DIFFUSION SYNTHETIC ACCELERATION OF S_N LINEAR NODAL SCHEMES IN WEIGHTED DIFFERENCE FORM

Igor Zmijarevic, Richard Sanchez and Daniele Lamponi
CEA Saclay
DM2S/SERMA
91191 Gif-sur-Yvette (France)
igor.zmijarevic@cea.fr, richard.sanchez@cea.fr

Keywords: Nodal; Transport; DSA

ABSTRACT

Higher order differencing schemes for solving discrete ordinates transport equations in multidimensional geometries, such as nodal and characteristics methods, are well developed and implemented in production codes. In order to have a stable and rapidly converging solution for the problems of reactor physics analysis it is important that these methods are accompanied by an efficient acceleration technique. This paper describes a consistently discretized diffusion synthetic acceleration for linear nodal transport schemes in weighted difference form. This is an extension of a previously developed and analyzed constant-flux scheme by Azmy. The new extension of the method is shown to be stable for any optical mesh size. Stability of results by Fourier analysis and results for simple model problems are presented.

1. INTRODUCTION

The consistent formulation of a diffusion synthetic acceleration (DSA) equation, derived directly from the discretized form of the transport equation following the methods of Alcouffe (1977) and Larsen (1982), encounters problems in the practical application to higher order multidimensional differencing schemes because of the relatively complex form of the resulting equation and the difficulties to solve it efficiently. Even the simplest XY diamond-differencing scheme results in a DSA system with nine-point coupling, but leaves a multiple choice for simplifications which give "nearly-consistent" DSA schemes, that are easier to solve (AboAlfaraj, 1991). Alternative approaches applied to nodal transport methods, like the method of Khalil (1985, 1988) in which the starting point is the continuous transport equation, or that of Azmy (1993) where the cell-centered discretization is used, yield DSA equations in a much simpler form. Several authors (Adams, 1988; Lawrence, 1986) have developed transport-based synthetic acceleration methods, where the acceleration equation is derived directly from the discretized transport equation and can be solved efficiently with minimal programming effort. However, for diffusive-mode-dominated problems, the use of transport acceleration may not be the optimal solution

because the diffusion equation can be solved faster. Azmy has shown that arbitrarily high-order multidimensional nodal methods can be cast in a weighted-difference form (Azmy, 1988) which is suitable for the formulation of the DSA algorithm. The stability and convergence of such DSA scheme has been proved for the lowest order nodal method, i.e. constant-surface flux (Azmy, 1985, 1987), but no attempt has been made to apply this algorithm to higher order differencing schemes. Here we describe the implementation of DSA to the linear and bilinear nodal approximations taking into account linearly anisotropic scattering.

The paper is organized as follows. In Sec. 2 we give a brief description of the linear flux approximations used in the IDT mock-up code (Zmijarevic, 1999), and the relation between the weighted difference formulation and the one used in the code for the S_N transport sweep. Sec. 3 describes the implementation of the DSA scheme for linear nodal approximations. Sec. 4 presents stability results by Fourier analysis in the case of an infinite homogeneous medium. Results for simple model problems are given in Sec. 5.

2. TRANSPORT SOLUTION WITH LINEAR FLUX APPROXIMATIONS

The IDT code and its one-group solver implemented in the Apollo2 code (Loubiere, 1999) solve the discrete ordinates transport equation in XY and XYZ geometry on a regular cartesian mesh, using either nodal or characteristics methods. For the sake of simplicity we consider the two-dimensional transport equation for a rectangular mesh cell of size $\Delta x \times \Delta y$ in terms of dimensionless variables $-1 \leq x, y \leq 1$.

$$(\omega_x \partial_x + \omega_y \partial_y + \sigma)\psi(x,y) = Q(x,y) \tag{1}$$

where the index of the discrete direction has been omitted, and $\omega_u = 2\mu_u/\Delta u$, $u = x, y$. The extension to three dimensional geometry is straightforward. Both nodal and characteristics methods use Legendre polynomial representations for the spatial dependence of the angular flux within the mesh cell interior and on its boundaries. The cell interior angular flux for each discrete direction is

$$\psi(x,y) = \sum_{mn \geq 0} (2m+1)(2n+1)\psi_{mn} P_m(x) P_n(y). \tag{2}$$

A similar expansion is used for the angular source. The incoming $(-)$ and exiting $(+)$ angular flux on a surface perpendicular to the x-axis in positive direction, for example, is

$$\psi^{x\pm}(y) = \psi(\pm 1, y) = \sum_{n \geq 0} (2n+1)\psi_n^{x\pm} P_n(y) \tag{3}$$

with analogous expressions for the flux on other surfaces. The spatial moments in the cell interior and on the boundaries are, respectively,

$$\psi_{mn} = \frac{1}{4}\int_{-1}^{+1} dx dy P_m(x) P_n(y) \psi(x,y) \quad \text{and} \quad \psi_n^{x\pm} = \frac{1}{2}\int_{-1}^{+1} dy P_n(y) \psi^{x\pm}(y). \tag{4}$$

Two different linear expansions are implemented in the code: a pure linear approximation, and a bilinear, which retains all the terms up to order 1. Taking the mn-th moment of Eq. (1) a set of balance equations for the spatial flux moment is obtained. In matrix form

$$\boldsymbol{\beta}^{+}\boldsymbol{\psi}^{+} - \boldsymbol{\beta}^{-}\boldsymbol{\psi}^{-} + \boldsymbol{\Sigma}\boldsymbol{\psi} = \boldsymbol{Q} \tag{5}$$

where the vectors $\boldsymbol{\psi}$, $\boldsymbol{\psi}^{\pm}$ and $\boldsymbol{Q}$ contain the spatial approximation of the flux and the source in one discrete direction, $\boldsymbol{\beta}^{\pm}$ are sparse matrices whose elements depend on direction cosines and mesh size, and $\boldsymbol{\Sigma}$ depends on direction, mesh size and total cross section. In the discrete ordinates method the angular flux within each mesh cell, ψ, and on its exiting surfaces, ψ^{+}, are computed in terms of the within cell source, Q, and the incoming fluxes on the incident surfaces, ψ^{-}. The solution algorithm, thus, can be expressed by

$$\boldsymbol{\psi} = \boldsymbol{C}\boldsymbol{Q} + \boldsymbol{I}\boldsymbol{\psi}^{-}, \tag{6}$$

$$\boldsymbol{\psi}^{+} = \boldsymbol{E}\boldsymbol{Q} + \boldsymbol{T}\boldsymbol{\psi}^{-}, \tag{7}$$

where the matrix elements of $\boldsymbol{C}$, $\boldsymbol{I}$, $\boldsymbol{E}$ and $\boldsymbol{T}$ represent respectively collision, surface-to-volume, escape and transmission coefficients. This is the form used in the S_N sweep routine of IDT with the matrix coefficients precomputed and stored for each mesh cell type and for each of the discrete directions of the first octant. Equation (5) together with Eqs. (6-7) show that the matrix coefficients satisfy the following conservation relations:

$$\boldsymbol{\beta}^{+}\boldsymbol{E} + \boldsymbol{\Sigma}\boldsymbol{C} = \mathbf{1}, \tag{8}$$

$$\boldsymbol{\beta}^{+}\boldsymbol{T} + \boldsymbol{\Sigma}\boldsymbol{I} = \boldsymbol{\beta}^{-}. \tag{9}$$

The solution algorithm can formally be cast into the weighted difference form:

$$\frac{1}{2}(\mathbf{1} + \boldsymbol{\alpha})\boldsymbol{\psi}^{+} + \frac{1}{2}(\mathbf{1} - \boldsymbol{\alpha})\boldsymbol{\psi}^{-} = \boldsymbol{A}\boldsymbol{\psi}, \tag{10}$$

where $\mathbf{1}$ is identity matrix. To obtain the expressions for $\boldsymbol{\alpha}$ and $\boldsymbol{A}$ we eliminate the source term in Eqs. (6-7) and replace the resulting $\boldsymbol{\psi}^{+}$ in Eq. (10):

$$\frac{1}{2}(\mathbf{1} + \boldsymbol{\alpha})(\boldsymbol{E}\boldsymbol{C}^{-1}\boldsymbol{\psi} - \boldsymbol{X}\boldsymbol{\psi}^{-}) = \boldsymbol{A}\boldsymbol{\psi} - \frac{1}{2}(\mathbf{1} - \boldsymbol{\alpha})\boldsymbol{\psi}^{-} \tag{11}$$

where $\boldsymbol{X} = \boldsymbol{E}\boldsymbol{C}^{-1}\boldsymbol{I} - \boldsymbol{T}$, This entails

$$\boldsymbol{A} = \frac{1}{2}(\mathbf{1} + \boldsymbol{\alpha})\boldsymbol{E}\boldsymbol{C}^{-1}, \tag{12}$$

$$\boldsymbol{\alpha} = (\mathbf{1} + \boldsymbol{X})^{-1}(\mathbf{1} - \boldsymbol{X}). \tag{13}$$

For the case of nodal methods the resulting $\boldsymbol{\alpha}$ is diagonal. This result was obtained in a different way by Azmy (1988). In linear nodal scheme, for example, and for moment ordering as $\boldsymbol{\psi} = \begin{pmatrix}\psi_{00} & \psi_{10} & \psi_{01}\end{pmatrix}^{T}$ and $\boldsymbol{\psi}^{\pm} = \begin{pmatrix}\psi_0^{x\pm} & \psi_1^{x\pm} & \psi_0^{x\pm} & \psi_1^{x\pm}\end{pmatrix}^{T}$, the diagonal of $\boldsymbol{\alpha}$ contains values $(\alpha_1^x, \alpha_0^x, \alpha_1^y, \alpha_0^y)$, where α_N^x is the weight of one-dimensional nodal scheme of order N. Equation (10) yields explicit expressions for the angular fluxes on exiting surfaces. However, this is not possible for the case of the method of characteristics because $\boldsymbol{\alpha}$ is a full matrix.

3. DSA FOR LINEAR NODAL SCHEMES IN XY GEOMETRY

We just showed that the two systems given respectively by Eqs. (6-7) and Eqs. (5,10) are equivalent. The first system allows to implement a unique computational routine, regardless of the method used, but the second is more suitable to formulate a DSA equation because for nodal methods Eqs. (10) has a simple form. Here we describe the construction of the acceleration equations. The following notation will be used: $\psi^{nm}_{d,i,j}$ is the nm-th nodal moment of the angular flux in the interior of mesh cell (i,j) in discrete direction d, $\psi^{n}_{d,i\pm1/2,j}$ is the n-th nodal moment of the angular flux at the mesh cell boundary in discrete direction d, $\phi^{nm}_{\ell k,i,j}$ is the nm-th nodal moment of the ℓk-th angular moment of the flux in the interior of the mesh cell (i,j), $\phi^{n}_{\ell k,i\pm1/2,j}$ is the n-th nodal moment of the ℓk-th angular moment of the flux at the mesh cell boundary. A similar notation holds for the directional source, $Q^{mn}_{d,i,j}$, and its moments, $q^{nm}_{\ell k,i,j}$. Also h_i and h_j are the mesh cell thicknesses, respectively, in x and y directions, l is inner iteration index and $\tilde{l}$ denotes the $(l+1)$-th unaccelerated iteration. The angular flux and its moments are related via discrete ordinates quadrature:

$$\psi^{nm}_{d}=\sum_{\ell k}Y_{\ell k,d}\phi^{nm}_{\ell k},\qquad \phi^{nm}_{\ell k}=L_{\ell k}\psi^{nm}_{d}=\sum_{d}w_d Y_{\ell k,d}\psi^{nm}_{d}, \tag{14}$$

where $Y_{\ell k,d}$ are spherical harmonics expansion coefficients. In bilinear nodal approximation, the system of four balance equations for the angular flux in each discrete direction and for each mesh cell is

$$\frac{\mu_d}{h_i}\big(\psi^{0,\tilde{l}}_{d,i+1/2,j}-\psi^{0,\tilde{l}}_{d,i-1/2,j}\big)+\frac{\eta_d}{h_j}\big(\psi^{0,\tilde{l}}_{d,i,j+1/2}-\psi^{0,\tilde{l}}_{d,i,j-1/2}\big)+\sigma_{i,j}\psi^{00,\tilde{l}}_{d,i,j}=Q^{00,l}_{d,i,j}, \tag{15}$$

$$\frac{\mu_d}{h_i}\big(\psi^{0,\tilde{l}}_{d,i+1/2,j}+\psi^{0,\tilde{l}}_{d,i-1/2,j}-2\psi^{00,\tilde{l}}_{d,i,j}\big)+\frac{\eta_d}{h_j}\big(\psi^{1,\tilde{l}}_{d,i,j+1/2}-\psi^{1,\tilde{l}}_{d,i,j-1/2}\big)+\sigma_{i,j}\psi^{10,\tilde{l}}_{d,i,j}=Q^{10,l}_{d,i,j}, \tag{16}$$

$$\frac{\mu_d}{h_i}\big(\psi^{1,\tilde{l}}_{d,i+1/2,j}-\psi^{1,\tilde{l}}_{d,i-1/2,j}\big)+\frac{\eta_d}{h_j}\big(\psi^{0,\tilde{l}}_{d,i,j+1/2}+\psi^{0,\tilde{l}}_{d,i,j-1/2}-2\psi^{00,\tilde{l}}_{d,i,j}\big)+\sigma_{i,j}\psi^{01,\tilde{l}}_{d,i,j}=Q^{01,l}_{d,i,j}, \tag{17}$$

$$\frac{\mu_d}{h_i}\big(\psi^{1,\tilde{l}}_{d,i+1/2,j}+\psi^{1,\tilde{l}}_{d,i-1/2,j}-2\psi^{01,\tilde{l}}_{d,i,j}\big)+\frac{\eta_d}{h_j}\big(\psi^{1,\tilde{l}}_{d,i,j+1/2}+\psi^{1,\tilde{l}}_{d,i,j-1/2}-2\psi^{10,\tilde{l}}_{d,i,j}\big)+\sigma_{i,j}\psi^{11,\tilde{l}}_{d,i,j}=Q^{11,l}_{d,i,j}. \tag{18}$$

This system is solved together with four weighted-difference nodal equations:

$$\frac{1}{2}(1+\alpha^x_d)\psi^{n}_{d,i+1/2,j}+\frac{1}{2}(1-\alpha^x_d)\psi^{n}_{d,i-1/2,j}=\psi^{0n}_{d,i,j}+3\alpha^x_d\psi^{1n}_{d,i,j}\quad,\quad n=0,1, \tag{19}$$

$$\frac{1}{2}(1+\alpha^y_d)\psi^{m}_{d,i,j+1/2}+\frac{1}{2}(1-\alpha^y_d)\psi^{m}_{d,i,j-1/2}=\psi^{m0}_{d,i,j}+3\alpha^y_d\psi^{m1}_{d,i,j}\quad,\quad m=0,1. \tag{20}$$

The source moments in these equations depend on the flux from the previous iteration:

$$Q^{nm,l}_{d,i,j}=\sum_{\ell\geq0}(2\ell+1)\sigma_{s\ell,i,j}\sum_{k\geq0}^{\ell}Y_{\ell k}\phi^{nm,l}_{\ell k}+Q^{nm,l}_{\text{ext},d,i,j}. \tag{21}$$

In IDT the implementation of the method has been limited to symmetric quadrature formulas. Following Larsen (1982) and Azmy (1987) we adopt the expansion

$$\alpha_d^x = \sum_{\ell k} Y_{\ell k,d}\alpha_{\ell k}^x = \alpha_{00}^x + 3\mu\alpha_{10}^x + 3\eta\alpha_{11}^x + \alpha_R^x, \tag{22}$$

where $\alpha_{\ell k} = L_{\ell k}\alpha$, and α_R contains all the terms with spherical harmonics expansion coefficients of greater than one. Because α^x and α^y are odd functions of the corresponding direction cosine: $\alpha_{00}^x = \alpha_{11}^x = \alpha_{10}^y = 0$. Therefore the P_1 expansion gives a cubic error. Here we follow Larsen's standard "four-step" procedure. First we project the previous system on each of the spherical harmonics. By applying L_{00} and L_{10} on balance equations (15-18) by we obtain

$$\frac{1}{h_i}\big(\phi_{10,i+1/2,j}^{0,\tilde{l}} - \phi_{10,i-1/2,j}^{0,\tilde{l}}\big) + \frac{1}{h_j}\big(\phi_{11,i,j+1/2}^{0,\tilde{l}} - \phi_{11,i,j-1/2}^{0,\tilde{l}}\big) + \sigma_{i,j}\phi_{00,i,j}^{00,\tilde{l}} = q_{00,i,j}^{00,l} \tag{23}$$

$$\frac{1}{h_i}\big(\phi_{10,i+1/2,j}^{0,\tilde{l}} + \phi_{10,i-1/2,j}^{0,\tilde{l}} - 2\phi_{10,i,j}^{00,\tilde{l}}\big) + \frac{1}{h_j}\big(\phi_{11,i,j+1/2}^{1,\tilde{l}} - \phi_{11,i,j-1/2}^{1,\tilde{l}}\big) + \sigma_{i,j}\phi_{00,i,j}^{10,\tilde{l}} = q_{00,i,j}^{10,l} \tag{24}$$

$$\frac{1}{h_i}\big(\phi_{10,i+1/2,j}^{1,\tilde{l}} - \phi_{10,i-1/2,j}^{1,\tilde{l}}\big) + \frac{1}{h_j}\big(\phi_{11,i,j+1/2}^{0,\tilde{l}} + \phi_{11,i,j-1/2}^{0,\tilde{l}} - 2\phi_{11,i,j}^{00,\tilde{l}}\big) + \sigma_{i,j}\phi_{00,d,i,j}^{01,\tilde{l}} = q_{00,i,j}^{01,l} \tag{25}$$

$$\frac{1}{h_i}\big(\phi_{10,i+1/2,j}^{1,\tilde{l}} + \phi_{10,i-1/2,j}^{1,\tilde{l}} - 2\phi_{10,i,j}^{01,\tilde{l}}\big) + \frac{1}{h_j}\big(\phi_{11,d,i,j+1/2}^{1,\tilde{l}} + \phi_{11,d,i,j-1/2}^{1,\tilde{l}} - 2\phi_{11,i,j}^{10,\tilde{l}}\big) + \sigma_{i,j}\phi_{00,i,j}^{11,\tilde{l}} = q_{00,i,j}^{11,l} \tag{26}$$

and

$$\frac{1}{h_i}\frac{1}{3}\big(\phi_{00,i+1/2,j}^{0,\tilde{l}} - \phi_{00,i-1/2,j}^{0,\tilde{l}}\big) + \sigma_{i,j}\phi_{10,i,j}^{00,\tilde{l}} = q_{10,i,j}^{00,l} \tag{27}$$

$$\frac{1}{h_i}\frac{1}{3}\big(\phi_{00,i+1/2,j}^{0,\tilde{l}} + \phi_{00,i-1/2,j}^{0,\tilde{l}} - 2\phi_{00,i,j}^{00,\tilde{l}}\big) + \sigma_{i,j}\phi_{10,i,j}^{10,\tilde{l}} = q_{10,i,j}^{10,l} \tag{28}$$

$$\frac{1}{h_i}\frac{1}{3}\big(\phi_{00,i+1/2,j}^{1,\tilde{l}} - \phi_{00,i-1/2,j}^{1,\tilde{l}}\big) + \sigma_{i,j}\phi_{10,d,i,j}^{01,\tilde{l}} = q_{10,i,j}^{01,l} \tag{29}$$

$$\frac{1}{h_i}\frac{1}{3}\big(\phi_{00,i+1/2,j}^{1,\tilde{l}} + \phi_{00,i-1/2,j}^{1,\tilde{l}} - 2\phi_{00,i,j}^{01,\tilde{l}}\big) + \sigma_{i,j}\phi_{10,i,j}^{11,\tilde{l}} = q_{10,i,j}^{11,l}. \tag{30}$$

Four equations more are obtained analogously applying L_{11}. Next, operating with L_{00} and L_{10} on weighted-difference equations (19) gives

$$\frac{1}{2}\big(\phi_{00,i+1/2,j}^{n,\tilde{l}} + \phi_{00,i-1/2,j}^{n,\tilde{l}}\big) + \frac{3\alpha_{10}^x}{2}\big(\phi_{10,i+1/2,j}^{n,\tilde{l}} - \phi_{10,i-1/2,j}^{n,\tilde{l}}\big) + \frac{\alpha_R^x}{2}\big(\phi_{00,i+1/2,j}^{n,\tilde{l}} - \phi_{00,i-1/2,j}^{n,\tilde{l}}\big) = \phi_{00,i,j}^{0n,\tilde{l}} + 9\big(\alpha_{10}^x\phi_{10,i,j}^{1n,\tilde{l}} + \alpha_R^x\phi_{00,i,j}^{1n,\tilde{l}}\big) \tag{31}$$

$$\frac{1}{2}\big(\phi_{10,i+1/2,j}^{n,\tilde{l}} + \phi_{10,i-1/2,j}^{n,\tilde{l}}\big) + \frac{\alpha_{10}^x}{2}\big(\phi_{00,i+1/2,j}^{n,\tilde{l}} - \phi_{00,i-1/2,j}^{n,\tilde{l}}\big) + \frac{\alpha_R^x}{2}\big(\phi_{10,i+1/2,j}^{n,\tilde{l}} - \phi_{10,i-1/2,j}^{n,\tilde{l}}\big) = \phi_{10,i,j}^{0n,\tilde{l}} + 3\big(\alpha_{10}^x\phi_{00,i,j}^{1n,\tilde{l}} + \alpha_R^x\phi_{10,i,j}^{1n,\tilde{l}}\big) \tag{32}$$

Similar equations for y direction are obtained by applying L_{00} and L_{11}. The set of all these equations is the P_1 approximation of Eqs. (15-20) Note that two pairs of equations are decoupled from this set, i.e. y-current components on vertical faces and x-currents on horizontal ones. These equations state that the diamond relation holds for the corresponding moments.

In the "four-step" procedure a new set of equations is constructed from Eqs. (23-32) and the remaining analogues by replacing the iteration index $\tilde{l}$ by $l+1$ in all terms except in those containing α_R. These two sets of equations are subtracted in order to provide a system of equations for the correction terms:

$$f^{mn}_{\ell k,i,j} = \phi^{mn,l+1}_{\ell k,i,j} - \phi^{mn,\tilde{l}}_{\ell k,i,j} \qquad \text{and} \qquad f^{n}_{\ell k,i\pm 1/2,j} = \phi^{n,l+1}_{\ell k,i\pm 1/2,j} - \phi^{n,\tilde{l}}_{\ell k,i\pm 1/2,j}. \tag{33}$$

The resulting system of twenty equations, having for unknowns all angular moments of cell-interior and surface flux correction terms, must be manipulated in order to obtain a linear algebraic system suitable for inversion. Elimination of all cell-interior moments and all anisotropic moments of cell-surface correction terms yields, after a considerable algebraic manipulation, a set of equations relating isotropic components of surface flux corrections:

$$\begin{pmatrix} \boldsymbol{D}_{x0} & \mathbf{0} & \boldsymbol{X}_{00} & \boldsymbol{X}_{01} \\ \mathbf{0} & \boldsymbol{D}_{x1} & \boldsymbol{X}_{10} & \boldsymbol{X}_{11} \\ \boldsymbol{Y}_{00} & \boldsymbol{Y}_{01} & \boldsymbol{D}_{y0} & \mathbf{0} \\ \boldsymbol{Y}_{10} & \boldsymbol{Y}_{11} & \mathbf{0} & \boldsymbol{D}_{y1} \end{pmatrix} \begin{pmatrix} \boldsymbol{f}_{x0} \\ \boldsymbol{f}_{x1} \\ \boldsymbol{f}_{y0} \\ \boldsymbol{f}_{y1} \end{pmatrix} = \boldsymbol{g} \tag{34}$$

Here the $\boldsymbol{f}_{yn}$ and $\boldsymbol{f}_{xn}$, $n = 0, 1$, are flux correction vectors for, respectively, horizontal and vertical cell interfaces, $\boldsymbol{D}$ are tridiagonal matrices, and $\boldsymbol{X}$ and $\boldsymbol{Y}$ are sparse matrices with a maximum of four non-zero elements per row.

In the present implementation, system (34) is solved iteratively by the alternating direction method accelerated by successive over-relaxation using the estimated spectral radius.

4. FOURIER ANALYSIS

Fourier analysis allows to investigate the convergence properties for a simple model, in this case, an infinite, homogeneous medium with a uniform spatial mesh. Evidently, such a model is far from real practical applications with cross sections discontinuities and non-uniform mesh, but experience shows that often the spectral radii are not far from the theoretically predicted values.

We introduce the Fourier ansatz:

$$\phi^{mn,l} = e^{i\vec{\lambda}\vec{r}}, \qquad \phi^{mn,l+1} = \omega_{mn} e^{i\vec{\lambda}\vec{r}}, \qquad \phi^{mn,\tilde{l}} = \Phi^{mn,\tilde{l}} e^{i\vec{\lambda}\vec{r}},$$
$$\psi^{mn,\tilde{l}} = \Psi^{mn,\tilde{l}} e^{i\vec{\lambda}\vec{r}} \qquad \psi^{x\pm,n} = \Psi^{x\pm,n} e^{i\vec{\lambda}\vec{r}} e^{\pm\lambda_x \Delta x/2} \qquad \psi^{y\pm,m} = \Psi^{y\pm,m} e^{i\vec{\lambda}\vec{r}} e^{\pm\lambda_y \Delta y/2}$$

Then, from the equations for a free transport iteration we express the $\tilde{l}$ Fourier components in terms of the l components. Next, we compute the source $\boldsymbol{g} = \boldsymbol{\psi}^{\tilde{l}} - \boldsymbol{\psi}^{l}$ and replace it in

acceleration equations (34). Finally, since $\boldsymbol{f} = \boldsymbol{\psi}^{l+1} - \boldsymbol{\psi}^{\tilde{l}}$ this equation can be written as $\boldsymbol{\psi}^{l+1} = \boldsymbol{\omega}\boldsymbol{\psi}^{l}$. Because of complicated general algebraic expressions, the eigenvalues in the search for spectral radius were computed numerically. Given the quadrature formula, mesh size and scattering ratio, c, we search for

$$\sup_{mn}\Big(\sup_{\lambda_x,\lambda_y}\omega_{mn}(\lambda_x,\lambda_y)\Big)$$

in the interval of periodicity for $(\lambda_x, \lambda y) \in (0, \pi/2)$ using steps of order 10^{-2}. The matrix eigenvalues are computed using the LAPACK package (Anderson, 1999).

Figure 1 shows Fourier analysis results for the linear and bilinear schemes in the case of isotropic scattering and for a scattering ratio c=0.999, for the mesh sizes varying separately in x and y it the range of 10^{-5} to 10. The calculations were done with the S_8 level symmetric formula. The DSA scheme is shown to be stable in the whole range and the maximum spectral ratio found for S_4, S_8 and S_{16} sets does not exceed the value of 0.2276. The behavior of both approximations is similar to that of the constant-constant approximation which has been analyzed previously (Azmy, 1987), although they are more sensitive to the mesh size. The investigation of spectral radii was carried out for the values of scattering ratio strictly less than one in order to avoid singularities which are due to the manner in which the Fourier analysis is implemented numerically. However, DSA equation itself is not singular for c=1.

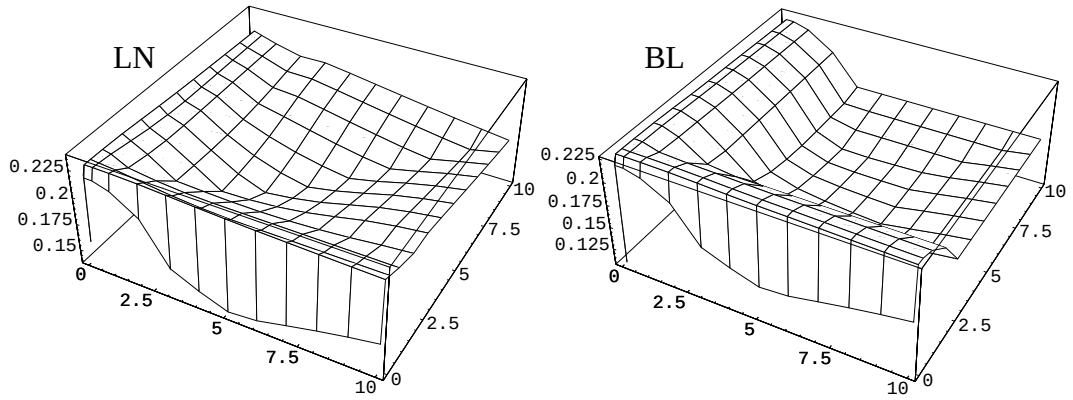


Figure 1: Results of XY geometry Fourier analysis: spectral radius versus mesh size for the linear (LN) and bilinear (BL) schemes with S_8 quadrature and scattering ratio=0.999

5. NUMERICAL RESULTS

We present here two classical model problems in XY geometry: the EIR2 benchmark (Fig. 2) and the Iron-Water Shielding problem (Fig. 3). Further, in order to investigate

the efficiency of DSA for cases with high heterogeneity, we considered Burre's Suite of Test Problems (BSTeP given in Fig. 4). For the latter example, in which the total cross section and mesh size vary in intervals of several orders of magnitude, we choose the same cases investigated by Azmy (2000). All cases are one-group fixed source problems. The required pointwise convergence was 10^{-4} for all flux moments.

Tables 1 to 4 respectively show the gain in number of iterations and CPU time for the first two problems run on a DEC Alpha workstation for different angular quadratures and spatial meshes. A significant speedup is obtained on coarse meshes, while the efficiency diminishes with mesh refinement. This is due to the large number of iterations needed to solve the DSA equation with the alternating direction method which exhibits a very slow convergence on small meshes. In fact, only in the cases that converge in 6 to 8 iterations for the first two problems the acceleration equation satisfies the requested convergence criterion. On finer meshes DSA equation remains unconverged after 100 alternating directions iterations, which consequently increases the number of inner iterations. Pushing the acceleration equation to a full convergence results in a less significant gain in CPU time. Table 5 shows the results for BSTeP using the linear nodal scheme. The bilinear scheme has a very similar behavior. It can be noted that for small values of cross sections and small mesh sizes the acceleration does not contribute much, or not at all, because the problem is leakage dominated and the diffusive modes which are suppressed by DSA are very weak. On the contrary, for larger geometrical problems, especially for optically large meshes, DSA is very efficient. Unaccelerated iterations in these cases reached the imposed limit of 3000 iterations without converging.

6. CONCLUSION

Based on ideas that have already been applied to the lower-order nodal methods, we have formulated and implemented an unconditionally stable, consistently formulated DSA scheme for linear multidimensional nodal methods in weighted-difference form. The final acceleration equation retains a relatively simple form for interface current equations with matrix coefficients which are inexpensive to compute. The method gives a rapid convergence for cases having a scattering ratio close to 1. However, work is still to be done in improving the method for the solution of acceleration equation. The alternating direction scheme which is presently used converges slowly on small meshes. Work on a multigrid algorithm is in progress.

The ad-hoc use of nodal DSA to accelerate the characteristic transport scheme, which is evidently inconsistent, shows a satisfying behavior in many cases. For meshes having small aspect ratio the behavior is nearly the same as for the nodal transport scheme, but fails for large values of $\Delta x/\Delta y$. This was expected because the coupling of flux moments in nodal and characteristics schemes is of different nature. It remains to analyze the parity properties of weighting coefficients for the characteristics method. Although the $\boldsymbol{\alpha}$ and $\boldsymbol{A}$ matrices in Eq. (10) are full, many elements vanish under integration in angle, so the consistent DSA again contains sparse matrices.

REFERENCES

[AboAlfaraj 1991] T. G. AboAlfaraj, E. W. Larsen. The Efficiency of Linear and Nonlinear Implementations of Diffusion Synthetic Acceleration. In *Proceedings of ANS Conference on Advances in Mathematics, Computations, and Reactor Physics*, Pittsburgh, PA, (1991).

[Adams 1988] M. L. Adams, W. R. Martin. Boundary Projection Acceleration: A New Approach to Synthetic Acceleration of Transport Calculations. *Nucl. Sci. Eng.*, **100,** 177, 1988.

[Alcouffe 1977] R. E. Alcouffe. Diffusion Synthetic Acceleration Method for the Diamond-Differenced Discrete-Ordinates Equations. *Nucl. Sci. Eng.*, **64,** 344, 1977.

[Anderson 1999] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition, 1999.

[Azmy 1985] Y. Y. Azmy, J. J. Dorning. Diffusion Synthetic Acceleration of Multi-Dimensional Discrete Nodal Transport Method. In *Proceedings of ANS Topical Meeting on Advances in Nuclear Engineering Computational Methods*, Knoxville, 1985, page 440.

[Azmy 1987] Y. Y. Azmy, E. W. Larsen. Fourier Analysis of Diffusion Synthetic Acceleration Method for Weighted Diamond Differencing Schemes in Cartesian Geometries. *Nucl. Sci. Eng.*, **95,** 106, 1987.

[Azmy 1988] Y. Y. Azmy. The Weighted Diamond-Difference Form of Nodal Transport Methods. *Nucl. Sci. Eng.*, **98,** 29, 1988.

[Azmy 1993] Y. Y. Azmy. Cell-Centered Imposed Diffusion Synthetic Acceleration for Weighted Difference Transport Methods. *Nucl. Sci. Eng.*, **115,** 265, 1993.

[Azmy 2000] Y. Y. Azmy. Acceleration of Multidimensional Discrete Ordinate Methods Via Adjacent-Cell Preconditioners. *Nucl. Sci. Eng.*, **136,** 202, 2000.

[Khalil 1985] H. Khalil. A Nodal Diffusion Technique for Synthetic Acceleration of Nodal S_n Calculations. *Nucl. Sci. Eng.*, **90,** 263, 1985.

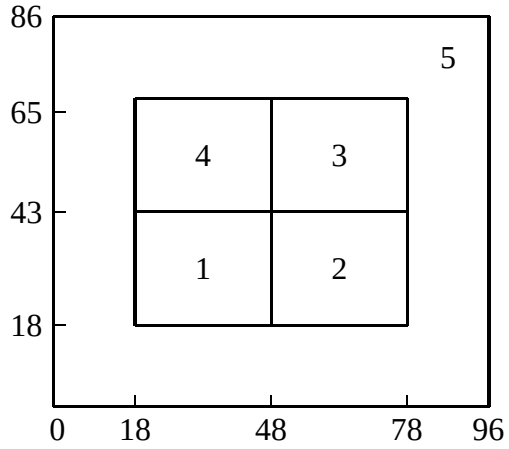
[Khalil 1988] H. Khalil. Effectiveness of a Consistently Formulated Diffusion-Synthetic Acceleration Differencing Approach. *Nucl. Sci. Eng.*, **98,** 226, 1988.

[Larsen 1982] E. W. Larsen. Unconditionally Stable Diffusion-Synthetic Acceleration Methods for the Slab Geometry Discrete Ordinates Equations. Part I: Theory. *Nucl. Sci. Eng.*, **82,** 47, 1982.

[Lawrence 1986] R. D. Lawrence. An Interface-Current Approach to Synthetic Acceleration of Three Dimensional Discrete Ordinates Transport Methods. *Trans. Am. Nucl. Soc.*, **53,** 280, 1986.

[Loubiere 1999] Loubiere, S., Sanchez, R., Coste, M., Hebert., A, Stankovski., Z., Van Der Gucht., C. and Zmijarevic, I. APOLLO2 Twelve Years Later. In *Proc. Int. Conf. on Math. and Computations, Reactor Physics and Environmental Analysis in Nucl. Applications,* Madrid, (1999), page 1298.

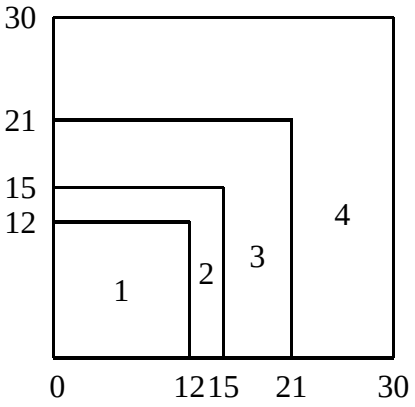
[Zmijarevic 1999] I. Zmijarevic. Multidimensional Discrete Ordinates Nodal and Characteristics Methods for the Apollo2 Code. In *Proc. Int. Conf. on Math. and Computations, Reactor Physics and Environmental Analysis in Nucl. Applications,* Madrid, (1999), page 1587.



zone	source	σ	σ_{s0}	σ_{s1}
1	1	.60	.53	.27
2	0	.48	.20	.02
3	1	.70	.66	.30
4	0	.65	.50	.15
5	0	.90	.89	.40

vacuum boundary conditions

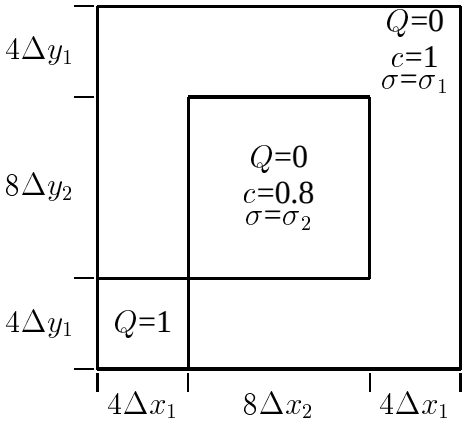
Figure 2: EIR2 benchmark



zone	source	σ	σ_{s0}	σ_{s1}
1	1	3.3333	3.3136	0.9256
2	0	3.3333	3.3136	0.9256
3	0	1.3333	1.1077	0.0367
4	0	3.3333	3.3136	0.9256

vacuum boundary at top and right
reflective at left and bottom

Figure 3: Iron Water Shielding Problem



Mesh: 16×16
$\Delta x_1 = \Delta x_2 = 1$
σ_1, σ_2 : {0.01, 0.1, 1, 10}
$\Delta y_1, \Delta y_2$: {0.01, 0.1, 1}

vacuum boundary conditions

Figure 4: Burre's Suite of Test Problems

Table 1: Results for IAEA EIR-2 benchmark with linearly anisotropic scattering and pointwise convergence of 10^{-4} versus mesh size and S_N order: number of free (SI) and accelerated (DSA) iterations versus number of meshes along an axis (N_x) and S_N order.

		linear			bilinear		
S_N	N_x	SI	DSA	gain	SI	DSA	gain
S_4	8	498	6	83.0	549	6	91.5
	16	535	6	89.2	531	6	88.5
	32	526	6	87.7	526	7	75.1
	64	526	12	43.8	526	14	37.6
	128	526	38	13.8	526	42	12.5
S_8	8	496	6	82.7	541	7	77.3
	16	535	7	76.4	531	6	88.5
	32	526	6	87.7	526	7	75.1
	64	526	12	43.8	526	14	37.6
	128	526	38	13.8	526	42	12.5
S_{16}	8	496	7	70.9	539	7	77.0
	16	535	6	89.2	531	6	88.5
	32	526	6	87.7	526	7	75.1
	64	526	12	43.8	526	14	37.6
	128	526	38	13.8	526	42	12.5

Table 2: Results for IAEA EIR-2 benchmark with linearly anisotropic scattering and pointwise convergence of 10^{-4} versus mesh size and S_N order: CPU time in seconds for free (SI) and accelerated (DSA) iterations versus number of meshes along an axis (N_x) and S_N order.

		linear			bilinear		
S_N	N_x	SI	DSA	gain	SI	DSA	gain
S_4	8	0.33	0.01	33.0	0.46	0.01	46.0
	16	1.27	0.07	18.1	1.66	0.08	20.8
	32	5.25	0.28	18.8	7.09	0.37	19.2
	64	27.13	2.82	9.6	39.22	3.81	10.3
	128	145.47	43.36	3.4	221.99	57.72	3.8
S_8	8	0.99	0.02	49.5	1.35	0.03	45.0
	16	3.96	0.12	33.0	5.06	0.12	42.2
	32	15.75	0.41	38.4	21.70	0.56	38.8
	64	83.01	4.11	20.2	119.54	5.96	20.1
	128	490.66	68.07	7.2	657.49	94.99	6.9
S_{16}	8	3.75	0.06	62.5	5.18	0.08	64.8
	16	13.61	0.21	64.8	17.65	0.26	67.9
	32	54.63	0.85	64.3	71.87	1.24	58.0
	64	292.17	8.67	33.7	415.44	13.86	30.0
	128	1572.31	156.61	10.0	2168.19	222.20	9.8

Table 3: Results for iron-water benchmark with linearly anisotropic scattering and pointwise convergence of 10^{-4} versus mesh size (Δ in cm) and S_N order: number of free (SI) and accelerated (DSA) iterations.

		linear			bilinear		
S_N	Δ	SI	DSA	gain	SI	DSA	gain
S_4	3.0	1189	6	198.2	1134	6	189.0
	1.0	1174	7	167.7	1166	8	145.8
	0.5	1169	15	77.9	1168	19	61.5
S_8	3.0	1181	7	168.7	1133	6	188.8
	1.0	1174	7	167.7	1166	8	145.8
	0.5	1169	15	77.9	1168	19	61.5
S_{16}	3.0	1178	8	147.3	1133	6	188.8
	1.0	1173	7	167.6	1167	8	145.9
	0.5	1169	15	77.9	1168	19	61.5

Table 4: Results for iron-water benchmark with linearly anisotropic scattering and pointwise convergence of 10^{-4} versus mesh size (Δ in cm) and S_N order: CPU time in seconds for free (SI) and accelerated (DSA) iterations.

		linear			bilinear		
S_N	Δ	SI	DSA	gain	SI	DSA	gain
S_4	3.0	1.13	0.03	37.7	1.33	0.04	33.3
	1.0	9.68	0.29	33.4	12.45	0.39	31.9
	0.5	41.00	2.89	14.2	53.08	4.02	13.2
S_8	3.0	3.27	0.05	65.4	3.96	0.05	79.2
	1.0	28.96	0.41	70.6	37.49	0.56	66.9
	0.5	121.69	4.09	29.8	166.09	5.98	27.8
S_{16}	3.0	11.52	0.12	96.0	13.73	0.10	137.3
	1.0	101.39	0.84	120.7	128.38	1.18	108.8
	0.5	462.02	7.81	59.2	565.53	12.00	47.1

Table 5: Results for BSTeP: number of iterations needed to attain the pointwise convergence of 10^{-4} for accelerated (upper) and unaccelerated (lower) using S_6 quadrature and linear nodal scheme.

σ_2	Δy_2	σ_1											
		0.01			0.1			1			10		
		Δy_1			Δy_1			Δy_1			Δy_1		
		0.01	0.1	1	0.01	0.1	1	0.01	0.1	1	0.01	0.1	1
0.01	0.01	4	6	6	6	8	8	8	20	12	19	74	53
		4	6	6	6	9	15	10	32	183	45	380	3000*
	0.1	5	6	6	7	9	10	13	21	28	24	45	146
		5	5	6	7	10	15	22	45	175	432	868	3000
	1	8	6	6	10	10	11	19	21	39	40	70	174
		8	7	7	13	13	15	79	84	170	2778	2313	3000
0.1	0.01	4	9	6	5	8	7	8	15	6	16	35	13
		4	6	6	5	9	15	12	36	182	48	393	3000
	0.1	6	7	7	7	9	8	11	12	9	16	21	24
		7	8	7	7	12	15	21	42	173	402	868	3000
	1	10	10	9	7	7	9	11	10	12	15	17	23
		13	12	11	15	14	17	79	84	153	2327	2287	3000
1	0.01	6	8	7	6	8	7	6	12	5	11	17	5
		6	7	7	6	10	16	8	31	179	39	443	3000
	0.1	13	14	11	9	14	9	8	10	6	11	12	7
		18	18	13	19	23	19	23	55	159	365	1251	3000
	1	15	14	13	11	10	9	7	7	6	8	8	6
		58	46	43	58	49	40	99	101	162	2477	2506	3000
10	0.01	14	16	11	12	13	9	9	13	7	10	11	6
		17	17	13	16	16	20	16	38	160	44	296	3000
	0.1	14	17	13	13	14	10	10	12	7	11	10	6
		54	50	49	60	52	43	54	50	143	362	796	3000
	1	12	13	12	10	11	9	8	9	7	29	10	7
		57	59	49	55	55	50	106	91	167	2693	2835	3000

* unconverged after 3000 iterations