

Self-Collision Rebalancing Technique for the MCI Characteristics Solver

G. J. Wu and R. Roy

Institut de génie nucléaire, École Polytechnique de Montréal
P.O.Box 6079, Station CV, Montréal, Québec, Canada H3C 3A7

Abstract

In this paper, we present a new acceleration technique, called Self-Collision Rebalancing (SCR) technique, for the method of characteristics applied to solving the transport equation in general 3D geometries with isotropic boundary conditions. Isotropic sources and scattering are assumed. By scanning the tracking lines, the total incoming flux is calculated for each discretized region and is thereafter supposed constant. An energy group rebalancing algorithm is applied for each region by using his self-collision probabilities. The out-going boundary current is adjusted according to the rebalanced flux in the conjunctive region. This SCR technique needs very little programming effort for its implementation and requires very little memory. Application of the SCR method significantly accelerates the resolution of multigroup problems without disturbing the final results.

1 Introduction

The usual deterministic method used for solving the neutron transport equation in the lattice code DRAGON is the Collision Probabilities (CP) method. A new alternative solution scheme, using the Method Of Characteristics (MOC), for both 2D (module MOCC using cyclic tracking lines [1]) and 3D geometries (module MCI using non-cyclic tracking lines [2]), is under development. The differential form of the Boltzmann equation will be solved after following the tracking lines (also called “characteristics”) of the system[3]. The resolution of the transport equation over all segments of each tracking line at each iteration demands a great deal of CPU time. Compared with CP method, the MOC usually needs more iterations for the following reasons:

1. For every iteration of the multigroup MCI solver, the integration lines are swept and the flux is simultaneously obtained after sweeping all the energy groups; this Jacobi scheme is less efficient than the Gauss-Seidel scheme generally used for the CP method;
2. Neglecting the usual down-scattered form of the cross-section matrices and the disadvantages that come with the Jacobi scheme, the MCI method internally calculates the angular flux, so the self-scattering reduction, which can be used to accelerate the CP solvers in producing the scalar flux, is not directly possible;
3. Even for a fixed source one-group problem without self-scattering, the MCI method needs more than one internal iteration to converge because the incoming starting current is guessed. This means that, for a multigroup problem, even the highest energy group where the up-scattering is absent could not converge at the first iteration.

This third reason is not valid when the cyclic tracking technique[4, 5] is used, like in MOCC[1] and CACTUS[6], where no current is necessary at the boundary.

Several acceleration techniques are developed and used for the MOC. For example, CACTUS uses an energy group rebalancing algorithm through solving a homogeneous eigenvalue problem[7]. Although the MOC usually needs more iterations, MOCC can be more than 1.5 times faster than the standard CP method in DRAGON for several typical benchmarks[1] because of the less numerous set of tracking lines in 2D geometries. In the past few years, other acceleration techniques for 1D and 2D geometries, such as the two-step acceleration method[8] and the transport synthetic acceleration method[9], were also introduced and were used or could be used for the MOC. But none of these is done in 3D geometries where the number of tracking lines is significantly increased. A more robust acceleration technique is therefore needed for the MOC in 3D geometries, particularly for the high-scattering problem because of the second reason mentioned above. In this paper, we will introduce a new acceleration method, called Self-Collision Rebalancing (SCR) method. This method combines the MOC and CP method through the self-collision probabilities. Use of this method in the 3D characteristics module MCI substantially reduces the total number of iterations, particularly for the problems having a high scattering ratio such as CANDU supercell calculations.

In the next section, we will briefly present the general formalism of the MOC. We will then introduce the SCR technique which is specially developed for the MOC with isotropic boundary conditions. We will compare the MCI execution times to the standard CP method for the Mosteller benchmark in a 3D geometry and we will present the CPU benefits by using the SCR technique without losing accuracy. Some discussions will be presented in the last section.

2 The characteristics formalism

Assuming a finite domain V split into homogeneous regions, each region having a volume V_j , the average (one-group) flux Φ_j is given by:

$$\begin{aligned} V_j \Phi_j &= \int_{V_j} d^3r \int_{4\pi} d^2\Omega \Phi(\vec{r}, \hat{\Omega}) \\ &= \int_{\Upsilon} d^4T \int_{-\infty}^{+\infty} dt \chi_{V_j}(\vec{T}, t) \Phi(\vec{p} + t\hat{\Omega}, \hat{\Omega}) \end{aligned} \quad (1)$$

A characteristic line \vec{T} (tracking line) is determined by its orientation (solid angle $\hat{\Omega}$) along with a reference starting point \vec{p} for the line. To cover the Υ domain, an EQ_n angular quadrature set sustained by uniform weights is used and the starting point \vec{p} is chosen by scanning the plane $\pi_{\hat{\Omega}}$ perpendicular to the selected direction $\hat{\Omega}$. In the above, the variable t refers to the local coordinates on the tracking line and the function $\chi_{V_j}(\cdot, \cdot)$ is defined as 1 if the tracking line passes through the region j , and 0 otherwise.

For a chosen line $\vec{T} = (\hat{\Omega}, \vec{p})$, the segment lengths L_k and numbers N_k for each region encountered along the line are ordered in the traveling direction, i.e. $\{L_0, L_1, L_2, \dots, L_{K-1}\}$ and the segment lengths are renormalized to preserve the true volumes. The crossing points between regions and their corresponding angular fluxes are defined as:

$$\begin{aligned} \vec{r}_{k+1} &= \vec{r}_k + L_k \hat{\Omega} \\ \phi_k(\vec{T}) &= \Phi(\vec{r}_k, \hat{\Omega}) \end{aligned} \quad (2)$$

where \vec{r}_0 and \vec{r}_K are respectively the entry and exit point of line \vec{T} to and from the domain V . We define the segment integrated angular flux and the region integrated scalar flux by the followings:

$$L_k \bar{\phi}_k(\vec{T}) = \int_0^{L_k} dt \Phi(\vec{r}_k + t\hat{\Omega}, \hat{\Omega}) \quad (3)$$

$$V_j \Phi_j = \int_{\Upsilon} d^4T \sum_k \delta_{jN_k} L_k \bar{\phi}_k(\vec{T}) \quad (4)$$

where δ is the usual Kronecker symbol, and where the summation of k runs over all possible integers for every line. Thus, all tracking lines are accepted, but only the contributions of segments crossing region j are added together.

Assuming an isotropic source of Q neutrons/cm³/sec, the one-group neutron transport equation may be written on a tracking line \vec{T} crossing the domain as:

$$\frac{d\phi(\vec{r}_0 + s\hat{\Omega}, \hat{\Omega})}{ds} + \Sigma_{tr}(\vec{r}_0 + s\hat{\Omega})\phi(\vec{r}_0 + s\hat{\Omega}, \hat{\Omega}) = \frac{Q(\vec{r}_0 + s\hat{\Omega})}{4\pi} \quad (5)$$

where \vec{r}_0 is the entry point, s the distance measured from \vec{r}_0 on \vec{T} , and ϕ , Σ_{tr} the angular flux and the transport-corrected total macroscopic cross-section.

By assuming a known incoming angular flux $\phi(\vec{r}_0, \hat{\Omega}) = \phi_0$ and constant properties $\Sigma_{tr,j}$, Q_j in each region V_j , one can easily obtain, from (5), the angular flux value at each crossing point and the averaged flux at each segment:

- when $\Sigma_{tr,N_k} \neq 0$,

$$\phi_{k+1} = \phi_k e^{-\tau_k} + q_{N_k} (1 - e^{-\tau_k}) \quad (6)$$

$$\bar{\phi}_k(\vec{T}) = (\phi_k - q_{N_k}) \frac{1 - e^{-\tau_k}}{\tau_k} + q_{N_k} \quad (7)$$

- when $\Sigma_{tr,N_k} = 0$,

$$\phi_{k+1} = \phi_k \quad (8)$$

$$\bar{\phi}_k = \phi_k \quad (9)$$

where we used $\tau_k = \Sigma_{tr,N_k} L_k$ and $q_j = \frac{Q_j}{4\pi\Sigma_{tr,j}}$ in the above equations.

When the averaged flux is available for all the segments, the region integrated flux can be easily computed by combining equations (4) and (7) which gives (assuming $\Sigma_{tr,j} \neq 0$):

$$\Sigma_{tr,j} V_j \Phi_j = \int_{\Upsilon} \sum_k \delta_{jN_k} (\phi_k - q_j) (1 - e^{-\tau_k}) + \Sigma_{tr,j} V_j q_j \quad (10)$$

At the boundary, we will consider an isotropic reflection condition:

$$J_{-, \alpha} = \beta_{\alpha} J_{+, \alpha} \quad (11)$$

where S_{α} is a sub-surface of ∂V and β_{α} the *albedo* factor on S_{α} . For a tracking line \vec{T} whose entry point \vec{r}_0 is on S_{α} , we use

$$\phi_0 = \frac{4}{S_{\alpha}} J_{-, \alpha}, \quad (12)$$

as the incoming flux value at \vec{r}_0 on \vec{T} .

The out-going current can be obtained by:

$$\begin{aligned} J_{+, \alpha} &= \int_{S_{\alpha}} d^2 \vec{r}_b \int_{\hat{\Omega} \cdot \vec{N} > 0} d^2 \Omega \hat{\Omega} \cdot \vec{N} \phi(\vec{r}_b, \hat{\Omega}) \\ &= \int_{\Upsilon} d^4 T \chi_{S_{\alpha}}(\vec{T}, t_K) \phi_K \quad \text{with } t_K = |\vec{r}_K - \vec{p}| \end{aligned} \quad (13)$$

where \vec{N} is the outer normal at the point $\vec{r}_b \in S_{\alpha}$.

We start the multigroup iteration scheme by guessing initial flux and incoming current. We compute the integrated angular flux on all the segments of the tracking lines and we sum the contributed values to obtain the region averaged flux and the out-going currents which will be used to evaluate the scattering source and the in-coming currents for the next iteration. This procedure will continue until the convergence is achieved. A global rebalancing schema and an one-parameter acceleration schema are used in the multigroup iteration scheme.

This method of characteristics is implanted as a solver named MCI in the DRAGON code. In a previous paper, it was shown that numerical results were similar to those of the CP solver EXCELL[2].

3 Self-Collision Rebalancing Technique

In this section, we will describe the new acceleration scheme called the Self-Collision Rebalancing (SCR) technique.

We can rewrite the multigroup form of (3) as following:

$$\tau_k^g \bar{\phi}_k^g(\vec{T}) = \phi_k^g(\vec{T})(1 - e^{-\tau_k^g}) + \frac{Q_{N_k}^g}{4\pi \Sigma_{t,N_k}^g} (-1 + \tau_k^g + e^{-\tau_k^g}) \quad (14)$$

From (4), the region average flux in V_j can be obtained by:

$$\begin{aligned} \Phi_j^g &= \frac{1}{\Sigma_{t,j}^g V_j} \int_{\Upsilon} d^4T \sum_k \delta_{jN_k} \phi_k^g(\vec{T}) (1 - e^{-\tau_k^g}) \\ &+ \frac{Q_j^g}{4\pi \Sigma_{t,j}^g V_j} \int_{\Upsilon} d^4T \sum_k \delta_{jN_k} \frac{-1 + \tau_k^g + e^{-\tau_k^g}}{\Sigma_{t,j}^g} \end{aligned} \quad (15)$$

We introduce the following notations:

$$\Phi_{j,\text{in}}^g = \frac{1}{\Sigma_{t,j}^g V_j} \int_{\Upsilon} d^4T \sum_k \delta_{jN_k} \phi_k^g(\vec{T}) (1 - e^{-\tau_k^g}) \quad (16)$$

$$p_{jj}^g = \frac{1}{4\pi \Sigma_{t,j}^g V_j} \int_{\Upsilon} d^4T \sum_k \delta_{jN_k} \frac{-1 + \tau_k^g + e^{-\tau_k^g}}{\Sigma_{t,j}^g} \quad (17)$$

so that (15) becomes:

$$\Phi_j^g = \Phi_{j,\text{in}}^g + p_{jj}^g Q_j^g \quad (18)$$

For a multigroup problem, the source is composed of fission source (or external source) F_j^g and scattering source as shown in the following:

$$Q_j^g = \sum_{g'=1}^G \Sigma_{s,j}^{g \leftarrow g'} \Phi_j^{g'} + F_j^g \quad (19)$$

By substituting (19) in (18), we obtain a G dimensional linear system for each region V_j :

$$\sum_{g'=1}^G (\delta_{gg'} - p_{jj}^g \Sigma_{s,j}^{g'-g'}) \Phi_j^{g'} = \Phi_{j,\text{in}}^g + p_{jj}^g F_j^g \quad (20)$$

We now calculate the out-going current from the sub-surfaces. From (6) and (13), we obtain:

$$\begin{aligned} J_{+, \alpha}^g &= \int_{\Upsilon} d^4T \chi_{S_\alpha}(\vec{T}, t_K) \left(\phi_K^g(\vec{T}) e^{-\tau_K^g} + \frac{Q_{N_K}^g}{4\pi \Sigma_{t, N_K}^g} (1 - e^{-\tau_K^g}) \right) \\ &= J_{\alpha, 0}^g + P_{j_\alpha \alpha}^g V_{j_\alpha} Q_{j_\alpha}^g \end{aligned} \quad (21)$$

where the index j_α indicates the last region encountered before sub-surface S_α and where the following notations are used:

$$J_{\alpha, 0}^g = \int_{\Upsilon} d^4T \chi_{S_\alpha}(\vec{T}, t_K) \phi_K^g(\vec{T}) e^{-\tau_K^g} \quad (22)$$

$$P_{j_\alpha \alpha}^g = \frac{1}{4\pi \Sigma_{t, j_\alpha}^g V_{j_\alpha}} \int_{\Upsilon} d^4T \chi_{S_\alpha}(\vec{T}, t_K) (1 - e^{-\tau_K^g}) \quad (23)$$

For the CP method, the reduced collision probability in a 3D domain can also be calculated after integrating over all tracking lines:

$$\begin{aligned} p_{ij} &= \frac{1}{V_i} \int_{V_j} d^3r \int_{V_i} d^3r' \frac{\exp(-\tau(\vec{r}', \vec{r}))}{4\pi |\vec{r}' - \vec{r}|^2} \\ &= \frac{1}{4\pi V_i} \int_{\Upsilon} d^4T \int_{s \in V_j} ds \int_{s' \in V_i} ds' e^{-\tau\{s' \leq s\}} \end{aligned} \quad (24)$$

after the global change of variables $d^3r d^3r' = d^4T ds ds' |\vec{r}' - \vec{r}|^2$. The optical path contributions $\tau(s' \leq s)$ from i to j are dependent on the local coordinates (s', s) of each tracking line.

Assuming a region V_j is convex, i.e. a tracking line can cross this region only once, and the intersected segment with this region is numbered to k_0 . We

consider now the reduced collision probability from V_j to itself, also called Self-Collision probability. The integrand of (24) becomes then:

$$\begin{aligned}
 & \int_{t_{k_0-1}}^{s_{t_0}} ds' \int_{s'}^{t_{k_0}} ds \exp(-\Sigma_{tr,j}(s - s')) \\
 &= \frac{L_{k_0}}{\Sigma_{tr,j}} - \frac{1 - e^{-\Sigma_{tr,j}L_{k_0}}}{\Sigma_{tr,j}^2} \\
 &= \sum_k \delta_{jN_k} \frac{-1 + \Sigma_{tr,j}L_k + e^{-\Sigma_{tr,j}L_k}}{\Sigma_{tr,j}^2} \tag{25}
 \end{aligned}$$

where t_k is the local coordinate of \vec{r}_k on \vec{T} . From (24) and (25), we conclude that: the quantity p_{jj}^g defined by (17), if the region V_j is convex, is nothing else but the reduced collision probability from V_j to itself for the group g .

When V_j is not convex, the p_{jj}^g will be different from the conventional reduced collision probability. But the neutrons traveling between more than two segments of the same region will not be lost, they will be considered as neutrons coming from outside of the region and will be added to $\Phi_{j,in}^g$.

Based on the same assumption, we can show that the $P_{j\alpha,\alpha}$, defined by (23), is just the leakage probability of region j_α through the sub-surface S_α .

For a given source F_j^g (fission source or external source) and a guessed flux (or the flux obtained at the last iteration), the computation procedure is the following:

- Update the source Q_j^g by (19) and evaluate the $\Phi_{j,in}^g$ and $J_{\alpha,0}^g$ by the method of characteristics;
- Solve the linear system (20) for each region in order to locally rebalance the flux Φ_j^g ;
- Update the source Q_j^g by using the rebalanced flux obtained at the last step and calculate the out-going current $J_{+,\alpha}$ using (21) for all sub-surfaces.

The system (20) is solved by the Gauss-Seidel iterative scheme. The results of the rebalancing system (20) do not need to be converged completely. The convergence criteria is therefore set to be a tenth of the maximum flux error observed at the last inner iteration and the total iteration number for resolving (20) is limit to 5.

The SCR technique is totally consistent with the MCI method for the following reasons:

- A CP technique is used for each region and the CP method was already shown equivalent to the MCI method with the same tracking file[1];
- No approximation is made at interfaces, so the solution converge theoretically to the exact solution when the number of angles and the density of tracking lines go to infinity.

This SCR technique can be easily implemented and needs only one more diagonal matrix to be saved, which has the size of flux-current vector. Computing the self-collision probabilities is much easier than computing the conventional CP. The CPU consumed for computing the self-collision probabilities is comparable to that consumed in an inner iteration.

4 Numerical results

Other techniques are also used to reduce the execution time:

- A pre-calculated table of the exponential function is used over an optical track length up to 15.5 cm with a tabulation interval of 1/512 cm[7];
- the total inner iterations for the n -th outer iteration is limit to n , because requiring full inner convergence for the first outer iterations would just be wasteful.

A. Mosteller benchmark

The Doppler coefficient of reactivity has been analyzed at pin-cell level by Mosteller et al.[10]. The PWR cell has a pitch of 1.2609 cm with an outer cladding radius of 0.45802 cm and a fuel radius of 0.39306 cm. Using an 89-group ENDF/B-V library provided by AECL Chalk River Laboratory, we first performed the self-shielding of cross sections with the standard SHI module of DRAGON using a Livolant-Jeanpierre correction using collision probabilities on the non-discretized geometry and isotropic returns at the external faces[11]. The cases studied include fuel pins of five different enrichments at each of two fuel temperatures, 600K and 900K.

This 2D standard benchmark was used to test the module MOCC, which is a 2D solver of the code DRAGON based on cyclic characteristics technique, and the results obtained were compared to those of CASMO-4[1]. In order to generate a similar 3D test, the third dimension of this benchmark is extended from 0 cm to 1 cm with isotropic reflection at the boundary. A series of calculations with different quadrature degrees and different tracking densities were carried out for each option of the benchmark. We remark that the results are converged for the EQ_6 and 160 lines/cm² and they were presented at the [Table 1](#) and were compared with those of the standard CP solver of DRAGON. By using the same set of tracking lines, few differences are observed in the results, this numerically confirm the equivalence of these two methods[1].

The CPU aspect on an IBM workstation (RISC-6000-43P-140 at 233MHz) is shown for both MCI and DRAGON calculations. A fixed quadrature EQ_6 is used and the tracking line density varies from 20 lines/cm² to 500 lines/cm². The CPU shown in the [Figure 1](#) is the average over all options of the benchmark for each set of tracking lines. The MCI module is as fast as the CP method implemented in DRAGON for this benchmark. We also remark that, for a fixed geometry, execution time increases almost linearly with the num-

Fuel Enrichment (wt%)	Fuel Temperature (K)	MCI	DRAGON
0.711	600	0.66945	0.66945
	900	0.66169	0.66169
1.6	600	0.96537	0.96542
	900	0.95538	0.95542
2.4	600	1.10377	1.10382
	900	1.09309	1.09314
3.1	600	1.18197	1.18202
	900	1.17101	1.17105
3.9	600	1.24532	1.24536
	900	1.23376	1.23379

Table 1: Eigenvalues from MCI with SCR

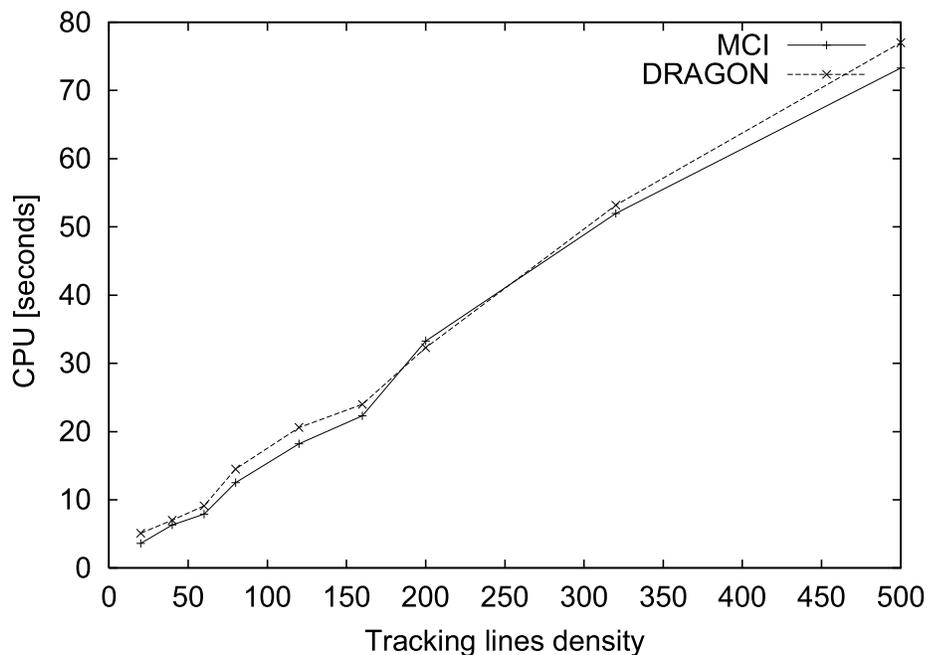


Figure 1: Mean execution times for MCI and DRAGON on a IBM workstation of type RISC-6000-43P-140 at 233MHz

ber of tracking lines for both methods. For a CP method, the CPU is mainly consumed by the formulation of the CP matrices and the resolution of the obtained systems takes very little time. For the MOC, the self-collision probabilities are calculated without an important effort like for the CP method. On the other hand, at each sweep of the tracking lines, exponential calculations are done following all lines and at each segment. Consequently, the iteration scheme consumes most of the CPU time.

B. Supercells of G2

We now consider some 3D Gentilly-2 supercells[12]. Each of these supercells is composed of two horizontal fuel bundles and one vertical adjuster with a symmetry factor 4. Six types of stainless steel adjuster rods are studied each at a series of different residence time (7 burnup steps) of the fuel in the core. Tracking is performed by EXCELT module of the lattice code DRAGON with isotropic reflection option. A EQ_4 angular quadrature and a 2.5 lines/cm² tracking density on the perpendicular plane were used. The transport equation is then solved by a critical buckling search.

The variations of the cross section properties were compared with those of CP solver of DRAGON. We now want to show the accuracy and the efficiency of the SCR technique. The multigroup diffusion coefficients are calculated with and without the use of SCR technique, the relative differences are drawn in the [Figure 2](#) for all the six adjuster types. The difference is always less than 10^{-6} , which means that the SCR technique does not significantly affect the accuracy of the MCI module in these cases.

For each type of adjuster, the multigroup iteration is started by a flat flux for the fresh fuel and is started with the last obtained flux at the other residence time of the fuel. At each residence time, two calculations are done: one with the adjuster completely insert and one with the adjuster completely extracted. The number of iterations presented on [Table 2](#), which is the sum of the two calculations, are averaged over all the residence time except the first calculation with fresh fuel. The CPU used for solving the system (20) is usually much smaller than that used for the step calculations on the tracking line segments. For rough timing estimation, we can suppose that an iteration uses the same quantity of CPU either with or without the SCR technique. In taking account of the self-collision's evaluation, the SCR technique makes a gain from 32% to 55% on CPU and these without changing the final results.

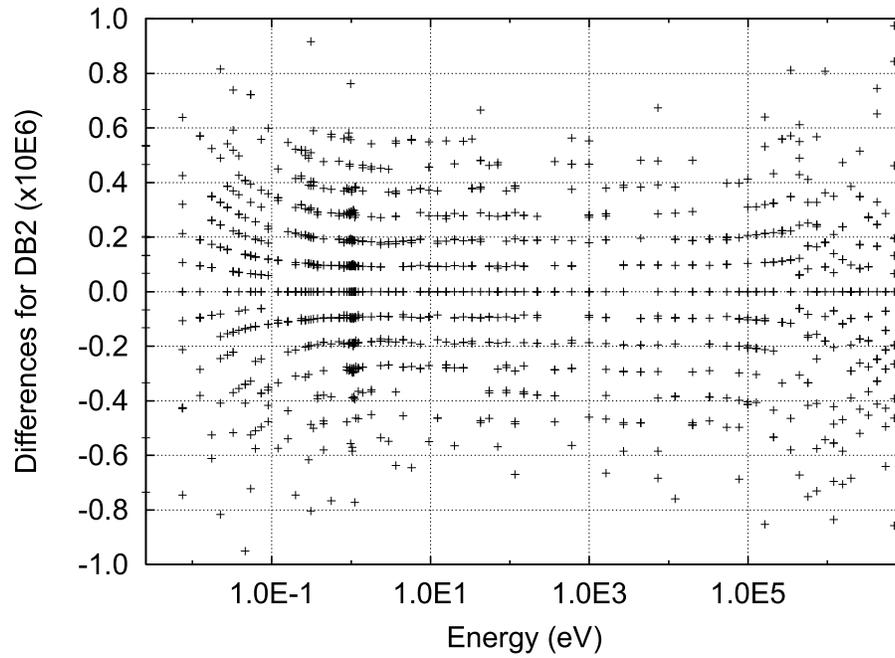


Figure 2: Relative difference for the multigroup diffusion coefficient

Adjuster type	It. nb. for the First Calculation		It. nb. for the Next Calculations	
	MCI	SCR	MCI	SCR
BCAINT	112	52	59.5	37.0
BCAOUT	125	51	58.7	37.7
BCBINT	126	53	58.0	38.7
BCCINT	115	52	55.2	37.0
BCCOUT	117	48	52.7	37.7
BCDINT	120	51	57.2	32.8

Table 2: Iteration numbers for CANDU reactor's adjuster calculation

5 Conclusions

The SCR technique is easy to implement, only the self-collision probabilities need to be calculated which requires much less effort than the standard CP. These probabilities are stored in a vector which has the size of a flux-current vector. Use of the SCR technique does not affect the final results because of the intrinsic consistency between the MOC and the CP method. In spite of its simplicity, the SCR technique considerably accelerated the multigroup resolution of the neutron transport problem by MOC in a 3D geometry. However, use of this technique in cyclic characteristics method in 2D geometries does not seem to yield much success. The reason for this is probably that, in two dimensions, the cyclic characteristics method does not suffer convergence trouble caused by the boundary current.

Although the MCI module observed a comparable speed to the standard CP method for the Mosteller benchmark, the execution time for the resolution of a CANDU supercell problem remains 4 to 5 times longer than the standard CP method. This is because of the intrinsic high-scattering ratio of the heavy water used in CANDU reactor. More effort will be invested to accelerate the resolution of the CANDU supercell problem.

The tracking lines are generated by the EXCELT module in DRAGON and stored in a file. MCI will read these tracking lines from the file when it needs them. For a problem involving many regions, the size of this file may be very large and exceed the machine storage capacity. For the next step, we will integrate the capacity of generating tracking lines in the MCI module, and the MCI will generate the tracking lines, instead of read them from a file, each time when it needs them. Once these developments are available, accurate solutions to very large 3D transport problems should become affordable.

Acknowledgments

This work has been carried out partly with the help of a grant from the Natural Science and Engineering Research Council of Canada.

References

- [1] R. Roy, "The Cyclic Characteristics Method," *Int. Conf. Physics of Nuclear Science and Technology*, Long Island, October 5–8, 1998.
- [2] G. J. Wu and R. Roy, "A New Characteristics Algorithm for 3D Transport Calculations," *Proc. 19th CNS Annual Conf.*, Toronto, Ontario, Canada, October 18-21 (1998).
- [3] J. R. Askew, "A Characteristics Formulation of the Neutron Transport Equation in Complicated Geometries," Report AEEW-M 1108, UK-AEE, Winfrith (1972).
- [4] R. Roy, G. Marleau, A. Hébert and D. Rozon, "A Cyclic Tracking Procedure for Collision Probability Calculations in 2-D Lattices," *Top. Meet. in Advances in Mathematics, Computations, and Reactor Physics*, Pittsburgh, Pennsylvania, April 28 - May 1, 1991
- [5] R. Roy, "Anisotropic Scattering for Integral Transport Codes. Part 2. Cyclic Tracking and its Application to XY Lattices," *Ann. Nucl. Energy*, **18**, 511-524 (1991).
- [6] M. J. Halsall, "CACTUS, A Characteristics Solution to the Neutron Transport Equation in Complicated Geometries," Report AEEW-R 1291, United Kingdom Atomic Energy Establishment, Winfrith (1980).
- [7] M. J. Halsall, "WIMS8 - Speed with accuracy" *Int. Conf. Physics of Nuclear Science and Technology*, Long Island, October 5–8, 1998.

- [8] E. W. Larsen and W. F. Miller, Jr., "A Two-Step Acceleration Method for Transport Problems," *Trans. Am. Nucl. Soc.*, **52**, 416 (1986).
- [9] M. R. Zika and M. L. Adams, "Transport Synthetic Acceleration for the Long-Characteristics Discretization," *Joint Int. Conf. on Mathematical Methods and Supercomputing for Nuclear Applications*, Saratoga Springs, NY, October 5-9, (1997).
- [10] R. D. Mosteller, L. D. Eisenhart, R. C. Little, W. J. Eich and J. Chao, "Benchmark Calculations for the Doppler Coefficient of Reactivity," *Ann. Nucl. Energy*, **107**, 265-271 (1991).
- [11] G. Marleau, A. Hébert and R. Roy, "A User's Guide for DRAGON," Report IGE-174 Rev. 3, École Polytechnique de Montréal (December 1997).
- [12] R. Roy *et al*, "Modelling of CANDU Reactivity Control Devices with the Lattice Code DRAGON", *Ann. Nucl. Energy*, **21**, 115-132 (1994).