1D fusion method for 3D whole-core transport calculations," *Trans. Am. Nucl. Soc.* **88**, 594 (2003).
- [14] G. L. G. Sleijpen and D. R. Fokkema, "BiCGstab(l) for linear equations involving unsymmetric matrices with complex spectrum," *Electronic Transactions on Numerical Analysis*, **11**, 1 (1993)
<<http://www.emis.ams.org/journals/ETNA/vol.1.1993/pp11-32.dir/pp11-32.pdf>>.
- [15] W. Boyd et al., "The OpenMOC method of characteristics neutral particle transport code," *Ann. Nucl. Energy* **68**, 43 (2014).
- [16] H. J. Yoo and N. Z. Cho, "3D/2D rotational plane slicing method for 3D whole-core transport calculation," *Ann. Nucl. Energy* (accepted for publication) (2015).
- [17] A. Yamamoto et al., "Derivation of optimum polar angle quadrature set for the method of characteristics based on approximation error for the Bickley function," *J. Nucl. Sci. Technol.* **44**, 129 (2007).
- [18] D. Lee et al., "Application of SP3 approximation to MOX transient analysis in PARCS," *Trans. Am. Nucl. Soc.* **91**, 252 (2004).
- [19] V. F. Boyarinov, A. E. Kondrushin, and P. A. Fomichenko, "Benchmark on deterministic time-dependent transport calculations without spatial homogenisation," PHYSOR 2014, Kyoto, Japan, September 28 – October 3, 2014.
- [20] Oak Ridge Leadership Computing Facility, "Oak Ridge Leadership Computing Facility," 2012
<<http://www.olcf.ornl.gov/computing-resources/jaguar/>>.