Whole-Core Heterogeneous Transport Calculations and Their Comparison with Diffusion Results

Gil Soo Lee, Nam Zin Cho, and Ser Gi Hong

Korea Advanced Institute of Science and Technology
Department of Nuclear Engineering
373-1 Kusong-dong, Yusong-gu
Taejon, Korea 305-701

Abstract

The CRX code based on the method of characteristics is extended to treat whole-core heterogeneous calculation. Since the heterogeneous transport calculation for such a large scale problem requires large computer memory, a modular ray tracing in which all lattice cells have the same ray distribution for each direction was used to reduce the computer memory requirement. In this scheme, the ray tracing is performed on only different types of cells. Therefore, this ray tracing scheme can reduce significantly the time in tracking along neutron paths and the computer memory for storing track lengths. Also, a parallelization scheme in angular domain rather than in spatial domain and the coarse mesh/coarse group rebalance (CMR/CGR) method in inner and outer iterations were implemented for further reduction of the computing time. To show the effectiveness of the extended CRX code, it is applied to a small core (i.e., 8x8 whole core) heterogeneous calculation. The numerical results show that the extensions of CRX are effective in saving the computer memory and in reducing the computing time. A comparison with the diffusion nodal method shows maximum 4.92% difference in assembly power.

I. INTRODUCTION

Recently, the method of characteristics (MOC)([1],[2],[3],[4],[5],[6],[7],[8]) which combines desirable features of the integral transport and S_N methods has been considered as an effective methodology in the lattice calculation. This method gives accurate solutions in complex geometries, strong absorber problems, strongly anisotropic problems and so on because its calculation preserves the simplicity of the S_N method. It divides directions like in S_N and for each direction performs transport calculation like the collision probability method by integrating the differential form of the within-group transport equation along its parallel characteristic lines. For better accuracy, it needs many rays and fine angle divisions.

Like most of the transport methods, MOC also requires long computing times and large computer memory for large scale problems such as multiassemblies and whole cores. Therefore, it has been nearly impossible to use the transport theory methods in the heterogeneous calculation for large scale problems. For this reason, the original ray tracing scheme in which the ray tracing is performed on whole problem was modified so that it is sufficient to perform the ray tracing on only different types of cells. The modified ray tracing scheme requires

that all lattice cells have the same size and the angular quadrature set is selected so that all lattice cells have the same ray distribution for each direction. Therefore, the ray tracing is performed only for different cell types and it leads to significant reduction of computing time and computer memory. Also, it must be noted that with this modified scheme, the rays on external or reflective boundaries can return to the exactly same position and it leads to improvement of accuracy of the solution. To further reduce the computing time and save the computer memory, a parallelization scheme that does not require an iterative procedure was implemented by decomposing the angular domain, and coarse mesh/coarse group rebalance method (CMR/CGR) was used to reduce computing time of outer and inner iteration. The CRX code with these extensions[9] described above is applied to a small test core problem with MOX fuels. The results are compared with those of the diffusion nodal code AFEN.

II. THEORY AND METHODOLOGY

II.1 The Methodology of the CRX Code

To describe the method of characteristics, the starting equation is the within-group transport equation in discrete ordinates form :

$$\left[\hat{\Omega}_n \cdot \vec{\nabla} + \sigma_g(\vec{r})\right] \psi_g(\vec{r}, \hat{\Omega}_n) = \sum_{g'=1}^{G} \sum_{l=0}^{L} \sum_{m=-l}^{l} Y_{ml}(\hat{\Omega}_n) \sigma_{slgg'}(\vec{r}) \phi_{lg'}^{m}(\vec{r})$$

$$+ \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} (\nu\sigma_f)_{g'} \phi_{g'} = q_{g,n} \tag{1}$$

where w_n is the weight for direction $\hat{\Omega}_n$, L is the order of anisotropy of scattering ($L=3$ in the current version of CRX) and the moments ϕ_{lg}^{m} are given by

$$\phi_{lg}^{m}(\vec{r}) = \sum_{n=1}^{N} w_n Y_{ml}^{*}(\hat{\Omega}_n) \psi_g(\vec{r}, \hat{\Omega}_n), \tag{2}$$

where N is the number of directions. Eq.(1) can be rewritten in the following differential form :

$$\sin\theta_n \frac{d\psi_{g,n}}{dp} + \sigma_g \psi_{g,n} = q_{g,n}, \tag{3}$$

where p is the projected coordinate on $x - y$ plane of the coordinate along the neutron trajectory for direction $\hat{\Omega}_n$ and θ_n is the polar angle. The equation for the outgoing flux along a ray in a computational mesh with the flat source approximation is obtained by analytically integrating Eq.(3) :

$$\psi_{g,n,l}^{out} = \psi_{g,n,l}^{in} e^{-\sigma_g L_{n,l}/\sin\theta_n} + \frac{q_{g,n}}{\sigma_g}(1 - e^{-\sigma_g L_{n,l}/\sin\theta_n}), \tag{4}$$

where $L_{n,l}$ is the track length of the l'th ray for direction $\hat{\Omega}_n$ in the mesh. The average flux along the ray included in the mesh for direction $\hat{\Omega}_n$ is obtained by integrating Eq.(3). The

equation is given as follows :

$$\bar{\psi}_{g,n,l} = \frac{q_{g,n}}{\sigma_g} + \sin\theta_n \frac{(\psi_{g,n,l}^{in} - \psi_{g,n,l}^{out})}{\sigma_g L_{n,l}}. \tag{5}$$

However, to perform the scattering source iteration, the average angular flux over the computational mesh is required for generation of the source. The equation for the average flux over the computational mesh is obtained by summing the average fluxes (Eq.(5)) over the rays passing through the mesh. The equation is given as follows :

$$\bar{\psi}_{g,n} = \frac{q_{g,n}}{\sigma_g} + \frac{\sin\theta_n}{A\sigma_g} \sum_{l \in \text{mesh}(i,j)} \delta_n \left(\psi_{g,n,l}^{in} - \psi_{g,n,l}^{out}\right), \tag{6}$$

where δ_n represents the spacing between two adjacent rays for direction $\hat{\Omega}_n$, and A is the area of the mesh. In the method of characteristics, Eq.(4) and Eq.(6) with a general tracking module are complete for transport calculation.

II.2 Whole-Core Heterogeneous Calculation

In the original ray tracing scheme of the CRX code, the ray tracing is performed over the whole problem and the computed neutron track lengths for all azimuthal directions and for the whole problem are stored for use in the scattering source iteration. Therefore, the scheme can treat any angular quadrature set and any geometrical structure of the problem, although it would require large computer memory and consume long computing time for large scale problems such as the whole-core heterogeneous calculation. Thus, the scheme can be very ineffective for large scale problems that are comprised of lattice cells even with the same geometric dimensions. Therefore, we have modified the scheme so that the ray tracing can be performed only for different geometrical cell types and therefore, the computing time and computer memory are reduced significantly in the ray tracing. First, the sizes of the lattice cells are assumed to be the same. In order for all lattice cells to have the same ray distribution, the angular azimuthal quadrature set ([10], [11], see Figure 1) proposed by Filippone *et al* is selected. The azimuthal angles (φ) for first octant are determined by

$$\tan(\varphi_r) = (A/B) \times \frac{r}{(N_\varphi + 1 - r)}, \ r = 1, 2, 3, \cdots, N_\varphi, \tag{7}$$

where N_φ is the number of azimuthal angles for first octant, A and B are the sizes of the lattice cell for x and y directions, respectively. Then, the number of rays for each direction in a lattice cell is equal to $N_\varphi + 1$. Therefore, if the number of rays for one lattice cell is specified, the number of azimuthal angles for an octant is determined. The number of rays used is typically more than 20. Therefore, the number of azimuthal angles is too large and it may lead to very long computing time. Thus, we use only a subset (N_φ') of these angles selected by considering the symmetry about the 45 degree line (see Figure 1). With this scheme, the ray distributions for all lattice cells become the same. Another merit of this angular quadrature set is the improvement of the accuracy due to the return of the rays to the exactly same position on the external or reflective boundaries.

Furthermore, the rays are swept through the whole problem domain, e.g., starting from an external boundary, reflecting at a reflective boundary (if any), and then ending at an external
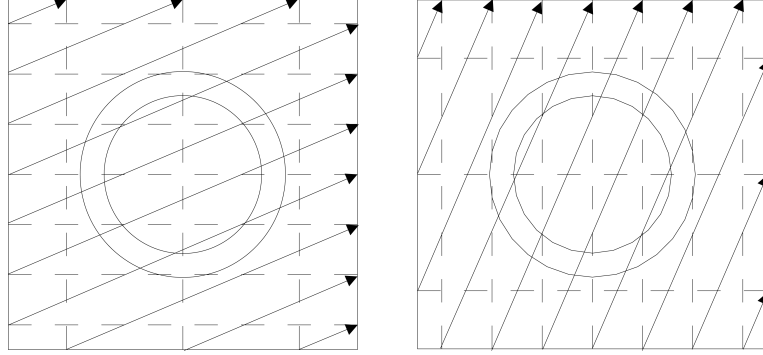
(a) Ray distribution when r=3

(b) Ray distribution when r=7
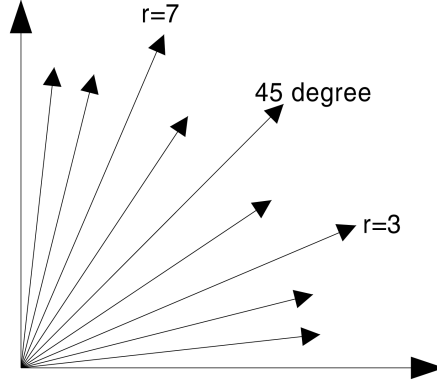
(c) Azimuthal angles (N_φ=9, N'_φ=2)

Figure 1: Ray distributions and azimuthal directions

boundary.

II.3 Parallel Computation

Parallel computation of the CRX code was performed with MPI (Message Passing Interface). The message passing model posits a set of processors that have only local memory but are able to communicate with other processors by sending and receiving messages. The sum of angular fluxes, rebalance factors, and flux moments are transferred. Each processor calculates for $N_{angle}/N_{processor}$ directions. So each computer has the following calculated results :

$$\phi_{lg}^{m,P}(\vec{r}) = \sum_{n=1}^{n \in n_P} w_n Y_{ml}^*(\hat{\Omega}_n)\psi_g(\vec{r}, \hat{\Omega}_n), \tag{8}$$

where P represents the processor number, and n_P represents the angle set which is calculated

in processor P. After this procedure, the above data are collected and summed as follows :

$$\phi_{lg}^m(\vec{r}) = \sum_P \phi_{lg}^{m,P}(\vec{r}). \tag{9}$$

Then these results are distributed to each processor for next iteration. Then leakage terms are collected to calculate rebalance factors in master processor (one of the processors). These rebalanced factors are then distributed to each processor and multiplied to all of the flux moments in each processor.

The ray sweeping strategy in Section II.2 with the angularly decomposed parallel computation does not require iteration on the assumed interface angular fluxes, in contrast to the domain decomposition parallel scheme in Ref. [12].

The efficiency of this parallel computation depends on the number of computational meshes since the communication time increases as the number of meshes increases (in our scheme, the amount of communication is proportional to the number of meshes). This method is effective in problems in which the communication time is a small portion of the total time (i.e., communication time plus computing time).

III. NUMERICAL RESULTS

We solved a 8×8 whole core problem to show the effectiveness of the CRX code with some extensions. This problem consists of eight MOX fuel assemblies and twenty four UO_2 fuel assemblies. The description of this problem is given in Figures 2 and 3. For comparison, the AFEN code that is a nodal diffusion code was applied to this problem. In fact, the CRX code and the AFEN code were applied to only 1/4 core due to the symmetry. For this problem, the material cross sections were extracted from the HELIOS code. For the AFEN code, two group cross sections and discontinuity factors were prepared by the HELIOS code through the condensation and homogenization procedure. On the other hand, the CRX code used the seven group transport corrected cross sections[13] also prepared by HELIOS. Figure 6 shows the eigenvalues and normalized assembly power distributions calculated with CRX and AFEN.

In this numerical test, 8 azimuthal and 2 polar angles per an octant, and 40 rays per a cell in each direction were used in the CRX code. The convergence criteria of 10^{-6} for eigenvalue and 10^{-5} for fission source were used. Figures 4 and 5 show the normalized cell power and 7th group flux distributions, respectively. The parallel efficiency is shown in Figure 7. The parallel speedup with 16 CPUs is 14.14, efficiency of 88% and computing time is 3568 sec, about 1 hour. We have experienced that the computing time is proportional to the number of rays but the accuracy is not sensitive to it. Therefore the computing time can be reduced further by reducing the number of rays without loss of accuracy. This parallel calculation was performed on KAIST*GALAXY cluster. KAIST*GALAXY cluster consists of dual PentiumII-300 front-end and 16 Celeron 300A compute-nodes.
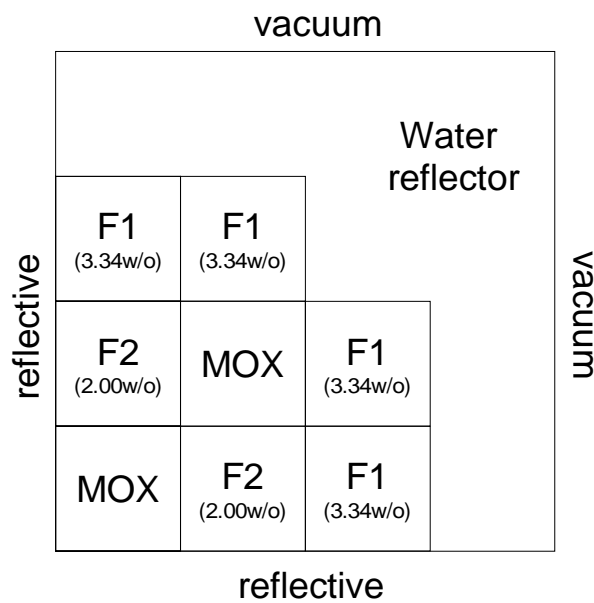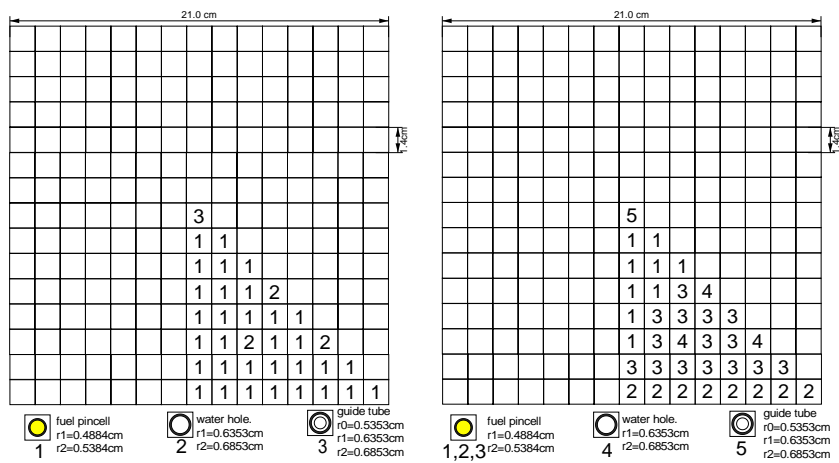
Figure 2: Core Configuration

(a) Assembly Type F1 & F2   (b) Assembly Type MOX
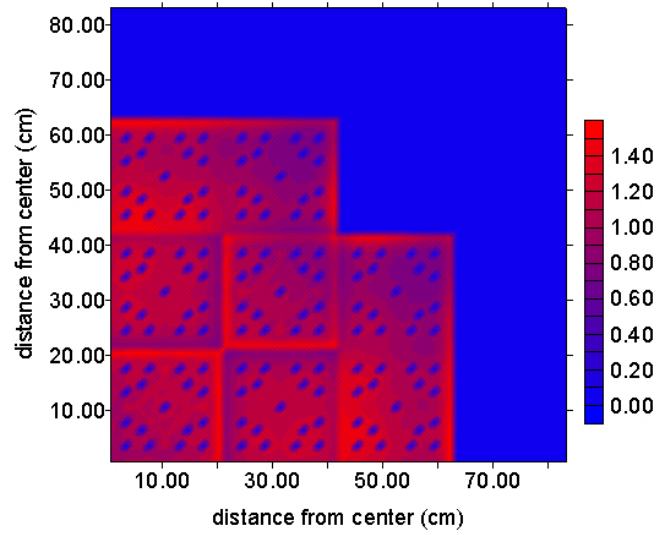
Figure 3: Assembly Description
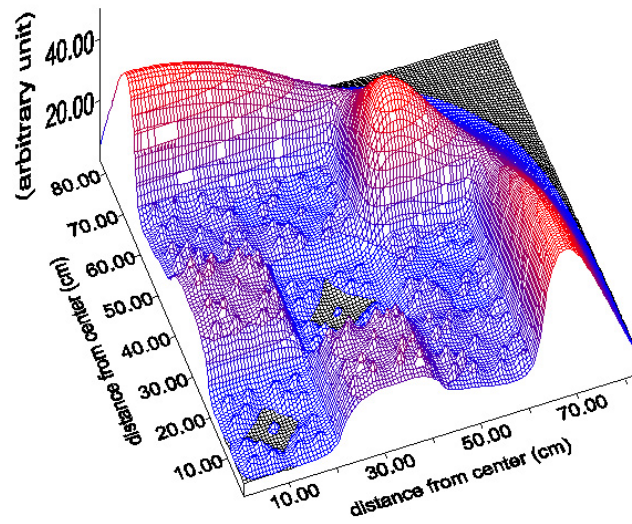
Figure 4: Normalized Cell Power Distribution

Figure 5: Cell Averaged 7th Group Neutron Flux Distribution

			k_{eff}
		CRX	1.153122
		AFEN	1.152953
			(-0.014^b)
1.109 1.158(4.42)	0.866 0.887(2.42)		
0.995 0.946(-4.92)	1.000 1.005(0.5)	0.866 0.887(2.42)	
1.060 $1.013(-4.43^a)$	0.995 0.946(-4.92)	1.109 1.158(4.42)	: CRX : AFEN

[a] power difference (%) between AFEN and CRX

[b] k_{eff} difference (%) between AFEN and CRX

$$\frac{(\text{AFEN}) - (\text{CRX})}{(\text{CRX})} \times 100$$

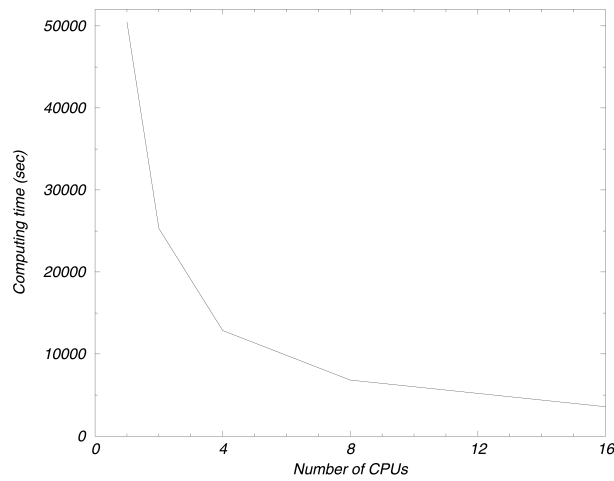Figure 6: Normalized Assembly Power Distributions and Eigenvalues

Figure 7: Computing Time versus Number of CPUs

IV. CONCLUDING REMARKS

The method of characteristics in the CRX code was extended for heterogeneous transport calculations of large scale problems. First, a modular ray tracing scheme was devised to reduce the computer memory and computing time for the neutron track lengths. With this modular ray tracing scheme, the ray tracing is performed on only different types of cells but this ray tracing scheme requires that all cells have the same dimensions. Second, parallel computation with an angular decomposition was also implemented to further reduce the computing time and computer memory. To show the effectiveness of these extensions, the CRX code with these extensions was applied to a 8×8 whole core test problem having MOX fuels. The numerical results show that the CRX code with these extensions is very effective in reducing the computer memory and the computing time. Also the results of the CRX code were compared with those of the AFEN code based on a nodal diffusion method, showing the differences of 0.014% in eigenvalue and maximum 4.92 % in assembly power.

References

[1] N. Z. Cho and S. G. Hong, "CRX : A Transport Theory Code for Cell and Assembly Calculations Based on Characteristic Method," *Proc. International Conf. on the Physics of Reactors*, September 16-20, Mito, Japan, Vol. 1A, p.250, 1996.

[2] S. G. Hong and N. Z. Cho, "CRX : A Code for Rectangular and Hexagonal Lattices Based on the Method of Characteristics," *Annals of Nuclear Energy*, **25**, 547, 1998.

[3] M. J. Halsall, "CACTUS, A Characteristics Solutions to the Neutron Transport Equations in Complicated Geometries," AEEW-R-1291, U.K. Atomic Energy Authority, 1980.

[4] M. D. Brough and C. T. Chudley, "Characteristic Ray Solutions of the Transport Equation," *Advances in Nuclear Science an Technology*, Vol. 12, Plenum Press, New York,1980.

[5] L. Goldberg, J. Vujic, A. Leonard, and R. Stachowski, "The Characteristics Method in General Geometry," *Trans. Am. Nucl. Soc.*, **112**, 16, 1992.

[6] R. Roy, "The Cyclic Characteristics Method with Anisotropic Scattering," *Proc. Int. Conf. Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*, p.1225, Madrid, Spain, September 27-30, 1999.

[7] R. E. Alcouffe and E. W. Larsen, "A Review of Characteristic Methods Used to Solve the Linear Transport Equations," *Proc. Int. Topl. Advances in Mathematical Methods for the Solution of Nuclear Engineering Problems*, Munich, FRG, April 27-29, 1981, American Nuclear Society, 1981.

[8] S. G. Hong and N. Z. Cho, "Method of Characteristic Direction Probabilities for Heterogeneous Lattice Calculation," *Nuclear Science and Engineering*, **132**, 65-77, 1999.

[9] G. S. Lee, N. Z. Cho, and S. G. Hong, "Acceleration and Parallelization of the Method of Characteristics for Lattice and Whole-Core Heterogeneous Calculations," *Proc. PHYSOR 2000*, May 7-11, 2000, Pittsburgh, USA (to be presented).

[10] W. L. Filippone, S. Woolf, and R. J. Lavigne, "Particle Transport Calculations with the Method of Streaming Rays," *Nuclear Science and Engineering*, **77**, 119, 1981.

[11] S. G. Hong and N. Z. Cho, "Extension of Streaming Rays Method for Streaming Dominant Neutron Transport Problems," *Journal of Korean Nuclear Society*, **28**, 320, 1996.

[12] S. Kosaka and E. Saji, "The Characteristics Transport Calculation for a Multi-Assembly System Using Neutron Path Linking Technique," *Proc. Int. Conf. Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*, p.1890, Madrid, Spain, September 27-30, 1999.

[13] R. Stamm'ler, J. Casal, A. Ferri and E. Villarino., "User's Manual for HELIOS," 05DEC94, Studsvik, 1994.