A Fast Multigrid Algorithm for Isotropic Transport Problems I: Pure Scattering *

T. Manteuffel†, S. McCormick‡, J. Morel§, S. Oliveira§ and G. Yang§

Abstract

A multigrid method for solving the 1-D slab-geometry $S_N$ equations with isotropic scattering and no absorption is presented. This scheme is highly compatible with massively parallel computer architectures and represents a first step towards similar multigrid methods for the $S_N$ equations in curvilinear and multi-dimensional geometries. Extensive theoretical analyses are given for our scheme which indicate that it is extremely efficient. In fact, the method is so efficient that it very nearly represents an exact solution technique. Results from calculations are presented which validate the theoretical results. The case with absorption is treated in a sequel to this paper [12].

§1. Introduction

In this paper we describe a fast method for solving the equations used to model the transport of neutral particles with isotropic scattering in slab geometry. These problems are important in many application such as nuclear reactor design, radiation therapy in medical science, radiation effects on global weather, and are a fundamental part of many algorithms used to model more complicated applications such as satellite electronics shielding. For example, a multigrid technique for the solution of highly forward-peaked anisotropic scattering in slab geometry requires the solution of an isotropic problem at the coarsest level (Morel and Manteuffel [15]). The main focus of this paper is the extensive theoretical results which we have obtained for our multigrid method. The matrix analysis techniques which we use are quite different from the Fourier analysis techniques traditionally used in numerical transport theory (e.g., see Morel and Manteuffel [15]). Since these matrix techniques yield information which is not usually available from a Fourier analysis, they represent a valuable alternative to Fourier analysis.

*This work was supported by AFOSR under grant AFOSR 86-0126, the NSF under grant DMS-8704169, the DOE under grant DE-FG02-90ER25086 and Los Alamos National Laboratory.
†Program in Applied mathematics, University of Colorado at Boulder, Boulder, CO
‡Computer Research Group (C-3), Los Alamos National Laboratory, Los Alamos, NM
§Center for Computational Mathematics, University of Colorado of Denver, Denver, CO
The linear Boltzmann equation, which is used to model neutral-particle transport, reduces to the following equation in slab geometry with isotropic scattering. The physical domain is a semi-infinite slab of width \( b - a \) in the \( x \) dimension. Although three dimensional, we assume the flux of particles is independent of the \( y \) and \( z \) coordinates. Thus, the Linear Boltzmann equation reduces to

\[
\mu \frac{\partial \psi}{\partial x} + \sigma_t \psi = \frac{\sigma_s}{2} \int_{-1}^{1} \psi(x, \mu') \, d\mu' + q(x, \mu),
\]

for \( x \in (a, b), \mu \in [-1, 1] \), where \( \psi(x, \mu) \) represents the flux of particles at position \( x \) traveling at an angle \( \theta \) from the \( x \)-axis (\( \mu = \cos(\theta) \)). Here \( \sigma_t \, dx \) represents the expected number of interactions (absorptive or scattering) that a particle will have in traveling a distance \( dx \). In a similar manner, \( \sigma_s \, dx \) represents the expected number of scattering interactions while \( \sigma_a = \sigma_t - \sigma_s \) represents the expected number of absorptive interactions. Finally, \( q(x, \mu) \) represents the particle source.

The solution is well defined if the flux of particles entering the slab is given as boundary conditions:

\[
\psi(a, \mu) = g_a(\mu), \quad \psi(b, -\mu) = g_b(\mu), \quad \mu \in (0, 1).
\]

The problem \((1.1)\) becomes difficult to solve in the optically dense or thick diffusion limit. Physically, the thick limit implies that the mean-free-path between collisions that is small compared to the width of the slab. If, in addition, the material allows very little absorption, the model resembles diffusion away from boundaries and sources. Mathematically, we have

\[
\sigma_t \to \infty, \quad \frac{\sigma_s}{\sigma_t} \to 1.
\]

Dividing \((1.1)\) by \( \sigma_t \) and taking the limits in \((1.3)\) yields

\[
\psi(x, \mu) = \frac{1}{2} \int_{-1}^{1} \psi(x, \mu') \, d\mu',
\]

which admits any \( \psi(x, \mu) \) that is independent of \( \mu \). Thus, in this limit \((1.1)\) is singularly perturbed with a large near null space. The problems near the thick diffusion limit are difficult to solve for traditional transport methods.

Numerical approximations of \((1.1)\) suffer several difficulties in this limit. Many difference schemes yield inaccurate solutions in this limit even for well behaved solutions. For a discussion of this issue see Larsen and Morel\([8]\). One difference scheme that behaves well in the thick limit is the Modified Linear Discontinuous Scheme (\(MILD)\)(\([8]\)). Not only does it give the proper behavior in the thick limit but it is very accurate. The algorithm described in this paper will address the solution of the transport equation with \(S_N\) discretization in angle (See Section 2) using \(MILD\) in space.

Even when the difference scheme behaves properly in the thick limit it may be difficult to solve the resulting discrete equations. The Diffusion Synthetic Acceleration (\(DSA\)
scheme was developed to address this difficulty (Alcouffe[2]). DSA is based upon the recognition that in the thick limit the transport of particles becomes diffusive in nature. The flux is nearly independent of direction except near the boundaries and sources. DSA involves solving a diffusion equation as a preconditioning for the transport equations (See Faber and Manteuffel[5]). In theory, it achieves a convergence factor bounded by approximately .23 per iteration independent of $\sigma_t$ or the cell width $h$.

In this paper we present a multigrid algorithm that is much faster than DSA and avoids some of the problems associated with DSA. In general, the DSA algorithm requires a certain degree of consistency between the $S_N$ spatial discretization scheme and the diffusion spatial discretization scheme. In recent work Larsen [6], Morel and Larsen [14] and Adams and Martin [1] have derived methods for finding consistent diffusion schemes. In [1] a scheme was developed in which the convergence factor goes to zero as $\sigma_t h$ gets large. However, DSA works because the transport model becomes diffusive in the thick limit. Such problems are usually amenable to multigrid techniques directly. Moreover, the difficulties associated with deriving and solving consistent DSA diffusion equations increase with higher dimensions.

The multigrid techniques described here have a natural generalization to higher dimensions and, as mentioned previously, can be used in conjunction with the multigrid algorithm for anisotropic problems described in Morel and Manteuffel[15]. While any generalization to higher dimensions is fraught with peril, we feel that the multigrid algorithm algorithms presented here can be successfully extended.

Although similar in spirit, our multigrid method is much more efficient than those of Nowak [16] [17], Nowak and Larsen [18], Barnett, Morel, and Harris [3] and Nowak, Morel and Harris [19]. These methods require an expensive relaxation and do not achieve a convergence factor that approaches zero in the thick diffusion limit.

In this paper, we concentrate on pure scattering, that is, no absorption, which is modeled by setting $\sigma_a = \sigma_t$. Convergence proofs for both the thick and thin limits are developed. The efficacy of the algorithm is demonstrated with numerical examples. The model with absorption, i. e., $\sigma_a < \sigma_t$, is addressed in a sequel to this paper, [12]. In this paper piecewise linear discontinuous elements are used for the spatial discretization. In [12] piecewise kinked-linear discontinuous elements are used in the thick cell limit. The proofs in this paper are used to develop asymptotic estimates for the algorithm in [12].

The remainder of this paper is organized as follows. In Section 2 we describe the discrete transport equation. We use an $S_N$ approximation in direction (angle), and modified piecewise linear discontinuous elements (MLD) for the spatial discretization. In Section 3 we develop a two-cell, red/black, block $\mu$-line relaxation. The fact that the scattering operator is rank-1 allows for a very efficient $\mu$-line relaxation. Letting $h_i$ be the width of cell $i$, we show

a) for max($\sigma_t h_{2i-1}, \sigma_t h_{2i}) \ll 1$ at two neighboring cells $2i-1$ and $2i$, the $\mu$-line relaxation leaves the error linear across the two cells up to order $O(\max(\sigma_t^2 h_{2i-1}^2, \sigma_t^2 h_{2i}^2))$. 

3
b) for \( \min_i (\sigma_i h_{2i-1}, \sigma_i h_{2i}) \gg 1 \), the \( \mu \)-line relaxation leaves the error linear across the two cells up to order \( O(\max_i (\frac{1}{\sigma_i h_{2i-1}}, \frac{1}{\sigma_i h_{2i}})) \).

In Section 4 we combine the relaxation with a multigrid algorithm and derive two-grid convergence rates, \( \rho \); 
\[ \begin{align*}
a) & \text{ for } \max_i (\sigma_i h_i) \ll 1, \quad \rho = O(\max_i (\sigma_i h_i)^2), \\
b) & \text{ for } \min_i (\sigma_i h_i) \gg 1, \quad \rho = O(\min_i (\frac{1}{\sigma_i h_i^2})). \\
\end{align*} \]

with a slowest rate of \( \rho \sim 0.01 \) occurring for \( \sigma_i h \sim 0.01 \). These rates make a single \( V(1,1) \) cycle nearly an exact solver, especially in the thick and thin limits. The cost of a \( V(1,1) \) cycle is on the same order as 4 sweeps across the computational grid. This represents a significant savings over DSA. Moreover, the algorithm was designed to be highly parallel. We have implemented the algorithm on the Thinking Machines Inc., CM-2 at Los Alamos National Labs. These results appear in [Manteuffel et al., [10], [11]].

In Section 5.2 we present numerical results for a DSA scheme for the MLD equations. Section 6 contains concluding remarks.

§2. The Discrete Model

The angle dependence in (1.1) is discretized by a \( S_N \) approximation (c.f. Lewis and Miller [9]), that is, we assume

\[
\psi(x, \mu) = \sum_{l=0}^{N-1} (2l + 1) \phi_l(x) p_l(\mu),
\]

where \( p_l(\mu) \) is the \( l^{th} \) Legendre polynomial and \( (2l + 1) \) is a normalization factor. If standard Galerkin formulation is used, a system of equations, for the \( \phi_l(x) \)'s, called the moment equations, is established. The moments may be found by

\[
\phi(x) = \frac{1}{2} \int_{-1}^{1} \psi_1(x, \mu^{'}) p_l(\mu^{'}) d\mu^{'} = \sum_{k=1}^{N} \omega_k \psi_1(x, \mu_k) p_l(\mu_k) d\mu^{'} ,
\]

where \( \omega_1, \ldots, \omega_N \) and \( \mu_1, \ldots, \mu_N \) are the Gauss quadrature weights and points respectively. The second equality is exact because of the assumption (2.1). Thus, finding \( \psi_j(x) = \psi(x, \mu_j) \) for \( j=1, \ldots, N \) is equivalent to finding the moments. A simple transformation
of the moment equations yields a system for the $\psi_j(x)$’s, called the flux equations, as follows

\[
(2.3) \quad \mu_j \frac{\partial \psi_j(x)}{\partial x} + \sigma_t \psi_j(x) = \sigma_s \sum_{k=1}^{N} \omega_k \psi_k(x) + q_j(x),
\]

for $j=1,\ldots,N$.

For our purposes here we choose to differentiate the positive and negative direction. Assume $N$ is even and let $n = N/2$, the Gauss quadrature points are symmetric about the origin and can be denoted as $-\mu_n < \ldots < -\mu_1 < 0 < \mu_1 < \ldots,\mu_n$. Likewise, the weight associated with $-\mu_j$ will equal the weight associated with $\mu_j$. We define

\[
(2.4a) \quad \psi_j^+(x) = \psi(x, \mu_j),
\]

\[
(2.4b) \quad \psi_j^-(x) = \psi(x, -\mu_j),
\]

for $j = 1, \ldots, n$. Equation (2.3) becomes the pair of equations

\[
(2.5a) \quad \mu_j \frac{\partial \psi_j^+(x)}{\partial x} + \sigma_t \psi_j^+(x) = \sigma_s \sum_{k=1}^{n} \omega_k (\psi_k^+(x) + \psi_k^-(x)) + q_j^+(x),
\]

\[
(2.5b) \quad \mu_j \frac{\partial \psi_j^-(x)}{\partial x} + \sigma_t \psi_j^-(x) = \sigma_s \sum_{k=1}^{n} \omega_k (\psi_k^+(x) + \psi_k^-(x)) + q_j^-(x),
\]

for $j = 1,\ldots, n$. The boundary conditions (1.2) become

\[
(2.6a) \quad \psi_j^+(a) = g_a(\mu_j),
\]

\[
(2.6b) \quad \psi_j^-(b) = g_b(\mu_j),
\]

for $j = 1,\ldots, n$.

The space dependence is now discretized by the Modified Linear Discontinuous difference scheme (Larsen and Morel[8]). The $MLD$ scheme can be derived by using finite element methods. We first describe the Linear Discontinuous difference scheme (LD)[8]. Consider a set of grid points given by

\[
(2.7) \quad a = x_{-\frac{1}{2}} < x_{\frac{1}{2}} < \ldots < x_{m+\frac{1}{2}} = b.
\]

These represent cell edges. The centers are given by

\[
(2.8) \quad x_i = \frac{1}{2}(x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}}) \quad \text{for} \quad i = 1,\ldots, m,
\]
and the cell width is \( h_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} \) and \( m \) is the number of cells. We start by forming piecewise linear trial spaces. However, a different space is used for \( \psi_j^+(x) \) than for \( \psi_j^-(x) \). For \( \psi_j^+(x) \) (for each \( j \)), we choose break points at

\[(2.9a) \quad x_{1 \over 2} < \hat{x}_i^+ < x_{3 \over 2} < ... < x_{i-1 \over 2} < \hat{x}_i^+ < x_{i+1 \over 2} < ... \]

with \( \hat{x}_i^+ = x_{i-\frac{1}{2}} + \epsilon_i \), while for \( \psi_j^-(x) \) (for each \( j \)) we choose break points at

\[(2.9b) \quad x_{1 \over 2} < \hat{x}_i^- < x_{3 \over 2} < ... < x_{i-1 \over 2} < \hat{x}_i^- < x_{i+1 \over 2} < ... \]

with \( \hat{x}_i^- = x_{i+\frac{1}{2}} - \epsilon_i \). Here \( \epsilon_i \) is small. Bases elements for the two spaces, which we denote as \( V_{h,+} \) and \( V_{h,-} \), are depicted in Figures 1 and 2 respectively.

![Figure 1](image1.png)

**Figure 1**

Bases for \( V_{h,+} \)

![Figure 2](image2.png)

**Figure 2**

Bases for \( V_{h,-} \)

We assume that \( \psi_j^+(x) \) and \( \psi_j^-(x) \) can be expressed as a linear combination of the bases elements in \( V_{h,+} \) and \( V_{h,-} \) respectively. We denote

\[(2.10a) \quad \psi_{i+\frac{1}{2},j}^+ = \psi_j^+(x_{i+\frac{1}{2}}), \quad \psi_{i+\frac{1}{2},j}^- = \psi_j^-(x_{i+\frac{1}{2}}), \]

\[(2.10b) \quad \hat{\psi}_{i,j}^+ = \psi_j^+(\hat{x}_i^+), \quad \hat{\psi}_{i,j}^- = \psi_j^-(\hat{x}_i^-), \]

We also denote

\[(2.10c) \quad \psi_{i,j}^+ = \psi_j^+(x_i), \quad \psi_{i,j}^- = \psi_j^-(x_i), \]
It is important to note that $\hat{\psi}_{i,j}^+$ and $\hat{\psi}_{i,j}^-$ can be expressed in terms of values of $\psi_{j}^{+}(x)$ and $\psi_{j}^{-}(x)$ at the cell edges and cell center. Since $\psi_{j}^{+}(x)$ and $\psi_{j}^{-}(x)$ are assumed to be linear, $\hat{\psi}_{i,j}^+$ and $\hat{\psi}_{i,j}^-$ can be found as an extrapolation from the center and edge. We have

$$\hat{\psi}_{i,j}^+ = 2\left( \frac{h_i - \epsilon_i}{h_i} \right) \psi_{i,j}^+ - \left( \frac{h_i - 2\epsilon_i}{h_i} \right) \psi_{i+\frac{1}{2},j}^+,$$

(2.11a)

$$\hat{\psi}_{i,j}^- = 2\left( \frac{h_i - \epsilon_i}{h_i} \right) \psi_{i,j}^- - \left( \frac{h_i - 2\epsilon_i}{h_i} \right) \psi_{i-\frac{1}{2},j}^-.$$

(2.11b)

The Linear Discontinuous (LD) difference Scheme is derived by applying a standard Galerkin formulation, then taking the limit as $\max(\epsilon_i) \to 0$. One way to construct the equations is to plug the trial space representation into (2.5a) and (2.5b) and integrate over each cell against the constant function and then against the linear function, $(x - x_i)$. This will yield a set of equations involving $\epsilon_i$. The final equations are found by setting $\epsilon_i=0$.

The Modified Linear Discontinuous Scheme (MLD) is found in a similar manner. As in the LD scheme, (2.5a) and (2.5b) are integrated over each cell against the constant function. Then after setting $\epsilon_i = 0$ we get the following equations

$$\mu_j (\psi_{i+\frac{1}{2},j}^+ - \psi_{i-\frac{1}{2},j}^-) + \sigma_i \left( \frac{\psi_{i,j}^+ + \psi_{i+\frac{1}{2},j}^+}{2} \right) h_i = \sigma_s \sum_{k=1}^{n} \omega_k (\hat{\psi}_{i,k}^+ + \psi_{i+\frac{1}{2},k}^+ + \hat{\psi}_{i,k}^- + \psi_{i-\frac{1}{2},k}^-) h_i + q_{ij}^+ h_i,$$

(2.12a)

$$\mu_j (\psi_{i+\frac{1}{2},j}^- - \psi_{i-\frac{1}{2},j}^+) + \sigma_i \left( \frac{\psi_{i,j}^- + \psi_{i-\frac{1}{2},j}^-}{2} \right) h_i = \sigma_s \sum_{k=1}^{n} \omega_k (\hat{\psi}_{i,k}^+ + \psi_{i+\frac{1}{2},k}^+ + \hat{\psi}_{i,k}^- + \psi_{i-\frac{1}{2},k}^-) h_i + q_{ij}^- h_i.$$

(2.12b)

These are referred to as the balance equations. Instead of integration against linear functions, as in LD, now the equations (2.5a) are collocated at $x_i^-$, equations (2.5b) are collocated at $x_i^+$, and the limit is taken as $\max_{i} \epsilon_i \to 0$. This yields the following equations referred to as edge equations.

$$\mu_j \left( \frac{\psi_{i+\frac{1}{2},j}^+ - \psi_{i,j}^+}{h_i} \right) + \sigma_i \psi_{i+\frac{1}{2},j} = \sigma_s \sum_{k=1}^{n} \omega_k (\psi_{i,k}^+ + \psi_{i+\frac{1}{2},k}^-) + q_{i+\frac{1}{2},j}^+,$$

(2.13a)

$$\mu_j \left( \frac{\psi_{i+\frac{1}{2},j}^- - \psi_{i-\frac{1}{2},j}^-}{h_i} \right) + \sigma_i \psi_{i-\frac{1}{2},j} = \sigma_s \sum_{k=1}^{n} \omega_k (\psi_{i,k}^- + \psi_{i-\frac{1}{2},k}^-) + q_{i-\frac{1}{2},j}^-,$$

(2.13b)

This may be formalized as integration of (2.5a) against $\delta(x - x_i^-$) and (2.5b) against $\delta(x - x_i^+)$. Note from (2.11a, b) that we may replace $\hat{\psi}_{i,k}^+$ and $\hat{\psi}_{i,k}^-$ with $2\psi_{i,k}^+ - \psi_{i+\frac{1}{2},k}^+$ and
\[ 2\psi^+_{i,k} - \psi^-_{i,\frac{k}{2}} \] respectively. The boundary conditions are

\begin{equation}
\frac{\psi^+}{\psi^-} = g^+_{i,j}, \quad \frac{\psi^-}{\psi^+} = g^-_{i,j}, \quad j = 1, 2, \ldots, n.
\end{equation}

We see from the edge equations and balance equations that it is the product of \(\sigma\) and \(h\) that will determine the property of the discretized equations. We will use \(\sigma h\) as a parameter in our discussion and set \(\gamma = \frac{\sigma h}{\sigma^2} = 1\).

In block matrix form, these equations become

\begin{equation}
B_i(\psi^+_{i,\frac{k}{2}} - \psi^-_{i,\frac{k}{2}}) + \psi^+ = R(\psi^+ + \psi^-) + g^+,
\end{equation}

\begin{equation}
2B_i(\psi^+_{i,\frac{k}{2}} - \psi^-_{i,\frac{k}{2}}) + \psi^+ = R(\psi^+ + 2\psi^- - \psi^-_{i,\frac{k}{2}}) + g^+,
\end{equation}

for \(i = 1, \ldots, m\), and

\begin{equation}
B_i(\psi^-_{i,\frac{k}{2}} - \psi^-_{i,\frac{k}{2}}) + \psi^- = R(\psi^+ + \psi^-) + g^-,
\end{equation}

\begin{equation}
2B_i(\psi^-_{i,\frac{k}{2}} - \psi^-_{i,\frac{k}{2}}) + \psi^- = R(\psi^- + 2\psi^+ - \psi^+_{i,\frac{k}{2}}) + g^-,
\end{equation}

for \(i = 1, \ldots, m\), with boundary conditions

\begin{equation}
\psi^+_{\frac{k}{2}} = g^+_i, \quad \psi^-_{m+\frac{k}{2}} = g^-_m,
\end{equation}

where \(B_i\) and \(R\) are defined as

\begin{equation}
B_i = \begin{bmatrix}
\frac{\mu}{\sigma h_i} & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & \frac{\mu}{\sigma h_i}
\end{bmatrix}, \quad R = \begin{bmatrix}
1 \\
\vdots \\
1\end{bmatrix} \begin{bmatrix}
\omega_1 & \cdots & \omega_n
\end{bmatrix} = 1 \omega^T,
\end{equation}

and

\begin{equation}
\psi^+_{i,\frac{k}{2}} = (\psi^+_{i,\frac{k}{2}}, \ldots, \psi^+_{i,\frac{k}{2}n})^T, \quad \psi^+ = (\psi^+_{i_1}, \ldots, \psi^+_{i_n})^T,
\end{equation}

\begin{equation}
\psi^-_{i,\frac{k}{2}} = (\psi^-_{i,\frac{k}{2}}, \ldots, \psi^-_{i,\frac{k}{2}n})^T, \quad \psi^- = (\psi^-_{i_1}, \ldots, \psi^-_{i_n})^T.
\end{equation}

We can write equation (2.15) in matrix form as

\begin{equation}
\begin{bmatrix}
A_1 & -C_1 & \cdots & 0 \\
-D_2 & A_2 & -C_2 & \cdots \\
\vdots & \ddots & \ddots & \ddots \\
-D_i & A_i & -C_i & \cdots \\
\cdots & \cdots & \ddots & \ddots \\
-D_m & A_m & \cdots & -D_m
\end{bmatrix} \begin{bmatrix}
\psi^-_1 \\
\psi^-_2 \\
\vdots \\
\psi^-_n
\end{bmatrix} = \begin{bmatrix}
Q_1 \\
Q_2 \\
\vdots \\
Q_n
\end{bmatrix},
\end{equation}

\[ 8 \]
where \( Q \) is the right-hand side and

\[
A_i = \begin{bmatrix}
I + 2B_i - R & -2R & -2B_i & R \\
0 & I - R & -R & B_i \\
B_i & -R & I - R & 0 \\
R & -2B_i & -2R & I + 2B_i - R
\end{bmatrix},
\]

\( i = 1, \ldots, m, \)

\[
C_i = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
B_i & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}, \quad i = 1, \ldots, m - 1,
\]

\[
D_i = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & B_i \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}, \quad i = 2, \ldots, m,
\]

\[
\Psi_i = (\hat{\psi}_i, \psi_i^+, \hat{\psi}_i, \psi_i^+)^T, \quad i = 1, \ldots, m.
\]

In the following section, we will examine the \( x \)-line relaxation, also know as a transport sweep or source iteration, and the \( \mu \)-line relaxation and their properties.

§3 Relaxation

Suppose the right-hand sides of (2.15a-d) were known. The positive angles would then satisfy a system of equations involving the matrix

\[
\begin{bmatrix}
I & B_1 \\
-2B_1 & I + 2B_1 \\
-B_2 & I \\
-2B_2 & I + 2B_2 \\
& & \ddots
\end{bmatrix}
\begin{bmatrix}
\psi^+_1 \\
\psi^+_2 \\
\psi^+_3 \\
\psi^+_4 \\
& \ddots
\end{bmatrix}
\]

\( \Psi_1 \)

The first block equation represents the boundary conditions. This block lower triangular matrix could be solved in a forward sweep by inverting the block matrices as follows:

\[
\begin{bmatrix}
I & B_i \\
-2B_i & I + 2B_i
\end{bmatrix}^{-1} \left( I + 2B_i + 2B_i^2 \right)^{-1} \begin{bmatrix}
I + 2B_i & -B_i \\
2B_i & I
\end{bmatrix}.
\]
Since $B_i$ is diagonal, this is a trivial operation. Likewise, the negative angles could be found in a backward sweep.

This process represents $x$-line relaxation (also known as a source iteration or transport sweep). If we reorder the unknowns by cells and examine the effect of a complete $x$-line relaxation, the resulting error equation for a single cell can be written as

\[
\begin{bmatrix}
I + 2B_i & 0 & -2B_i & 0 \\
0 & I & 0 & -B_i \\
-B_i & 0 & I & 0 \\
0 & -2B_i & 0 & I + 2B_i
\end{bmatrix}
\begin{bmatrix}
\xi_{i+\frac{1}{2}}^- \\
\xi_{i+\frac{1}{2}}^+ \\
\xi_{i-\frac{1}{2}}^+ \\
\xi_{i-\frac{1}{2}}^-
\end{bmatrix}^{i+1} =
\begin{bmatrix}
0 \\
0 \\
B_i \xi_{i+\frac{1}{2}}^- \\
0
\end{bmatrix}^{i+1}.
\]

Here $l$ is the relaxation step. When $\sigma_i h_i \gg 1$, $x$-line relaxation is not effective. Equation (2.16a) reveals that, when $\sigma_i h_i \rightarrow \infty$, $B_i$ approaches zero. Then, an $x$-line relaxation nearly reproduces the component of the error that is independent of angle. Since this does not affect any spatial frequency, the resulting error is not smooth in space. Any error that is independent of angle will not be suppressed regardless of its spatial frequency. This makes $x$-line relaxation inappropriate for use in a multigrid scheme, since even if $\sigma_i h_i$ is small on the finest grid, it will become large on coarser grids.

On the other hand, we can rearrange equation (2.17) to reveal the form of two-cell red-black block $\mu$-line relaxations. By two-cell red-black block $\mu$-line relaxation, we mean that at cells $i$ and $i + 1$, we set $\psi_{i+\frac{1}{2}}^+$ and $\psi_{i+\frac{1}{2}}^-$ as boundary values and solve all other interior values in cells $i$ and $i + 1$ simultaneously. This relaxation is carried out in a red-black ordering. Since the relaxation uses a red-black ordering, we can look at each 2-cell pair individually. For 2-cell pair $i$ and $i + 1$, the errors at these two cells after the relaxation will be

\[
\begin{bmatrix}
\xi_{i+\frac{1}{2}}^- \\
\xi_{i+\frac{1}{2}}^+ \\
\xi_{i-\frac{1}{2}}^+ \\
\xi_{i-\frac{1}{2}}^-
\end{bmatrix} =
\begin{bmatrix}
A_i & -C_i \\
-D_{i+1} & A_{i+1}
\end{bmatrix}^{-1}
\begin{bmatrix}
0 \\
B_i \xi_{i+\frac{1}{2}}^- \\
0 \\
0
\end{bmatrix}.
\]

The inversion of the

\[
\begin{bmatrix}
A_i & -C_i \\
-D_{i+1} & A_{i+1}
\end{bmatrix}
\]

is inexpensive since $R$ is rank one and $I$, $B_i$, and
\( B_{i+1} \) are diagonal matrices. To show this, we write matrix \[
\begin{bmatrix}
 A_i & -C_i \\
 -D_{i+1} & A_{i+1}
\end{bmatrix}
\]
as
\[
\begin{bmatrix}
 I + 2B_i & 0 & -2B_i & 0 \\
 0 & I & 0 & B_i \\
 B_i & 0 & I & 0 \\
 0 & -2B_i & 0 & I + 2B_i \\
-2B_{i+1} & 0 & I + 2B_i & -B_i \\
 B_{i+1} & 0 & I & 0 \\
 0 & -2B_{i+1} & 0 & I + 2B_{i+1}
\end{bmatrix}
\]

where \( \mathbf{1} = (1, 1, \ldots, 1)^T \) and \( \mathbf{w}^T = (\omega_1, \omega_2, \ldots, \omega_n) \). The first matrix in (3.5) can be easily inverted since it is sparse and its components \( I, B_i \), and \( B_{i+1} \) are diagonal matrices. The second term is a rank four matrix. If we denote (3.5) as
\[
\begin{bmatrix}
 A_i & -C_i \\
 -D_{i+1} & A_{i+1}
\end{bmatrix} = A_0 - V W^T,
\]
where \( A_0 \) is the first matrix in (3.5) and \( V, W^T \) are the two rank four matrices in (3.5), then the Sherman-Morrison formulas yield
\[
\begin{bmatrix}
 A_i & -C_i \\
 -D_{i+1} & A_{i+1}
\end{bmatrix}^{-1} = A_0^{-1} + A_0^{-1} V (I - W^T A_0^{-1} V)^{-1} W^T A_0^{-1}.
\]

Since \( A_0 \) can be easily inverted, the main thing is to invert \( (I - W^T A_0^{-1} V) \). The matrix \( (I - W^T A_0^{-1} V) \) is a \( 4 \times 4 \) matrix and is given by
\[
(I - W^T A_0^{-1} V) = \begin{bmatrix}
 a_{11} & 0 & a_{13} & a_{14} \\
 0 & a_{22} & a_{23} & a_{24} \\
 a_{31} & a_{32} & a_{33} & 0 \\
 a_{41} & a_{42} & 0 & a_{44}
\end{bmatrix},
\]

where
\[
a_{11} = a_{22} = \sum_{j=1}^{n} \frac{2\omega_j (\mu_j + 2 \frac{\mu_j^2}{\sigma_i h_i})}{\sigma_i h_i (1 + 2 \frac{\omega_j}{\sigma_i h_i} + 2 \frac{\mu_j^2}{\sigma_i h_i})},
\]

11
(3.9b) \[ a_{33} = a_{44} = \sum_{j=1}^{n} \frac{2\omega_j(\mu_j + 2\frac{\mu_j^2}{\sigma_{j\tilde{h}_{i+1}}})}{\sigma_j\tilde{h}_{i+1}(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})}, \]

(3.9c) \[ a_{13} = -\sum_{j=1}^{n} \frac{2\omega_j\frac{\mu_j^3}{\sigma_j^2\tilde{h}_{i+1}}}{(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})}, \]

(3.9d) \[ a_{14} = -\sum_{j=1}^{n} \frac{2\omega_j\mu_j^3}{\sigma_j^2\tilde{h}_{i+1}^2(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})}, \]

(3.9e) \[ a_{23} = -\sum_{j=1}^{n} \frac{2\omega_j(\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + \frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}} + \frac{\mu_j^3(1 + \frac{\mu_j}{\sigma_j\tilde{h}_{i+1}})}{\sigma_j^2\tilde{h}_{i+1}^2})}{(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})}, \]

(3.9f) \[ a_{24} = -\sum_{j=1}^{n} \frac{2\omega_j\frac{\mu_j^3}{\sigma_j^2\tilde{h}_{i+1}}(1 + \frac{\mu_j}{\sigma_j\tilde{h}_{i+1}})}{(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})}, \]

(3.9g) \[ a_{31} = -\sum_{j=1}^{n} \frac{2\omega_j(\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + \frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}} + \frac{\mu_j^3(1 + \frac{\mu_j}{\sigma_j\tilde{h}_{i+1}})}{\sigma_j^2\tilde{h}_{i+1}^2})}{(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})}, \]

(3.9h) \[ a_{32} = -\sum_{j=1}^{n} \frac{2\omega_j(\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + \frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}} + \frac{\mu_j^3(1 + \frac{\mu_j}{\sigma_j\tilde{h}_{i+1}})}{\sigma_j^2\tilde{h}_{i+1}^2})}{(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})}, \]

(3.9i) \[ a_{41} = -\sum_{j=1}^{n} \frac{2\omega_j\frac{\mu_j^3}{\sigma_j^2\tilde{h}_{i+1}}}{\sigma_j^2\tilde{h}_{i+1}^2(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})}, \]

(3.9j) \[ a_{42} = -\sum_{j=1}^{n} \frac{2\omega_j(\frac{\mu_j^3}{\sigma_j^2\tilde{h}_{i+1}})}{(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})(1 + 2\frac{\mu_j}{\sigma_j\tilde{h}_{i+1}} + 2\frac{\mu_j^2}{\sigma_j^2\tilde{h}_{i+1}^2})}, \]
Since \((I - W^TA_0^{-1}V)\) is a 4 by 4 matrix, its inversion can be obtained inexpensively and explicitly by a direct method. It is obvious that no matter how many moments are used in (2.1), the \(\mu\)-line relaxation can be carried out with the same insignificant expense.

The next three theorems examine the properties of two-cell red-black block \(\mu\)-line relaxation. In the first theorem we consider the \(S_2\) problem, that is, one positive and one negative \(\mu\). We assume the number of cells \(m\) is an even number.

**Theorem 1** For \(S_2\), the errors after two-cell red-black \(\mu\)-line relaxation will be piecewise linear across two cells.

Proof: Since we use red-black relaxation, we can just examine each pair of two cells individually. When there is only one angle in the positive direction and one in the negative direction, \(R = \frac{1}{2}; B_k = \frac{\mu_k}{\sigma_{2k}}, k = 1, ..., m\). At cells \(2i - 1\) and \(2i\), after a relaxation, the error equation (3.4) becomes

\[
\begin{bmatrix}
e^{+}_{2i-\frac{1}{2}} \\
e^{-}_{2i-1} \\
e^{+}_{2i-1} \\
e^{-}_{2i-\frac{1}{2}} \\
e^{+}_{2i} \\
e^{-}_{2i+\frac{1}{2}}
\end{bmatrix} = \begin{bmatrix}
A_{2i-1} & -C_{2i-1} \\
-D_{2i} & A_{2i}
\end{bmatrix}^{-1} \begin{bmatrix}
0 \\
B_{2i-1}e^{+}_{2i-\frac{1}{2}} \\
0 \\
0 \\
0 \\
B_{2i}e^{+}_{2i+\frac{1}{2}}
\end{bmatrix}
\]

(3.10)

By setting \(n = 1\), the explicit form of \(\begin{bmatrix} A_{2i-1} & -C_{2i-1} \\ -D_{2i} & A_{2i} \end{bmatrix}^{-1}\) can be easily obtained by Sherman-Morrison formula (3.7). After multiplication in (3.10), we have

\[
\begin{bmatrix}
e^{+}_{2i-\frac{1}{2}} \\
e^{-}_{2i-1} \\
e^{+}_{2i-1} \\
e^{-}_{2i-\frac{1}{2}} \\
e^{+}_{2i} \\
e^{-}_{2i+\frac{1}{2}}
\end{bmatrix} = \begin{bmatrix}
1 + \frac{h_{2i-1}}{h_{2i}} \\
1 + \frac{h_{2i-1}}{h_{2i}} + \frac{\mu_1}{\sigma_{2h_{2i}}} \\
1 + \frac{h_{2i-1}}{h_{2i}} + \frac{2\mu_1}{\sigma_{2h_{2i}}} \\
1 + \frac{h_{2i-1}}{h_{2i}} \frac{1}{\sigma_{2h_{2i}}} \\
\frac{1}{2} + \frac{\mu_1}{\sigma_{2h_{2i}}} \\
\frac{1}{2} + \frac{\mu_1}{\sigma_{2h_{2i}}}
\end{bmatrix} \begin{bmatrix}
\sigma_{1}h_{2i} \\
2\mu_1 + \sigma_{1}h_{2i-1} + \sigma_{1}h_{2i} \\
1 + \frac{h_{2i-1}}{2h_{2i}} + \frac{\mu_1}{\sigma_{2h_{2i}}} \\
1 + \frac{h_{2i-1}}{2h_{2i}} + \frac{2\mu_1}{\sigma_{2h_{2i}}} \\
\frac{1}{2} + \frac{2\mu_1}{\sigma_{2h_{2i}}} \\
\frac{1}{2} + \frac{2\mu_1}{\sigma_{2h_{2i}}}
\end{bmatrix} \begin{bmatrix}
e^{+}_{2i-\frac{1}{2}} \\
e^{-}_{2i-1} \\
e^{+}_{2i-1} \\
e^{-}_{2i-\frac{1}{2}} \\
e^{+}_{2i} \\
e^{-}_{2i+\frac{1}{2}}
\end{bmatrix}
\]

(3.11)
The errors fit the following linear formulas:

\[(\text{3.12a})\]

\[
\varepsilon_k^- = \frac{\sigma_i h_{2i}}{2 \mu_1 + \sigma_i h_{2i-1} + \sigma_i h_{2i}} \left[ \frac{2 \mu_1}{\sigma_i h_{2i-1}} \right] + \frac{1}{2} + \frac{2 \mu_1}{\sigma_i h_{2i}} \left[ \frac{1}{1 + \frac{\sigma_i h_{2i-1}}{2h_{2i-1}}} \right] + \frac{\xi_k}{2} \varepsilon_{2i}^- + \frac{\xi_k}{2} \varepsilon_{2i+1}^- .
\]

where \( k = 2i - \frac{3}{2}, 2i - 1, 2i - \frac{1}{2}, 2i \), and \( \xi_k = 0, \frac{h_{2i-1}}{2}, h_{2i-1}, h_{2i-1} + \frac{h_{2i}}{2} \) correspondingly and

\[(\text{3.12b})\]

\[
\varepsilon_k^+ = \frac{\sigma_i h_{2i}}{2 \mu_1 + \sigma_i h_{2i-1} + \sigma_i h_{2i}} \left[ \frac{2 \mu_1}{\sigma_i h_{2i-1}} \right] + \frac{1}{2} + \frac{2 \mu_1}{\sigma_i h_{2i}} \left[ \frac{1}{1 + \frac{\sigma_i h_{2i-1}}{2h_{2i-1}}} \right] + \frac{\xi_k}{2} \varepsilon_{2i}^- + \frac{\xi_k}{2} \varepsilon_{2i+1}^- ,
\]

where \( k = 2i - 1, 2i - \frac{1}{2}, 2i, 2i + \frac{1}{2}, \) and \( \xi_k = \frac{h_{2i-1}}{2}, h_{2i-1}, h_{2i-1} + \frac{h_{2i}}{2}, h_{2i-1} + h_{2i} \) correspondingly. Since after relaxation the errors fit linear formulas at both the positive and negative angle direction, the errors are linearly distributed across two cells. \( \square \)

Let us define a new cell \( i \) on a coarser grid \( \hat{h} \) whose length \( \hat{h} \) is \( \hat{h} = h_{2i-1} + h_{2i} \) and it covers cells \( 2i - 1 \) and cell \( 2i \). We can find values at the cell center and the edges of cell \( i \) of grid \( \hat{h} \) by letting \( \xi_k = 0, \xi_k = \frac{h_{2i-1} + h_{2i}}{2} \) and \( \xi_k = h_{2i-1} + h_{2i} \) in (3.12). Thus, we have

\[(\text{3.13a})\]

\[
\varepsilon_{-\frac{1}{2}} \hat{h}^- = \frac{\sigma_i h_{2i}}{2 \mu_1 + \sigma_i h_{2i-1} + \sigma_i h_{2i}} \left[ \frac{1}{2} + \frac{2 \mu_1}{\sigma_i h_{2i-1}} \right] + \frac{h_{2i-1}}{2} \varepsilon_{2i-\frac{1}{2}}^- + \frac{h_{2i-1}}{2} \varepsilon_{2i+\frac{1}{2}}^- ,
\]

\[(\text{3.13b})\]

\[
\varepsilon_{\frac{1}{2}} \hat{h}^+ = \frac{\sigma_i h_{2i}}{2 \mu_1 + \sigma_i h_{2i-1} + \sigma_i h_{2i}} \left[ \frac{1}{2} + \frac{2 \mu_1}{\sigma_i h_{2i-1}} \right] + \frac{h_{2i-1}}{2} \varepsilon_{2i-\frac{1}{2}}^+ + \frac{h_{2i-1}}{2} \varepsilon_{2i+\frac{1}{2}}^- ,
\]

\[(\text{3.13c})\]

\[
\varepsilon_{-\frac{1}{2}} \hat{h}^- = \frac{\sigma_i h_{2i}}{2 \mu_1 + \sigma_i h_{2i-1} + \sigma_i h_{2i}} \left[ \frac{1}{2} + \frac{2 \mu_1}{\sigma_i h_{2i-1}} \right] + \frac{h_{2i-1}}{2} \varepsilon_{2i-\frac{1}{2}}^- + \frac{h_{2i-1}}{2} \varepsilon_{2i+\frac{1}{2}}^- ,
\]

\[(\text{3.13d})\]

\[
\varepsilon_{\frac{1}{2}} \hat{h}^+ = \frac{\sigma_i h_{2i}}{2 \mu_1 + \sigma_i h_{2i-1} + \sigma_i h_{2i}} \left[ \frac{1}{2} + \frac{2 \mu_1}{\sigma_i h_{2i-1}} \right] + \frac{h_{2i-1}}{2} \varepsilon_{2i-\frac{1}{2}}^+ + \frac{h_{2i-1}}{2} \varepsilon_{2i+\frac{1}{2}}^- ,
\]
We can then write (3.11) in terms of $e_i^{-(\hat{h})}, e_i^{+(\hat{h})}, e_i^{-(\frac{h}{2})}, e_i^{+(\frac{h}{2})}$ as

$$
\begin{bmatrix}
e_i^{-(\frac{h}{2})} \\
e_i^{+(\frac{h}{2})} \\
e_i^{-(h)} \\
e_i^{+(h)}
\end{bmatrix} =
\begin{bmatrix}
I & 0 & 0 & 0 \\
0 & \frac{2h_i + h_{i+1}}{h_i + h_{i+1}} I & \frac{h_{i+1}}{h_i + h_{i+1}} I & 0 \\
0 & \frac{h_{i+1}}{h_i + h_{i+1}} I & \frac{h_i}{h_i + h_{i+1}} I & \frac{h_i - h_{i+1}}{h_i + h_{i+1}} I \\
0 & \frac{h_i - h_{i+1}}{h_i + h_{i+1}} I & \frac{h_{i+1}}{h_i + h_{i+1}} I & 0 \\
0 & 0 & 0 & I
\end{bmatrix}
\begin{bmatrix}
e_i^{-(\frac{h}{2})} \\
e_i^{+(\frac{h}{2})} \\
e_i^{-(h)} \\
e_i^{+(h)}
\end{bmatrix}.
$$

Let

$$
T_{1,i} =
\begin{bmatrix}
I & 0 & 0 & 0 \\
0 & \frac{h_{i+1} + h_i}{h_{i+1} + h_i} I & 0 & \frac{-h_{i+1}}{h_{i+1} + h_i} I \\
0 & \frac{h_i}{h_{i+1} + h_i} I & 0 & \frac{h_i - h_{i+1}}{h_{i+1} + h_i} I \\
0 & \frac{h_i - h_{i+1}}{h_{i+1} + h_i} I & 0 & \frac{h_{i+1}}{h_i + h_{i+1}} I \\
0 & 0 & 0 & I
\end{bmatrix},
$$

$$
T_{2,i} =
\begin{bmatrix}
h_{i+1} - h_i \frac{h_{i+1}}{h_{i+1} + h_i} I & 0 & 2h_i \frac{h_{i+1}}{h_{i+1} + h_i} I & 0 \\
0 & \frac{h_i}{h_{i+1} + h_i} I & 0 & \frac{h_i - h_{i+1}}{h_{i+1} + h_i} I \\
0 & \frac{h_i - h_{i+1}}{h_{i+1} + h_i} I & 0 & \frac{h_{i+1}}{h_i + h_{i+1}} I \\
0 & 0 & 0 & I
\end{bmatrix}.
$$

Then after the relaxation, the errors can be written as

$$
\begin{bmatrix}
e_i^{-(\frac{h}{2})} \\
e_i^{+(\frac{h}{2})} \\
e_i^{-(h)} \\
e_i^{+(h)}
\end{bmatrix} =
\begin{bmatrix}
T_{1,1} & T_{1,3} \\
T_{2,1} & T_{2,3} \\
\vdots & \ddots \\
\vdots & \ddots & T_{1,m-1} & T_{2,m-1}
\end{bmatrix}
\begin{bmatrix}
e_i^{-(\frac{h}{2})} \\
e_i^{+(\frac{h}{2})} \\
e_i^{-(h)} \\
e_i^{+(h)}
\end{bmatrix}.
$$

We write (3.16) in multigrid notation as

$$(3.17)
\varepsilon_h = I_{2h}^h \varepsilon_{2h},
$$
where

\[
I_{2h}^k = \begin{bmatrix}
T_{1,1} & T_{1,3} & \cdots & T_{1,m-1} \\
T_{2,1} & T_{2,3} & \cdots & T_{2,m-1}
\end{bmatrix}.
\]

We use the notation \(2h\) to indicate a coarse grid although our grid is not uniform. In the nonuniform case, the mesh size \(h_i\) at cell \(i\) on the coarse grid is \(h_{2i-1} + h_{2i}\) of the fine grid instead of being twice as much as that in fine grid. In the next section, we will see that (3.16) means the errors after relaxation are in the range of interpolation.

In the multi-angle case, the errors will no longer be exactly linearly distributed across two cells after two-cell red-black block \(\mu\)-line relaxation. But, we can examine the property of the relaxation in two extreme cases. These are when \(\max_{k=(1,\ldots,m)} (\sigma_i h_k) \ll 1\) and \(\min_{k=(1,\ldots,m)} (\sigma_i h_k) \gg 1\). The next two theorems will give Taylor expansion forms of the errors after the relaxation.

**Theorem 2** When \(\max(\sigma_i h_{2i-1}, \sigma_i h_{2i}) \ll 1\), the errors after two-cell red-black block \(\mu\)-line relaxation will be piecewise linear across two cells up to the accuracy of \(O(\max(\sigma_i^2 h_{2i-1}^2, \sigma_i^2 h_{2i}^2))\).

**Proof:** We can write

\[
\begin{bmatrix}
A_{2i-1} & -C_{2i-1} \\
-D_{2i} & A_{2i}
\end{bmatrix}
\]

in (3.10) as

\[
\begin{bmatrix}
A_{2i-1} & -C_{2i-1} \\
-D_{2i} & A_{2i}
\end{bmatrix} = H_0 + H_1
\]

with

\[
H_0 = \begin{bmatrix}
\frac{2}{\sigma_i h_{2i-1}^2} M & 0 & \frac{2}{\sigma_i h_{2i-1}^2} M & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{\sigma_i h_{2i-1}^2} M & 0 & \frac{1}{\sigma_i h_{2i-1}^2} M & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{\sigma_i h_{2i-1}^2} M & 0 & \frac{1}{\sigma_i h_{2i-1}^2} M & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{\sigma_i h_{2i-1}^2} M & 0 & \frac{1}{\sigma_i h_{2i-1}^2} M & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{\sigma_i h_{2i-1}^2} M & 0 & \frac{1}{\sigma_i h_{2i-1}^2} M & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{\sigma_i h_{2i-1}^2} M & 0 & \frac{1}{\sigma_i h_{2i-1}^2} M
\end{bmatrix},
\]

16
\( H_1 = \begin{bmatrix} I - R & -2R & 0 & 0 & 0 & 0 \\ 0 & I - R & -R & 0 & 0 & 0 \\ 0 & -R & I - R & 0 & 0 & 0 \\ R & 0 & -2R & I - R & 0 & 0 \\ 0 & 0 & 0 & 0 & I - R & -2R \\ 0 & 0 & 0 & 0 & 0 & R \end{bmatrix}^{-1}, \)

and

\( M = \sigma_t h_i B_i = \begin{bmatrix} \mu_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mu_n \end{bmatrix}. \)

Then

\( \begin{bmatrix} A_{2i-1} & -C_{2i-1} \\ -D_{2i} & A_{2i} \end{bmatrix}^{-1} = (I + H_0^{-1}H_1)^{-1}H_0^{-1}. \)

It is trivial to obtain

\( H_0^{-1} = \begin{bmatrix} 0 & 0 & \sigma_t h_{2i-1} M^{-1} & 0 & 0 & \sigma_t h_{2i} M^{-1} \\ 0 & \sigma_t h_{2i-1} M^{-1} & 0 & \sigma_t h_{2i-1} M^{-1} & 0 & 0 \\ \sigma_t h_{2i-1} M^{-1} & 0 & \sigma_t h_{2i-1} M^{-1} & 0 & \sigma_t h_{2i} M^{-1} \end{bmatrix}^{-1}. \)

When \( \sigma_t h_{2i-1}, \sigma_t h_{2i} \ll 1, \|H_0^{-1}\| \ll 1. \) Since \( H_1 \) is a constant matrix, \( \|H_0^{-1}H_1\| \ll 1. \) So \( I + H_0^{-1}H_1 \) is invertible and

\( (I + H_0^{-1}H_1)^{-1} = I - H_0^{-1}H_1 + O(\max(\sigma_t^2 h_{2i-1}^2, \sigma_t^3 h_{2i}^3)). \)

Substituting (3.23) into (3.21), we have

\( \begin{bmatrix} A_{2i-1} & -C_{2i-1} \\ -D_{2i} & A_{2i} \end{bmatrix}^{-1} = H_0^{-1} - H_0^{-1}H_1H_0^{-1} + O(\max(\sigma_t^3 h_{2i-1}^3, \sigma_t^3 h_{2i}^3)). \)

When we carry out the arithmetic in (3.24) by using (3.20b) and (3.22) and substitute the
resulting formula into (3.10), we have the errors after relaxation as

(3.25) \[
\begin{bmatrix}
\frac{e_{2i-\frac{1}{2}}^-}{h_{2i-1}} \\
\frac{e_{2i-\frac{1}{2}}^+}{h_{2i-1}} \\
\frac{e_{2i-\frac{1}{2}}^+}{h_{2i-1}} \\
\frac{e_{2i+\frac{1}{2}}^-}{h_{2i-1}} \\
\frac{\sigma_t h_{2i}}{2}
\end{bmatrix}
= \begin{bmatrix}
I \\
0 \\
I \\
0 \\
0
\end{bmatrix}
\begin{bmatrix}
\frac{e_{2i-\frac{1}{2}}^+}{h_{2i-1}} + \sigma_t h_{2i-1} \\
\frac{1}{2}M^{-1}(I - R) \\
\frac{1}{2}M^{-1}R \\
\frac{1}{2}M^{-1}(I - R) \\
\frac{1}{2}M^{-1}R
\end{bmatrix}
+ \begin{bmatrix}
0 \\
I \\
0 \\
I \\
0
\end{bmatrix}
\begin{bmatrix}
-(1 + \frac{h_{2i-1}}{2h_{2i-1}})M^{-1}R \\
\frac{h_{2i-1}}{h_{2i}}M^{-1}(I - R) \\
-\frac{h_{2i-1}}{h_{2i}}M^{-1}(I - R) \\
\frac{1}{2}M^{-1}R \\
(1 + \frac{h_{2i-1}}{2h_{2i-1}})M^{-1}(I - R)
\end{bmatrix}
\begin{bmatrix}
e_{2i-\frac{1}{2}}^- \\
e_{2i+\frac{1}{2}}^- + O(max(\sigma_t^2 h_{2i-1}^2, \sigma_t^2 h_{2i}^2))
\end{bmatrix}
\]

When we define the same cell \( i \) on grid \( \hat{h} \) as in Theorem 1, we have

(3.26a) \[
\frac{e_{2i-\frac{1}{2}}^-}{h_{2i-1}} = (1 - \sigma_t h_{2i-1}(1 + \frac{h_{2i}}{h_{2i-1}}))M^{-1}(I - R)e_{2i-\frac{1}{2}}^+ + \sigma_t h_{2i}(1 + \frac{h_{2i-1}}{h_{2i}})M^{-1}R e_{2i+\frac{1}{2}}^-,
\]

(3.26b) \[
e_{2i+\frac{1}{2}}^+ = \sigma_t h_{2i-1}(\frac{1}{2} + \frac{h_{2i}}{2h_{2i-1}})M^{-1}R e_{2i-\frac{1}{2}}^- + (1 - \sigma_t h_{2i}(\frac{1}{2} + \frac{h_{2i-1}}{2h_{2i}}))M^{-1}(I - R)e_{2i+\frac{1}{2}}^-,
\]

(3.26c) \[
\frac{e_{2i-\frac{1}{2}}^-}{h_{2i-1}} = (1 - \sigma_t h_{2i-1}(\frac{1}{2} + \frac{h_{2i}}{2h_{2i-1}}))M^{-1}(I - R)e_{2i-\frac{1}{2}}^+ + \sigma_t h_{2i}(\frac{1}{2} + \frac{2h_{2i-1}}{h_{2i}})M^{-1}R e_{2i+\frac{1}{2}}^-,
\]

(3.26d) \[
\frac{e_{2i+\frac{1}{2}}^-}{h_{2i-1}} = \sigma_t h_{2i-1}(1 + \frac{h_{2i}}{h_{2i-1}})M^{-1}R e_{2i-\frac{1}{2}}^- + (1 - \sigma_t h_{2i}(1 + \frac{h_{2i-1}}{2h_{2i}}))M^{-1}(I - R)e_{2i+\frac{1}{2}}^-.
\]
We can write (3.25) in terms of \( e^{-(\frac{1}{2})}, e^{\frac{1}{2}}, e^{-(\frac{3}{2})}, e^{\frac{3}{2}} \) as

\[
\begin{bmatrix}
    e^{-(\frac{1}{2})}
    e^{\frac{1}{2}}
    e^{-(\frac{3}{2})}
    e^{\frac{3}{2}}
\end{bmatrix}
= \begin{bmatrix}
    T_{1,2i-1}
    T_{2,2i-1}
\end{bmatrix}
\begin{bmatrix}
    e^{-(\frac{1}{2})}
    e^{\frac{1}{2}}
    e^{-(\frac{3}{2})}
    e^{\frac{3}{2}}
\end{bmatrix}
+ O(max(\sigma_i^2 h^2_{2i-1}, \sigma_i^2 h^2_{2i})).
\]

Thus, the errors are piecewise linear across the two cells \(2i-1\) and \(2i\) up to the accuracy of \(O(max(\sigma_i^2 h^2_{2i-1}, \sigma_i^2 h^2_{2i}))\). \(\Box\)

We now focus our attention on the optically dense or thick limit.

**Lemma 1.** Suppose \(\sigma_i h_i \gg 1\); then, the asymptotic inversion of \(A_i\) defined by (2.18a) can be expanded as

\[
A_i^{-1} = \frac{\sigma_i h_i}{2c_0} \begin{bmatrix}
    R & 2R & 0 & -R \\
    0 & R & R & 0 \\
    0 & R & R & 0 \\
    -R & 0 & 2R & R
\end{bmatrix}
+ I
\]

\[
+ \frac{1}{2c_0} \begin{bmatrix}
    -2(MR + RM) & -2(MR + RM) & 2(MR + RM) & 2MR \\
    -2(MR - RM) & 0 & -2MR & -2MR \\
    -2(MR + RM) & 0 & 2MR & 2MR \\
    2MR & 0 & 2MR + RM & -2(MR + RM)
\end{bmatrix}
+ O\left(\frac{1}{\sigma_i h_i}\right).
\]

**Proof:** Write

\[
A_i = \begin{bmatrix}
    I + 2B_i - R & -2R & -2B_i & R \\
    0 & I - R & -R & B_i \\
    B_i & -R & I - R & 0 \\
    R & -2B_i & -2R & I + 2B_i - R
\end{bmatrix}
\]

\[
= \begin{bmatrix}
    I + 2B_i & 0 & -2B_i & 0 \\
    0 & I & 0 & B_i \\
    B_i & 0 & I & 0 \\
    0 & -2B_i & 0 & I + 2B_i
\end{bmatrix}
- \begin{bmatrix}
    1 & 0 & 0 & 0 \\
    1 & 1 & 1 & 1 \\
    1 & 1 & 1 & 1 \\
    0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
    \frac{1}{2}w^T & \frac{1}{2}w^T & \frac{1}{2}w^T & \frac{1}{2}w^T \\
    \frac{1}{2}w^T & \frac{1}{2}w^T & \frac{1}{2}w^T & \frac{1}{2}w^T \\
    \frac{1}{2}w^T & \frac{1}{2}w^T & \frac{1}{2}w^T & \frac{1}{2}w^T \\
    \frac{1}{2}w^T & \frac{1}{2}w^T & \frac{1}{2}w^T & \frac{1}{2}w^T
\end{bmatrix},
\]

where \(1 = (1, 1, ..., 1)^T\) and \(w^T = (\omega_1, \omega_2, ..., \omega_n)\). The first term is easily inverted while the second term is a rank two matrix. If we denote this as

\[
A_i = A_0 - VW^T,
\]

(3.30)
with \( A_0 \) denoting the first term and \( VW^T \) the second. Then the Sherman-Morrison formulas yield

\begin{equation}
A^{-1} = A_0^{-1} + A_0^{-1}V(I - W^TA_0^{-1}V)^{-1}W^TA_0^{-1}.
\end{equation}

\( A_0^{-1} \) is readily given by

\begin{equation}
A_0 = (I + 2B_i + 2B_i^2)^{-1} \begin{bmatrix} I & 0 & 2B_i & 0 \\
0 & I + 2B_i & 0 & -B_i \\
-B_i & 0 & I + 2B_i & 0 \\
0 & 2B_i & 0 & I \\
\end{bmatrix}.
\end{equation}

The matrix \( (I - W^TA_0^{-1}V) \) is \( 2 \times 2 \) and its inverse can be obtained as

\begin{equation}
(I - W^TA_0^{-1}V)^{-1} = \xi_i \begin{bmatrix} 1 & 0 \\
0 & 1 \\
\end{bmatrix},
\end{equation}

where

\begin{equation}
\xi_i = \frac{1}{1 - \zeta_i}, \quad \zeta_i = \sum_{j=1}^{N} \frac{2\omega_j(1 + \sigma_j)}{1 + 2\frac{\sigma_j}{\sigma_i} + 2(\frac{\sigma_j}{\sigma_i})^2}.
\end{equation}

In the optically dense media case, we expand \( \xi_i \) as

\begin{equation}
\xi_i = \frac{\sigma_i h_i}{2c_0} + O\left(\frac{1}{\sigma_i h_i}\right),
\end{equation}

and

\begin{equation}
(I + 2B_i + 2B_i^2)^{-1} = I - \frac{2}{\sigma_i h_i}M + \frac{2}{\sigma_i^2 h_i^2}M^2 + O\left(\frac{1}{\sigma_i^3 h_i^3}\right).
\end{equation}

Substituting (3.32), (3.33), (3.35) and (3.36) into (3.31), we have (3.28). \( \square \)

**Theorem 3.** Suppose \( \min_{k=(2i-1,2i)} (\sigma_i h_k) \gg 1 \); then, the errors after the two cell red-black \( \mu \)-line relaxation will be piecewise linear up to the accuracy of \( O(\max\left(\frac{1}{\sigma_i h_{2i-1}}, \frac{1}{\sigma_i h_{2i}}\right)) \).

**Proof:** When \( \min_{k=(2i-1,2i)} (\sigma_i h_k) \gg 1 \), we need to expand \( \begin{bmatrix} A_{2i-1} & -C_{2i-1} \\
-D_{2i} & A_{2i} \\
\end{bmatrix}^{-1} \). Write

\begin{equation}
\begin{bmatrix} A_{2i-1} & -C_{2i-1} \\
-D_{2i} & A_{2i} \\
\end{bmatrix} = \begin{bmatrix} A_{2i-1} & 0 \\
-D_{2i} & A_{2i} \\
\end{bmatrix} - \begin{bmatrix} 0 & C_{2i-1} \\
0 & 0 \\
\end{bmatrix} = A_0 - VW^T,
\end{equation}

where

\begin{equation}
VW^T = \begin{bmatrix} 0 & C_{2i-1} \\
0 & 0 \\
\end{bmatrix},
\end{equation}

\( \begin{bmatrix} A_{2i-1} & -C_{2i-1} \\
-D_{2i} & A_{2i} \\
\end{bmatrix}^{-1} = A_0^{-1} + A_0^{-1}V(I - W^TA_0^{-1}V)^{-1}W^TA_0^{-1}.
\end{equation}
with \( V = (0, 0, I, 0, 0, 0, 0, 0) \) and \( W^T = (0, 0, 0, 0, B_{2i-1}, 0, 0, 0) \). Its inversion can be obtained by Sherman-Morrison formula (3.31). We first note that

\[
(3.39) \\
A_0^{-1} = \begin{bmatrix}
A_{2i-1}^{-1} & 0 \\
A_{2i-1}^{-1} D_{2i} A_{2i}^{-1} & A_{2i}^{-1}
\end{bmatrix}.
\]

We can expand \( A_0^{-1} \) by using Lemma 1. By substituting the expansion of \( A_0^{-1} \) into \( (I - W^T A_0^{-1} V) \), we can then expand \( (I - W^T A_0^{-1} V)^{-1} \). The explicit expansion

\[
\begin{bmatrix}
A_{2i-1}^{-1} & -C_{2i-1} \\
-D_{2i} & A_{2i}
\end{bmatrix}^{-1}
\]

is omitted here because of its complexity. After the expansion, the error equation (3.10) at cells \( 2i-1 \) and \( 2i \) after two-cell \( \mu \)-line relaxation can be expanded as

\[
(3.40)
\]

\[
\left[
\begin{array}{c}
e_{2i-\frac{1}{2}}^-
e_{2i+\frac{1}{2}}^- \\
\varepsilon_{2i-1}^-
\varepsilon_{2i}^-
e_{2i+\frac{1}{2}}^-
\varepsilon_{2i}^+
\varepsilon_{2i+\frac{1}{2}}^-
\varepsilon_{2i}^+
\varepsilon_{2i+\frac{1}{2}}^-
\end{array}
\right]
= \begin{bmatrix}
\frac{1}{2} R & \frac{h_{2i-1}}{2(h_{2i-1}+h_{2i})} R \\
\left(\frac{1}{2} + \frac{1}{2(h_{2i-1}+h_{2i})} R\right) & \frac{h_{2i-1}}{2(h_{2i-1}+h_{2i})} R
\end{bmatrix}
\left[
\begin{array}{c}
\varepsilon_{2i-\frac{1}{2}}^+ \\
\varepsilon_{2i-1}^+ \\
\varepsilon_{2i}^+ \\
\varepsilon_{2i+\frac{1}{2}}^+ \\
\varepsilon_{2i}^+ \\
\varepsilon_{2i+\frac{1}{2}}^+
\end{array}
\right]
+ \begin{bmatrix}
0 \\
\frac{h_{2i-1}}{2(h_{2i-1}+h_{2i})} R \\
\frac{h_{2i-1}}{2(h_{2i-1}+h_{2i})} R
\end{bmatrix}
\begin{bmatrix}
e_{2i-\frac{1}{2}}^+ \\
\varepsilon_{2i-1}^+ \\
\varepsilon_{2i}^+ \\
\varepsilon_{2i+\frac{1}{2}}^+-O\left(\max\left(\frac{1}{\sigma_i h_{2i-1}}, \frac{1}{\sigma_i h_{2i}}\right)\right)\end{bmatrix}.
\]

As in Theorem 1 and 2 we define the same cell \( i \) on grid \( \hat{h} \) and obtain

\[
(3.41a) \\
\varepsilon_{2i-\frac{1}{2}}^-(\hat{h}) = R\varepsilon_{2i-\frac{1}{2}}^+;
\]

\[
(3.41b) \\
\varepsilon_{2i}^+(\hat{h}) = \frac{1}{2} R\varepsilon_{2i-\frac{1}{2}}^- + \frac{1}{2} R\varepsilon_{2i+\frac{1}{2}}^-;
\]

\[
(3.41c) \\
\varepsilon_{2i}^-(\hat{h}) = \frac{1}{2} R\varepsilon_{2i-\frac{1}{2}}^- + \frac{1}{2} R\varepsilon_{2i+\frac{1}{2}}^-;
\]

\[
(3.41d) \\
\varepsilon_{2i+\frac{1}{2}}^+(\hat{h}) = R\varepsilon_{2i+\frac{1}{2}}^-;
\]

Then (3.40) can be written as

\[
(3.42) \\
\left[
\begin{array}{c}
\varepsilon_{2i-\frac{1}{2}}^- \\
\varepsilon_{2i-1}^- \\
\varepsilon_{2i}^- \\
\varepsilon_{2i+\frac{1}{2}}^- \\
\varepsilon_{2i}^+ \\
\varepsilon_{2i+\frac{1}{2}}^+ \\
\varepsilon_{2i}^-
\end{array}
\right] = \begin{bmatrix}T_{1,2i-1} & T_{2,2i-1} \end{bmatrix}
\left[
\begin{array}{c}
\varepsilon_{2i-\frac{1}{2}}^-(\hat{h}) \\
\varepsilon_{2i}^+(\hat{h}) \\
\varepsilon_{2i}^-(\hat{h}) \\
\varepsilon_{2i+\frac{1}{2}}^+(\hat{h}) \\
\varepsilon_{2i}^- \\
\varepsilon_{2i+\frac{1}{2}}^-
\end{array}
\right] + \begin{bmatrix}
\varepsilon_{2i-\frac{1}{2}}^- \\
\varepsilon_{2i}^+ \\
\varepsilon_{2i}^- \\
\varepsilon_{2i+\frac{1}{2}}^+ \\
\varepsilon_{2i}^- \\
\varepsilon_{2i+\frac{1}{2}}^-
\end{array}
\right] + O\left(\max\left(\frac{1}{\sigma_i h_{2i-1}}, \frac{1}{\sigma_i h_{2i}}\right)\right),
\]

21
Therefore the leading terms in the errors after relaxation are piecewise linear across the two cells \(2i - 1\) and \(2i\) up to the accuracy of \(O\left(\max\left(\frac{1}{\sigma_{2i-1}}, \frac{1}{\sigma_{2i}}\right)\right)\). \(\square\)

The property of the leading error being piecewise linear across two cells will be very important in our multigrid algorithm. In the next section, we will introduce the notations and strategy of the multigrid solution technique.

§4 Multigrid Solution Technique

Define \(A^h\) be the fine grid operator, \(A^{2h}\) the coarse grid operator and \(I_{2h}^h\) to be interpolation operator which is give by (3.18). \(I_{2h}^h\) is defined as a restriction operator which will transfer residuals from the fine grid to coarse grid. We will give the exact form of \(I_{2h}^h\) later. Then, a three-level CS(Correction Scheme)(Brandt[4]) \(V(\nu_1, \nu_2)\) multigrid method proceeds as follows:

i) relax \(\nu_1\) times on \(A^h u^h = f^h\)

ii) calculate residual \(r^h = f^h - A^h u^h\) and transfer residual to grid \(2h\) \(f^{2h} = I_{2h}^h r^h\)

iii) relax \(\nu_1\) times on \(A^{2h} u^{2h} = f^{2h}\)

iv) calculate residual \(r^{2h} = f^{2h} - A^{2h} u^{2h}\) and transfer residual to grid \(4h\) \(f^{4h} = I_{4h}^{2h} r^{2h}\)

v) relax \(\nu_1 + \nu_2\) times on \(A^{4h} u^{4h} = f^{4h}\) or solve it exactly

vi) replace \(u^{2h} \leftarrow u^{2h} + I_{4h}^{2h} u^{4h}\) and relax \(\nu_2\) times on \(A^{2h} u^{2h} = f^{2h}\)

vii) replace \(u^h \leftarrow u^h + I_{2h}^h u^{2h}\) and relax \(\nu_2\) times on \(A^h u^h = f^h\)

We need to define interpolation and restriction operators for the modified linear discontinuous scheme. By the same finite element principle as in derivation of MLD scheme, we can derive the restriction operator \(I_{2h}^h\) and interpolation operator \(I_{2h}^h\).

From (2.17), the fine grid operator \(A^h\) is

\[
A^h = \begin{bmatrix}
A^h_1 & -C^h_1 & & & & \\
-D^h_2 & A^h_2 & -C^h_2 & & \\
& & & & & \ddots \\
& & & & & & -D^h_i & A^h_i & -C^h_i \\
& & & & & & & & & -D^h_m & A^h_m
\end{bmatrix},
\]

where

\[
A^h_i = \begin{bmatrix}
I + 2B^h_i & -R & -2R & -2B^h_i & R \\
0 & I - R & -R & B^h_i & \\
B^h_i & -R & I - R & 0 & \\
R & -2B^h_i & -2R & I + 2B^h_i & -R
\end{bmatrix},
\]

\[
C^h_i = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
B^h_i & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0
\end{bmatrix},
B^h_i = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
B^h_i & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0
\end{bmatrix},
B^h_i = \frac{1}{\sigma_{ih_i}} \begin{bmatrix}
\mu_1 & 0 & 0 & \cdots & 0 \\
0 & \cdots & 0 \\
0 & \cdots & 0
\end{bmatrix}.
\]

22
The coarse grid operator $A^{2h}$ is defined as

$$A^{2h} = I^h_h A^h I^h_{2h},$$

with restriction operator $I^h_{2h}$ defined by

$$I^h_{2h} = \begin{bmatrix} S_{1,1} & S_{1,2} & \cdots & \cdots & S_{m-1,1} & S_{m-1,2} \\ S_{1,2} & S_{3,1} & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \cdots & S_{m-1,1} & \cdots & \cdots \\ S_{m-1,1} & \cdots & \cdots & \cdots & S_{m-1,1} & S_{m-1,2} \end{bmatrix},$$

and interpolation operator $I^h_{2h}$ defined by (3.18). It is easy to show that $A^{2h}$ correspond to the MLD scheme on the coarse grid, that is,

$$A^{2h} = \begin{bmatrix} A_1^{2h} & -C_1^{2h} & -C_2^{2h} & \cdots & -C_i^{2h} & -C_m^{2h} \\ -D_2^{2h} & A_2^{2h} & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \cdots & A_i^{2h} & \cdots & \cdots \\ -D_m^{2h} & \cdots & \cdots & \cdots & A_m^{2h} & \cdots \\ -D_{m/2}^{2h} & \cdots & \cdots & \cdots & \cdots & A_{m/2}^{2h} \end{bmatrix},$$

where

$$A_1^{2h} = \begin{bmatrix} I + 2B_i^{2h} - R & -2R & -2B_i^{2h} & R \\ 0 & I - R & -R & B_i^{2h} \\ B_i^{2h} & -R & I - R & 0 \\ R & -2B_i^{2h} & -2R & I + 2B_i^{2h} - R \end{bmatrix},$$

$$C_i^{2h} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ B_i^{2h} & 0 & 0 & 0 \\ 0 & B_i^{2h} & 0 & 0 \end{bmatrix}, \quad D_i^{2h} = \begin{bmatrix} 0 & 0 & 0 & B_i^{2h} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad B_i^{2h} = \begin{bmatrix} \mu_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mu_n \end{bmatrix},$$

with $h_i = h_{2i-1} + h_{2i}$ and $i = 1, \ldots, m/2$.

In the next subsection, we analyze the convergence factors of the multigrid algorithm defined above.
§4.1 Analysis of Convergence Factors of the Multigrid Algorithm

We start our analysis with the $S_2$ problem.

**Theorem 4** For the $S_2$ angular discretization, that is, when there is only one angle in the positive direction and one angle in the negative direction, the multigrid algorithm with two-cell red-black block $\mu$-line relaxation will be an exact solver provided the coarsest grid is solved exactly.

Proof: Define $G^h$ be the two-cell red-black block $\mu$-line relaxation matrix. The errors after the relaxation will be

\[ e^h = G^h e^h_0, \]

where $e^h_0$ and $e^h$ are errors before and after relaxation respectively. From (3.16) in Theorem 1, the errors after the relaxation can be expressed as

\[ e^h = G^h e^h_0 = I_{2h}^h e^{2h}, \]

where $e^{2h}$ is defined on a coarse grid $\tilde{h}$ with mesh size of cell $i$ as $\tilde{h}_i = h_{2i-1} + h_{2i}$ and its cell edges at the same spatial position of left edge of cell $2i - 1$ and right edge of cell $2i$ of fine grid $h$. The error, $e^h_2$, after one step of a two-grid multigrid $V(1, 1)$ cycle can be written in matrix form as

\[ e^h_2 = G^h (e^h_2) - I_{2h}^h (A^{2h})^{-1} I_{2h}^h A^h G^h e^h_0 \]

Substituting (4.9) into (4.10), we have

\[ e^h_2 = G^h (I_{2h}^h e^{2h} - I_{2h}^h (A^{2h})^{-1} I_{2h}^h A^h I_{2h}^h e^{2h}). \]

Note that the coarse grid operator $A^{2h}$ is defined by (4.3). So (4.11) becomes

\[ e^h_2 = G^h (I_{2h}^h e^{2h} - I_{2h}^h (A^{2h})^{-1} A^{2h} e^{2h}) = 0. \]

This means that the errors will be completely eliminated by a multigrid V-cycle provided the coarse grid $2h$ is solved exactly. By induction, the coarse grid can be solved exactly if the coarsest grid solution is exact. The coarsest grid in our algorithm contains only two cells and can be exactly solved by two-cell block $\mu$-line relaxation. \square

For $S_N$ with $N > 2$, the multigrid algorithm will no longer be an exact solver. However, we can establish upper bounds on the convergence factors of the multigrid algorithm when

\[ \max_{k=1,...,m} (\sigma_t h_k) \ll 1 \text{ and } \min_{k=1,...,m} (\sigma_t h_k) \gg 1. \]

In the next two lemmas, we present upper bounds on the norm of the inversion of the operator $A^{2h}$. We use the notation $\|\cdot\|_{1,\infty}$ to indicate that the bounds hold for both the $L_1$ norm and the $L_\infty$ norm.
Lemma 2 When \( \max(\sigma_t \hat{h}_i, \sigma_t \hat{h}_{i+1}) \ll 1 \),

\[
\| \begin{bmatrix} \begin{array}{cc} A^{2h}_{i+1} & -C^{2h}_{i+1} \\ -D^{2h}_{i+1} & A^{2h}_{i+1} \end{array} \end{bmatrix}^{-1} \|_{1,\infty} \leq c_1 \max(\sigma_t \hat{h}_i, \sigma_t \hat{h}_{i+1}),
\]

where \( \hat{h}_i \) and \( \hat{h}_{i+1} \) are the coarse grid cell sizes and are defined as

\[
\hat{h}_i = h_{2i-1} + h_{2i}, \quad \hat{h}_{i+1} = h_{2i+1} + h_{2i+2},
\]

and \( c_1 \) is a constant independent of \( \sigma_t \hat{h}_i \) and \( \hat{h}_{i+1} \).

Proof: When \( \max_{k=(2i-1, \ldots, 2i+2)}(\sigma_t \hat{h}_k) \ll 1 \), then \( \max_{k=(i, i+1)}(\sigma_t \hat{h}_k) \ll 1 \). From (3.22), we can expand

\[
\begin{bmatrix} \begin{array}{cc} A^{2h}_{i+1} & -C^{2h}_{i+1} \\ -D^{2h}_{i+1} & A^{2h}_{i+1} \end{array} \end{bmatrix}^{-1} =
\]

\[
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

\[+O(\max(\sigma_t^2 \hat{h}_i^2, \sigma_t^2 \hat{h}_{i+1}^2)).\]

where \( M \) is defined by (3.20c). Since \( M \) is diagonal and constant, we can find constants \( c_1 \) such that

\[
\| \begin{bmatrix} \begin{array}{cc} A^{2h}_{i+1} & -C^{2h}_{i+1} \\ -D^{2h}_{i+1} & A^{2h}_{i+1} \end{array} \end{bmatrix}^{-1} \|_{1,\infty} < c_1 \max(\sigma_t \hat{h}_i, \sigma_t \hat{h}_{i+1}).
\]

To simplify our notation, let us define

\[
H_i = \begin{bmatrix} \begin{array}{cc} A^{2h}_{i+1} & -C^{2h}_{i+1} \\ -D^{2h}_{i+1} & A^{2h}_{i+1} \end{array} \end{bmatrix}^{-1} \begin{bmatrix} \begin{array}{cccc} S_{2i-1,1} & S_{2i-1,2} & 0 & 0 \\ 0 & 0 & S_{2i+1,1} & S_{2i+1,2} \end{array} \end{bmatrix}
\]

\[
U_{i+1} = \begin{bmatrix} \begin{array}{cc} A^{2h}_{i+1} & -C^{2h}_{i+1} \\ -D^{2h}_{i+1} & A^{2h}_{i+1} \end{array} \end{bmatrix}^{-1} \begin{bmatrix} \begin{array}{cccc} S_{2i-1,1} & S_{2i-1,2} & 0 & 0 \\ 0 & 0 & S_{2i+1,1} & S_{2i+1,2} \end{array} \end{bmatrix}
\]

25
where $A_k^h, C_k^h, D_k^h$ are defined by (4.7), $S_{k,1}, S_{k,2}$ are defined by (4.5) and $A_k^h, C_k^h, D_k^h$ are defined by (4.2).

**Lemma 3** When $\max(\sigma_t h_{2i-1}, \sigma_t h_{2i}, \sigma_t h_{2i+1}, \sigma_t h_{2i+2}) < 1$, then,

(4.18a) \[ ||H_i||_{1,\infty} \leq c_2 \frac{\max(\sigma_t (h_{2i-1} + h_{2i}), \sigma_t (h_{2i+1} + h_{2i+2}))}{\min(\sigma_t h_{2i-1}, \sigma_t h_{2i}, \sigma_t h_{2i+1}, \sigma_t h_{2i+2})}, \]

(4.18b) \[ ||U_{i+1}||_{1,\infty} \leq c_3 \frac{\max(\sigma_t (h_{2i-1} + h_{2i}), \sigma_t (h_{2i+1} + h_{2i+2}))}{\sigma_t (h_{2i+1} + h_{2i+2})}, \]

(4.18c) \[ ||V_{i+1}||_{1,\infty} \leq c_3 \frac{\max(\sigma_t (h_{2i+3} + h_{2i+4}), \sigma_t (h_{2i+5} + h_{2i+6}))}{\sigma_t (h_{2i+3} + h_{2i+4})}, \]

where $c_2$, and $c_3$ are constants independent of mesh sizes.

**Proof:** When $\max(\sigma_t h_{2i-1}, \sigma_t h_{2i}, \sigma_t h_{2i+1}, \sigma_t h_{2i+2}) < 1$, $||B_k^h|| < 1, k = 2i - 1, 2i, 2i + 1, 2i + 2$ in (4.2). By taking the dominant terms, we have

(4.19) \[ \begin{bmatrix} S_{2i-1,1} & S_{2i-1,2} & 0 & 0 \\ 0 & 0 & S_{2i,1} & S_{2i,2} \end{bmatrix} \begin{bmatrix} A_{2i-1}^h & -C_{2i-1}^h \\ -D_{2i}^h & A_{2i}^h \\ 0 & -D_{2i+1}^h & A_{2i+1}^h \\ 0 & 0 & -D_{2i+2}^h & A_{2i+2}^h \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} & 0 \\ 0 & H_{22} & H_{23} & H_{24} \end{bmatrix} + O(1), \]

where

(4.20a) \[ H_{11} = \begin{bmatrix} \frac{2}{\sigma_t h_{2i-1}} M & 0 & -\frac{2}{\sigma_t h_{2i-1}} M & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{\sigma_t (h_{2i-1} + h_{2i})} M & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \]

(4.20b) \[ H_{12} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sigma_t h_{2i}} M \\ 0 & 0 & 0 & 0 \\ 0 & -\frac{2}{\sigma_t h_{2i}} M & 0 & \frac{2}{\sigma_t h_{2i}} M \end{bmatrix}, \]
(4.20c) \[ H_{13} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{\sigma_t(h_{2i-1} + h_{2i})}M & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \]

(4.20d) \[ H_{22} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{\sigma_t(h_{2i+1} + h_{2i+2})}M & 0 & 0 & 0 \end{bmatrix}, \]

(4.20e) \[ H_{23} = \begin{bmatrix} \frac{2}{\sigma_t h_{2i+1}}M & 0 & -\frac{2}{\sigma_t h_{2i+1}}M & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{\sigma_t(h_{2i+1} + h_{2i+2})}M & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \]

(4.20f) \[ H_{24} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{1}{\sigma_t(h_{2i+1} + h_{2i+2})}M & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \]

Since \( M \) is constant, we can find a \( c_4 \) such that

(4.21) \[ \| S_{2i-1,1} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} A_{2i-1}^h & -C_{2i-1}^h & 0 \\ A_{2i}^h & A_{2i+1}^h & -C_{2i+1}^h \\ 0 & -D_{2i+1}^h & A_{2i+2}^h \end{bmatrix} \|_{1,\infty} \leq \frac{c_4}{\min(\sigma_t h_{2i-1}, \sigma_t h_{2i}, \sigma_t h_{2i+1}, \sigma_t h_{2i+2})}. \]

Combined with Lemma 2, we have

(4.22) \[ \| H_i \|_{1,\infty} \leq c_2 \frac{\max(\sigma_t(h_{2i-1} + h_{2i}), \sigma_t(h_{2i+1} + h_{2i+2}))}{\min(\sigma_t h_{2i-1}, \sigma_t h_{2i}, \sigma_t h_{2i+1}, \sigma_t h_{2i+2})}, \]

with \( c_2 = c_1 c_4 \).

To prove (4.18b), we have

(4.23) \[ S_{2i-1,1} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} U_{21} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \]

27
with

\[
U_{21} = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\frac{1}{\sigma_i(h_{2i+1} + h_{2i+2})} & M & 0 & 0 \\
0 & 0 & 0 & 0 \\
\end{bmatrix}.
\]

Using Lemma 2, (4.18b) is proved. The final result, (4.18c), can be proved in the same manner. □

We next assume that on the finest grid, the value of \(\sigma_i h_i\) does not vary too quickly; that is, we assume that for each \(i\)

\[
\frac{\max(\sigma_i(h_{2i-1} + h_{2i}), \sigma_i(h_{2i+1} + h_{2i+2}))}{\min(\sigma_i h_{2i-1}, \sigma_i h_{2i}, \sigma_i h_{2i+1}, \sigma_i h_{2i+2})} \leq c_5,
\]

where \(c_5\) is a bound independent of mesh sizes. With (4.25), we have

\[
||H_i||_{1,\infty} \leq c_6, \quad ||U_{i+1}||_{1,\infty} \leq c_7, \quad ||V_{i+1}||_{1,\infty} \leq c_7,
\]

where \(c_6 = c_2 c_5\) and \(c_7 = c_3 c_5\).

We arrange (4.1) as

\[
A^{2h} = Z_1 - Z_2,
\]

where

\[
Z_1 = \begin{bmatrix}
A_1^{2h} & -C_1^{2h} \\
-D_2^{2h} & A_2^{2h} \\
0 & \ddots \\
\end{bmatrix},
\]

\[
(4.28a)
\]

\[
Z_2 = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & C_2^{2h} & 0 \\
0 & D_3^{2h} & 0 & 0 \\
0 & 0 & \ddots & \ddots \\
\end{bmatrix},
\]

\[
(4.28b)
\]

\[
\]
Then

\[(A^{2h})^{-1} = (I - Z_1^{-1} Z_2)^{-1} Z_1^{-1}.\]

**Lemma 4** Suppose \(\max_{k=(1,\ldots,n)} (\sigma_i \hat{h}_k) \ll 1\) and (4.25) holds for all \(i\); then \(||(I - Z_1^{-1} Z_2)^{-1}||_2||\) is bounded by a constant independent of \(\sigma_i \hat{h}_i\).

Proof: By using (4.15), the leading term of \(I - Z_1^{-1} Z_2\) can be expanded as

\[
\begin{bmatrix}
I & Q_1 \\
Q_2 & I \\
& \ddots \\
& & & \ddots \\
& & & & \ddots \\
& & & & & Q_1 \\
& & & & & Q_2 & I
\end{bmatrix},
\]

where

\[
Q_1 = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix},
\]

\[
Q_2 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}.
\]

Notice that (4.30) can be obtained by performing elementary operations on an identity matrix. So

\[(I - Z_1^{-1} Z_2) = P_1 P_2 + O(\max_{k=(1,\ldots,n)} (\sigma_i \hat{h}_k)),\]

where \(P_1\) and \(P_2\) are elementary matrices independent of \(\sigma_i \hat{h}_i\). Thus

\[||(I - Z_1^{-1} Z_2)^{-1}||_2 < c_8,\]
where $c_{\delta}$ is a constant independent of $\sigma_{t} h_{k}$. \hfill $\Box$

With all the lemmas established, we present in the next Theorem an upper bound on the convergence factor of multigrid algorithm when $\max_{k=(1,\ldots,m)}(\sigma_{t} h_{k}) \ll 1$.

**Theorem 5** When $\max_{k=(1,\ldots,m)}(\sigma_{t} h_{k}) \ll 1$ and (4.25) holds for all $i$, a two-grid multigrid $V(1,0)$ with two-cell red-black block $\mu$-line relaxation will have a convergence factor which is less than $O\left(\max_{k=(1,\ldots,m)}\left(\sigma_{t}^{2} h_{k}^{2}\right)\right)$.

Proof: From Theorem 2, the errors after two-cell red-black block $\mu$-line relaxation can be expressed as

\begin{equation}
\epsilon_{h}^{k} = I_{2h}^{h} \epsilon_{h}^{2h} + O\left(\max_{k=(1,\ldots,m)}\left(\sigma_{t}^{2} h_{k}^{2}\right)\right) \epsilon_{0}^{h}.
\end{equation}

A multigrid $V(1,0)$ cycle in matrix form takes the form of

\begin{equation}
\epsilon_{r}^{h} = G^{h} \epsilon_{0}^{h} - I_{2h}^{h}\left(A^{2h}\right)^{-1}I_{h}^{2h} A^{h} G^{h} \epsilon_{0}^{h}.
\end{equation}

Substitute (4.34) into (4.35), we have

\begin{equation}
\epsilon_{r}^{h} = I_{2h}^{h} \epsilon_{h}^{2h} + O\left(\max_{k=(1,\ldots,m)}\left(\sigma_{t}^{2} h_{k}^{2}\right)\right) \epsilon_{0}^{h} - I_{2h}^{h}\left(A^{2h}\right)^{-1}I_{h}^{2h} A^{h} I_{2h}^{h} \epsilon_{h}^{2h} + I_{2h}^{h}\left(A^{2h}\right)^{-1}I_{h}^{2h} A^{h} O\left(\max_{k=(1,\ldots,m)}\left(\sigma_{t}^{2} h_{k}^{2}\right)\right) \epsilon_{0}^{h}.
\end{equation}

Note that

\begin{equation}
I_{2h}^{h} \epsilon_{h}^{2h} = I_{2h}^{h}\left(A^{2h}\right)^{-1}I_{h}^{2h} A^{h} I_{2h}^{h} \epsilon_{h}^{2h}.
\end{equation}

It is obvious that we only need to bound the norm of $I_{2h}^{h}\left(A^{2h}\right)^{-1}I_{h}^{2h} A^{h}$.

From (4.29), we obtain

\begin{equation}
\left(A^{2h}\right)^{-1} I_{h}^{2h} A^{h} = \left(I - Z_{1}^{-1} Z_{2}\right)^{-1} Z_{1}^{-1} I_{h}^{2h} A^{h},
\end{equation}

and

\begin{equation}
Z_{1}^{-1} I_{h}^{2h} A^{h} = \begin{bmatrix}
H_{1} & -U_{2} \\
-V_{2} & H_{3} \\
& \ddots \\
& & -V_{m-2} & H_{m-1}
\end{bmatrix}.
\end{equation}
By our assumption of (4.25) and (4.26), we have
\[
\|Z_1^{-1} I_h^2 A^h\|_{1,\infty} \leq c_9.
\]
For any matrix $A$, $\|A\|_2 \leq \sqrt{\|A\|_1 \|A\|_\infty}$. So
\[
\|Z_1^{-1} I_h^2 A^h\|_2 \leq c_9.
\]
The norms of $I_h^2$ are bounded by a constant. So
\[
\|I_h^2\|_2 < c_{10}.
\]
Combining Lemma 4 with Equations (4.41) and (4.42), we obtain
\[
\|\epsilon_h^f/\|\epsilon_h^b\|_2 < O\left( \max_{k=(1,\ldots,m)} (\sigma_i^2 h_k^2) \right).
\]

When $\min_{k=(1,\ldots,m)} (\sigma_i h_k) \gg 1$, we limit our analysis to only four cells on the fine grid with uniform mesh size $h$ and our numerical results imply that the analytical result is valid with nonuniform mesh size and large number of cells. The next theorem gives an upper bound on convergence factor of multigrid algorithm when $m = 4$ and $\sigma_i h \gg 1$. First, let us establish some space saving notation. Define the block matrices
\[
E_{j,l} = [0, \ldots, 0, I, 0, \ldots, 0],
\]
to have $l$ blocks of size $n \times n$ with the $I$ appearing as the $j$th block. Then we define another block matrix with $n \times n$ blocks, say
\[
W = [W_1, \ldots, W_k].
\]
The product $W^T E_{j,l}$ is a $k \times l$ block matrix with $n \times n$ blocks. Each block column is zero except the $j$th block column which contains $W^T$. For example, we may write $C_i$ in (2.18b) as
\[
C_i = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ B_i & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = [0, 0, B_i, 0]^T E_{1,4}.
\]

**Theorem 6**: When there are four uniform cells on the fine grid and $\sigma_i h \gg 1$, a multigrid $V(1, 1)$ cycle with two-cell red-black block $\mu$-line relaxation will have a convergence factor $\rho \leq c(\frac{1}{\sigma_i h})$ for some $c$ independent of $\sigma_i h$.

Proof: Define $A^h$ as the fine grid operator, then
\[
A^h = \begin{bmatrix} A & -C \\ -D & A \end{bmatrix},
\]
where

\[
A = \begin{bmatrix}
I + 2B - R & -2R & -2B & R & 0 & 0 & 0 & 0 \\
0 & I - R & -R & B & 0 & 0 & 0 & 0 \\
B & -R & I - R & 0 & -B & 0 & 0 & 0 \\
R & -2B & -2R & I + 2B - R & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & I + 2B - R & -2R & -2B & R \\
0 & 0 & 0 & -B & 0 & I - R & -R & B \\
0 & 0 & 0 & 0 & B & -R & I - R & 0 \\
0 & 0 & 0 & 0 & R & -2B & -2R & I + 2B - R \\
\end{bmatrix},
\]

and

\[
C = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & B & 0
\end{bmatrix}^T E_{1,s}, \quad D = \begin{bmatrix}
0 & B & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}^T E_{s,s}.
\]

The coarse grid operator \(A^{2h}\) is defined by

\[
A^{2h} = I_h^{2h} A^h I_h^{2h},
\]

with \(I_h^{2h}\) and \(I_h^{2h}\) defined by

\[
I_h^{2h} = \begin{bmatrix}
S_1 & S_2 & 0 & 0 \\
0 & 0 & S_1 & S_2 \\
\end{bmatrix}, \quad S_1 = \begin{bmatrix}
I & 0 & 0 & 0 \\
0 & 0.5I & 0 & 0 \\
0 & 0 & 0.5I & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}, \quad S_2 = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0.5I & 0 & 0 \\
0 & 0 & 0.5I & 0 \\
0 & 0 & 0 & I
\end{bmatrix},
\]

\[
I_h^{2h} = \begin{bmatrix}
T_1 & 0 \\
0 & T_2 \\
\end{bmatrix}, \quad T_1 = \begin{bmatrix}
I & 0 & 0 & 0 \\
0 & 1.5I & 0 & -0.5I \\
0.5I & 0 & 0.5I & 0 \\
0 & I & 0 & 0
\end{bmatrix}, \quad T_2 = \begin{bmatrix}
0 & 0 & I & 0 \\
0 & 0.5I & 0 & 0.5I \\
-0.5I & 0 & 1.5I & 0 \\
0 & 0 & 0 & I
\end{bmatrix}.
\]

The red-black relaxation matrix \(G^h\) is

\[
G^h = \begin{bmatrix}
A & 0 \\
-D & A
\end{bmatrix}^{-1} \begin{bmatrix}
0 & C \\
0 & 0
\end{bmatrix} = \begin{bmatrix}
0 & A^{-1}C \\
0 & A^{-1}DA^{-1}C
\end{bmatrix}.
\]

A \(V(1,1)\) multigrid cycle can be written in matrix form as

\[
\|e_h^h\| = G^h(G^h - I_h^{2h}(A^{2h})^{-1}I_h^{2h}A^hG^h)e_0^h,
\]

here \(e_0^h\) is the initial error and \(e_h^h\) is the error after the \(V(1,1)\) cycle.

Multiply \(G^h\) by \(A^h\) on the left to get

\[
A^h G^h = \begin{bmatrix}
0 & C - CA^{-1}DA^{-1}C \\
0 & 0
\end{bmatrix} = \begin{bmatrix}
0 & 0 \\
0 & I - A^{-1}DA^{-1}C
\end{bmatrix}.
\]
Note that

\[
(4.54) \quad I_{h}^{2h} \begin{bmatrix} C & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{2}B & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T E_{1,16}.
\]

The expansion of \((A^{2h})^{-1}I_{h}^{2h} \begin{bmatrix} C & 0 \\ 0 & 0 \end{bmatrix}\) is

\[
(4.55) \quad (A^{2h})^{-1}I_{h}^{2h} \begin{bmatrix} C & 0 \\ 0 & 0 \end{bmatrix} = \frac{\sigma_{h}}{4c_{2}} \begin{bmatrix} 0 \\ RM \\ 2RM \\ RM \\ 0 \end{bmatrix} E_{1,16}
\]

\[
+ \frac{1}{2c_{1}} \begin{bmatrix}
\frac{1}{2}RM + \frac{c_{1}}{2c_{2}} MRM \\
\frac{1}{8}(3 + \frac{3c_{1}}{c_{2}})RM - \frac{c_{1}}{4c_{2}} MRM - \frac{c_{1}}{4c_{2}} RM^2 \\
\frac{1}{8}(3 + \frac{3c_{1}}{c_{2}})RM - \frac{c_{1}}{4c_{2}} MRM - \frac{c_{1}}{4c_{2}} RM^2 \\
\frac{1}{4}(1 + \frac{3c_{1}}{c_{2}})RM - \frac{c_{1}}{4c_{2}} MRM - \frac{c_{1}}{4c_{2}} RM^2 \\
\frac{1}{8}(1 + \frac{3c_{1}}{c_{2}})RM - \frac{c_{1}}{4c_{2}} MRM - \frac{c_{1}}{4c_{2}} RM^2 \\
\frac{1}{2}RM + \frac{c_{1}}{2c_{2}} MRM
\end{bmatrix} E_{1,16} + O\left(\frac{1}{\sigma_{h}}\right),
\]

where \(M = \sigma_{h}B\) and \(c_{1} = \sum_{j=1}^{n} \omega_{j} \mu_{j}, c_{2} = \sum_{j=1}^{n} \omega_{j} \mu_{j}^2\) and \(c_{3} = \sum_{j=1}^{n} \omega_{j} \mu_{j}^3\). Note \(RM = c_{1} R, RM^2 = c_{2} R\) and \(RM^3 = c_{3} R\).

Expand \(I - A^{-1}DA^{-1}C\) as

\[
(4.56) \quad I - A^{-1}DA^{-1}C = I - \frac{1}{4c_{1}} \begin{bmatrix} 4RM \\ 3RM \\ 3RM \\ 2RM \\ 2RM \\ RM \\ RM \\ 0 \end{bmatrix} E_{1,8}.
\]
\[
- \frac{1}{8c_1^2 \sigma_i h} \begin{bmatrix}
-8c_2 RM - 4c_1 MRM - 4c_1 RM^2 \\
-5c_2 RM + 4c_1 MRM - \frac{1}{4}(13 - \frac{c_2}{c_1})c_1 RM^2 \\
-5c_2 RM - 4c_1 MRM - \frac{1}{4}(13 - \frac{c_2}{c_1})c_1 RM^2 \\
-2c_2 RM + 4c_1 MRM - \frac{1}{2}(5 - \frac{c_2}{c_1})c_1 RM^2 \\
-2c_2 RM - 4c_1 MRM - \frac{1}{2}(5 - \frac{c_2}{c_1})c_1 RM^2 \\
c_2 RM - 4c_1 MRM - \frac{1}{4}(5 - \frac{c_2}{c_1})c_1 RM^2 \\
c_2 RM + 4c_1 MRM - \frac{1}{4}(5 - \frac{c_2}{c_1})c_1 RM^2 \\
4c_2 RM + 4c_1 MRM
\end{bmatrix} E_{1,8} + O\left(\frac{1}{\sigma_i^2 h^2}\right)
\]

From (4.55) and (4.56), we get

\[
(4.57) \quad (A^{2h})^{-1} I_{h}^{2h} \begin{bmatrix}
C & 0 \\
0 & 0
\end{bmatrix} \begin{bmatrix}
0 & I - A^{-1} D A^{-1} C
0 & 0
\end{bmatrix} = \frac{1}{4c_1} \begin{bmatrix}
0 \\
2RM \\
2RM \\
4RM \\
4RM \\
2RM \\
2RM \\
0
\end{bmatrix} E_{9,16} + O\left(\frac{1}{\sigma_i h}\right).
\]

Then

\[
(4.58) \quad I_{2h}^{2h} (A^{2h})^{-1} I_{h}^{2h} A^{h} G^{h} = \frac{1}{4c_1} \begin{bmatrix}
0 \\
RM \\
RM \\
2RM \\
2RM \\
3RM \\
3RM \\
4RM \\
4RM \\
3RM \\
3RM \\
2RM \\
2RM \\
RM \\
RM \\
0
\end{bmatrix} E_{9,16} + O\left(\frac{1}{\sigma_i h}\right).
\]
The expansion of $G^h$ takes the form of

$$
G^h = \frac{1}{4c_1} \begin{bmatrix}
0 \\
\frac{1}{2}RM + \frac{c_1}{2c_2}RM^2 \\
\frac{1}{2}RM + \frac{c_1}{2c_2}RM^2 \\
RM + \frac{c_1}{2c_2}RM^2 \\
RM + \frac{c_1}{c_2}RM^2 \\
\frac{3}{2}RM + \frac{c_1}{2c_2}RM^2 \\
\frac{3}{2}RM + \frac{c_1}{c_2}RM^2
\end{bmatrix} E_{\sigma,16} + O(\frac{1}{\sigma_1 h}).
$$

So

$$
G^h G^h = \frac{1}{4c_1} \begin{bmatrix}
0 \\
RM \\
RM \\
2RM \\
2RM \\
3RM \\
3RM \\
4RM \\
4RM \\
3RM \\
3RM \\
2RM \\
2RM \\
RM \\
RM \\
0
\end{bmatrix} E_{\sigma,16} + O(\frac{1}{\sigma_1 h}).
$$
and

\begin{equation}
G^h I_{2h}^h (A^{2h})^{-1} I_R^2 A^h G^h = \frac{1}{4c_1} \begin{bmatrix}
0 \\
RM \\
RM \\
2RM \\
2RM \\
3RM \\
3RM \\
4RM \\
4RM \\
3RM \\
3RM \\
2RM \\
2RM \\
RM \\
RM \\
0
\end{bmatrix} E_{5,16} + O\left(\frac{1}{\sigma_1 h}\right),
\end{equation}

Combine (4.60) and (4.61), we can have

\begin{equation}
\|G^h (G^h - I_{2h}^h (A^{2h})^{-1} I_R^2 A^h G^h)\| \leq \frac{c}{\sigma_1 h}.
\end{equation}

here \(c\) is a constant independent of \(\sigma_1 h\). \(\square\)

§5 Computational Results

5.1 Multigrid Convergence Rates

The following computational results were conducted on a computational domain with \(m = 1024\) and \(S_8\), that is \(n = 4\). Our analytical results are independent of \(n\). From the computational results conducted on \(S_N\) with \(N\) ranging from 2 to 256, we observe a convergence performance independent of \(N\).

1. Convergence Factor for Uniform Grid

The results in Table 1 reveal that the convergence factor \(\rho\) is \(O(\sigma_1^3 h^3)\) when \(\sigma_1 h \ll 1\) and is \(O\left(\frac{1}{\sigma_1 h}\right)\) when \(\sigma_1 h \gg 1\). The computational results are even better than the analytical results. Our analysis only gives upper bounds on the convergence factor. We believe that the numerical results reflect the actual behavior. In Table 2, we present a range within which the maximal convergence factor occurs. Note that the maximal \(\rho\) occurs around \(\sigma_1 h = 0.01\).
<table>
<thead>
<tr>
<th>( \sigma_t , h )</th>
<th>convergence factor ( \rho )</th>
</tr>
</thead>
<tbody>
<tr>
<td>10^{-6}</td>
<td>0.23 \times 10^{-10}</td>
</tr>
<tr>
<td>10^{-5}</td>
<td>0.21 \times 10^{-7}</td>
</tr>
<tr>
<td>10^{-4}</td>
<td>0.16 \times 10^{-4}</td>
</tr>
<tr>
<td>10^{-3}</td>
<td>0.20 \times 10^{-2}</td>
</tr>
<tr>
<td>10^{-2}</td>
<td>0.97 \times 10^{-2}</td>
</tr>
<tr>
<td>10^{-1}</td>
<td>0.89 \times 10^{-2}</td>
</tr>
<tr>
<td>1</td>
<td>0.58 \times 10^{-3}</td>
</tr>
<tr>
<td>10^1</td>
<td>0.27 \times 10^{-4}</td>
</tr>
<tr>
<td>10^2</td>
<td>0.20 \times 10^{-6}</td>
</tr>
<tr>
<td>10^3</td>
<td>0.20 \times 10^{-8}</td>
</tr>
<tr>
<td>10^4</td>
<td>0.20 \times 10^{-10}</td>
</tr>
</tbody>
</table>

Table 1. Convergence Factors for Uniform Grid

<table>
<thead>
<tr>
<th>( \sigma_t , h )</th>
<th>convergence factor ( \rho )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.005</td>
<td>0.72 \times 10^{-2}</td>
</tr>
<tr>
<td>0.006</td>
<td>0.87 \times 10^{-2}</td>
</tr>
<tr>
<td>0.007</td>
<td>0.95 \times 10^{-2}</td>
</tr>
<tr>
<td>0.008</td>
<td>0.98 \times 10^{-2}</td>
</tr>
<tr>
<td>0.009</td>
<td>0.98 \times 10^{-2}</td>
</tr>
<tr>
<td>0.010</td>
<td>0.97 \times 10^{-2}</td>
</tr>
<tr>
<td>0.011</td>
<td>0.96 \times 10^{-2}</td>
</tr>
<tr>
<td>0.012</td>
<td>0.96 \times 10^{-2}</td>
</tr>
<tr>
<td>0.013</td>
<td>0.95 \times 10^{-2}</td>
</tr>
<tr>
<td>0.014</td>
<td>0.95 \times 10^{-2}</td>
</tr>
<tr>
<td>0.015</td>
<td>0.95 \times 10^{-2}</td>
</tr>
</tbody>
</table>

Table 2. Convergence Factors for Uniform Grid: Worst Case

2. Convergence Factor for Nonuniform Grid

We choose a highly variable \( \sigma_t h_i \) by letting \( \sigma_t h_i = c10^{2\eta_i}, i = 1, \ldots, m \) where \( \eta_i \) is a random number between \((-1, 1)\). Thus, neighboring cells can vary in width up to 4 orders of magnitude. Here, \( c \) is a constant chosen to control the range of cell widths. If \( c = 1 \), then \( \sigma_t h_i \) are random numbers between \((0.01, 100)\). If \( c \geq 100 \), \( \sigma_t h_i \geq 1 \). If \( c \leq 0.01 \), all \( \sigma_t h_i \leq 1 \). The results of Table 3 indicate that the performance of our multigrid scheme for a nonuniform grid is the same as for a uniform grid. Furthermore in both thin and thick limit cases, a nonuniform grid has the same convergence behavior as uniform grid.
<table>
<thead>
<tr>
<th>$c$</th>
<th>convergence factor $\rho$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1.0^{-6}$</td>
<td>$9.9 \times 10^{-10}$</td>
</tr>
<tr>
<td>$1.0^{-5}$</td>
<td>$8.5 \times 10^{-7}$</td>
</tr>
<tr>
<td>$1.0^{-4}$</td>
<td>$2.7 \times 10^{-4}$</td>
</tr>
<tr>
<td>$1.0^{-3}$</td>
<td>$5.4 \times 10^{-3}$</td>
</tr>
<tr>
<td>$1.0^{-2}$</td>
<td>$9.1 \times 10^{-4}$</td>
</tr>
<tr>
<td>$1.0^{-1}$</td>
<td>$3.3 \times 10^{-4}$</td>
</tr>
<tr>
<td>$1.0^{0}$</td>
<td>$6.1 \times 10^{-6}$</td>
</tr>
<tr>
<td>$1.0^{1}$</td>
<td>$2.4 \times 10^{-8}$</td>
</tr>
<tr>
<td>$1.0^{2}$</td>
<td>$2.3 \times 10^{-10}$</td>
</tr>
</tbody>
</table>

Table 3. Convergence Factors for Nonuniform Grids

5.2 DSA Convergence Rates

As was mentioned in the introduction, a competing algorithm for the solution of the isotropic $S_N$ equations in slab geometry is Diffusion Synthetic Acceleration (DSA). It is motivated by the observation that, in the thick limit, the solution of the linear Boltzman equation (1.1) becomes independent of angle except near boundaries and sources. If the $S_N$ problem were recast as an $S_2$ problem, then the zeroth and first Legendre moments of the solution could be found by solving a diffusion equation for the zeroth moment. DSA can be viewed as a preconditioning technique (Faber and Manteuffel [5]) or as a two level multigrid scheme (Larson [7]) in which an $S_2$ problem is used as a coarse level correction to an $S_N$ problem. In this context, a transport sweep or x-line relaxation (see3.3)) corresponds to multigrid relaxation and the $S_2$ problem represents the coarse level. DSA skips all of the intermediate levels. The multigrid in angle scheme developed in Morel and Manteuffel [15] visits levels $N, N/2, N/4, \ldots, 2$ and is effective for isotropic scattering as well as highly anisotropic scattering.

The overall DSA algorithm is very sensitive to the difference scheme used to solve the diffusion equation. In our numerical results we solve an $S_2$ equation discretized with a MLD scheme using Marshak boundary conditions as the coarse level correction. This is equivalent to solving a consistently differenced diffusion equation for the zeroth moment. In Table 4 and Table 5 we present convergence factors for a $V(1,1)$ multigrid cycle and for a DSA cycle. Here, the slab is assumed to have physical thickness 1. Thus, $\sigma_t$ represents the width of the slab measured in the number of mean-free-paths. The tests were performed using $S_8$ and a wide range of $\sigma_t$ and $m$, (the number of cells). The diagonals of these tables represent constant $\sigma_t h$. 

38
<table>
<thead>
<tr>
<th>$\sigma_i$</th>
<th>$m = 16$</th>
<th>$m = 64$</th>
<th>$m = 256$</th>
<th>$m = 1024$</th>
<th>$m = 4096$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$4^{-4}$</td>
<td>.72E-8</td>
<td>.10E-7</td>
<td>.10E-7</td>
<td>.10E-7</td>
<td>.10E-7</td>
</tr>
<tr>
<td>$4^{-3}$</td>
<td>.37E-6</td>
<td>.55E-6</td>
<td>.56E-6</td>
<td>.56E-6</td>
<td>.56E-6</td>
</tr>
<tr>
<td>$4^{-2}$</td>
<td>.17E-4</td>
<td>.16E-4</td>
<td>.18E-4</td>
<td>.26E-4</td>
<td>.27E-4</td>
</tr>
<tr>
<td>$4^{-1}$</td>
<td>.25E-3</td>
<td>.53E-3</td>
<td>.55E-3</td>
<td>.55E-4</td>
<td>.58E-4</td>
</tr>
<tr>
<td>1</td>
<td>.14E-2</td>
<td>.37E-2</td>
<td>.44E-2</td>
<td>.44E-2</td>
<td>.46E-2</td>
</tr>
<tr>
<td>64</td>
<td>.46E-4</td>
<td>.57E-3</td>
<td>.23E-2</td>
<td>.85E-2</td>
<td>.99E-2</td>
</tr>
<tr>
<td>256</td>
<td>.20E-5</td>
<td>.46E-4</td>
<td>.58E-3</td>
<td>.25E-2</td>
<td>.89E-2</td>
</tr>
<tr>
<td>1024</td>
<td>.12E-6</td>
<td>.18E-5</td>
<td>.47E-4</td>
<td>.58E-3</td>
<td>.26E-2</td>
</tr>
<tr>
<td>4096</td>
<td>.75E-8</td>
<td>.12E-6</td>
<td>.16E-5</td>
<td>.46E-4</td>
<td>.58E-3</td>
</tr>
<tr>
<td>8192</td>
<td>.45E-9</td>
<td>.75E-8</td>
<td>.12E-6</td>
<td>.14E-5</td>
<td>.47E-4</td>
</tr>
<tr>
<td>16384</td>
<td>.28E-10</td>
<td>.46E-9</td>
<td>.75E-8</td>
<td>.12E-6</td>
<td>.11E-5</td>
</tr>
<tr>
<td>32768</td>
<td>.18E-11</td>
<td>.30E-10</td>
<td>.47E-9</td>
<td>.75E-8</td>
<td>.12E-6</td>
</tr>
<tr>
<td>65536</td>
<td>.11E-12</td>
<td>.20E-11</td>
<td>.30E-10</td>
<td>.46E-9</td>
<td>.75E-8</td>
</tr>
</tbody>
</table>

Table 4. Convergence Factors for the Multigrid Algorithm

<table>
<thead>
<tr>
<th>$\sigma_i$</th>
<th>$m = 16$</th>
<th>$m = 64$</th>
<th>$m = 256$</th>
<th>$m = 1024$</th>
<th>$m = 4096$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$4^{-5}$</td>
<td>.190E-2</td>
<td>.190E-2</td>
<td>.190E-2</td>
<td>.190E-2</td>
<td>.190E-2</td>
</tr>
<tr>
<td>$4^{-3}$</td>
<td>.276E-1</td>
<td>.276E-1</td>
<td>.276E-1</td>
<td>.276E-1</td>
<td>.276E-1</td>
</tr>
<tr>
<td>$4^{-2}$</td>
<td>.836E-1</td>
<td>.834E-1</td>
<td>.834E-1</td>
<td>.835E-1</td>
<td>.835E-1</td>
</tr>
<tr>
<td>$4^{-1}$</td>
<td>.153</td>
<td>.151</td>
<td>.151</td>
<td>.152</td>
<td>.152</td>
</tr>
<tr>
<td>1</td>
<td>.188</td>
<td>.203</td>
<td>.204</td>
<td>.204</td>
<td>.205</td>
</tr>
<tr>
<td>4</td>
<td>.180</td>
<td>.217</td>
<td>.216</td>
<td>.217</td>
<td>.218</td>
</tr>
<tr>
<td>16</td>
<td>.142</td>
<td>.164</td>
<td>.204</td>
<td>.217</td>
<td>.219</td>
</tr>
<tr>
<td>64</td>
<td>.129</td>
<td>.143</td>
<td>.149</td>
<td>.165</td>
<td>.185</td>
</tr>
<tr>
<td>256</td>
<td>.125</td>
<td>.131</td>
<td>.143</td>
<td>.145</td>
<td>.160</td>
</tr>
<tr>
<td>1024</td>
<td>.124</td>
<td>.126</td>
<td>.131</td>
<td>.143</td>
<td>.146</td>
</tr>
<tr>
<td>4096</td>
<td>.124</td>
<td>.124</td>
<td>.126</td>
<td>.131</td>
<td>.142</td>
</tr>
<tr>
<td>8192</td>
<td>.124</td>
<td>.124</td>
<td>.126</td>
<td>.127</td>
<td>.132</td>
</tr>
<tr>
<td>16384</td>
<td>.124</td>
<td>.124</td>
<td>.124</td>
<td>.126</td>
<td>.126</td>
</tr>
<tr>
<td>32768</td>
<td>.124</td>
<td>.124</td>
<td>.124</td>
<td>.124</td>
<td>.126</td>
</tr>
<tr>
<td>65536</td>
<td>.124</td>
<td>.124</td>
<td>.124</td>
<td>.124</td>
<td>.124</td>
</tr>
</tbody>
</table>

Table 5. Convergence Factors for the DSA Algorithm
For the multigrid algorithm, the convergence factors are roughly equal along diagonals, with
the slowest rates occurring for \( \sigma_t h \) in the range \( 4^{-3} \times 0.0156 \) to \( 4^{-5} \times 0.97E - 3 \). For example, \( \sigma_t = 4^{-1}, m = 16 \) and \( \sigma_t = 64, m = 4096 \) are on the diagonal with \( \sigma_t h = 0.0156 \).

The behavior of the DSA algorithm is quite different. The best rate achieved for thick
problems is .124. Further, for any fixed \( \sigma_t \geq 4 \), the slab is thick but the cells become individually
thin as the number of cells increases. In this limit DSA saturates to a convergence factor of .23.
In our tests the the value of \( m \) was not quite large enough to achieve this limit, but a value of
.219 was reached for \( \sigma_t = 16 \) and \( m = 4096 \).

In all cases, the multigrid convergence factors were superior to the DSA convergence factors.
However, it is important to adjust for the relative amount of computational work required by
each algorithm. Of course, such measures will be machine dependent. On the Cray Y/MP, where
these tests were performed, we compared times for \( N=64 \) and \( m=1024 \). The \( V(1,1) \) cycle required
8.9 seconds and the DSA cycle required 3.7 seconds. The ratio of these times is 2.4. Table 6 is
the result of raising each term in Table 5 to the power 2.4. This is a more fair comparision
with Table 4 on a serial machine. Note that the multigrid algorithm is faster than DSA in all regimes.
On a parallel machine we expect the results to more heavily favor the multigrid algorithm. Both
algorithms can be implemented with parallel complexity \( O(\log(m)) \). In this context, however,
we expect the times required to perform a single DSA cycle and a single multigrid \( V(1,1) \) cycle
to be more nearly equal. A parallel version of the multigrid algorithm is described in [10], [11].
A parallel version of the DSA sweep has also been implemented on the Thinking Machines Inc.
CM-200 at Los Alamos National Laboratories [20].

<table>
<thead>
<tr>
<th>( \sigma_t )</th>
<th>( m = 16 )</th>
<th>( m = 64 )</th>
<th>( m = 256 )</th>
<th>( m = 1024 )</th>
<th>( m = 4096 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 4^{-3} )</td>
<td>.29E-6</td>
<td>.29E-6</td>
<td>.29E-6</td>
<td>.29E-6</td>
<td>.29E-6</td>
</tr>
<tr>
<td>( 4^{-4} )</td>
<td>.78E-5</td>
<td>.78E-5</td>
<td>.78E-5</td>
<td>.78E-5</td>
<td>.78E-5</td>
</tr>
<tr>
<td>( 4^{-3} )</td>
<td>.18E-3</td>
<td>.18E-3</td>
<td>.18E-3</td>
<td>.18E-3</td>
<td>.18E-3</td>
</tr>
<tr>
<td>( 4^{-2} )</td>
<td>.25E-2</td>
<td>.25E-2</td>
<td>.25E-2</td>
<td>.25E-2</td>
<td>.25E-2</td>
</tr>
<tr>
<td>( 4^{-1} )</td>
<td>.11E-1</td>
<td>.10E-1</td>
<td>.10E-1</td>
<td>.10E-1</td>
<td>.10E-1</td>
</tr>
<tr>
<td>1</td>
<td>.18E-1</td>
<td>.21E-1</td>
<td>.22E-1</td>
<td>.22E-1</td>
<td>.22E-1</td>
</tr>
<tr>
<td>4</td>
<td>.16E-1</td>
<td>.23E-1</td>
<td>.25E-1</td>
<td>.25E-1</td>
<td>.25E-1</td>
</tr>
<tr>
<td>16</td>
<td>.92E-2</td>
<td>.13E-1</td>
<td>.22E-1</td>
<td>.25E-1</td>
<td>.26E-1</td>
</tr>
<tr>
<td>64</td>
<td>.73E-2</td>
<td>.93E-2</td>
<td>.10E-1</td>
<td>.13E-1</td>
<td>.17E-1</td>
</tr>
<tr>
<td>256</td>
<td>.68E-2</td>
<td>.76E-2</td>
<td>.93E-2</td>
<td>.97E-2</td>
<td>.12E-1</td>
</tr>
<tr>
<td>1024</td>
<td>.66E-2</td>
<td>.69E-2</td>
<td>.76E-2</td>
<td>.93E-2</td>
<td>.98E-2</td>
</tr>
<tr>
<td>4096</td>
<td>.66E-2</td>
<td>.66E-2</td>
<td>.69E-2</td>
<td>.76E-2</td>
<td>.92E-2</td>
</tr>
<tr>
<td>8192</td>
<td>.66E-2</td>
<td>.66E-2</td>
<td>.69E-2</td>
<td>.70E-2</td>
<td>.77E-2</td>
</tr>
<tr>
<td>16384</td>
<td>.66E-2</td>
<td>.66E-2</td>
<td>.69E-2</td>
<td>.69E-2</td>
<td>.69E-2</td>
</tr>
<tr>
<td>32768</td>
<td>.66E-2</td>
<td>.66E-2</td>
<td>.66E-2</td>
<td>.66E-2</td>
<td>.69E-2</td>
</tr>
<tr>
<td>65536</td>
<td>.66E-2</td>
<td>.66E-2</td>
<td>.66E-2</td>
<td>.66E-2</td>
<td>.66E-2</td>
</tr>
</tbody>
</table>

Table 6. Adjusted Convergence Factors for the DSA Algorithm

40
In either setting, parallel or serial, a full multigrid algorithm can be implemented (cf. [13]). This version of multigrid starts on a coarse grid and moves toward finer grids. In general, an amount of work equal to two $V(1,1)$ cycles on the finest grid will yield a solution that is accurate to the level of discretization error. This would provide a savings in some regions of the tables. Moreover, full multigrid provides a natural framework for adaptive grid refinement.

§6 Conclusions

From our numerical and analytical results, we conclude that

- For $S_2$, two-cell, red-black, block $\mu$-line relaxation is a good smoother which makes the errors exactly in the range of interpolation after one relaxation. Thus, one multigrid $V(1,0)$ cycle is an exact solver for the $S_2$ problem.

- For $S_N$ with $N > 2$, the errors at two neighboring cells $2i - 1, 2i$ after one such relaxation will be in the range of interpolation up to accuracy of $O\left(\max\left(\frac{1}{\sigma_i h_{2i-1}}, \frac{1}{\sigma_i h_{2i}}\right)\right)$ when $\min(\sigma_i h_{2i-1}, \sigma_i h_{2i}) \geq 1$ and $O\left(\max(\sigma_i^2 h_{2i-1}^2, \sigma_i^2 h_{2i}^2)\right)$ when $\max(\sigma_i h_{2i-1}, \sigma_i h_{2i}) \ll 1$.

- The multigrid algorithm is an efficient solver which has a convergence factor of $O\left(\max_{k=1,\ldots,m} \left(\frac{1}{\sigma_i^2 h_k^2}\right)\right)$ for the thick limit and $O\left(\max_{k=1,\ldots,m} \left(\frac{1}{\sigma_i^2 h_k^2}\right)\right)$ for the thin limit with a maximal of $\rho = 0.0098$ occurring around $\sigma_i h = 0.01$ for the uniform grid.

- The multigrid algorithm performs equally well for uniform and nonuniform grids.

- In the thick limit a single $V(1,1)$ yields nearly an exact solution.

- The multigrid algorithm is more efficient than DSA in all regimes.

- The relaxation is implemented in a red-black ordering and is parallelizable.

The above results assumed no absorption. The problem is easy to solve if the absorption is sufficiently large, that is if $\gamma = \frac{\sigma_t}{\sigma_i}$ is bounded away from unity. In this case, smoothing alone is sufficient for a fast solution. With no absorption, $\gamma = 1$, the algorithm presented above provides nearly an exact solver. However, for the case in which $\gamma = 1 - O\left(\frac{1}{\sigma_i h}\right)$ in the thick limit, the above algorithm yields a convergence factor that does not go to zero in the thick limit. A minor adaptation to the algorithm has been developed that treats this special case and provides a convergence factor that does go to zero in the thick limit for all levels of absorption. It will appear in a subsequent report.

References


