NUCLEAR SCIENCE AND ENGINEERING: 125, 257-283 (1997)

A Transport Synthetic Acceleration Method for Transport Iterations

Gilles L. Ramone and Marvin L. Adams

Texas A&M University, Department of Nuclear Engineering, College Station, Texas 77843-3133

and

Paul F. Nowak

Lawrence Livermore National Laboratory, P.O. Box 808, L-18, Livermore, California 94551

Received April 15, 1996
Accepted August 19, 1996

Abstract—*A family of transport synthetic acceleration (TSA) methods for iteratively solving within-group scattering problems is presented. A single iteration in these schemes consists of a transport sweep followed by a low-order calculation, which itself is a simplified transport problem. The method for isotropic-scattering problems in X-Y geometry is described. Our Fourier analysis of a model problem for equations with no spatial discretization shows that a previously proposed TSA method is unstable in two dimensions but that our modifications make it stable and rapidly convergent. The same procedure for discretized transport equations, using the step characteristic and two bilinear discontinuous methods, shows that discretization enhances TSA performance. A conjugate gradient algorithm for the low-order problem is described, a crude quadrature set for the low-order problem is proposed, and the number of low-order iterations per high-order sweep is limited to a relatively small value. These features lead to simple and efficient improvements to the method. TSA is tested on a series of problems, and a set of parameters is proposed for which the method behaves especially well. TSA achieves a substantial reduction in computational cost over source iteration, regardless of discretization parameters or material properties, and this reduction increases with the difficulty of the problem.*

I. INTRODUCTION

Transport processes cover a wide range of physical problems that computationally require an iterative solution. In steady-state problems, discretization of the energy variable typically leads to a sequence of single-energy or one-group problems. Hence, essentially all transport problems of interest rely on the solution of steady-state one-energy-group problems.[1] We focus here on the iterative solution of such problems.

A need exists for an acceleration scheme that is easy to implement yet provides significant speedups on problems for which source iteration (SI) is unacceptably slow. After 25 yr of research and development, diffusion synthetic acceleration (DSA) has still not proved to be of practical use in some classes of transport problems.[2-8] These are typically multidimensional problems with advanced discretization schemes and/or complicated spatial grids. In such problems, it is extremely difficult to devise diffusion acceleration equations that are both effective at reducing iteration counts and easy to solve computationally. Also, implementing DSA in an existing transport code usually requires a significant coding effort. Finally, although diffusion equations in one spatial dimension require less solution effort than transport sweeps, their solution difficulty relative to transport increases with additional spatial dimensions. This further complicates the task of devising efficient DSA methods in multidimensional problems.

Our goal in this work is to develop an acceleration scheme that uses only transport equations. As is the case with DSA, a full iteration will consist of a transport sweep followed by a low-order calculation, but our low-order equation will be a transport (not diffusion)

equation. We therefore avoid "consistent differencing" issues because our low-order equation will be discretized exactly like the high-order transport equation being solved. Implementation will be simple because the low-order solution procedure will be a series of transport sweeps, the coding for which already exists. Finally, the cost of solving our low-order problems relative to a high-order transport sweep should be relatively independent of geometry, spatial discretization, and spatial grid. We show in this paper that we have been largely successful in achieving our goal: Our transport synthetic acceleration (TSA) method is easy to implement, efficient, and rapidly convergent for the class of problems we address here. We have yet to address problems with highly anisotropic scattering, opposing reflecting boundaries, or problems that are both extremely optically thick and highly scattering; thus, conclusions regarding the effectiveness of TSA for such problems must await further research.

The connection between synthetic acceleration and standard mathematical iterative techniques is often left unclear in the computational transport literature. To help make the connection, we remark that both DSA and TSA are preconditioned Richardson iteration methods. With DSA, the preconditioner is a discretized diffusion operator and with TSA, it is a discretized transport operator.

We begin with the discrete ordinates transport equation with isotropic scattering, for which the SI procedure is

$$\mathbf{\Omega}_m \cdot \nabla \psi_m^{(l+1/2)}(\mathbf{r}) + \sigma_t(\mathbf{r}) \psi_m^{(l+1/2)}(\mathbf{r}) = \frac{\sigma_s(\mathbf{r})}{4\pi} \phi^{(l)}(\mathbf{r}) + q_m(\mathbf{r}) , \quad (1a)$$

$$\phi^{(l+1/2)}(\mathbf{r}) = \sum_{m=1}^{M} w_m \psi_m^{(l+1/2)}(\mathbf{r}) , \quad (1b)$$

and

$$\phi^{(l+1)}(\mathbf{r}) = \phi^{(l+1/2)}(\mathbf{r}) . \quad (1c)$$

Physically, if $\phi^{(0)}$ is taken to be zero, $\phi^{(l)}$ represents the scalar flux because of particles that have undergone fewer than l collisions. If $\sigma_s \cong \sigma_t$, that is if the scattering ratio $c = \sigma_s/\sigma_t$ is very close to 1, and if the problem domain is large so that particles rarely leak out, we can expect the number of collisions per particle (and thus the number of iterations) to be arbitrarily high. This method is then too time consuming for practical use and can be susceptible to false convergence. (Unless convergence tests are specifically altered to prevent it, false convergence occurs when successive iterates are much closer to each other than to the converged solution, a situation that arises when an algorithm converges very slowly.)

The drawbacks of SI have motivated researchers to find acceleration techniques that could reduce the number of iterations while decreasing the computational cost. Those acceleration methods often consist of a transport sweep followed by a low-order calculation.[2-9] The transport sweep is simply the computation of the angular flux given the latest guess for the scattering source, as it is achieved in the SI method. However, the low-order calculation provides a correction to the flux, thereby producing a better scattering source for the next transport sweep.

Before we describe acceleration techniques in more detail, we develop an equation for the exact additive correction to $\psi^{(l+1/2)}$ that would give the converged solution. First, note that the converged solution satisfies

$$\mathbf{\Omega}_m \cdot \nabla \psi_m(\mathbf{r}) + \sigma_t(\mathbf{r}) \psi_m(\mathbf{r}) = \frac{\sigma_s(\mathbf{r})}{4\pi} \phi(\mathbf{r}) + q_m(\mathbf{r}) , \quad (2)$$

where the absence of iteration index denotes a converged solution. The exact additive corrections are

$$f_m^{(l+1/2)}(\mathbf{r}) = \psi_m(\mathbf{r}) - \psi_m^{(l+1/2)}(\mathbf{r}) , \quad (3a)$$

and

$$F^{(l+1/2)}(\mathbf{r}) = \sum_{m=1}^{M} w_m f_m^{(l+1/2)}(\mathbf{r}) . \quad (3b)$$

Then, if we subtract Eq. (1a) from Eq. (2), we obtain

$$\mathbf{\Omega}_m \cdot \nabla f_m^{(l+1/2)}(\mathbf{r}) + \sigma_t(\mathbf{r}) f_m^{(l+1/2)}(\mathbf{r}) = \frac{\sigma_s(\mathbf{r})}{4\pi} F^{(l+1/2)}(\mathbf{r}) + \frac{S^{(l+1/2)}(\mathbf{r})}{4\pi} , \quad (4a)$$

$$F^{(l+1/2)}(\mathbf{r}) = \sum_{m=1}^{M} w_m f_m^{(l+1/2)}(\mathbf{r}) , \quad (4b)$$

and

$$S^{(l+1/2)}(\mathbf{r}) = \sigma_s(\mathbf{r}) [\phi^{(l+1/2)}(\mathbf{r}) - \phi^{(l)}(\mathbf{r})] . \quad (4c)$$

An immediately convergent method is therefore given by one transport sweep [(Eqs. (1)] followed by Eqs. (4) for the additive correction, with

$$\phi^{(l+1)}(\mathbf{r}) = \phi^{(l+1/2)}(\mathbf{r}) + F^{(l+1/2)}(\mathbf{r}) . \quad (5)$$

However, problem (4) is just as difficult to solve as the original problem described by Eqs. (1).

Because diffusion problems are sometimes much easier to solve than transport problems, researchers have used a low-order diffusion operator in place of Eqs. (4). This technique, DSA, is defined as follows[2-8,10]:

$$\mathbf{\Omega}_m \cdot \nabla \psi_m^{(l+1/2)}(\mathbf{r}) + \sigma_t(\mathbf{r}) \psi_m^{(l+1/2)}(\mathbf{r}) = \frac{\sigma_s(\mathbf{r})}{4\pi} \phi^{(l)}(\mathbf{r}) + \frac{q(\mathbf{r})}{4\pi} , \quad (6a)$$

$$\phi^{(l+1/2)}(\mathbf{r}) = \sum_{m=1}^{M} w_m \psi_m^{(l+1/2)}(\mathbf{r}, \mathbf{\Omega}) , \quad (6b)$$

$$-\nabla\cdot\left[\frac{1}{3\sigma_t(\boldsymbol{r})}\nabla F(\boldsymbol{r})\right]+[\sigma_t(\boldsymbol{r})-\sigma_s(\boldsymbol{r})]F(\boldsymbol{r})=\sigma_s(\boldsymbol{r})[\phi^{(l+1/2)}(\boldsymbol{r})-\phi^{(l)}(\boldsymbol{r})]\ , \tag{6c}$$

and

$$\phi^{(l+1)}(\boldsymbol{r})=\phi^{(l+1/2)}(\boldsymbol{r})+F(\boldsymbol{r})\ . \tag{6d}$$

In practice, these equations must be discretized in space. This reveals the weakness of DSA: It is indeed a good accelerator if one is able to discretize the diffusion operator in a manner consistent with the discretized transport operator.[5,8] However, in many cases, the consistent discretizations are so complicated that it becomes too costly to solve the low-order problem. This is especially true on nonrectangular grids, which are becoming more common in modern codes.

Larsen and Miller followed another approach.[11] In their acceleration method, they replaced Eq. (4a) with the following easy-to-solve transport problem:

$$\boldsymbol{\Omega}_m\cdot\nabla f_m^{(l+1/2)}(\boldsymbol{r})+[\sigma_t(\boldsymbol{r})-\sigma_s(\boldsymbol{r})]f_m^{(l+1/2)}(\boldsymbol{r})=\frac{S^{(l+1/2)}(\boldsymbol{r})}{4\pi}\ . \tag{7}$$

Physically, this replacement involves ignoring scattering but keeping the absorption cross section the same. This results in a low-order problem that is solved in only one sweep. Larsen and Miller proved that this accelerator reduced the number of high-order iterations (transport sweeps) by more than a factor of 2 in most cases in slab geometry.[11] This demonstrated the efficiency of the method since each iteration is basically two transport sweeps. Yet, as we show in this paper, this method is unstable in two or three dimensions.

The rather crude approximation (ignoring all scattering) made by Larsen and Miller is responsible for the poor behavior of the Larsen-Miller method in multidimensional geometry. Therefore, we propose to keep a part of the scattering term in the low-order problem. To do this, we define a new coefficient β. With our method, the scattering cross section for the low-order problem is not σ_s (as it would be for the immediately convergent solution) nor 0 (as in the Larsen-Miller method), but is $(1-\beta)\sigma_s$, where β can take any value in the range [0,1]. To ensure that the low-order solution satisfies the same conservation equation as the exact correction, we subtract $\beta\sigma_s$ from σ_t; the resulting low-order problem is

$$\boldsymbol{\Omega}_m\cdot\nabla f_m^{(l+1/2)}(\boldsymbol{r})+[\sigma_t(\boldsymbol{r})-\beta\sigma_s(\boldsymbol{r})]f_m^{(l+1/2)}(\boldsymbol{r})=\frac{\sigma_s(\boldsymbol{r})(1-\beta)}{4\pi}F^{(l+1/2)}(\boldsymbol{r})+\frac{\sigma_s(\boldsymbol{r})}{4\pi}[\phi^{(l+1/2)}(\boldsymbol{r})-\phi^{(l)}(\boldsymbol{r})]\ . \tag{8}$$

(The conservation equation is the integral of this equation over all directions.)

We can now define a family of TSA methods for which an iteration is

$$\boldsymbol{\Omega}_m\cdot\nabla\psi_m^{(l+1/2)}(\boldsymbol{r})+\sigma_t(\boldsymbol{r})\psi_m^{(l+1/2)}(\boldsymbol{r})=\frac{\sigma_s(\boldsymbol{r})}{4\pi}\phi^{(l)}(\boldsymbol{r})+\frac{q(\boldsymbol{r})}{4\pi}\ , \tag{9a}$$

$$\phi^{(l+1/2)}(\boldsymbol{r})=\sum_{m=1}^{M}w_m\psi_m^{(l+1/2)}(\boldsymbol{r})\ , \tag{9b}$$

$$\boldsymbol{\Omega}_m\cdot\nabla f_m^{(l+1/2)}(\boldsymbol{r})+[\sigma_t(\boldsymbol{r})-\beta\sigma_s(\boldsymbol{r})]f_m^{(l+1/2)}(\boldsymbol{r})=\frac{\sigma_s(\boldsymbol{r})(1-\beta)}{4\pi}F^{(l+1/2)}(\boldsymbol{r})+\frac{\sigma_s(\boldsymbol{r})}{4\pi}[\phi^{(l+1/2)}(\boldsymbol{r})-\phi^{(l)}(\boldsymbol{r})]\ , \tag{9c}$$

$$F^{(l+1/2)}(\boldsymbol{r})=\sum_{m=1}^{M}w_m f_m^{(l+1/2)}(\boldsymbol{r})\ , \tag{9d}$$

and

$$\phi^{(l+1)}(\boldsymbol{r})=\phi^{(l+1/2)}(\boldsymbol{r})+F^{(l+1/2)}(\boldsymbol{r})\ . \tag{9e}$$

As we will show, for a wide range of β's, this scheme does decrease the number of high-order iterations. However, it is efficient only if the reduction in high-order iterations overcomes the computational cost of the low-order problem. The superscript $(l+\frac{1}{2})$ in Eqs. (9c), (9d), and (9e) denotes a converged solution of the corrections for the l'th transport sweep. If we use SI to solve this low-order problem, the scattering ratio will be

$$\tilde{c}=\frac{(1-\beta)c}{(1-\beta c)}\ , \tag{10}$$

where

$$c=\sigma_s/\sigma_t\ .$$

Since $\tilde{c}$ is always less than c (if $c<1.0$), the low-order problem will generally take fewer iterations to converge with SI than would the high-order one with SI. This tendency is further enhanced by an increased leakage probability, because the total cross section is now $\sigma_t-\beta\sigma_s$ in Eq. (9c) instead of σ_t for Eq. (9a).

As we shall see, these properties do not reduce computational cost enough to make the method efficient. To address this, we have devised further improvements. First, we can use a conjugate gradient (CG) method on the low-order problem [Eq. (9c)] (Ref. 12). The obvious question is: Why not use CG on the original problem? This might be possible if the high-order problem always had isotropic scattering, but we did not wish to impose such a restriction. However, we are willing to impose this restriction on the low-order problem. The angular discretization also requires some attention: Numerous directions are often needed for an accurate solution ψ, but there is no such requirement on the angular corrections. Hence, it may be possible to use a

cruder quadrature set for the low-order problem and thereby reduce the computational cost of the accelerator. Finally, we can set the number of low-order iterations per high-order iteration, N_{it}, to a finite value. The convergence of the low-order problem will, in that case, not be as tight as the convergence of the high-order problem. However, we find that even though more high-order iterations are necessary, the overall computational cost decreases.

The remainder of this paper is organized as follows. In Sec. II, we perform a Fourier analysis of the analytic transport synthetic accelerated discrete ordinate equations [Eqs. (9)] and describe the behavior of the spectral radius for different quadrature sets, various scattering ratios c, and different values of the coefficient β. In Sec. III, we discretize those equations in X-Y geometry, first with the step characteristic (SC) method, then with a family of bilinear discontinuous (BLD) finite element methods. We add the mesh spacings to the previous variables and perform a Fourier analysis of the discrete system. In Sec. IV, we discuss the efficiency of this accelerator and develop the improvements described previously. In Sec. V, we apply our accelerator to a series of problems in X-Y geometry. We quantify the effects of the CG method, the low-order quadrature set, and the iteration limit N_{it} by comparing them to TSA alone and to SI. Section VI is devoted to a summary and some concluding remarks.

II. CONTINUOUS TRANSPORT SYNTHETIC ACCELERATION

At this point, we have defined an iterative method but we do not know whether it is efficient or even convergent. One theoretical way to address this issue is to perform a Fourier analysis.[5] We define a simple model problem: an infinite medium with constant cross sections. (This seems restrictive, but previous experience demonstrates that if an iterative method performs well on this problem, it performs well on finite, heterogeneous problems.[5–7]) Thus, we define iteration errors as

$$\hat{\psi}_m^{(l+1/2)} = \psi_m^{(l+1/2)} - \psi_m \ , \tag{11a}$$

$$\hat{\phi}^{(l+1/2)} = \phi^{(l+1/2)} - \phi \ , \tag{11b}$$

and

$$\hat{\phi}^{(l)} = \phi^{(l)} - \phi \ . \tag{11c}$$

With these definitions, the equations for iteration errors are simply the original equations with no fixed source. If we focus on X-Y geometry, we obtain

$$\left(\mu_m \frac{\partial}{\partial x} + \eta_m \frac{\partial}{\partial y} + 1\right)\hat{\psi}_m^{(l+1/2)}(x,y) = \frac{c}{2\pi}\hat{\phi}^{(l)}(x,y) \ , \tag{12a}$$

$$\hat{\phi}^{(l+1/2)}(x,y) = \sum_{m=1}^{M} w_m \hat{\psi}_m^{(l+1/2)}(x,y) \ , \tag{12b}$$

$$\left[\mu_m \frac{\partial}{\partial x} + \eta_m \frac{\partial}{\partial y} + (1-\beta c)\right] f_m^{(l+1/2)}(x,y) = \frac{(1-\beta)c}{2\pi} F^{(l+1/2)}(x,y) + \frac{c}{2\pi}\left[\hat{\phi}^{(l+1/2)}(x,y) - \hat{\phi}^{(l)}(x,y)\right] \ , \tag{12c}$$

$$F^{(l+1/2)}(x,y) = \sum_{m=1}^{M} w_m f_m^{(l+1/2)}(x,y) \ , \tag{12d}$$

and

$$\hat{\phi}^{(l+1)}(x,y) = \hat{\phi}^{(l+1/2)}(x,y) + F^{(l+1/2)}(x,y) \ , \tag{12e}$$

where x and y are in units of mean free paths (mfp), $c = \sigma_s/\sigma$, and the quadrature weights sum to 2π.

For this model problem, we can represent the iteration error as a linear combination of plane waves, or modes. Those modes are the eigenfunctions of the iteration operator. Hence, each mode behaves independently from the others and a single iteration multiplies the mode's amplitude by a number we shall call ω, the corresponding eigenvalue of the operator. The worst behavior is therefore determined by the largest $|\omega|$, the spectral radius for the method. If the spectral radius is larger than unity, the method is divergent. If it is less than unity, the method converges, and the spectral radius determines the speed of convergence. We can find ω by inserting the following into the iteration scheme:

$$\hat{\psi}_m^{(l+1/2)}(x,y) = \omega^l a_m e^{i(\lambda_x x + \lambda_y y)} \ , \quad i \equiv \sqrt{-1} \ , \tag{13a}$$

$$\hat{\phi}^{(l+1/2)}(x,y) = \omega^l A e^{i(\lambda_x x + \lambda_y y)} \ , \tag{13b}$$

and

$$\hat{\phi}^{(l)}(x,y) = \omega^l B e^{i(\lambda_x x + \lambda_y y)} \ . \tag{13c}$$

Upon substituting Eqs. (13) into Eqs. (12), we obtain after considerable algebra the following expression for ω:

$$\omega = c\left[D_0 + \frac{(cD_0 - 1)D_1}{1 - (1-\beta)cD_1}\right] \ , \tag{14a}$$

where

$$D_0 = \frac{\sum_{m=1}^{M}\left[\frac{w_m}{1 + (\lambda_x\mu_m + \lambda_y\eta_m)^2}\right]}{\sum_{m=1}^{M} w_m} \tag{14b}$$

and

$$D_1 = \frac{\sum_{m=1}^{Mlo}\left[\frac{w_m(1-\beta c)}{(1-\beta c)^2 + (\lambda_x\mu_m + \lambda_y\eta_m)^2}\right]}{\sum_{m=1}^{Mlo} w_m} \ . \tag{14c}$$

Thus, ω depends on many parameters: the scattering ratio c, the two quadrature sets S_N and S_{Nlo}, the

wave numbers λ_x and λ_y, and the coefficient β. To analyze the behavior of ω, we can choose quadrature sets (the same for the high- and low-order problems unless otherwise noted) and values for β and c. Then, we plot ω as a function of the wave numbers λ_x and λ_y as shown in Fig. 1. For that particular case, we used an S_4 level symmetric quadrature set and chose $c = 0.9$ and $\beta = 0.8$.

Figure 1 shows that the problem is symmetric in λ_x and λ_y, as we expected from Eqs. (14). It appears that $\omega = 0$ for the flat mode ($\lambda_x = \lambda_y = 0$). The negative eigenvalues reach their largest magnitude for the high frequencies (large λ_x and λ_y) on the "diagonal modes" ($\lambda_x = \lambda_y$). Eigenvalues also become negative on secondary directions, corresponding approximately to $\lambda_x = \lambda_y/2$ (and $\lambda_y = \lambda_x/2$), but their magnitude is not as large as on the diagonal modes. Also, the largest positive eigenvalue is located on the one-dimensional modes ($\lambda_x = 0$, $0 \le \lambda_y \le \infty$, or vice versa). In this case, this is also the spectral radius.

Upon analyzing many graphs of this type for different quadrature sets, scattering ratios, and β's, we have found the following:

1. For every quadrature set tested, the largest $|\omega|$ occurs when $c = 1.0$ and $\lambda_x = \lambda_y$.

2. For $c > 0.95$, the Larsen-Miller method ($\beta = 1.0$) is unstable for all quadrature sets tested (S_2 to S_{16} level symmetric). The worst error results from high-frequency modes (large λ_x and λ_y), but many modes are unstable. However, we believe that for an infinite quadrature order, the Larsen-Miller method would be stable because each mode would behave as in slab geometry.

3. If $c = 1.0$, $\omega = \beta$ for the flat mode ($\lambda_x = \lambda_y = 0$).

4. If $c < 1.0$, $\omega = 0$ for the flat mode. This result is independent of the quadrature set.

5. The most positive value of ω, ρ^+, is always bounded below the product βc. For $c = 1.0$, it occurs when $\lambda_x = \lambda_y = 0$. For $c < 1.0$, it occurs on the one-dimensional modes, i.e., when one of the two wave numbers is 0. For β and c close to 1, the value of λ for which the peak occurs is close to 0, but as β and c decrease, this value increases.

6. The most negative value of ω, ρ^-, occurs on the diagonal mode ($\lambda_x = \lambda_y$) when λ_x and λ_y tend to infinity. However, when the quadrature order increases, $|\rho^-|$ decreases.

7. For all c and for $\beta < 1.0$, the spectral radius decreases and rapidly comes to an asymptotic value as the quadrature order increases. This is true at least for level-symmetric quadrature sets.

8. For a given quadrature order N, there are $N/2$ directions for which ω reaches a relative minimum but the absolute minimum is always located on the diagonal modes.

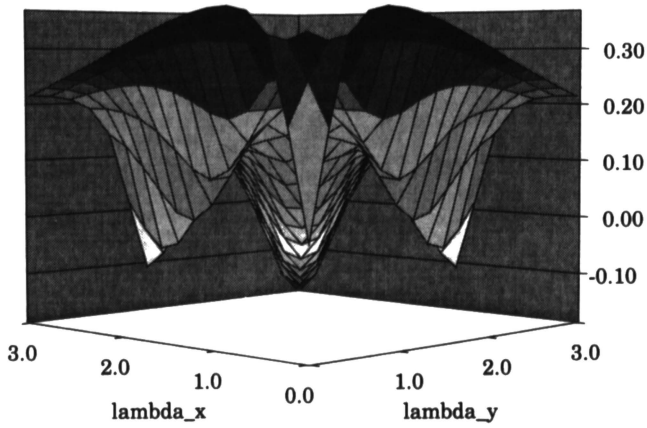

Fig. 1. Iteration eigenvalue as a function of λ_x and λ_y, for an S_4 level-symmetric quadrature set, $c = 0.9$, $\beta = 0.8$.

9. For a given high-order quadrature, the magnitudes of both the most negative and the most positive eigenvalues decrease as the low quadrature order increases. This is all the more relevant as c decreases.

10. For a sufficiently small β, $|\rho^-| < \rho^+ \le \beta c$ and the method is stable. "Sufficiently small" depends on the quadrature order and c. In Table I, we focus on $c = 1.0$ (the worst case). We give the maximum value of β for which $|\rho^-| < \rho^+ \le \beta c$ as a function of the high and low quadrature order. (A value of zero in Table I means $|\rho^-| > \rho^+$ unless $\beta = 0$.)

We now try to illustrate the most important trends of the TSA methods. In Fig. 2, we present ρ^+ and ρ^- as a function of β, given $c = 0.9$ and an S_4 level-symmetric quadrature set for both the high- and low-order problems. We see that the Larsen-Miller method ($\beta = 1$) is unstable ($\rho^- = -1.12$). We also can see that both ρ^+ and $|\rho^-|$ are increasing functions of β. When

TABLE I

Maximum β Such That $|\rho^-| < \rho^+ \le \beta c$

	S_n					
S_{nlo}	S_2	S_4	S_6	S_8	S_{12}	S_{16}
S_2	0.51	0.0	0.0	0.0	0.0	0.0
S_4		0.84	0.84	0.81	0.79	0.78
S_6			0.84	0.81	0.79	0.78
S_8				0.90	0.89	0.89
S_{12}					0.93	0.93
S_{16}						0.94

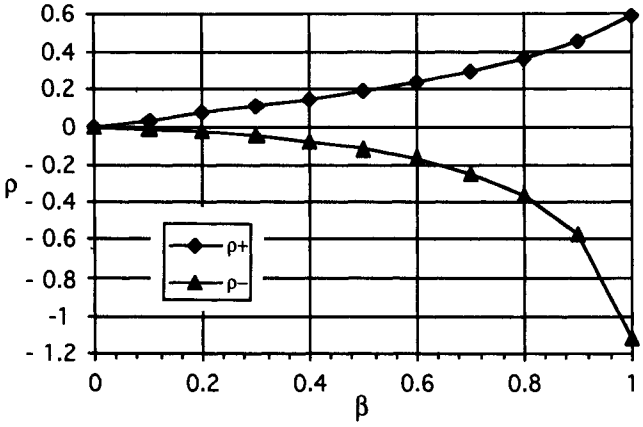

Fig. 2. Most negative and most positive eigenvalue for an S_4 level-symmetric quadrature set, $c = 0.9$, various β's.

β is sufficiently small (in this case $\beta = 0.8$), ρ^+ is also the spectral radius, and it increases almost linearly with β. If we used a lower scattering ratio for a given β, ρ^+ and ρ^- would both have a smaller magnitude than the values reported in Fig. 2. This is also true for finer quadrature sets.

In Fig. 3, we present ω as a function of λ for different values of c, given an S_4 level-symmetric quadrature set for both the high- and the low-order problems, the one-dimensional error modes, and $\beta = 0.9$. For this case, the spectral radius is also the most positive eigenvalue regardless of the value of c. We see that when $c = 1.0$, $\omega = \beta$ for the flat mode, whereas for all other c's, $\omega = 0$. It also appears that the spectral radius decreases much faster than linearly when c decreases. This trend holds for any β or quadrature set.

Figure 4 illustrates the behavior of ω as a function of λ for different level-symmetric quadrature sets (identical for the high- and low-order problems), $c = 0.999$, the diagonal error modes, and $\beta = 0.8$. We can see from the graph that the most negative eigenvalues occur for the "high frequency" modes (large λ's) and that their magnitude decreases as the quadrature order increases. This is especially true if we use an S_4 instead of an S_2 quadrature set, the decrease being far less spectacular for higher orders.

To make the low-order problem as inexpensive as possible, it might be best to have different quadrature sets for the high- and low-order problems. In Table II, we present ρ^+ and ρ^- for different level-symmetric low-order quadrature sets, an S_{16} level-symmetric high-order quadrature set, $c = 0.99$, and $\beta = 0.8$. We notice the same trends as in the previous graph concerning the negative modes, although this time, the high-order quadrature is fixed. This emphasizes the dominant role played by the low-order quadrature on the behavior of negative modes. Both ρ^+ and ρ^- are decreasing functions of the low quadrature order, but whereas ρ^- keeps on getting closer to zero as N_{lo} increases, ρ^+ very rapidly approaches an asymptotic value (in this case 0.630). Thus, the low-order quadrature set also has an effect on the spectral radius, and it does not need to be fine to yield a nearly optimal performance. In the case shown in Table II, an S_4 quadrature set might be sufficient. (Note that ρ^- really is the same for S_4 and S_6; this is not a typing error.)

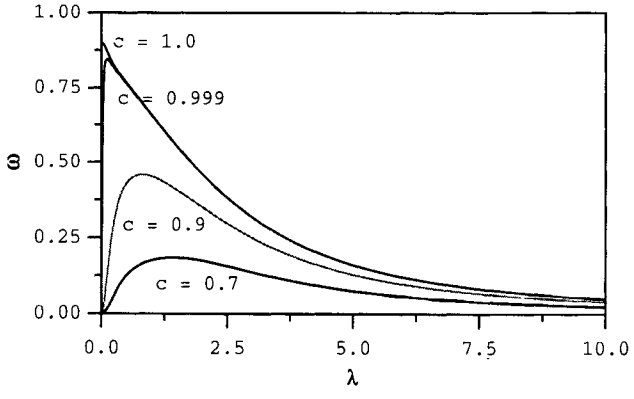

Fig. 3. Iteration eigenvalue as a function of $\lambda = \lambda_x$, $\lambda_y = 0$, for an S_4 level-symmetric quadrature set, $\beta = 0.9$, various c's.

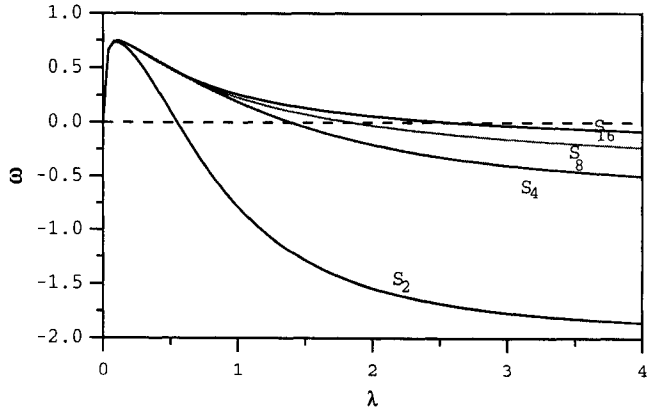

Fig. 4. Iteration eigenvalue as a function of $\lambda = \lambda_x = \lambda_y$, for various level-symmetric quadrature sets, $c = 0.999$, $\beta = 0.8$.

TABLE II

ρ^+ and ρ^- for Various Low Level-Symmetric Quadrature Order, an S_{16} Level-Symmetric High Quadrature Set, $c = 0.99$, $\beta = 0.8$

	S_2	S_4	S_6	S_8	S_{12}	S_{16}
ρ^+	0.725	0.638	0.632	0.631	0.630	0.630
ρ^-	−4.186	−0.818	−0.818	−0.465	−0.287	−0.251

TABLE III

ρ^+ and ρ^- for Various Low Level-Symmetric Quadrature Order, an S_{16} Level-Symmetric High Quadrature Set, $c = 0.99$, $\beta = 0.2$

	S_2	S_4	S_6	S_8	S_{12}	S_{16}
ρ^+	0.282	0.147	0.147	0.147	0.147	0.147
ρ^-	−1.07	−0.167	−0.167	−0.074	−0.028	−0.016

In addition, Table III shows the effect of a smaller β ($\beta = 0.2$) on the spectral radius, given the same scattering ratio ($c = 0.99$) and quadrature sets. As we expected, ρ^+ and ρ^- have substantially smaller magnitudes than in Table II, and an S_4 low-order quadrature set would lead to convergence in a few iterations, the spectral radius being only 0.167.

In this section, we have summarized the behavior of our TSA iteration scheme without spatial discretization. First, we defined the parameters that the TSA eigenvalues depend upon: the wave numbers λ_x and λ_y, the scattering ratio c, the high and low quadrature sets, and the parameter β. We found that the Larsen-Miller TSA method was unstable in two dimensions, at least for the discrete ordinates transport equation. We noted, however, that it is only one member in a family of TSA methods, among which many are stable. We introduced the notion of compromise that should exist between the choice of the spectral radius and the corresponding computational cost. We analyzed a family of TSA methods for simple model problems (infinite medium, constant cross sections, no spatial discretization), and we presented results illustrating a variety of interesting trends. In Sec. III, we theoretically examine the effects of spatial discretization on the performances and trends of TSA.

III. DISCRETIZED TRANSPORT SYNTHETIC ACCELERATION

One of the most attractive features of TSA is that the spatial discretization of the low-order problem is straightforward: because the high- and low-order problems are both transport problems, their discretization is identical. This is a clear advantage over DSA, for which much care is needed to assure the consistency of the discretized transport and diffusion equations.

In the first part of this section, we analyze the SC spatial discretization[13] because of its similarity to the long characteristic method, a more complicated discretization technique used in many reactor-analysis codes. In addition, we develop an analysis for two BLD methods[14] that are often used for transport calculations. Because there is no question of consistency concerning the spatial discretization for TSA, we expect the performances to be similar for most spatial discretization methods.

III.A. Step Characteristics Spatial Discretization

III.A.1. Fourier Analysis

Step characteristics is a spatial discretization method that requires the knowledge of the average angular fluxes on incident edges to compute the angular fluxes on exiting edges. The SC equations are

$$\frac{\mu_m}{\Delta x_i}\left[\psi^{(l+1/2)}_{m,i+1/2,J} - \psi^{(l+1/2)}_{m,i-1/2,J}\right] + \frac{\eta_m}{\Delta y_J}\left[\psi^{(l+1/2)}_{m,i,J+1/2} - \psi^{(l+1/2)}_{m,i,J-1/2}\right] + \sigma_{t_{i,J}}\psi^{(l+1/2)}_{m,i,J} = S^{(l)}_{m,i,J} , \tag{15a}$$

$$S^{(l)}_{m,i,j} = \frac{\sigma_{s_{i,J}}}{2\pi}\phi^{(l)}_{i,j} + q_{m,i,J} , \tag{15b}$$

$$\psi^{(l+1/2)}_{m,i\pm1/2,J} = \frac{S^{(l)}_{m,i,J}}{\sigma_{t_{i,J}}} + \gamma_{m_{i,J}}\left[\psi^{(l+1/2)}_{m,i,j\mp1/2} - \frac{S^{(l)}_{m,i,J}}{\sigma_{t_{i,J}}}\right]\times\left(\frac{1-e^{-\tau_{m_{i,J}}}}{\tau_{m_{i,J}}}\right) + \left[\psi^{(l+1/2)}_{m,i\mp1/2,j} - \frac{S^{(l)}_{m,i,J}}{\sigma_{t_{i,J}}}\right]\times(1-\gamma_{m_{i,J}})e^{-\tau_{m_{i,J}}} , \tag{15c}$$

and

$$\psi^{(l+1/2)}_{m,i,j\pm1/2} = \frac{S^{(l)}_{m,i,j}}{\sigma_{t_{i,J}}} + \alpha_{m_{i,J}}\left[\psi^{(l+1/2)}_{m,i\mp1/2,J} - \frac{S^{(l)}_{m,i,j}}{\sigma_{t_{i,J}}}\right]\times\left(\frac{1-e^{-\tau_{m_{i,J}}}}{\tau_{m_{i,J}}}\right) + \left[\psi^{(l+1/2)}_{m,i,j\mp1/2} - \frac{S^{(l)}_{m,i,J}}{\sigma_{t_{i,J}}}\right]\times(1-\alpha_{m_{i,J}})e^{-\tau_{m_{i,J}}} , \tag{15d}$$

where

$$\tau_{m_{i,J}} = \sigma_{t_{i,J}}\min\left(\frac{\Delta x_i}{|\mu_m|}, \frac{\Delta y_J}{|\eta_m|}\right) , \tag{16a}$$

$$\alpha_{m_{i,J}} = \min\left(1, \left|\frac{\mu_m \Delta y_J}{\eta_m \Delta x_i}\right|\right) , \tag{16b}$$

$$\gamma_{m_{i,J}} = \min\left(1, \left|\frac{\eta_m \Delta x_i}{\mu_m \Delta y_J}\right|\right) , \tag{16c}$$

$$i \pm \tfrac{1}{2} = i + \tfrac{1}{2} \quad \text{for } \mu_m > 0 \text{ and } i - \tfrac{1}{2} \text{ for } \mu_m < 0 , \tag{16d}$$

$$j \pm \tfrac{1}{2} = j + \tfrac{1}{2} \quad \text{for } \eta_m > 0 \text{ and } j - \tfrac{1}{2} \text{ for } \eta_m < 0 , \tag{16e}$$

and Δx_i and Δy_j are the dimensions of the cell (i,j). The subscripts $i+\frac{1}{2}$, $i-\frac{1}{2}$, $j+\frac{1}{2}$, and $j-\frac{1}{2}$ refer respectively to the right, left, top, and bottom of the cell (i,j). Hence, for example, $\psi_{m,i+1/2,j}$ is the angular flux at the right face of cell (i,j) in the direction $\mathbf{\Omega}_m$. Our quadrature weights sum to 2π.

We now perform a Fourier analysis of the discretized model problem. First, we write the ansatz for the angular and scalar flux iteration errors of interest:

$$\hat{\psi}^{(l+1/2)}_{m,i,j} = \omega^l a_m e^{k(\lambda_x x_i + \lambda_y y_j)} , \quad k \equiv \sqrt{-1} , \tag{17a}$$

$$\hat{\psi}^{(l+1/2)}_{m,i+1/2,j} = \omega^l b_m e^{k(\lambda_x x_{i+1/2} + \lambda_y y_j)} , \tag{17b}$$

$$\hat{\psi}^{(l+1/2)}_{m,i,j+1/2} = \omega^l d_m e^{k(\lambda_x x_i + \lambda_y y_{j+1/2})} , \tag{17c}$$

$$\hat{\phi}^{(l+1/2)}_{i,j} = \omega^l A e^{k(\lambda_x x_i + \lambda_y y_j)} , \tag{17d}$$

and

$$\hat{\phi}^{(l)}_{i,j} = \omega^l B e^{k(\lambda_x x_i + \lambda_y y_j)} . \tag{17e}$$

After considerable algebra, which is unique to the SC method, we obtain the following expression for the ratio A/B:

$$\frac{A}{B} = c\left\{1 - \frac{2}{\pi\chi}\left[\begin{array}{l}\sin\left(\frac{\theta_x}{2}\right)\sum\limits_{\mu_m,\eta_m>0} w_m \frac{\mu_m}{\Delta x}\left(\frac{Nb_I}{D_I} + \frac{Nb_{II}}{D_{II}}\right) \\ + \sin\left(\frac{\theta_y}{2}\right)\sum\limits_{\mu_m,\eta_m>0} w_m \frac{\eta_m}{\Delta y}\left(\frac{Nd_I}{D_I} + \frac{Nd_{IV}}{D_{IV}}\right)\end{array}\right]\right\}, \tag{18}$$

where $\chi = \sigma_t/\sigma_t = 1.0$ and $\theta_x = \lambda_x \Delta x$, $\theta_y = \lambda_y \Delta y$. Expressions for Nb_I, Nb_{II}, Nd_I, Nd_{IV}, D_I, D_{II}, and D_{IV} as well as a complete derivation of A/B, are given in Appendix A.

If we were only considering SI, the ratio A/B would be the eigenvalue ω, but with TSA it is only an intermediate result. We form the discretized transport equation for the corrections:

$$\frac{\mu_m}{\Delta x(1-\beta c)}\left[f^{(l+1/2)}_{m,i+1/2,j} - f^{(l+1/2)}_{m,i-1/2,j}\right] + \frac{\eta_m}{\Delta y(1-\beta c)}\left[f^{(l+1/2)}_{m,i,j+1/2} - f^{(l+1/2)}_{m,i,j-1/2}\right] + f^{(l+1/2)}_{m,i,j}$$
$$= \frac{c}{2\pi}\frac{\left[(1-\beta)F^{(l+1/2)}_{i,j} + \phi^{(l+1/2)}_{i,j} - \phi^{(l)}_{i,j}\right]}{1-\beta c} . \tag{19}$$

Following similar steps as previously described, we define

$$f^{(l+1/2)}_{m,i,j} = \omega^l \tilde{a}_m e^{i(\lambda_x x_i + \lambda_y y_j)} , \tag{20a}$$

$$f^{(l+1/2)}_{m,i+1/2,j} = \omega^l \tilde{b}_m e^{i(\lambda_x x_{i+1/2} + \lambda_y y_j)} , \tag{20b}$$

$$f^{(l+1/2)}_{m,i,j+1/2} = \omega^l \tilde{d}_m e^{i(\lambda_x x_i + \lambda_y y_{j+1/2})} , \tag{20c}$$

$$F^{(l+1/2)}_{i,j} = \omega^l \tilde{A} e^{i(\lambda_x x_i + \lambda_y y_j)} , \tag{20d}$$

and

$$\frac{(1-\beta)F^{(l+1/2)}_{i,j} + \left[\phi^{(l+1/2)}_{i,j} - \phi^{(l)}_{i,j}\right]}{(1-\beta c)} = \omega^l \tilde{B} e^{i(\lambda_x x_i + \lambda_y y_j)} . \tag{20e}$$

We obtain the ratio $\tilde{A}/\tilde{B}$:

$$\frac{\tilde{A}}{\tilde{B}} = c\left\{1 - \frac{2}{\pi\tilde{\chi}}\left[\begin{array}{l}\sin\left(\frac{\theta_x}{2}\right)\sum\limits_{\mu_m,\eta_m>0} w_m \frac{\mu_m}{\Delta x}\left(\frac{Nb_I}{D_I} + \frac{Nb_{II}}{D_{II}}\right) \\ + \sin\left(\frac{\theta_y}{2}\right)\sum\limits_{\mu_m,\eta_m>0} w_m \frac{\eta_m}{\Delta y}\left(\frac{Nd_I}{D_I} + \frac{Nd_{II}}{D_{II}}\right)\end{array}\right]\right\}, \tag{21}$$

with

$$\tilde{\chi} = \tilde{\sigma}_t/\sigma_t = 1.0 - \beta c . \tag{22}$$

Finally, substituting Eqs. (17) and (20) into Eq. (9e) yields

$$\omega = \frac{A}{B} + \frac{\frac{\tilde{A}}{\tilde{B}}\left(\frac{A}{B} - 1\right)}{1 - \beta c - \frac{\tilde{A}}{\tilde{B}}(1-\beta)} . \tag{23}$$

Hence, we need to create a subroutine that computes the ratio A/B. This can also compute $\tilde{A}/\tilde{B}$, and we can determine the TSA spectral radius for an SC discretization of our model problem.

III.A.2. Results

Most of the properties described in Sec. III.A.1 are still valid when the problem is discretized. Therefore, we focus mainly on the new parameters that replaced

the wave numbers: Δx, Δy, θ_x, and θ_y. Discretization introduces a periodicity in the behavior of the eigenvalues so that we need only consider the domain $\theta_x \in [0, \pi]$, $\theta_y \in [0, \pi]$. One of those periods is presented in Fig. 5 for $\beta = 0.8$, $c = 0.99$, an S_4 level-symmetric quadrature set, and a spatial mesh whose cells are each 100×100 mfp.

If we compare Fig. 5 to Fig. 1, we see that the spectral radius is located on the diagonal modes (instead of the one-dimensional modes). We also witness a dramatic decrease of the spectral radius from 0.38 to 0.0047. Importantly, the negative eigenvalues have vanished.

We have analyzed many of those graphs and reached the following conclusions:

1. Most of the properties concerning β, c, and the high- and low-order quadrature sets still hold when the scheme is discretized:

   a. The spectral radius is an increasing function of β.

   b. The spectral radius is an increasing function of c.

   c. If $c < 1.0$, $\omega = 0$ for the flat mode ($\theta_x = \theta_y = 0$).

   d. For all c and for $\beta < 1.0$, the spectral radius is a decreasing function of the high-order quadrature set but rapidly comes to an asymptotic value.

   e. For a given high-order quadrature set, the spectral radius decreases as the low quadrature order increases. This behavior is enhanced as c decreases.

2. When $c = 1.0$, the most positive value of ω, ρ^+, is no longer β except in the fine-mesh limit. Also, ρ^+ occurs at $\theta_x = \theta_y = 0.0$ until Δx and Δy are both >6.0 mfp. Then, it is reached for $\theta_x/\Delta x = \theta_y/\Delta y = \pi$.

3. The worst case for ρ^+ is still $c = 1.0$, but ρ^+ is far less than β when the cells become optically thick. For S_4/S_4 level-symmetric quadrature sets and $\beta = 0.8$, ρ^+ is 0.799 for $\Delta x = \Delta y = 0.01$ mfp but only 0.227 when $\Delta x = \Delta y = 10.0$ mfp.

4. In the fine-mesh limit ($\Delta x \to 0$, $\Delta y \to 0$), we observe the same behavior as without spatial discretization.

5. The spectral radius decreases as cell size increases.

6. Negative eigenvalues vanish as the mesh coarsens.

7. The spectral radius is located on the diagonal modes or on the one-dimensional modes. The diagonal modes correspond to $\theta_x/\Delta x = \theta_y/\Delta y$ and the one-dimensional modes to $\theta_x = 0$ or $\theta_y = 0$.

8. The Larsen-Miller method is stable for a greater range when the equations are discretized. Table IV shows the values we found as a function of square cells size for S_{16} level-symmetric quadrature sets.

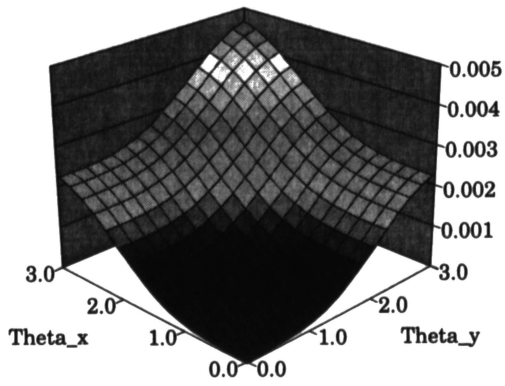

Fig. 5. Iteration eigenvalues as a function of θ_x and θ_y for the SC method, $\beta = 0.8$, $c = 0.99$, $\Delta x = \Delta y = 100$ mfp, S_4/S_4 level-symmetric quadrature sets.

TABLE IV

Maximum Scattering Ratio for Stability of the Larsen-Miller Method, S_{16}/S_{16} Level-Symmetric Quadrature Sets

$\Delta x = \Delta y$ (mfp)	0.01	0.1	1.0	10.0	100.0
c	0.95	0.96	0.97	0.997	0.9997

In Table V, we present the spectral radius as a function of the mesh size for an S_2 level-symmetric quadrature set, $\beta = 0.9$ and $c = 0.999$. We can see that for fine meshes (Δx and $\Delta y < 1$ mfp) the method is unstable with this β and quadrature set, but that the method

TABLE V

Spectral Radius Versus Mesh Size for the SC Method, $\beta = 0.9$, $c = 0.999$, S_2 Level-Symmetric Quadrature Set

	Δy							
Δx	0.01	0.1	0.3	1.0	3.0	10.0	30.0	100.0
0.01	4.358							
0.1	1.328	3.969						
0.3	0.881	0.879	3.282					
1.0	0.881	0.880	0.877	1.907				
3.0	0.881	0.880	0.877	0.851	0.709			
10.0	0.881	0.880	0.877	0.854	0.717	0.456		
30.0	0.881	0.880	0.878	0.855	0.719	0.557	0.264	
100.0	0.881	0.880	0.878	0.855	0.720	0.591	0.308	0.092

rapidly becomes stable as either Δx or Δy becomes large.

In Table VI, we focus on the effect of a smaller β ($\beta = 0.8$ instead of 0.9) and a finer quadrature set (S_4 instead of S_2). Those changes make the method unconditionally stable. As in Table V, the spectral radius decreases monotonically as cell size increases.

In Table VII, we emphasize the dramatic effect of a small amount of absorption: We again use $\beta = 0.8$ and an S_4 level-symmetric quadrature set but now $c = 0.9$ instead of 0.999 in Table VI. The spectral radius drops significantly to 0.368 in the worst case (fine-mesh limit).

Finally, in Table VIII, we show that even if we use an S_2 quadrature set (the most troublesome), we only need to take a lower β (0.5 instead of 0.9 in Table IV) to reach unconditional stability. However, such a low β makes the low-order problem more difficult to solve, so the TSA method may be less effective for S_2 problems than for other S_N problems. (This is mainly of academic interest because S_2 is almost never used for practical calculations.) Table VIII also illustrates the rapid decrease in spectral radius as mesh cells become optically thicker.

TABLE VI

Spectral Radius Versus Mesh Size for the SC Method, $\beta = 0.8$, $c = 0.999$, S_4 Level-Symmetric Quadrature Set

	Δy							
Δx	0.01	0.1	0.3	1.0	3.0	10.0	30.0	100.0
0.01	0.746							
0.1	0.746	0.743						
0.3	0.746	0.745	0.734					
1.0	0.746	0.739	0.739	0.688				
3.0	0.746	0.746	0.741	0.695	0.508			
10.0	0.746	0.746	0.741	0.699	0.498	0.275		
30.0	0.746	0.746	0.741	0.699	0.496	0.275	0.138	
100.0	0.746	0.746	0.741	0.700	0.495	0.293	0.131	0.043

TABLE VII

Spectral Radius Versus Mesh Size for the SC Method, $\beta = 0.8$, $c = 0.9$, S_4 Level-Symmetric Quadrature Set

	Δy							
Δx	0.01	0.1	0.3	1.0	3.0	10.0	30.0	100.0
0.01	0.368							
0.1	0.369	0.362						
0.3	0.369	0.369	0.348					
1.0	0.369	0.368	0.358	0.292				
3.0	0.369	0.368	0.361	0.303	0.168			
10.0	0.369	0.368	0.362	0.307	0.209	0.142		
30.0	0.369	0.368	0.362	0.308	0.238	0.115	0.037	
100.0	0.369	0.368	0.362	0.309	0.248	0.121	0.024	0.005

III.B. Bilinear Discontinuous Methods

III.B.1. Fourier Analysis

Bilinear discontinuous methods utilize four basis functions to approximate the angular flux in a cell:

$$b_1(x,y) = 1 \ , \tag{24a}$$

$$b_2(x,y) = \frac{2(x-x_i)}{\Delta x_i} \ , \tag{24b}$$

$$b_3(x,y) = \frac{2(y-y_i)}{\Delta y_i} \ , \tag{24c}$$

and

$$b_4(x,y) = b_2(x,y)b_3(x,y) \ . \tag{24d}$$

The BLD approximation is

$$\psi(x,y) = \psi_{ij} b_1(x,y) + \psi_{ij}^x b_2(x,y) + \psi_{ij}^y b_3(x,y) + \psi_{ij}^{xy} b_4(x,y) \ , \quad x,y \in \text{cell } ij \ . \tag{25}$$

If we expand the angular flux in bilinear basis functions, then multiply the transport equation by each basis function, and integrate it over cell (i,j), we find

$$\frac{\mu_m}{\Delta x_i}\left[\psi_{m,i+1/2,j}^{(l+1/2)} - \psi_{m,i-1/2,j}^{(l+1/2)}\right] + \frac{\eta_m}{\Delta y_j}\left[\psi_{m,i,j+1/2}^{(l+1/2)} - \psi_{m,i,j-1/2}^{(l+1/2)}\right] + \sigma_{i,j}\psi_{m,i,j}^{(l+1/2)} = \frac{c}{2\pi}\phi_{i,j}^{(l)} \ , \tag{26a}$$

$$\frac{\kappa_{m,i,j}^x \mu_m}{\Delta x_i}\left[\psi_{m,i+1/2,j}^{(l+1/2)} + \psi_{m,i-1/2,j}^{(l+1/2)} - 2\psi_{m,i,j}^{(l+1/2)}\right] + \frac{\gamma_{m,i,j}^y \eta_m}{\Delta y_j}\left[\psi_{m,i,j+1/2}^{x,(l+1/2)} - \psi_{m,i,j-1/2}^{x,(l+1/2)}\right] + \sigma_{i,j}\psi_{m,i,j}^{x,(l+1/2)} = \frac{c}{2\pi}\phi_{i,j}^{x,(l)} \ , \tag{26b}$$

TABLE VIII

Spectral Radius Versus Mesh Size for the SC Method, $\beta = 0.5$, $c = 0.999$, S_2 Level-Symmetric Quadrature Set

	Δy							
Δx	0.01	0.1	0.3	1.0	3.0	10.0	30.0	100.0
0.01	0.499							
0.1	0.477	0.473						
0.3	0.477	0.475	0.459					
1.0	0.477	0.476	0.466	0.375				
3.0	0.477	0.476	0.468	0.090	0.163			
10.0	0.477	0.476	0.469	0.395	0.210	0.094		
30.0	0.477	0.477	0.469	0.396	0.225	0.107	0.035	
100.0	0.477	0.477	0.469	0.397	0.230	0.117	0.037	0.011

$$\frac{\gamma^x_{m,i,j}\mu_m}{\Delta x_i}\left[\psi^{y,(l+1/2)}_{m,i+1/2,j}-\psi^{y,(l+1/2)}_{m,i-1/2,j}\right]+\frac{\kappa^y_{m,i,j}\eta_m}{\Delta y_j}\left[\psi^{(l+1/2)}_{m,i,j+1/2}+\psi^{(l+1/2)}_{m,i,j-1/2}-2\psi^{(l+1/2)}_{m,i,j}\right]+\sigma_{i,j}\psi^{y,(l+1/2)}_{m,i,j}=\frac{c}{2\pi}\phi^{y,(l)}_{i,j}\ , \tag{26c}$$

and

$$\frac{\delta^x_{m,i,j}\mu_m}{\Delta x_i}\left[\psi^{y,(l+1/2)}_{m,i+1/2,j}+\psi^{y,(l+1/2)}_{m,i-1/2,j}-2\psi^{y,(l+1/2)}_{m,i,j}\right]+\frac{\delta^y_{m,i,j}\eta_m}{\Delta y_j}\left[\psi^{x,(l+1/2)}_{m,i,j+1/2}+\psi^{x,(l+1/2)}_{m,i,j-1/2}-2\psi^{x,(l+1/2)}_{m,i,j}\right]+\sigma_{i,j}\psi^{xy,(l+1/2)}_{m,i,j}=\frac{c}{2\pi}\phi^{xy,(l)}_{i,j}\ , \tag{26d}$$

where we have replaced some constants with the parameters κ, γ, and δ, which we discuss shortly. Equations (26) contain eight unknowns; thus, to close the system we need four more equations. These are simple "upstream" equations at cell edges, which can be written as

$$\psi_{m,i\pm1/2,j}=\psi_{m,i,j}\pm\psi^x_{m,i,j}\ , \tag{27a}$$

$$\psi^y_{m,i\pm1/2,j}=\psi^y_{m,i,j}\pm\psi^{xy}_{m,i,j}\ , \tag{27b}$$

$$\psi_{m,i,j\pm1/2}=\psi_{m,i,j}\pm\psi^y_{m,i,j}\ , \tag{27c}$$

and

$$\psi^x_{m,i,j\pm1/2}=\psi^x_{m,i,j}\pm\psi^{xy}_{m,i,j}\ . \tag{27d}$$

Given that the parameters κ, γ, and δ can be varied in Eqs. (26), we now have a family of BLD methods (see Refs. 14 and 15 for more details). The important point here is that we can analyze TSA for this entire family of methods. When we do so we find, after a bit of matrix algebra, the following:

$$\omega\boldsymbol{B}=\left\{\frac{c}{4\pi}\mathbf{X}+\left[\mathbf{I}_4-\frac{c(1-\beta)}{2\pi(1-\beta c)}\tilde{\mathbf{X}}\right]^{-1}\times\tilde{\mathbf{X}}\left(\frac{c}{4\pi}\mathbf{X}-\mathbf{I}_4\right)\frac{c}{2\pi(1-\beta c)}\right\}\boldsymbol{B}\ , \tag{28}$$

where $\mathbf{X}$ and $\tilde{\mathbf{X}}$ are 4×4 matrices whose elements depend on θ_x, θ_y, Δx, Δy, $c\beta$, and the high- and low-order quadrature sets.

Thus, we have an explicit formula for the iteration operator. We compute its eigenvalues using a mathematical library (Lapack) and are left with four eigenvalues per mode (instead of one with SC); some of them are complex. Our analysis code retains only the largest in magnitude of these four.

III.B.2. Results

We focused on the standard BLD and the simple corner balance (SCB) methods,[15] which differ only in the values of the coefficients κ, γ, and δ introduced in Eqs. (26). For standard BLD, these are

$$\kappa^x_m=\kappa^y_m=3.0\ , \tag{29a}$$

$$\gamma^x_m=\gamma^y_m=1.0\ , \tag{29b}$$

and

$$\delta^x_m=\delta^y_m=3.0\ , \tag{29c}$$

and for SCB they are

$$\kappa^x_m=\kappa^y_m=1.0\ , \tag{30a}$$

$$\gamma^x_m=\gamma^y_m=1.0\ , \tag{30b}$$

and

$$\delta^x_m=\delta^y_m=1.0\ . \tag{30c}$$

As expected, we observed the same trends as with SC. We find that in general, spectral radii are larger for BLD and SCB than for SC because their spectral radii do not decrease as rapidly for coarse meshes. (In the fine-mesh limit, all methods behave like the continuous equations.)

In Table IX and Table X, we present results for $\beta=0.8$ and an S_4 quadrature set. Like with SC, TSA for BLD and SCB is unconditionally stable. The spectral radius decreases more slowly as the mesh size increases than was true with SC (see Table VI) but the basic trends are the same for all methods. Because the low-order problem becomes easier to solve as β increases, it is very encouraging to note that the spectral radius can be small even with a relatively high value of β (such as 0.8).

An even more attractive solution might be to choose a lower value for β. (This will be practical only if we find efficient means to reduce the computational cost of the low-order problem, which is the object of Sec. IV.) We will find it useful to know the overall spectral radius for a small β, such as 0.2, which we report in Table XI. The spectral radius is only 0.18 in the worst

TABLE IX

Spectral Radius Versus Mesh Size for the BLD Method, $\beta=0.8$, $c=0.999$, S_4 Level-Symmetric Quadrature Set

	Δy							
Δx	0.01	0.1	0.3	1.0	3.0	10.0	30.0	100.0
0.01	0.746							
0.1	0.746	0.745						
0.3	0.746	0.745	0.745					
1.0	0.746	0.746	0.746	0.746				
3.0	0.746	0.745	0.745	0.746	0.745			
10.0	0.746	0.745	0.745	0.746	0.745	0.739		
30.0	0.746	0.745	0.745	0.746	0.745	0.734	0.719	
100.0	0.746	0.745	0.745	0.746	0.745	0.739	0.708	0.615

TABLE X

Spectral Radius Versus Mesh Size for the SCB Method, $\beta = 0.8$, $c = 0.999$, S_4 Level-Symmetric Quadrature Set

	Δy							
Δx	0.01	0.1	0.3	1.0	3.0	10.0	30.0	100.0
0.01	0.745							
0.1	0.745	0.745						
0.3	0.745	0.745	0.745					
1.0	0.745	0.745	0.745	0.745				
3.0	0.745	0.745	0.745	0.745	0.744			
10.0	0.745	0.745	0.745	0.745	0.743	0.732		
30.0	0.735	0.745	0.745	0.745	0.742	0.721	0.691	
100.0	0.745	0.745	0.745	0.745	0.742	0.722	0.655	0.433

TABLE XI

Spectral Radius Versus Mesh Size for the BLD Method, $\beta = 0.2$, $c = 0.999$, S_4 Level-Symmetric Quadrature Set

	Δy							
Δx	0.01	0.1	0.3	1.0	3.0	10.0	30.0	100.0
0.01	0.182							
0.1	0.182	0.182						
0.3	0.182	0.182	0.182					
1.0	0.182	0.182	0.182	0.182				
3.0	0.182	0.182	0.182	0.182	0.181			
10.0	0.182	0.182	0.182	0.182	0.181	0.176		
30.0	0.182	0.182	0.182	0.182	0.181	0.174	0.170	
100.0	0.182	0.182	0.182	0.182	0.181	0.173	0.160	0.098

case, so that only a few high-order iterations are necessary to converge. This is very encouraging and provides motivation to find methods to decrease the computational cost of the low-order problem for relatively small β.

In this section, we analyzed TSA of the SC, BLD, and SCB spatial discretizations. We concluded that the properties observed for continuous TSA are still valid when applied to discretized equations. In addition, despite a slower rate of convergence for BLD and SCB than for SC, we observed the same major trends for all spatial discretization methods: The spectral radius is a decreasing function of the mesh size, and as the cells become optically thicker, the negative eigenvalues vanish. We also emphasized that a relatively high β (0.8) associated with an S_4 level-symmetric quadrature set was sufficient to make the method unconditionally stable. Finally, we showed that only a few high-order iterations are necessary to reach convergence for a small value of β (such as 0.2), but we note that this has little practical value unless we find a way to efficiently solve the low-order problem. This is the subject of Sec. IV.

IV. EFFICIENCY ISSUES

The analysis of the TSA family for both continuous and discretized transport equations showed that the number of high-order iterations could be reduced substantially compared to SI. However, the computational cost of the low-order problem remains to be addressed. Indeed, the reduction in high-order iterations might not be sufficient to compensate for the amount of work needed to solve the low-order problem as it has been defined until now. We propose three means of reducing this computational cost:

1. implementing a CG method on the low-order problem
2. setting the maximum number of low-order iterations per high-order iteration, N_{it}, to a finite value; i.e., not allowing the low-order problem to converge very tightly
3. use a cruder quadrature set ($N = N_{lo}$) in the low-order problem.

We first describe these features, then we discuss their combined effect on the method. Finally, we propose values for β, N_{lo}, and N_{it}.

IV.A. Conjugate Gradient Method

In the previous sections, we defined a transport accelerator to reduce the number of high-order iterations compared with SI. However, for each high-order iteration, we assumed that the low-order transport problem was solved. If we use SI to solve the low-order problem, the spectral radius of its solution procedure, at least in an infinite uniform medium, is the effective scattering ratio of that problem:

$$\tilde{\rho} = \frac{(1-\beta)c}{(1-\beta c)} \quad . \tag{31}$$

In Table XII, we give the value of $\tilde{\rho}$ for different scattering ratios and β varying from 0 to 1.0. As expected, $\tilde{\rho}$ is a decreasing function of β. This emphasizes the trade-off between an easy-to-solve low-order problem

TABLE XII

Low-Order Spectral Radius Using SI

	β							
c	1	0.95	0.9	0.8	0.6	0.4	0.2	0
0.999	0	0.980	0.990	0.995	0.997	0.998	0.998	0.999
0.99	0	0.832	0.908	0.952	0.975	0.983	0.987	0.99
0.95	0	0.487	0.655	0.792	0.884	0.919	0.938	0.95
0.9	0	0.310	0.474	0.643	0.783	0.843	0.878	0.9
0.7	0	0.104	0.189	0.318	0.483	0.583	0.651	0.7
0.5	0	0.048	0.091	0.167	0.286	0.375	0.444	0.5

(large β) with a larger number of high-order iterations, and a more difficult low-order problem (small β) with a smaller number of high-order iterations. It also appears that as c becomes closer to 1.0, $\tilde{\rho}$ is still high, even for a large value of β such as 0.9, and the low-order problem will require a large number of iterations to converge.

In Table XIII, we compute the number of iterations necessary to reduce the low-order error by a factor of 10. In addition, Table XIV gives the number of iterations needed to reduce the high-order error by a factor of 10 when no accelerator is used. If we focus on $\beta = 0.8$, a safe value for all quadrature sets (except S_2 for which a smaller β is needed), we note that it takes roughly a factor of 5 fewer iterations to converge the low-order problem than to converge the high-order problem using SI. This, by itself, is quite discouraging, for if the high-order problem requires more than five iterations to converge (a reasonable assumption), the CPU time will be higher for TSA than for SI. However, these figures stand for an infinite medium. In a real problem, leakage acts like absorption and accelerates convergence. Since $\sigma_t(1-\beta c) < \sigma_t$, this effect is greater on the low-order problem. Moreover, we have already analyzed the possibility of using a cruder quadrature set for the low-order problem. This does not reduce the number of iterations but acts on the CPU time, because less computation is necessary.

TABLE XIII

Number of Iterations Needed to Reduce Low-Order Error by a Factor of 10 with SI

	c					
β	0.999	0.99	0.95	0.9	0.7	0.5
1	0	0	0	0	0	0
0.95	116.1	12.5	3.2	2.0	1.0	0.8
0.9	231.2	23.9	5.5	3.1	1.4	1.0
0.8	461.2	46.7	9.9	5.2	2.0	1.3
0.6	921.3	92.3	18.6	9.4	3.2	1.8
0.4	1381.3	137.9	27.4	13.5	4.3	2.3
0.2	1841.4	183.5	36.1	17.7	5.9	3.1
0.0	2301.4	229.1	44.9	21.8	6.5	3.3

TABLE XIV

Number of Iterations Needed to Reduce High-Order Error by a Factor of 10 Without TSA

c	0.999	0.99	0.95	0.9	0.7	0.5
Iterations	2301.4	229.1	44.9	21.8	6.5	3.3

We can act directly on the low-order convergence rate by implementing CG acceleration on the low-order problem.[12] This algebraic method lowers the spectral radius but requires the underlying operator to be positive definite and symmetric. In general, CG can accelerate the iterative solution of the system $\mathbf{A}\boldsymbol{F} = \boldsymbol{s}$, where $\mathbf{A}$ is symmetric positive definite (SPD). For our application, the matrix $\mathbf{A}$ is given by

$$\mathbf{A} = \mathbf{I} - \mathbf{M} , \tag{32}$$

where $\mathbf{M}$ is the SI matrix for the low-order problem. The source vector is

$$\boldsymbol{s} = \mathbf{M}_s \boldsymbol{q} , \tag{33a}$$

where

$$\boldsymbol{q} = \frac{\sigma_s}{4\pi} [\boldsymbol{\phi}^{(l+1/2)} - \boldsymbol{\phi}^{(l)}] , \tag{33b}$$

and where $\mathbf{M}_s$ is a matrix whose definition is not important for our purpose. With these definitions, the CG algorithm can be written as

$$\begin{aligned}
&\boldsymbol{r}^{(0)} = \boldsymbol{d}^{(0)} = \mathbf{A}\boldsymbol{F}^{(0)} - \boldsymbol{s} , \\
&\boldsymbol{\Omega}\cdot\nabla \boldsymbol{f}_m^{(l+1/2)} + \sigma_t(1-\beta c)\boldsymbol{f}_m^{(l+1/2)} = \frac{\sigma_s}{2\pi}\boldsymbol{d}^{(l)} , \\
&\boldsymbol{d}^{(l+1/2)} = \sum_{m=1}^{M_{lo}} w_m \boldsymbol{f}_m^{(l+1/2)} , \\
&\mathbf{A}\boldsymbol{d}^{(l)} = \boldsymbol{d}^{(l)} - \boldsymbol{d}^{(l+1/2)} , \\
&\alpha_l = \frac{\langle \boldsymbol{r}^{(l)}, \boldsymbol{r}^{(l)} \rangle}{\langle \boldsymbol{d}^{(l)}, \mathbf{A}\boldsymbol{d}^{(l)} \rangle} , \\
&\boldsymbol{F}^{(l+1)} = \boldsymbol{F}^{(l)} + \alpha_l \boldsymbol{d}^{(l)} , \\
&\boldsymbol{r}^{(l+1)} = \boldsymbol{r}^{(l)} - \alpha_l \mathbf{A}\boldsymbol{d}^{(l)} , \\
&\beta_{l+1} = \frac{\langle \boldsymbol{r}^{(l+1)}, \boldsymbol{r}^{(l+1)} \rangle}{\langle \boldsymbol{r}^{(l)}, \boldsymbol{r}^{(l)} \rangle} ,
\end{aligned} \tag{34}$$

and

$$\boldsymbol{d}^{(l+1)} = \boldsymbol{r}^{(l+1)} + \beta_{l+1}\boldsymbol{d}^{(1)} ,$$

where the scalar product $\langle \boldsymbol{u}, \boldsymbol{v} \rangle$ is defined as

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = (1-\beta) \sum_{i=1}^{N} (\sigma_{si} u_i v_i \Delta x_i \Delta y_i) , \tag{35}$$

where N is the number of spatial unknowns. This definition is crucial, because the matrix $\mathbf{A}$ is symmetric only for this scalar product. The easiest way to see this is to note that, given isotropic scattering and no spatial discretization, $\mathbf{A}$ becomes the self-adjoint integral operator:

$$A\phi(\boldsymbol{r}) = \phi(\boldsymbol{r}) - \int_{Volume} \sigma_s(\boldsymbol{r}')\phi(\boldsymbol{r}') \frac{e^{-\alpha(\boldsymbol{r}',\boldsymbol{r})}}{4\pi|\boldsymbol{r}'-\boldsymbol{r}|^2} d^3r' , \tag{36}$$

where $\alpha(\boldsymbol{r}',\boldsymbol{r})$ is the optical distance (number of mfp's) on the line between $\boldsymbol{r}'$ and $\boldsymbol{r}$. Note that

$$\begin{aligned}
&\int_{Volume} \sigma_s(\boldsymbol{r})g(\boldsymbol{r})Af(\boldsymbol{r})\,d^3r \\
&= \int_{Volume} \sigma_s(\boldsymbol{r})\left[g(\boldsymbol{r})f(\boldsymbol{r}) - \int_{Volume} \sigma_s(\boldsymbol{r}')g(\boldsymbol{r})f(\boldsymbol{r}') \right. \\
&\qquad \left. \times \frac{e^{-\alpha(\boldsymbol{r}',\boldsymbol{r})}}{4\pi|\boldsymbol{r}'-\boldsymbol{r}|^2}\,d^3r'\right] d^3r \\
&= \int_{Volume} \sigma_s(\boldsymbol{r}')f(\boldsymbol{r}')\left[g(\boldsymbol{r}') - \int_{Volume} \sigma_s(\boldsymbol{r})g(\boldsymbol{r}) \right. \\
&\qquad \left. \times \frac{e^{-\alpha(\boldsymbol{r}',\boldsymbol{r})}}{4\pi|\boldsymbol{r}'-\boldsymbol{r}|^2}\,d^3r\right] d^3r' \\
&= \int_{Volume} \sigma_s(\boldsymbol{r}')f(\boldsymbol{r}')Ag(\boldsymbol{r}')\,d^3r' \ . \qquad (37)
\end{aligned}$$

This means that if we define the scalar product

$$\langle f,g\rangle \equiv \int_{Volume} \sigma_s(\boldsymbol{r})g(\boldsymbol{r})f(\boldsymbol{r})\,d^3r \ , \qquad (38)$$

then we have

$$\langle g,Af\rangle \equiv \langle Ag,f\rangle \ . \qquad (39)$$

Thus, our inner product needs to be the discrete version of a volume integral with σ_s as a multiplier, as in Eq. (35).

Given anisotropic scattering, the integral operator corresponding to the matrix **A** is not symmetric under any inner product. The physical problem being asymmetric, we cannot use CG on a problem that has anisotropic scattering. This is why we do not propose to use CG to accelerate high-order transport sweeps. If the high-order problem does have anisotropic scattering, we envision that TSA would still use a low-order problem with isotropic scattering, which would permit the use of CG for solving the low-order problem. (If scattering is sufficiently forward-peaked this probably will not work.) These issues will be addressed in future work.

The aforementioned manipulations show that the integral operator A is self-adjoint when scattering is isotropic. It is also easy to show that A is positive definite: Because $A = I - M$, where M is the iteration operator for SI, each eigenvalue of A is 1 − an eigenvalue of M. In every problem for which SI converges, the eigenvalues of M are strictly <1, thus the eigenvalues of A are strictly positive.

An obvious question is whether discretization changes these properties of the operator A. That is, is the discrete version **A** of the integral operator A SPD? The positivity of the eigenvalues follows from $\mathbf{A} = \mathbf{I} - \mathbf{M}$, with the eigenvalues of **M** strictly <1. Symmetry is a more complicated question, however. At this point, we have determined that a fairly wide variety of transport discretizations do give rise to symmetric **A** matrices, at least given the correct inner products and given symmetric quadrature sets.

We emphasize that the matrix **A** is never directly constructed. Its action on a vector $\boldsymbol{d}$ is determined by a transport sweep, with $\boldsymbol{d}$ playing the role of a scalar flux, as shown in Eq. (34). Thus, a CG iteration requires very little extra work relative to an SI.

Theory predicts that with CG the low-order spectral radius should be[12]

$$\tilde{\rho}_{cg} = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} \ , \qquad (40)$$

where κ is the condition number of the matrix **A**. To evaluate this number, we recall that

$$\mathbf{A} = \mathbf{I} - \mathbf{M} \ , \qquad (41)$$

where **M** is the SI matrix. In an infinite medium, **M** has eigenvalues ranging from 0 to $\tilde{\rho}$. Hence, **I** − **M** has eigenvalues ranging from to $1-\tilde{\rho}$ to 1. Since, by definition

$$\kappa = \frac{\max|\lambda_A|}{\min|\lambda_A|} \ , \qquad (42)$$

we have

$$\kappa = \frac{1}{1-\tilde{\rho}} \ . \qquad (43)$$

In Table XV, we give the number of iterations needed to reduce the error by a factor of 10 when CG is used. Note that in the more difficult cases (i.e., when c is close to 1.0), CG slows down but still achieves a large reduction in iteration count relative to SI (Table XIV). Reduction from hundreds to tens is the norm and should therefore make CG computationally much more efficient than SI for solving the low-order problem.

IV.B. Limits on Low-Order Iterations

Conjugate gradient with a cruder quadrature for the low-order problem might be sufficient to reduce

TABLE XV

Number of Iterations Needed to Reduce Low-Order Error by a Factor of 10 with CG

	β							
c	1	0.95	0.9	0.8	0.6	0.4	0.2	0
0.999	0	8.2	11.5	16.3	23.0	28.2	32.5	36.4
0.99	0	2.6	3.7	5.2	7.3	8.9	10.3	11.5
0.95	0	1.3	1.7	2.3	3.2	3.9	4.5	5.1
0.9	0	1.0	1.3	1.7	2.3	2.8	3.2	3.5
0.7	0	0.6	0.8	1.0	1.3	1.5	1.7	1.9
0.5	0	0.5	0.6	0.7	0.9	1.1	1.2	1.3

computational cost relative to SI, but we investigate one extra parameter: limiting the maximum number of low-order iterations per high-order iteration, N_{it}. We believe that it is not necessary to converge the low-order problem tightly. More high-order iterations may then be needed, but we expect the overall computational cost to decrease substantially and be considerably lower than that of SI. We explore the effects of this in Sec. V where we numerically test our TSA scheme for a variety of simple problems in X-Y geometry.

V. NUMERICAL RESULTS

To test TSA, we modified an existing code that contained the family of BLD methods described in Sec. III. Implementing TSA proved very simple because only two principal subroutines were added to the existing code. We focused on the acceleration of the within-group iteration, so all test problems had a single energy group.

The first test compares the performance of the various options derived in the previous sections. We considered the problem:

1. 100 × 100 mfp homogeneous medium
2. $c = 0.99$
3. 4 × 4 mfp cells
4. S_8 high-order quadrature set
5. isotropic incident flux on the left face of the mesh
6. vacuum boundary conditions on other faces of the mesh
7. volumetric isotropic source

A diagram of the problem is given in Fig. 6, and results are reported in Table XVI.

Using SI as a reference for both the number of high-order iterations and the CPU time, we first set β to 0.2 but let the low-order problems converge and use the same quadrature set (S_8) for both the high- and low-order problems. Note a dramatic decrease in high iteration count: from 897 for SI to only 5 with TSA. Yet, this is not sufficient to compensate the computational cost of the low-order problem: The overall CPU time increases from 650 to 1140 s. However, the effect of the CG method is spectacular, as case 3 shows: only 201 low-order iterations remain necessary instead of 1455 previously. Here, TSA begins to show true promise: It is at this point four times faster than SI.

Another way to reduce the computational cost of the low-order problem is to use a cruder quadrature set (S_2 instead of S_8) as case 4 shows. The CPU reduction relative to SI is comparable to that achieved by CG but the reasons are very different: Whereas CG reduces the number of low-order iterations, using S_2 reduces the cost per low-order iteration. This results in one additional high-order iteration but the overall CPU time remains competitive: 154 s instead of 650 s for SI.

In case 5, we combine CG and the crude low-order quadrature to obtain a remarkable reduction: TSA performs 22 times faster than SI. However, so far we have let the low-order problem converge for each transport

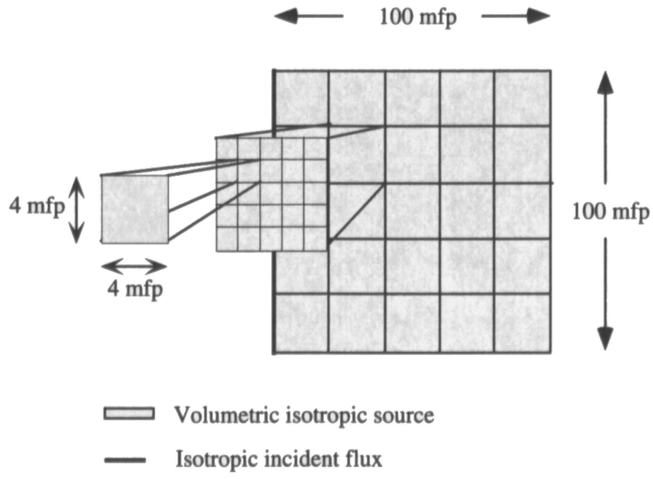

Fig. 6. Test problem 1.

TABLE XVI

Comparison of TSA and SI for an S_8 High-Order Quadrature Set, $\Delta x = \Delta y = 4$ mfp and $c = 0.99$

Case	Method	High-Order Iterations	Low-Order Iterations	CPU	Gain
1	SI	897	---	650	1
2	$\beta = 0.2$	5	1455	1140	0.6
3	$\beta = 0.2$ + CG	5	201	158	4
4	$\beta = 0.2 + S_2$	6	1517	154	4
5	$\beta = 0.2$ + CG + S_2	6	225	29	22
6	$\beta = 0.2$ + CG + $S_2 + N_{it} = 10$	7	68	13	49
7	$\beta = 0.2$ + CG + $S_2 + N_{it} = 7$	7	49	11	58

sweep. Limiting each low-order problem to ten iterations further enhances the performance of TSA, making it 49 times faster than SI. Finally, a further reduction to seven iterations produced a factor of 58 overall speedup. These results are comparable to results that one would expect with a consistent DSA method. However, TSA is much simpler to derive and implement than DSA.

In Table XVII, we consider the same physical problem but for different scattering ratios. Again, we use an S_2 low-order quadrature set, $\beta = 0.2$, and $N_{it} = 7$. Table XVII shows that as c gets closer to 1, TSA provides more and more advantage over SI; i.e., when c increases from 0.99 to 0.999, the gain over SI increases from 58 to 165.

In Table XVI, the parameters $\beta = 0.2$ and $N_{it} = 7$ proved extremely efficient. However, there could be an even better combination. Therefore, in Table XVIII, we show the computational cost obtained for the physical problem described in Fig. 6, as β varies from 0.1 to 0.9 and N_{it} from 1 to 10. Also, we give the CPU time corresponding to a converged solution of the low-order problem ($N_{it} = \infty$). For each value of β, the smallest computational cost is underlined. We see that as β increases, the optimal value of N_{it} decreases, which confirms the theory developed in Sec. IV: When β increases, it takes fewer iterations for the low-order problem to converge. In addition, we have highlighted the region in which the CPU time performs the best. The range for β and N_{it} is fairly broad, which means that in implementing TSA, we need not be overly concerned with finding the optimal values for these parameters. We somewhat arbitrarily chose $\beta = 0.2$ and $N_{it} = 7$ for the remainder of our tests.

The results reported in Table XVI suggest that an S_2 quadrature set is the most economic solution for the low-order problem. This trend is confirmed in Table XIX: We consider the physical problem described in Fig. 6 but vary both the high- and low-order quadrature sets. Note that the S_2 low-order quadrature set always performs more efficiently than higher low-order quadratures. (We believe that for high-order quadrature sets finer than S_{16}, an S_4 low-order quadrature set may be more appropriate than S_2 in many problems.) Also, as Table III shows, an S_4 low-order quadrature may be needed for fine-mesh problems, because S_2 can be unstable. Note also that the computational gain of TSA over SI is an increasing function of the high quadrature order. This is expected because SI performs all computations with the maximum number of angles, whereas TSA relies on a cruder quadrature set for a substantial part of its calculation, namely the low-order problem.

In Table XX, we focus on TSA performances for the combination $\beta = 0.2$ and $N_{it} = 7$ as a function of the cell dimensions. We consider a slightly different problem than the one described in Fig. 6: Sources and

TABLE XVII

CPU Time as a Function of c

c	SI	TSA	Gain
0.9	69	10.1	6.83
0.99	650	11.1	58
0.999	4135	25	165
0.9999	9410	42	224
0.99999	10700	44	244

TABLE XVIII

CPU Time as a Function of N_{it} and β

N_{it}	β								
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
1	44.46	35.78	36.23	41.03	32.29	30.89	50.22	39.72	32.43
2	21.84	24.66	22.45	20.88	20.76	16.72	19.69	15.40	35.29
3	18.33	19.11	16.92	16.45	15.22	13.75	16.91	25.56	40.39
4	16.23	14.73	13.54	13.28	13.51	16.37	22.30	29.22	41.43
5	13.18	13.61	11.71	12.74	14.48	19.30	28.16	28.74	45.30
6	14.15	11.88	12.03	14.20	16.27	19.53	24.15	30.07	51.32
7	12.93	11.14	15.23	14.34	16.12	20.91	24.42	34.09	55.65
8	11.73	11.66	13.71	15.10	16.47	20.15	25.92	33.63	59.13
9	10.79	12.79	12.44	15.33	17.62	21.06	26.17	37.15	64.34
10	11.19	13.35	13.05	15.31	18.58	22.04	28.28	37.54	64.46
Infinity	28.61	29.74	31.97	33.70	41.20	47.31	48.18	58.86	81.66

TABLE XIX

Influence of Quadrature Sets on the Ratio (SI CPU Time/TSA CPU Time), $c = 0.99$, $\beta = 0.2$, $N_{it} = 7$

Low Order	High Order					
	S_2	S_4	S_6	S_8	S_{12}	S_{16}
S_2	13	30	47	58	73	90
S_4		14	24	35	55	70

TABLE XX

TSA Speedup Relative to SI as a Function of Cell Size, S_8/S_2 Quadrature Sets, $c = 0.99$, $\beta = 0.2$, $N_{it} = 7$

Δx	Δy				
	1.0	2.0	5.0	10.0	25.0
1.0	41	42	42	40	43
2.0	47	46	46	42	40
5.0	47	48	54	49	44
10.0	53	46	55	50	43
25.0	45	48	46	45	39

incident fluxes are identical, but the problem is reduced to 50 × 50 mfp and the dimensions of a cell, Δx and Δy, become variables as we can see in Fig. 7.

After solving the problem with SI and TSA, we compute the ratio of their computational costs and conclude that the computational speedup of TSA over SI is roughly independent of cell size.

Finally, we extend our results to a heterogeneous problem. It is very similar to the first test problem:

1. $c = 0.99$ for the whole problem
2. 25 × 25 square cells
3. S_8 high-order quadrature set
4. isotropic source on the left face of the mesh
5. vacuum boundary conditions on other faces of the mesh
6. volumetric isotropic source.

However, as we can see in Fig. 8, we use a different material in the four regions located at the bottom left corner of the problem.

In Table XXI, we present the computational gain of TSA over SI for square cells ($\Delta x = \Delta y$) in both media. The total cross sections are σ_{t1} and σ_{t2} in material 1 and 2, respectively, as they are defined in Fig. 8.

Despite isolated variations which, again, we attribute to differences in CPU load, the speedup is roughly an increasing function of the optical thickness of the problem. The relatively lower speedups of TSA for small values of ($\sigma_{t1}\Delta x, \sigma_{t2}\Delta x$) are due to the preponderance of leakage in optically thin problems, which makes SI relatively fast. Note that even for very heterogeneous problems such as $\sigma_{t1}\Delta x = 0.1$ mfp and $\sigma_{t2}\Delta x = 100.0$ mfp, TSA achieves a speedup factor of 40. That is, we see no performance degradation in heterogeneous problems.

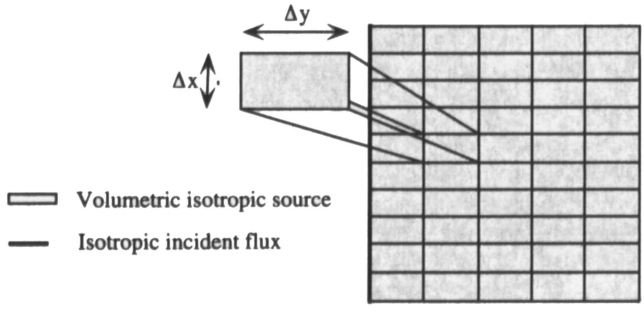

Fig. 7. Problem solved in Table XX.

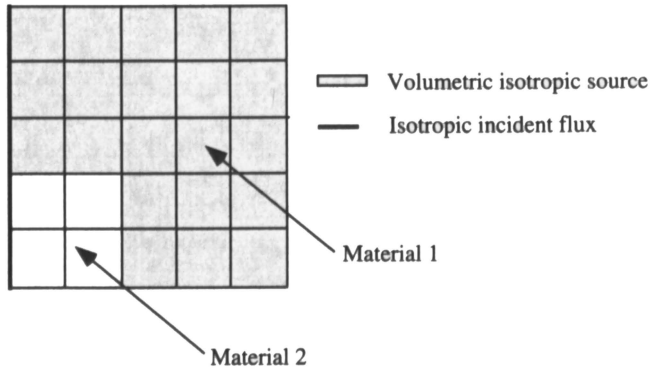

Fig. 8. Description of the problem solved in Table XXI.

TABLE XXI

TSA Speedup Relative to SI for a Heterogeneous Medium, S_8/S_2 Quadrature Sets, $c = 0.99$, $\beta = 0.2$, $N_{it} = 7$

$\sigma_{t2}\Delta x$ (mfp)	$\sigma_{t1}\Delta x$ (mfp)				
	0.1	1.0	10.0	100.0	1000.0
0.1	13	26	23	48	72
1.0	14	47	58	51	65
10.0	27	61	74	84	75
100.0	40	46	84	73	78
1000.0	45	52	73	71	89

TABLE XXII

DSA Speedup Relative to SI for a Heterogeneous Medium, S_8 Quadrature Set, $c = 0.99$*

$\sigma_{t2}\Delta x$ (mfp)	$\sigma_{t1}\Delta x$ (mfp)		
	0.1	10.0	1000.0
0.1	4	94	84
10.0	96	126	141
1000.0	87	116	109

*See text for DSA details.

To show that TSA is comparable to DSA for problems of this type, in Table XXII, we present DSA results[a] for a subset of the problems considered previously. The results were generated using the BLD transport discretization and a BLD diffusion discretization.[6] The diffusion equation was solved using a very efficient multigrid scheme that is an improvement over that described in Ref. 16; to our knowledge, this is the most efficient DSA scheme in existence. We see that overall this DSA method is faster than our TSA, but not overwhelmingly so, and of course it is much more difficult to implement than TSA.

In this section, we have compared the efficiency of TSA to that of SI. We found that taking $\beta = 0.2$, $N_{it} = 7$, and an S_2 low-order quadrature set leads to very good results for a wide range of problems. We have emphasized that the choice of β and N_{it} does not have to be precise and that many combinations lead to good results as long as β is <0.5. In addition, we have shown that as problems become more difficult, the computational speedup of TSA relative to SI increases dramatically. A reduction in time execution by a factor of 50 or more is the norm for a scattering ratio >0.99 (precisely the range for which this accelerator was needed). Finally, we have found that TSA provides significant speedups relative to SI for all cell sizes and material properties we tested. That is, TSA helps the most when it is most needed, but it even speeds up problems for which SI is already acceptably fast.

VI. CONCLUSION

We have developed a family of TSA methods for the iterative solution of transport problems. A single TSA iteration involves an ordinary transport sweep followed by the solution of a transport problem that has a reduced scattering cross section and a coarser angular discretization. (The algorithm can be viewed as Richardson iteration with a coarse transport operator used as a preconditioner.) We have analyzed, implemented, and tested our TSA algorithm for several spatial discretizations of the discrete ordinates transport equation in X-Y geometry. Implementation was very simple; very little new coding was added to our existing discrete ordinates program. Our results are extremely encouraging: For the problems considered, we observe iteration counts and CPU speedups comparable to the best DSA methods. Further, we find that our speedups relative to SI increase as problems become iteratively more difficult—TSA helps most when it is needed most.

However, we have not answered several questions that are beyond the scope of this paper but are nevertheless of practical importance. We briefly discuss them here.

1. *Anisotropic scattering.* In this paper, we have focused on problems with isotropic scattering. Many problems of interest, however, contain anisotropic scattering. We believe that for mild anisotropy, our TSA algorithm will remain effective. We propose to simply ignore the higher order scattering terms in the low-order problem and proceed as before. Our belief that this will work is based on the fact that DSA, whose low-order equation accounts only for zeroth- and first-order scattering coefficients, remains effective for mildly anisotropic scattering.[17] We could include anisotropic scattering in the low-order problem, but this causes asymmetry and thus does not allow use of the CG method.

Given highly anisotropic scattering, encountered in many photon and electron transport problems, the simple strategy outlined previously will probably fail. However, in this case we can turn to the multigrid-in-angle method of Morel and Manteuffel,[18] and simply use TSA (with CG) to accelerate the convergence of the problem on the crudest grid (S_2 or S_4 quadrature set).

2. *Upscattering.* Given problems with significant upscattering (i.e., scattering from low-energy groups to higher energy groups), it is often not sufficient to merely accelerate within-group iterations: the group-to-group scattering iteration also requires acceleration. Such problems often arise in the analyses of nuclear reactors. We believe that TSA can be extended to address upscattering, in much the same way that DSA was extended by Adams and Morel.[19]

3. *Eigenvalue problems.* Most problems in nuclear reactor analysis are posed as eigenvalue problems, and power iteration is almost always the basis for their iterative solution. In large systems, power iteration can be quite slow to converge, and thus begs for some acceleration. An obvious candidate for this is shifted power iteration[20] (SPI); however, SPI requires at each step the solution of a problem with a significant amount of upscattering. This has deterred the widespread use of SPI. However, if an algorithm for quickly solving

[a]These results were provided by T. A. Wareing of Los Alamos National Laboratory.

upscattering problems is available, SPI becomes attractive. Our belief is that once TSA is extended to upscattering problems, it will be a relatively simple extension to make it work in conjunction with SPI to quickly solve eigenvalue problems.

4. *Optically thick, highly scattering problems.* Given problems that are extremely optically thick (thousands of mfp's) and very highly scattering ($c > 0.9999$), the TSA algorithm described here continues to exhibit spectacular speedups relative to SI but nevertheless becomes far slower than one would like. Such problems often arise in radiative heat transfer. The source of TSA's difficulty is apparently that the low-order problem itself becomes difficult to converge, even with CG. We are exploring possible solutions; among these is the idea of "nested" TSA iterations, in which a low-order transport problem is itself accelerated by TSA (possibly boosted by CG).

5. *Effect of discretization on properties of operator A.* In this paper, we showed that an integral operator A is self-adjoint and positive definite. An important question is whether discrete versions of this operator are SPD. We showed in Sec. IV that the discrete operators are positive definite for problems of interest, but we did not fully address the question of symmetry. We have found that given the spatial discretizations studied in this paper (SC, BLD, and SCB), the inner product described by Eq. (35) does in fact "symmetrize" the operator. Our preliminary investigations indicate that "symmetrizing" inner products can also be found relatively easily for most finite element and characteristic spatial discretizations.

However, some spatial discretizations may introduce asymmetry into the transport operator. In particular, it may be that recently developed nonlinear methods may produce asymmetries. If this is true, then we may be unable to use CG in the low-order problem, and an alternative may need to be investigated. Possibilities include Krylov-subspace (or other) methods that do not require symmetry;[21] however, most such methods are more difficult to implement and more costly to use than CG. Note that CG should not necessarily be abandoned for asymmetric low-order operators: Although CG does not produce the solution of the "real" low-order problem in such cases, the solution it produces can be very effective as an additive correction in TSA (Ref. 22).

6. *Opposing reflecting boundaries.* Given problems with reflecting boundaries that are opposite one another, it is not possible to completely update the angular flux with a single transport sweep. That is, on any given transport sweep the problem is driven not only by a scattering source that uses information from a previous iteration, but also by some incident fluxes from a previous iteration. This raises several complications. First, it requires the addition of an angular flux residual as a driving term in the low-order equation. (This is not difficult.) Second, and perhaps most important, it appears that the presence of angular-flux unknowns on the boundary may significantly complicate the CG acceleration of the low-order iteration.

Thus, the TSA methods developed in this work appear very promising but much work remains before they can be applied to all problems of practical interest. We are currently investigating these questions and plan to report our results in future communications.

APPENDIX A

DERIVATION OF A/B FOR THE STEP CHARACTERISTIC SPATIAL DISCRETIZATION

In this section, we present some details of the Fourier analysis for the SC method as it was introduced in Sec. III. We also give expressions for the coefficients Nb_I, Nb_{II}, Nd_I, Nd_{II}, D_I, and D_{II}.

First, we recall the discretized transport equation in X-Y geometry for the SC method, using the same notation as in Sec. III:

$$\frac{\mu_m}{\Delta x_i}\left[\psi^{(l+1/2)}_{m,i+1/2,j}-\psi^{(l+1/2)}_{m,i-1/2,j}\right]+\frac{\eta_m}{\Delta y_j}\left[\psi^{(l+1/2)}_{m,i,j+1/2}-\psi^{(l+1/2)}_{m,i,j-1/2}\right]+\sigma_{t_{i,j}}\psi^{(l+1/2)}_{m,i,j}=S^{(l)}_{m,i,j}\ , \tag{A.1}$$

where

$$\psi^{(l+1/2)}_{m,i\pm1/2,j}=\frac{S^{(l)}_{m,i,j}}{\sigma_{t_{i,j}}}+\gamma_{m_{i,j}}\left[\psi^{(l+1/2)}_{m,i,j\mp1/2}-\frac{S^{(l)}_{m,i,j}}{\sigma_{t_{i,j}}}\right]\left(\frac{1-e^{-\tau_{m_{i,j}}}}{\tau_{m_{i,j}}}\right)+\left[\psi^{(l+1/2)}_{m,i\mp1/2,j}-\frac{S^{(l)}_{m,i,j}}{\sigma_{t_{i,j}}}\right](1-\gamma_{m_{i,j}})e^{-\tau_{m_{i,j}}} \tag{A.2a}$$

and

$$\psi^{(l+1/2)}_{m,i,j\pm1/2}=\frac{S^{(l)}_{m,i,j}}{\sigma_{t_{i,j}}}+\alpha_{m_{i,j}}\left[\psi^{(l+1/2)}_{m,i\mp1/2,j}-\frac{S^{(l)}_{m,i,j}}{\sigma_{t_{i,j}}}\right]\left(\frac{1-e^{-\tau_{m_{i,j}}}}{\tau_{m_{i,j}}}\right)+\left[\psi^{(l+1/2)}_{m,i,j\mp1/2}-\frac{S^{(l)}_{m,i,j}}{\sigma_{t_{i,j}}}\right](1-\alpha_{m_{i,j}})e^{-\tau_{m_{i,j}}}\ . \tag{A.2b}$$

To conduct our Fourier analysis, we set

$$\hat{\psi}^{(l+1/2)}_{m,i,j}=\omega^l a_m e^{k(\lambda_x x_i+\lambda_y y_j)}\ ,\quad k\equiv\sqrt{-1}\ , \tag{A.3a}$$

$$\hat{\psi}^{(l+1/2)}_{m,i+1/2,J} = \omega^l b_m e^{k(\lambda_x x_{i+1/2}+\lambda_y y_J)} , \tag{A.3b}$$

$$\hat{\psi}^{(l+1/2)}_{m,i,j+1/2} = \omega^l d_m e^{k(\lambda_x x_i+\lambda_y y_{j+1/2})} , \tag{A.3c}$$

$$\hat{\phi}^{(l+1/2)}_{i,J} = \omega^l A e^{k(\lambda_x x_i+\lambda_y y_J)} , \tag{A.3d}$$

and

$$\hat{\phi}^{(l+1/2)}_{i,J} = \omega^l B e^{k(\lambda_x x_i+\lambda_y y_J)} . \tag{A.3e}$$

From Eq. (A.1) we get

$$2k\frac{\mu_m b_m}{\Delta x}\sin\left(\frac{\theta_x}{2}\right) + 2kd_m\frac{\eta_m d_m}{\Delta y}\sin\left(\frac{\theta_y}{2}\right) + \chi a_m = \chi\frac{cB}{2\pi} , \tag{A.4}$$

where Δx and Δy are in unit of mfp and $\chi = \sigma_t/\sigma_t = 1.0$.

Equations (A.2a) and (A.2b) give

$$b_m[e^{k(\theta_x/2)} - e^{-k(\theta_x/2)}(1-\gamma_m)e^{-\tau_m}] = \frac{cB}{2\pi}\hat{r}_m + d_m e^{\mp k(\theta_y/2)}\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) , \quad \mu_m > 0 , \quad \eta_m \gtrless 0 , \tag{A.5a}$$

$$b_m[e^{-k(\theta_x/2)} - e^{k(\theta_x/2)}(1-\gamma_m)e^{-\tau_m}] = \frac{cB}{2\pi}\hat{r}_m + d_m e^{\mp k(\theta_y/2)}\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) , \quad \mu_m < 0 , \quad \eta_m \gtrless 0 , \tag{A.5b}$$

$$d_m[e^{k(\theta_y/2)} - e^{-k(\theta_y/2)}(1-\alpha_m)e^{-\tau_m}] = \frac{cB}{2\pi}\hat{s}_m + b_m e^{\mp k(\theta_x/2)}\alpha_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) , \quad \eta_m > 0 , \quad \mu_m \gtrless 0 , \tag{A.5c}$$

and

$$d_m[e^{-k(\theta_y/2)} - e^{k(\theta_y/2)}(1-\alpha_m)e^{-\tau_m}] = \frac{cB}{2\pi}\hat{s}_m + b_m e^{\mp k(\theta_x/2)}\alpha_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) , \quad \eta_m < 0 , \quad \mu_m \gtrless 0 , \tag{A.5d}$$

where

$$\hat{r}_m = 1 - (1-\gamma_m)e^{-\tau_m} - \gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) \tag{A.6a}$$

and

$$\hat{s}_m = 1 - (1-\alpha_m)e^{-\tau_m} - \alpha_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) . \tag{A.6b}$$

Also, b_m and d_m are complex numbers that can be written as

$$b_m = b_{\mathrm{Re},m} + \sqrt{-1}\,b_{\mathrm{Im},m} \tag{A.7a}$$

and

$$d_m = d_{\mathrm{Re},m} + \sqrt{-1}\,d_{\mathrm{Im},m} . \tag{A.7b}$$

In Sec. A.I, we focus on the imaginary parts of b_m and d_m. The reason will become apparent later on. Since their definition depends on the angular quadrant in which μ_m and η_m are located, we consider these quadrants one by one.

A.I. DERIVATION OF $b_{\mathrm{Im},m}$ AND $d_{\mathrm{Im},m}$

A.I.A. First Quadrant ($\mu_m > 0$, $\eta_m > 0$)

Equations (A.5a) and (A.5c) give

$$D_I = \left\{\cos\left(\frac{\theta_x+\theta_y}{2}\right)\left[1 - \alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \cos\left(\frac{\theta_x-\theta_y}{2}\right)[2-(\alpha_m+\gamma_m)]e^{-\tau_m}\right\}^2$$
$$+ \left\{\sin\left(\frac{\theta_x+\theta_y}{2}\right)\left[1 + \alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \sin\left(\frac{\theta_x-\theta_y}{2}\right)(\gamma_m-\alpha_m)e^{-\tau_m}\right\}^2 , \tag{A.8a}$$

$$
\begin{aligned}
Nb_I = &\cos\left(\frac{\theta_y}{2}\right)\left\{\hat{s}_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) + \hat{r}_m[1-(1-\alpha_m)e^{-\tau_m}]\right\} \\
&\times\left\{\sin\left(\frac{\theta_x+\theta_y}{2}\right)\left[1+\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \sin\left(\frac{\theta_x-\theta_y}{2}\right)(\gamma_m-\alpha_m)e^{-\tau_m}\right\} \\
&+\sin\left(\frac{\theta_y}{2}\right)\left\{\hat{s}_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) - \hat{r}_m[1+(1-\alpha_m)e^{-\tau_m}]\right\} \\
&\times\left\{\cos\left(\frac{\theta_x+\theta_y}{2}\right)\left[1-\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \cos\left(\frac{\theta_x-\theta_y}{2}\right)[2-(\alpha_m+\gamma_m)]e^{-\tau_m}\right\}, \qquad \text{(A.8b)}
\end{aligned}
$$

and

$$
\begin{aligned}
Nd_I = &\cos\left(\frac{\theta_x}{2}\right)\left\{\hat{r}_m\alpha_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) + \hat{s}_m[1-(1-\gamma_m)e^{-\tau_m}]\right\} \\
&\times\left\{\sin\left(\frac{\theta_x+\theta_y}{2}\right)\left[1+\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \sin\left(\frac{\theta_x-\theta_y}{2}\right)(\gamma_m-\alpha_m)e^{-\tau_m}\right\} \\
&+\sin\left(\frac{\theta_x}{2}\right)\left\{\hat{r}_m\alpha_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) - \hat{s}_m[1+(1-\gamma_m)e^{-\tau_m}]\right\} \\
&\times\left\{\cos\left(\frac{\theta_x-\theta_y}{2}\right)\left[1-\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \cos\left(\frac{\theta_x-\theta_y}{2}\right)[2-(\alpha_m+\gamma_m)]e^{-\tau_m}\right\}. \qquad \text{(A.8c)}
\end{aligned}
$$

With those definitions, we have

$$b_{\mathrm{Im},m} = -\frac{cB}{2\pi}\frac{Nb_I}{D_I} \qquad \text{(A.9a)}$$

and

$$d_{\mathrm{Im},m} = -\frac{cB}{2\pi}\frac{Nd_I}{D_I}\ . \qquad \text{(A.9b)}$$

Note that we can find directly Nd_I from Nb_I by replacing all parameters in Eq. (A.8b) by their counterparts in Eq. (A.8c), following the rule:

$$
\begin{aligned}
b_m &\leftrightarrow d_m\ , \\
\alpha_m &\leftrightarrow \gamma_m\ , \\
\gamma_m &\leftrightarrow \alpha_m\ , \\
\theta_x &\leftrightarrow \theta_y\ , \\
\theta_y &\leftrightarrow \theta_x\ , \\
\hat{r}_m &\leftrightarrow \hat{s}_m\ ,
\end{aligned}
$$

and

$$\hat{s}_m \leftrightarrow \hat{r}_m\ . \qquad \text{(A.10)}$$

A.I.B. Second Quadrant ($\mu_m > 0$, $\eta_m < 0$)

Equations (A.5b) and (A.5d) give

$$
\begin{aligned}
D_{II} = &\left\{\cos\left(\frac{\theta_x-\theta_y}{2}\right)\left[1-\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \cos\left(\frac{\theta_x+\theta_y}{2}\right)[2-(\alpha_m+\gamma_m)]e^{-\tau_m}\right\}^2 \\
&+\left\{\sin\left(\frac{\theta_x-\theta_y}{2}\right)\left[1+\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \sin\left(\frac{\theta_x+\theta_y}{2}\right)[\gamma_m-\alpha_m]e^{-\tau_m}\right\}^2, \qquad \text{(A.11a)}
\end{aligned}
$$

$$Nb_{II} = \cos\left(\frac{\theta_y}{2}\right)\left\{\hat{s}_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) + \hat{r}_m[1-(1-\alpha_m)e^{-\tau_m}]\right\}$$
$$\times\left\{\sin\left(\frac{\theta_x-\theta_y}{2}\right)\left[1+\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \sin\left(\frac{\theta_x+\theta_y}{2}\right)[\gamma_m-\alpha_m]e^{-\tau_m}\right\}$$
$$+\sin\left(\frac{\theta_y}{2}\right)\left\{-\hat{s}_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) + \hat{r}_m[1+(1-\alpha_m)e^{-\tau_m}]\right\}$$
$$\times\left\{\cos\left(\frac{\theta_x-\theta_y}{2}\right)\left[1-\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \cos\left(\frac{\theta_x+\theta_y}{2}\right)[2-(\alpha_m+\gamma_m)]e^{-\tau_m}\right\}, \quad \text{(A.11b)}$$

and

$$Nd_{II} = \cos\left(\frac{\theta_x}{2}\right)\left\{\hat{r}_m\alpha_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) + \hat{s}_m[1-(1-\gamma_m)e^{-\tau_m}]\right\}$$
$$\times\left\{\sin\left(\frac{\theta_x-\theta_y}{2}\right)\left[1+\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \sin\left(\frac{\theta_x+\theta_y}{2}\right)[\gamma_m-\alpha_m]e^{-\tau_m}\right\}$$
$$+\sin\left(\frac{\theta_x}{2}\right)\left\{\hat{r}_m\alpha_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) - \hat{s}_m[1+(1-\gamma_m)e^{-\tau_m}]\right\}$$
$$\times\left\{\cos\left(\frac{\theta_x-\theta_y}{2}\right)\left[1-\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] - \cos\left(\frac{\theta_x+\theta_y}{2}\right)[2-(\alpha_m+\gamma_m)]e^{-\tau_m}\right\}. \quad \text{(A.11c)}$$

Again, with those definitions, we have

$$b_{\mathrm{Im},m} = -\frac{cB}{2\pi}\frac{Nb_{II}}{D_{II}} \quad \text{(A.12a)}$$

and

$$d_{\mathrm{Im},m} = -\frac{cB}{2\pi}\frac{Nd_{II}}{D_{II}}. \quad \text{(A.12b)}$$

We can get directly Nd_{II} from Nb_{II} by using the relations

$$b_m \leftrightarrow d_m,$$
$$\alpha_m \leftrightarrow \gamma_m,$$
$$\gamma_m \leftrightarrow \alpha_m,$$
$$\theta_x \leftrightarrow -\theta_y,$$
$$\theta_y \leftrightarrow -\theta_x,$$
$$\hat{r}_m \leftrightarrow \hat{s}_m,$$

and

$$\hat{s}_m \leftrightarrow \hat{r}_m. \quad \text{(A.13)}$$

A.I.C. Third Quadrant ($\mu_m < 0$, $\eta_m < 0$)

Once the work has been done for the first quadrant, we can deduct the results for the third quadrant by simply replacing θ_x by $-\theta_x$ and θ_y by $-\theta_y$. Hence, we get

$$D_{III} = D_I, \quad \text{(A.14a)}$$

$$Nb_{III} = -Nb_I, \quad \text{(A.14b)}$$

and

$$Nd_{III} = -Nd_I. \quad \text{(A.14c)}$$

Therefore, the expressions for $b_{\mathrm{Im},m}$ and $d_{\mathrm{Im},m}$ become

$$b_{\mathrm{Im},m} = \frac{cB}{2\pi}\frac{Nb_I}{D_I} \quad \text{(A.15a)}$$

and

$$d_{\mathrm{Im},m} = \frac{cB}{2\pi}\frac{Nd_I}{D_I}. \quad \text{(A.15b)}$$

A.I.D. Fourth Quadrant ($\mu_m < 0$, $\eta_m > 0$)

We can follow the same steps as in case *III* by replacing θ_x by $-\theta_x$ and θ_y by $-\theta_y$ in the second quadrant. We obtain the following relations:

$$D_{IV} = D_{II}, \quad \text{(A.16a)}$$

$$Nb_{IV} = -Nb_{II}, \quad \text{(A.16b)}$$

and

$$Nd_{IV} = -Nd_{II}. \quad \text{(A.16c)}$$

Finally, we deduce the equations for $b_{\mathrm{Im},m}$ and $d_{\mathrm{Im},m}$:

$$b_{\mathrm{Im},m} = \frac{cB}{2\pi}\frac{Nb_{II}}{D_{II}} \tag{A.17a}$$

and

$$d_{\mathrm{Im},m} = \frac{cB}{2\pi}\frac{Nd_{II}}{D_{II}} \; . \tag{A.17b}$$

A.II. $b_{\mathrm{Re},m}$ AND $d_{\mathrm{Re},m}$

So far, we have not taken the real part of b_m and d_m into account. Instead of deriving all cases as we did for their imaginary parts, we propose to focus only on one quadrant ($\mu_m > 0$, $\eta_m > 0$) to prove that the overall contribution of $b_{\mathrm{Re},m}$ and $d_{\mathrm{Re},m}$ to the determination of ω is null. By similarity, with the derivation of $b_{\mathrm{Im},m}$, we define

$$\begin{aligned} Nb\mathrm{Re}_I = \cos\left(\frac{\theta_y}{2}\right)\Bigg\{ & \hat{s}_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right) \\ & + \hat{r}_m[1-(1-\alpha_m)e^{-\tau_m}]\Bigg\} \\ \times \Bigg\{ & \cos\left(\frac{\theta_x+\theta_y}{2}\right)\left[1-\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] \\ & - \cos\left(\frac{\theta_x-\theta_y}{2}\right)[2-(\alpha_m+\gamma_m)]e^{-\tau_m}\Bigg\} \\ + \sin\left(\frac{\theta_y}{2}\right)\Bigg\{ & \hat{r}_m[1+(1-\alpha_m)e^{-\tau_m}] \\ & - \hat{s}_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)\Bigg\} \\ \times \Bigg\{ & \sin\left(\frac{\theta_x+\theta_y}{2}\right)\left[1+\alpha_m\gamma_m\left(\frac{1-e^{-\tau_m}}{\tau_m}\right)^2\right] \\ & - \sin\left(\frac{\theta_x-\theta_y}{2}\right)(\gamma_m-\alpha_m)e^{-\tau_m}\Bigg\} \; . \end{aligned} \tag{A.18}$$

Thus, for the first quadrant,

$$b^I_{\mathrm{Re},m} = \frac{cB}{2\pi}\frac{Nb\mathrm{Re}_I}{D_I} \; . \tag{A.19}$$

To get $b_{\mathrm{Re},m}$ for the third quadrant ($\mu_m < 0$, $\eta_m < 0$), we only need to replace θ_x by $-\theta_x$ and θ_y by $-\theta_y$ in the equations. A look at Eq. (A.18) clearly indicates that

$$Nb\mathrm{Re}_{III} = Nb\mathrm{Re}_I \; . \tag{A.20}$$

Besides, we already know from Eq. (13a) that

$$D_{III} = D_I \; . \tag{A.21}$$

Hence, to each angle of the quadrature located in the first quadran corresponds an angle in the third quadrature such that

$$b^I_{\mathrm{Re},m} = b^{III}_{\mathrm{Re},m'} \; . \tag{A.22}$$

Besides, those angles verify

$$\mu_m = -\mu_{m'} \; ,$$

$$\eta_m = -\eta_{m'} \; ,$$

and

$$w_m = w_{m'} \; . \tag{A.23}$$

This derivation can be extended to the second and fourth quadrant so that

$$\begin{aligned} \sum_{m=1}^{M}\left(\frac{\mu_m}{\Delta x}w_m b_{\mathrm{Re},m}\right) &= \sum_{\mu_m>0}\left(\frac{\mu_m}{\Delta x}w_m b_{\mathrm{Re},m}\right) \\ &\quad + \sum_{\mu_m<0}\left(\frac{\mu_m}{\Delta x}w_m b_{\mathrm{Re},m}\right) \\ &= \sum_{\mu_m>0}\left(\frac{\mu_m}{\Delta x}w_m b_{\mathrm{Re},m}\right) \\ &\quad - \sum_{\mu_m>0}\left(\frac{\mu_m}{\Delta x}w_m b_{\mathrm{Re},m}\right) = 0 \; . \end{aligned} \tag{A.24}$$

Similarly,

$$\sum_{m=1}^{M}\left(\frac{\eta_m}{\Delta y}w_m d_{\mathrm{Re},m}\right) = 0 \; . \tag{A.25}$$

Thus, the real part of b_m and d_m do not contribute to the derivation of ω.

A.III. FORMULA FOR A/B

If we multiply Eq. (A.4) by the weight w_m and carry a summation over all angles of the quadrature, we find

$$2\sqrt{-1}\left[\sin\left(\frac{\theta_x}{2}\right)\sum_{m=1}^{M}\left(\frac{\mu_m}{\Delta x}w_m b_m\right) + \sin\left(\frac{\theta_y}{2}\right)\sum_{m=1}^{M}\left(\frac{\eta_m}{\Delta y}w_m d_m\right)\right] + \chi A = \chi cB \; . \tag{A.26}$$

Recalling Eqs. (A.24) and (A.25), Eq. (A.26) becomes

$$-2\left[\sin\left(\frac{\theta_x}{2}\right)\sum_{m=1}^{M}\left(\frac{\mu_m}{\Delta x}\,w_m b_{\mathrm{Im},m}\right)+\sin\left(\frac{\theta_y}{2}\right)\sum_{m=1}^{M}\left(\frac{\eta_m}{\Delta y}\,w_m d_{\mathrm{Im},m}\right)\right]+\chi A=\chi cB\ . \tag{A.27}$$

Equations (A.15) and (A.16) then allow us to perform some factorizations:

$$-4\left[\sin\left(\frac{\theta_x}{2}\right)\sum_{\mu_m>0}\left(\frac{\mu_m}{\Delta x}\,w_m b_{\mathrm{Im},m}\right)+\sin\left(\frac{\theta_y}{2}\right)\sum_{\eta_m>0}\left(\frac{\eta_m}{\Delta y}\,w_m d_{\mathrm{Im},m}\right)\right]+\chi A=\chi cB\ . \tag{A.28}$$

Finally, we use Eqs. (A.9) and (A.12) and come to the following formula for A/B:

$$\frac{A}{B}=c\left\{1-\frac{2}{\pi\chi}\left[\begin{array}{l}\sin\left(\frac{\theta_x}{2}\right)\sum_{quad\ I}w_m\,\frac{\mu_m}{\Delta x}\left(\frac{Nb_I}{D_I}+\frac{Nb_{II}}{D_{II}}\right)\\ +\sin\left(\frac{\theta_y}{2}\right)\sum_{quad\ I}w_m\,\frac{\eta_m}{\Delta y}\left(\frac{Nd_I}{D_I}+\frac{Nd_{IV}}{D_{IV}}\right)\end{array}\right]\right\}\ . \tag{A.29}$$

APPENDIX B

DEFINITION OF THE MATRICES $\mathbf{M}_m$

In this section we give an expression for the matrices $\mathbf{M}_m$, which are introduced in Sec. III. Their derivation is based on the relation between the angular flux at the center of a cell and at its border:

$$\psi^{(l+1/2)}_{m,i+1/2,j}=\psi^{(l+1/2)}_{m,i,j}+\psi^{x,(l+1/2)}_{m,i,j}\ ,\quad \mu_m>0\ , \tag{B.1a}$$

$$\psi^{(l+1/2)}_{m,i+1/2,j}=\psi^{(l+1/2)}_{m,i,j}-\psi^{x,(l+1/2)}_{m,i,j}\ ,\quad \mu_m<0\ , \tag{B.1b}$$

$$\psi^{(l+1/2)}_{m,i,j+1/2}=\psi^{(l+1/2)}_{m,i,j}+\psi^{y,(l+1/2)}_{m,i,j}\ ,\quad \eta_m>0\ , \tag{B.1c}$$

and

$$\psi^{(l+1/2)}_{m,i,j+1/2}=\psi^{(l+1/2)}_{m,i,j}-\psi^{y,(l+1/2)}_{m,i,j}\ ,\quad \eta_m<0\ . \tag{B.1d}$$

If we substitute these equations into Eqs. (28), we obtain a formula for $\mathbf{M}_m$ that depends on the angular quadrant. For readability, we give the matrix column by column.

B.I. FIRST QUADRANT ($\mu_m>0$, $\eta_m>0$)

$$\text{First column}=\begin{bmatrix}\frac{\mu_m}{\Delta x}(1-e^{-j\theta_x})\\ \quad+\frac{\eta_m}{\Delta y}(1-e^{-j\theta_y})+\sigma\\ \frac{\kappa^x_m\mu_m}{\Delta x}(e^{-j\theta_x}-1)\\ \frac{\kappa^y_m\eta_m}{\Delta y}(e^{-j\theta_y}-1)\\ 0.0\end{bmatrix}\ ,\quad j\equiv\sqrt{-1}\ .$$

$$\text{Second column}=\begin{bmatrix}\frac{\mu_m}{\Delta x}(1-e^{-j\theta_x})\\ \frac{\kappa^x_m\mu_m}{\Delta x}(1-e^{-j\theta_x})\\ \quad+\frac{\gamma^y_m\eta_m}{\Delta y}(1-e^{-j\theta_y})+\sigma\\ 0.0\\ \frac{\delta^y_m\eta_m}{\Delta y}(e^{-j\theta_y}-1)\end{bmatrix}\ .$$

$$\text{Third column}=\begin{bmatrix}\frac{\eta_m}{\Delta y}(1-e^{-j\theta_y})\\ 0.0\\ \frac{\gamma^x_m\mu_m}{\Delta x}(1-e^{-j\theta_x})\\ \quad+\frac{\kappa^y_m\eta_m}{\Delta y}(1-e^{-j\theta_y})+\sigma\\ \frac{\delta^x_m\mu_m}{\Delta x}(e^{-j\theta_x}-1)\end{bmatrix}\ .$$

$$\text{Fourth column}=\begin{bmatrix}0.0\\ \frac{\gamma^y_m\eta_m}{\Delta y}(1-e^{-j\theta_y})\\ \frac{\gamma^y_m\mu_m}{\Delta x}(1-e^{-j\theta_x})\\ \frac{\delta^x_m\mu_m}{\Delta x}(1+e^{-j\theta_x})\\ \quad+\frac{\delta^y_m\eta_m}{\Delta y}(1+e^{-j\theta_y})+\sigma\end{bmatrix}\ .$$

B.II. SECOND QUADRANT ($\mu_m > 0$, $\eta_m < 0$)

$$\text{First column} = \begin{bmatrix} \frac{\mu_m}{\Delta x}(1 - e^{-j\theta_x}) \\ \quad + \frac{\eta_m}{\Delta y}(e^{j\theta_y} - 1) + \sigma \\ \frac{\kappa_m^x \mu_m}{\Delta x}(e^{-j\theta_x} - 1) \\ \frac{\kappa_m^y \eta_m}{\Delta y}(e^{j\theta_y} - 1) \\ 0.0 \end{bmatrix}.$$

$$\text{Second column} = \begin{bmatrix} \frac{\mu_m}{\Delta x}(1 - e^{-j\theta_x}) \\ \frac{\kappa_m^x \mu_m}{\Delta x}(1 + e^{-j\theta_x}) \\ \quad + \frac{\gamma_m^y \eta_m}{\Delta y}(e^{j\theta_y} - 1) + \sigma \\ 0.0 \\ \frac{\delta_m^y \eta_m}{\Delta y}(e^{-j\theta_y} - 1) \end{bmatrix}.$$

$$\text{Third column} = \begin{bmatrix} \frac{\eta_m}{\Delta y}(1 - e^{j\theta_y}) \\ 0.0 \\ \frac{\gamma_m^x \mu_m}{\Delta x}(1 - e^{-j\theta_x}) \\ \quad - \frac{\kappa_m^y \eta_m}{\Delta y}(1 + e^{j\theta_y}) + \sigma \\ \frac{\delta_m^x \mu_m}{\Delta x}(e^{-j\theta_x} - 1) \end{bmatrix}.$$

$$\text{Fourth column} = \begin{bmatrix} 0.0 \\ \frac{\gamma_m^y \eta_m}{\Delta y}(1 - e^{j\theta_y}) \\ \frac{\gamma_m^x \mu_m}{\Delta x}(1 - e^{-j\theta_x}) \\ \frac{\delta_m^x \mu_m}{\Delta x}(1 + e^{-j\theta_x}) \\ \quad - \frac{\delta_m^y \eta_m}{\Delta y}(1 + e^{j\theta_y}) + \sigma \end{bmatrix}.$$

B.III. THIRD QUADRANT ($\mu_m < 0$, $\eta_m < 0$)

$$\text{First column} = \begin{bmatrix} \frac{\mu_m}{\Delta x}(e^{j\theta_x} - 1) \\ \quad + \frac{\eta_m}{\Delta y}(e^{j\theta_y} - 1) + \sigma \\ \frac{\kappa_m^x \mu_m}{\Delta x}(e^{j\theta_x} - 1) \\ \frac{\kappa_m^y \eta_m}{\Delta y}(e^{j\theta_y} - 1) \\ 0.0 \end{bmatrix}.$$

$$\text{Second column} = \begin{bmatrix} \frac{\mu_m}{\Delta x}(1 - e^{j\theta_x}) \\ -\frac{\kappa_m^x \mu_m}{\Delta x}(1 + e^{j\theta_x}) \\ \quad + \frac{\gamma_m^y \eta_m}{\Delta y}(e^{j\theta_y} - 1) + \sigma \\ 0.0 \\ \frac{\delta_m^y \eta_m}{\Delta y}(e^{j\theta_y} - 1) \end{bmatrix}.$$

$$\text{Third column} = \begin{bmatrix} \frac{\eta_m}{\Delta y}(1 - e^{j\theta_y}) \\ 0.0 \\ \frac{\gamma_m^x \mu_m}{\Delta x}(e^{j\theta_x} - 1) \\ \quad - \frac{\kappa_m^y \eta_m}{\Delta y}(1 + e^{j\theta_y}) + \sigma \\ \frac{\delta_m^x \mu_m}{\Delta x}(e^{j\theta_x} - 1) \end{bmatrix}.$$

$$\text{Fourth column} = \begin{bmatrix} 0.0 \\ \frac{\gamma_m^y \eta_m}{\Delta y}(1 - e^{j\theta_y}) \\ \frac{\gamma_m^x \mu_m}{\Delta x}(1 - e^{j\theta_x}) \\ -\frac{\delta_m^x \mu_m}{\Delta x}(1 + e^{j\theta_x}) \\ \quad - \frac{\delta_m^y \eta_m}{\Delta y}(1 + e^{j\theta_y}) + \sigma \end{bmatrix}.$$

B.IV. FOURTH QUADRANT ($\mu_m < 0$, $\eta_m > 0$)

$$\text{First column} = \begin{bmatrix} \frac{\mu_m}{\Delta x}(e^{j\theta_x} - 1) \\ \quad + \frac{\eta_m}{\Delta y}(1 - e^{-j\theta_y}) + \sigma \\ \frac{\kappa_m^x \mu_m}{\Delta x}(e^{j\theta_x} - 1) \\ \frac{\kappa_m^y \eta_m}{\Delta y}(e^{-j\theta_y} - 1) \\ 0.0 \end{bmatrix}.$$

$$\text{Second column} = \begin{bmatrix} \frac{\mu_m}{\Delta x}(1 - e^{j\theta_x}) \\ -\frac{\kappa_m^x \mu_m}{\Delta x}(1 + e^{j\theta_x}) \\ \quad + \frac{\gamma_m^y \eta_m}{\Delta y}(1 - e^{-j\theta_y}) + \sigma \\ 0.0 \\ \frac{\delta_m^y \eta_m}{\Delta y}(e^{-j\theta_y} - 1) \end{bmatrix}.$$

$$\text{Third column} = \begin{bmatrix} \frac{\eta_m}{\Delta y}(1 - e^{-j\theta_y}) \\ 0.0 \\ \frac{\gamma_m^x \mu_m}{\Delta x}(e^{j\theta_x} - 1) \\ \quad + \frac{\kappa_m^y \eta_m}{\Delta y}(1 + e^{-j\theta_y}) + \sigma \\ \frac{\delta_m^x \mu_m}{\Delta x}(e^{j\theta_x} - 1) \end{bmatrix}.$$

$$\text{Fourth column} = \begin{bmatrix} 0.0 \\ \frac{\gamma_m^y \eta_m}{\Delta y}(1 - e^{-j\theta_y}) \\ \frac{\gamma_m^x \mu_m}{\Delta x}(1 - e^{j\theta_x}) \\ -\frac{\delta_m^x \mu_m}{\Delta x}(1 + e^{j\theta_x}) \\ \quad + \frac{\delta_m^y \eta_m}{\Delta y}(1 + e^{-j\theta_y}) + \sigma \end{bmatrix}.$$

ACKNOWLEDGMENTS

We thank T. Wareing for providing DSA timings that we used for comparison. Part of this work was supported by the Lawrence Livermore National Laboratory under subcontract B307875. Part of this work was supported by NSF grant CCR-9302782. Work by the first author (GLR) was in partial fulfillment of the requirements for the Master of Science degree in Nuclear Engineering at Texas A&M University.

REFERENCES

1. E. E. LEWIS and W. F. MILLER, Jr., *Computational Methods of Neutron Transport*, Wiley, New York (1984).

2. H. J. KOPP, "Synthetic Method Solution of the Transport Equation," *Nucl. Sci. Eng.*, **17**, 65 (1963).

3. W. H. REED, "The Effectiveness of Acceleration Techniques for Iterative Methods in Transport Theory," *Nucl. Sci. Eng.*, **45**, 245 (1971).

4. R. E. ALCOUFFE, "Diffusion Synthetic Acceleration Methods for the Diamond-Differenced Discrete-Ordinates Equations," *Nucl. Sci. Eng.*, **64**, 344 (1977).

5. 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).

6. M. L. ADAMS and W. R. MARTIN, "Boundary Projection Acceleration: A New Approach to Synthetic Acceleration of Transport Calculations," *Nucl. Sci. Eng.*, **100**, 177 (1988).

7. D. R. McCOY and E. W. LARSEN, "Unconditionally Stable Diffusion-Synthetic Acceleration Methods for the Slab Geometry Discrete Ordinates Equations. Part II: Numerical Results," *Nucl. Sci. Eng.*, **82**, 64 (1982).

8. M. L. ADAMS and W. R. MARTIN, "Diffusion Synthetic Acceleration of Discontinuous Finite Element Transport Iterations," *Nucl. Sci. Eng.*, **111**, 145 (1992).

9. V. Y. GOL'DIN, "A Quasi-Diffusion Method of Solving the Kinetic Equation," *USSR Comp. Math. Phys.*, **4**, 136 (1967).

10. H. KHALIL, "Effectiveness of a Consistently Formulated Diffusion-Synthetic Acceleration Differencing Approach," *Nucl. Sci. Eng.*, **98**, 226 (1988).

11. E. W. LARSEN and W. F. MILLER, Jr., "A Two-Step Acceleration Method for Transport Problems," *Trans. Am. Nucl. Soc.*, **52**, 416 (1986).

12. J. R. SHEWCHUK, "An Introduction to the Conjugate Gradient Without the Agonizing Pain," School of Computer Science, Carnegie Mellon University (1994).

13. R.E ALCOUFFE, E. W. LARSEN, W. F. MILLER, Jr., and B. R. WIENKE, "Computational Efficiency of Numerical Methods for the Multigroup, Discrete-Ordinates Neutron Transport Equations: The Slab Geometry Case," *Nucl. Sci. Eng.*, **71**, 111 (1979).

14. T. A. WAREING, E. W. LARSEN, and M. L. ADAMS, "Diffusion Accelerated Discontinuous Finite Element Schemes for the Sn Equations in Slab and X,Y Geometries," *Proc. Topl. Mtg. Advances in Mathematics, Computations, and Reactor Physics*, Pittsburgh, Pennsylvania, April 28–May 2, 1991, Vol. 3, Sec. 11.1, 2-1 (1991).

15. M. L. ADAMS, "A New Transport Discretization Scheme for Arbitrary Spatial Meshes in XY Geometry," *Proc. Topl. Mtg. Advances in Mathematics, Computations, and Reactor Physics*, Pittsburgh, Pennsylvania, April 28–May 2, 1991, Vol. 3, Sec. 13.2, 2-1 (1991).

16. J. E. MOREL, J. E. DENDY, Jr., and T. A. WAREING, "Diffusion-Accelerated Solution of the Two-Dimensional S_n Equations with Bilinear-Discontinuous Differencing," *Nucl. Sci. Eng.*, **115**, 304 (1993).

17. M. L. ADAMS and T. A. WAREING, "Diffusion-Synthetic Acceleration Given Anisotropic Scattering, General Quadratures, and Multidimensions," *Trans. Am. Nucl. Soc.*, **68**, 203 (1993).

18. J. E. MOREL and T. A. MANTEUFFEL, "An Angular Multigrid Acceleration Technique for S_n Equations with Highly Forward-Peaked Scattering," *Nucl. Sci. Eng.*, **107**, 330 (1991).

19. B. T. ADAMS and J. E. MOREL, "A Two-Grid Acceleration Scheme for the Multigroup S_n Equations with Neutron Upscattering," *Nucl. Sci. Eng.*, **115**, 253 (1993).

20. T. M. SUTTON, "Wielandt Iteration as Applied to the Nodal Expansion Method," *Nucl. Sci. Eng.*, **98**, 169 (1988).

21. S. F. ASHBY, P. N. BROWN, M. R. DORR, and A. C. HINDMARSH, "Preconditioned Iterative Methods for Discretized Transport Equations," *Proc. Topl. Mtg. Advances in Mathematics, Computations, and Reactor Physics*, Pittsburgh, Pennsylvania, April 28–May 2, 1991, Vol. 2, Sec. 6.1, 2-1 (1991).

22. C. L. CASTRIANNI and M. L. ADAMS, "A Synthetically Accelerated Nonlinear Corner-Balance Spatial Discretization for Transport on Arbitrary Grids" (submitted for publication).