**NE 255, Fa16**
**Equation Discretization**
**October 6, 2016**

---

So far we've dealt with

- Discretization of *time* using finite difference method (Taylor expand points and combine)
- Discretization of *energy* using the multigroup approximation, where we assume group-integrated values.
- *Expanding sources*, in particular scattering, in spherical harmonics–which we reduce to Legendre Polynomials in the case of azimuthal symmetry.
- Discretization of *angle* using either
  - $S_N$: get solutions along specific angle sets (quadrature points), use corresponding quadrature weights to integrate over angle
  - $P_N$: expand the angular flux in spherical harmonics, which we only do in 1-D so Legendre polynomials in practice, and solve a set of coupled equations for each expansion term (with closure relations at $n = 0$ and $n = N + 1$).
  - $SP_N$: we take the 1-D $P_N$ equations and transform them to 3-D by replacing the 1-D diffusion operators with the 3-D diffusion operator and replacing the derivatives at the boundary with the outward normal derivatives. We also replace $\phi_{l'}$ by a vector for odd $l'$.

When we do all of this we get $t = 0, \ldots, T$ equations in time, $g = 0, \ldots, G$ equations in energy, and a number of equations in angle that depends on which approach we take. However, we still have one major item to deal with...

# Space

(Largely from Evans, some from Vujíc and Lewis and Miller)

There are *many* spatial discretization choices out there. What you choose can depend on the geometry and physical properties, as well as if you're using Cartesian or curvilinear formulations. Fundamentally, we can characterize the differencing schemes in a few ways

- cell balance, which includes
  - Finite Difference Method (FDM) – using point value solution

  - Finite Volume Method (FVM) – using cell-averaged value solution
- finite element (FEM) – using basis function for expansion:
  - Piecewise linear: hat functions
  - Piecewise quadratic or cubic basis functions
  - Piecewise higher order Gauss-Legendre polynomials
- Spectral and Pseudo Spectral Methods – using orthogonal global series as the basis function:
  - Fourier series
  - Bessel, Chebyshev, Legendre series

We'll talk about cell balance and finite element methods; in nuclear we have specific versions of these. For example, Denovo, the 3-D Cartesian mesh deterministic code from ORNL, offers these choices:

- Simplified $P_N$: finite volume
- Discrete ordinates: weighted diamond difference (WDD) without flux fixup is equivalent to a Crank-Nicolson method; cell balance.
- Discrete ordinates: weighted diamond difference with flux fixup to zero (WDD-FF) is a nonlinear method; cell balance
- Discrete ordinates: theta weighted diamond difference (TWD) is a nonlinear method; cell balance
- Discrete ordinates: linear discontinuous (LD) is a Galerkin method formed from the basis set $\{1, x, y, z\}$; FEM
- Discrete ordinates: bilinear discontinuous (BLD) in 2-D; FEM
- Discrete ordinates: trilinear discontinuous (TLD) is a Galerkin method formed from the basis set $\{1, x, y, z, xy, yz, xz, xyz\}$ and maintains the asymptotic diffusion limit on the grid used in Denovo; FEM
- Discrete ordinates: step characteristics (SC) in 2- or 3-D does not produce negative fluxes and does not have oscillatory behavior; can be written as a cell balance or finite element scheme

The WDD, WDD-FF, TWD, LD, BLD, and TLD schemes are all second-order, and the SC scheme is first-order.

---

Aside

*Galerkin Methods*: Converts a continuous problem to a discrete problem - equivalent to converting the equation to a weak formulation. Typically one gives some constraints on the function space to characterize the space with a finite set of basis functions.

*weak form*: The main idea of the weak form is to multiply the entire governing equation by an arbitrary function, often referred to as a "weight function," and then integrate over the entire domain. Then, integration by parts is applied when possible to transfer some amount of the differentiation requirements on the solution to the weight function. The reason that this is performed is that we can use lower-order basis functions for expanding the solution than would have been possible with the un-modified equation, or the "strong form." For instance, in the equation $d^2u(x)/dx^2 = 0$, it is required that $u(x)$ be twice-differentiable. While physically we may have reason to believe that $u(x)$ should always be second-order (based on a particular physical application), the basis functions used in the finite element method are often only piecewise continuous, such that their second derivatives are not continuous. Integrating over the second derivatives would be similar to integrating a heaviside function, and could be undefined based on the region of the domain over which we're integrating. However, we still want to use these simple basis functions because they allow us to expend relatively little cost in evaluation of those functions, and their piecewise continuous nature means that we can force them to exist only over each finite element (this is a big deal - this allows us to have very sparse matrices in our problem). So, the weak form allows us to transfer half of the differentiations to the weight function, which then means that we only require $u(x)$ to be once-differentiable. So, the essential purpose of the weak form is to turn the differential equation into an integral equation and introduce an arbitrary function, the weight function, so as to lessen the burden on the numerical algorithm in evaluating derivatives of the solution.

A strong form of the governing equations along with boundary conditions states the conditions at every point over a domain that a solution must satisfy. On the other hand a weak form states the conditions that the solution must satisfy in an integral sense. (As an aside, weak formulations strictly require that the solution obey the Dirichlet boundary conditions, but not necessarily the Neumann boundary conditions). Examples of weak forms are variational formulations or weighted residual formulations.

Weak formulations naturally promote computing approximate solutions to challenging problems, and are 'equivalent' to strong forms.

---

LANL's PARTISN has similar choices, but their new Capsaicin code has unstructured mesh and uses Discontinuous Finite Element Method (DFEM): Linear, Bars, Triangles, Quadrilaterals, Polygons. They also have an ability to deal with non-convex meshes using Continuous FEM. Finally, they have structured meshes in 3-D with DD and LD.

INL's Rattle$S_N$ake uses finite element methods for unstructured higher-order meshes as well, and has capabilities for both discontinuous and continuous Galerkin.

We'll use the diagram in Figure 1 to think through our discretization schemes.

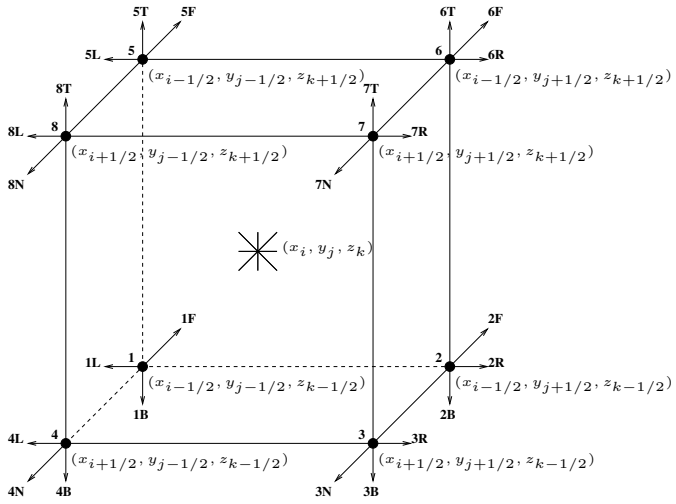


Figure 1: General mesh cell used to derive discrete spatial equations. The adjacent cell points are given using the notation $N \rightarrow +x$, $F \rightarrow -x$, $L \rightarrow -y$, $R \rightarrow +y$, $B \rightarrow -z$, and $T \rightarrow +z$.

For any given group, angle, and source, the transport equation can be reduced to

$$\hat{\Omega} \cdot \nabla \psi(\mathbf{r}) + \Sigma_t(\mathbf{r}) \psi(\mathbf{r}) = s(\mathbf{r}) \,, \tag{1}$$

where $s(\mathbf{r})$ is a total accumulated source. In operator form, this equation is

$$\mathbf{L}\psi = s \,, \tag{2}$$

where $\mathbf{L}$ is the differential transport operator ($\hat{\Omega} \cdot \nabla + \Sigma_t(\mathbf{r})$). We will be required to perform operations of the type

$$\psi = \mathbf{L}^{-1} s \,, \tag{3}$$

to solve discrete forms of Eq. (2).

An *operator* is a mapping from one vector space or module to another. Operators are of critical importance to both linear algebra and functional analysis, and they find application in many other fields of pure and applied mathematics.

For all the spatial differencing schemes discussed below, $\mathbf{L}$ can be *implicitly* formed as a lower-left triangular matrix and inverted by "sweeping" through the mesh in the direction of particle flow. In effect, the discretized form of Eq. (1) is solved in each cell. The outgoing fluxes become input

to the downwind cells, or in other words, each cell looks "upwind" to find its incoming fluxes. Once all the incoming fluxes are defined on the entering faces of a cell, the outgoing fluxes can be calculated, and the process is repeated until the entire mesh is solved for a given angle. For each cell, the entering and exiting faces are defined by

$$\hat{\Omega} \cdot \mathbf{n} < 0 \,, \quad \text{(entering face)} \tag{4}$$

$$\hat{\Omega} \cdot \mathbf{n} > 0 \,, \quad \text{(exiting face)} \,. \tag{5}$$

Mathematically, this is called a *wavefront* solver. The operation $\mathbf{L}^{-1}$ is regularly referred to as a *sweep* in the nuclear engineering and transport communities.

## Finite Difference

Finite Difference gives pointwise values on a grid.

We (quickly) covered the principles of finite difference when we did time discretization. We used points to approximate derivatives and were able to obtain a corresponding expression for the *Local Truncation Error* (LTE).

A note about *convergence*: the solution of the finite difference equations should converge to the true solution of the PDE as grid spacing (mesh size) goes to zero.

We're actually going to skip this formulation; I suspect you can figure it out.

## Finite Volume

Finite Volume methods approximate the average integral on a reference volume. This handles discontinuities much better–why might that be? If cell boundaries line up with material boundaries and we integrate half way into each cell, we capture the impact of the neighboring materials. Thus, the cell-balance equation can be derived by integrating Eq. (1) over the mesh cell in Fig. 1, which yields a statement of conservation of particles within the mesh cell.

**Weighted Diamond Difference**

To integrate the differential term, we will note

$$\int_{x_{i-1/2}}^{x_{i+1/2}} dx\, \frac{\partial \psi}{\partial x} = \int_{x_{i-1/2}}^{x_{i+1/2}} \partial\psi = \psi_{i+1/2} - \psi_{i-1/2}\,.$$

For the other terms we will use the *midpoint integration rule*. We won't derive that rule here, but know that the midpoint rule comes from open Newton Cotes with Lagrange polynomials (a way to make integration rules) using $n = 0$ (which uses one point only):

$$\int_a^b f(x)dx = \int_{x_{i-1/2}}^{x_{i+1/2}} f(x)dx = hf(x_i) + \frac{h^3}{3} f''(\xi_i)\,.$$

For us, $x_{i+1/2} - x_{i-1/2} = h \equiv \Delta_i$. Thus, applying $\iiint(\cdot)\, dxdydz$, dividing by differential volume, and separating flux into $x$, $y$, and $z$ differential components gives (letting $\Sigma_{ijk} \equiv \Sigma_{t,ijk}$),

$$\frac{\mu}{\Delta_i}(\psi_{i+1/2} - \psi_{i-1/2}) + \frac{\eta}{\Delta_j}(\psi_{j+1/2} - \psi_{j-1/2}) + \frac{\xi}{\Delta_k}(\psi_{k+1/2} - \psi_{k-1/2}) + \Sigma_{ijk}\psi_{ijk} = s_{ijk}\,. \quad (6)$$

Note that we now need a way to relate the center ($\psi_{ijk}$) and edge ($\psi_{n\pm1/2}$) fluxes to one another! The diamond-difference method (using Lewis & Miller and Exnihilo manual) is derived by closing Eq. (6) with a weighted average of the face-edge fluxes, which is equivalent to a Crank-Nicolson method in space. (note that Lewis & Miller uses a slightly different scheme that is mathematically equivalent)

$$\begin{aligned}
\psi_i &= \frac{1}{2}\big((1+\alpha_i)\psi_{i+1/2} + (1-\alpha_i)\psi_{i-1/2}\big) \\
&\quad -1 \le \alpha \le 1 \\
\psi_{i+1/2} &= \frac{2}{(1+\alpha_i)}\psi_{ijk} - \frac{(1-\alpha_i)}{(1+\alpha_i)}\psi_{i-1/2}\,, \quad \mu > 0\ (\psi_{i-1/2} \text{ is incoming}) \\
\psi_{i-1/2} &= \frac{2}{(1-\alpha_i)}\psi_{ijk} - \frac{(1+\alpha_i)}{(1-\alpha_i)}\psi_{i+1/2}\,, \quad \mu < 0\ (\psi_{i+1/2} \text{ is incoming})
\end{aligned}$$

The plus/minus depends on direction–going in $\mu > 0$ the $\psi_{i-1/2}$ is the flux incoming and $\psi_{i+1/2}$ is outgoing. This flips for $\mu < 0$. We will use $\bar{\psi}$ to indicate the incoming fluxes on each face to help us keep track of which direction we're going.

The $\alpha$ terms are weighting factors such that $\alpha = 0$ gives the classic diamond-difference equations (the central flux is the average of the incoming and outgoing). $\alpha = \pm 1$ gives the step-difference equations in which the center flux is determined entirely from the incoming flux. Setting $\alpha = \pm 1$

yields a first-order spatial differencing scheme. The default behavior of Denovo for WDD uses $\alpha = 0$, which gives the second-order diamond-difference method.

Substituting and rearranging the cell-balance equation with this closure yields the following system of equations,

$$
\begin{aligned}
&\mu \gtrless 0\ ,\ \eta \gtrless 0\ ,\ \xi \gtrless 0 \\
&\qquad \psi_{ijk} = \frac{s_{ijk} + \frac{2}{(1\pm\alpha_i)}\frac{|\mu|}{\Delta_i}\bar{\psi}_{i\mp 1/2} + \frac{2}{(1\pm\alpha_j)}\frac{|\eta|}{\Delta_j}\bar{\psi}_{j\mp 1/2} + \frac{2}{(1\pm\alpha_k)}\frac{|\xi|}{\Delta_k}\bar{\psi}_{k\mp 1/2}}{\Sigma_{ijk} + \frac{2}{(1\pm\alpha_i)}\frac{|\mu|}{\Delta_i} + \frac{2}{(1\pm\alpha_j)}\frac{|\eta|}{\Delta_j} + \frac{2}{(1\pm\alpha_k)}\frac{|\xi|}{\Delta_k}}\ , \\
&\qquad \psi_{i\pm 1/2} = \frac{2}{(1\pm\alpha_i)}\psi_{ijk} - \frac{(1\mp\alpha_i)}{(1\pm\alpha_i)}\bar{\psi}_{i\mp 1/2}\ , \\
&\qquad \psi_{j\pm 1/2} = \frac{2}{(1\pm\alpha_j)}\psi_{ijk} - \frac{(1\mp\alpha_j)}{(1\pm\alpha_j)}\bar{\psi}_{j\mp 1/2}\ , \\
&\qquad \psi_{k\pm 1/2} = \frac{2}{(1\pm\alpha_k)}\psi_{ijk} - \frac{(1\mp\alpha_k)}{(1\pm\alpha_k)}\bar{\psi}_{k\mp 1/2}\ .
\end{aligned}
\tag{7}
$$

Now, given incoming boundary conditions, we can march through our spatial mesh along each direction and solve for (1) the cell-center flux and then (2) the outgoing flux in each cell. We use the outgoing flux as the incoming flux for the next cell. Going through all of the angles in some sensible order is called a sweep–where we've inverted the streaming + interaction term.

Many codes, like Denovo, provide a version of WDD that **can correct negative fluxes**. When the outgoing flux is less than zero, we set the face-edge flux to zero and recalculate $\psi_{ijk}$ and the new edge fluxes. This process is repeated until all the outgoing fluxes are greater than or equal to zero. This method is nonlinear in that the corrected solution of Eq. (7) depends on $\psi$.

### Theta Weighted Diamond

Another nonlinear cell-balance scheme is TWD. This scheme uses the incoming fluxes to calculate weighting factors that permit the calculation of cell-centered and outgoing fluxes that vary smoothly between the step and diamond-difference approximations. The weighting factors are

calculated from the following system of equations,

$$\begin{aligned}
&\mu \gtrless 0\ ,\ \eta \gtrless 0\ ,\ \xi \gtrless 0 \\
&\qquad 1-a = \frac{s_{ijk}V\theta_s + (|\eta|B\bar{\psi}_{j\mp 1/2} + |\xi|C\bar{\psi}_{k\mp 1/2})\theta + |\mu|A\bar{\psi}_{i\mp 1/2}}{(\Sigma_{ijk}V + 2|\eta|B + 2|\xi|C)\bar{\psi}_{i\mp 1/2}}\ , \\
&\qquad 1-b = \frac{s_{ijk}V\theta_s + (|\mu|A\bar{\psi}_{i\mp 1/2} + |\xi|C\bar{\psi}_{k\mp 1/2})\theta + |\eta|B\bar{\psi}_{j\mp 1/2}}{(\Sigma_{ijk}V + 2|\mu|A + 2|\xi|C)\bar{\psi}_{j\mp 1/2}}\ , \\
&\qquad 1-c = \frac{s_{ijk}V\theta_s + (|\mu|A\bar{\psi}_{i\mp 1/2} + |\eta|B\bar{\psi}_{j\mp 1/2})\theta + |\xi|C\bar{\psi}_{k\mp 1/2}}{(\Sigma_{ijk}V + 2|\mu|A + 2|\eta|B)\bar{\psi}_{k\mp 1/2}}\ ,
\end{aligned} \tag{8}$$

where

$$A = \Delta_j\Delta_k\ , \quad B = \Delta_i\Delta_k\ , \quad C = \Delta_i\Delta_j\ , \quad V = \Delta_i\Delta_j\Delta_k\ . \tag{9}$$

The theta-weighting factors, $\theta$ and $\theta_s$, are set to values between 0 and 1. By default Denovo uses the *theta-weighted* model from the TORT code in which $\theta = \theta_s = 0.9$.

The weighting parameters are bounded between the diamond-difference and step approximations,

$$\frac{1}{2} \leq \{a, b, c\} \leq 1\ . \tag{10}$$

Using these factors, the cell-centered and outgoing fluxes are

$$\begin{aligned}
\psi_{ijk} &= \frac{s_{ijk}V + \frac{|\mu|A}{a}\bar{\psi}_{i\mp 1/2} + \frac{|\eta|B}{b}\bar{\psi}_{j\mp 1/2} + \frac{|\xi|C}{c}\bar{\psi}_{k\mp 1/2}}{\Sigma_{ijk}V + \frac{|\mu|A}{a} + \frac{|\eta|B}{b} + \frac{|\xi|C}{c}}\ , \\
\psi_{i\pm 1/2} &= \frac{1}{a}\psi_{ijk} - \frac{(1-a)}{a}\bar{\psi}_{i\mp 1/2}\ , \\
\psi_{j\pm 1/2} &= \frac{1}{b}\psi_{ijk} - \frac{(1-b)}{b}\bar{\psi}_{j\mp 1/2}\ , \\
\psi_{k\pm 1/2} &= \frac{1}{c}\psi_{ijk} - \frac{(1-c)}{c}\bar{\psi}_{k\mp 1/2}\ .
\end{aligned} \tag{11}$$

TWD and WDD-FF produce uniformly positive fluxes when the source, $s_{ijk}$, is greater than zero.

## Finite Element

Finite element schemes are derived from the weak form of Eq. (1). We begin the derivation by integrating Eq. (1) over a single element, $e$, and multiplying by a weighting function for the element,

$w_{e\,n}$,

$$\int_{V_e} w_{e\,n}(\hat{\Omega}\cdot\nabla\psi + \Sigma_t\psi)\,dV = \int_{V_e} w_{e\,n} s\,dV\ , \tag{12}$$

where the element defined in Fig. 1 has $dV = dxdydz$. The weight functions are defined over the range $n \in [1, N]$ such that the set $w_{e\,n}$ is linearly independent. Furthermore, we make the assumption that all elements in the orthogonal grid have equal weight functions, so $w_{e\,n} \equiv w_n$. Applying Green's theorem to the gradient term gives the weak form of the transport equation,

$$\oint_{\partial V_e} w_n \hat{\mathbf{n}}\cdot\hat{\Omega}\psi\,dA - \int_{V_e}\hat{\Omega}\cdot\nabla w_n\psi\,dV + \int_{V_e} w_n\Sigma_t\psi\,dV = \int_{V_e} w_n s\,dV\ , \quad n = 1,\ldots,N\ . \tag{13}$$

Now, we expand the angular flux in the following basis,

$$\psi = \sum_{m=1}^{N} b_m(\mathbf{r})\psi_e^{(m)}\ , \quad \mathbf{r}\in\partial V_e\ . \tag{14}$$

---

Aside: In a continuous Galerkin finite element method, the variable $u = u(x)$ is approximated globally in a (piecewise linear) continuous manner. In contrast, in a discontinuous Galerkin finite element method, the variable $u = u(x)$ is approximated globally in a discontinuous manner and locally in each element in a (piecewise linear) continuous way. What we're doing is discontinuous Galerkin.

With discontinuous Galerkin methods, continuity at the nodes is *not* enforced. This leads to a much greater number of unknowns when compared to the continuous Galerkin method. For instance, consider four Cartesian mesh cells that meet at a node. At this node, with the discontinuous Galerkin method, we have four unknkowns. With the continuous Galerkin method, however, because continuity at this node is required, there is only one unknown. The difference in the number of unknowns is even greater for higher-order basis functions. So why use the discontinuous Galerkin method? One big advantage is (* I think *) that it allows us to formulate our problem in terms of a transport sweep that is now familiar to us from the finite difference methods we showed previously. With the continuous Galerkin method, you cannot sweep through the mesh because the solution over each individual mesh cell depends on the solutions over the adjacent cells, and a sweep would effectively mean that you're trying to solve each cell alone. This is equivalent to trying to solve a singular system, and is not a valid solution approach. With the continuous Galerkin method, a large matrix system $\mathbf{A}x = b$ is solved, where this matrix system applies to the entire mesh, and not to a single cell. On the other hand, with the discontinuous Galerkin method, the solution over each cell can be phrased in terms of a transport sweep because continuity is not

imposed at nodes. Phrasing a finite element method in terms of a transport sweep allows the ability to re-use code already developed for the finite differencing sweeps.

---

Applying the Galerkin finite element approximation in which $w_n = b_n$, Eq. (13) becomes

$$\oint_{\partial V_e} b_n \hat{\mathbf{n}} \cdot \hat{\Omega} \psi \, dA - \mathcal{T}\Psi + \Sigma_e \mathcal{M}\Psi = \mathcal{M}S \,, \tag{15}$$

where the elements of the matrices and vectors are defined

$$\begin{aligned} [\mathcal{T}]_{nm} &= \int_{V_e} \hat{\Omega} \cdot \nabla b_n b_m \, dV \,, \\ [\mathcal{M}]_{nm} &= \int_{V_e} b_n b_m \, dV \,, \\ \Psi &= \begin{pmatrix} \psi_e^{(1)} & \psi_e^{(2)} & \cdots & \psi_e^{(N)} \end{pmatrix}^T \,, \\ S &= \begin{pmatrix} s_e^{(1)} & s_e^{(2)} & \cdots & s_e^{(N)} \end{pmatrix}^T \,, \end{aligned} \tag{16}$$

and the source has been expanded in the same basis as the angular flux. The angular fluxes in the surface term come from upwind cells or boundary conditions.

The size and composition of the matrices in Eq. (15) are dependent on the mesh and the basis functions chosen to represent the unknowns.

### Linear Discontinuous

The LD method defines a basis over the set $\{1, x, y, z\}$ with the shape functions

$$b_1 = 1 \,, \quad b_2 = \frac{2(x - x_{ijk})}{\Delta_i} \,, \quad b_3 = \frac{2(y - y_{ijk})}{\Delta_j} \,, \quad b_4 = \frac{2(z - z_{ijk})}{\Delta_k} \,. \tag{17}$$

Using these shape functions in Eq. (15) and analytically evaluating the integrals give the following system of equations,

$$\begin{aligned}
\frac{\mu}{\Delta_i}(\psi_{i+1/2}-\psi_{i-1/2})+\frac{\eta}{\Delta_j}(\psi_{j+1/2}-\psi_{j-1/2})+\frac{\xi}{\Delta_k}(\psi_{k+1/2}-\psi_{k-1/2})+\Sigma_t\psi^a &= s^a\ ,\\
\frac{3\mu}{\Delta_i}(\psi_{i+1/2}+\psi_{i-1/2}-2\psi^a)+\frac{\eta}{\Delta_j}(\psi^x_{j+1/2}-\psi^x_{j-1/2})+\frac{\xi}{\Delta_k}(\psi^x_{k+1/2}-\psi^x_{k-1/2})+\Sigma_t\psi^x &= s^x\ ,\\
\frac{\mu}{\Delta_i}(\psi^y_{i+1/2}-\psi^y_{i-1/2})+\frac{3\eta}{\Delta_j}(\psi_{j+1/2}+\psi_{j-1/2}-2\psi^a)+\frac{\xi}{\Delta_k}(\psi^y_{k+1/2}-\psi^y_{k-1/2})+\Sigma_t\psi^y &= s^y\ ,\\
\frac{\mu}{\Delta_i}(\psi^z_{i+1/2}-\psi^z_{i-1/2})+\frac{\eta}{\Delta_j}(\psi^z_{j+1/2}-\psi^z_{j-1/2})+\frac{3\xi}{\Delta_k}(\psi_{k+1/2}+\psi_{k-1/2}-2\psi^a)+\Sigma_t\psi^z &= s^z\ .
\end{aligned} \tag{18}$$

Here,

$$\psi_e^{(1)}=\psi^a\ ,\quad \psi_e^{(2)}=\psi^x\ ,\quad \psi_e^{(3)}=\psi^y\ ,\quad \psi_e^{(4)}=\psi^z\ , \tag{19}$$

such that $\psi^a$ is the average angular flux in the element and $\psi^{x|y|z}$ are the slopes in the $(x,y,z)$ directions. The same notation is used for the expanded source, $s$.

Equations (14) and (17) are used to develop upwind expressions for $\psi$,

$$\psi_{i\pm1/2}=\psi^a\pm\psi^x\ ,\quad \psi_{j\pm1/2}=\psi^a\pm\psi^y\ ,\quad \psi_{k\pm1/2}=\psi^a\pm\psi^z\ ,\quad \{\mu,\eta,\xi\}\gtrless 0\ , \tag{20}$$

and the upwinded slopes are

$$\psi^{(y|z)}_{i\pm1/2}=\psi^{(y|z)}\ ,\quad \psi^{(x|z)}_{j\pm1/2}=\psi^{(x|z)}\ ,\quad \psi^{(x|y)}_{k\pm1/2}=\psi^{(x|y)}\ . \tag{21}$$

For each direction, these expressions are inserted into Eq. (18), and the resulting $4\times4$ system is solved for $\Psi$. For efficiency, the solution to the matrix is precomputed, and $\Psi$ is calculated by evaluating four statements (**four** unknowns per cell).

### TriLinear Discontinuous

The TLD method is the only spatial discretization that maintains the asymptotic diffusion limit on the grid used. The TLD equations are derived by expanding $\Psi$ in the basis

$$\{1,x,y,z,xy,yz,xz,xyz\}\ .$$

thus, there are **eight** unknowns per cell. We define the unknowns at each node of the cell as indicated by the cardinal numbers in Fig. 1. The basis functions are

$$
\begin{aligned}
b_1 &= \left(\frac{x_{i+1/2}-x}{\Delta_i}\right)\left(\frac{y_{j+1/2}-y}{\Delta_j}\right)\left(\frac{z_{k+1/2}-z}{\Delta_k}\right), \\
b_2 &= \left(\frac{x_{i+1/2}-x}{\Delta_i}\right)\left(\frac{y-y_{j-1/2}}{\Delta_j}\right)\left(\frac{z_{k+1/2}-z}{\Delta_k}\right), \\
b_3 &= \left(\frac{x-x_{i-1/2}}{\Delta_i}\right)\left(\frac{y-y_{j-1/2}}{\Delta_j}\right)\left(\frac{z_{k+1/2}-z}{\Delta_k}\right), \\
b_4 &= \left(\frac{x-x_{i-1/2}}{\Delta_i}\right)\left(\frac{y_{j+1/2}-y}{\Delta_j}\right)\left(\frac{z_{k+1/2}-z}{\Delta_k}\right), \\
b_5 &= \left(\frac{x_{i+1/2}-x}{\Delta_i}\right)\left(\frac{y_{j+1/2}-y}{\Delta_j}\right)\left(\frac{z-z_{k-1/2}}{\Delta_k}\right), \\
b_6 &= \left(\frac{x_{i+1/2}-x}{\Delta_i}\right)\left(\frac{y-y_{j-1/2}}{\Delta_j}\right)\left(\frac{z-z_{k-1/2}}{\Delta_k}\right), \\
b_7 &= \left(\frac{x-x_{i-1/2}}{\Delta_i}\right)\left(\frac{y-y_{j-1/2}}{\Delta_j}\right)\left(\frac{z-z_{k-1/2}}{\Delta_k}\right), \\
b_8 &= \left(\frac{x-x_{i-1/2}}{\Delta_i}\right)\left(\frac{y_{j+1/2}-y}{\Delta_j}\right)\left(\frac{z-z_{k-1/2}}{\Delta_k}\right).
\end{aligned}
\tag{22}
$$

Substituting these into Eq. (15) and evaluating the integrals analytically, we derive the TLD equations for the cells. We are skipping this because it's a mess and waaay more than we need to go through in this class.

The order of error for LD compared to TLD is the same on bricks–with other mesh shapes it could be different (I *think*). However, TLD preserves the asymptotic diffusion limit whereas LD does not. That is the principal advantage. TLD should also have a lower coefficient, i.e.

LD $\propto a(\Delta x)^2$
TLD $\propto b(\Delta x)^2$

and $a > b$. TLD should also preserve the asymptotic diffusion limit at boundaries whereas LD will not. Finally, TLD should do better at sharp boundary discontinuities compared to LD.

The rate of convergence for the *continuous* Galerkin method is:

$$
\|u - u^h\|_{H^1\Omega} \leq ch^{\min(r-1,p)} \tag{23}
$$

where $H^1$ indicates the Hilbert-space norm, $c$ is an unknown constant, $h$ is the mesh spacing, $r$ is the regularity, and $p$ is the polynomial order of the approximating function. This theorem comes from functional analysis, and applies for *any* mesh and *any* element type (i.e. shape of element).

The regularity of smooth functions is $\infty$, and the lowest theoretical value of $r$ is 3/2, but for the types of problems we're going to be solving the minimum in the exponent of $h$ is going to be dictated by $p$. Hence, in the $H^1$ norm, linear elements give first-order convergence, quadratic elemetns second-order, and so on. When measured in the $L^2$ norm, on the other hand, the error is proportional to $h^{p+1}$, and linear elements are second-order, quadratic are third-order, and so on. If this same theorem applies to the discontinuous finite element method (* I think it would, though I haven't looked at the derivation of the theorem above to see if at any point it assumes continuity at the nodes *), then this explains why all the LD, BLD, and TLD methods are second-order convergent, since their basis functions are all linear in $x$, $y$, and $z$ (note that a basis function $xyz$ is still considered to be linear because $x$, $y$, and $z$ do not appear to powers greater than 1). Higher-order methods would achieve better convergence, and hence the finite element method is not always second-order convergent.

## Step Characteristics

The primary advantage of the SC scheme is that it produces uniformly positive solutions. SC gives positive fluxes as long as the source is positive, and it does not require non-linear flux fix-ups like WDD-FF or TWD. Also, it does not suffer from oscillatory behavior like step differencing. Because of these properties, SC is an ideal choice for calculating adjoint importance maps for use in hybrid Monte Carlo methods. However, if very high accuracy is required, LD and TLD are better choices because they are second-order methods, whereas SC is first-order.

The SC spatial discretization can be derived using the cell-balance or finite element formalism. We will use the slice-balance method.First, define a minimum step size through the cell defined in Fig. 1,

$$d = \min\left(\frac{\Delta_i}{|\mu|}, \frac{\Delta_j}{|\eta|}, \frac{\Delta_k}{|\xi|}\right) . \tag{24}$$

We now define a new set of indices, $\{i, j, k\} \rightarrow \{p, q, r\}$, such that $p$ is associated with the index of $d$, and $\{q, r\}$ are associated with the remaining indices. The slice fluxes are defined

$$\begin{aligned}
\psi_0^m &= \bar{s} + \mathrm{e}^{-\Sigma_t d}(\psi_{\text{inc}}^m - \bar{s}) , \\
\psi_1^m &= \bar{s} + \frac{1}{\Sigma_t d}(\psi_{\text{inc}}^m - \psi_0^m) , \\
\psi_2^m &= \bar{s} + \frac{2}{\Sigma_t d}(\psi_{\text{inc}}^m - \psi_1^m) , \\
m &= p, q, r ,
\end{aligned} \tag{25}$$

and $\bar{s} = s/\Sigma_t$. For each slice a normalized distance is defined,

$$\delta_m = \left( \frac{|\hat{\Omega}|_m}{\Delta_m} \right) d \,, \tag{26}$$

such that $\delta_p = 1$ and $\delta_{q,r} \leq 1$. The areas of each slice are

$$A_{pq} = \frac{\delta_p \delta_r}{2} \,, \quad B_{pq} = \delta_p (1 - \delta_r) \,, \quad C_{pp} = (1 - \delta_q)(1 - \delta_r) \,. \tag{27}$$

By using these definitions, the outgoing fluxes are

$$\begin{aligned} \psi_p &= A_{qp} \psi_2^q + A_{rp} \psi_2^r + B_{qp} \psi_1^q + B_{rp} \psi_1^r + C_{pp} \psi_0^p \,, \\ \psi_q &= A_{pq} \psi_2^p + A_{rq} \psi_2^r + B_{pq} \psi_1^p \,, \\ \psi_r &= A_{pr} \psi_2^p + A_{qr} \psi_2^q + B_{pr} \psi_1^p \,. \end{aligned} \tag{28}$$

Finally, the cell-centered flux is

$$\psi = \bar{s} + \frac{1}{\Sigma_t d} \sum_m \delta_m (\psi_{\text{inc}}^m - \psi_m) \,, \quad m = p, q, r \,. \tag{29}$$

Because these equations depend inversely on $\Sigma_t d$, we need to handle cases where $\Sigma_t d \ll 1$, which includes vacuum regions. In Denovo, when $\Sigma_t d \leq 0.025$ we expand $\exp(-\Sigma_t d)$ in a $O(7)$ Taylor-series. Using the expansion in Eqs. (25) through (29) yields formulas for the outgoing and cell-centered flux that do not vary inversely proportional to $\Sigma_t d$.