Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2008) 2187–2194
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Short Note

Extension of Kershaw diffusion scheme to hexahedral meshes

Milad Fatenejad *, Gregory A. Moses

Department of Engineering Physics, University of Wisconsin-Madison, 1500 Engineering Drive, Madison, WI 53706, USA

Received 17 April 2007; received in revised form 4 November 2007; accepted 5 November 2007
Available online 12 November 2007

Keywords: Kershaw; Diffusion; Non-orthogonal mesh

1. Introduction

Numerical solution of the time dependent diffusion equation on non-orthogonal meshes in two spatial dimensions is an essential feature of typical Lagrangian radiation hydrodynamics simulation codes used for designing inertial confinement fusion targets and modeling other high energy density plasmas. Electron thermal conduction is treated with a non-linear diffusion equation and radiative transfer is often modeled using multi-group flux-limited diffusion. For two dimensions, an initially orthogonal mesh can become non-orthogonal due to hydrodynamic flow irregularities and the diffusion equation must be numerically differenced on this non-orthogonal mesh. A novel approach to this problem was reported by Kershaw [1], where he used a variational method to derive the difference operator corresponding to the continuous diffusion operator on a non-orthogonal r–z mesh. Kershaw's method leads to a nine-point differencing stencil and tractable positive definite matrix solutions for problems of practical significance. It reduces to the standard five-point differencing stencil for orthogonal meshes. For this reason Kershaw's method is often used as a benchmark for comparison of more recent, higher order methods [2].

In this paper, the discretization scheme developed by Kershaw is extended to three dimensional non-uniform hexahedral x–y–z meshes. As three dimensional radiation hydrodynamics simulations become more commonplace, this three dimensional Kershaw scheme can be a viable approach to solving the diffusion equation. While higher order discretizations exist, the 3D Kershaw method has the benefit of using only zone centered unknowns with a local computational stencil making it relatively easy to implement in Lagrangian codes. Along these lines, the detailed difference equations for the resulting 19 point computational stencil are presented in this paper. While this scheme has limited accuracy, it can still serve as a benchmark for comparison of more elaborate higher order, but more expensive schemes.

This extension shares many of the same properties as the original method. The resulting matrix has the benefit of being symmetric positive definite (SPD). However, it is not an "M-matrix" and therefore negative

* Corresponding author. Tel.: +1 608 265 4329.
E-mail address: mfatenejad@wisc.edu (M. Fatenejad).

doi:10.1016/j.jcp.2007.11.001

answers are not mathematically forbidden. The convergence properties are also similar to the original method. In general, first order convergence is expected, except on non-smooth meshes where the accuracy is degraded. A numerical test is presented which demonstrates this property. Furthermore, the difference equations reduce to the standard second-order, 7-point scheme as the mesh becomes orthogonal and uniform, therefore higher order accuracy is expected in this case.

2. Discretizing the diffusion operator

The process for discretizing the diffusion operator closely follows Kershaw's derivation of the two dimensional r–z method. The first step involves expressing the diffusion operator as a function of continuous variables K, L, and M as shown in (1) through (3). The variable, j, defined in (3) is the Jacobian which transforms from the physical variables (X, Y, Z) to (K, L, M), and is simply the zone volume. The lack of the factors of $\mathbf{R}$ that appear in Kershaw's original derivation is the key mathematical difference between the cylindrical and Cartesian formulations.

$$\begin{aligned}\nabla\cdot D\nabla f = \frac{1}{j}\Bigg\{&\frac{\partial}{\partial K}\left(\frac{D}{j}|\mathbf{R}_K|^2\frac{\partial f}{\partial K}\right)+\frac{\partial}{\partial L}\left(\frac{D}{j}|\mathbf{R}_L|^2\frac{\partial f}{\partial L}\right)+\frac{\partial}{\partial M}\left(\frac{D}{j}|\mathbf{R}_M|^2\frac{\partial f}{\partial M}\right)+\frac{\partial}{\partial L}\left(\frac{D}{j}\mathbf{R}_K\cdot\mathbf{R}_L\frac{\partial f}{\partial K}\right)\\&+\frac{\partial}{\partial K}\left(\frac{D}{j}\mathbf{R}_L\cdot\mathbf{R}_K\frac{\partial f}{\partial L}\right)+\frac{\partial}{\partial M}\left(\frac{D}{j}\mathbf{R}_K\cdot\mathbf{R}_M\frac{\partial f}{\partial K}\right)+\frac{\partial}{\partial K}\left(\frac{D}{j}\mathbf{R}_M\cdot\mathbf{R}_K\frac{\partial f}{\partial M}\right)\\&+\frac{\partial}{\partial M}\left(\frac{D}{j}\mathbf{R}_L\cdot\mathbf{R}_M\frac{\partial f}{\partial L}\right)+\frac{\partial}{\partial L}\left(\frac{D}{j}\mathbf{R}_M\cdot\mathbf{R}_L\frac{\partial f}{\partial M}\right)\Bigg\}\end{aligned} \tag{1}$$

where

$$\mathbf{R}_K=\begin{pmatrix}\frac{\partial y}{\partial L}\frac{\partial z}{\partial M}-\frac{\partial y}{\partial M}\frac{\partial z}{\partial L}\\ \frac{\partial z}{\partial L}\frac{\partial x}{\partial M}-\frac{\partial z}{\partial M}\frac{\partial x}{\partial L}\\ \frac{\partial x}{\partial L}\frac{\partial y}{\partial M}-\frac{\partial x}{\partial M}\frac{\partial y}{\partial L}\end{pmatrix},\quad \mathbf{R}_L=\begin{pmatrix}\frac{\partial y}{\partial M}\frac{\partial z}{\partial K}-\frac{\partial y}{\partial K}\frac{\partial z}{\partial M}\\ \frac{\partial z}{\partial M}\frac{\partial x}{\partial K}-\frac{\partial z}{\partial K}\frac{\partial x}{\partial M}\\ \frac{\partial x}{\partial M}\frac{\partial y}{\partial K}-\frac{\partial x}{\partial K}\frac{\partial y}{\partial M}\end{pmatrix},\quad \mathbf{R}_M=\begin{pmatrix}\frac{\partial y}{\partial K}\frac{\partial z}{\partial L}-\frac{\partial y}{\partial L}\frac{\partial z}{\partial K}\\ \frac{\partial z}{\partial K}\frac{\partial x}{\partial L}-\frac{\partial z}{\partial L}\frac{\partial x}{\partial K}\\ \frac{\partial x}{\partial K}\frac{\partial y}{\partial L}-\frac{\partial x}{\partial L}\frac{\partial y}{\partial K}\end{pmatrix} \tag{2}$$

and,

$$j=\det\begin{pmatrix}\frac{\partial x}{\partial K}&\frac{\partial y}{\partial K}&\frac{\partial z}{\partial K}\\ \frac{\partial x}{\partial L}&\frac{\partial y}{\partial L}&\frac{\partial z}{\partial L}\\ \frac{\partial x}{\partial M}&\frac{\partial y}{\partial M}&\frac{\partial z}{\partial M}\end{pmatrix} \tag{3}$$

A variational approach, which is embodied in (4), is used for the discretization. The surface integral term which would normally appear has been set to zero as in the 2D method, because the function, f, is assumed to be zero on the boundary at this point in the derivation. Although this is not always the case physically (boundary conditions where f is non-zero are allowed and are discussed later), all of the terms involving non-zero boundary values are moved to the right hand side of the diffusion equation after the discretization occurs. In other words, the resulting diffusion matrix, A, always has the form consistent with an $f=0$ boundary condition and cancelling this term is justified.

$$\int f\nabla\cdot D\nabla f\,\mathrm{d}x\mathrm{d}y\mathrm{d}z=-\int D(\nabla f)^2\mathrm{d}x\mathrm{d}y\mathrm{d}z=-\int\left[\sqrt{\frac{D}{j}}\left(\mathbf{R}_K\frac{\partial}{\partial K}+\mathbf{R}_L\frac{\partial}{\partial L}+\mathbf{R}_M\frac{\partial}{\partial M}\right)f\right]^2\mathrm{d}K\mathrm{d}L\mathrm{d}M \tag{4}$$

The discrete form of the variational relationship is given in (5). To make this transition, the variables K, L, and M are now allowed to assume only integer values and are no longer continuous functions. At this point, they represent the mesh indices where the 3D mesh indexing is shown in Fig. 1.

$$-\frac{1}{8}\sum_{(K,L,M)}\sum_{i=1}^{8}[B^i f]^2_{K,L,M}=+\sum_{(K,L,M)}f_{K,L,M}(Af)_{K,L,M}V_{K,L,M} \tag{5}$$

Here, A is the matrix representing the finite difference form of the diffusion operator, V is the zone volume, and B is the matrix representation of the term in brackets in (4). The summation over i appears because there are

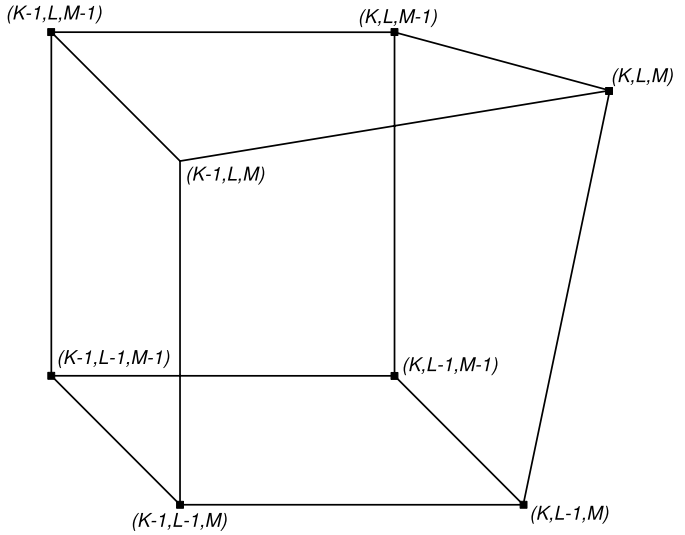

Fig. 1. Indexing of mesh points for zone K, L, M.

eight equally valid representations of the matrix that must be equally averaged to maintain the rotational symmetry of the problem. The matrix B is defined in (6).

$$(Bf)_{K,L,M} = \mathscr{K}_{K,L,M}\mathbf{R}_K + \mathscr{L}_{K,L,M}\mathbf{R}_L + \mathscr{M}_{K,L,M}\mathbf{R}_M \tag{6}$$

where,

$$\begin{aligned}
\mathscr{K}_{K,L,M} &= \Sigma_{K,L,M}(f_{K+1,L,M} - f_{K,L,M}) \approx \sqrt{\frac{D}{j}}\frac{\partial f}{\partial K},\\
\mathscr{L}_{K,L,M} &= \Lambda_{K,L,M}(f_{K,L+1,M} - f_{K,L,M}) \approx \sqrt{\frac{D}{j}}\frac{\partial f}{\partial L}, \text{ and}\\
\mathscr{M}_{K,L,M} &= \Gamma_{K,L,M}(f_{K,L,M+1} - f_{K,L,M}) \approx \sqrt{\frac{D}{j}}\frac{\partial f}{\partial M}.
\end{aligned} \tag{7}$$

The terms Σ, Λ, and Γ represent the face-centered coefficient $(D/j)^{1/2}$ computed on the $K+1$, $L+1$, and $M+1$ faces of the zones, respectively. These terms are evaluated by averaging quantities in the zones neighboring each face. For example, Σ can be averaged as follows:

$$\Sigma_{K,L,M} = \left[2/\left(j_{K,L,M}/D_{K,L,M} + j_{K+1,L,M}/D_{K+1,L,M}\right)\right]^{1/2} \tag{8}$$

Λ is averaged between zones L and $L+1$, and Γ is averaged between zones M and $M+1$. These quantities can be computed on either zone face in each direction which leads to the following eight representations for the quantity (Bf). The 2D method requires four B matrices for the same reason.

$$\begin{aligned}
(B^1 f)_{K,L,M} &= \mathscr{K}_{K,L,M}\mathbf{R}_K + \mathscr{L}_{K,L,M}\mathbf{R}_L + \mathscr{M}_{K,L,M}\mathbf{R}_M\\
(B^2 f)_{K,L,M} &= \mathscr{K}_{K-1,L,M}\mathbf{R}_K + \mathscr{L}_{K,L,M}\mathbf{R}_L + \mathscr{M}_{K,L,M}\mathbf{R}_M\\
(B^3 f)_{K,L,M} &= \mathscr{K}_{K,L,M}\mathbf{R}_K + \mathscr{L}_{K,L-1,M}\mathbf{R}_L + \mathscr{M}_{K,L,M}\mathbf{R}_M\\
(B^4 f)_{K,L,M} &= \mathscr{K}_{K,L,M}\mathbf{R}_K + \mathscr{L}_{K,L,M}\mathbf{R}_L + \mathscr{M}_{K,L,M-1}\mathbf{R}_M\\
(B^5 f)_{K,L,M} &= \mathscr{K}_{K-1,L,M}\mathbf{R}_K + \mathscr{L}_{K,L-1,M}\mathbf{R}_L + \mathscr{M}_{K,L,M}\mathbf{R}_M\\
(B^6 f)_{K,L,M} &= \mathscr{K}_{K-1,L,M}\mathbf{R}_K + \mathscr{L}_{K,L,M}\mathbf{R}_L + \mathscr{M}_{K,L,M-1}\mathbf{R}_M\\
(B^7 f)_{K,L,M} &= \mathscr{K}_{K,L,M}\mathbf{R}_K + \mathscr{L}_{K,L-1,M}\mathbf{R}_L + \mathscr{M}_{K,L,M-1}\mathbf{R}_M\\
(B^8 f)_{K,L,M} &= \mathscr{K}_{K-1,L,M}\mathbf{R}_K + \mathscr{L}_{K,L-1,M}\mathbf{R}_L + \mathscr{M}_{K,L,M-1}\mathbf{R}_M
\end{aligned} \tag{9}$$

The vectors $\mathbf{R}_K$, $\mathbf{R}_L$, and $\mathbf{R}_M$ can now be discretized by differencing the coordinate variables with respect to K, L, and M.

$$\left(\frac{\partial \xi}{\partial K}\right)_{K,L,M} = \frac{1}{4}\left(\xi_{K,L,M} + \xi_{K,L-1,M} + \xi_{K,L,M-1} + \xi_{K,L-1,M-1} - \xi_{K-1,L,M} - \xi_{K-1,L-1,M} - \xi_{K-1,L,M-1} - \xi_{K-1,L-1,M-1}\right) \tag{10}$$

where ξ represents either x, y, or z. Then, for example, the discretized $(\mathbf{R}_K)_{K,\,L,\,M}$ can be computed as shown in (11).

$$(\mathbf{R}_K)_{K,L,M} = \begin{pmatrix} \left(\frac{\partial y}{\partial L}\right)_{K,L,M}\left(\frac{\partial z}{\partial M}\right)_{K,L,M} - \left(\frac{\partial y}{\partial M}\right)_{K,L,M}\left(\frac{\partial z}{\partial L}\right)_{K,L,M} \\ \left(\frac{\partial z}{\partial L}\right)_{K,L,M}\left(\frac{\partial x}{\partial M}\right)_{K,L,M} - \left(\frac{\partial z}{\partial M}\right)_{K,L,M}\left(\frac{\partial x}{\partial L}\right)_{K,L,M} \\ \left(\frac{\partial x}{\partial L}\right)_{K,L,M}\left(\frac{\partial y}{\partial M}\right)_{K,L,M} - \left(\frac{\partial x}{\partial M}\right)_{K,L,M}\left(\frac{\partial y}{\partial L}\right)_{K,L,M} \end{pmatrix}, \tag{11}$$

Using these definitions along with the B^i matrices, one can compute the elements of the diffusion matrix, A, according to (12).

$$VA = -\frac{1}{8}\sum_{i=1}^{8}(B^i)^T \cdot (B^i) \tag{12}$$

The result of this computation is a 19-point stencil, which is expected. In the original 2D method, Kershaw uses a Taylor expansion to show that the discrete operator must satisfy six equations. However, a six point coupling could not be used while preserving symmetry, which led him to a nine-point stencil. This same exercise can be carried out in three dimensions. In this case ten equations must be satisfied, and a 19-point scheme is the smallest stencil that still preserves rotational symmetry. The matrix elements are listed in (13) through (22). Since the matrix is symmetric, only the upper triangular elements are shown.

$$\begin{aligned} V_{K,L,M}A_{(K,L,M),(K,L,M)} &= -\frac{1}{2}\Big\{\Sigma^2_{K,L,M}(\mathbf{R}_K)^2_{K+1,L,M} + \Lambda^2_{K,L,M}(\mathbf{R}_L)^2_{K,L+1,M} + \Gamma^2_{K,L,M}(\mathbf{R}_M)^2_{K,L,M+1} + \Gamma^2_{K,L,M-1}(\mathbf{R}_M)^2_{K,L,M-1} \\ &\quad + \Lambda^2_{K,L-1,M}(\mathbf{R}_L)^2_{K,L-1,M} + \Sigma^2_{K-1,L,M}(\mathbf{R}_K)^2_{K-1,L,M} + \left(\Gamma^2_{K,L,M} + \Gamma^2_{K,L,M-1}\right)(\mathbf{R}_M)^2_{K,L,M} \\ &\quad + \left(\Lambda^2_{K,L,M} + \Lambda^2_{K,L-1,M}\right)(\mathbf{R}_L)^2_{K,L,M} + \left(\Sigma^2_{K,L,M} + \Sigma^2_{K-1,L,M}\right)(\mathbf{R}_K)^2_{K,L,M} \\ &\quad + (\Gamma_{K,L,M} - \Gamma_{K,L,M-1})(\Lambda_{K,L,M} - \Lambda_{K,L-1,M})(\mathbf{R}_L)_{K,L,M} \cdot (\mathbf{R}_M)_{K,L,M} \\ &\quad + (\Gamma_{K,L,M} - \Gamma_{K,L,M-1})(\Sigma_{K,L,M} - \Sigma_{K-1,L,M})(\mathbf{R}_K)_{K,L,M} \cdot (\mathbf{R}_M)_{K,L,M} \\ &\quad + (\Lambda_{K,L,M} - \Lambda_{K,L-1,M})(\Sigma_{K,L,M} - \Sigma_{K-1,L,M})(\mathbf{R}_K)_{K,L,M} \cdot (\mathbf{R}_L)_{K,L,M}\Big\} \end{aligned} \tag{13}$$

$$\begin{aligned} V_{K,L,M}A_{(K,L,M),(K+1,L,M)} &= -\frac{1}{4}\Sigma_{K,L,M}\Big\{-2\Sigma_{K,L,M}\left((\mathbf{R}_K)^2_{K+1,L,M} + (\mathbf{R}_K)^2_{K,L,M}\right) + (\Gamma_{K+1,L,M} - \Gamma_{K+1,L,M-1})(\mathbf{R}_K)_{K+1,L,M} \cdot (\mathbf{R}_M)_{K+1,L,M} \\ &\quad + (\Gamma_{K,L,M-1} - \Gamma_{K,L,M})(\mathbf{R}_K)_{K,L,M} \cdot (\mathbf{R}_M)_{K,L,M} + (\Lambda_{K+1,L,M} - \Lambda_{K+1,L-1,M})(\mathbf{R}_K)_{K+1,L,M} \cdot (\mathbf{R}_L)_{K+1,L,M} \\ &\quad + (\Lambda_{K,L-1,M} - \Lambda_{K,L,M})(\mathbf{R}_K)_{K,L,M} \cdot (\mathbf{R}_L)_{K,L,M}\Big\} \end{aligned} \tag{14}$$

$$V_{K,L,M}A_{(K,L,M),(K-1,L+1,M)} = -\frac{1}{4}\left\{\Sigma_{K-1,L+1,M}\Lambda_{K,L,M}(\mathbf{R}_K)_{K,L+1,M} \cdot (\mathbf{R}_L)_{K,L+1,M} + \Lambda_{K-1,L,M}\Sigma_{K-1,L,M}(\mathbf{R}_K)_{K-1,L,M} \cdot (\mathbf{R}_L)_{K-1,L,M}\right\} \tag{15}$$

$$V_{K,L,M}A_{(K,L,M),(K,L+1,M)} = -\frac{1}{4}\Lambda_{K,L,M}\Big\{-2\Lambda_{K,L,M}\Big((\mathbf{R}_L)^2_{K,L+1,M}+(\mathbf{R}_L)^2_{K,L,M}\Big)+(\Gamma_{K,L+1,M}-\Gamma_{K,L+1,M-1})(\mathbf{R}_L)_{K,L+1,M}\cdot(\mathbf{R}_M)_{K,L+1,M} + (\Gamma_{K,L,M-1}-\Gamma_{K,L,M})(\mathbf{R}_L)_{K,L,M}\cdot(\mathbf{R}_M)_{K,L,M}+(\Sigma_{K,L+1,M}-\Sigma_{K-1,L+1,M})(\mathbf{R}_L)_{K,L+1,M}\cdot(\mathbf{R}_K)_{K,L+1,M} + (\Sigma_{K-1,L,M}-\Sigma_{K,L,M})(\mathbf{R}_L)_{K,L,M}\cdot(\mathbf{R}_K)_{K,L,M}\Big\} \tag{16}$$

$$V_{K,L,M}A_{(K,L,M),(K+1,L+1,M)} = \frac{1}{4}\Big\{\Sigma_{K,L,M}\Lambda_{K+1,L,M}(\mathbf{R}_K)_{K+1,L,M}\cdot(\mathbf{R}_L)_{K+1,L,M}+\Lambda_{K,L,M}\Sigma_{K,L+1,M}(\mathbf{R}_K)_{K,L+1,M}\cdot(\mathbf{R}_L)_{K,L+1,M}\Big\} \tag{17}$$

$$V_{K,L,M}A_{(K,L,M),(K,L-1,M+1)} = -\frac{1}{4}\Big\{\Lambda_{K,L-1,M+1}\Gamma_{K,L,M}(\mathbf{R}_L)_{K,L,M+1}\cdot(\mathbf{R}_M)_{K,L,M+1}+\Lambda_{K,L-1,M}\Gamma_{K,L-1,M}(\mathbf{R}_L)_{K,L-1,M}\cdot(\mathbf{R}_M)_{K,L-1,M}\Big\} \tag{18}$$

$$V_{K,L,M}A_{(K,L,M),(K-1,L,M+1)} = -\frac{1}{4}\Big\{\Sigma_{K-1,L,M+1}\Gamma_{K,L,M}(\mathbf{R}_K)_{K,L,M+1}\cdot(\mathbf{R}_M)_{K,L,M+1}+\Sigma_{K-1,L,M}\Gamma_{K-1,L,M}(\mathbf{R}_K)_{K-1,L,M}\cdot(\mathbf{R}_M)_{K-1,L,M}\Big\} \tag{19}$$

$$V_{K,L,M}A_{(K,L,M),(K,L,M+1)} = \frac{1}{4}\Gamma_{K,L,M}\Big\{2\Gamma_{K,L,M}\Big((\mathbf{R}_M)^2_{K,L,M+1}+(\mathbf{R}_M)^2_{K,L,M}\Big)+(\Lambda_{K,L-1,M+1}-\Lambda_{K,L,M+1})(\mathbf{R}_L)_{K,L,M+1}\cdot(\mathbf{R}_M)_{K,L,M+1} + (\Lambda_{K,L,M}-\Lambda_{K,L-1,M})(\mathbf{R}_L)_{K,L,M}\cdot(\mathbf{R}_M)_{K,L,M}+(\Sigma_{K-1,L,M+1}-\Sigma_{K,L,M+1})(\mathbf{R}_M)_{K,L,M+1}\cdot(\mathbf{R}_K)_{K,L,M+1} + (\Sigma_{K,L,M}-\Sigma_{K-1,L,M})(\mathbf{R}_M)_{K,L,M}\cdot(\mathbf{R}_K)_{K,L,M}\Big\} \tag{20}$$

$$V_{K,L,M}A_{(K,L,M),(K+1,L,M+1)} = \frac{1}{4}\Big\{\Sigma_{K,L,M}\Gamma_{K+1,L,M}(\mathbf{R}_K)_{K+1,L,M}\cdot(\mathbf{R}_M)_{K+1,L,M}+\Sigma_{K,L,M+1}\Gamma_{K,L,M}(\mathbf{R}_K)_{K,L,M+1}\cdot(\mathbf{R}_M)_{K,L,M+1}\Big\} \tag{21}$$

$$V_{K,L,M}A_{(K,L,M),(K,L+1,M+1)} = \frac{1}{4}\Big\{\Lambda_{K,L,M}\Gamma_{K,L+1,M}(\mathbf{R}_L)_{K,L+1,M}\cdot(\mathbf{R}_M)_{K,L+1,M}+\Lambda_{K,L,M+1}\Gamma_{K,L,M}(\mathbf{R}_L)_{K,L,M+1}\cdot(\mathbf{R}_M)_{K,L,M+1}\Big\} \tag{22}$$

These expressions result in the formation of a symmetric positive definite matrix, and reduce to the 2D x–y Kershaw equations in problems with uniformity in the z direction. In the case of an orthogonal grid the equations greatly simplify since

$$(\mathbf{R}_K)_{K,L,M}(\mathbf{R}_L)_{K,L,M} =$$
$$(\mathbf{R}_K)_{K,L,M}(\mathbf{R}_M)_{K,L,M} =$$
$$(\mathbf{R}_L)_{K,L,M}(\mathbf{R}_M)_{K,L,M} = 0.$$

It can then be shown that as the mesh becomes uniform and orthogonal, the 3D Kershaw equations reduce to the standard seven-point, second-order scheme.

The boundary conditions are addressed through the use of ghost zones which surround the physical domain. A description of the boundaries along the K direction follows, and a similar procedure can be followed along the L and M directions. Zones that are part of the physical mesh lie between indices K_s and K_e. Using this notation, the ghost zones would be labeled $K_s - 1$ and $K_e + 1$.

The values of $\mathbf{R}_K$, $\mathbf{R}_L$, and $\mathbf{R}_M$ in the ghost zones must be computed. Simply let $(\mathbf{R}_L)_{K_s-1} = (\mathbf{R}_M)_{K_s-1} = 0$ and $(\mathbf{R}_L)_{K_e+1} = (\mathbf{R}_M)_{K_e+1} = 0$. The value of $\mathbf{R}_K$ is computed as shown earlier, however only mesh points on the boundary of the physical domain, points with index $K_s - 1$ and K_e, are used, and the points with index $K_s - 2$ and $K_e + 1$ are not needed.

Next, the value of Σ must be computed on faces between the physical and ghost zones. For a zero-flux (Neumann) boundary condition, simply let $\Sigma_{K_s-1} = \Sigma_{K_e} = 0$. For fixed f (Dirichlet) boundary conditions, Σ can be computed as usual, while setting $j_{K_s-1} = j_{K_e+1} = 0$. The matrix is computed using Eqs. (13) through

(22), but terms multiplying the boundary value of f must be moved to the right hand side of the discretized diffusion equation. Finally, let $\Lambda = \Gamma = 0$ on surfaces between ghost zones.

This procedure is easily replicated in the L and M directions. Note that the difference equations never require the value of $\mathbf{R}_K$, $\mathbf{R}_L$, or $\mathbf{R}_M$, be specified on the problem corners or edges. Furthermore the boundary conditions do not require the definition of the position of mesh points on the outward face of the ghost zones.

3. Computational results

The matrix B used to compute the diffusion matrix is defined as

$$(Bf)_{K,L,M} \approx \sqrt{\frac{D}{j}}\left(\mathbf{R}_K \frac{\partial}{\partial K} + \mathbf{R}_L \frac{\partial}{\partial L} + \mathbf{R}_M \frac{\partial}{\partial M}\right) f, \tag{23}$$

and as in the 2D method [1], by defining B as a function of the vectors $\mathbf{R}_K$, $\mathbf{R}_L$, and $\mathbf{R}_M$ an assumption is made that the coordinates are a smooth function of the logical variables K, L, and M. This leads to a requirement that the mesh lines be smooth in order to obtain first order accuracy, which was demonstrated in [2] for the 2D method. In this section, the result of a convergence test is shown on smooth/non-orthogonal and non-smooth/non-orthogonal meshes to demonstrate the convergence properties of the 3D method.

The two meshes used for the convergence test are shown in Fig. 2. The non-smooth, non-orthogonal mesh is shown on the left [3] and is simply an extension of the original Kershaw Z-mesh which appeared in [1]. It has the property that it does not become smoother as the mesh is refined. The smooth, non-orthogonal is illustrated on the right of Fig. 2. This mesh remains non-orthogonal as the resolution is increased while also becoming smoother. The convergence test involves a simple problem where the top boundary is fixed at a temperature of 1 eV, the bottom boundary is fixed at 0 ev, and a steady-state solution is used to compute the error. The conductivity is set to a constant value of 1×10^{11} ergs/cm/eV/s, resulting in a linear steady-state temperature distribution in the z direction. The physical domain is cube with sides of length 20 μm. Fig. 3 shows both meshes with temperature contour lines superimposed. Because the temperature is treated as a zone centered quantity, the mesh lines in this figure trace the zone centers rather than the vertices. This helps reduce artificial distortions in the plotting caused by interpolating the temperatures to the position of the vertices.

$$\text{error} = \sqrt{\frac{\sum_{K,L,M}\left(f^{\text{exact}}_{K,L,M} - f^{\text{numeric}}_{K,L,M}\right)^2}{\sum_{K,L,M}\left(f^{\text{exact}}_{K,L,M}\right)^2}} \tag{24}$$

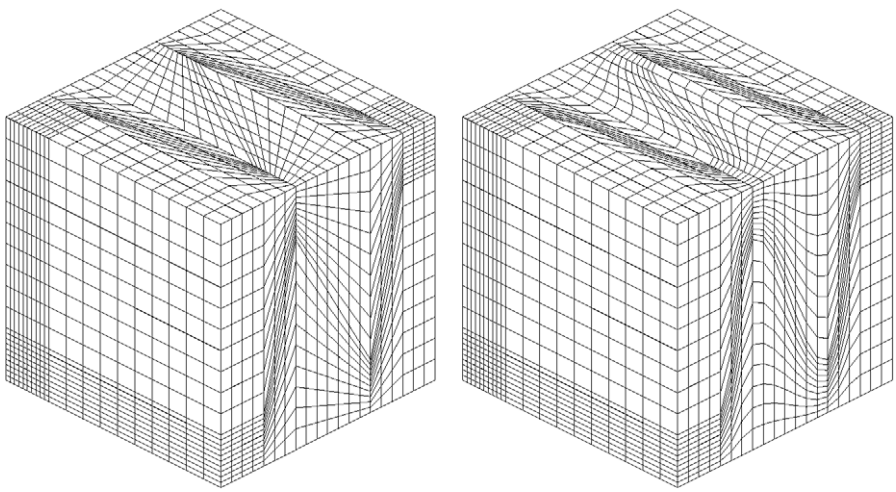

Fig. 2. The non-smooth, non-orthogonal 3D Kershaw mesh (left) and the smooth, non-orthogonal 3D Kershaw mesh (right).

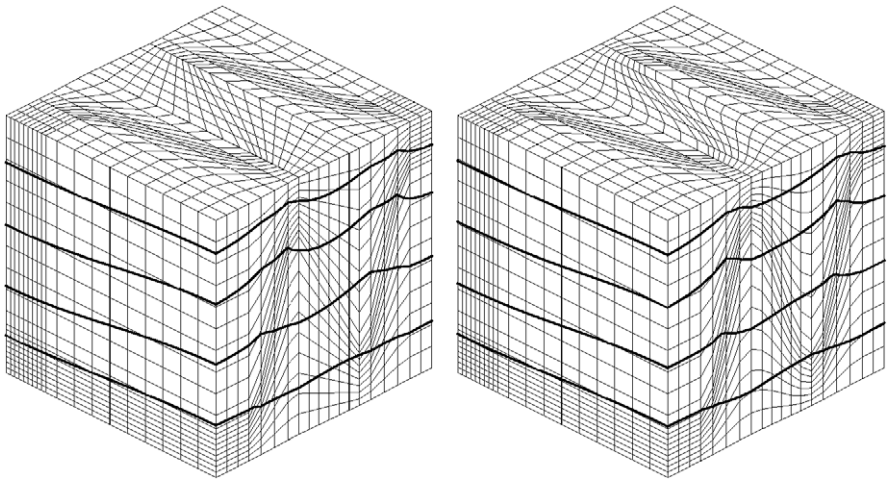

Fig. 3. The non-smooth (left) and smooth (right) Kershaw meshes with mesh lines intersecting zone centers rather than vertices. Thick black lines represent temperature contours.

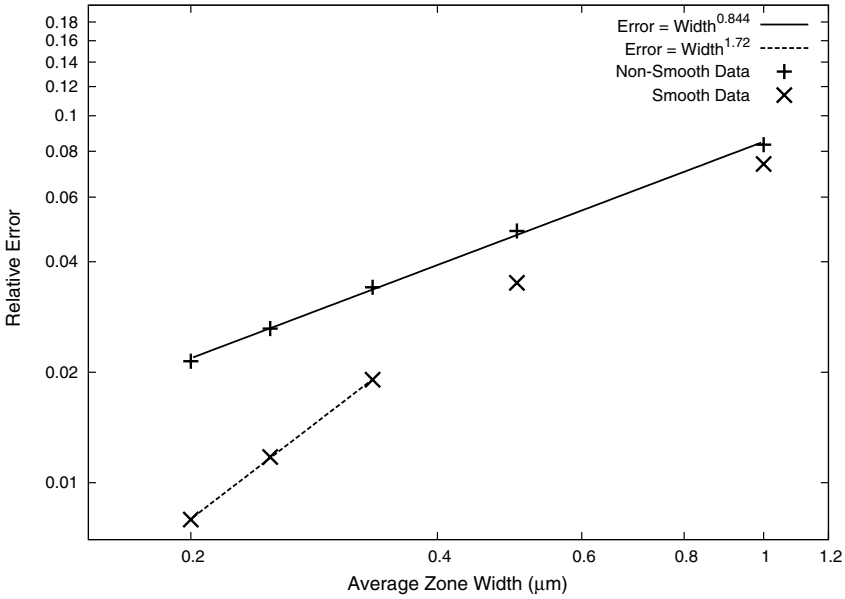

Fig. 4. Convergence test results.

The error was computed using the L_2 norm shown in (24). A plot of the convergence test results is shown in Fig. 4. The average zone width is computed by simply dividing the number of grid points in a direction by the width of the domain, in this case, 20 μm. For example, an average zone width of 1 μm corresponds to using twenty grid points in the I, J, and K directions. For larger zone widths (fewer mesh points), the lines on the smooth mesh are still somewhat jagged. Once a sufficient number of points are added the problem falls into the asymptotic regime and the order of convergence can be determined. For this reason, the line in Fig. 4 corresponding to the smooth mesh is only drawn through three of the five-points. Overall, the behavior is as expected. The 3D scheme is at least first order convergent on smooth meshes, and has difficulty converging on non-smooth meshes. This is the same significant drawback that is present in the 2D method.

4. Conclusions

Expressions for the matrix elements of a symmetric positive definite matrix A that approximates the diffusion operator on non-orthogonal 3D x–y–z meshes are derived using the Kershaw formulation. This results in a 19-point stencil. Convergence testing of this scheme on non-orthogonal 3D meshes confirms that its properties are consistent with those of the original Kershaw scheme. The method is relatively simple to implement in simulation codes, which comes at the cost of accuracy. Specifically, the method is generally less than second order accurate and care must be taken when the mesh is not smooth, as this can further degrade the accuracy. The resulting matrix is not an M-matrix which can lead to negative temperatures, although non were obtained in the convergence test that was run. However, within these limitations, the 3D method has the potential to serve as a viable method for solving the diffusion equation on non-orthogonal meshes in 3D radiation hydrodynamics simulations. This scheme is currently implemented in the 3D version of the DRACO laser fusion simulation code [4].

Acknowledgments

This work was supported by the University of Rochester Laboratory for Laser Energetics. The authors wish to thank Marvin Adams and James Morel for their help and advice.

References

[1] D.S. Kerhsaw, Differencing of the diffusion equations in Lagrangian hydrodynamics code, J. Comput. Phys. 39 (1981) 375.
[2] J.E. Morel, J.E. Dendy Jr., M.L. Hall, S.W. White, A cell-centered Lagrangian-mesh diffusion differencing scheme, J. Comput. Phys. 103 (1992) 286.
[3] J.E. Morel, M.L. Hall, M.J. Shashkov, A local support-operators diffusion discretization scheme for hexahedral meshes, J. Comput. Phys. 170 (2001) 338.
[4] M. Fatenejad, G.A. Moses, DRACO development for 3D simulations, Bull. APS 51 (2006) 209.