Symmetrization of the transport operator and Lanzcos' iterations

Richard Sanchez and Simone Santandrea

Commissariat à l'Energie Atomique
Direction des Etudes Nucléaires
Service d'Etudes de Réacteurs et de Modélisations Avancées
CEA de Saclay (France)

richard.sanchez@cea.fr, simone.santandrea@cea.fr

Keywords: characteristics, acceleration, symmetrization, Lanczos

Abstract

Acceleration techniques are essential for the efficient iterative solution of the transport equation for reactor applications. In earlier work we have introduced an asymptotic synthetic acceleration method. In a radically different approach we show how to construct a scalar product such that the transport operator becomes symmetric, and apply Lanczos' iterations. An advantage of Lanczos' method is that it treats on the same footing all angular flux moments and boundary angular fluxes. Numerical tests for realistic assembly geometries are used to compare the efficiency of Lanczos' and the synthetic acceleration methods.

1 Introduction

Today reactor analysis methods are based on whole-core, coarse-group diffusion or simplified transport calculations where cells and/or assemblies are represented by properly collapsed and homogenized cross sections (Lefebvre, 1993). To correctly account for fine-spectrum effects, a first step in reactor core calculations consists of a series of fine-group, cell and/or assembly transport calculations in which the main geometrical heterogeneities are fully represented. These calculations are then used to provide a meaningful set of collapsed and homogenized cross sections for use in the whole-core calculation. Because of their ability to represent local heterogeneities, collision probability (CP) methods (Sanchez, 1982; Lewis, 1984) have been until recently the preferred tool for detailed cell and assembly transport calculations. Collision probability methods achieve this via the use of a finite set of trajectories that allows them to effectively explore the geometrical heterogeneities within the domain. However, multidimensional CP methods lack the ability to account for scattering anisotropy and require the introduction of dubious transport corrections. Furthermore, CP methods yield full collision matrices that set a practical limit to the size (i.e., total number of regions) of the problems that can be treated within reasonable computer resources. These drawbacks can be removed by the use of the trajectories in an iterative solution of the transport equation such as is done with the method of characteristics (MOC). The method of characteristics has proved to be an advantageous tool for the solution of the transport equation in unstructured meshes and, therefore, for realistic applications to reactor analysis. The first implementation of the MOC in an industrial code goes back to the seventies with the work of Halsall (1980), but it is only in recent years that the MOC has been introduced in most of the industrial codes (Goldberg, 1995; Hong, 1998; Sanchez, 2000; Smith, 2000).

Contrarily to propagation problems such as in shielding calculations, reactor transport problems are often characterized by collision-dominated regimes. and, therefore, require a large number of iterations for convergence. As for any iterative solution of the transport equation, the efficient use of the MOC for the solution of transport problems for reactor analysis requires acceleration techniques. We have recently introduced (Sanchez, 2000) an asymptotic synthetic acceleration (ASA) for use with the MOC in unstructured meshes.

The ASA is based on exact cell balance equations for the cell scalar fluxes coupled to cell approximated transmission equations for the intercell currents. The cell transmission equations are obtained by assuming that the angular fluxes entering and leaving through cell interfaces are constant in space and isotropic in angle. To ensure numerical consistency, escape and transmission coefficients are numerically evaluated with the same trajectories that are used in the MOC. The ASA provides a linear system of equations for the cell scalar fluxes and intercell currents that, after elimination of the scalar fluxes, is solved with the diagonally preconditioned biconjugated gradient stabilized method (Barret, 1994).

In this work we present a new acceleration method that consists of applying a classical projective iterative method (Saad, 1996). This approach was inspired from the work of Zika and Adams (2000). These authors have also consider the iterative solution for the MOC, and have used the fact that the associated numerical operator is symmetric with respect to the usual volume scalar product to solve the equations by conjugated gradient iterations. This work was extended to the treatment of reflecting boundary conditions in a second paper (Zika, 2000 bis). A sopposed to linear iterative methods, projective iterative methods are non linear and exhibit a quadratic convergence. Projective iterative methods are very attractive for use in the acceleration of transport iterations because they are easy to implement and they accelerate simultaneously all angular flux components. The iterative solution of the algebraic equation $A\overrightarrow{x} = \overrightarrow{s}$ requires successive applications of iterator A. However, for most projective methods a necessary condition to ensure convergence is that A be symmetric. Pioneer work in symmetrization of the transport operator was done by Vladimirov (1963). Vladimirov's ideas have been recently applied to show that the spectrum of the one-group transport operator is real under severe constraints for the anisotropic cross sections (Shani, 1995). However, neither one of these works deals with full anisotropy of scattering. In the present work we have used a constructive technique to derive a scalar product for which the iterator is symmetric, allowing thus the application of Lanczos' iterations.

A brief summary of the MOC as applied to unstructured meshes is given in Section 2. The point of view is that in (Sanchez, 2000). In that section we review the approximations used for volume and interface angular fluxes; we also discuss boundary conditions and show that usual boundary conditions obey the reciprocity principle, a fact that will be useful later. Lanczos' acceleration is the object of Section 3. Here we show how to symmetrize the iterator for the case of heterogeneous boundary conditions and anisotropic scattering and, then, for the general case with homogeneous boundary conditions. The weights for the scalar products are obtained in a direct, constructive manner. However, to ensure the positivity of the weights it is necessary to apply a conditioner to the iterator. The section ends with a discussion of the boundary conditions. The demonstrations in this section are done for the continuous form of the operator. The extension of the proof to the discretized MOC operator is given in the appendix. Tests and numerical comparisons are given in Section 4 for source and eigenvalue problems in realistic RBMK and PWR assembly geometries. A closing discussion and conclusions are presented in the last section.

2 The method of characteristics

In this section we briefly review the equations of the method of characteristics for unstructured meshes. The method is applied to obtain a numerical iterative solution of the one group transport equation in a geometrical domain D of boundary ∂D:

$$\left.\begin{array}{ll} (\Omega.\nabla + \Sigma)\psi^{(n)} = q^{(n-1)}, & x \in X, \\ \psi^{(n)} = \beta\psi^{(n-1)} + \psi_0, & x \in \partial_- X. \end{array}\right\} \tag{1}$$

In this equation n denotes the iteration index, $x = (\mathbf{r}, \Omega)$ indicates a generic point in phase space $X = \{x; \mathbf{r} \in D, \Omega \in (4\pi)\}$, $\Sigma(\mathbf{r})$ is the total cross section and

$$q = H\psi + S \tag{2}$$

is the emission density, where $(H\psi)(x) = \int_{(4\pi)} \Sigma_s(\mathbf{r}, \Omega' \cdot \Omega)\psi(\mathbf{r}, \Omega')d\Omega'$ is the usual scattering operator and S stands for the external source.

The angular flux ψ entering through the boundary $\partial_- X = \{x; \mathbf{r} \in \partial D, \Omega \in (2\pi)_-\}$ can be either a known angular flux ψ_0 or the result of particles leaving the domain and reentering it again. The latter can be

expressed in terms of an albedo operator:

$$(\beta\psi)(x) = \int_{\partial D} dS' \int_{(2\pi_+)} \beta(x' \to x)\psi(x') \, |\Omega' \cdot \mathbf{n}'| \, d\Omega', \quad x \in \partial_- X \ , \tag{3}$$

where $\mathbf{n}$ is the unit vector normal to the boundary at $\mathbf{r}$. The kernel of the albedo operator, $\beta(x' \to x)$, equals the angular flux entering the domain at x due to one particle leaving the domain at x'.

2.1 Collision term

In the method of characteristics the discrete ordinates approximation is used for the angular approximation. The discrete angular directions are the set of angular directions in the angular quadrature formula that is also used to evaluate the collision term:

$$S_N = \{w_n, \Omega_n, n = 1, N\} \Longrightarrow \frac{1}{4\pi} \int_{(4\pi)} f(\Omega) d\Omega \sim \sum_n w_n f(\Omega_n),$$

where w_n is the angular weight associated to direction Ω_n.

The calculation of the collision contribution is based on the classical expansion of the collision term on spherical harmonics:

$$\Sigma_s(\mathbf{r}, \Omega' \cdot \Omega) = \frac{1}{4\pi} \sum_{k \geq 0}^{K} \Sigma_{sk}(\mathbf{r}) \sum_{|l| \leq k} A_{kl}(\Omega) A_{kl}(\Omega'). \tag{4}$$

In our implementation we have used real spherical harmonics:

$$A_{kl}(\Omega) = \alpha_{k|l|} P_k^{|l|}(\mu) \times \begin{cases} \cos l\varphi, & l \geq 0, \\ \sin |l| \, \varphi & l < 0. \end{cases}$$

Here $\Omega = (\mu, \phi)$, the $\alpha_{kl} = \sqrt{(2 - \delta_{l0})(k-l)!/(k+l)!}$ are normalization constants and $P_k^l(\mu)$ are the familiar Legendre functions:

$$P_k^l(\mu) = (1 - \mu^2)^{l/2} \frac{d^l P_k(\mu)}{d\mu^l},$$

where P_k is the Legendre polynomial of order k. In the following we will use vector notation for the angular components and write $\overrightarrow{A}(\Omega) = \{A^\rho(\Omega), \rho = 1, N_K\}$, where a single index has been used to denote a spherical harmonic, $\rho \equiv (k, l)$, and N_K is the total number of terms in expansion (4). These spherical harmonics satisfy the orthogonality relations

$$\frac{1}{4\pi} \int_{(4\pi)} \overrightarrow{A}(\Omega) \overrightarrow{A}(\Omega) d\Omega = diag\{\frac{1}{2k+1}\}.$$

Finally, with the help of (4), the emission density in (2) can be written in vector notation as

$$q(\mathbf{r}, \Omega) = \overrightarrow{A}(\Omega) \cdot \overrightarrow{q}(\mathbf{r}).$$

In this expression $\overrightarrow{q}(\mathbf{r}) = \Sigma_s(\mathbf{r}) \overrightarrow{\phi}(\mathbf{r}) + \overrightarrow{S}(\mathbf{r})$, Σ_s is a diagonal matrix with entries $\Sigma_s^{\rho\rho} = \Sigma_{sk}$ and the angular flux moments are evaluated with the help of the angular quadrature formula:

$$\overrightarrow{\phi}(\mathbf{r}) = \frac{1}{4\pi} \int_{(4\pi)} \overrightarrow{A}(\Omega) \psi(\mathbf{r}, \Omega) d\Omega \sim \sum_n w_n \overrightarrow{A}(\Omega_n) \psi(\mathbf{r}, \Omega_n). \tag{5}$$

2.2 Basic equations

In our implementation of the method of characteristics the geometrical domain D is decomposed into a set of homogeneous regions $\{D_i, i = 1, N_{reg}\}$ on which we use a flat source approximation:

$$q(x) \sim \sum_i \theta_i(\mathbf{r}) q_i(\Omega) = \sum_i \theta_i(\mathbf{r}) \overrightarrow{A}(\Omega) \cdot \overrightarrow{q}_i, \ x \in X, \tag{6}$$

where $\theta_i(\mathbf{r})$ is the characteristic function of homogeneous domain D_i and $\overrightarrow{q}_i = \Sigma_{si} \overrightarrow{\phi}_i + \overrightarrow{S}_i$ is the average value of the emission density in region i. In a multigroup setting the external source accounts for fissions and scattering from the other groups.

At every iteration a fixed set of trajectories is used to compute the angular fluxes within the domain. For each angular direction in the quadrature formula a set of parallel trajectories is tracked over the surface $\partial D_\perp(\Omega)$ orthogonal to the direction Ω; each trajectory has a weight $w_\perp(t)$ equal to the cross sectional area associated to the trajectory.

In the method of characteristics the angular flux is assumed to be constant across the transverse cross sectional area of the trajectory:

$$\psi(x) \sim \sum_{t//\Omega} \theta_t(\mathbf{r}_\perp) \psi_t(z, \Omega), \ x \in X. \tag{7}$$

In this equation Ω is a direction in the angular quadrature set, the sum in t is done over all trajectories in direction Ω, $\mathbf{r}_\perp$ denotes a position on $\partial D_\perp(\Omega)$, $\theta_t(\mathbf{r}_\perp)$ is the characteristic function associated to the cross sectional area of trajectory t, z is the coordinate along the trajectory, and $\psi_t(z, \Omega)$ is the angular flux obtained by analytical integration of Eq. (1) along the trajectory:

$$\psi(\mathbf{r}_\perp, z, \Omega) = e^{-\tau(z_{in}, z)} \psi(\mathbf{r}_\perp, z_{in}, \Omega) + \int_{z_{in}}^{z} e^{-\tau(z', z)} q(\mathbf{r}_\perp, z', \Omega) dz',$$

where z_{in} is the entering point of the trajectory in the domain and $\tau(z', z)$ is the optical distance along the trajectory between points z' and z.

Let i be a region intersected by trajectory t. By integrating Eq. (1) along the path of trajectory t across region i one obtains a propagation equation giving the angular flux leaving the region in terms of the incoming angular flux and the internal sources:

$$\psi_{out,i}^{(n)}(t) = e^{-\Sigma_i R_i(t)} \psi_{in,i}^{(n)}(t) + \frac{1 - e^{-\Sigma_i R_i(t)}}{\Sigma_i} q_i^{(n-1)}(\Omega), \tag{8}$$

where $R_i(t)$ is the length of the trajectory within the region and Ω is the direction of the trajectory. This equation is used to iteratively compute the angular flux across region boundaries along each trajectory .

The updated flux values for the next iteration are obtained from

$$\overrightarrow{\phi}_i^{(n)} = \sum_n w_n \overrightarrow{A}(\Omega_n) \psi_i^{(n)}(\Omega_n),$$

where $\psi_i^{(n)}(\Omega)$ are the cell mean angular fluxes. These fluxes are calculated from the cell balance equation obtained by integration of Eq. (1) over the volume of a cell:

$$\Sigma_i \psi_i^{(n)}(\Omega) = q_i^{(n-1)}(\Omega) - \frac{1}{V_i} \sum_{t//\Omega, t \cap i} w_\perp(t) [\psi_{out,i}^{(n)}(t) - \psi_{in,i}^{(n)}(t)]. \tag{9}$$

Here V_i is the volume of the cell and the sum in t is done for all trajectories with direction Ω that intersect cell i. The sum is updated during the sweep every time a trajectory with direction Ω crosses cell i. Equations (8) and (9) are the basic equations of the method of characteristics.

2.3 Boundary conditions

Boundary conditions are specified by the nature of the kernel of the albedo operator. In our implementation we have considered two kind of boundary conditions: geometrical motions and approximated albedo conditions .

Geometrical motions, such as translation, rotation or axial symmetry, map the outgoing boundary $\partial_+ X = \{x; \mathbf{r} \in \partial D, \Omega \in (2\pi)_+\}$ onto the incoming boundary $\partial_- X$. This map associates an exiting angular flux to an entering angular flux, providing thus an explicit expression for the boundary condition:

$$\psi_-^{(n)}(x) = \psi_+^{(n-1)}(g^{-1}x),\ x \in \partial_- X, \tag{10}$$

where $g : \partial_+ X \to \partial_- X$ is the transformation generated by the geometrical motion. This transformation is an angle-preserving, one-to-one mapping with the property $(Rg)^2 = 1$, where R is the direction inversion operator, $(Rf)(\Omega) = f(-\Omega)$. This property of g expresses the physical fact that if the angular flux leaving at x_+ enters at x_-, then the angular flux leaving Rx_- enters at Rx_+. This means that the boundary condition generated by a geometrical motion complies with reciprocity (Sanchez, 1998). The kernel of the corresponding albedo operator is:

$$\beta(x' \to x)\,|\Omega \cdot \mathbf{n}| = \delta_{\partial D}(\mathbf{r} - g\mathbf{r}')\delta_2(\Omega \cdot g\Omega'), \tag{11}$$

where we have used the fact that the map generated by the geometrical motion preserves angles, $g^{-1}\Omega \cdot g^{-1}\mathbf{n} = \Omega \cdot \mathbf{n}$, $\delta_{\partial D}$ is a delta function on the surface of the domain, and $\delta_2(\Omega \cdot \Omega')$ is Placzek's delta (Case, 1953): $\int_{(4\pi)} d\Omega' \delta_2(\Omega \cdot \Omega') f(\Omega') = f(\Omega)$.

The second type of boundary conditions is used to implement either a white albedo condition or a piecewise constant approximation of a geometrical motion (Sanchez, 2000 bis). To do this we decompose the boundary into a set of surfaces $\partial D = \{\alpha\}$ and the exiting and entering angular domains into a set of angular subdomains $(2\pi\pm) = \{\rho\}_\pm$ and introduce a piecewise constant angular approximations for the entering and exiting angular fluxes:

$$\psi_\pm(x) \sim \sum_{\alpha,\rho} \theta^\rho_{\pm,\alpha}(x)\psi^\rho_{\pm,\alpha},\ x \in \partial_\pm X, \tag{12}$$

where $\theta^\rho_{\pm,\alpha}(x) = \theta_\alpha(\mathbf{r})\theta^\rho_\pm(\Omega)$ is the characteristic function of domain (α, ρ) and, in order to preserve reciprocity, we select $\theta^\rho_{-,\alpha}(x) = R\theta^\rho_{+,\alpha}(x)$. Again, a one-to-one mapping g is used to associate to each outgoing element (α, ρ) and incoming one $(g\alpha, g\rho)$. The boundary condition relays on this mapping:

$$\beta(x' \to x) = \sum_{\alpha,\rho} \frac{\theta^{g\rho}_{g\alpha}(x)}{c^{g\rho}_{g\alpha}} \beta_\alpha \theta^\rho_\alpha(x'), \tag{13}$$

where $c^\rho_\alpha = \int_{(\alpha)} dS \int_{(\rho)} d\Omega\,|\Omega \cdot \mathbf{n}|$ is the measure of domain (α, ρ). According to this boundary condition, a fraction β_α of the total current exiting through domain (α, ρ) reenters the medium uniformly and isotropically via the image domain $(g\alpha, g\rho)$. Once more, to insure reciprocity, we assume that $(Rg)^2 = 1$. When g is the identity map (13) represents a piecewise white albedo condition; otherwise, with $\beta_\alpha = 1$ formula (13) is used to introduce an approximation of the boundary condition generated by a geometrical motion.

3 Symmetrization of the iterator

The solution of an iterative linear problem can be very efficiently speeded up with the help of a non linear acceleration method. In this section we investigate the application of Lanczos' acceleration to the iterative solution of transport problem (1). We show that preconditioning of the iterator allows to construct a scalar product for which the iterator is symmetric, fulfilling thus the prerequisites for the convergence of Lanczos' iterations. This ensures that the iterations will converge in at most N iterations, where N is the size of the problem.

We shall treat the continuous case and relegate to the appendix the extension of our results to the discretized characteristic equations. In the continuous case an iteration consists of a sweep along all trajectories crossing the domain to produce updated values for the angular flux moments and the incoming angular fluxes in terms of the corresponding values from the previous iteration. The iteration can be written in operator notation as:

$$\begin{pmatrix} \overrightarrow{\phi} \\ \psi_- \end{pmatrix}^{(n)} = B \begin{pmatrix} \overrightarrow{\phi} \\ \psi_- \end{pmatrix}^{(n-1)} = \begin{pmatrix} C & I \\ E & T \end{pmatrix} \begin{pmatrix} \overrightarrow{\phi} \\ \psi_- \end{pmatrix}^{(n-1)}, \tag{14}$$

where, for our analysis, we have only retained homogeneous terms and exclude all source terms. In this equation $\overrightarrow{\phi}(\mathbf{r})$ is the vector of angular flux moments and $\psi_-(x)$ is the angular flux entering the surface of the domain. In the upper equation, operators C and I stand, respectively, for the contributions from volumetric collisions and incoming particles. In the lower equation operators E and T represent the contributions to the entering flux from the reflection, respectively, of particles escaping after a collision and particles transmitted without collisions.

The iterator used in Lanczos' algorithm is $1 - B$. By noting $\Xi = (\overrightarrow{\phi}, \psi)$ the vector in Eq. (14) on which iterations are performed, we define a scalar product of the form:

$$\Xi_1 \circ \Xi_2 = (\overrightarrow{\phi}_1, \overrightarrow{\phi}_2) + < \psi_1, \psi_2 >, \tag{15}$$

where $(,)$ and $<,>$ have to be defined so that the iterator is selfadjoint. To simplify the analysis we consider first the case with no albedo at the boundary.

3.1 Case with fixed incoming angular flux

The upper equation for $\overrightarrow{\phi}(\mathbf{r})$ in (14) is readily obtained by computing the angular flux in (5) by integration along the back trajectory. After some easy manipulations we obtain the following form for operators C and I:

$$(C\overrightarrow{\phi})(\mathbf{r}) = \int_D d\mathbf{r}' k(\mathbf{r}' \to \mathbf{r}) \overrightarrow{A}(\Omega_R) \overrightarrow{A}(\Omega_R) \cdot (\Sigma_s \overrightarrow{\phi})(\mathbf{r}'),$$
$$(I\psi_-)(\mathbf{r}) = \int_{\partial D} dS' \, |\Omega_R \cdot \mathbf{n}'| \, k(\mathbf{r}' \to \mathbf{r}) \overrightarrow{A}(\Omega_R) \psi_-(\mathbf{r}', \Omega_R).$$

In these equations $\Sigma_s(\mathbf{r})$ is the diagonal scattering matrix, $k(\mathbf{r}' \to \mathbf{r})$ is the familiar first collision kernel $e^{-\tau(\mathbf{r}',\mathbf{r})} / (4\pi |\mathbf{r} - \mathbf{r}'|^2)$, where $\tau(\mathbf{r}', \mathbf{r})$ is the optical distance between points $\mathbf{r}'$ and $\mathbf{r}$, and Ω_R is the unit vector pointing from $\mathbf{r}'$ to $\mathbf{r}$.

For the case with no albedo at the boundary a Lanczos' iteration consists of applying operator $1 - C$ to vector $\overrightarrow{\phi}(\mathbf{r})$. But, because of the presence of the angular functions and of the scattering matrix in its kernel, operator C is non symmetric for the usual scalar product. To symmetrize this operator we introduce a weighted scalar product of the form

$$(\overrightarrow{\phi}_1, \overrightarrow{\phi}_2) = \int_D d\mathbf{r} (\overrightarrow{\phi}_1 \cdot V \overrightarrow{\phi}_2)(\mathbf{r}), \tag{16}$$

where $V(\mathbf{r})$ is a positive definite symmetric matrix that has to be determined so that C is selfadjoint for the above scalar product, $(\overrightarrow{\phi}_1, C\overrightarrow{\phi}_2) = (C\overrightarrow{\phi}_1, \overrightarrow{\phi}_2)$. From this last condition we obtain the following equation for the weight:

$$V(\mathbf{r}) \overrightarrow{A}(\Omega_R) \otimes \Sigma_s(\mathbf{r}') \overrightarrow{A}(\Omega_R) = \Sigma_s(\mathbf{r}) \overrightarrow{A}(-\Omega_R) \otimes V(\mathbf{r}') \overrightarrow{A}(-\Omega_R).$$

Then, by noticing that $\overrightarrow{A}(-\Omega) = S\overrightarrow{A}(\Omega)$, where S is a diagonal matrix with components $S^\rho = (-)^k$, we obtain the solution $V(\mathbf{r}) = \Sigma_s(\mathbf{r})S$. This weight is diagonal and therefore symmetric but, except in the case of isotropic scattering, it is not positive and cannot be utilized to define a scalar product. One may lift this inconvenience, however, by preconditioning the iterator. We do this by applying the diagonal matrix $C_\Phi = sg(V) = sg(\Sigma_s)S$ to the upper equation in (14) and by using the positive weight $V(\mathbf{r}) = |\Sigma_s(\mathbf{r})|$ to define the scalar product. It can then be readily checked that the operator $C_\Phi(1 - C)$ is selfadjoint with respect to the scalar product defined with this new weight.

3.2 Case with albedo boundary conditions

The lower equation in (14) is defined by the albedo boundary condition in (1). The operators in this equation can be made explicit by applying the albedo operator in (3) to the exiting angular flux computed by integration along the back trajectory. In particular, the reflected transmission operator $T \equiv \beta U$ is the product of the albedo operator and the propagation operator

$$(U\psi_{in})(\mathbf{r},\Omega) = e^{-\tau(\mathbf{r},\mathbf{r}_{in})}\psi_{in}(\mathbf{r}_{in},\Omega),$$

where $\mathbf{r}_{in}$ is the point on the surface ∂D at which trajectory through $(\mathbf{r},\Omega)$ enters the domain.

With homogeneous boundary conditions one has to consider the full operator B in (14) with the total scalar product in (15). Since the volumetric product component has already been defined, we investigate now the surface scalar product

$$< \psi_1, \psi_2 >= \frac{1}{4\pi}\int_{\partial D} dS \int_{(2\pi_-)} d\Omega\, |\Omega\cdot\mathbf{n}|\, (\psi_1 W \psi_2)(x), \tag{17}$$

where, similarly as was done for the volumetric scalar product, the positive definite, symmetric weight operator W has to be defined so that this scalar product makes T selfadjoint.

In order to derive operator W we need to analyze the behavior of β and U with respect to the scalar product. The structure of the albedo can be arbitrary, but we assume here that the kernel $\beta(x' \to x)$ obeys reciprocity (Sanchez, 1998) which is the case for all practical applications. This entails that operator β has the property:

$$< \psi_-, \beta\psi_+ >_-=< R\beta R\psi_-, \psi_+ >_+,\ \forall\psi_-, \psi_+,$$

where R is the operator that changes the sign of Ω, ψ_- and ψ_+ are two regular test functions respectively defined over the domains $\partial_- X$ and $\partial_+ X$, and where we have introduced the natural scalar products:

$$< \psi_1, \psi_2 >_\pm = \int_{\partial D} dS \int_{(2\pi_\pm)} d\Omega\, |\Omega\cdot\mathbf{n}|\, (\psi_1\psi_2)(x).$$

Notice that we have taken inspiration of the structure of these scalar products to define (17) which can be viewed as $[1/(4\pi)] < \psi_1, W\psi_2 >_-$.

As for operator U, one easily checks that $U^\dagger = RUR$, that is, U satisfies reciprocity:

$$< \psi_+, U\psi_- >_+ =< RUR\psi_+, \psi_- >_-,\ \forall\psi_-, \psi_+.$$

We are now in a position to determine W from the condition $< \psi_1, T\psi_2 >=< T\psi_1, \psi_2 >$. By using the properties of β and U in this relation one obtains the following equations for the weight:

$$RU\beta RW \equiv W\beta U,$$

which has the solution $W\beta \equiv R$. From this expression one can obtain $W \equiv R\beta^{-1}$, under the condition that the albedo operator admits an inverse. Later we will show that this is the case in practical applications. The weight operator W is symmetric but, as before, it is not positive so we have to use a preconditioner in the lower equation of (14). We have found that the use of preconditioner $C_\psi = R\beta^{-1}$ with the new weight $W \equiv 1$ makes operator $C_\psi(1-T)$ selfadjoint.

3.3 Lanczos' iterator

After preconditioning, Lanczos' iterator for problem (1) reads:

$$\begin{pmatrix} C_\Phi & 0 \\ 0 & C_\psi \end{pmatrix} - \begin{pmatrix} C_\Phi C & C_\Phi I \\ C_\psi E & C_\psi T \end{pmatrix}. \tag{18}$$

We have defined the preconditioners, $C_\Phi = sg(\Sigma_s)S$ and $C_\psi = R\beta^{-1}$, and the positive definite weights for the scalar products, $V(\mathbf{r}) = |\Sigma_s(\mathbf{r})|$ and $W \equiv 1$, so the diagonal terms of the iterator are selfadjoint.

However, for the entire operator to be selfadjoint, the off-diagonal terms have also to be the adjoint one of each other:

$$(\overrightarrow{\phi}, C_\Phi I\psi_-) =< C_\psi E\overrightarrow{\phi}, \psi_- >, \ \forall \overrightarrow{\phi}, \psi_-.$$

This property can be easily checked out. By observing that the β^{-1} operator in C_ψ cancels out the β operator in E, we can write:

$$(C_\psi E\overrightarrow{\phi})(\mathbf{r}, \Omega) = R\int_0^{x_{in}} dx e^{-\tau(\mathbf{r}_x, \mathbf{r})}\overrightarrow{A}(\Omega)\cdot(\Sigma_s\overrightarrow{\phi})(\mathbf{r}_x),$$

where $\mathbf{r}_x = \mathbf{r} - x\Omega$, and x_{in} is the length within the medium of the trajectory through $(\mathbf{r}, \Omega)$. With the help of this result we obtain:

$$\begin{aligned} < C_\psi E\overrightarrow{\phi}, \psi_- > = \int_{\partial D} dS \int_D d\mathbf{r}_x \left|\Omega_R\cdot\mathbf{n}\right| k(\mathbf{r}_x \to \mathbf{r}) \times \\ \times \ \overrightarrow{A}(-\Omega_R)\cdot(\Sigma_s\overrightarrow{\phi})(\mathbf{r}_x)\psi_-(\mathbf{r}, \Omega_R), \end{aligned}$$

where Ω_R is the unit vector pointing from $\mathbf{r}_x$ to $\mathbf{r}$. Use of the expression for $I\psi_-$ in $(\overrightarrow{\phi}, C_\Phi I\psi_-)$ gives the same result.

3.4 Analysis of boundary conditions

We consider here the two boundary conditions discussed in Section 2. Our aim is to show that these boundary conditions obey reciprocity and define an invertible albedo operator. The albedo of the boundary condition defined by a geometrical motion, as in (11), has been shown to obey reciprocity. Moreover, the corresponding albedo operator has an inverse with kernel:

$$\beta^{-1}(x' \to x)\left|\Omega\cdot\mathbf{n}\right| = \delta_{\partial D}(\mathbf{r}' - g\mathbf{r})\delta_2(\Omega'\cdot g\Omega).$$

Next, we consider a pointwise albedo condition of the form:

$$\beta(x' \to x) = \frac{\beta(\mathbf{r})}{\pi}\delta_{\partial D}(\mathbf{r}' - \mathbf{r}).$$

Clearly, this albedo satisfies reciprocity and has the inverse:

$$\beta^{-1}(x' \to x) = \frac{1}{\pi\beta(\mathbf{r})}\delta_{\partial D}(\mathbf{r}' - \mathbf{r}).$$

Although this albedo condition is the natural albedo for the continuous case, in practical applications one uses, however, the piecewise constant white boundary condition defined in (13). The corresponding albedo operator complies with reciprocity if the condition

$$\beta_\alpha c_\alpha^\rho = \beta_{g\alpha} c_{g\alpha}^{g\rho}$$

is satisfied, which happens to be the case in all practical applications. However, the inverse does not exists in the continuous case; this is because the albedo operator averages the boundary angular flux on the exiting domain (α, ρ). However, if one introduces an approximation for the boundary angular fluxes as in (12), then the albedo operator has the inverse:

$$\beta^{-1}(x \to x') = \sum_{\alpha,\rho}\frac{\theta_{g\alpha}^{g\rho}(x)}{\beta_{g\alpha}c_{g\alpha}^{g\rho}}\theta_\alpha^\rho(x').$$

4 Numerical tests

In this section we compare Lanczos' iterations to the earlier ASA scheme and to a new DP_1 scheme under development (Santandrea, 2001). An attractive characteristic of Lanczos' acceleration is that it requires a single calculation step per iteration, in contrast with ASA and DP_1 that need a separated acceleration step. Moreover, Lanczos' acceleration treats all components, angular flux moments and boundary angular moments, on the same footing which, further, entails that all angular flux moments are accelerated. This has to be compared to ASA and DP_1 for which acceleration of high angular moments is based on non consistent approximations. These properties of Lanczos' scheme makes it attractive and easy to implement.

In this section we illustrate with a few examples the use of the DP_1 and Lanczos' accelerations. We think that it is fair to say that the original ASA (DP_0) acceleration gives on the average the best gains, but the new methods were better in some calculations. In particular, the DP_1 method results in less accelerated iterations than the ASA. However, the mean time per iteration is greater and there is no reduction in the total computing time. With coarse tracking sets Lanczos' acceleration was always faster than ASA and DP_1.

We have not tried to be exhaustive in our numerical examples. In the first subsection we consider a one-group source problem for a RBMK assembly, and in the last subsection we analyze a three-group eigenvalue problem for a PWR rodded assembly. We calculate both problems with specularly reflected boundary conditions and with piecewise approximated boundary conditions. Furthermore, to compare acceleration performances in a broad spectrum of conditions, we have run our calculations for two types of tracking: a coarse tracking and a refined tracking. The first tracking corresponds to a routine calculation, while the second is typical of a reference calculation.

4.1 RBMK assembly

Our first example is a RBMK assembly of 25 cm of side and specular reflection on its boundary. The assembly contains an inner fuel region with 18 fuel elements of radius 0.6814 cm, and a central inert tube. The fuel region is surrounded by a tube force of external radius 4.3 cm. The assembly is schematized in Fig. 1, where the flat flux regions are indicated. Internal axial symmetries have been used to reduce the calculation domain to one fourth of the assembly with a total of 192 flat flux regions. We have run a series of one-group, source calculations with P_1 cross sections obtained by collapsing of a 99-group APOLLO2 calculation (Sanchez, 1988). The APOLLO2 multigroup calculation predicted a nearly uniform thermal source in the graphite. Accordingly, for our calculations we have adopted an uniform unit source in the graphite.

Two kind of boundary conditions have been analyzed: exact conditions with the use of cyclic trajectories and approximated conditions based on the use of a piecewise, in angle and space, approximation. In the first case only region-averaged angular flux moments have to be accelerated, while for the second the piecewise boundary angular fluxes also have to be accelerated.

To better show the complementary aspects of the different types of acceleration we have calculated the assembly first with a coarse tracking set and then with a refined one. The coarse tracking set was constructed with four horizontal uniformly spaced angles (in $\pi/2$) and only one vertical angle, whereas eight uniformly spaced angles (in $\pi/2$) and four Gauss-Legendre vertical angles (Sanchez, 2000 bis) were used in the refined tracking set. Table 1 summarizes the characteristics of the tracking sets. A constant distance between trajectories was used in the calculations with exact boundary conditions, whereas a non-uniform Gauss-Legendre quadrature for the spacing between trajectories (Sanchez, 1997) was applied in the calculations with approximated boundary conditions. This quadrature is obtained by first projecting all mesh discontinuities onto the direction orthogonal to tracking and then applying a Gauss-Legendre quadrature to each of the segments so defined. This explains the larger number of trajectories observed in Table 1 for the approximated treatment of the boundary condition. Furthermore, whereas for approximated boundary conditions trajectories end as soon as they hit the boundary, cyclic trajectories bounce off the domain boundary until they reach the initial entering point. Therefore cyclic trajectories contain much more intersections than open trajectories. Clearly the total number of intersections is a good measure of the numerical work involved in one iteration.

The results for both tracking sets are given in Table 2. The convergence criterion to stop the iterations was a maximum relative error of 10^{-5}. All accelerated iterations converged to the correct flux values within

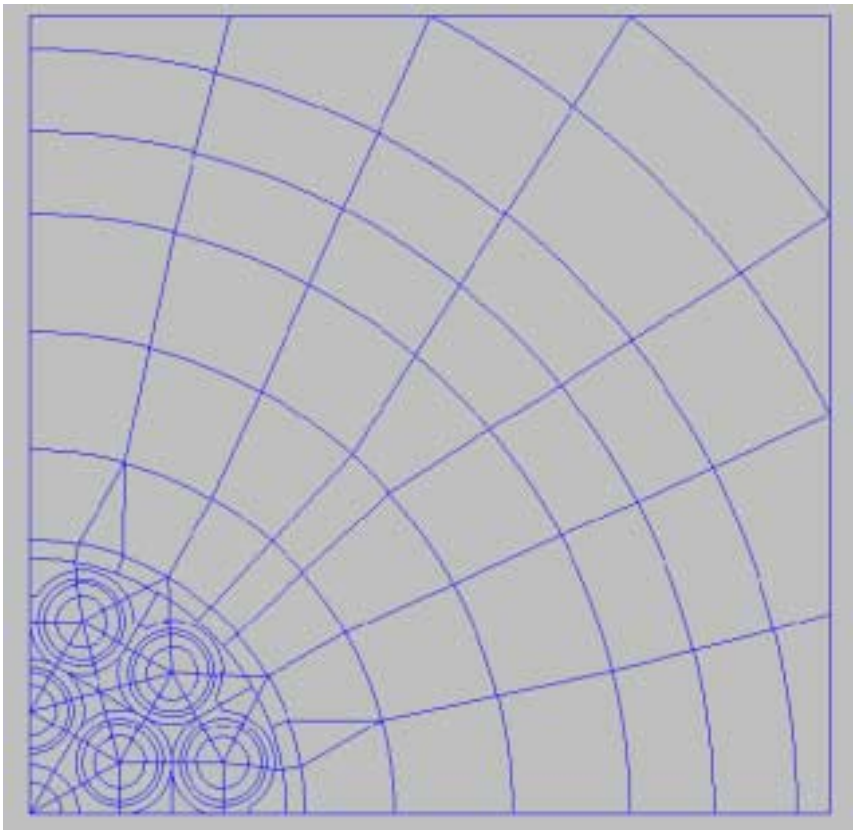

Figure 1: Region partition for the one-group, source calculation of the 1/4 of the RBMK assembly. A uniform source of intensity 1 exists in the external graphite moderator. The domain, with a total of 192 flat flux regions, is specularly reflected on its four sides.

a few iterations. However, this was not the case for free iterations. Because of the high thermalization of the assembly, free iterations converge very slowly and the relative convergence criterion is satisfied before true convergence is achieved for the same criterion. Only after we increased the relative convergence criterion to 10^{-6}, the free iterations gave the same result, to the precision 10^{-5}, than the accelerated iterations. Because of the faster convergence rate the latter did not encounter this problem.

For the coarse tracking set the time spend in a free iteration is or the order of the time spend in the acceleration stage, whereas the former time is much greater than the latter for the refined tracking. Thus, we should anticipate that for the coarse tracking case the best acceleration is the one that involves less computing time, while for the refined tracking the best performance would arise from the acceleration that minimizes the number of iterations. The results in Table 2 confirm our expectations and shows the efficiency of acceleration for this highly thermalized assembly. We discuss first the calculations with the coarse tracking set. Although not reducing considerably the number of iterations, Lanczos' acceleration is the fastest. The explanation for this is the small overwork involved in each iteration. Synthetic DP_0 and DP_1 accelerations minimize the number of iterations but are penalized for the time involved in the acceleration stage. For this problem ASA is comparable with Lanczos, whereas DP_1 acceleration is strongly penalized by its heavy arithmetics. The ASA and DP_1 accelerations perform much better for the calculations with refined tracking because here the cost of the acceleration is small as compared to that of a free iteration. Hence, the best acceleration is the one that gives the smallest number of iterations. Indeed, it can be observed that the DP_1 scheme, even though it involves a greater numerical effort than the ASA, becomes competitive, and one may foresee that this scheme would be the best if the number of trajectories is further increased. Lanczos' acceleration lags behind but offers still a good gain in computational time.

Table 1: Characteristics of the tracking sets for the RBMK assembly calculation. Coarse tracking is based on 4 horizontal uniformly spaced angles and 1 vertical angle. Refined tracking uses 8 horizontal uniformly spaced angles and 4 vertical Gauss-Legendre angles.

Tracking	Boundary condition	Number of trajectories	Number of tracks	mean traj. spacing
Coarse	Approx.	1956	44034	0.50 cm
	Exact	58	8365	0.10 cm
Refined	Approx.	4550	97236	0.10 cm
	Exact	335	168856	0.01 cm

Table 2: Results for the RBMK assembly with the coarse and the refined quadrature formulas. The calculations were performed for the two types of boundary conditions with free iterations and with different types of acceleration: a = total iteration time in seconds, b = mean time per inner iteration in seconds, c = number of inner iterations.

	Coarse tracking		Refined tracking	
	Approx.	Exact	Approx.	Exact
Free (10^{-5})	54.50^a	12.37	265.0	415.7
	0.112^b	0.032	0.561	1.039
	487^c	386	472	400
Free (10^{-6})	78.55	16.92	346.1	565.9
	0.123	0.033	0.554	1.112
	640	508	624	509
ASA	3.24	1.59	10.83	23.45
	0.191	0.093	0.637	1.172
	17	17	17	20
DP_1	10.51	34.03	14.04	22.39
	0.808	2.43	1.080	1.493
	13	14	13	15
Lanczos	3.18	0.94	17.21	36.1
	0.110	0.034	0.555	1.094
	29	28	31	33

4.2 PWR assembly

Figure 2 shows a typical PWR rodded assembly with a 1/8 assembly symmetry. The fuel is enriched at 3.25%. The figure shows the location of the absorber cells, endowed with an internal absorbing zone and two external moderating zones, and the presence of a central water hole. For our calculations we used the region partition shown in the figure with a total of 487 regions.

Multigroup selfshielded cross sections for the heavy isotopes in the fuel, the natural zirconium present in the cladding and the absorbing resonant isotopes situated in the absorber pellets were determined with the simplified multicell treatment of APOLLO2 (Sanchez, 1988). We have run multigroup eigenvalue calculations for 3 macrogroups, 2 fast and 1 thermal, with cross sections collapsed from an APOLLO2 99-group calculation. As for the previous case, we have considered exact and approximated boundary conditions with coarse and refined angular quadratures. The characteristics of the tracking sets are given in Table 3. The angular quadratures used for these trackings are the same that those utilized for the RBMK assembly. Also, as before, a projective technique together with a Gauss-Legendre formula were used to obtain trajectory spacing for the case with approximate boundary condition.

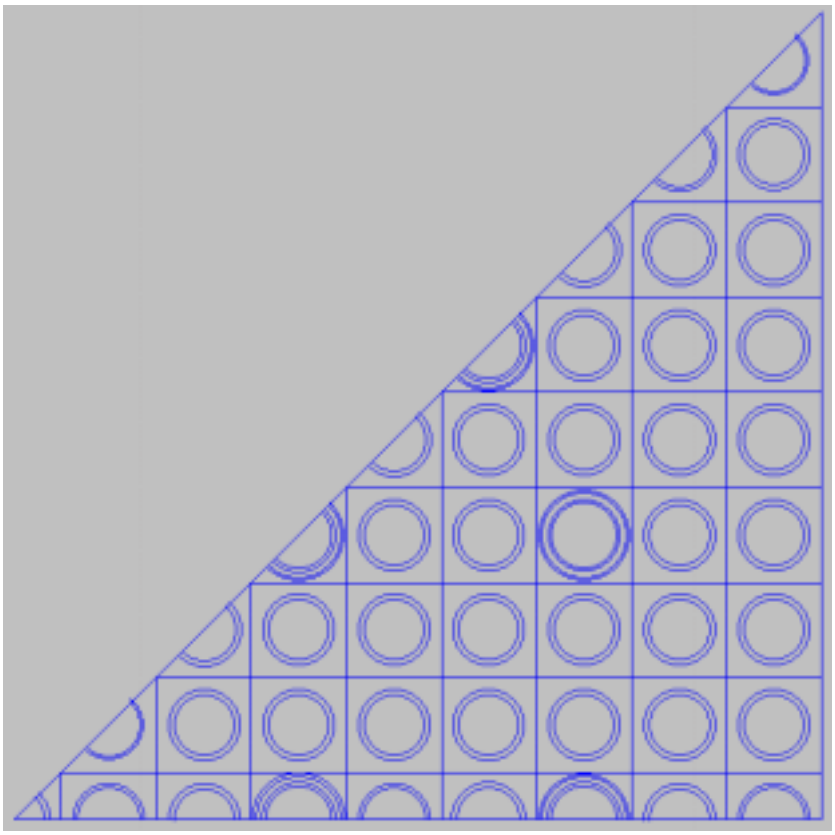

Figure 2: Region partition for the three-group, eigenvalue calculation of the 1/8 of the PWR assembly. The assembly, with a total of 192 flat flux regions, is specularly reflected on its three sides.

The results of the calculations are given in Table 4, where we have also indicated the corresponding eigenvalues. The physics of the PWR rodded assembly case are markedly different from those of the RBMK assembly considered in the previous subsection. The rodded assembly is strongly subcritical and dominated by absorption which implies that the acceleration would be much less efficient. Because of this the trends observed in the RBMK problem are amplified in the present case. As can be seen in the table, the acceleration is of practically no use when using the coarse tracking set. Again the 'best' computation time is obtained with Lanczos' iterations but the gain is minimal. As for the ASA and DP_1 acceleration schemes, they are heavily penalized by the overhead work needed for the acceleration. In particular, we notice that the DP_1 acceleration is not as efficient as in the RBMK case in reducing the number of iterations. This is aggravated by the heavy time overload required for the acceleration step, resulting in nonsensical computation times.

We turn now to a discussion of the results with detailed tracking. We note first that the eigenvalue increases of about 2000 pcm (1 pcm = 10^{-5}) which gives an idea of the 'worth' of the results obtained with the coarse tracking set. The best acceleration is now the ASA that, although not been as spectacular as for the heavily moderated RBMK case, gives a decent gain. The DP_1 and Lanczos' acceleration schemes result in very small acceleration factors.

5 Conclusions

We have considered the application of a classical projective iterative method to the acceleration of the MOC in unstructurede meshes. Basically, projective methods iteratively improve a guess vector by increasing the dimension of the initial Krilov space. We have selected Lanczos' iterative scheme because, in contrast with the conjugated gradient method, it does not require that the operator be definitive positive, a property that, depending on the type of boundary conditions, it is only true in the case of isotropic scattering. However,

Table 3: Characteristics of the tracking sets for the PWR assembly calculation. Coarse tracking is based on 4 horizontal uniformly spaced angles and 1 vertical angle. Refined tracking uses 8 horizontal uniformly spaced angles and 4 vertical Gauss-Legendre angles.

Tracking	Boundary condition	Number of trajectories	Number of tracks	mean traj. spacing
Coarse	Approx.	482	11708	0.10 cm
	Exact	5	9950	0.10 cm
Refined	Approx.	4698	115214	0.02 cm
	Exact	15	30332	0.05 cm

Table 4: Results for the PWR assembly with the coarse and the refined quadrature formulas. The calculations were performed for the two types of boundary conditions with free iterations and with different types of acceleration: a = total iteration time in seconds, b = mean time per inner iteration in seconds, c = number of inner iterations.

	Coarse tracking		Refined tracking	
	Approx.	Exact	Approx.	Exact
k_{eff}	0.82060	0.81495	0.83473	0.83118
Free	38.07^a	15.40	678.0	186.5
	0.040^b	0.027	0.663	0.301
	956^c	562	1022	620
ASA	50.95	20.30	246.1	79.30
	0.156	0.105	0.764	0.413
	327	194	322	192
DP_1	457.90	138.70	525.07	169.9
	1.431	0.730	1.672	0.894
	320	190	314	190
Lanczos	31.95	14.14	592.77	162.9
	0.039	0.029	0.705	0.322
	816	481	841	506

a necessary condition for the convergence of Lanczos' iterations is that the operator has to be symmetrical. Under this condition it is known that the iterations converge in at most N steps, where N is the dimension of the system. Because the transport operator is not symmetric, except for isotropic scattering, we have introduced in a constructive way weighted scalar products for which the operator is symmetric. This was done by direct inspection of each one of the scalar products: the volume and surface components and, then, the surface to volume and volume to surface cross terms. A problem we found is that the direct construction of the scalar product gives a weight operator that is not positive definite and, therefore, cannot be accepted to define a true scalar product. A way to circumvent this is to use part or the entire weight to define a preconditioning operator: by choosing the sign of the weight to define the preconditioner we were able to obtain a definitive positive scalar product for the preconditioned iterator. This was possible for the volume weight but not for the surface one. For the latter we decided to use the entire weight as preconditioner and utilize the natural, non-weighed scalar product, for the preconditioned iterator. Since the same operation could also be done for the volume weight, it is in order to question which choice is better: a partial distribution of the weight between preconditioner and scalar product or the inclusion of the entire weight in the preconditioner and the use of the natural scalar product? Since both choices involve the same amount of operations, clearly the best choice is the one that minimizes the number of iterations or, in other words, that minimizes the spectral radius of the iterator. Our numerical tests proved that the best results were obtained by using as much as the weight as possible in the scalar product, which explains why we choose to only use the sign of the weight to

precondition the volume equations. In contrast with the DP $_N$ scheme, that requires a transport sweep and an acceleration step, a Lanczos' iteration consists only of a modified transport sweep followed by a few scalar products to compute the new vector. Moreover, Lanczos' method treats on the same foot all the angular flux moments and boundary fluxes. These two features make Lanczos' method a very attractive and easy to implement acceleration scheme.

We have used realistic RBMK and PWR assembly, source and eigenvalue test cases to establish a comparison between Lanczos' method, the earlier developed ASA method and the DP $_1$ method that is under development. The methods were compared with exact and approximated treatments of reflective boundary conditions. In the first case we used cyclic trajectories, whereas for the second the trajectories end at the boundaries, where they are used to compute entering flux components. We used two different sets of trajectories for each one of the calculations: a coarse trajectory set and a refined one. The number of angular directions and the track spacing for the coarse set are characteristics of fast, routine calculations, while those for the refined set are typical of a reference calculation. On the average, ASA acceleration gave the more stable results. The DP$_1$ method always gave the best performances in number of iterations but not in total computation time. The reason is that the overtime needed to solve the acceleration equations increases very fast with the size of the problem. This is a consequence of the poor behavior of the BICGST iterations. In the future we will try to precondition this iterator to improve its performance. A positive aspect of DP $_1$ acceleration is that it can become competitive when using very refined tracking sets, as our results for the RBMK assembly show.

Our numerical tests show that the iterator of Lanczos' method is not as efficient as the DP $_N$ method in reducing the spectral radius, with the consequence that the number of Lanczos' iterations is always greater than those for the DP$_N$. Nevertheless, Lanczos' exhibits a behavior close to the asymptotic synthetic method in that it efficiently accelerates diffusion modes but is not too efficient in the presence of heavy absorption. Nevertheless, Lanczos' method requires the smallest storage overhead and for routine calculations gives the best results. The reason is that with coarse tracking the acceleration step for the ASA is as expensive as a free iteration and, therefore, greater than the effort required for a Lanczos' iteration. Lanczos' iterations are, thus, the best candidate for routine calculations. Another advantage of Lanczos' method is that it is as robust as free iterations and always converges to the exact result. We have observed cases for which the ASA and DP$_1$ methods diverge. This may happen when the initial guess is far from the solution and the difference between two iterates gives an acceleration source that results in strong negative corrections, breaking down the positivity of the MOC scheme.

References

[1] R. Barret et al., 'Templates for the solution of linear systems: building blocks for iterative methods, SIAM, 1994.

[2] K.M. Case, F. de Hoffmann and G. Placzek , 'Introduction to the theory of neutron diffusion,' U.S. Government Printing Office, Washington, D.C., 1953.

[3] L. Goldberg, J. Vujic, A. Leonard and R. Stachowski, 'The Characteristics Method in General Geometry,' *Trans. Am. Nucl. Soc*.., **73**, 173, 1995.

[4] M.J. Halsall, 'CACTUS, A Characteristics Solution to the Neutron Transport Equations in Complicated Geometries,' AEEW-R 1291, Atomic Energy Establishment, Winfrith, Dorchester, Dorset, Apr. 1980.

[5] S.G. Hong and N.Z. Cho, 'CRX: A Code for Rectangular and Hexagonal Lattices Based on the Method of Characteristics,' *Ann. Nucl. Energy*, Vol. 25, No. 8, pp. 547-565, 1998.

[6] J.C. Lefebvre and C. Garzenne, 'Transport Diffusion Equivalence Method Used in the EDF Neutronic Calculation Scheme,' Int. Conf. on Simulation Methods in Nuclear Engineering, Montreal, Canada, 1993

[7] E.E. Lewis and W.F. Miller, Jr., 'Computational Methods of Neutron Transport,' John Wiley & Sons, NY, 1984.

[8] Y. Saad, 'Iterative Methods for Sparse Systems,' PWS Publishing Company, Boston, 1996.

[9] D.C. Shani, E.B. Dahl and N.G. Sjostrand, 'Real Criticality Eigenvalues of the One-speed Linear Transport Operator,' Transport Theory and Stat. Physics, 24(9), 1295-1317, 1995.

[10] R. Sanchez and N.J. McCormick, *Nucl. Sci. Eng.*, **80**,481-535, 1982.

[11] R. Sanchez, J. Mondot, Z. Stankovski, A. Cossic, I. Zmijarevic, 'APOLLO2: A User Oriented, Portable, Modular Code for Multigroup Transport Assembly Calculations,' *Nucl. Sci. Eng.*, **100**, 352-362, 1988.

[12] R. Sanchez, 'Duality, Green's functions and all that. *Transport Theory and Statistical Physics* 27(5-7), 445-478, 1998.

[13] R. Sanchez and A. Chetaine, 'A Synthetic Acceleration for a Two-Dimensional Characteristic Method in Unstructured Meshes,' *Nucl. Sci. Eng.*, **136**, 122-139, 2000.

[14] R. Sanchez, L. Mao and S. Santandrea, 'Treatment of boundary conditions in trajectory-based deterministic transport methods,' submitted to *Nucl. Sci. Eng.*, 2000.

[15] R. Sanchez, 'New Developments in the Interface-Current Module TDT,' *Int. Top. Mtg on Mathematical Methods and Supercomputing for Nuclear Applications*, Saratoga, New York, Oct.1997.

[16] S. Santandrea and R. Sanchez, 'Acceleration techniques for the characteristic method in unstructured meshes,'submitted to *Ann. Nucl. Energy*, 2001.

[17] K.S. Smith, 'CASMO-4 Characteristics Methods for Two-Dimensional PWR and BWR Core Calculations,' *Trans. Am. Nucl. Soc.*, **83**, 294-296, 2000.

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

[19] M.R. Zika and M.L. Adams, 'Transport Synthetic Acceleration with Opposing Reflecting Boundary Conditions,' *Nucl. Sci. Eng.*, **134**,159-170, 2000.x

[20] V.S. Vladimirov, 'Mathematical Problems in the One-Velocity Theory of Particle Transport,' Atomic Energy of Canada Ltd., Chalk River, Report AECL-1661, 1963.

A Application of the symmetrization to the method of characteristics

The symmetrization technique that we have demonstrated for the continuous transport operator can be readily extended to numerical approximations of the transport equation, such as in the method of characteristics. To obtain a discretized solution of the transport equation, one introduces the approximation

$$\overrightarrow{\phi}(\mathbf{r}) \sim \sum_i \theta_i(\mathbf{r})\, \overrightarrow{\phi}_i, \ \mathbf{r} \in D,$$

for the angular flux moments and an approximation as in (12) for the boundary angular fluxes. The set of piecewise constant expansion functions $\{\theta_i(\mathbf{r}), \theta^{\rho}_{\pm,\alpha}(x)\}$ defines a subspace $\mathcal{E}_0$ of the space of regular functions $\mathcal{E}$ on which acts the continuous iterator in (14). However, the action of the iterator on a function $\Xi_0 \in \mathcal{E}_0$ does not necessarily result in a function in the same subspace. A numerical approximation of the continuous problem can be obtained by projecting back onto $\mathcal{E}_0$ the result of applying the iterator to a vector in $\mathcal{E}_0$. This procedure, that is also known as a Galerkin projection, produces the following discrete iterator:

$$P \begin{pmatrix} C_\Phi(1-C) & C_\Phi I \\ C_\psi E & C_\psi(1-T) \end{pmatrix} P. \tag{19}$$

In this expression $P : \mathcal{E} \to \mathcal{E}_0$ is the projector:

$$P \equiv \begin{pmatrix} P_\Phi & 0 \\ 0 & P_\psi \end{pmatrix},$$

with components:

$$(P_\Phi \overrightarrow{f})(\mathbf{r}) = \sum_i \theta_i(\mathbf{r}) \frac{1}{V_i} \int_{(i)} d\mathbf{r}' \overrightarrow{f}(\mathbf{r}'),$$
$$(P_\psi f)(x) = \sum_{\alpha,\rho} \theta^\rho_{-,\alpha}(x) \frac{1}{c^\rho_\alpha} \int_{(\alpha)} dS' \int_{(\rho-)} d\Omega' \, |\Omega' \cdot \mathbf{n}'| \, f(x').$$

On the premises that the projectors are selfadjoint with respect to their respective scalar products, $(f, P_\Phi g) = (P_\Phi f, g)$ and $< f, P_\psi g >=< P_\psi f, g >$, one can easily check that the numerical iterator (19) is symmetric with respect to scalar product (15). Indeed:

$$(\overrightarrow{f}, P_\Phi C_\Phi (1-C) P_\Phi \overrightarrow{g}) = (P_\Phi \overrightarrow{f}, C_\Phi (1-C) P_\Phi \overrightarrow{g}) =$$
$$(C_\Phi (1-C) P_\Phi \overrightarrow{f}, P_\Phi \overrightarrow{g}) = (P_\Phi C_\Phi (1-C) P_\Phi \overrightarrow{f}, \overrightarrow{g}),$$

and, similarly,

$$(\overrightarrow{f}, P_\Phi C_\Phi I P_\psi g) = (P_\Phi \overrightarrow{f}, C_\Phi I P_\psi g) =$$
$$< C_\psi E P_\Phi \overrightarrow{f}, P_\psi g > =< P_\psi C_\psi E P_\Phi \overrightarrow{f}, g >,$$

and

$$< f, P_\psi C_\psi (1-T) P_\psi g > =< P_\psi f, C_\psi (1-T) P_\psi g >=$$
$$< C_\psi (1-T) P_\psi f, P_\psi g > =< P_\psi C_\psi (1-T) P_\psi f, g > .$$

It remains to verify that P_Φ and P_ψ are selfadjoint. We find that P_ψ is selfadjoint and, under the condition $V P_\Phi = P_\Phi V$, P_Φ is also selfadjoint. The commutation relation between V and P_Φ requires Σ_s to be regionwise constant. We notice that this is the case for the method of characteristics because in this method all physical properties are taken to be constant on each region. However, the enlightened reader might suspect that (19) is not exactly the iterator of the method of characteristics, because for the latter the integrations implicit in operators C, I, E and T are approximately calculated over a finite set of trajectories. To complete our proof we have to prove then that the approximate operators satisfy the same properties of adjointness that the original ones. But, indeed, this is the case because the set of numerical trajectories are computed only for half of the total directions. Then, in the method of characteristics (as well as in the collision probability method) a trajectory in direction Ω is used twice in the calculations, first in the original direction and then, by reverting the order of the intersections, in the opposite direction. This procedure complies with reciprocity and ensures, thus, that the approximate operators are selfadjoint. This applies, in particular, to the approximate operators for E and T, when the albedo conditions are those discussed in Section 2.