21st Annual Conference of the Canadian Nuclear Society
Toronto, Ontario, Canada / June 11-14, 2000

New Development of the Three-Dimensional Characteristics Solver MCI in DRAGON

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
E-mail: wu@meca.polymtl.ca

Abstract

In this paper, we present the recent development of our 3D characteristics solver MCI. The Self-Collision Rebalancing (SCR) technique uses the self-collision probabilities to reduce the total iteration number. The proof of equivalence between the Collision Probabilities (CP) method and the characteristics method is given for finite 3D domains. The Track Merging Technique (TMT) is another new technique used by our characteristics solver. The neighboring lines are merged together in order to reduce the total number of lines. The TMT is an approximation of second order and can also be used by other tracking lines dependent solver.

1 Introduction

DRAGON[1] is a lattice code which uses the Collision Probabilities (CP) method for solving the neutron transport equation in a arbitrary geometrical domain. As an alternative solution scheme, the Method Of Characteristics (MOC) is currently under development for both 2D (module MOCC[9], using cyclic tracking lines) and 3D geometries (module MCI[2, 3], using non-cyclic tracking lines). The main goal of this development is to obtain the capability to treat large problems with a very accurate method for reference calculations.

The MOC has been used in the nuclear physics since the seventies. Because of its intrinsic properties, most of the standard acceleration techniques could not be used (namely diffusion synthetic acceleration cannot be directly used in most MOC codes due to their restricted geometrical application) or are not efficient enough [3]. Several acceleration techniques have been developed which are particular to the MOC. CACTUS, the cyclic characteristics solver of WIMS, has used the same algorithm for many years: the energy group rebalancing algorithm [6]. This algorithm consists in solving a homogeneous problem at the end of every iteration and rescaling the characteristics flux according to the homogeneous solution. Larger spatial problems benefit less from the energy rebalancing of the homogeneous calculation because the spatial dependence of the flux solution converges much more slowly. Zika and Adams employ a Transport Synthetic Acceleration (TSA) in their long characteristics code [7]. Like others synthetics methods, the idea is to solve a low-order approximation which is represented using a simplified transport operator in order to accelerate the high-order approximation which is the original transport operator. The low-order equation used in TSA is a modified transport problem in which the scattering cross section is artificially reduced. The low-order equation is also solved by a long characteristics method but using a cruder angular quadrature and a coarser ray spacing than the high order problem. These two simplifications significantly reduce the number of unknowns in the low order equations and result in considerable savings in computational cost. Unfortunatly, for several categories of problems, restriction and prolongation operations must be defined in order to map between the high and low order boundary angular grids.

In Korea, the Angular Dependent Rebalance (ADR) iteration method was studied when applying to the Step Characteristic (SC) scheme for transport problems in equilateral triangular meshes where the unknowns are surface-averaged and the volume-averaged quantities [10]. The rebalancing factor is only defined on the edges of the triangular meshes and its angular dependency is assumed to be uniform for each sextant. However, it will be very difficult to generate the ADR method for an arbitrary geometry and the accuracy of the SC scheme is not very good. In the environment of the interface-current code TDT (part of APOLLO2), Sanchez and Chetaine have developed a characteristics method for unstructured two-dimensional geometries [8]. Based on piecewise uniform and isotropic approximations for cell entering and exiting fluxes, they have developed a synthetic acceleration technique for their characteristics method. However, the synthetic acceler-

ation equations are nonsymmetrical and the size of the system of equations is very large. There are two methods for the inner iteration acceleration in the MCCG code (Method of Characteristics in Complicated Geometry): the Current/Flux Ratio method (CFR) and the Consistent Diffusion Differencing method (CDD) [5]. The first one is based upon the ratio of the outgoing one-side current and the total flux after transport sweep. The second one is generated using the consistent diffusion differencing approach for ray-tracing [4].

We have also developed an acceleration method for our characteristics module MCI: the Self-Collision Rebalancing method (SCR). Based on the equivalence of the Collision Probabilities (CP) method and the Method Of Characteristics (MOC), the SCR uses the self-collision probabilities (collision probabilities from one region to itself) in order to rebalance the energy distribution of the scalar flux for each region separately. A new technique now used by our characteristics solver is called the Track Merging Technique (TMT) of the tracking lines. Using this technique, two neighboring tracking lines crossing the same regions in the same order are merged together for one line associated with the sum of the weights of each of them. The TMT method is generally very efficient because more than a half of tracking lines are merged together without loosing accuracy.

The equivalence of the CP method and the MOC was already proved for infinite domains[9]. In this paper, we will first show a similar equivalence but for finite domains where isotropic boundary conditions apply on all external surfaces. We will then develop the SCR method, and show that it can be used in conjunction with our new TMT approximation. We will also demonstrate that this new merging technique provides an approximation of order two for the segment-length differences over successive tracking lines. Numerical results will show the consistence of the SCR method and the performance of TMT acceleration on the usual MCI solver.

2 Characteristics in Finite 3D Domains

Assuming a finite domain split into homogeneous regions V_j, the average (group-dependent) flux Φ_j is given by:

$$\begin{aligned} V_j\Phi_j &= \int_{V_j} d^3r \int_{4\pi} d^2\Omega\, \Phi(\vec{r},\hat{\Omega}) \\ &= \int_{\Gamma} d^4T \int_{t_0}^{t_K} dt\, \chi_j(\vec{T},t)\Phi(\vec{p}+t\hat{\Omega},\hat{\Omega}) \end{aligned} \tag{1}$$

The boundary of the domain is also split into surfaces S_α, and the outward surface current $J_{+,\alpha}$ is given by:

$$\begin{aligned} J_{+,\alpha} &= \int_{S_\alpha} d^2r_S \int_{2\pi} d^2\Omega\, \hat{\Omega}\cdot\hat{N}_+\Phi(\vec{r}_S,\hat{\Omega}) \\ &= \int_{\Gamma} d^4T\, \chi_\alpha(\vec{T},t_K)\Phi(\vec{p}+t_K\hat{\Omega},\hat{\Omega}) \end{aligned} \tag{2}$$

A characteristics line $\vec{T}$ is determined by its orientation $\hat{\Omega}$ along with a reference starting point $\vec{p}$ for the line. The variable t refers to the local coordinates on the tracking line and the function $\chi_j(\vec{T}$,t) (the function $\chi_\alpha(\vec{T}$,t)) is defined as 1 if the point $\vec{p}+t\hat{\Omega}$ (the point $\vec{p}+t_K\hat{\Omega}$) on the line $\vec{T}$ is in the region V_j (on the surface S_α), and 0 otherwise. We assume that a tracking line $\vec{T}$ changes regions K times when going through the domain, and the crossing points $\vec{r}_k = \vec{p}+t_k\hat{\Omega}$ are ordered in the neutron traveling direction. The Γ domain is covered by a quadrature set of solid angles and by scanning the plane $\pi_{\hat{\Omega}}$ perpendicular to the selected direction $\hat{\Omega}$ for the starting point $\vec{p}$. The d^4T element is then composed of a solid angle element $d^2\Omega$ multiplied by the corresponding plane element d^2p.

For a chosen line $\vec{T}$, the following data are required for the characteristics calculation:

- $L_k = t_k - t_{k-1}$: the length of the segment k;
- N_k: the region number of the segment k for $k = 1,\ldots,K$,
  the entering surface number N_0 and the exit surface number N_{K+1}.

The angular fluxes on the crossing points are defined as:

$$\phi_k(\vec{T}) = \Phi(\vec{r}_k,\hat{\Omega}), \qquad k = 0,1,\ldots,K. \tag{3}$$

For each segment on the line, we define the integrated angular flux as:

$$L_k\bar{\phi}_k(\vec{T}) = \int_0^{L_k} dt\, \Phi(\vec{r}_{k-1} + t\hat{\Omega}, \hat{\Omega}), \qquad k = 1, \ldots, K. \tag{4}$$

Using the above definitions, the integrated flux in Eq. (1) becomes:

$$V_j\Phi_j = \int_\Gamma d^4T \sum_{k=1}^{K} \delta_{jN_k} L_k \bar{\phi}_k(\vec{T}) \tag{5}$$

and the surface current in Eq (2) becomes:

$$J_\alpha = \int_\Gamma d^4T\, \delta_{\alpha N_{K+1}} \phi_K(\vec{T}) \tag{6}$$

where δ is the usual Kronecker symbol. Note that the value of K depends of the line $\vec{T}$. The segment lengths are renormalized to preserve the true volumes, i.e. to insure the following equality:

$$V_j = \int_{\pi_{\hat{\Omega}}} d^2p \sum_{k=1}^{K} \delta_{jN_k} L_k. \tag{7}$$

In order to simplify the writing of the equations, the following local notations are introduced:

$$\sigma_k = \Sigma_{N_k}, \qquad q_k = \frac{Q_{N_k}}{4\pi}. \tag{8}$$

The differential transport equation for angular flux on the segment k is then:

$$\left(\frac{d}{ds} + \sigma_k\right) \Phi(\vec{r}_{k-1} + s\hat{\Omega}, \hat{\Omega}) = q_k \tag{9}$$

for $s \in [0, L_k]$. The general solution to (9) on the segment is:

$$\Phi(\vec{r}_{k-1} + t\hat{\Omega}, \hat{\Omega}) = \phi_{k-1}(\vec{T})e^{-\sigma_k t} + q_k \frac{1 - e^{-\sigma_k t}}{\sigma_k} \tag{10}$$

Knowing the inward angular flux $\phi_{k-1}(\vec{T})$, the outward angular flux value at $t = L_k$ can thus be computed as:

$$\phi_k(\vec{T}) = \phi_{k-1}(\vec{T})e^{-\sigma_k L_k} + q_k \frac{1 - e^{-\sigma_k L_k}}{\sigma_k} \tag{11}$$

The integrated angular flux can also be found after integration of (10) over the segment k:

$$L_k\bar{\phi}_k(\vec{T}) = \phi_{k-1}(\vec{T})\frac{1-e^{-\sigma_k L_k}}{\sigma_k} + q_k\left(\frac{L_k}{\sigma_k} - \frac{1-e^{-\sigma_k L_k}}{\sigma_k^2}\right) \tag{12}$$

In void regions, i.e. $\sigma_k = 0$, we have:

$$\bar{\phi}_k(\vec{T}) = \phi_k(\vec{T}) = \phi_{k-1}(\vec{T}) \tag{13}$$

In the practice, the repeated computation of the exponential function is particularly costly for the characteristics solver. Therefore, we have adopted the tabulated exponentials used in CACTUS of WIMS8 [6]. However, the exact exponentials can also be chosen by the user and, in that case, the tabulated exponentials are used in the first iterations and the resolution is completed with the exact exponentials.

To complete the characteristics resolution, we introduce the following isotropic reflection boundary condition:

$$J_{-,\alpha} = J_{+,\alpha} \tag{14}$$

$$\phi_0(\vec{T}) = \frac{1}{\pi S_\alpha} J_{-,\alpha} \tag{15}$$

where S_α is the entering surface of the line $\vec{T}$.

3 Equivalence with Collision Probabilities Solver

To simplify the notations, we introduce the following segment-dependent factors for $\sigma_k \neq 0$:

$$c_k(\vec{T}) = \frac{1-e^{-\sigma_k L_k}}{\sigma_k} \tag{16}$$

$$d_k(\vec{T}) = \frac{L_k}{\sigma_k} - \frac{1-e^{-\sigma_k L_k}}{\sigma_k^2} \tag{17}$$

For void regions, i.e. $\sigma_k = 0$, the above factors are defined by their limit values respectively:

$$c_k(T) = L_k, \qquad d_k(\vec{T}) = \frac{1}{2}L_k^2 \tag{18}$$

Our intent is now to recursively add all contributions appearing in the generic term $L_k\bar{\phi}_k(\vec{T})$ and explicitly eliminate, from the formula, the flux value $\phi_k(\vec{T})$ on the crossing points $\vec{r}_k$ for $k = 1, 2, \ldots, K$. We thus start from:

$$L_k\bar{\phi}_k(\vec{T}) = q_k d_k(\vec{T}) + \phi_{k-1}(\vec{T})c_k(\vec{T}) \tag{19}$$

The inward angular flux in the above equation may be eliminated explicitly by successively applying the following recurrence relation:

$$\phi_l(\vec{T}) = \phi_{l-1}(\vec{T})e^{-\sigma_l L_l} + q_l c_l(\vec{T}) \tag{20}$$

for $l = k-1, k-2, \ldots, 1$. We end up with an equation without intermediate angular flux values on the crossing points:

$$L_k\bar{\phi}_k(\vec{T}) = q_k d_k(\vec{T}) + \sum_{l=1}^{k-1} q_l c_l(\vec{T})e^{-\tau_{l,k-1}}c_k(\vec{T}) + \phi_0(\vec{T})e^{-\tau_{1,k-1}}c_k(\vec{T}) \tag{21}$$

where τ_{nk} is the optical path defined by:

$$\tau_{nk} = \begin{cases} \displaystyle\sum_{l=n}^{k} \sigma_l L_l, & \text{if } n \leq k \\ 0, & \text{otherwise.} \end{cases} \tag{22}$$

Similarly, we can calculate the outward flux of the domain by:

$$\phi_K(\vec{T}) = \phi_0(\vec{T})e^{-\tau_{1K}} + \sum_{l=1}^{K} q_l c_l e^{-\tau_{l+1,K}} \tag{23}$$

It's now possible to group all contributions for the segments pertinent to region j coming from isotropic sources in other regions i and from the isotropic current on the inward surface α:

$$\sum_k \delta_{jN_k} L_k\bar{\phi}_k = J_{-,\alpha}F_{\alpha j}(\vec{T}) + \sum_i \frac{Q_i}{4\pi}F_{ij}(\vec{T}) \tag{24}$$

and with similar symbols we can rewrite Eq (23) as

$$\phi_K = J_{-,\alpha} F_{\alpha\beta}(\vec{T}) + \sum_i \frac{Q_i}{4\pi} F_{i\beta}(\vec{T}) \tag{25}$$

where the F's in the above equations are computed as:

$$F_{ij}(\vec{T}) = \delta_{ij} \sum_{k=1}^{K} \delta_{jN_k} d_k(\vec{T}) + \sum_{k=1}^{K} \sum_{l=1}^{k-1} \delta_{jN_k} \delta_{iN_l} c_l(\vec{T}) e^{-\tau_{l,k-1}} c_k(\vec{T}) \tag{26}$$

$$F_{\alpha j}(\vec{T}) = \delta_{\alpha N_0} \sum_{k=1}^{K} \delta_{jN_k} e^{-\tau_{1,k-1}} \tag{27}$$

$$F_{i\beta}(\vec{T}) = \sum_{k=1}^{K} \delta_{\beta N_{K+1}} \delta_{iN_k} c_k e^{-\tau_{k+1,K}} \tag{28}$$

$$F_{\alpha\beta}(\vec{T}) = \delta_{\alpha N_0} \delta_{\beta N_{K+1}} e^{-\tau_{1K}} \tag{29}$$

In the CP method, the definitions of different probabilities are:

$$p_{ij} = \frac{1}{4\pi V_j} \int_{V_i} d^3r \int_{V_j} d^3r' \, \frac{\exp(-\tau(\vec{r}', \vec{r}))}{|\vec{r}' - \vec{r}|^2} \tag{30}$$

$$p_{\alpha j} = \frac{1}{\pi S_\alpha} \int_{V_j} d^3r \int_{S_\alpha} d^2r'_S \, (\hat{\Omega} \cdot \hat{N}_-) \frac{\exp(-\tau(\vec{r}'_S, \vec{r}))}{|\vec{r}'_S - \vec{r}|^2} \tag{31}$$

$$P_{i\beta} = \frac{1}{4\pi V_i} \int_{S_\beta} d^2r_S \int_{V_j} d^3r' \, (\hat{\Omega} \cdot \hat{N}_+) \frac{\exp(-\tau(\vec{r}', \vec{r}_S))}{|\vec{r}' - \vec{r}_S|^2} \tag{32}$$

$$P_{\alpha\beta} = \frac{1}{\pi S_\beta} \int_{S_\beta} d^2r_S \int_{S_\alpha} d^2r'_S \, (\hat{\Omega} \cdot \hat{N}_+)(\hat{\Omega} \cdot \hat{N}_-) \frac{\exp(-\tau(\vec{r}'_S, \vec{r}_S))}{|\vec{r}'_S - \vec{r}_S|^2} \tag{33}$$

If we denote the optical path contributions $\tau(\vec{r}', \vec{r})$ on the local coordinates $s' \leq s$ of each tracking line, after the following change of variables:

$$d^3r \, d^3r' = d^4T \, ds \, ds' \, |\vec{r}' - \vec{r}|^2 \tag{34}$$

$$d^3r \, d^2r'_S = d^4T \, ds \, |\vec{r}'_S - \vec{r}|^2 \tag{35}$$

$$d^2r_S \, d^3r' = d^4T \, ds' \, |\vec{r}' - \vec{r}_S|^2 \tag{36}$$

$$d^2r_S \, d^2r'_S = d^4T \, |\vec{r}'_S - \vec{r}_S|^2 \tag{37}$$

the Eqn (30)-(33) become:

$$p_{ij} = \frac{1}{4\pi V_j} \int_\Gamma d^4T \int_{s\in V_j} ds \int_{s'\in V_i} ds' e^{-\tau(s',s)} \tag{38}$$

$$p_{\alpha j} = \frac{1}{\pi S_\alpha} \int_\Gamma d^4T \chi_\alpha(\vec{T},0) \int_{s\in V_j} ds\, e^{-\tau(0,s)} \tag{39}$$

$$P_{i\beta} = \frac{1}{4\pi V_i} \int_\Gamma d^4T \chi_\beta(\vec{T},s_K) \int_{s'\in V_i} ds'\, e^{-\tau(s',s_K)} \tag{40}$$

$$P_{\alpha\beta} = \frac{1}{\pi S_\beta} \int_\Gamma d^4T\, \chi_\alpha(\vec{T},0)\chi_\beta(\vec{T},s_K) e^{-\tau(0,s_K)} \tag{41}$$

It is easily shown that the analytical expressions of these probabilities are:

$$p_{ij} = \frac{1}{4\pi V_j} \int_\Gamma d^4T\, F_{ij}(\vec{T}) \tag{42}$$

$$p_{\alpha j} = \frac{1}{\pi S_\alpha} \int_\Gamma d^4T\, F_{\alpha j}(\vec{T}) \tag{43}$$

$$P_{i\beta} = \frac{1}{4\pi V_i} \int_\Gamma d^4T\, F_{i\beta}(\vec{T}) \tag{44}$$

$$P_{\alpha\beta} = \frac{1}{\pi S_\beta} \int_\Gamma d^4T\, F_{\alpha\beta}(\vec{T}) \tag{45}$$

The above leads to the following equations:

$$V_j\Phi_j = \sum_\alpha J_{-,\alpha} p_{\alpha j} + \sum_i Q_i V_i p_{ij} \tag{46}$$

$$J_{+,\beta} = \sum_\alpha J_{-,\alpha} P_{\alpha\beta} + \sum_i Q_i V_i P_{i\beta} \tag{47}$$

which are equivalent to Eqn (5)-(6).

4 Self-Collision Rebalancing

We now consider the multigroup problem. From Eqn (5), (6), (11) and (12), we obtain:

$$\Phi_j^g = \Phi_{j,\text{in}}^g + p_{jj}^g Q_j \tag{48}$$

$$J_{+,\alpha}^g = J_{\alpha,0}^g + P_{j_\alpha\alpha}^g V_{j_\alpha} Q_{j_\alpha}^g \tag{49}$$

where

$$\Phi^g_{j,\text{in}} = \frac{1}{\Sigma^g_j V_j} \int_\Gamma d^4T \sum_{k=1}^{K} \delta_{jN_k} \phi^g_{k-1}(\vec{T})(1 - e^{-\tau^g_{kk}}) \tag{50}$$

$$J^g_{\alpha,0} = \int_\Gamma d^4T \, \delta_{\alpha N_{K+1}} \phi^g_{K-1}(\vec{T}) e^{-\tau^g_{K,K}} \tag{51}$$

$$p^g_{jj} = \frac{1}{4\pi\Sigma^g_j V_j} \int_\Gamma d^4T \sum_{k=1}^{K} \delta_{jN_k} \frac{-1 + \tau^g_{kk} + e^{-\tau^g_{kk}}}{\sigma^g_k} \tag{52}$$

$$P^g_{j_\alpha \alpha} = \frac{1}{4\pi V_{j_\alpha}} \int_\Gamma d^4T \, \delta_{j_\alpha N_K} \delta_{\alpha N_{K+1}} \frac{1 - e^{-\tau^g_{KK}}}{\sigma^g_k} \tag{53}$$

In the above equations, the index j_α denote the last region encountered by the tracking line $\vec{T}$ before leaving from the surface α. Assuming the region j is convex, Eq (26) for F_{jj} becomes:

$$\begin{aligned} F_{jj}(\vec{T}) &= \sum_{k=1}^{K} \delta_{jN_k} d_k(\vec{T}) \\ &= \sum_{k=1}^{K} \delta_{jN_k} \frac{-1 + \tau^g_{kk} + e^{-\tau^g_{kk}}}{\sigma_k} \end{aligned} \tag{54}$$

and Eq (28) for $F_{j_\alpha \alpha}$ becomes:

$$\begin{aligned} F_{j_\alpha \alpha}(\vec{T}) &= \delta_{j_\alpha N_K} \delta_{\alpha N_{K+1}} c_K(\vec{T}) \\ &= \delta_{j_\alpha N_K} \delta_{\alpha N_{K+1}} \frac{1 - e^{-\tau^g_{KK}}}{\sigma^g_k} \end{aligned} \tag{55}$$

So, Eq (52) defines the reduced collision probabilities from the region j to itself (called the reduced *self-collision probability*) and Eq (53) defines the leakage probabilities from the region j_α to the surface α.

In the multigroup MCI solver, we assume that the source is composed of a fission source S^g_{fj} and a scattering term. For a specific outer iteration, the source term after the n-th inner iteration is given by:

$$Q^{g,n}_j = \sum_{g'=1}^{G} \Sigma^{g \leftarrow g'}_{s,j} \Phi^{g',n}_j + S^g_{fj} \tag{56}$$

We use the superscript index $n+\frac{1}{2}$ to denote the quantities obtained from the characteristics sweep of the source $Q_j^{g,n}$ and by $n+1$ for the quantities obtained after applying the SCR technique. By substituting Eq (56) in Eq (48), we obtain the rebalancing system for the region V_j:

$$\sum_{g'=1}^{G}\left(\delta_{gg'}-p_{jj}^{g}\Sigma_{s,j}^{g\leftarrow g'}\right)\Phi_j^{g',n+1}=\Phi_{j,\mathrm{in}}^{g,n+\frac{1}{2}}+p_{jj}^{g}S_{fj}^{g} \tag{57}$$

The above SCR system is solved by an iterative method for all the regions one after another. Finally, the outward currents are updated by Eq (49) using the updated source $Q_{j_\alpha}^{g,n+1}$.

5 Track Merging Technique

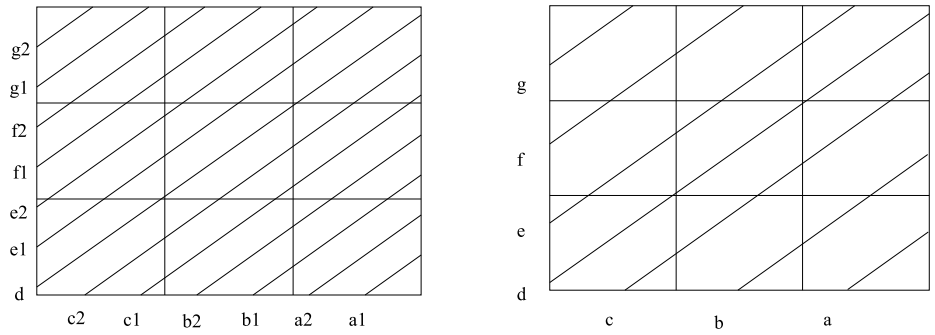
In the process of generating all the tracking lines for a large domain, some neighboring lines often cross the same regions in exactly the same order as shown in (a) of Fig 1 and the corresponding segments can have very similar lengths. The finer the tracking mesh, the more this situation will occur. We can therefore suspect that these kinds of neighboring lines having identical nuclear properties may be merged together in order to save computing time. The resulting line takes the averaged segments lengths and additional weights of the original lines as its properties. It is clear that, if the original lines are normalized, the merged lines are also normalized and verify the equality of Eq (7). In Fig 1, (b) shows the merged lines set of (a). We will now prove that the approximation of TMT is of second order with respect to the tracking length variation.

Assuming the lengths of two neighboring segments are $L'=L+\varepsilon$ and $L''=L-\varepsilon$ respectively and the merging segment length is therefore L. The inward fluxes on these three segments are supposed to be identical. From (11), the outward fluxes on them are consequently:

$$\phi'_{\mathrm{out}}=\phi_{\mathrm{in}}e^{-\sigma(L+\varepsilon)}+\frac{q}{\sigma}\left(1-e^{-\sigma(L+\varepsilon)}\right) \tag{58}$$

$$\phi''_{\mathrm{out}}=\phi_{\mathrm{in}}e^{-\sigma(L-\varepsilon)}+\frac{q}{\sigma}\left(1-e^{-\sigma(L-\varepsilon)}\right) \tag{59}$$

$$\phi_{\mathrm{out}}=\phi_{\mathrm{in}}e^{-\sigma L}+\frac{q}{\sigma}\left(1-e^{-\sigma L}\right) \tag{60}$$



(a) Before merging (b) After merging

Figure 1: Track merging technique.

and the integrated angular fluxes are:

$$\sigma L'\bar{\phi}' = \phi_{\text{in}}\left(1 - e^{-\sigma(L+\varepsilon)}\right) + \frac{q}{\sigma}\left(\sigma(L+\varepsilon) - 1 + e^{-\sigma(L+\varepsilon)}\right) \tag{61}$$

$$\sigma L''\bar{\phi}'' = \phi_{\text{in}}\left(1 - e^{-\sigma(L-\varepsilon)}\right) + \frac{q}{\sigma}\left(\sigma(L-\varepsilon) - 1 + e^{-\sigma(L-\varepsilon)}\right) \tag{62}$$

$$\sigma L\bar{\phi} = \phi_{\text{in}}\left(1 - e^{-\sigma L}\right) + \frac{q}{\sigma}\left(\sigma L - 1 + e^{-\sigma L}\right) \tag{63}$$

Assuming that:

$$\min(\phi_{\text{in}}, \frac{q}{\sigma}) \geq M_1 > 0, \qquad \max(\phi_{\text{in}}, \frac{q}{\sigma}) \leq M_2 \tag{64}$$

and then, we can get:

$$\left|\frac{\phi'_{\text{out}} + \phi''_{\text{out}} - 2\phi_{\text{out}}}{2\phi_{\text{out}}}\right| = \frac{\left|(\phi_{\text{in}} - \frac{q}{\sigma})\left(e^{-\sigma(L+\varepsilon)} + e^{-\sigma(L-\varepsilon)} - 2e^{-\sigma L}\right)\right|}{2\left|\phi_{\text{in}}e^{-\sigma L} + \frac{q}{\sigma}\left(1 - e^{-\sigma L}\right)\right|} \tag{65}$$

$$\leq \frac{M_2}{2M_1}\left(e^{-\sigma\varepsilon} + e^{\sigma\varepsilon} - 2\right) \tag{66}$$

$$= O(\varepsilon^2) \tag{67}$$

and similarly

$$\left|\frac{L'\bar{\phi}' + L''\bar{\phi}'' - 2L\bar{\phi}}{2L\bar{\phi}}\right| = \frac{\left|(\phi_{\text{in}} - \frac{q}{\sigma})\left(e^{-\sigma(L+\varepsilon)} + e^{-\sigma(L-\varepsilon)} - 2e^{-\sigma L}\right)\right|}{2\left|\phi_{\text{in}}(1-e^{-\sigma L}) + \frac{q}{\sigma}(\sigma L - 1 + e^{-\sigma})\right|} \tag{68}$$

$$\leq \frac{M_2}{2M_1\sigma L}\left(e^{-\sigma\varepsilon} + e^{\sigma\varepsilon} - 2\right) \tag{69}$$

$$= O(\varepsilon^2) \tag{70}$$

The TMT method will thus combine two or more neighboring tracking lines having this kind of similarity, and the density of the resulting track will be modified proportionally according to the original one. We will now show the effectiveness of this technique that preserves the solution accuracy.

6 Numerical Results

Our tests are based on the 3D Gentilly-2 adjuster supercell calculation. As shown in Fig 2, there are two horizontal fuel bundles and one vertical adjuster with a symmetry factor 4. The Transport equation is solved by the B1 critical buckling search. The resulting fluxes are then Homogenized and the nuclear properties are condensed to 2 energy groups keeping the small up-scattering effect from the thermal to the fast group. An EQ_4 angular quadrature is used for all the calculations. The transport equation is resolved many times with different densities and different mesh sizes. As shown in Fig 2, we refer $N = 1$ as normal mesh and $N = 2$ as fine mesh. The normal mesh has 46 regions and 35 surfaces. The fine mesh has 308 regions and 127 surfaces. For this paper, the tracking line density for the fine mesh is fixed at 25 lines/cm^2. It is the minimum for which every region is covered. The classic CP method of DRAGON is used to compare results.

First of all, we will show the performance of SCR used with the one parameter acceleration method. In the Fig 3, the inner iteration accuracy is drawn for the MCI in both the cases with and without the SCR technique. The one parameter acceleration is always used. The inner iterations are ordered from the first outer iteration to the last. Although the one parameter acceleration is very good for the CP method, it is clearly shown that it is

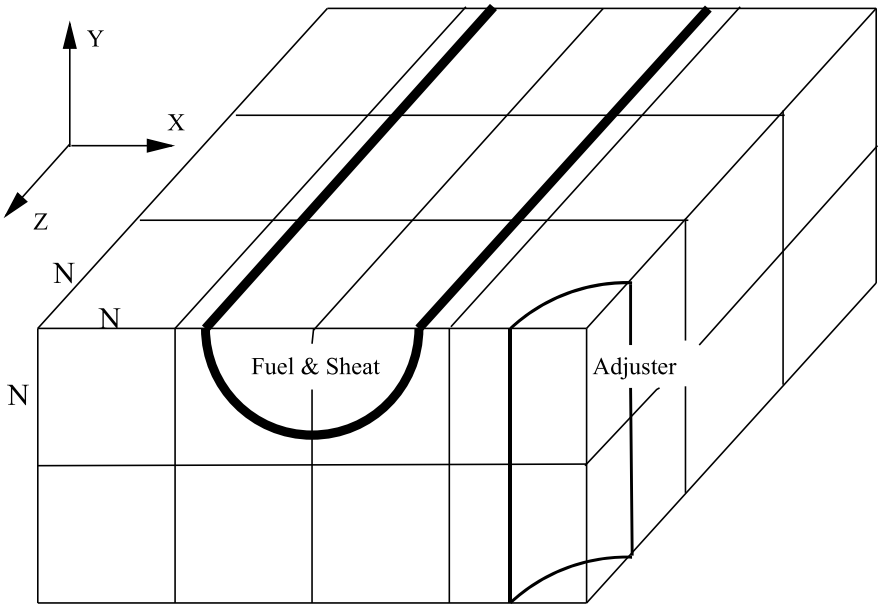


Figure 2: Adjuster supercell of Gentilly-2.

not good enough for the 3D characteristics solver MCI. The application of the SCR technique has substantially accelerated the characteristics solver resolution scheme.

We have examined the depletion calculations with density of 4 lines/cm^2 for eight time steps. The normal mesh is used for our depletion calculations. The tracking lines are merged and SCR technique is also applied for the MCI solver. The tabulated exponentials are used for the resolution of the problem. We sketch the B^2 variations of both DRAGON and MCI in Fig 4. We can see that the results are almost identical for these two methods in the depletion calculations.

We have examined the $\Delta\Sigma$ calculations for the adjuster supercell. The tracking lines densities used for the normal mesh are 1.5, 2.5 and 4.0 lines/cm^2 and the density used for the fine mesh is 25 lines/cm^2 because each region must be crossed by some lines. The $\Delta\Sigma$ results of the solver MCI, for which

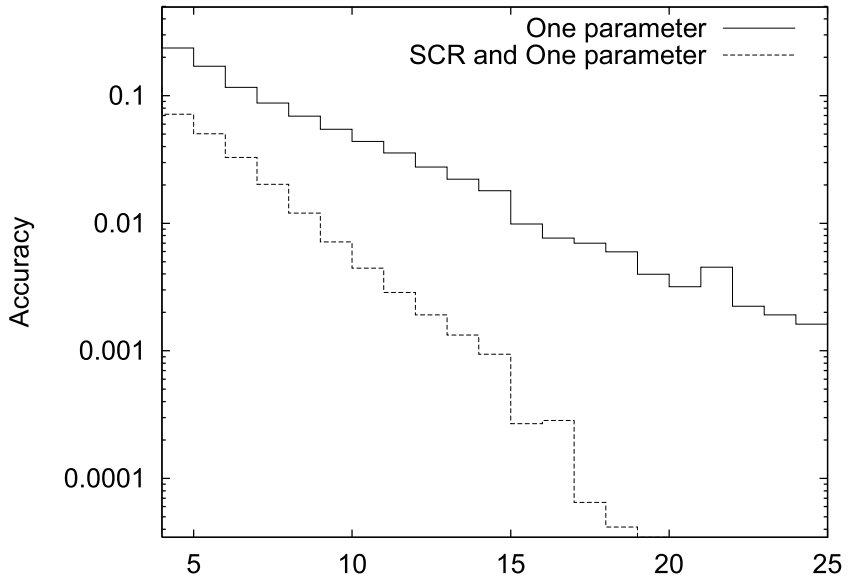


Figure 3: Convergence of the MCI solver with and without SCR technique.

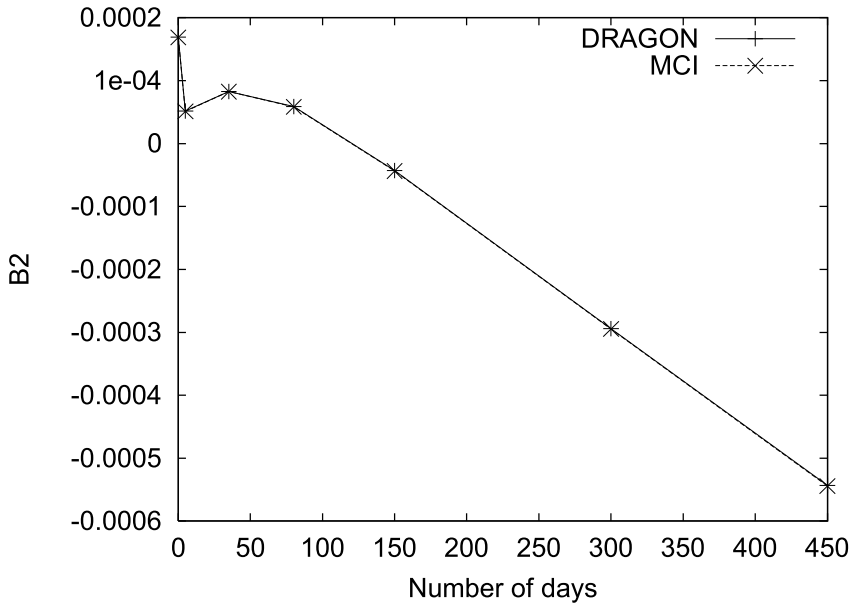


Figure 4: Variation of B^2 with residence time for the BCAINT adjuster supercell.

Incremental cross-section	Densities (lines/cm^2)			
	1.5	2.5	4.0	25.0
$\Delta\Sigma_t^1$	7.1123E-04	7.1126E-4	7.1281E-4	7.0435E-4
$\Delta\Sigma_a^1$	1.3313E-05	1.3309E-5	1.3375E-5	1.3400E-5
$\Delta\Sigma_{s0}^{1\leftarrow 1}$	6.2978E-04	6.3012E-4	6.3069E-4	6.2189E-4
$\Delta\Sigma_{s0}^{1\leftarrow 2}$	1.6773E-07	1.6758E-7	1.6888E-7	1.6131E-7
$\Delta\Sigma_t^2$	3.3516E-04	3.3447E-4	3.3724E-4	3.3107E-4
$\Delta\Sigma_a^2$	2.6757E-04	2.6677E-4	2.6967E-4	2.6583E-4
$\Delta\Sigma_{s0}^{2\leftarrow 1}$	6.8128E-05	6.7828E-5	6.8737E-5	6.9062E-5
$\Delta\Sigma_{s0}^{2\leftarrow 2}$	6.7385E-05	6.7512E-5	6.7407E-5	6.5095E-5

Table 1: Adjuster $\Delta\Sigma$ values for BCAINT.

the SCR, TMT and tabulated exponentials are all used, are presented in Table 1. We can see that the mesh and tracking line density normally used does not lead to a converged result.

To show the accuracy of different approaches, the MCI results are compared to the corresponding DRAGON results where the classic CP method is used, no lines are merged and exponentials are computed exactly. Several relative differences, defined by:

$$\frac{\Delta\Sigma_{\text{MCI}} - \Delta\Sigma_{\text{DRAGON}}}{\Delta\Sigma_{\text{DRAGON}}} \times 100. \tag{71}$$

are shown in Table 2. The options used by MCI solver are the following:

- Column A: merged lines, tabulated exponentials;
- Column B: merged lines, exact exponentials;
- Column C: original lines, tabulated exponentials;
- Column D: original lines, exact exponentials;

Using all the acceleration techniques, the MCI solver is now very competitive in speed with the classic methods of DRAGON. For problems having many regions, the MCI is faster. The CPU times shown in Table 3

Density	A	B	C	D
1.5	0.0338[a]	0.0338	0.0042	0.0042
	0.1279[b]	0.1354	0.0677	0.0677
	0.0222[c]	0.0238	0.0000	-0.0016
	-0.0209[d]	0.0358	0.0358	0.0089
	0.0973[e]	0.0973	0.0374	0.0262
	-0.5608[f]	-0.3010	0.0162	-0.1313
2.5	0.0619	0.0576	0.0239	0.0239
	0.1354	0.1580	0.0752	0.0752
	0.0524	0.0444	0.0191	0.0191
	0.0628	0.0628	0.0539	0.0539
	0.1238	0.1238	0.0375	0.0375
	-0.2261	-0.2113	0.0857	0.0931
4.0	0.0463	0.0463	0.0042	0.0084
	0.1573	0.1647	0.0524	0.0524
	0.0365	0.0349	0.0048	0.0048
	0.0089	0.0178	0.0000	0.0089
	0.1299	0.1299	0.0149	0.0149
	-0.4534	-0.4386	-0.0162	0.0281
25.0	0.0043	0.0043	0.0085	0.0000
	0.0971	0.0971	0.0822	0.0747
	-0.0064	-0.0113	-0.0016	-0.0129
	-0.0815	-0.1177	0.0181	-0.0634
	0.1130	0.1130	0.0942	0.0866
	-0.8498	-1.0203	-0.3092	-0.6655

Table 2: Relative difference between MCI and DRAGON (in %).

[a] Relative difference for $\Delta\Sigma^1$
[b] Relative difference for $\Delta\Sigma^1_a$
[c] Relative difference for $\Delta\Sigma^{1\leftarrow 1}_{s0}$
[d] Relative difference for $\Delta\Sigma^2$
[e] Relative difference for $\Delta\Sigma^2_a$
[f] Relative difference for $\Delta\Sigma^{2\leftarrow 2}_{s0}$

Density	Nb. of lines		CPU time (sec.)	
(lines/cm^2)	(before)	(after)	DRAGON	MCI
1.5	41441	21014	299	757
2.5	68391	29447	449	1042
4.0	109751	39980	667	1470
25.0	693939	312826	41032	33142

Table 3: CPU time and number of lines used by MCI and DRAGON.

are obtained on a Pentium-166 with 64 MegaBytes physical memory and 128 MegaBytes virtual memory. The operating system is Linux. It is interesting to note that, for the fine mesh case, DRAGON has used almost all of the physical and virtual memory but the MCI solver has used only a small amount of the physical memory.

7 Conclusion and perspective

The development of the TMT method was shown to be very valuable in decreasing the CPU times required to obtain accurate characteristics solutions. In the near future, the TMT method could serve as the basis for an automatic tracking procedure that will scan large 3D domains to produce a minimal tracking file containing much of the geometric data. One of the main benefit of this TMT process is that it would not depend on nuclear properties: the tracking file for various reactivity mechanisms (even containing small detailled regions) could thus be pre-processed by the user to an imposed degree of accuracy.

Acknowledgments

This work was partly funded by the Natural Science and Engineering Research Council of Canada and by the Candu Owners Group in its support for DRAGON.

References

[1] G. Marleau, A. Hébert and R. Roy. A User's Guide for DRAGON. Technical Report IGE-174 Rev. 3, École Polytechnique de Montréal, December 1997.

[2] G.J. Wu and R. Roy. A New Characteristics Algorithm for 3D Transport Calculations. In *19th Annual Conference of the Canadian Nuclear Society*, Toronto, Ontario, Canada, October 1998.

[3] G.J. Wu and R. Roy. Self-Collision Rebalancing Technique for the MCI Characteristics Solver. In *20th Annual Conference of the Canadian Nuclear Society*, Montreal, Quebec, Canada, 1999.

[4] H. Khalil. Effectiveness of a Consistently Formulated Diffusion Synthetic Acceleration Differencing Approach. *Ann. Nucl. Energy*, 98:226–243, 1988.

[5] I.R. Suslov. Solution of Transport Equation in 2- and 3-Dimensional Irregular Geometry by the Method of Characteristics. In *Mathematical Methods and Supercomputing in Nuclear Applications*, Karlsruhe, Germany, April 1994.

[6] M.J. Halsall. WIMS8 - SPEED WITH ACCURACY. In *Int. Conf. Physics of Nuclear Science and Technology*, pages 103–108, Long Island, October 1998.

[7] M.R. Zika and M.L. Adams. Transport Synthetic Acceleration for Long-Characteristics Assembly-Level Problems. *Nucl. Sci. Eng.*, 134:135–158, 2000.

[8] R. Sanchez and A. Chetaine. Synthetic Acceleration for a 2D Characteristic Method in Non Regular Meshes. In *Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*, Madrid, Spain, September 1999.

[9] R. Roy. The Cyclic Characteristics Method. In *Int. Conf. Physics of Nuclear Science and Technology, Long Island, October 5 – 8, 1998*, Long Island, 1998.

[10] Ser Gi Hong and Nam Zin Cho. Angular Dependent Rebalance (ADR) Iteration for Discrete-Ordinates Transport Problems in Equilateral Triangular Meshes. In *Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*, Madrid, Spain, September 1999.