A Second-Order Cell-Centered Diffusion Difference Scheme for Unstructured Hexahedral Lagrangian Meshes

Michael L. Hall
Jim E. Morel
Group CIC-19, Radiation Transport Team
P.O. Box 1663, MS-B265
Los Alamos National Laboratory
Los Alamos, New Mexico 87545 USA
Email: `hall@lanl.gov`

10 / 24 / 96

Outline

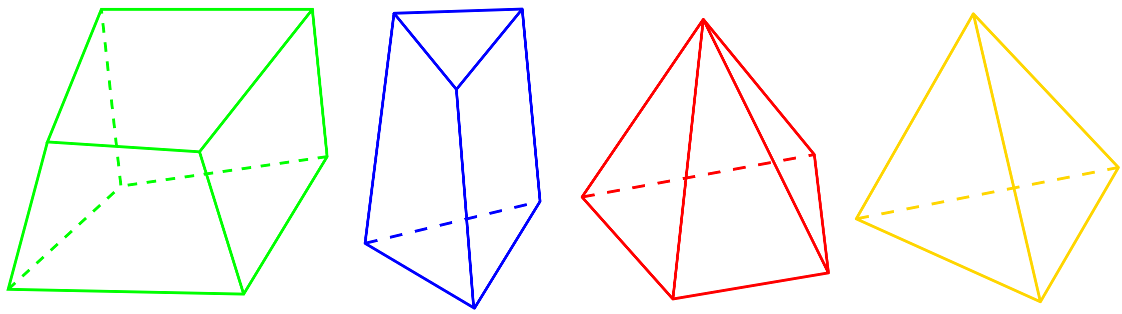
- Problem Justification
- Equation Set
- Properties of the Method
- Discretization
- Algebraic Solution
- Method Summary
- Implementation
- Results
- Future Work

Problem Justification:
Diffusion Applications

- Heat Conduction
- Fluid Flow
- Radiation Transport
  - Diffusion
  - Diffusion Synthetic Acceleration
  - Simplified Spherical Harmonic (SP_N) Methods

Problem Justification:
Mesh Description

- 3-Dimensional
- Hexahedra and Degenerate Hexahedra (Prisms, Pyramids, Tetrahedra)



- Unstructured
  - Block Structured
  - Curved geometries

Equation Set:

$$\alpha\frac{\partial\Phi}{\partial t} - \overrightarrow{\nabla} \cdot D \overrightarrow{\nabla} \Phi + \overrightarrow{\nabla} \cdot \overrightarrow{J} + \sigma\Phi = S$$

Which can be written

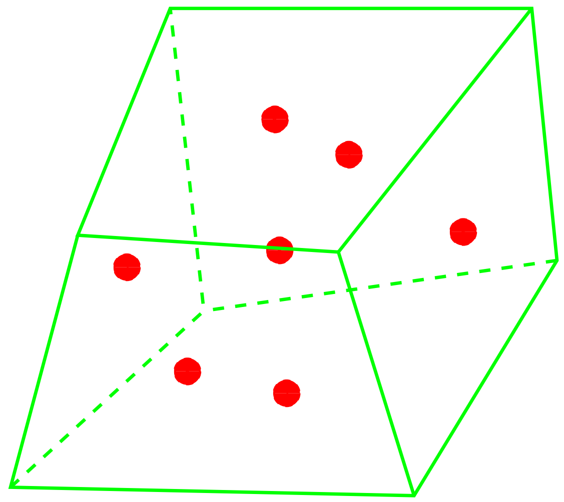
$$\alpha\frac{\partial\Phi}{\partial t} + \overrightarrow{\nabla} \cdot \overrightarrow{F} + \sigma\Phi = S$$

$$\overrightarrow{F} = -D \overrightarrow{\nabla} \Phi + \overrightarrow{J}$$

Where

Φ	$=$	Intensity
$\overrightarrow{F}$	$=$	Flux
D	$=$	Diffusion Coefficient
α	$=$	Time Derivative Coefficient
σ	$=$	Removal Coefficient
S	$=$	Intensity Source Term
$\overrightarrow{J}$	$=$	Flux Source Term

Properties of the Method



- Cell-centered (balance equations are done over a cell)

- Cell-centered and face-centered unknowns (required to rigorously treat material discontinuities)

- Unstructured mesh

- Derivation valid for 1-D, 2-D, and 3-D geometries

- Preserves homogeneous linear solutions, second-order accurate

Properties of the Method

- Reduces to the standard cell-centered 7-point operator for an orthogonal mesh

- Local energy conservation is maintained

- Unsymmetric matrix system

- Extension of the method described in

  Morel, J. E., J. E. Dendy, Jr., Michael L. Hall, and Stephen W. White. A Cell-Centered Lagrangian-Mesh Diffusion Differencing Scheme. Journal of Computational Physics, 103(2):286-299, December 1992.

  to 3-D unstructured meshes, with an alternate derivation.

Discretization: Conservation Equation

Integrate the conservation equation over the cell volume,

$$\int_{V_c} \alpha \frac{\partial \Phi}{\partial t}\, dV + \int_{V_c} \overrightarrow{\nabla} \cdot \overrightarrow{F}\, dV + \int_{V_c} \sigma \Phi \, dV = \int_{V_c} S\, dV$$

Define cell averages and use Gauss' Theorem:

$$\alpha_c \frac{\partial \Phi_c}{\partial t} V_c + \int_A \overrightarrow{F} \cdot \overrightarrow{dA} + \sigma_c \Phi_c V_c = S_c V_c$$

Discretize temporally and evaluate flux integral:

$$\frac{\alpha_c V_c}{\Delta t}\left(\Phi_c^{n+1} - \Phi_c^n\right) + \sum_f \overrightarrow{F_f^{n+1}} \cdot \overrightarrow{A_f} + \sigma_c \Phi_c^{n+1} V_c = S_c V_c$$

Discretization: Flux Terms

We need to express $\overrightarrow{F_f^{n+1}}$ in terms of Φ^{n+1}.

Start with the flux equation:

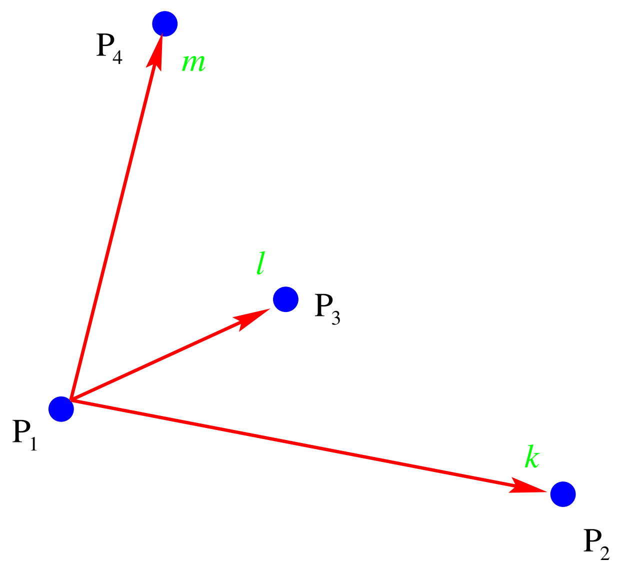
$$\overrightarrow{F_f^{n+1}} = -D_{c,f}\,\overrightarrow{\nabla}\,\Phi + \overrightarrow{J_f}$$

The flux source, $\overrightarrow{J_f}$, is known.

The diffusion coefficient is known within a cell, but may be discontinuous at the cell face.

The discretization must accurately model material discontinuities.

Discretization: Flux Terms



The values of Φ at four non-planar points are needed to determine the gradient. Any four non-planar points $(\overrightarrow{P_1}, \overrightarrow{P_2}, \overrightarrow{P_3}, \overrightarrow{P_4})$ define a coordinate system in terms of three vectors,

$$
\begin{aligned}
\hat{k} &= \overrightarrow{P_2} - \overrightarrow{P_1} \\
\hat{l} &= \overrightarrow{P_3} - \overrightarrow{P_1} \\
\hat{m} &= \overrightarrow{P_4} - \overrightarrow{P_1}
\end{aligned}
$$

Discretization: Flux Terms

A Jacobian matrix converts between the (k, l, m) coordinate system and the (x, y, z) coordinate system:

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial k} & \frac{\partial x}{\partial l} & \frac{\partial x}{\partial m} \\ \frac{\partial y}{\partial k} & \frac{\partial y}{\partial l} & \frac{\partial y}{\partial m} \\ \frac{\partial z}{\partial k} & \frac{\partial z}{\partial l} & \frac{\partial z}{\partial m} \end{bmatrix} \begin{bmatrix} P_k \\ P_l \\ P_m \end{bmatrix}$$

which is represented as:

$$\overrightarrow{P}_{x,y,z} = \mathbf{J} \overrightarrow{P}_{k,l,m}$$

Note that an equally valid reverse transformation from the (x, y, z) coordinate system to the (k, l, m) coordinate system could have been used, with a Jacobian matrix equal to $\mathbf{J}^{-1}$.

Discretization: Flux Terms

Since the four points are located along the axes in (k, l, m)-space, but not in (x, y, z)-space, it is easier to take the derivatives needed for the forward Jacobian than the reverse Jacobian:

$$
\begin{aligned}
\mathbf{J} &= \begin{bmatrix} \frac{\partial x}{\partial k} & \frac{\partial x}{\partial l} & \frac{\partial x}{\partial m} \\ \frac{\partial y}{\partial k} & \frac{\partial y}{\partial l} & \frac{\partial y}{\partial m} \\ \frac{\partial z}{\partial k} & \frac{\partial z}{\partial l} & \frac{\partial z}{\partial m} \end{bmatrix} \\
&= \begin{bmatrix} \left(\overrightarrow{P_2} - \overrightarrow{P_1} \right) & \left(\overrightarrow{P_3} - \overrightarrow{P_1} \right) & \left(\overrightarrow{P_4} - \overrightarrow{P_1} \right) \end{bmatrix} \\
&= \begin{bmatrix} \hat{k} & \hat{l} & \hat{m} \end{bmatrix}
\end{aligned}
$$

Discretization: Flux Terms

Returning to the consideration of the gradient term, expand the k, l and m derivatives of Φ using the chain rule:

$$\begin{bmatrix} \frac{\partial \Phi}{\partial k} \\ \frac{\partial \Phi}{\partial l} \\ \frac{\partial \Phi}{\partial m} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial k} & \frac{\partial y}{\partial k} & \frac{\partial z}{\partial k} \\ \frac{\partial x}{\partial l} & \frac{\partial y}{\partial l} & \frac{\partial z}{\partial l} \\ \frac{\partial x}{\partial m} & \frac{\partial y}{\partial m} & \frac{\partial z}{\partial m} \end{bmatrix} \begin{bmatrix} \frac{\partial \Phi}{\partial x} \\ \frac{\partial \Phi}{\partial y} \\ \frac{\partial \Phi}{\partial z} \end{bmatrix}$$

$$= \mathbf{J}^T \overrightarrow{\nabla} \Phi$$

or, solving for $\overrightarrow{\nabla} \Phi$ and inserting the derivative definitions,

$$\overrightarrow{\nabla} \Phi = \mathbf{J}^{-T} \begin{bmatrix} \frac{\partial \Phi}{\partial k} \\ \frac{\partial \Phi}{\partial l} \\ \frac{\partial \Phi}{\partial m} \end{bmatrix} = \mathbf{J}^{-T} \begin{bmatrix} \Phi_2 - \Phi_1 \\ \Phi_3 - \Phi_1 \\ \Phi_4 - \Phi_1 \end{bmatrix}$$

Discretization: Flux Terms

Now that we know how to represent gradients from any four points in (x, y, z)-space in (k, l, m)-space, which points do we choose?

We are limited to adding points within the cell to maintain a rigorous treatment of material discontinuities.

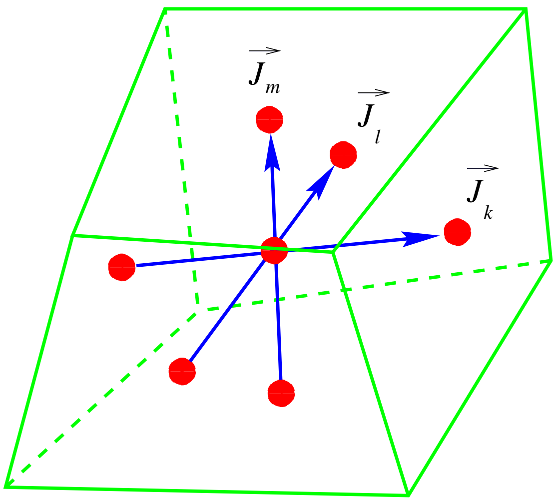
Adding four different points for each of the six faces results in twenty-five unknowns per cell, including the cell center, which is clearly untenable.

Fortunately, there is a better solution…

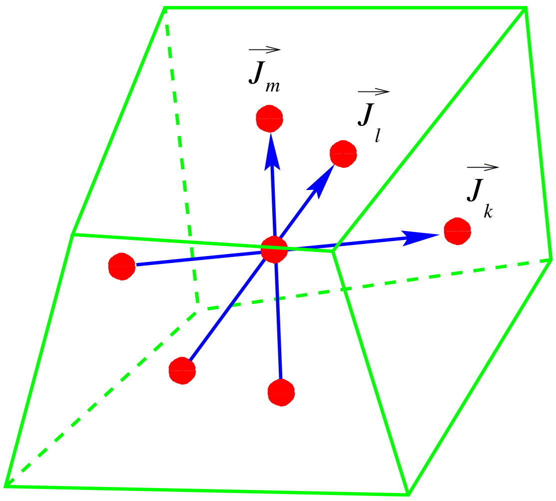
Discretization: Flux Terms

Four points are not the only way to determine a gradient: three lines that intersect in a single point can also be used.

If we place a point (and therefore an unknown Φ) in the center of each face, the three lines formed by connecting opposing faces all intersect at the cell center. A single Jacobian matrix per cell will be sufficient.



Discretization: Flux Terms



If the vectors connecting the face centers of opposite faces are denoted $\overrightarrow{J_k}$, $\overrightarrow{J_l}$, and $\overrightarrow{J_m}$ for the k, l, and m directions, then the Jacobian matrix is given by:

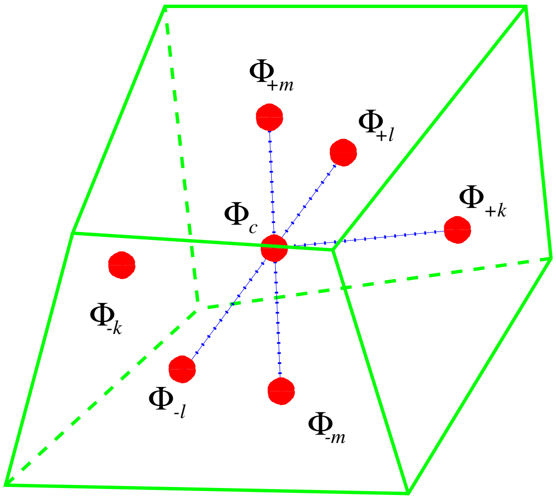
$$\mathbf{J} = \left[\begin{array}{ccc} \overrightarrow{J_k} & \overrightarrow{J_l} & \overrightarrow{J_m} \end{array} \right]$$

and the inverse transpose matrix is:

$$\mathbf{J}^{-T} = \frac{1}{|\mathbf{J}|} \left[\begin{array}{ccc} \left(\overrightarrow{J_l} \times \overrightarrow{J_m} \right) & \left(\overrightarrow{J_m} \times \overrightarrow{J_k} \right) & \left(\overrightarrow{J_k} \times \overrightarrow{J_l} \right) \end{array} \right]$$

Discretization: Flux Terms

There are seven unknowns in each cell. The gradient for each face is represented by the cell value for the $\mathbf{J}^{-T}$ matrix multiplied by the k, l, and m derivative vector for that face.



Minor direction derivatives (for example, the l and m derivatives on the $+k$ face) are evaluated across the full cell, and major direction derivatives use a half cell.

For example,

$$\overrightarrow{F_{+k}^{n+1}} = -D_{c,f}\,\mathbf{J}^{-T}\begin{bmatrix} 2\left(\Phi_{+k}^{n+1} - \Phi_{c}^{n+1}\right) \\ \Phi_{+l}^{n+1} - \Phi_{-l}^{n+1} \\ \Phi_{+m}^{n+1} - \Phi_{-m}^{n+1} \end{bmatrix} + \overrightarrow{J_f}$$

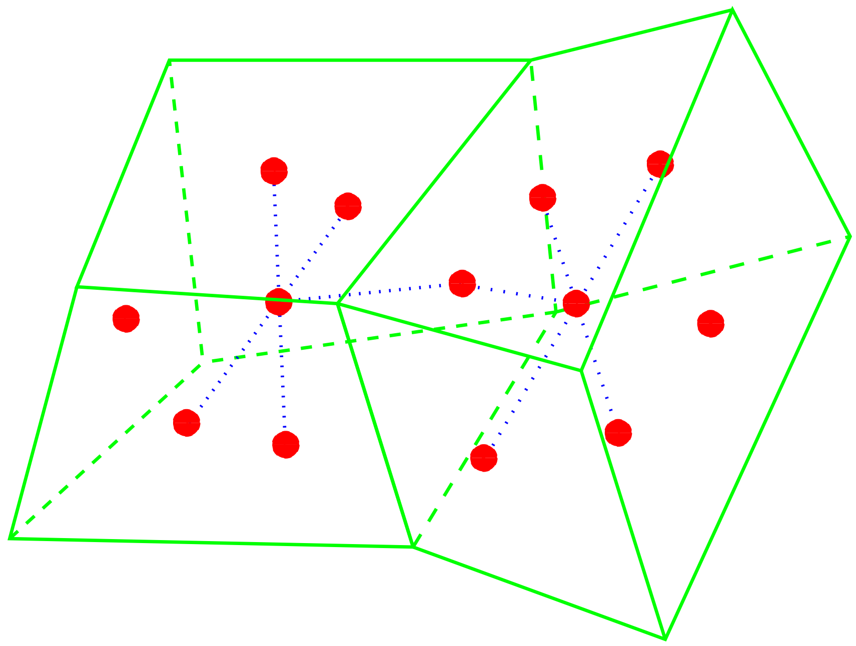
Discretization: Cell Face Equations

The cell center (conservation) equation has been discretized, but we have added 3 extra unknowns per cell.

At each cell face, we apply a continuity of flux condition:

$$-\overrightarrow{F^{n+1}_{c1,f}} \cdot \overrightarrow{A_{c1,f}} - \overrightarrow{F^{n+1}_{c2,f}} \cdot \overrightarrow{A_{c2,f}} = 0$$

where $c1$ and $c2$ are the two cells that share the face f.



Discretization: Boundary Conditions

The boundary conditions only affect the cell face equations. On the boundaries, a Robin boundary condition is specified:

$$\beta_1 \, \Phi_f^{n+1} - \beta_2 \, \overrightarrow{F_{c,f}^{n+1}} \cdot \overrightarrow{A_{c,f}} = \beta_3 \, \Phi_{bc}$$

where β_1, β_2 and β_3 can be specified to match

- homogeneous: $\Phi_f^{n+1} = 0,$
- reflective: $-\overrightarrow{F_{c,f}^{n+1}} \cdot \hat{n}_{c,f} = 0,$
- vacuum: $\frac{1}{2}\Phi_f^{n+1} - \overrightarrow{F_{c,f}^{n+1}} \cdot \hat{n}_{c,f} = 0,$
- Dirichlet: $\Phi_f^{n+1} = \Phi_{bc},$
- Neumann: $-\overrightarrow{F_{c,f}^{n+1}} \cdot \hat{n}_{c,f} = -\Phi_{bc}$, or
- source boundary conditions:

  $\frac{1}{2}\Phi_f^{n+1} - \overrightarrow{F_{c,f}^{n+1}} \cdot \hat{n}_{c,f} = \frac{1}{2}\Phi_{bc}.$

Algebraic Solution

Main Matrix System:

- Unsymmetric – must use unsymmetric solver
- Size is $(4n_c + n_b/2)$ squared
- Maximum of 11 non-zero elements per row

Preconditioner for Krylov Space methods is a Low-Order Matrix System:

- Assume orthogonal: drop out minor directions in flux terms
- Symmetric – can use standard CG solver
- Size is n_c squared
- Maximum of 7 non-zero elements per row

Method Summary

- Cell-centered, unstructured mesh
- Derivation valid for 1-D, 2-D, and 3-D geometries
- Preserves linear homogeneous solutions, second-order accurate
- Reduces to the standard cell-centered 7-point operator for an orthogonal mesh
- Local energy conservation is maintained
- Material discontinuities are rigorously treated
- Unsymmetric matrix system
- Solves for $(4n_c + n_b/2)$ unknowns, but only cell centers (n_c) remain between timesteps

Implementation:
The Augustus Code Package

Author:	Michael L. Hall (1/94 - present)
Architectures:	Sun (SunOS and Solaris), SGI (IRIX), HP (HP-UX), IBM (AIX)
Language:	Fortran-77, plans for Fortran-90
Solver Packages:	JTpack (by John Turner, LANL) for Krylov Space methods, UMFPACK (by Tim Davis, U of FL) for sparse direct methods
Installations:	SNLA ALEGRA hydrodynamics code, LANL TELLURIDE low-speed flow code, Solver for the Spartan SP_N radiation transport code.
Status:	Completed, active development of new features
Availability:	Email **`hall@lanl.gov`** and we'll talk

Implementation:
The Augustus Code Package

Spatial Mesh:

Dimension	Geometries	Type of Elements
1-D	spherical, cylindrical or cartesian	line segments
2-D	cylindrical or cartesian	quadrilaterals or triangles
3-D	cartesian	hexahedra or degenerate hexahedra (tetrahedra, prisms, pyramids)

all with an unstructured (arbitrarily connected) format.

Results: Second-Order Proof

- 3-D Random Mesh (del = .4)
- Constant properties, No removal
- Source = Qx^2
- Reflective boundaries on 4 sides
- Vacuum boundary conditions on opposite sides
- Analytic solution - Quartic:
  $\Phi(x, y, z) = \Phi(x) = a + bx + cx^4$

Results: Second-Order Proof

New Method:

Problem Size (cells)	$\frac{\lVert\Phi_{\text{exact}}-\Phi\rVert_2}{\lVert\Phi_{\text{exact}}\rVert_2}$	Error Ratio
$5 \times 5 \times 5$	$1.0248{\times}10^{-2}$	
$10 \times 10 \times 10$	$2.6190{\times}10^{-3}$	3.91
$20 \times 20 \times 20$	$6.6082{\times}10^{-4}$	3.96
$40 \times 40 \times 40$	$1.6530{\times}10^{-4}$	4.00

Orthogonal 7-Pt Solution:

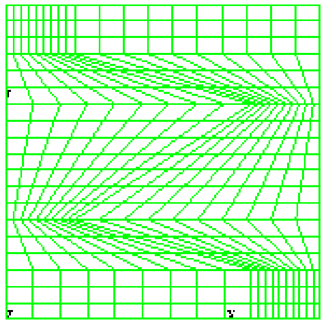
Problem Size (cells)	$\frac{\lVert\Phi_{\text{exact}}-\Phi\rVert_2}{\lVert\Phi_{\text{exact}}\rVert_2}$	Error Ratio
$5 \times 5 \times 5$	$1.0202{\times}10^{-2}$	
$10 \times 10 \times 10$	$2.6205{\times}10^{-3}$	3.92
$20 \times 20 \times 20$	$6.5952{\times}10^{-4}$	3.97
$40 \times 40 \times 40$	$1.6515{\times}10^{-4}$	3.99

Results: Sample Problem

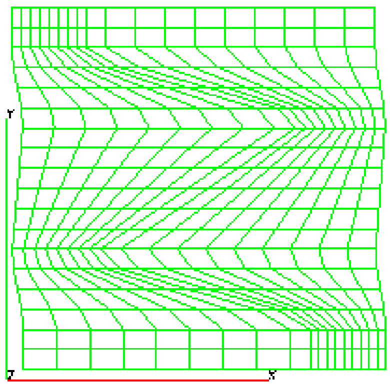
- 3-D Kershaw-Squared Mesh
- Constant properties
- No removal or sources
- Reflective boundaries on 4 sides
- Source and vacuum boundary conditions on opposite sides
- Analytic solution - linear
- Grid size - $20 \times 20 \times 20 = 8000$ nodes, 6859 cells
- 50 timesteps, 15 s / timestep on IBM RS/6000 Scalable POWERparallel System, SP2

Results: Sample Problem

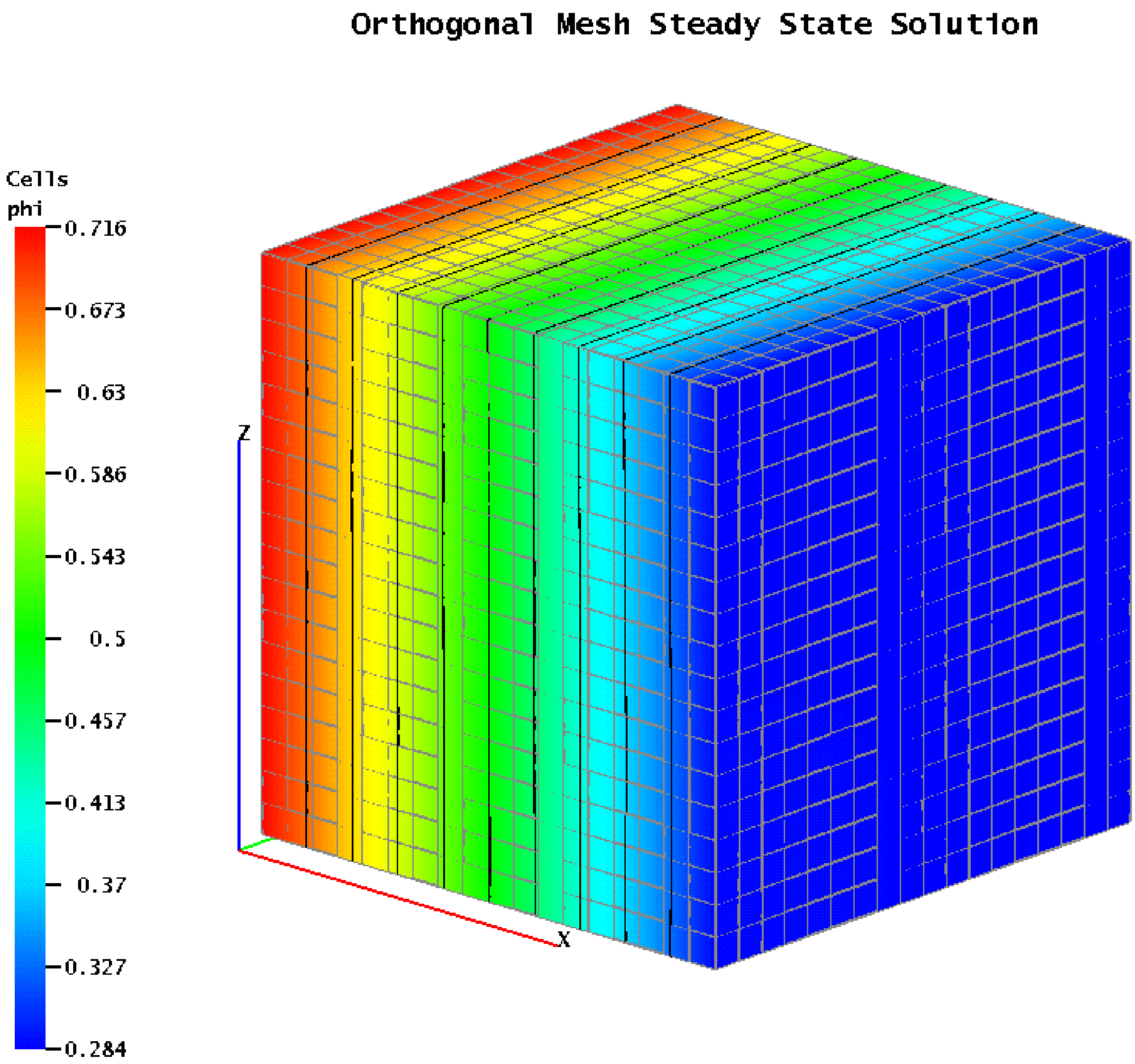
Actual Mesh (Cell Nodes)



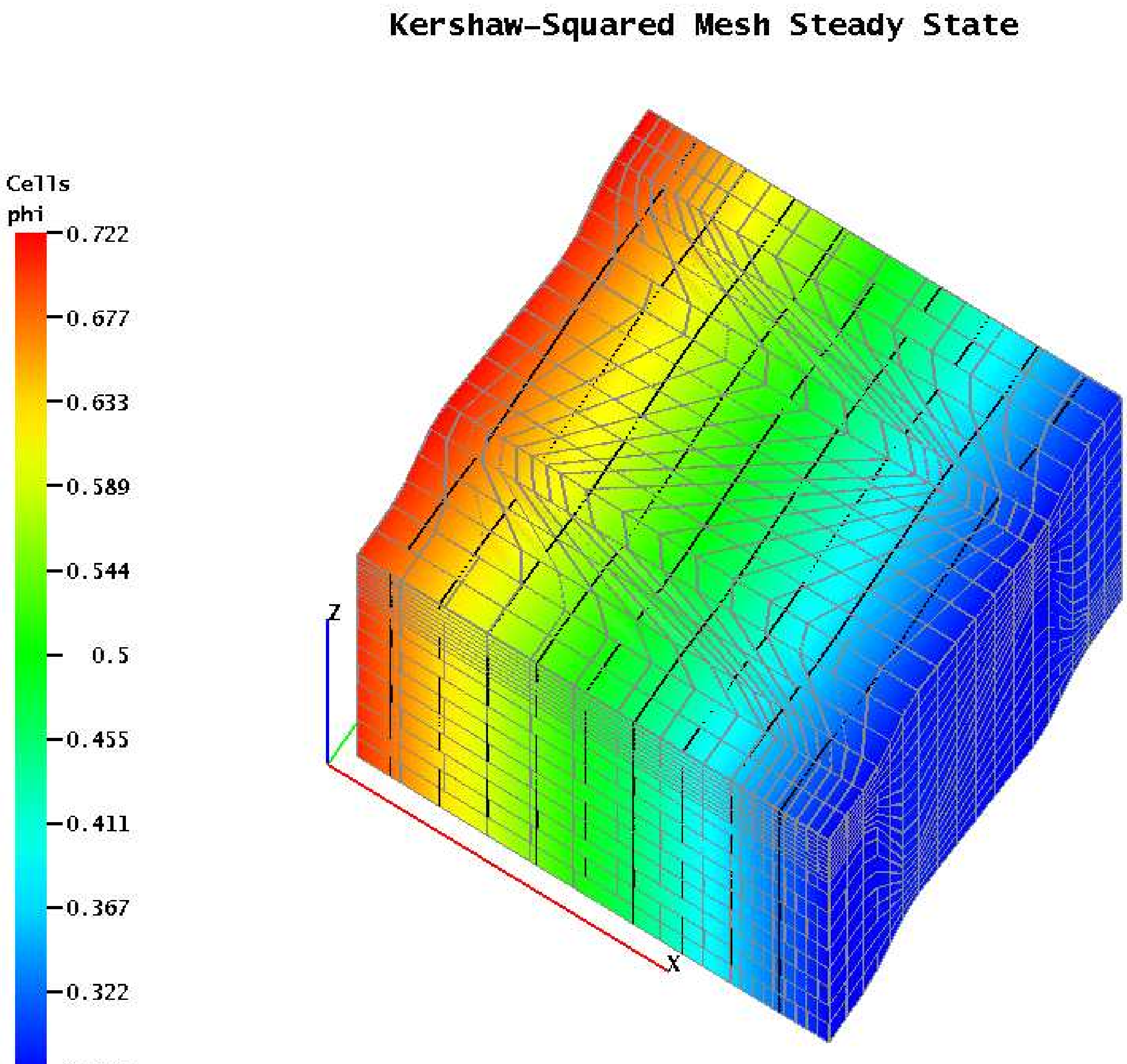
Dual Mesh (Cell Centers)



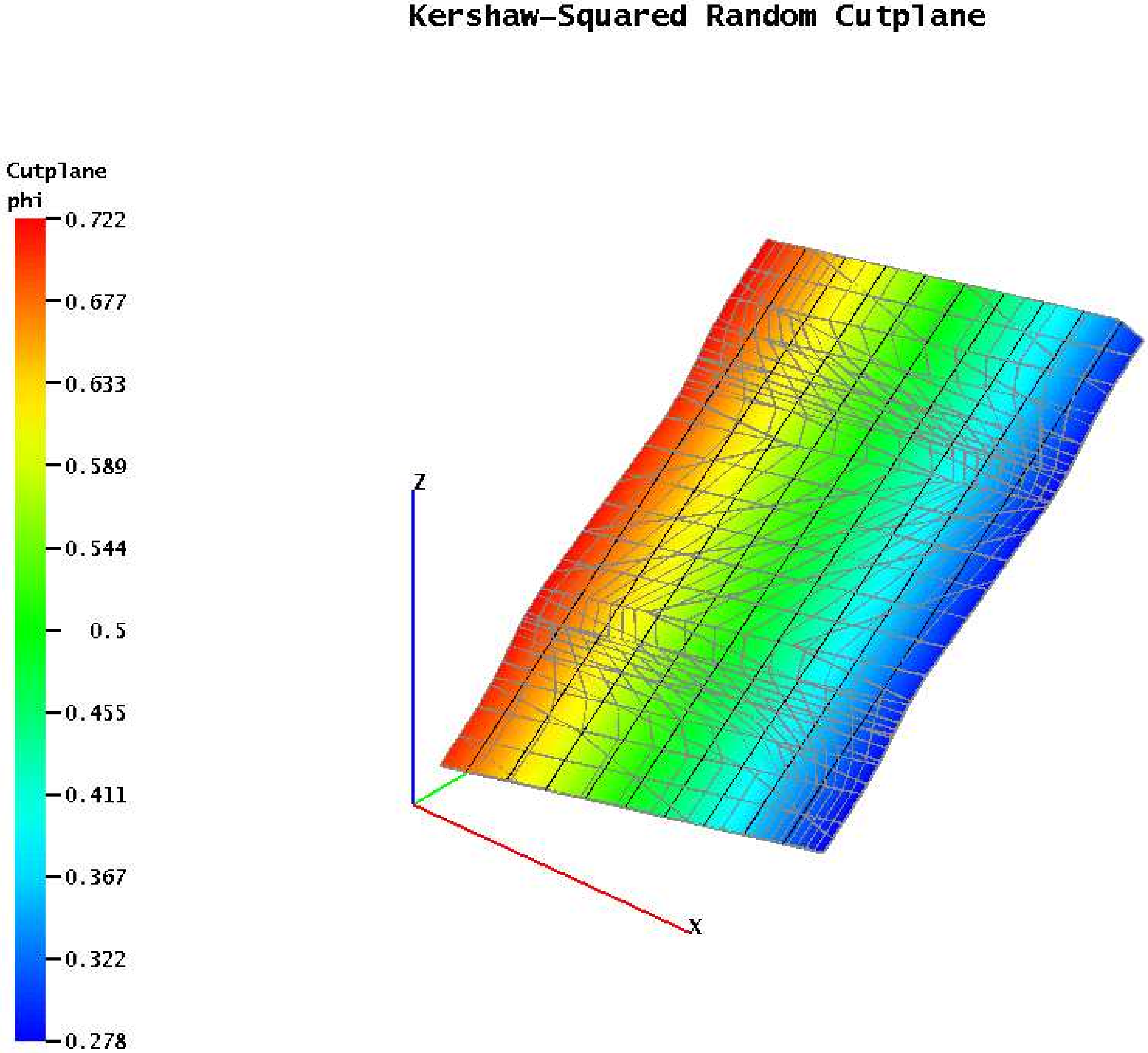
Results: Sample Problem



Results: Sample Problem



Results: Sample Problem



Future and Concurrent Work

- SPARTAN Code Package

- Support Operator Method

- 2-D Symmetric Method

- MHD Equations