THE PIECEWISE LINEAR DISCONTINUOUS FINITE ELEMENT METHOD APPLIED TO THE RZ AND XYZ TRANSPORT EQUATIONS

A Dissertation

by

TERESA S. BAILEY

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2008

Major Subject: Nuclear Engineering

THE PIECEWISE LINEAR DISCONTINUOUS FINITE ELEMENT METHOD APPLIED TO THE RZ AND XYZ TRANSPORT EQUATIONS

A Dissertation

by

TERESA S. BAILEY

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Marvin Adams
Committee Members,	Jim Morel
	Warren Miller
	Jean-Luc Guermond
	Michael Zika
Head of Department,	Raymond Juzaitis

May 2008

Major Subject: Nuclear Engineering

ABSTRACT

The Piecewise Linear Discontinuous Finite Element Method Applied to the RZ and XYZ Transport Equations. (May 2008)
Teresa S. Bailey, B.S., Oregon State University; M.S., Texas A&M University
Chair of Advisory Committee: Dr. Marvin Adams

In this dissertation we discuss the development, implementation, analysis and testing of the Piecewise Linear Discontinuous Finite Element Method (PWLD) applied to the particle transport equation in two-dimensional cylindrical (RZ) and three-dimensional Cartesian (XYZ) geometries. We have designed this method to be applicable to radiative-transfer problems in radiation-hydrodynamics systems for arbitrary polygonal and polyhedral meshes. For RZ geometry, we have implemented this method in the Capsaicin radiative-transfer code being developed at Los Alamos National Laboratory. In XYZ geometry, we have implemented the method in the Parallel Deterministic Transport code being developed at Texas A&M University.

We discuss the importance of the thick diffusion limit for radiative-transfer problems, and perform a thick diffusion-limit analysis on our discretized system for both geometries. This analysis predicts that the PWLD method will perform well in this limit for many problems of physical interest with arbitrary polygonal and polyhedral cells. Finally, we run a series of test problems to determine some useful properties of the method and verify the results of our thick diffusion limit analysis.

Finally, we test our method on a variety of test problems and show that it compares favorably to existing methods. With these test problems, we also show that our method performs well in the thick diffusion limit as predicted by our analysis. Based on PWLD's solid finite-element foundation, the desirable properties it shows under

analysis, and the excellent performance it demonstrates on test problems even with highly distorted spatial grids, we conclude that it is an excellent candidate for radiative-transfer problems that need a robust method that performs well in thick diffusive problems or on distorted grids.

For my father, Lee Bailey

ACKNOWLEDGEMENTS

First, I would like to thank Dr. Marvin Adams for being an excellent graduate advisor. Dr. Adams has exhibited much patience in the process of educating me about computational transport, and it has truly been an honor to work with him on this research. I also am appreciative of my committee members. I have learned a great deal from Dr. Jim Morel and Dr. Jean-Luc Guermond, and I appreciate the interest that Dr. Warren Miller and Dr. Michael Zika have shown in this research. Also, I am grateful to Dr. Ragusa for acting as a substitute committee member during my defense.

I acknowledge the Computational Science Graduate Fellowship and Lawrence Livermore National Laboratory for completely funding this dissertation work. I am exceedingly fortunate to have received support from both of these organizations.

The first part of the research was performed during a summer practicum at Los Alamos National Laboratory. I am grateful to Jae Chang, Jim Warsa, and Kelly Thompson for helping me navigate the Capsaicin code and teaching me many practical things about methods development. The second part of this research was implemented in a code being developed at Texas A&M. I am grateful to Tim Smith and Gabriel Tanase of the Computer Science department for their willingness to help me learn the code and resolve programming issues.

At Texas A&M University, I have relied on the friendship and expertise of many fellow graduate students to survive the graduate school experience. I am particularly grateful to Amy Maslowski, Alex Maslowski, Cable Kurwitz, Kevin Clarno, Josh Jarrell, Trey Johansen, and Melanie Sarzynski for many illuminating discussions on a variety of useful topics.

I am also thankful for the encouragement and advice I have received from the Montague family, the Putnam family, the Sarzynski family, Dawna and Jeremy Vicars, Thea and John Browning, Natalie Richards, the Edman family, and the people at Pleasanton Presbyterian Church. I do not think I would have completed this degree without the influence of these people in my life.

Most importantly, I am grateful for the support of my family throughout this process. My father, Lee Bailey, is sometimes more fascinated by my research than I am. My mother, Jeanette Bailey, has always been willing to help me accomplish whatever I am working toward. I appreciate my parents' obvious sacrifices, continual encouragement, and good advice. I am grateful to my sister Lenore for always taking care of me, and always being interested in what I am doing. My sister Becky is exceptionally wise, and I appreciate her ability to understand me. I am extremely grateful for my sister Katie for reasons too numerous to list. Katie fills in the gaps in my knowledge and experience and makes my life easy. Finally, I am thankful to God for providing me with the opportunity to finish this research.

TABLE OF CONTENTS

CHAPTER Page

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

CHAPTER I
INTRODUCTION

Description of the problem

In this research, we introduce Piecewise Linear Discontinuous Finite Element Discretizations of the RZ and XYZ geometry linear Boltzmann transport equations for arbitrary polygonal and polyhedral grids. The linear Boltzmann transport equation is a mathematical statement of particle conservation, which we generally write as

$$\frac{1}{v(E)}\frac{\partial \psi\left(\vec{r},\vec{\Omega},E,t\right)}{\partial t}+\vec{\Omega}\bullet\vec{\nabla}\psi\left(\vec{r},\vec{\Omega},E,t\right)+\sigma\left(\vec{r},E,t\right)\psi\left(\vec{r},\vec{\Omega},E,t\right)=S\left(\vec{r},\vec{\Omega},E,t\right). \quad (1.1)$$

To make this problem well-posed, we define boundary conditions on the surface of the domain

$$\psi\left(\vec{r}_s,\vec{\Omega},E,t\right)=F\left(\vec{r}_s,\vec{\Omega},E,t\right),\ \vec{n}\bullet\vec{\Omega}<0, \quad (1.2)$$

where F is some known function, $\vec{n}$ is the outward going normal of the surface of the domain, and $\vec{r}_s$ is some point on the surface of the problem domain. We have assumed that the surface of the problem domain is not re-entrant, which means that once a particle leaves the system by streaming across the domain surface, it cannot re-enter the problem unless the boundary conditions specify its re-entrance. We also define an initial condition

$$\psi\left(\vec{r},\vec{\Omega},E,t\right)=\psi_0\left(\vec{r},\vec{\Omega},E\right), \quad (1.3)$$

where ψ_0 is the initial distribution of the particles inside the problem domain [1].

This dissertation follows the style of Journal of Computational Physics.

The fundamental unknown, ψ, in Eq. (1.1) is known as the angular flux or angular intensity. In some fields of study the angular intensity is written as I. We will use both

sets of terminology interchangeably in this dissertation. The angular intensity is dependent on three spatial coordinates ($\vec{r}$), two directional coordinates ($\vec{\Omega}$), energy (E), and time (t). Physically, the angular flux can be interpreted as the particle path-length-rate density at a spatial coordinate $\vec{r}$, along a specific direction $\vec{\Omega}$, at a specific energy E, at time t. Each term in Eq. (1.1) represents a physical process. The first term is the time rate of change of the angular flux; the second term on the left hand side of the equation represents the particles streaming through space in straight lines without interacting; the third term on the left hand side of the equation describes the total interaction rate; and the term on the right hand side of the equation is a general particle source term that is dependent on the physical system. Our transport equation is a statement of particle conservation in six dimensional phase space, where the angular intensity is a density quantity in that phase space.

Because of the complexity of the equation and the large number of independent variables, analytic solutions exist only for simple problems. For this reason, we use numerical techniques to produce approximate solutions for problems of practical interest. However, numerical solutions of the transport equation have proven to be computationally expensive both in the time required to solve the problems and the memory required to store the solution. Any new solution techniques applied to the transport equation should retain or improve the accuracy, speed, and memory footprint of existing methods for the calculations of interest.

This form of the Boltzmann equation written above has many assumptions built into it. We will discuss five of these assumptions. First, we assume that the particles we are describing do not interact with each other, but only with electrons and nuclei in the background material. This assumption is generally true because the number of particles

in a system is generally many orders of magnitude less than the number of atoms in the background material. This assumption makes the Boltzmann equation "linear." A second assumption we make in Eq. (1.1) is that the particles are not influenced by any outside forces such as gravity, magnetic, or electric fields, which is why the particles travel in straight lines between collisions. This assumption limits us to describing neutral particles such as neutrons and photons, or charged particles in the absence of significant fields. Third, we assume that the quantum mechanical wave properties are negligible compared to the distances that particles travel between collisions. As a result, we can treat particles as having a velocity and a position defined by a point location. Our fourth assumption is that the background material with which particles interact is moving isotropically. This means that a particle sees the same distribution of atoms moving away from it as it sees moving towards it. In short, the background material has no preferred direction of travel with respect to the particles we are transporting. (Later we discuss procedures for treating problems in which the background material does have significant bulk motion.) Finally, we assume that all particle interactions occur instantaneously, which means that any emission of a particle due to an interaction (such as a fission or scattering event in a nuclear reactor) occurs immediately following the interaction. A few physically significant processes exist that are exceptions to this assumption, such as the emission of delayed neutrons from fission products in nuclear reactors; however, this assumption holds true for most interactions [2]. This list of assumptions is not exhaustive, but covers most major physical limitations built into Eq. (1.1), and these assumptions provide guidelines to the types of problems we can model with the linear Boltzmann equation.

As we have previously implied, the linear Boltzmann transport equation is often used to model neutron populations in nuclear reactors and other systems involving nuclear fission events. Having high-fidelity solutions to the transport equation allows the nuclear engineering community to develop accurate power profiles inside nuclear reactor cores, optimize the use of the nuclear fuel, determine kinetics parameters to safely

operate the reactor, and design shielding around reactor plants. Furthermore, the transport equation is useful in modeling spent fuel storage configurations to prevent inadvertent criticality events. The research we are presenting here can certainly be applied to nuclear reactor systems involving neutrons, but we did not intend our research for this purpose.

A second, emerging field of study that uses the transport equation is the medical physics field. Many medical imaging techniques use photons as a non-destructive way to examine the inside of biological systems. Also, radiation therapy is an established form of treatment for many diseases. The transport equation can model these systems, and if we can solve this equation efficiently, we can help to optimize imaging and therapy. Again, the methods we develop in this dissertation can be applied to these systems, but we did not design our methods for this system.

Other systems that can be modeled using the transport equation are photon interactions in atmospheric systems, neutron and photon interactions in oil well logging systems, and other industrial applications.

The physical application we are interested in modeling with the methods we have developed is radiative transfer, which is important in inertial confinement fusion systems, astrophysics, and many laboratory settings involving plasmas. Absorption and emission of radiation in the form of photons is a key mechanism by which energy is transferred in these systems. In order to model these systems, we couple the radiative transfer equations to hydrodynamics equations (such as the Euler or Navier-Stokes equations), which describe the motion of the background material, along with matter energy-balance and possibly other equations. This background material is usually in a plasma state and these systems typically involve high energy densities (HED). In some parts of these HED systems the material is optically thick, meaning the photon opacities are very large, causing the photons to have a very short average distance between

collisions, known as a mean free path. As a result, energy and momentum are exchanged between the photons and the background material causing a significant impact on the behavior of both the radiation intensity and the material motion and properties. In order to model this energy and momentum exchange between the material and the radiation, we have to couple the radiative transfer equations with the equations describing the material behavior. During this coupling, it is advantageous to discretize both systems on the same spatial grid. Furthermore, because the photons have a very short mean free path in parts of the problem, we cannot always resolve this mean free path with our meshes. As a result, we require that our numerical methods recover an asymptotic limit known as the diffusion limit, which will produce accurate numerical results for cells containing many mean free paths.

The focus of this research is the development, implementation and characterization of new spatial discretizations for the radiative transfer equations in RZ and XYZ geometries. These new methods are Piecewise Linear Discontinuous Finite Element Methods, whose unique feature is that they utilize Piecewise Linear basis functions. Stone and Adams first describe the PWL functions and applied them to DFEM discretization of the XY transport equation [3,4]. Bailey, Adams, Yang, and Zika have since applied these basis functions to a continuous finite element discretization of the diffusion equation [5]. We require that our new methods perform well in the diffusion limit for grids composed of arbitrary polygons (RZ) and polyhedra (XYZ). These cell types are sometimes required by the discretization of the hydrodynamic equations.

A few methods exist to solve the radiation transport equation on arbitrary polyhedral and polygonal grids. Adams has proven the properties which cause discontinuous finite element methods to perform well in the asymptotic diffusion limit [6]. As a result, many of the methods that exist to solve these radiation transport problems are discontinuous finite element methods. The first such methods were Linear Discontinuous Finite Element Methods on triangular and tetrahedral meshes, Bi-Linear Discontinuous Finite

Element Methods on quadrilaterals (2D), and Tri-Linear Discontinuous Finite Element methods on hexahedra (3D) [7]. These methods are computationally efficient and well understood, but can only be used on their specific grid types when the problem requires that the method satisfy the diffusion limit [6]. As a result, they are not sufficiently general for our arbitrary polygonal and polyhedral meshes. A second method is the Wachspress Discontinuous Finite Element Method, which uses Wachspress rational basis functions. The Wachspress method does satisfy the diffusion limit on most polygonal and polyhedral meshes. It can also handle cells with curved edges and faces. However, integrals of the basis functions must be performed numerically, creating significant computational expense [8,9]. In addition, Wachspress functions cannot be applied to cells with interior angles equal to or greater than 180 degrees, which in our applications is a severe limitation. The Upstream Corner Balance Method is a third option for these radiation transport problems. This method is computational efficient because, unlike discontinuous finite element methods, it can solve the unknowns in a cell one at a time. Discontinuous finite element methods usually invert an NxN matrix in each cell, where N is the number of vertices in the cell. In three dimensions, this NxN matrix solve becomes a non-trivial computational expense. Furthermore, the Upstream Corner Balance method performs well in the diffusion limit on arbitrary polygonal and polyhedral cells with moderate distortion [10,11,12]. However, when these cells become sufficiently distorted, which occurs often in the Lagrangian solution of the hydrodynamic equations, the method loses accuracy in general and its solution can sometimes oscillate. A final method that has the potential to solve these radiation transport problems is a Characteristic method. For nuclear reactor problems, this method has proven to be both computationally efficient and accurate for arbitrary polygonal and polyhedral cells. However, in has not been proven in general how to ensure that the method satisfies the diffusion limit on these cell types, and in fact it appears that a successful characteristic method will need basis functions with the same properties as those of a successful discontinuous finite element method [13]. Thus, the

present work may enable characteristic methods to join the list of viable options for the radiative transfer problems we consider.

This list of potential methods to solve the radiation transport equation of arbitrary polygonal and polyhedral grids is very short. Furthermore, each of the existing methods has distinct advantages and troublesome disadvantages. The goal of our new method is to match the advantages of the existing methods, while mitigating their disadvantages.

Overview of chapters

This introductory chapter briefly describes the linear Boltzmann transport equation and physical problems that we can model with this equation. We note that the transport equation must be solved numerically for most physical systems of interest. Finally, we describe the role that radiation transport has in various HED systems and define goals for the new spatial discretizations we are developing to apply to these problems.

Chapter II discusses the radiative transfer problem in more depth. We begin this chapter by outlining how the radiation transport equation is coupled to the Euler equations, which we use here as a model for hydrodynamics, an equation of state, and the matter energy-balance equation. Then we show that the diffusion equation is an asymptotic limit of the radiative transfer equations, which are the radiation transport equation coupled to the matter energy-balance equation. As a result, we note that in order for spatial discretizations of this system to be accurate, they must mimic to some extent the behavior of the exact solution in the diffusion limit. In particular, we note that if a candidate method cannot perform well in the diffusion limit of the mono-energetic, uncoupled transport equation, it has no chance of performing well in more realistic diffusive problems.

In Chapter III, we briefly discuss basic discretization techniques for four of the seven independent variables in the transport equation—time, energy, and the two direction

variables. The result of applying these discretizations to the transport equation is a set of mono-energetic, single direction equations to discretize only in the spatial variable to generate a full numerical solution. We also briefly discuss the iterative methods that we use to solve the resultant linear system of equations that we generate when we discretize the equation. Finally, we define the types of spatial grids on which we will spatially discretize the transport equation and briefly describe a variety of spatial discretizations that can potentially generate accurate solutions on these grids.

Chapter IV begins with a brief description of the RZ transport equation and the special requirements of its angular discretization. We apply a general discontinuous finite element method to the RZ equation and introduce the PWL basis functions in two dimensions. Next, we discuss how to build the single cell PWLD matrix using standard finite element mapping techniques, resulting in a full definition of the method in RZ geometry. We then perform an asymptotic analysis of the PWLD method in RZ geometry. A result of this analysis is that "lumping" the method can improve the PWLD solution in the diffusion limit. We define two types of lumping for finite element methods in RZ geometry. This next section of this chapter is a description of Capsaicin, an RZ transport code being developed at Los Alamos National Laboratory. We describe the process we used to implement PWLD in Capsaicin, focusing on how we were able to leverage existing pieces of the code to efficiently build and test the PWLD method. We conclude this chapter with the results of test problems that demonstrate PWLD's accuracy, robustness, and generality. We compare PWLD to Bi-Linear Discontinuous Finite Element Method (BLD) on quadrilateral cells.

Chapter V begins with a general description of discontinuous finite element methods applied to the XYZ transport equation. We examine some of the unique problems we encounter with three-dimensional elements and potential solutions to these problems. Then we describe the three dimensional PWL basis functions and show how to build the single-cell coefficient matrix using standard finite element mapping techniques. We

conclude this chapter with the asymptotic analysis of the XYZ PWLD discretization. Again, we note that lumping can improve the method's solution in the diffusion limit.

In Chapter VI we describe the implementation and testing of the PWLD method in the Parallel Deterministic Transport code (PDT) being developed at Texas A&M University. We discuss linear and quadratic manufactured solutions that we derive to test the PWLD method and also the Tri-Linear Discontinuous Finite Element Method (TRILD). The TRILD method is a standard against which we can compare PWLD. We use the quadratic test problem to determine the truncation error of both methods in the thin limit and the thick diffusion limit on orthogonal and random grids.

In Chapter VII, we conclude with a discussion of the PWLD methods we have developed. We summarize the results from our asymptotic analysis and test problem results, draw some general conclusions about these results, and suggest ideas for future research.

CHAPTER II
AN INTRODUCTION TO RADIATION HYDRODYNAMICS

The purpose of this chapter is to show how the linear Boltzmann transport equation fits into a multiphysics radiation hydrodynamics model. We begin this chapter by outlining the Euler equations, which describe fluid flow in inertial confinement fusion and other high energy density (HED) systems. We then introduce the radiative transfer equations, which couple the radiation transport equation and a material energy balance equation. In order to model the whole system, we discuss how to couple the Euler equations to the transport equations, focusing on understanding the bulk energy and momentum transfer terms in each equation and ignoring the mathematical details. We conclude this chapter with a simple asymptotic diffusion limit analysis to show that in order for a radiation transport method to accurately describe this physical system, it must satisfy this diffusion limit.

The Euler equations

The Euler Equations are a set of three equations that can be used to describe fluid motion that admits shock waves. In this chapter we use these equations to describe how to couple radiative transfer equations to fluid-flow equations. Each equation is a conservation equation of a specific physical quantity. When we write these equations, all dependent variables are dependent on position, $\vec{r}$, and time, *t*, but we omit specifying this dependence in the equations. Our description of this physics is very simple and based on many, more detailed sources [1,14,15,16].

The first equation, often called the continuity equation, is a conservation of mass statement.

$$\frac{\partial}{\partial t}\rho + \vec{\nabla}\bullet\left[\rho\vec{u}\right] = 0\,, \tag{2.1}$$

where ρ is the density of the fluid and $\vec{u}$ is the velocity of the fluid. In Eq. (1.1), we are conserving "specific" mass, or density. This equation states that the time rate of change of mass in any portion of the system is equal to the mass flow rate across the boundary of that portion.

The second Euler Equation is a conservation of momentum equation.

$$\frac{\partial}{\partial t}\left[\rho\vec{u}\right]+\vec{\nabla}\bullet\left[\rho\vec{u}\otimes\vec{u}\right]+\vec{\nabla}p=0\,, \tag{2.2}$$

where ρ and $\vec{u}$ have been previously defined and p is the pressure in the system. Eq. (2.2) is really a system of three equations for the three spatial coordinates that make up the vector components in the equation. It takes into account any forces acting on the fluid that will affect its motion. As Eq. (2.2) is written, no external forces are acting on the fluid. To include external forces or other outside sources of momentum, for example momentum due to photon interaction with the fluid, we add a source term to the right hand side of the equation. No viscous forces are included in the momentum equation. If we included viscous terms, they would act as a dampening effect on shocks.

We write a conservation of total energy statement as the third Euler equation.

$$\frac{\partial}{\partial t}\left[\frac{1}{2}\rho u^2+\rho e\right]+\vec{\nabla}\bullet\left[\left(\frac{1}{2}\rho u^2+\rho e+p\right)\vec{u}\right]=0\,, \tag{2.3}$$

where ρ, $\vec{u}$, and p have been previously defined and e is the specific internal energy of the fluid. Eq. (2.3) states that energy is conserved between kinetic energy, internal energy, and energy changes due to pressure changes of the fluid. Like the momentum conservation equation in Eq. (2.2), the energy conservation equation in Eq. (2.3) does not include external sources of energy. To include these sources, we would add a source term to the right hand side of the equation. Our energy equation also neglects heat conduction through the fluid [15].

Eqs. (1.1)-(2.3) form a system of three equations (two scalar equations and one vector equation) and four unknowns, ρ, $\vec{u}$, p, and e. In order to close this system, we use an equation of state to relate fluid pressure to the internal energy and density.

$$p = p(\rho, e) \tag{2.4}$$

Often, this relationship is written in terms of temperature as well.

$$T = T(\rho, e) \tag{2.5}$$

These four equations, the Euler equations and the equation of state, fully describe fluid motion without viscosity or external sources of momentum and energy. It is possible to derive these relationships from an asymptotic expansion of a non-linear Boltzmann equation for the particles that make up the fluid, where particle-particle interactions are not neglected. The leading-order result of this expansion is the Euler equations. The first order-result of the expansion is the Navier-Stokes equations [14].

The radiative transfer equations

We begin our discussion of the radiation transport equation with the linear Boltzmann transport equation we introduced in Chapter I.

$$\frac{1}{v(E)}\frac{\partial \psi(\vec{r},\vec{\Omega},E,t)}{\partial t} + \vec{\Omega}\bullet\vec{\nabla}\psi(\vec{r},\vec{\Omega},E,t) + \sigma(\vec{r},E,t)\psi(\vec{r},\vec{\Omega},E,t) = S(\vec{r},\vec{\Omega},E,t), \tag{2.6}$$

where

$\vec{r} \equiv$ particle location

$\vec{\Omega} \equiv$ particle direction of travel

$E \equiv$ particle energy

$t \equiv$ time

$v \equiv$ particle speed

$\psi \equiv$ particle path length rate density per unit direction at energy E at time t

$\sigma \equiv$ total interaction cross section

$S \equiv$ total source of particles

For photon transport, this equation becomes

$$\frac{1}{c}\frac{\partial I\left(\vec{r},\vec{\Omega},\nu,t\right)}{\partial t}+\vec{\Omega}\bullet\vec{\nabla}I\left(\vec{r},\vec{\Omega},\nu,t\right)+\sigma\left(\vec{r},\nu,t\right)I\left(\vec{r},\vec{\Omega},\nu,t\right)=S\left(\vec{r},\vec{\Omega},\nu,t\right), \qquad (2.7)$$

where

$h\nu = E$

$c \equiv$ photon speed, speed of light

$I = h\nu\psi \equiv$ photon intensity

$\sigma \equiv$ total photon opacity

The source term for radiation transport in these problems is written as

$$S\left(\vec{r},\vec{\Omega},\nu,t\right)=\frac{\sigma_s}{4\pi}\phi\left(\vec{r},\nu,t\right)+\sigma_a B\left(\nu,T\right), \qquad (2.8)$$

where ϕ is the scalar intensity defined as

$$\phi\left(\vec{r},\nu,t\right)=\int_{4\pi} d\Omega I\left(\vec{r},\vec{\Omega},\nu,t\right), \qquad (2.9)$$

σ_s is the scattering opacity, σ_a is the absorption opacity, and $B\left(\nu,T\right)$ is the Planck function for photon emission from materials. Substituting Eq. (2.8) into Eq. (2.7) results in the radiation transport equation for radiative transfer problems.

$$\frac{1}{c}\frac{\partial I\left(\vec{r},\vec{\Omega},\nu,t\right)}{\partial t}+\vec{\Omega}\bullet\vec{\nabla}I\left(\vec{r},\vec{\Omega},\nu,t\right)+\sigma\left(\vec{r},\nu,t\right)I\left(\vec{r},\vec{\Omega},\nu,t\right)=\frac{\sigma_s}{4\pi}\phi\left(\vec{r},\nu,t\right)+\sigma_a B\left(\nu,T\right) \quad (2.10)$$

The first source term in Eq. (2.8) represents isotropic scattering of photons. In inertial confinement fusion systems, this type of scattering usually occurs due to mono-chromatic scattering of photons off of electrons. This scattering interaction, often called Thomson scattering, is modeled as isotropic scattering for simplicity; however, Thomson scattering is anisotropic, and is a low-energy limit of Compton scattering. At very high energies, when photon energies become close to the rest mass of the electrons, Compton scattering begins to become a non-negligible effect in these problems, but we will neglect this type of scattering in this chapter [15]. The second source term in Eq. (2.8) is the source of photons emitted from the background material. In this system, photons are absorbed by the electrons of the background material. A fraction of these photons are then re-emitted isotropically in direction with a frequency distribution given by the Planck function:

$$B\left(\nu,T\right)=\frac{4\pi h\nu^3}{c^2}\frac{1}{e^{h\nu/kT(\vec{r})}-1}. \quad (2.11)$$

Note that the Planck function is dependent on temperature, *T*. As a result, in order for us to accurately describe radiation transport we need to account for the change in the source of photons due to changes in the temperature of the material in the system. To do this, we couple the radiation transport equation to an energy conservation equation that describes temperature changes in the system.

$$C_v\left(\vec{r},T\right)\frac{\partial T}{dt}=\int_0^\infty \sigma_a\left(\vec{r},\nu,T\right)\left[\phi\left(\vec{r},\nu,t\right)-4\pi B\left(\nu,T\right)\right]d\nu, \quad (2.12)$$

where C_v is the specific heat capacity of the background material. Eq. (2.12) says that the rate of change of the matter internal energy is the rate at which photons deposit

energy minus the rate at which matter emits energy. We will refer to the coupled system of Eqs. (2.10) and (2.12) as the radiative transfer equations.

If we look at the limit of this system as $C_v\left(\vec{r},T\right)\to 0$, we should find that all absorbed photons get re-emitted because the background material has no ability to absorb the photon energy. When we set C_v to zero in Eq. (2.12), the result is

$$\begin{gathered}\int_0^\infty \sigma_a\left(\vec{r},\nu,T\right)\phi\left(\vec{r},\nu,t\right)d\nu=\int_0^\infty \sigma_a\left(\vec{r},\nu,T\right)4\pi B\left(\nu,T\right)d\nu,\\ \tilde{\sigma}_a\left(\vec{r},T\right)\Phi\left(\vec{r},t\right)=\tilde{\sigma}_{a,B}\left(\vec{r},T\right)acT^4,\end{gathered} \tag{2.13}$$

where $\tilde{\sigma}_a$ is the energy integrated absorption opacity that is weighted by the scalar intensity, $\tilde{\sigma}_{a,B}$ is the energy integrated absorption opacity weighted by the Planck function, Φ is the energy integrated scalar intensity, and *a* is the radiation constant defined as

$$a=\frac{8\pi^5k^4}{15h^3c^3}. \tag{2.14}$$

Eq. (2.13) indicates that the total absorption rate of photons over all energies is equal to the total re-emission rate over all energies. We then integrate the radiation transport, Eq. (2.10), over all frequency,

$$\frac{1}{c}\frac{\partial I\left(\vec{r},\vec{\Omega},t\right)}{\partial t}+\vec{\Omega}\bullet\vec{\nabla}I\left(\vec{r},\vec{\Omega},t\right)+\tilde{\sigma}\left(\vec{r},t\right)I\left(\vec{r},\vec{\Omega},t\right)=\frac{\tilde{\sigma}_s}{4\pi}\Phi\left(\vec{r},t\right)+\tilde{\sigma}_{a,B}acT^4, \tag{2.15}$$

where we keep the same notation for the angular intensity, but note that this intensity is energy integrated, and the total and scattering opacities are also energy integrated and weighted by the scalar flux. We then substitute the result from Eq. (2.13) into the mono-energetic transport equation in Eq. (2.15), resulting in

$$\frac{1}{c}\frac{\partial I\left(\vec{r},\vec{\Omega},t\right)}{\partial t}+\vec{\Omega}\bullet\vec{\nabla}I\left(\vec{r},\vec{\Omega},t\right)+\tilde{\sigma}\left(\vec{r},t\right)I\left(\vec{r},\vec{\Omega},t\right)=\frac{\tilde{\sigma}_s}{4\pi}\Phi\left(\vec{r},t\right)+\frac{\tilde{\sigma}_a}{4\pi}\Phi\left(\vec{r},t\right). \quad (2.16)$$

The result of this energy-integrated, zero-heat-capacity case is that the system becomes purely scattering. Physically, all photons are absorbed and immediately re-emitted with a new direction. The problem will become diffusive when the system is optically thick, which means the total opacity becomes large. As a result, if a method cannot model this diffusive transport behavior, it will fail for these types of problems.

Coupling the radiation transport equation to the Euler equations

In order to begin coupling the radiation transport process to the fluid motion process, we need to define some physically relevant quantities that are interchangeable in both systems. The first quantity we define is the radiative energy density.

$$\Phi=\frac{1}{c}\int_0^\infty\int_{4\pi}I\left(\vec{\Omega},\nu\right)d\Omega d\nu\,. \quad (2.17)$$

We also define a radiative flux (which looks like an energy-integrated current to nuclear engineers)

$$\vec{F}=\int_0^\infty\int_{4\pi}\vec{\Omega}I\left(\vec{\Omega},\nu\right)d\Omega d\nu\,, \quad (2.18)$$

and a radiation pressure tensor

$$P_{i,j}=\frac{1}{c}\int_0^\infty\int_{4\pi}\Omega_i\Omega_j I\left(\vec{\Omega},\nu\right)d\Omega d\nu\,. \quad (2.19)$$

It is interesting to note that if we take the zeroth angular moment of the radiation transport equation, Eq. (2.10), and then integrate over all frequencies, we get an energy balance equation.

$$\frac{\partial \Phi}{\partial t} + \vec{\nabla} \bullet \vec{F} = \int_0^\infty \sigma_a \left[B(\nu, T) - \phi \right] d\nu \tag{2.20}$$

The first angular moment of this equation integrated over all frequencies represents a momentum balance equation.

$$\frac{1}{c^2} \frac{\partial \vec{F}}{\partial t} + \vec{\nabla} \bullet \underline{\underline{P}} + \frac{\sigma_a}{c} \vec{F} = 0 \tag{2.21}$$

At this point, in order to couple the radiation system to the fluid motion system, we must define a frame of reference and transform all physical quantities into it. Because the background material with which photons are interacting is moving, we cannot assume that the distribution of electron velocities is isotropic in the lab frame. (For many other transport problems, it is reasonable to assume that the background particles move isotropically in the lab frame.) As a result, to properly calculate photon-matter interactions we must transform the photon frequency and direction to the reference frame of the moving fluid. The details of this procedure are mathematically complex, especially if the fluid moves fast enough to require relativistic treatment. A good source for details on these transformations can be found in References [1,14,15,16]. We will present a summary of the results of these transformations. For the rest of this section, we have dropped the independent-variable arguments unless they are needed for clarity.

After the frame of reference transformations, the radiation transport equation can be written as

$$\frac{1}{c} \frac{\partial I}{\partial t} + \vec{\Omega} \bullet \vec{\nabla} I = Q \tag{2.22}$$

where

$$\begin{aligned} Q &= R\tilde{Q} \\ \tilde{Q} &= -\tilde{\sigma} I + \frac{\tilde{\sigma}_s}{4\pi} \tilde{\phi} + \tilde{\sigma}_a \tilde{B} \end{aligned} \tag{2.23}$$

and quantities with a "~" have been transformed to the reference frame of the moving fluid (the "fluid" frame). In Eq. (2.23), *R*, represents a transformation from the fluid frame back to the laboratory frame in which it is most convenient to represent the streaming operator. We now formally define a radiation energy-exchange source and a radiation momentum-exchange source.

$$\begin{aligned} S_{re} &= \int_0^\infty \int_{4\pi} \tilde{Q} d\Omega d\nu \\ \vec{S}_{rp} &= \frac{1}{c} \int_0^\infty \int_{4\pi} \vec{\Omega} \tilde{Q} d\Omega d\nu, \end{aligned} \tag{2.24}$$

and substitute them into the Euler equations (Eqs. (1.1)-(2.3)).

$$\begin{aligned} \frac{\partial}{\partial t} \rho + \vec{\nabla} \bullet \left[\rho \vec{u} \right] &= 0, \\ \frac{\partial}{\partial t} \left[\rho \vec{u} \right] + \vec{\nabla} \bullet \left[\rho \vec{u} \otimes \vec{u} \right] + \vec{\nabla} p &= -\vec{S}_{rp}, \\ \frac{\partial}{\partial t} \left[\frac{1}{2} \rho u^2 + \rho e \right] + \vec{\nabla} \bullet \left[\left(\frac{1}{2} \rho u^2 + \rho e + p \right) \vec{u} \right] &= -S_{re}. \end{aligned} \tag{2.25}$$

Eqs. (2.22)-(2.25) represent a fully coupled general set of radiation hydrodynamics equations. Note that the source of energy exchange in the energy-balance Euler equation is exactly opposite the source in the radiation transport equation, so that energy must be conserved in the entire system. It is also interesting to note that we can write the radiation energy balance equation, Eq. (2.20), and radiation momentum conservation equation, Eq. (2.21) for this coupled system. The balance equation becomes

$$\frac{\partial \Phi}{\partial t} + \vec{\nabla} \bullet \vec{F} = S_{re} \tag{2.26}$$

and the momentum conservation equation becomes

$$\frac{\partial \vec{F}}{\partial t} + \vec{\nabla} \underline{\underline{P}} = -\vec{S}_{rp} \,. \tag{2.27}$$

Eqs. (2.26) and (2.27) are a system of two equations with three unknowns, Φ, $\vec{F}$, and $\underline{\underline{P}}$. It is possible to solve these two equations instead of the radiation transport equation if we develop closure relationships for the three unknowns. The diffusion closure relationship is

$$\begin{aligned} \underline{\underline{P}} &= \frac{1}{3}\underline{\underline{I}}\Phi \\ \frac{\partial \vec{F}}{\partial t} &= 0, \end{aligned} \tag{2.28}$$

where we have developed a Fick's law relationship for the pressure tensor [14,15]. Solving the system of Eqs. (2.26), (2.27), and (2.28) is analogous to solving the diffusion equations for neutronics problems.

Limit of the radiative transfer equations

As we have previously implied, we can gain physical insight into these problems by examining mathematical limits of the equations. From the radiation transport perspective an extremely useful limit to examine is the thick diffusive limit that we can derive from an asymptotic analysis. It is possible to examine this diffusion limit for the fully coupled radiation hydrodynamics equations, but in our analysis, we will assume the background material is not moving. Physically, we scale the problem such that the absorption-re-emission process is the dominant photon interaction mechanism, and mono-chromatic scattering tends toward zero. As a result, our coupled system simplifies to Eqs. (2.10) and (2.12).

In the following asymptotic analysis, terms in both equations are scaled by a small number ε, as shown below, and the opacities and unknown functions are assumed to be *O(1)*. This scaling causes the absorption and re-emission of particles to be the dominant radiation-matter interaction processes. The standard "equilibrium" diffusion equation is invariant under this scaling, which offers hope that the forthcoming analysis may

produce this equation. See [17] for details. Here we are following the procedure outlined by Adams and Nowak [10].

$$\begin{aligned}\frac{\varepsilon}{c}\frac{\partial I}{dt}\left(\vec{r},\vec{\Omega},\nu,t\right)+\vec{\Omega}\bullet\vec{\nabla}I\left(\vec{r},\vec{\Omega},\nu,t\right)+\left[\frac{\sigma_a\left(\vec{r},\nu,T\left(\vec{r}\right)\right)}{\varepsilon}+\varepsilon\sigma_s\left(\vec{r},\nu\right)\right]I\left(\vec{r},\vec{\Omega},\nu,t\right)\\ =\varepsilon\frac{\sigma_s\left(\vec{r},\nu,T\left(\vec{r}\right)\right)}{4\pi}\phi\left(\vec{r},\vec{\Omega},\nu,t\right)+\frac{\sigma_a\left(\vec{r},\nu,T\left(\vec{r}\right)\right)}{\varepsilon}B\left(\nu,T\left(\vec{r}\right)\right)\end{aligned} \tag{2.29}$$

and

$$\begin{aligned}&\varepsilon C_p\left(\vec{r},T\left(\vec{r}\right)\right)\frac{\partial T\left(\vec{r}\right)}{dt}\\ &=\int_0^\infty\int_{4\pi}\frac{\sigma_a\left(\vec{r},\nu,T\left(\vec{r}\right)\right)}{\varepsilon}\left[I\left(\vec{r},\vec{\Omega},\nu,t\right)-B\left(\nu,T\left(\vec{r}\right)\right)\right]d\Omega d\nu.\end{aligned} \tag{2.30}$$

All unknown functions, I, T, σ_a, σ_s, and B, are then expanded in terms of ε.

$$\begin{aligned}I\left(\vec{r},\vec{\Omega},\nu,t\right)&=I^{(0)}+\varepsilon I^{(1)}+\varepsilon^2 I^{(2)}...\\ \phi\left(\vec{r},\vec{\Omega},\nu,t\right)&=\phi^{(0)}+\varepsilon\phi^{(1)}+\varepsilon^2\phi^{(2)}...\\ T\left(\vec{r},t\right)&=T^{(0)}+\varepsilon T^{(1)}+\varepsilon^2 T^{(2)}...\\ \sigma_a\left(\vec{r},\nu,T\right)&=\sigma_a^{(0)}+\varepsilon\sigma_a^{(1)}+\varepsilon^2\sigma_a^{(2)}...\\ \sigma_s\left(\vec{r},\nu,T\right)&=\sigma_s^{(0)}+\varepsilon\sigma_s^{(1)}+\varepsilon^2\sigma_s^{(2)}...\\ B\left(\nu,T\right)&=B^{(0)}+\varepsilon B^{(1)}+\varepsilon^2 B^{(2)}...\end{aligned} \tag{2.31}$$

and substituted into Eqs. (2.29) and (2.30).

$$\begin{aligned}
&\frac{\varepsilon}{c}\frac{\partial}{dt}\left(I^{(0)}+\varepsilon I^{(1)}+\varepsilon^2 I^{(2)}...\right)+\vec{\Omega}\bullet\vec{\nabla}\left(I^{(0)}+\varepsilon I^{(1)}+\varepsilon^2 I^{(2)}...\right)\\
&\quad+\frac{\left(\sigma_a^{(0)}+\varepsilon\sigma_a^{(1)}+\varepsilon^2\sigma_a^{(2)}...\right)}{\varepsilon}\left(I^{(0)}+\varepsilon I^{(1)}+\varepsilon^2 I^{(2)}...\right)\\
&\quad+\varepsilon\left(\sigma_s^{(0)}+\varepsilon\sigma_s^{(1)}+\varepsilon^2\sigma_s^{(2)}...\right)\left(I^{(0)}+\varepsilon I^{(1)}+\varepsilon^2 I^{(2)}...\right)\\
&\quad=\varepsilon\frac{\left(\sigma_s^{(0)}+\varepsilon\sigma_s^{(1)}+\varepsilon^2\sigma_s^{(2)}...\right)}{4\pi}\left(\phi^{(0)}+\varepsilon\phi^{(1)}+\varepsilon^2\phi^{(2)}...\right)\\
&\quad+\frac{\left(\sigma_a^{(0)}+\varepsilon\sigma_a^{(1)}+\varepsilon^2\sigma_a^{(2)}...\right)}{\varepsilon}\left(B^{(0)}+\varepsilon B^{(1)}+\varepsilon^2 B^{(2)}...\right)
\end{aligned} \tag{2.32}$$

and

$$\begin{aligned}
&\varepsilon C_p\left(\vec{r},T\right)\frac{\partial}{dt}\left(T^{(0)}+\varepsilon T^{(1)}+\varepsilon^2 T^{(2)}...\right)\\
&=\int_0^\infty\int_{4\pi}\frac{\left(\sigma_a^{(0)}+\varepsilon\sigma_a^{(1)}+\varepsilon^2\sigma_a^{(2)}...\right)}{\varepsilon}\begin{bmatrix}\left(I^{(0)}+\varepsilon I^{(1)}+\varepsilon^2 I^{(2)}...\right)\\-\left(B^{(0)}+\varepsilon B^{(1)}+\varepsilon^2 B^{(2)}...\right)\end{bmatrix}d\Omega d\nu.
\end{aligned} \tag{2.33}$$

The O(*1/ε*) terms are collected. From Eq. (2.32)

$$\frac{\sigma_a^{(0)}}{\varepsilon}\left(I^{(0)}\right)=\frac{\sigma_a^{(0)}}{\varepsilon}B^{(0)}\left(\nu,T^{(0)}\right), \tag{2.34}$$

resulting in

$$I^{(0)}=B^{(0)}\left(\nu,T^{(0)}\right). \tag{2.35}$$

We then integrate Eq. (2.35) over all angles, resulting in

$$\phi^{(0)}=4\pi B^{(0)}\left(\nu,T^{(0)}\right). \tag{2.36}$$

Physically this result means that the leading-order intensity is isotropic in angle and Planckian in energy. Next, we collect the *O(1)* terms in Eq. (2.32).

$$\vec{\Omega}\bullet\vec{\nabla}I^{(0)}+\left[\sigma_a I\right]^{(1)}=\left[\sigma_a B\right]^{(1)} \tag{2.37}$$

where

$$\begin{aligned}\left[\sigma_a I\right]^{(1)} &= \sigma_a^{(0)} I^{(1)} + \sigma_a^{(1)} I^{(0)} \\ \left[\sigma_a B\right]^{(1)} &= \sigma_a^{(0)} B^{(1)} + \sigma_a^{(1)} B^{(0)}.\end{aligned} \tag{2.38}$$

As a result,

$$\left[\sigma_a I\right]^{(1)} = -\vec{\Omega}\bullet\vec{\nabla} I^{(0)} + \left[\sigma_a B\right]^{(1)}. \tag{2.39}$$

We now define a quantity

$$\vec{J} = \int_{4\pi} d\Omega \vec{\Omega} I\left(\vec{\Omega}\right), \tag{2.40}$$

which is known as *current*. We find the *O(ε)* current by integrating Eq. (2.39) over all directions, Ω.

$$\left[\sigma_a \vec{J}\right]^{(1)} = -\int_{4\pi} d\Omega \vec{\Omega}\vec{\Omega}\bullet\vec{\nabla} I^{(0)} + \int_{4\pi} d\Omega \vec{\Omega}\left[\sigma_a B\right]^{(1)} \tag{2.41}$$

The result of these integrals is

$$\left[\sigma_a \vec{J}\right]^{(1)} = -\frac{1}{3}\vec{\nabla} I^{(0)}. \tag{2.42}$$

The integral involving the Planckian cancels because B is isotropic and $\vec{\Omega}$ is an odd function. The next step is to expand the $\left[\sigma_a \vec{J}\right]^{(1)}$ term.

$$\left[\sigma_a \vec{J}\right]^{(1)} = \sigma_a^{(0)} \vec{J}^{(1)} + \sigma_a^{(1)} \vec{J}^{(0)} = -\frac{1}{3}\vec{\nabla} I^{(0)} \tag{2.43}$$

We note that

$$\vec{J}^{(0)} = \int_{4\pi} d\Omega \vec{\Omega} I^{(0)} = \int_{4\pi} d\Omega \vec{\Omega} B^{(0)} = 0\,, \tag{2.44}$$

resulting in

$$J^{(1)} = -\frac{1}{3\sigma_a^{(0)}} \vec{\nabla} I^{(0)} = -\frac{1}{3\sigma_a^{(0)}} \vec{\nabla} B^{(0)}\,. \tag{2.45}$$

Next, we collect the *O(ε)* terms in Eqs. (2.32) and (2.33). From Eq. (2.30)

$$\frac{1}{c}\frac{\partial}{dt}\left(I^{(0)}\right) + \vec{\Omega}\bullet\vec{\nabla}\left(I^{(1)}\right) + \left[\sigma_a I\right]^{(2)} + \sigma_s I^{(0)} = \frac{\sigma_s}{4\pi}\phi^{(0)} + \left[\sigma_a B\right]^{(2)}\,. \tag{2.46}$$

We recall that the leading order angular intensity is isotropic, causing the scattering terms in Eq. (2.46) to cancel.

$$\frac{1}{c}\frac{\partial}{dt}\left(I^{(0)}\right) + \vec{\Omega}\bullet\vec{\nabla}\left(I^{(1)}\right) + \left[\sigma_a I\right]^{(2)} = \left[\sigma_a B\right]^{(2)}\,. \tag{2.47}$$

When rearranged, Eq. (2.47) becomes

$$\left[\sigma_a I\right]^{(2)} = -\frac{1}{c}\frac{\partial}{dt}\left(I^{(0)}\right) - \vec{\Omega}\bullet\vec{\nabla}\left(I^{(1)}\right) + \left[\sigma_a B\right]^{(2)}\,. \tag{2.48}$$

From Eq. (2.33)

$$C_p\left(\vec{r},T\right)\frac{\partial}{dt}\left(T^{(0)}\right) = \int_0^\infty \int_{4\pi} \left[\sigma_a I\right]^{(2)} - \left[\sigma_a B\right]^{(2)} d\Omega d\nu\,. \tag{2.49}$$

Substituting Eq. (2.48) into Eq. (2.49) results in

$$C_p\left(\vec{r},T\right)\frac{\partial}{dt}\left(T^{(0)}\right) = -\int_0^\infty \int_{4\pi} \frac{1}{c}\frac{\partial}{dt}\left(I^{(0)}\right) + \vec{\Omega}\bullet\vec{\nabla}\left(I^{(1)}\right) d\Omega d\nu\,, \tag{2.50}$$

which can be simplified to

$$C_p\left(\vec{r},T\right)\frac{\partial}{dt}\left(T^{(0)}\right) + \frac{1}{c}\frac{\partial}{dt}\int_0^\infty \int_{4\pi} \left(I^{(0)}\right) d\Omega d\nu + \vec{\nabla}\bullet\int_0^\infty \int_{4\pi} \vec{\Omega} I^{(1)} d\Omega d\nu = 0\,. \tag{2.51}$$

Note that $\vec{J}^{(1)} = \int_{4\pi} \vec{\Omega} I^{(1)} d\Omega$, causing Eq. (2.51) to become

$$C_p(\vec{r},T)\frac{\partial}{dt}\left(T^{(0)}\right)+\frac{1}{c}\frac{\partial}{dt}\int_0^\infty\int_{4\pi}\left(I^{(0)}\right)d\Omega d\nu+\vec{\nabla}\bullet\int_0^\infty \vec{J}^{(1)}d\nu=0. \tag{2.52}$$

The substitution of Eq. (2.45) into Eq. (2.52) for the current results in

$$\begin{aligned} &C_p(\vec{r},T)\frac{\partial}{dt}\left(T^{(0)}\right)+\frac{1}{c}\frac{\partial}{dt}\int_0^\infty\int_{4\pi}\left(I^{(0)}\right)d\Omega d\nu \\ &-\vec{\nabla}\bullet\int_0^\infty\frac{1}{3\sigma(\nu)^{(0)}}\vec{\nabla}B(\nu,T)^{(0)}d\nu=0. \end{aligned} \tag{2.53}$$

The integral in the third term of Eq. (2.53) must be treated specially. This integral is

$$\begin{aligned} &\int_0^\infty\frac{1}{\sigma(\nu)^{(0)}}\vec{\nabla}B(\nu,T)^{(0)}d\nu \\ &=\int_0^\infty\frac{1}{\sigma(\nu)^{(0)}}\left(\frac{\partial}{\partial x}B(\nu,T)^{(0)}\hat{i}+\frac{\partial}{\partial y}B(\nu,T)^{(0)}\hat{j}+\frac{\partial}{\partial z}B(\nu,T)^{(0)}\hat{k}\right)d\nu. \end{aligned} \tag{2.54}$$

Each derivative can be simplified using the chain rule. For example

$$\frac{\partial}{\partial x}B(\nu,T)^{(0)}\hat{i}=\left(\frac{\partial B}{\partial T^4}\right)\left(\frac{\partial T^4}{\partial x}\right)\hat{i}=\left(\frac{\partial B}{\partial T}\frac{\partial T}{\partial T^4}\right)\left(\frac{\partial T^4}{\partial x}\right)\hat{i}=\frac{1}{4T^3}\left(\frac{\partial B}{\partial T}\right)\left(\frac{\partial T^4}{\partial x}\right)\hat{i}. \tag{2.55}$$

If this chain rule is performed on each derivative in the integral the result is

$$\begin{aligned} &\int_0^\infty\frac{1}{\sigma(\nu)^{(0)}}\vec{\nabla}B(\nu,T)^{(0)}d\nu \\ &=\int_0^\infty\frac{1}{\sigma(\nu)^{(0)}}\frac{1}{4T^3}\left(\frac{\partial B}{\partial T}\right)\left(\left(\frac{\partial T^4}{\partial x}\right)\hat{i}+\left(\frac{\partial T^4}{\partial y}\right)\hat{j}+\left(\frac{\partial T^4}{\partial z}\right)\hat{k}\right)d\nu, \end{aligned} \tag{2.56}$$

which simplifies to

$$\int_0^\infty \frac{1}{\sigma(\nu)^{(0)}} \vec{\nabla} B(\nu,T)^{(0)} d\nu = \frac{1}{4T^3}\left[\vec{\nabla}\left(T^{(0)}\right)^4\right]\int_0^\infty \frac{1}{\sigma(\nu)^{(0)}}\left(\frac{\partial B}{\partial T}\right)d\nu \,. \tag{2.57}$$

We can multiply and divide Eq. (2.57) by $\int_0^\infty \left(\frac{\partial B}{\partial T}\right)d\nu$, resulting in

$$\int_0^\infty \frac{1}{\sigma(\nu)^{(0)}} \vec{\nabla} B(\nu,T)^{(0)} d\nu = \frac{1}{4T^3}\left[\vec{\nabla}\left(T^{(0)}\right)^4\right]\left[\frac{\int_0^\infty \frac{1}{\sigma(\nu)^{(0)}}\left(\frac{\partial B}{\partial T}\right)d\nu}{\int_0^\infty \left(\frac{\partial B}{\partial T}\right)d\nu}\right]\int_0^\infty \left(\frac{\partial B}{\partial T}\right)d\nu \,. \tag{2.58}$$

The term in the brackets in Eq. (2.58) is an "averaged" inverse opacity value known as the inverse of the Rosseland mean opacity [9].

$$\frac{1}{\sigma_R} \equiv \left[\frac{\int_0^\infty \frac{1}{\sigma(\nu)^{(0)}}\left(\frac{\partial B}{\partial T}\right)d\nu}{\int_0^\infty \left(\frac{\partial B}{\partial T}\right)d\nu}\right] \tag{2.59}$$

Using this definition, Eq. (2.54) becomes

$$\int_0^\infty \frac{1}{\sigma(\nu)^{(0)}} \vec{\nabla} B(\nu,T)^{(0)} d\nu = \frac{1}{4\sigma_R T^3}\left[\vec{\nabla}\left(T^{(0)}\right)^4\right]\frac{\partial}{\partial T}\int_0^\infty B d\nu \,. \tag{2.60}$$

The integral of the Planckian over all frequencies is

$$\int_0^\infty B d\nu = \int_0^\infty \frac{4\pi h\nu^3}{c^2}\frac{1}{e^{h\nu/kT}-1}d\nu = acT^4 \,, \tag{2.61}$$

where the radiation constant, a, is defined as

$$a = \frac{8\pi^5 k^4}{15h^3c^3} \tag{2.62}$$

When the two values from Eqs. (2.62)and (2.61) are inserted into Eq. (2.60), the result is

$$\int_0^\infty \frac{1}{\sigma(\nu)^{(0)}} \vec{\nabla} B(\nu,T)^{(0)} d\nu = \frac{1}{4\sigma_R T^3}\left[\vec{\nabla}\left(T^{(0)}\right)^4\right]\frac{\partial}{\partial T}\left(acT^4\right), \tag{2.63}$$

which can be simplified to

$$\int_0^\infty \frac{1}{\sigma(\nu)^{(0)}} \vec{\nabla} B(\nu,T)^{(0)} d\nu = \frac{ac}{\sigma_R}\left[\vec{\nabla}\left(T^{(0)}\right)^4\right]. \tag{2.64}$$

From this result, Eq. (2.53) becomes

$$C_p(\vec{r},T)\frac{\partial}{dt}\left(T^{(0)}\right) + \frac{1}{c}\frac{\partial}{dt}\int_0^\infty \int_{4\pi}\left(I^{(0)}\right) d\Omega d\nu - \vec{\nabla}\bullet\frac{ac}{3\sigma_R}\left[\vec{\nabla}\left(T^{(0)}\right)^4\right] = 0. \tag{2.65}$$

We now define a radiation energy density as

$$E_R(\vec{r},t) = \frac{1}{c}\int_0^\infty d\nu \int_{4\pi} d\Omega I\left(\vec{r},\vec{\Omega},\nu,t\right), \tag{2.66}$$

and a radiation temperature in terms of the radiation energy as

$$E_R = aT_R^4, \tag{2.67}$$

where *a* is the radiation constant given in Eq. (2.14). For this limit, the leading order radiation energy density is

$$E_R^{(0)}(\vec{r},t) = \frac{1}{c}\int_0^\infty d\nu \int_{4\pi} d\Omega I^{(0)}\left(\vec{r},\vec{\Omega},\nu,t\right) = \frac{1}{c}\int_0^\infty d\nu \int_{4\pi} d\Omega B^{(0)}\left(\nu,T^{(0)}\right). \tag{2.68}$$

As a result, the leading order radiation temperature is given by

$$E_R^{(0)}(\vec{r},t) = a\left(T^{(0)}\right)^4. \tag{2.69}$$

Using the definition of the radiation energy density from Eq. (2.69) Eq. (2.65) becomes

$$C_p\left(\vec{r},T\right)\frac{\partial}{dt}\left(T^{(0)}\right)+a\frac{\partial}{dt}\left(T^{(0)}\right)^4-\vec{\nabla}\bullet\frac{ac}{3\sigma_R}\left[\vec{\nabla}\left(T^{(0)}\right)^4\right]=0 \tag{2.70}$$

The first term of Eq. (2.54) is often neglected, because it is often slowly varying compared to the other terms. The result is a diffusion equation for T^4.

$$a\frac{\partial}{dt}\left(T^{(0)}\right)^4-\vec{\nabla}\bullet\frac{ac}{3\sigma_R}\nabla\left(T^{(0)}\right)^4=0 \tag{2.71}$$

Because the result of this analysis is a diffusion equation, we can infer that in order for a discretization of the radiative transfer equations to be accurate, it must result in an accurate diffusion discretization when the same analysis is applied to the discrete equations. Furthermore, if the analysis is applied to the system where C_p approaches zero, the result will also be that the leading order energy density will satisfy a diffusion equation. When a method's leading-order discrete solution satisfies an accurate discretization of the same diffusion equation that is satisfied by the leading-order analytic solution, and also satisfies accurate boundary conditions, we say that the method "satisfies" the diffusion limit. If a method cannot satisfy the diffusion limit when applied to the simplified radiative transfer system where C_p approaches zero, it will not satisfy the diffusion limit for a more physically complex problem. As a result, it is imperative that we test a method's ability to satisfy this diffusion limit to determine if the method will be accurate in this simplified problem.

One might argue that this analysis indicates that solutions from a direct discretization of the equilibrium diffusion equation would be sufficient for our problems of interest. These diffusion solutions would be significantly cheaper in computation cost and many robust methods exist to solve the diffusion equation for a variety of problems. However, the diffusion limit analysis assumes that the whole system is physically diffusive and this assumption is not practical for real problems. It is not hard to imagine a system with regions of extremely high density and extremely low density materials. In such a system, the high density regions would be diffusive or optically thick, but the low

density regions would be optically thin. If we model the entire system with diffusion we have created a significant error in our model before we even apply a numerical approximation. Furthermore, if we can model the system with diffusion where it is diffusive and transport where it is not diffusive, we still have the problem of correctly determining the behavior of the system at the interface between the two regions. Finally, it is far too simplistic to think of a region as being completely diffusive or not – there is a continuum. Diffusion cannot be trusted in regions that are not highly diffusive, but a transport discretization cannot be trusted in regions that are "almost" diffusive unless that transport discretization is known to be accurate in diffusive regions. The bottom line from this discussion is that a transport discretization that is accurate in both thin limits and in the thick diffusive limit is the safest path to a solution in which we have confidence.

The grand challenge of developing new methods to solve the transport equation for radiative transfer problems is to design methods that satisfy the leading order diffusion limit, while being sufficiently computationally efficient so that the accuracy gain from these transport calculations outweigh the extra computational cost over the more approximate diffusion solutions. If we meet the challenge of developing transport methods that perform well in the diffusion limit and are sufficiently inexpensive, we not only can add accuracy to existing physical models, we can also begin to solve previously unsolvable problems.

Summary

In this chapter we briefly describe the set of radiation hydrodynamics equations used to model high energy density systems. The Euler equations are used to model fluid flow of all materials in the system. The radiative transfer equations are used to describe photon transport and its interaction with the background material through the absorption, scattering, and emission processes. We then briefly describe how to couple the two sets of equations together to arrive at a mathematical radiation hydrodynamics model. We

conclude the chapter with a simple diffusion limit analysis to show the physical properties under which the transport problem becomes a diffusion problem. We note that our transport methods must satisfy this limit to be an accurate model for our physical problems.

CHAPTER III
GENERAL DISCRETIZATIONS OF THE TRANSPORT EQUATION

In this chapter we present common discretization techniques for the time, energy, and angular independent variables in the transport equation. We also discuss the variety of methods used to solve the resultant linear system of equations. We conclude this chapter with a discussion of potential spatial discretizations that we can apply to our time, energy, and angular discretized transport equation, focusing on methods that should perform well in optically thick diffusive regions even when the spatial mesh is composed of arbitrary polygonal and polyhedral cells.

Operator splitting and time differencing

To solve our time-dependent coupled radiation-hydrodynamics equations we can use a strategy known as operator splitting. Our discussion of this strategy relies heavily on Castor [15]. To begin, we write our system of equations in terms of a simple first order differential equation.

$$\frac{d\vec{X}}{dt} = A\left[\vec{X}\right] + B\left[\vec{X}\right] + \ldots \tag{3.1}$$

where $\vec{X}$ represents all unknown quantities in the problem and *A*, *B*,… represent various operators for the different types of physics in the coupled problem. We could explicitly time difference this problem resulting in

$$\frac{\vec{X}^{t+1} - \vec{X}^{t}}{\Delta t} = A\left[\vec{X}^{t}\right] + B\left[\vec{X}^{t}\right] + \ldots . \tag{3.2}$$

However, explicit time differencing of this nature is subject to limitations on the time step size, Δt, making this method conditionally stable. In our case, the time step limitation for the hydrodynamics physics model is the Courant limit

$$\frac{c_s \Delta t}{\Delta x} < 1, \tag{3.3}$$

where c_s is the speed of sound in the material and Δx is the mesh size of the spatial discretization. The time step limitation for radiation transport for explicit time differencing is the radiation Courant limit

$$\frac{c \Delta t}{\Delta x} < 1, \tag{3.4}$$

where c is the speed of light [15]. In this limit, a particle is not allowed to cross a spatial cell in one time step. Solving the hydrodynamics system is tractable given its time step constraints; however, because the speed of light is so large, solving the radiation transport equation explicitly requires exceedingly small time steps making the entire time dependent calculation prohibitively expensive. As a result, we are forced to apply an implicit time differencing scheme to the transport equation, which produces an unconditionally stable method. An implicit time differencing scheme is written as

$$\frac{\vec{X}^{t+1} - \vec{X}^{t}}{\Delta t} = A\left[\vec{X}^{t+1}\right] + B\left[\vec{X}^{t+1}\right] + \ldots \tag{3.5}$$

We can employ an explicit time differencing of the hydrodynamics system and an implicit time differencing of the radiation transport system if we employ operator splitting. Operator splitting splits the time derivative of $\vec{X}$ into k pieces, so that there are k partial time steps inside one actual time step.

$$\begin{aligned}
\frac{\vec{X}^{\,t+\frac{1}{k}} - \vec{X}^{\,t}}{\Delta t} &= A\left[\vec{X}^{\,t+\frac{1}{k}}\right], \\
\frac{\vec{X}^{\,t+\frac{2}{k}} - \vec{X}^{\,t+\frac{1}{k}}}{\Delta t} &= B\left[\vec{X}^{\,t+\frac{2}{k}}\right], \\
&\vdots \\
\frac{\vec{X}^{\,t+1} - \vec{X}^{\,t+\frac{k-1}{k}}}{\Delta t} &= Z\left[\vec{X}^{\,t+1}\right]
\end{aligned} \tag{3.6}$$

The operators are split and solved individually over one partial time step. We have shown each operator to be differenced implicitly in Eq. (3.6); however, we can choose to solve either explicitly or implicitly over the partial time step. In the radiation-hydrodynamics case, we would split the time step into two sequential operations,

$$\begin{aligned}
\frac{\vec{X}^{\,t+\frac{1}{2}} - \vec{X}^{\,t}}{\Delta t} &= H\left[\vec{X}^{\,t}\right], \\
\frac{\vec{X}^{\,t+1} - \vec{X}^{\,t+\frac{1}{2}}}{\Delta t} &= R\left[\vec{X}^{\,t+1}\right]
\end{aligned} \tag{3.7}$$

and solve the hydrodynamics operation, *H*, explicitly over the first partial time step then use its results to solve the radiation operation, *R*, implicitly over the remaining partial time step. In this case, *H* operates only on dependent variables in the hydrodynamics equations and *R* operates only on dependent variables in the radiative transfer equations. As a result, we have successfully split our operators and can now focus on solving the radiation transport piece of the problem by itself. It is interesting to note that if we used a diffusion approximation to the radiation transport equation, we would decrease the severity imposed on the time step limit for an explicit time differencing, but it is still not enough to warrant an explicit solution technique [15].

After we have applied operator splitting to the problem, we have isolated the radiative transfer equations we presented in Chapter II as the radiation transport piece of the problem. These equations are

$$\frac{1}{c}\frac{\partial I\left(\vec{r},\vec{\Omega},\nu,t\right)}{\partial t}+\vec{\Omega}\bullet\vec{\nabla}I\left(\vec{r},\vec{\Omega},\nu,t\right)+\sigma\left(\vec{r},\nu,t\right)I\left(\vec{r},\vec{\Omega},\nu,t\right) = \frac{1}{4\pi}\sigma_s\phi\left(\vec{r},\nu,t\right)+\sigma_a B\left(\nu,T\right) \tag{3.8}$$

and

$$C_v\left(\vec{r},T\right)\frac{\partial T}{dt}=\int_0^\infty \sigma_a\left(\vec{r},\nu,T\right)\left[\phi\left(\vec{r},\nu,t\right)-4\pi B\left(\nu,T\right)\right]d\nu\,, \tag{3.9}$$

where

$$B\left(\nu,T\right)=\frac{4\pi h\nu^3}{c^2}\frac{1}{e^{h\nu/kT(\vec{r})}-1}\,. \tag{3.10}$$

The difficulty in applying an implicit time differencing to this system lies in the coupling between Eqs. (3.8) and (3.9) through the Planck function, *B*, which is dependent on temperature, which can change in time. This coupling makes the system non-linear. As a result, we first linearize the system and then solve it implicitly [14,15]. We begin by implicitly differencing the angular intensity and the temperature:

$$\frac{1}{c}\frac{I^{t+1}-I^t}{\Delta t}+\vec{\Omega}\bullet\vec{\nabla}I^{t+1}+\sigma I^{t+1}=\frac{\sigma_s}{4\pi}\phi^{t+1}+\sigma_a B^*\left(\nu,T\right) \tag{3.11}$$

and

$$C_v\frac{T^{t+1}-T^t}{\Delta t}=\int_0^\infty \sigma_a\left[\phi^{t+1}-4\pi B^*\left(\nu,T\right)\right]d\nu\,. \tag{3.12}$$

We then linearize the system by linearizing the Planck function.

$$B^{*} = B^{old} + \frac{\partial B^{old}}{\partial T}\left(T^{t+1} - T^{old}\right), \tag{3.13}$$

and substitute this into Eqs. (3.11) and (3.12), resulting in

$$\frac{1}{c}\frac{I^{t+1} - I^{t}}{\Delta t} + \vec{\Omega}\bullet\vec{\nabla}I^{t+1} + \sigma I^{t+1} = \frac{\sigma_s}{4\pi}\phi^{t+1} + \sigma_a\left[B^{old} + \frac{\partial B^{old}}{\partial T}\left(T^{t+1} - T^{old}\right)\right] \tag{3.14}$$

and

$$C_v\frac{T^{t+1} - T^{t}}{\Delta t} = \int_0^{\infty}\sigma_a\left[\phi^{t+1} - 4\pi\left[B^{old} + \frac{\partial B^{old}}{\partial T}\left(T^{t+1} - T^{old}\right)\right]\right]d\nu\,, \tag{3.15}$$

which is a linear set of equations. In this set of equations, we assume that the opacities are evaluated at T^{old}. Next, we solve Eq. (3.15) for $\left(T^{t+1} - T^{old}\right)$

$$T^{t+1} - T^{old} = \frac{1}{\dfrac{C_v}{\Delta t} + \displaystyle\int_0^{\infty}\sigma_a\frac{\partial B^{old}}{\partial T}d\nu}\left[\int_0^{\infty}\sigma_a\phi^{t+1}d\nu - 4\pi\int_0^{\infty}\sigma_a B^{old}d\nu\right] \tag{3.16}$$

and substitute it into Eq. (3.14)

$$\begin{aligned}&\frac{1}{c}\frac{I^{t+1} - I^{t}}{\Delta t} + \vec{\Omega}\bullet\vec{\nabla}I^{t+1} + \sigma I^{t+1}\\ &= \frac{\sigma_s}{4\pi}\phi^{t+1} + \sigma_a\left[B^{old} + \frac{\partial B^{old}}{\partial T}\left\{\frac{1}{\dfrac{C_v}{\Delta t} + \displaystyle\int_0^{\infty}\sigma_a\frac{\partial B^{old}}{\partial T}d\nu}\left[\int_0^{\infty}\sigma_a\phi^{t+1}d\nu - 4\pi\int_0^{\infty}\sigma_a B^{old}d\nu\right]\right\}\right].\end{aligned} \tag{3.17}$$

Eq. (3.17) can be written in the form

$$\vec{\Omega}\bullet\vec{\nabla}I + \tilde{\sigma}I = \frac{1}{4\pi}\sigma_s\left(\nu\right)\phi\left(\nu\right) + \Lambda\Gamma\int_0^{\infty}\sigma_a\left(\nu'\right)\phi\left(\nu'\right)d\nu' + S \tag{3.18}$$

where

$$\Lambda = \frac{\int_0^\infty \sigma_a \frac{\partial B^{old}}{\partial T} d\nu}{\frac{C_p}{\Delta t} + \int_0^\infty \sigma_a \frac{\partial B^{old}}{\partial T} d\nu},$$

$$\Gamma = \frac{\sigma_a \frac{\partial B^{old}}{\partial T}}{\int_0^\infty \sigma_a \frac{\partial B^{old}}{\partial T} d\nu}, \tag{3.19}$$

$$\tilde{\sigma} = \sigma + \frac{1}{c\Delta t},$$

$$S = \frac{I^t}{c\Delta t} + \sigma_a B^{old} - \Lambda\Gamma \int_0^\infty \sigma_a B^{old} d\nu.$$

We have successfully generated a time-independent transport equation for intensity at time *t+1*, the end of the time step. This equation is now dependent on frequency, direction, and position only. We note that the Planck function does not have to be defined explicitly in Eq. (3.17). If we apply a Newton-Raphson iteration, we can converge the Planck function such that it is defined at the end of the time step, making the equation implicit in *I*, *T*, and *B(T)*.

At this point, one could employ either a Monte Carlo or a deterministic method to solve the transport equation. We note that we have already introduced discretization error into our solution from the time differencing scheme. This discretization error will be found in both Monte Carlo and deterministic methods. A Monte Carlo method is applied to the transport system by statistically sampling probability functions of known physical quantities, and using the statistical results to move particles through the system. Monte Carlo methods do not require discretization of independent variables except for purposes of tallying integrals of the intensity over pre-defined portions of the phase-space volume. However, a large number of particle histories must be simulated in order to generate accurate integrals over a large number of small phase-space volumes, and this is

computationally expensive for many problems of practical interest. Deterministic methods readily generate the intensity throughout phase-space, but this intensity is the result of discrete approximations that introduce error. For the remainder of our discussion, we focus on deterministic methods.

Multigroup frequency discretization

We now turn our attention to discretizing the frequency or energy variable. A standard way to discretize this variable is to apply a multigroup method. The multigroup method breaks the frequency variable into discrete frequency intervals and "groups" the photons according to these intervals [18]. Once the boundaries of the groups are defined, we can integrate our time independent transport equation over the energy range in group *g*. Here, we define our group structure to be numbered from lowest frequency to highest frequency. This ordering is opposite that in typical neutronics calculations.

$$\begin{aligned}&\int_{\nu_{g-1/2}}^{\nu_{g+1/2}} d\nu\left(\vec{\Omega}\bullet\vec{\nabla}I(\nu)+\tilde{\sigma}I(\nu)\right)\\ &=\frac{1}{4\pi}\int_{\nu_{g-1/2}}^{\nu_{g+1/2}} d\nu\sigma_s(\nu)\phi(\nu)+\int_{\nu_{g-1/2}}^{\nu_{g+1/2}} d\nu\Lambda(\nu)\Gamma\int_0^{\infty}\sigma_a(\nu')\phi(\nu')d\nu'+\int_{\nu_{g-1/2}}^{\nu_{g+1/2}} d\nu S(\nu).\end{aligned} \tag{3.20}$$

We define a group angular intensity to be the intensity integrated over the group frequencies.

$$I_g=\int_{\nu_{g-1/2}}^{\nu_{g+1/2}} d\nu I(\nu). \tag{3.21}$$

Also, we define approximate average multigroup opacities through a weighted average over the energy group, where the weighting function, *f*, is some estimate of the intensity spectrum over that group. We write these multigroup opacities as

$$\sigma_g = \frac{\int_{\nu_{g-1/2}}^{\nu_{g+1/2}} d\nu \sigma(\nu) f(\nu)}{\int_{\nu_{g-1/2}}^{\nu_{g+1/2}} d\nu f(\nu)}. \tag{3.22}$$

Often, we approximate this weighting function, *f*, as the Planck function, which makes sense based on the equilibrium result shown in Chapter II. We also note that if we have an integral over the entire frequency range, this integral simplifies to a sum of the group parameter over all the groups:

$$\int_0^\infty d\nu F(\nu) = \sum_{g=1}^{G} \int_{\nu_{g-1/2}}^{\nu_{g+1/2}} d\nu F(\nu) = \sum_{g=1}^{G} F_g . \tag{3.23}$$

Given the definitions in Eqs. (3.21)-(3.23), we can now write Eq. (3.20) in multigroup form as

$$\vec{\Omega} \bullet \vec{\nabla} I_g + \tilde{\sigma}_g I_g = \frac{1}{4\pi} \sigma_{s,g} \phi_g + \frac{1}{4\pi} \Lambda \Gamma_g \sum_{g'=1}^{G} \sigma_{a,g'} \phi_{g'} + S_g , \tag{3.24}$$

where

$$\begin{aligned}
\Lambda &= \frac{\sum_{g=1}^{G} \bar{\sigma}_{a,g} \frac{\partial B_g}{\partial T}}{\frac{C_p}{\Delta t} + \sum_{g=1}^{G} \bar{\sigma}_{a,g} \frac{\partial B_g}{\partial T}}, \\
\Gamma_g &= \frac{\bar{\sigma}_{a,g} \frac{\partial B_g}{\partial T}}{\sum_{g'=1}^{G} \bar{\sigma}_{a,g'} \frac{\partial B_{g'}}{\partial T}}, \\
\tilde{\sigma}_g &= \sigma_g + \frac{1}{c\Delta t}, \\
S_g &= \frac{I_g^t}{c\Delta t} + \hat{\sigma}_{a,g} B_g - \Lambda \Gamma_g \sum_{g'=1}^{G} \hat{\sigma}_{a,g'} B_{g'}.
\end{aligned} \tag{3.25}$$

Each of the $\sigma_{a,g}$ in Eq. (3.25) have slightly different notations because each opacity can be defined with its own, distinct weighting spectrum. Eq. (3.24) generates a coupled system of time-independent, one-group transport equations dependent on direction and position. This multigroup method will be accurate if the averaged opacities are accurate. If the shape of the opacity over the group energy range is slowly varying the opacity will be accurate for that energy; or if the weighting spectrum is an accurate representation of the angular intensity spectrum, the group opacity will be accurate.

Angular discretizations

Before we begin our discussion of how to angularly discretize Eq. (3.24), we need to complicate the equation. Throughout this dissertation we have assumed that all scattering results in an isotropic distribution of particles. In other words, if a particle undergoes a scattering interaction, it is equally likely to scatter to any direction. This assumption, although sometimes a sufficiently good approximation, is not true in general. For example, the dominant scattering event for photons in many radiative-transfer systems is Compton scattering. Compton scattering is not isotropic. As a result, we modify Eq. (3.24) to be general for any type of scattering.

$$\vec{\Omega}\bullet\vec{\nabla}I\left(\vec{\Omega}\right)+\tilde{\sigma}I\left(\vec{\Omega}\right)=\int_{4\pi}d\Omega'\tilde{\sigma}_s\left(\vec{\Omega}'\rightarrow\vec{\Omega}\right)I\left(\vec{\Omega}'\right)+S \tag{3.26}$$

We begin by developing an approximation for the scattering term, the first term on the right hand side of Eq. (3.26). If we assume that the material is isotropic, so that interaction rates are independent of the initial direction of the photon, then the scattering opacity depends only on the scattering angle, or the cosine between the incoming direction and the outgoing direction. As a result, we can rewrite the scattering opacity as

$$\tilde{\sigma}_s\left(\vec{\Omega}'\rightarrow\vec{\Omega}\right)=\frac{1}{2\pi}\tilde{\sigma}_s\left(\vec{\Omega}'\bullet\vec{\Omega}\right)=\frac{1}{2\pi}\tilde{\sigma}_s\left(\mu_0\right). \tag{3.27}$$

We can now expand this opacity in terms of Legendre polynomials. When we truncate this expansion, we have approximated the scattering opacity as an L^{th} –order polynomial.

$$\tilde{\sigma}_s\left(\vec{\Omega}'\to\vec{\Omega}\right)=\frac{1}{2\pi}\tilde{\sigma}_s\left(\mu_0\right)=\sum_{l=0}^{\infty}\frac{2l+1}{4\pi}\sigma_s^l P_l\left(\mu_0\right)\approx\sum_{l=0}^{L}\frac{2l+1}{4\pi}\sigma_s^l P_l\left(\mu_0\right), \qquad (3.28)$$

where σ_s^l is the l^{th} angular moment of the scattering opacity.

$$\sigma_s^l=\int_{-1}^{1} d\mu_0\sigma_s\left(\mu_0\right)P_l\left(\mu_0\right). \qquad (3.29)$$

To complete our approximation of the scattering term, we develop an angular approximation for the angular intensity by expanding the angular intensity in terms of spherical harmonics functions, Y [18].

$$I\left(\vec{\Omega}'\right)=\frac{1}{4\pi}\sum_{k=0}^{\infty}\sum_{m=-k}^{k}\phi_{k,m}Y_{k,m}\left(\vec{\Omega}'\right), \qquad (3.30)$$

where $\phi_{k,m}$ are the angular moments of the angular intensity

$$\phi_{k,m}=\int_{4\pi} d\Omega' I\left(\vec{\Omega}'\right)Y_{k,m}^{*}\left(\vec{\Omega}'\right). \qquad (3.31)$$

In Eq. (3.31), $Y_{k,m}^{*}\left(\vec{\Omega}'\right)$ are the complex conjugates of the spherical harmonics functions. We note that the spherical harmonics functions are a complete set of basis functions for the space defined by $\vec{\Omega}$ and they are orthogonal:

$$\int_{4\pi} d\Omega Y_{k,m}\left(\vec{\Omega}\right)Y_{k',m'}^{*}\left(\vec{\Omega}\right)=4\pi\delta_{kk'}\delta_{mm'}. \qquad (3.32)$$

We now substitute Eqs. (3.30) and (3.28) into the scattering term, resulting in

$$\int_{4\pi} d\Omega'\tilde{\sigma}_s\left(\vec{\Omega}'\to\vec{\Omega}\right)I\left(\vec{\Omega}'\right)=\int_{4\pi} d\Omega'\sum_{l=0}^{\infty}\frac{2l+1}{4\pi}\sigma_s^l P_l\left(\mu_0\right)\frac{1}{4\pi}\sum_{k=0}^{\infty}\sum_{m=-k}^{k}\phi_{k,m}Y_{k,m}\left(\vec{\Omega}'\right). \qquad (3.33)$$

We can now use the Spherical Harmonics addition Theorem

$$P_l\left(\mu_0\right) = P_l\left(\vec{\Omega}'\bullet\vec{\Omega}\right) = \frac{1}{2l+1}\sum_{n=-l}^{l} Y_{l,n}^{*}\left(\vec{\Omega}'\right)Y_{l,n}\left(\vec{\Omega}\right), \tag{3.34}$$

to simplify Eq. (3.33)

$$\begin{aligned}
&\int_{4\pi} d\Omega'\tilde{\sigma}_s\left(\vec{\Omega}'\rightarrow\vec{\Omega}\right)I\left(\vec{\Omega}'\right) \\
&= \int_{4\pi} d\Omega'\sum_{l=0}^{\infty}\frac{2l+1}{4\pi}\sigma_s^l\left(\frac{1}{2l+1}\sum_{n=-l}^{l} Y_{l,n}^{*}\left(\vec{\Omega}'\right)Y_{l,n}\left(\vec{\Omega}\right)\right)\left(\frac{1}{4\pi}\sum_{k=0}^{\infty}\sum_{m=-k}^{k}\phi_{k,m}Y_{k,m}\left(\vec{\Omega}'\right)\right) \\
&= \sum_{l=0}^{\infty}\frac{1}{4\pi}\sigma_s^l\sum_{n=-l}^{l}Y_{l,n}\left(\vec{\Omega}\right)\frac{1}{4\pi}\sum_{k=0}^{\infty}\sum_{m=-k}^{k}\phi_{k,m}\int_{4\pi} d\Omega' Y_{l,n}^{*}\left(\vec{\Omega}'\right)Y_{k,m}\left(\vec{\Omega}'\right).
\end{aligned} \tag{3.35}$$

We now substitute Eq. (3.32) for the integral in Eq. (3.35).

$$\int_{4\pi} d\Omega'\tilde{\sigma}_s\left(\vec{\Omega}'\rightarrow\vec{\Omega}\right)I\left(\vec{\Omega}'\right) = \sum_{l=0}^{\infty}\frac{1}{4\pi}\sigma_s^l\sum_{n=-l}^{l}Y_{l,n}\left(\vec{\Omega}\right)\phi_{l,n}\,. \tag{3.36}$$

Because the spherical harmonics functions are orthogonal, the integration picks out the *l,n* angular flux moment, greatly simplifying the approximation. Eq. (3.36) is exact for any type of scattering. If we truncate the summation in Eq. (3.36) to *L* terms, this expression can still be exact as long as the scattering in the problem can be perfectly described with only *L* moments of the scattering opacity, or if the angular flux can be perfectly described with moments through order *L*. We substitute Eq. (3.36) into Eq. (3.26) to generate a time-independent, single-energy, transport equation with an approximated scattering source

$$\vec{\Omega}\bullet\vec{\nabla}I\left(\vec{\Omega}\right) + \tilde{\sigma}I\left(\vec{\Omega}\right) = \sum_{l=0}^{L}\frac{1}{4\pi}\sigma_s^l\sum_{n=-l}^{l}Y_{l,n}\left(\vec{\Omega}\right)\phi_{l,n} + S\,, \tag{3.37}$$

where

$$\phi_{l,n} = \int_{4\pi} d\Omega I\left(\vec{\Omega}\right) Y_{l,n}^{*}\left(\vec{\Omega}\right). \tag{3.38}$$

The system of equations, Eqs. (3.37) and (3.38), can now be solved for the angular intensity, $I\left(\vec{\Omega}\right)$. However, because $I\left(\vec{\Omega}\right)$ appears on both sides of Eq. (3.37), we have to discretize the angular variable. We can apply this discretization to either $I\left(\vec{\Omega}\right)$, or to the integral in Eq. (3.38).

To discretize $I\left(\vec{\Omega}\right)$ directly, we approximate the angular intensity in terms of some selected set of basis functions.

$$I\left(\vec{\Omega}\right) = \sum_{m=1}^{M} I_m u_m\left(\vec{\Omega}\right), \tag{3.39}$$

We then substitute this expansion into Eqs. (3.37) and (3.38)

$$\vec{\Omega}\bullet\vec{\nabla}\sum_{m=1}^{M} I_m u_m\left(\vec{\Omega}\right) + \tilde{\sigma}\sum_{m=1}^{M} I_m u_m\left(\vec{\Omega}\right) = \sum_{l=0}^{L}\frac{1}{4\pi}\sigma_s^l\sum_{n=-l}^{l} Y_{l,n}\left(\vec{\Omega}\right)\phi_{l,n} + S \tag{3.40}$$

and

$$\phi_{l,n} = \sum_{m=1}^{M} I_m \int_{4\pi} d\Omega u_m\left(\vec{\Omega}\right) Y_{l,n}^{*}\left(\vec{\Omega}\right) \tag{3.41}$$

We combine Eqs. (3.40) and (3.41), multiply by a weighting function and integrate over all direction space, resulting in

$$\begin{aligned}&\int_{4\pi} d\Omega v_i\left(\vec{\Omega}\right)\left\{\vec{\Omega}\bullet\vec{\nabla}\sum_{m=1}^{M} I_m u_m\left(\vec{\Omega}\right) + \tilde{\sigma}\sum_{m=1}^{M} I_m u_m\left(\vec{\Omega}\right)\right\}\\ &= \int_{4\pi} d\Omega v_i\left(\vec{\Omega}\right)\left\{\sum_{l=0}^{L}\frac{1}{4\pi}\sigma_s^l\sum_{n=-l}^{l} Y_{l,n}\left(\vec{\Omega}\right)\sum_{m=1}^{M} I_m\int_{4\pi} d\Omega u_m\left(\vec{\Omega}\right) Y_{l,n}^{*}\left(\vec{\Omega}\right) + S\right\}.\end{aligned} \tag{3.42}$$

We apply *M* weight functions to Eq. (3.42), creating a system of *M* equations and *M* unknowns. Often spherical harmonics functions are used as the basis functions in the functional expansion of the angular intensity. A disadvantage of expanding the angular intensity in terms of basis functions is that most sets of basis functions are smooth; therefore, they do not represent discontinuities in the angular flux well.

A second way to discretize the angular variable, which we have used in this research, is known as the discrete ordinates method. To apply this method, we approximate the angular flux moment integrals in Eq. (3.38) using quadrature sets:

$$\phi_{l,n} = \int_{4\pi} d\Omega I\left(\vec{\Omega}\right) Y_{l,n}^{*}\left(\vec{\Omega}\right) \approx \sum_{m=1}^{M} w_m I\left(\vec{\Omega}_m\right) Y_{l,n}^{*}\left(\vec{\Omega}_m\right). \tag{3.43}$$

We substitute this approximation into Eq. (3.40)

$$\vec{\Omega}\bullet\vec{\nabla} I\left(\vec{\Omega}\right) + \tilde{\sigma} I\left(\vec{\Omega}\right) = \frac{1}{4\pi}\sum_{l=0}^{L}\sigma_s^l \sum_{n=-l}^{l} Y_{l,n}\left(\vec{\Omega}\right)\sum_{m=1}^{M} w_m I\left(\vec{\Omega}_m\right) Y_{l,n}^{*}\left(\vec{\Omega}_m\right) + S\,. \tag{3.44}$$

The right hand side of Eq. (3.44) depends on a set of *M* angular intensities defined at the quadrature points $\vec{\Omega}_m$. As a result, we have to solve Eq. (3.44) at each discrete quadrature point.

$$\vec{\Omega}\bullet\vec{\nabla} I_m + \tilde{\sigma} I_m = \frac{1}{4\pi}\sum_{l=0}^{L}\sigma_s^l \sum_{n=-l}^{l} Y_{l,n}\left(\vec{\Omega}\right)\sum_{m=1}^{M} w_m I_m Y_{l,n}^{*}\left(\vec{\Omega}_m\right) + S\,, \tag{3.45}$$

where

$$I_m \equiv I\left(\vec{\Omega}_m\right). \tag{3.46}$$

This method also generates a set of *M* equations with *M* unknowns, but the coupling in this method occurs only in the scattering source. The main disadvantage of the discrete ordinates method is commonly referred to as ray effects. Because we solve the equation

at discrete directions in the direction space, we bias the solution along those discrete directions.

Finally, we observe that Eq. (3.45) suggests an application of a simple iterative method, which we call source iteration. During source iteration we guess a solution of the angular intensity and substitute this guess into the scattering source of the equation. We then solve the equation for every quadrature point over all space and determine if the new solution is sufficiently close to the guessed solution. If the new solution equals the guessed solution we have iterated to convergence; if not, we begin a new iterative step with the solution from our previous iteration as the new guess in the scattering source. Source iteration is a very useful tool for solving these problems. It has a very unique physical significance in that each iteration represents a scattering event for a particle.

Iterative methods for solving the angularly discretized transport equation

To demonstrate source iteration as a linear algebra problem, we rewrite the linear system of equations in Eq. (3.44) in matrix notation.

$$\underline{\underline{T}}\vec{\psi} - \underline{\underline{S}}\vec{\psi} = \vec{Q}\,, \tag{3.47}$$

where $\underline{\underline{T}}$ is the transport operator, or the left hand side operator in Eq. (3.44), $\underline{\underline{S}}$ is the scattering operator, $\vec{Q}$ is the fixed source vector, and $\vec{\psi}$ is the angular flux unknown vector. We can now apply source iteration to Eq. (3.47).

$$\begin{aligned} \vec{\psi}^{\,p+1} &= \underline{\underline{T}}^{-1}\underline{\underline{S}}\vec{\psi}^{\,p} + \underline{\underline{T}}^{-1}\vec{Q}, \\ \vec{\psi}^{\,0} &= \vec{v}, \end{aligned} \tag{3.48}$$

where *p* represents the iteration number and $\vec{v}$ represents our initial guess. If we can find $\underline{\underline{T}}^{-1}$ easily Eq. (3.48) represents a potentially successful iterative method. If we

have applied a spatial discretization to the *T* operator, we find $\underline{\underline{T}}^{-1}$ by solving each space dependent discrete direction equation independently. This process is called "sweeping."

To determine the robustness of the iterative method is, we examine how the method approaches its solution by first defining an iteration error

$$e^p = \psi^{converged} - \psi^p . \tag{3.49}$$

We then write Eq. (3.48) for the converged solution and for the *p+1* iteration.

$$\begin{aligned} \vec{\psi}^{converged} &= \underline{\underline{T}}^{-1}\underline{\underline{S}}\vec{\psi}^{converged} + \underline{\underline{T}}^{-1}\vec{Q} \\ \vec{\psi}^{p+1} &= \underline{\underline{T}}^{-1}\underline{\underline{S}}\vec{\psi}^{p} + \underline{\underline{T}}^{-1}\vec{Q} \end{aligned} \tag{3.50}$$

If we subtract the two equations in Eq. (3.50), and substitute the definition of the iteration error into the resultant equation, we arrive at an expression for how the error changes with each iteration.

$$\begin{aligned} &\vec{e}^{\,p+1} = \underline{\underline{T}}^{-1}\underline{\underline{S}}\vec{e}^{\,p} = \sum_{n=1}^{p} \lambda_i^n \alpha_i v_i \\ &\text{where} \quad \vec{e}^{\,0} = \sum_i \alpha_i v_i \quad \text{and} \quad \underline{\underline{T}}^{-1}\underline{\underline{S}} v_i = \lambda_i v_i \ . \end{aligned} \tag{3.51}$$

Here λ_i represents the eigenvalue of $\underline{\underline{T}}^{-1}\underline{\underline{S}}$ associated with the eigenvector v_i. Eq. (3.51) indicates that the error in the *i*th component of the solution is decreased by a factor of $\frac{1}{\lambda_i}$ after each iteration. As a result, if the largest eigenvalue of $\underline{\underline{T}}^{-1}\underline{\underline{S}}$ is less than 1, the method will be unconditionally convergent. For our matrix and for optically thick problems, the largest eigenvalue approaches $\frac{\sigma_s}{\sigma}$, which is often called the scattering ratio and denoted by *c*. As a result, source iteration is unconditionally convergent. However, as the problem becomes scattering dominated the scattering ratio approaches one, and the factor by which the error is decreased during each iteration is very small.

As a result, source iteration can take a prohibitively large number of iterations to converge in highly scattering problems. Our problems are effectively scattering dominated, so better iterative methods are necessary to solve our problem.

In order to find new iterative methods to solve our problem, we will recast our matrix problem into a mathematically familiar form. First we define a scattering source term as

$$\vec{\Lambda} = \underline{\underline{S}}\vec{\psi}\,, \tag{3.52}$$

and substitute it into Eq. (3.48)

$$\vec{\psi}^{\,p+1} = \underline{\underline{T}}^{-1}\vec{\Lambda}^{\,p} + \underline{\underline{T}}^{-1}\vec{Q}\,. \tag{3.53}$$

We can then multiply Eq. (3.53) by $\underline{\underline{S}}$ resulting in

$$\vec{\Lambda}^{\,p+1} = \underline{\underline{S}}\,\underline{\underline{T}}^{-1}\vec{\Lambda}^{\,p} + \underline{\underline{S}}\,\underline{\underline{T}}^{-1}\vec{Q}\,. \tag{3.54}$$

Now, we are iterating to find the scattering source instead of the angular intensity, but we are still using the same operator as our iteration matrix. We can now see that the equation we are trying to solve with the iterative method given in Eq. (3.54) is

$$\underline{\underline{H}}\,\vec{\Lambda} = \underline{\underline{S}}\,\underline{\underline{T}}^{-1}\vec{Q}\,, \tag{3.55}$$

where $\underline{\underline{H}} = \left(I - \underline{\underline{S}}\,\underline{\underline{T}}^{-1}\right)$. We now apply the well-known Richardson iteration to Eq. (3.55)

$$\vec{\Lambda} = \left(\underline{\underline{I}} - \underline{\underline{H}}\right)\vec{\Lambda} + \underline{\underline{S}}\,\underline{\underline{T}}^{-1}\vec{Q}\,, \tag{3.56}$$

and simplify it using the definition of *H* and *Λ*

$$\vec{\psi} = \underline{\underline{T}}^{-1}\underline{\underline{S}}\vec{\psi} + \underline{\underline{T}}^{-1}\vec{Q}\,. \tag{3.57}$$

This result is the same as directly applying source iteration to the angular intensity. As a result, source iteration is equivalent to Richardson iteration applied to the scattering source [18]. Because we have found a way to cast our problem as a standard linear algebra iterative technique in which we iterate on the scattering source, we can now use all of the mathematical tools developed to improve the performance of these techniques. One such technique is to precondition the matrix with a good approximate matrix whose inverse is easier to find. A standard approximation to the transport solution is a diffusion solution. As a result, using a diffusion matrix as a preconditioner to the Richardson iteration is very effective at accelerating source iteration. We often refer to preconditioning as acceleration. Furthermore, because we now understand our iterative methods in a mathematical setting, we can apply better iterative methods, such as Conjugate Gradient or GMRES, to our system, with or without preconditioning. Unlike Richardson iteration, if we use a diffusion matrix as a preconditioner for Krylov methods we are freed somewhat from the requirement for strict consistency and we can apply the same spatial diffusion approximation to different spatially discretized transport equations [19, 20].

Applying a spatial discretization

We have now successfully discretized the time, frequency, and angular variables of the radiation transport equation producing a steady state, mono-frequency spatially dependent transport equation along one quadrature direction. The focus of this research is to develop new spatial discretizations for this form of the transport equation. When we compare spatial discretizations, we will assume that each discretization has the same time, frequency and angular discretization.

In order to spatially discretize the transport equation, we generally divide the physical domain into a mesh composed of cells or zones, and apply approximations over those cells. For this research, we are required to develop methods that are successful on meshes composed of arbitrary polygonal cells (2D RZ) and arbitrary polyhedral cells

(3D XYZ). An arbitrary polygon is any two dimensional shape that has straight edges that do not cross each other. Examples of polygonal cells are shown in Figure 3.1.

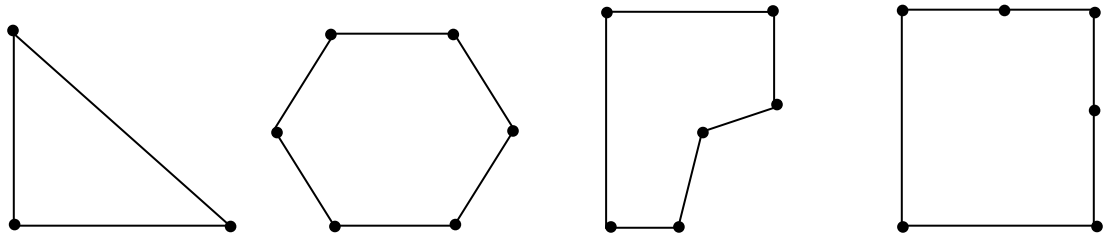

Figure 3.1: Arbitrary polygons

An arbitrary polyhedral cell is a three-dimensional shape with an arbitrary number of faces. Examples of these cell types are shown in Figure 3.2. We note that our definition of a polyhedral cell allows for faces constructed from non-co-planar vertices. We will discuss how we treat these faces later.

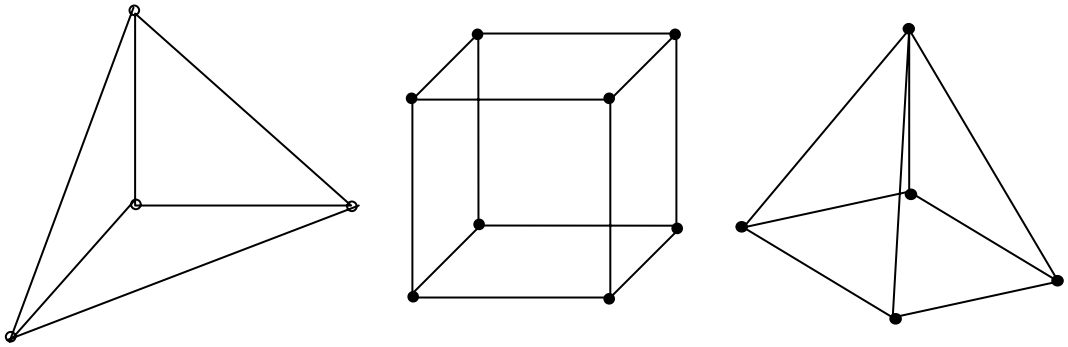

Figure 3.2: Arbitrary polyhedra

Generally, we would like to develop discretizations for these cell types for two reasons. First, using these cell types can potentially reduce unknowns in our mesh, while maintaining symmetry within the mesh. We show this potential reduction in unknown

count for a two-dimensional hexahedral cell versus the same space divide symmetrically using quadrilaterals in Figure 3.3. We illustrate assuming an unknown for every vertex in every cell, which is standard for transport discretizations that perform well in thick diffusive problems. In this example, a method that can directly treat a hexagon uses six unknowns whereas a quadrilateral-based method uses 24.

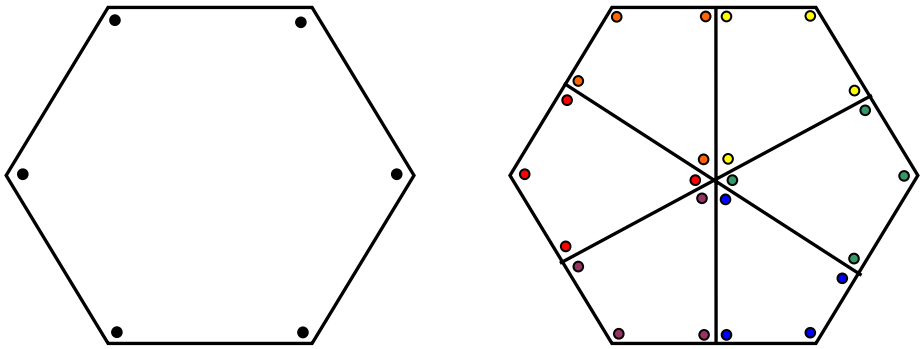

Figure 3.3: Two-dimensional cells with the position of the unknowns

Secondly, we can use the concept of the arbitrary polygonal and polyhedral cell to generate methods for adaptive mesh refinement (AMR) problems. In Figure 3.4 we show a cell with neighboring cells that have been refined such that the faces between the cell and its neighbors do not line up. This front face of this cell has four neighbors. If we think of this cell as a 9-face polyhedron instead of a hexahedron, we can apply our methods that work on polyhedral cells directly to this problem without having to further approximate the coupling between cells.

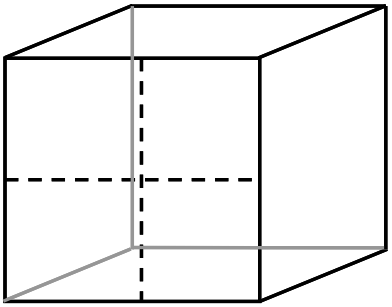

Figure 3.4: An AMR cell in three dimensions

When developing methods for radiation transport problems discretized on arbitrary polygons and polyhedra, we require that the methods be accurate and robust, satisfy important physical limits such as the diffusion limit, and are computationally efficient. Very few spatial discretizations exist that meet any of our requirements. As a result, we have developed the Piecewise Linear Discontinuous Finite Element Method (PWLD) for RZ and XYZ, which uses piecewise linear basis functions developed by Stone and Adams. Stone and Adams have shown this method to be successful for XY radiation transport problems [3,4].

Another possible method is a discontinuous finite element method that uses Wachspress rational basis functions. This method is accurate in the diffusion limit for our problems of interest. However, integrals the Wachspress functions must be performed numerically, creating a significant computational cost. Furthermore, using Wachspress basis functions for AMR problems is completely intractable because the basis functions become delta functions in this case [8,9].

A third potential method for our problem is the Upstream Corner Balance method (UCB), which also is accurate in the thick diffusion limit on our problems. This method divides the cells into sub-cell volumes and makes approximations in the sub-cells. Unlike traditional discontinuous finite element methods, UCB is able to sweep through the sub-cells instead of creating a cell based matrix to invert. This sub-cell sweeping

property makes UCB extremely computationally efficient. However, current versions of UCB can experience solution oscillations when cells become sufficiently distorted and have discontinuous incoming intensities [11,12].

Another discretization that has the potential to meet our discretization requirements is a characteristics method. These methods approximate the source shape and outgoing intensity, while exactly inverting the left hand side of the spatially dependent transport equation. Characteristics methods are typically used in nuclear reactor calculations where it is not vital for the method to satisfy the diffusion limit. The type of characteristics methods used in these reactor calculations do not satisfy the diffusion limit, but are extremely accurate in other limits and are computationally efficient. The reason why these methods do not satisfy the diffusion limit is because their source approximation is not complex enough to admit this limit. Applying the PWL interpolating functions as the approximation for the source will potentially enable characteristics methods to satisfy the diffusion limit [13]. This type of method has not yet been examined in detail.

Finally, we have the option of solving the radiation transport problem with a standard Linear Discontinuous Finite Element Method on triangles and tetrahedra, a Bilinear Discontinuous Finite Element Method on quadrilaterals, or a Trilinear Discontinuous Finite Element Method on hexahedra. These methods applied to their cell types are accurate in the thick diffusion limit. Furthermore, extensive analysis has been performed on these methods, and mesh generation is well defined for their cell types. However, these methods are not applicable for general polygons and polyhedra, so they do not meet our requirements [7].

We note one more significant detail about our candidate spatial discretizations—many of them are discontinuous finite element methods (DFEMs). DFEMs have many nice mathematical properties. They are locally conservative for every cell and they are

second order accurate if the basis functions are linear. Furthermore, Adams has defined clear conditions under which DFEMs will satisfy the thick diffusion limit, allowing us to analyze our methods simply [6].

Summary

In this chapter, we briefly describe the entire discretization process for the radiation transport equation as it applies to the coupled system of radiation-hydrodynamics equations. We begin by separating the radiation system from the hydrodynamics system using operator splitting. We then apply an implicit time differencing scheme to the resultant radiative transfer equations noting that implicit time differencing is necessary for stability due to the extremely small time step required by an explicit solution. Next we apply a standard multi-group approximation to the steady state radiation transport equation, resulting in a set of coupled steady state, mono-frequency transport equations. Then, we discuss two ways to discretize the direction variables, and note that the discrete ordinates approximation lends itself to a solution technique called source iteration. We show that source iteration is really a standard linear algebra iterative method for solving a matrix called Richardson's iteration. After we apply operator splitting, the implicit time differencing, a multi-group frequency approximation, and a discrete ordinates angular discretization, we produce a steady state, mono-frequency, spatially dependent transport equation along one direction. To conclude our discretization, we define arbitrary polygonal and polyhedral meshes over which we wanted to apply a spatial discretization, and briefly described all known methods that are potentially successful spatial discretizations for the radiation transport problem.

In the next chapter we formally introduce the Piecewise Linear Discontinuous Finite Element Method applied to the RZ transport equation. We note that any further discussion of spatial discretizations in this dissertation assumes that the transport equation we are discretizing is the equation resulting from the implicit time

discretization, the multi-group frequency discretization, and the discrete ordinates angular discretization.

CHAPTER IV
PWLD IN RZ GEOMETRY

In this chapter we begin with some comments about the unique requirements of solving the discrete ordinates RZ transport equation. We change notation slightly from the previous chapters: the angular intensity unknown, I, becomes ψ. We make this change to use standard Nuclear Engineering notation.

We then apply a general discontinuous finite element discretization (DFEM) to the steady state, mono-frequency, discrete ordinates RZ transport equation. At this point, we introduce the Piecewise Linear basis functions (PWL) and apply them to our DFEM formulation resulting in the Piecewise Linear Discontinuous Finite Element Method (PWLD). When we are developing this method we take special care to show its close relationship to the DFEM that uses standard linear functions, know as linear discontinuous (LD). We then briefly discuss our implementation of this method into the Capsaicin code developed at Los Alamos National Laboratory, again noting the similarities between the PWLD and LD implementations in this code.

The subsequent section of this chapter is an asymptotic diffusion limit analysis for the RZ PWLD method. In this analysis we follow examples give by Adams [6] and Palmer [21]. This analysis provides motivation for lumping our method. As a result, we discuss two lumping methods that improve the behavior of our PWLD method in the diffusion limit.

We conclude this chapter with a discussion of a variety of problems with which we tested our method in the Capsaicin code. We note that the PWLD method produced exactly the same solution as the LD method on a triangular mesh, which is expected. We also note that the PWLD method performs as accurately and robustly as the DFEM that uses bi-linear basis functions (BLD) on arbitrary quadrilateral grids. We further

note that, unlike the BLD method, the PWLD method is extendable to arbitrary polygonal meshes. As a result, we conclude that PWLD is an excellent candidate spatial discretization of the RZ transport equation for arbitrary polygonal meshes.

Development of the PWLD method in RZ geometry

The direction and space dependent RZ particle transport equation is written in conservation form as:

$$\begin{aligned}\frac{\mu}{r}\frac{\partial}{\partial r}r\psi\left(r,z,\omega,\xi\right)-\frac{1}{r}\frac{\partial}{\partial\omega}\eta\psi\left(r,z,\omega,\xi\right)+\xi\frac{\partial}{\partial z}\psi\left(r,z,\omega,\xi\right)+\sigma\left(r,z\right)\psi\left(r,z,\omega,\xi\right)\\ =\int_{4\pi}\sigma_s\left(r,z,\vec{\Omega}\bullet\vec{\Omega}'\right)\psi\left(r,z,\vec{\Omega}'\right)d\Omega'+S\left(r,z,\omega,\xi\right),\end{aligned} \tag{4.1}$$

where ψ is the unknown angular intensity, σ is the macroscopic total cross section, σ_s is the macroscopic scattering cross section, and $\vec{\Omega}$ is the direction in three dimensional space about which the particle is traveling, where

$$\begin{aligned}\mu&=\left(1-\xi^2\right)^{1/2}\cos\omega,\\ \eta&=\left(1-\xi^2\right)^{1/2}\sin\omega,\\ \xi&=\cos\Omega,\\ \omega&=\cos^{-1}\left(\mu\right).\end{aligned} \tag{4.2}$$

We apply a discrete ordinates angular discretization to Eq. (1.1). We recall from Chapter III, that the scattering kernel can be defined in terms of spherical harmonics functions:

$$\int_{4\pi}\sigma_s\left(r,z,\vec{\Omega}\bullet\vec{\Omega}'\right)\psi\left(r,z,\vec{\Omega}'\right)d\Omega'=\sum_{l=0}^{L}\frac{2l+1}{4\pi}\sigma_{s,l}\left(r,z\right)\sum_{k=-l}^{l}Y_{l,k}\left(\vec{\Omega}\right)\phi_{l,k}\ , \tag{4.3}$$

where ϕ_l is the l^{th} angular moment of the angular flux defined as

$$\phi_{l,k} = \int_{4\pi} d\Omega \psi\left(\vec{\Omega}\right) Y_{l,k}^{*}\left(\vec{\Omega}\right) \approx \sum_{n=1}^{N}\sum_{m=1}^{M_n} w_{m,n}\psi\left(\vec{\Omega}_{m,n}\right) Y_{l,k}^{*}\left(\vec{\Omega}_{m,n}\right) \tag{4.4}$$

and $\sigma_{s,l}$ is the l^{th} coefficient in the expansion of the scattering cross section.

For these calculations, it is standard to use a level-symmetric quadrature set that has N levels of discrete ξ values, each the nth level having M_n number of discrete μ values. Other quadrature sets are also used. We do not restrict the quadrature set in our work except to require that it be symmetric, with the weights summing to 4π. For the discrete ordinate discretization, we solve for the angular flux at every (m,n) discrete direction in the quadrature set.

One main difficulty in solving the RZ transport equation with the discrete ordinates angular discretization is the angular derivative term, which is the second term in Eq. (1.1). We discretize this angular derivative term with angular differencing coefficients such that a constant solution in space and angle is preserved:

$$\begin{aligned} &\frac{\mu_{m,n}}{r}\frac{\partial}{\partial r} r\psi_{m,n}\left(r,z\right) + \frac{\alpha^{n}_{m+1/2}\psi_{m+1/2,n}\left(r,z\right) - \alpha^{n}_{m-1/2}\psi_{m-1/2,n}\left(r,z\right)}{r w_{m,n}} + \xi_{m,n}\frac{\partial}{\partial z}\psi_{m,n}\left(r,z\right) \\ &+\sigma\psi_{m,n}\left(r,z\right) = \frac{1}{4\pi}\sum_{l=0}^{L}\sigma_{s,l}\left(r,z\right)\sum_{k=-l}^{l} Y_{l,k}\left(\vec{\Omega}_{m,n}\right)\phi_{l,k}\left(r,z\right) + \frac{1}{4\pi}S\left(r,z\right), \end{aligned} \tag{4.5}$$

where

$$\begin{aligned} \alpha^{n}_{m+1/2} &= \alpha^{n}_{m-1/2} - \mu_{m,n} w_{m,n}, \\ \alpha^{n}_{1/2} &= \alpha^{n}_{M_n+1/2} = 0, \end{aligned} \tag{4.6}$$

and we have assumed an isotropic fixed source. Given the definitions in Eq. (4.6), it is easy to show that the alphas preserve these constant solutions. If we have a constant solution then the gradient of that solution should be zero. As a result, for constant solutions

$$\frac{\mu_{m,n}}{r}\frac{\partial}{\partial r}r\psi_{m,n}(r,z)+\frac{\alpha^{n}_{m+\frac{1}{2}}\psi_{m+\frac{1}{2},n}(r,z)-\alpha^{n}_{m-\frac{1}{2}}\psi_{m-\frac{1}{2},n}(r,z)}{rw_{m,n}} \\ +\xi_{m,n}\frac{\partial}{\partial z}\psi_{m,n}(r,z)=0. \tag{4.7}$$

Substituting in $\psi = c$ results in

$$\frac{\mu_{m,n}c}{r}+\frac{\left(\alpha^{n}_{m+\frac{1}{2}}-\alpha^{n}_{m-\frac{1}{2}}\right)c}{rw_{m,n}}=0. \tag{4.8}$$

From Eq. (4.6), we know that

$$\left(\alpha^{n}_{m+\frac{1}{2}}-\alpha^{n}_{m-\frac{1}{2}}\right)=-\mu_{m,n}w_{m,n}\,, \tag{4.9}$$

forcing Eq. (4.8) to be true. Returning to Eq.(4.5), we now have successfully differenced the angular derivative term, but we have only one equation with 2 unknowns: $\psi_{m,n},\psi_{m+\frac{1}{2},n}$. We can assume the *n-1/2* term is known from old information, which implies a sweeping order, from $\mu=-\left(1-\xi^{2}\right)^{\frac{1}{2}}$ to $\mu=\left(1-\xi^{2}\right)^{\frac{1}{2}}$. As a result, we must solve for the $\mu=-\left(1-\xi^{2}\right)^{\frac{1}{2}}$ starting direction angular flux on each ξ-level, $\psi_{1/2,n}$, which is the angular flux associated with $\omega=\pi$. The resultant transport equation for the starting direction is equivalent to a transport equation in XY geometry along the starting direction, which is given by $\mu=-\left(1-\xi_{n}\right)^{\frac{1}{2}}$ and ξ_n.

$$-\left(1-\xi_{n}\right)^{\frac{1}{2}}\frac{\partial}{\partial r}\psi_{\frac{1}{2},n}(r,z)+\xi_{n}\frac{\partial}{\partial z}\psi_{\frac{1}{2},n}(r,z)+\sigma(r,z)\psi_{\frac{1}{2},n}(r,z) \\ =\frac{1}{4\pi}\sum_{l=0}^{L}\sigma_{s,l}(r,z)\sum_{k=-l}^{l}Y_{l,k}\left(\vec{\Omega}_{m,n}\right)\phi_{l,k}(r,z)+\frac{1}{4\pi}S(r,z) \tag{4.10}$$

We must develop one more angularly dependent relationship to relate the *m* and *m*+1/2 intensities and close the system. We define a weighted diamond relationship between the angular flux defined at the discrete ordinate and the angular fluxes at points in between the discrete ordinate directions. We define this relationship so that the a linear solution in μ will be preserved

$$\psi_{m,n} = \tau_{m,n}\psi_{m+\frac{1}{2},n} + \left(1-\tau_{m,n}\right)\psi_{m-\frac{1}{2},n}, \tag{4.11}$$

where the values of τ are defined such that

$$\mu_{m,n} = \tau_{m,n}\mu_{m+\frac{1}{2},n} + \left(1-\tau_{m,n}\right)\mu_{m-\frac{1}{2},n}, \tag{4.12}$$

and

$$\begin{aligned} \mu_{m+\frac{1}{2},n} &= \left(1-\xi_n^2\right)^{\frac{1}{2}}\cos\left(\gamma_{m+\frac{1}{2},n}\right), \\ \gamma_{m+\frac{1}{2},n} &= \gamma_{m-\frac{1}{2},n} + \pi\frac{w_{m,n}}{\sum_{m=1}^{M} w_{m,n}}. \end{aligned} \tag{4.13}$$

For non-starting directions, the equation resulting from the application of the angular discretization and the assumption of isotropic scattering and fixed source is

$$\begin{aligned} \frac{\mu_{m,n}}{r}\frac{\partial}{\partial r}r\psi_{m,n}(r,z) + \frac{\dfrac{\alpha^n_{m+\frac{1}{2}}\psi_{m,n}(r,z)}{\tau_{m,n}} - \left(\dfrac{1-\tau_{m,n}}{\tau_{m,n}}\alpha^n_{m+\frac{1}{2}} + \alpha^n_{m-\frac{1}{2}}\right)\psi_{m-\frac{1}{2},n}(r,z)}{rw_{m,n}} \\ +\xi_{m,n}\frac{\partial}{\partial z}\psi_{m,n}(r,z) + \sigma(r,z)\psi_{m,n}(r,z) \\ = \frac{1}{4\pi}\sum_{l=0}^{L}\sigma_{s,l}(r,z)\sum_{k=-l}^{l}Y_{l,k}\left(\vec{\Omega}_{m,n}\right)\phi_{l,k}(r,z) + \frac{1}{4\pi}S(r,z). \end{aligned} \tag{4.14}$$

From now on we will refer to the right hand side of Eq. (4.14) as *S*.

Finite element spatial discretization

In order to apply a general discontinuous finite element discretization to Eq. (4.14), we multiply the equation by a weight function, *v*, multiply by *r*, and integrate over a cell volume

$$\int_{A_{cell}} dAv_i \left\{ \mu_{m,n} \frac{\partial}{\partial r} r\psi_{m,n} + \frac{\frac{1}{\tau_{m,n}} \alpha^n_{m+1/2} \psi_{m,n}}{w_{m,n}} + \xi_{m,n} r \frac{\partial}{\partial z} \psi_{m,n} + \sigma r \psi_{m,n} \right\} = \int_{A_{cell}} dAv_i \left\{ rS + \frac{\left(\frac{1-\tau_{m,n}}{\tau_{m,n}} \alpha^n_{m+1/2} + \alpha^n_{m-1/2} \right)}{w_{m,n}} \psi_{m-1/2,n} \right\}, \tag{4.15}$$

where $dA = drdz$ and $A_{cell} = \Delta r \Delta z$.

Terms with derivatives on the left hand side of Eq. (4.15) are evaluated using Gauss's Divergence Theorem, resulting in

$$\oint_{\partial A_{cell}} dsv_i \vec{n} \bullet \vec{\Omega}_{m,n} r\tilde{\psi}_{m,n} - \int_{A_{cell}} dAr\psi_{m,n} \vec{\Omega}_{m,n} \bullet \vec{\nabla} v_i + \int_{A_{cell}} dAv_i \left[\frac{\frac{1}{\tau_{m,n}} \alpha^n_{m+1/2}}{w_{m,n}} \psi_{m,n} + \sigma r \psi_{m,n} \right] = \int_{A_{cell}} dAv_i \left\{ rS + \frac{\left(\frac{1-\tau_{m,n}}{\tau_{m,n}} \alpha^n_{m+1/2} + \alpha^n_{m-1/2} \right)}{w_{m,n}} \psi_{m-1/2,n} \right\}, \tag{4.16}$$

where $\tilde{\psi}_{m,n}$ denotes an angular flux unknown on the boundary of the cell. These surface quantities are determined by an upwinding condition.

$$\tilde{\psi}_{m,n} = \begin{cases} \psi_{m,n,cell} \;\; if \;\; \vec{n} \bullet \vec{\Omega}_{m,n} > 0 \\ \psi_{m,n,upwind\ cell} \;\; if \;\; \vec{n} \bullet \vec{\Omega}_{m,n} < 0 \end{cases} \tag{4.17}$$

We perform Gauss Divergence a second time on the second term in Eq. (4.16), resulting in

$$\begin{aligned} &\oint_{\partial A_{cell}} ds v_i r \vec{n} \bullet \vec{\Omega}_{m,n} \left[\tilde{\psi}_{m,n} - \psi_{m,n} \right] + \int_{A_{cell}} dA v_i \vec{\Omega}_{m,n} \bullet \vec{\nabla} \left(r \psi_{m,n} \right) \\ &\quad + \int_{A_{cell}} dA v_i \left[\frac{1}{\tau_{m,n}} \frac{\alpha^n_{m+1/2}}{w_{m,n}} \psi_{m,n} + \sigma r \psi_{m,n} \right] \\ &= \int_{A_{cell}} dA v_i \left\{ rS + \frac{\left(\frac{1-\tau_{m,n}}{\tau_{m,n}} \alpha^n_{m+1/2} + \alpha^n_{m-1/2} \right)}{w_{m,n}} \psi_{m-1/2,n} \right\}. \end{aligned} \tag{4.18}$$

The solution and source can be expanded in terms of basis functions, *u*

$$\begin{aligned} \psi_{m,n}\left(\vec{r}\right) &= \sum_{j=1}^{J} \psi_{m,n,j} u_j\left(\vec{r}\right), \\ S\left(\vec{r}\right) &= \sum_{j=1}^{J} S_j u_j\left(\vec{r}\right), \end{aligned} \tag{4.19}$$

and substituted into Eq. (4.18)

$$\oint_{\partial A_{cell}} ds v_i \vec{n}\bullet\vec{\Omega}_{m,n} r\left\{\left(\sum_{j=1}^{J}\tilde{\psi}_{m,n,j}u_j(\vec{r})\right)-\left(\sum_{j=1}^{J}\psi_{m,n,j}u_j(\vec{r})\right)\right\}$$
$$+\int_{A_{cell}} dA v_i \vec{\Omega}_{m,n}\bullet\vec{\nabla}\left[r\left(\sum_{j=1}^{J}\psi_{m,n,j}u_j(\vec{r})\right)\right]$$
$$+\int_{A_{cell}} dA v_i\left[\frac{\frac{1}{\tau_{m,n}}\alpha^{n}_{m+1/2}}{w_{m,n}}\left(\sum_{j=1}^{J}\psi_{m,n,j}u_j(\vec{r})\right)+\sigma r\left(\sum_{j=1}^{J}\psi_{m,n,j}u_j(\vec{r})\right)\right] \qquad (4.20)$$
$$=\int_{A_{cell}} dA v_i\left\{r\left(\sum_{j=1}^{J}S_j u_j(\vec{r})\right)+\frac{\left(\frac{1-\tau_{m,n}}{\tau_{m,n}}\alpha^{n}_{m+1/2}+\alpha^{n}_{m-1/2}\right)}{w_{m,n}}\sum_{j=1}^{J}\psi_{m-1/2,n,j}u_j(\vec{r})\right\}.$$

Eq. (4.20) represents a general form of any DFEM applied to the RZ transport equation. The general DFEM form of the its corresponding starting-direction equation is

$$\oint_{\partial A_{cell}} ds v_i \vec{n}\bullet\vec{\Omega}_{1/2,n}\left\{\left(\sum_{j=1}^{J}\tilde{\psi}_{1/2,n,j}u_j(\vec{r})\right)-\left(\sum_{j=1}^{J}\psi_{1/2,n,j}u_j(\vec{r})\right)\right\}$$
$$+\int_{A_{cell}} dA v_i\vec{\Omega}_{1/2,n}\bullet\vec{\nabla}\left[\left(\sum_{j=1}^{J}\psi_{1/2,n,j}u_j(\vec{r})\right)\right]+\int_{A_{cell}} dA v_i\sigma\left(\sum_{j=1}^{J}\psi_{1/2,n,j}u_j(\vec{r})\right) \qquad (4.21)$$
$$=\int_{A_{cell}} dA v_i\left(\sum_{j=1}^{J}S_j u_j(\vec{r})\right).$$

This equation can easily be derived using the same process that we used previously to derive the RZ equation. We will focus upon Galerkin FEMs, which means $v_i = u_i$ in Eqs. (4.20) and (4.21).

Definition of PWL basis functions

The PWL basis functions, developed by Stone and Adams for application to XY transport solutions on arbitrary polygonal grids, are designed to linearly interpolate

functions on arbitrary polygons [3]. In order to build these functions, we divide each polygonal cell into subcells called sides. A side is a triangle defined by choosing a center point in the cell and connecting the center point to two adjacent vertices. Figure 4.1 shows an example of a side in a hexagonal cell.

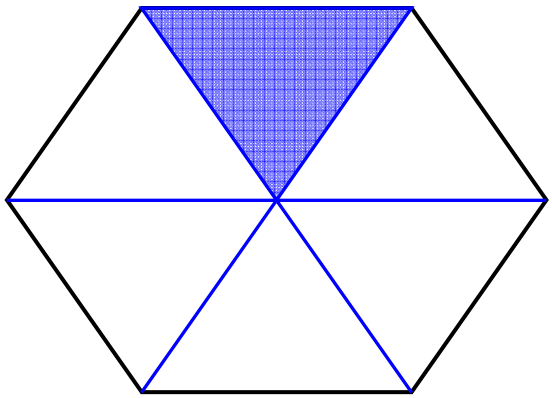

Figure 4.1: The shaded triangle in the hexagon represents a side in a two-dimensional cell

The general mathematical form of the PWL basis functions in two dimensions is

$$u_i(r,z) = t_i(r,z) + \beta_i t_c(r,z), \tag{4.22}$$

where the t functions are standard linear functions defined triangle by triangle. For example, t_i equals one at the i-th vertex and decreases linearly to zero at the cell center and each cell vertex of each side that touches point i. t_c is one at the cell center and decrease linearly to zero at each vertex in the cell. The β_i value is a weighting parameter for the cell center points, and is defined such that

$$\vec{r}_c \equiv \text{ cell midpoint } = \sum_{i=1}^{N} \beta_i \vec{r}_i \,. \tag{4.23}$$

All calculations presented in this dissertation use a value of

$$\beta_i = \frac{1}{N}, \tag{4.24}$$

where N is the number of sides in the cell. To see how these basis functions are constructed we will build one over a pentagonal cell. First, we know that $t_i(r,z)$ has components in two triangular sides inside any cell shape, which is shown in Figure 4.2 .

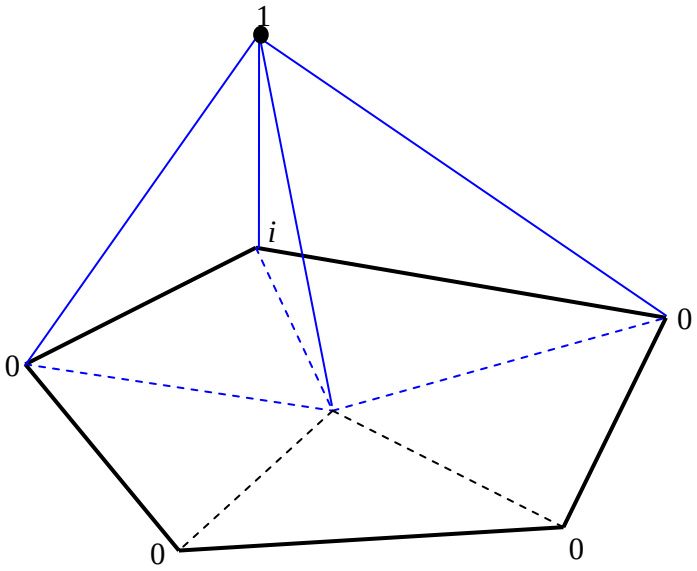

Figure 4.2: The $t_i(r,z)$ components of the basis function u_i, which are defined at support point *i*. The value of this basis function is equal to 1 at its support point and zero at all other vertices

We also know that $t_c(r,z)$ has a component in every side in the cell and is shown in Figure 4.3.

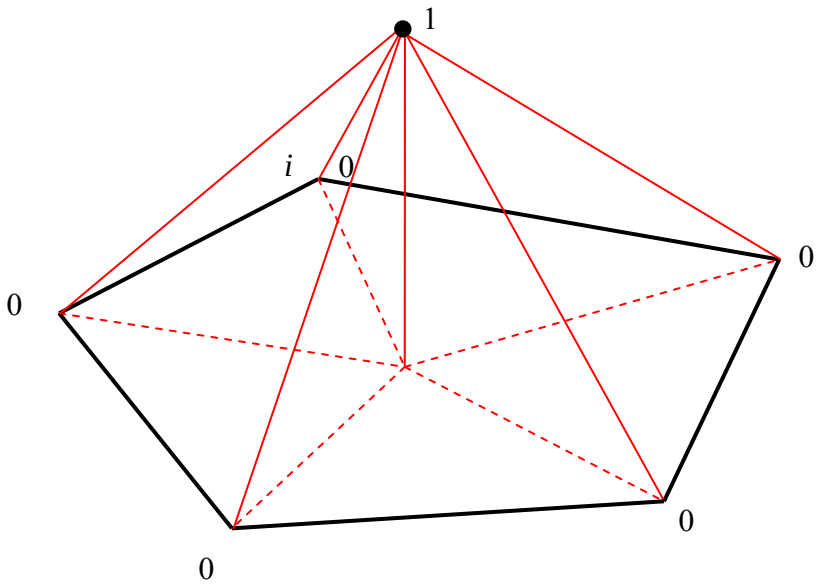

Figure 4.3: The $t_c(r,z)$ components for all basis functions, *u*, defined in the cell. The value of the top of the "tent" is equal to 1, but is multiplied by β_j for each basis to create each basis function

When all components are linearly combined, the result is a PWL basis function, shown in Figure 4.4.

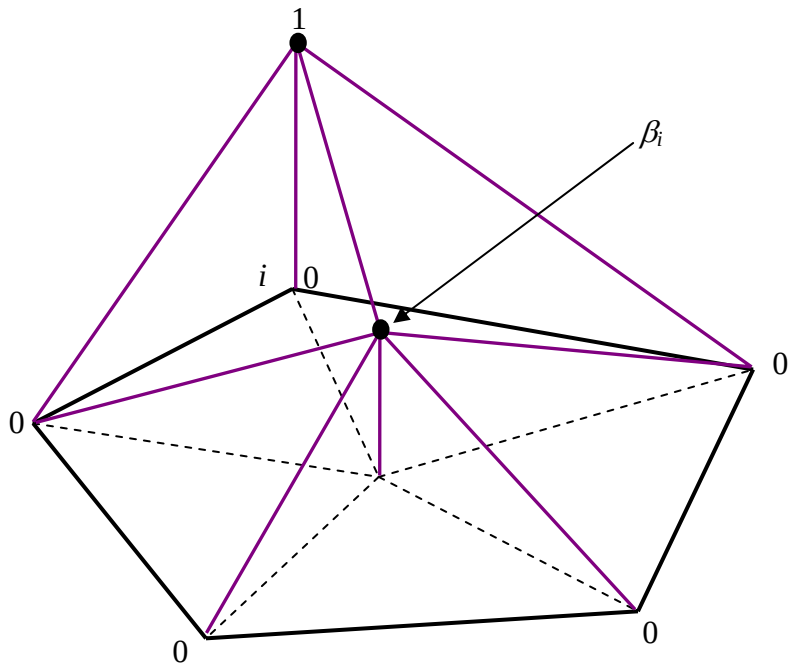

Figure 4.4: A PWL basis function for support point *i* on an arbitrary polygon

One interesting result of the construction of the basis functions is that the PWL basis functions collapse to LD basis functions (or standard linear functions) on triangular cells. As a result, the PWLD method should produce exactly the same solution as LD for any problem on grids composed of triangles.

Because the PWL basis functions are built on triangular side subcell volumes, in order to apply them to the general DFEM formulation in Eqs. (4.20) and (4.21) we divide the integrals over the area of a cell into a sum of integrals over the sides of the cell, resulting in:

$$\sum_{\substack{e= \\ edges \in cell}} \int_e ds u_i \vec{n}_e \bullet \vec{\Omega}_{m,n} r \left\{ \left(\sum_{j=1}^{J} \tilde{\psi}_{m,n,j} u_j(\vec{r}) \right) - \left(\sum_{j=1}^{J} \psi_{m,n,j} u_j(\vec{r}) \right) \right\}$$

$$+ \sum_{\substack{s= \\ side \in cell}} \int_s dA u_i \vec{\Omega}_{m,n} \bullet \vec{\nabla} \left[r \left(\sum_{j=1}^{J} \psi_{m,n,j} u_j(\vec{r}) \right) \right]$$

$$+ \sum_{\substack{s= \\ side \in cell}} \int_s dA u_i \left[\frac{\frac{1}{\tau_{m,n}} \alpha^n_{m+1/2}}{w_{m,n}} \left(\sum_{j=1}^{J} \psi_{m,n,j} u_j(\vec{r}) \right) + \sigma r \left(\sum_{j=1}^{J} \psi_{m,n,j} u_j(\vec{r}) \right) \right] \tag{4.25}$$

$$= \sum_{\substack{s= \\ side \in cell}} \int_s dA u_i \left\{ r \left(\sum_{j=1}^{J} S_j u_j(\vec{r}) \right) + \frac{\left(\frac{1-\tau_{m,n}}{\tau_{m,n}} \alpha^n_{m+1/2} + \alpha^n_{m-1/2} \right)}{w_{m,n}} \sum_{j=1}^{J} \psi_{m-1/2,n,j} u_j(\vec{r}) \right\}$$

and

$$\sum_{\substack{e= \\ edges \in cell}} \int_e ds u_i \vec{n}_e \bullet \vec{\Omega}_{1/2,n} \left\{ \left(\sum_{j=1}^{J} \tilde{\psi}_{1/2,n,j} u_j(\vec{r}) \right) - \left(\sum_{j=1}^{J} \psi_{1/2,n,j} u_j(\vec{r}) \right) \right\}$$

$$+ \sum_{\substack{s= \\ sides \in cell}} \int_s dA u_i \vec{\Omega}_{1/2,n} \bullet \vec{\nabla} \left[\left(\sum_{j=1}^{J} \psi_{1/2,n,j} u_j(\vec{r}) \right) \right] + \sum_{sides \in cell} \int_{side} dA u_i \sigma \left(\sum_{j=1}^{J} \psi_{1/2,n,j} u_j(\vec{r}) \right) \tag{4.26}$$

$$= \sum_{\substack{s= \\ sides \in cell}} \int_s dA u_i \left(\sum_{j=1}^{J} S_j u_j(\vec{r}) \right).$$

We now note that each integral over a side is an integral of standard linear functions on that triangular side.

Solving integrals on mapped triangles

We now solve the integrals for the RZ case, given in Eq. (4.25). First, we define a reference triangle in RZ, which we show in Figure 4.5. We solve the integrals of the linear basis functions on arbitrary triangles by mapping to this reference triangle [22].

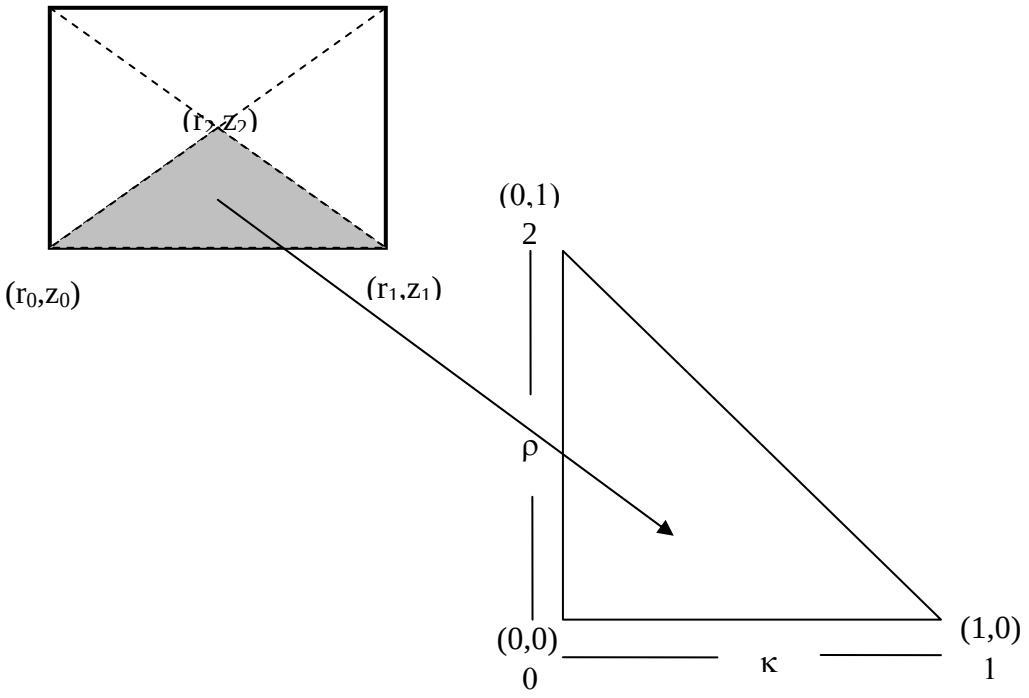

Figure 4.5: Reference triangle for linear functions in RZ.

The standard linear basis functions for this reference triangle are

$$\begin{aligned} u_0 &= 1-\kappa-\rho, \\ u_1 &= \kappa, \\ u_2 &= \rho. \end{aligned} \tag{4.27}$$

We can map the necessary integrals of the basis functions, from integrals over the reference triangle to integrals over the real triangle.

$$\int_0^{\Delta z} dz \int_{r_{\min}(z)}^{r_{\max}(z)} dr\ f(r,z) = \int_0^1 d\rho \int_0^{(1-\rho)} d\kappa\ f(\kappa,\rho)\det(J), \tag{4.28}$$

where we have assumed the method is Galerkin and we define

$$J = \begin{pmatrix} \dfrac{\partial r}{\partial \kappa} & \dfrac{\partial r}{\partial \rho} \\ \dfrac{\partial z}{\partial \kappa} & \dfrac{\partial z}{\partial \rho} \end{pmatrix} \tag{4.29}$$

and

$$\begin{aligned} r(\kappa,\rho) &= r_0 + (r_1 - r_0)\kappa + (r_2 - r_0)\rho, \\ z(\kappa,\rho) &= z_0 + (z_2 - z_0)\rho + (z_1 - z_0)\kappa, \\ \det(J) &= \begin{vmatrix} (r_1 - r_0) & (r_2 - r_0) \\ (z_1 - z_0) & (z_2 - z_0) \end{vmatrix} = 2A_{side}. \end{aligned} \tag{4.30}$$

These relationships imply that

$$drdz = \det|J|\, d\kappa d\rho. \tag{4.31}$$

We also note that

$$\begin{aligned} \frac{\partial r(\kappa,\rho)}{\partial \kappa} &= (r_1 - r_0), \\ \frac{\partial r(\kappa,\rho)}{\partial \rho} &= (r_2 - r_0), \\ \frac{\partial z(\kappa,\rho)}{\partial \kappa} &= (z_1 - z_0), \\ \frac{\partial z(\kappa,\rho)}{\partial \rho} &= (z_2 - z_0). \end{aligned} \tag{4.32}$$

The within cell gradient term for the real triangle can be simplified by using the chain rule on the gradient before we integrate.

$$\begin{aligned}&\int_0^{\Delta z}dz\int_{r_{\min}(z)}^{r_{\max}(z)}dr\,u_i(r,z)\vec{\nabla}\left(ru_j(r,z)\right)\\&=\int_0^{\Delta z}dz\int_{r_{\min}(z)}^{r_{\max}(z)}dr\,u_i(r,z)u_j(r,z)\vec{\nabla}(r)+\int_0^{\Delta z}dz\int_{r_{\min}(z)}^{r_{\max}(z)}dr\,ru_i(r,z)\vec{\nabla}\left(u_j(r,z)\right).\end{aligned}\tag{4.33}$$

Note that the gradient of r is the radial unit vector, and that the gradient of u_j is a constant because the basis functions are linear functions. These simplifications result in

$$\begin{aligned}\int_0^{\Delta z}dz\int_{r_{\min}(z)}^{r_{\max}(z)}dr\,u_i(r,z)\vec{\nabla}\left(ru_j(r,z)\right)&=\left(\int_0^{\Delta z}dz\int_{r_{\min}(z)}^{r_{\max}(z)}dr\,u_i(r,z)u_j(r,z)\right)\hat{i}\\&+\vec{\nabla}\left(u_j(r,z)\right)\int_0^{\Delta z}dz\int_{r_{\min}(z)}^{r_{\max}(z)}dr\,ru_i(r,z).\end{aligned}\tag{4.34}$$

The first matrix of integrals on the right hand side is

$$\int_0^{\Delta y}dy\int_{x_{\min}(y)}^{x_{\max}(y)}dx u_i(x,y)u_j(x,y)=\int_0^1 d\rho\int_0^{(1-\rho)}d\kappa u_i(\kappa,\rho)u_j(\kappa,\rho)\det(J).\tag{4.35}$$

In matrix form, this set of integrals becomes

$$\frac{A_{side}}{12}\begin{pmatrix}2&1&1\\1&2&1\\1&1&2\end{pmatrix}.\tag{4.36}$$

It is interesting to note that the integrals of the angular derivative term have the same form as Eq. (4.35). Therefore, the angular derivative term's side matrix is determined by Eq. (4.36).

We substitute this matrix into Eq. (4.34) resulting in

$$\int_0^{\Delta z}\int_{r_{\min}(z)}^{r_{\max}(z)} dz\, dr\, u_i(r,z)\vec{\nabla}\left(ru_j(r,z)\right)=\frac{1}{12}\begin{pmatrix}2&1&1\\1&2&1\\1&1&2\end{pmatrix}A_{side}\hat{i}$$
$$+\vec{\nabla}\left(u_j(r,z)\right)\int_0^{\Delta z}\int_{r_{\min}(z)}^{r_{\max}(z)} dz\, dr\, ru_i(r,z). \tag{4.37}$$

The second matrix of integrals on the right hand side can be solved using the same mapping process we used to solve the mass matrix integrals. We first note that

$$\vec{\nabla}\left(u_j(r,z)\right)=-\frac{1}{\det(J)}\vec{l}_{opp,j}=-\frac{1}{2A_{side}}\vec{l}_{opp,j}\,, \tag{4.38}$$

where $\vec{l}_{opp,j}$ is the outward unit normal of the edge opposite vertex *j*. We then expand *r* in the integral in terms of the basis functions.

$$\int_0^{\Delta z}\int_{r_{\min}(z)}^{r_{\max}(z)} dz\, dr\, ru_i(r,z)=\int_0^{\Delta z}\int_{r_{\min}(z)}^{r_{\max}(z)} dz\, dr\, u_i(r,z)\sum_{k=0}^{2} r_k u_k(r,z)$$
$$=\int_0^{\Delta z}\int_{r_{\min}(z)}^{r_{\max}(z)} dz\, dr\, u_i(r,z)\left(r_0u_0(r,z)+r_1u_1(r,z)+r_2u_2(r,z)\right). \tag{4.39}$$

Putting the integral and gradient components of this matrix together results in

$$\vec{\nabla}\left(u_j(r,z)\right)\int_0^{\Delta z}\int_{r_{\min}(z)}^{r_{\max}(z)} dz\, dr\, ru_i(r,z)=$$
$$-\left[\int_0^{\Delta z}\int_{r_{\min}(z)}^{r_{\max}(z)} dz\, dr\, u_i(r,z)\left(r_0u_0(r,z)+r_1u_1(r,z)+r_2u_2(r,z)\right)\right]\frac{1}{2A_{side}}\vec{l}_{opp,j}. \tag{4.40}$$

These integrals have already been evaluated in Eq. (4.36). For example, the *0th* row of the matrix will be

$$-\left[\frac{1}{12}A_{side}\left(2r_0+r_1+r_2\right)\right]\frac{1}{2A_{side}}\vec{l}_{opp,j}=-\left[\frac{1}{24}\left(2r_0+r_1+r_2\right)\right]\vec{l}_{opp,j}\,, \tag{4.41}$$

where $\vec{l}_{opp,j}$ has the same definition as before. We can now construct the gradient matrix in its *i* and *j* components.

$$\begin{aligned}
&-\frac{1}{24}\begin{pmatrix} (2r_0+r_1+r_2)l_{opp,0,r} & (2r_0+r_1+r_2)l_{opp,1,r} & (2r_0+r_1+r_2)l_{opp,2,r} \\ (r_0+2r_1+r_2)l_{opp,0,r} & (r_0+2r_1+r_2)l_{opp,1,r} & (r_0+2r_1+r_2)l_{opp,2,r} \\ (r_0+r_1+2r_2)l_{opp,0,r} & (r_0+r_1+2r_2)l_{opp,1,r} & (r_0+r_1+2r_2)l_{opp,2,r} \end{pmatrix}\hat{i}, \\
&-\frac{1}{24}\begin{pmatrix} (2r_0+r_1+r_2)l_{opp,0,z} & (2r_0+r_1+r_2)l_{opp,1,z} & (2r_0+r_1+r_2)l_{opp,2,z} \\ (r_0+2r_1+r_2)l_{opp,0,z} & (r_0+2r_1+r_2)l_{opp,1,z} & (r_0+2r_1+r_2)l_{opp,2,z} \\ (r_0+r_1+2r_2)l_{opp,0,z} & (r_0+r_1+2r_2)l_{opp,1,z} & (r_0+r_1+2r_2)l_{opp,2,z} \end{pmatrix}\hat{j}.
\end{aligned} \tag{4.42}$$

As a result

$$\begin{aligned}
&\int_0^{\Delta z} dz \int_{r_{\min}(z)}^{r_{\max}(z)} dr\, b_i(r,z)\vec{\nabla}\left(r b_j(r,z)\right) = \\
&\left\{ \begin{array}{c} \dfrac{A_{side}}{12}\begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} \\ -\dfrac{1}{24}\begin{pmatrix} (2r_0+r_1+r_2)l_{opp,0,r} & (2r_0+r_1+r_2)l_{opp,1,r} & (2r_0+r_1+r_2)l_{opp,2,r} \\ (r_0+2r_1+r_2)l_{opp,0,r} & (r_0+2r_1+r_2)l_{opp,1,r} & (r_0+2r_1+r_2)l_{opp,2,r} \\ (r_0+r_1+2r_2)l_{opp,0,r} & (r_0+r_1+2r_2)l_{opp,1,r} & (r_0+r_1+2r_2)l_{opp,2,r} \end{pmatrix} \end{array} \right\}\hat{i} \\
&-\frac{1}{24}\begin{pmatrix} (2r_0+r_1+r_2)l_{opp,0,z} & (2r_0+r_1+r_2)l_{opp,1,z} & (2r_0+r_1+r_2)l_{opp,2,z} \\ (r_0+2r_1+r_2)l_{opp,0,z} & (r_0+2r_1+r_2)l_{opp,1,z} & (r_0+2r_1+r_2)l_{opp,2,z} \\ (r_0+r_1+2r_2)l_{opp,0,z} & (r_0+r_1+2r_2)l_{opp,1,z} & (r_0+r_1+2r_2)l_{opp,2,z} \end{pmatrix}\hat{j}.
\end{aligned} \tag{4.43}$$

We also define a parameter for each side edge internal to the cell that limits to one as the *r* coordinates of the cell become close relative to their distance from the axis. When this parameter goes to one, the method limits to the XY PWLD discretization. Furthermore, as this parameter goes to one, the resultant discretization can be used as the discretization for the starting direction equation, Eq. (4.26).

$$\gamma_0 = \frac{r_0}{r_2}$$
$$\gamma_1 = \frac{r_1}{r_2} \tag{4.44}$$

When Eq. (4.44) is substituted into Eq. (4.43), it simplifies to

$$\int_0^{\Delta z} dz \int_{r_{\min}(z)}^{r_{\max}(z)} dr\, u_i(r,z)\vec{\nabla}\left(r u_j(r,z)\right) =$$
$$\left\{ \begin{array}{c} \dfrac{A_{tri}}{12}\begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} \\ -\dfrac{r_2}{24}\begin{pmatrix} (2\gamma_0+\gamma_1+1)l_{opp,0,r} & (2\gamma_0+\gamma_1+1)l_{opp,1,r} & (2\gamma_0+\gamma_1+1)l_{opp,2,r} \\ (\gamma_0+2\gamma_1+1)l_{opp,0,r} & (\gamma_0+2\gamma_1+1)l_{opp,1,r} & (\gamma_0+2\gamma_1+1)l_{opp,2,r} \\ (\gamma_0+\gamma_1+2)l_{opp,0,r} & (\gamma_0+\gamma_1+2)l_{opp,1,r} & (\gamma_0+\gamma_1+2)l_{opp,2,r} \end{pmatrix} \end{array} \right\}\hat{i} \tag{4.45}$$
$$-\frac{r_2}{24}\begin{pmatrix} (2\gamma_0+\gamma_1+1)l_{opp,0,z} & (2\gamma_0+\gamma_1+1)l_{opp,1,z} & (2\gamma_0+\gamma_1+1)l_{opp,2,z} \\ (\gamma_0+2\gamma_1+1)l_{opp,0,z} & (\gamma_0+2\gamma_1+1)l_{opp,1,z} & (\gamma_0+2\gamma_1+1)l_{opp,2,z} \\ (\gamma_0+\gamma_1+2)l_{opp,0,z} & (\gamma_0+\gamma_1+2)l_{opp,1,z} & (\gamma_0+\gamma_1+2)l_{opp,2,z} \end{pmatrix}\hat{j}.$$

The mass matrix is considerably easier to derive. We begin by writing the integrals as

$$\int_0^{\Delta z} dz \int_{r_{\min}(z)}^{r_{\max}(z)} dr\, r u_i(r,z) u_j(r,z) = \int_0^1 d\rho \int_0^{(1-\rho)} d\kappa\, r(\kappa,\rho) u_i(\kappa,\rho) u_j(\kappa,\rho) \det(J). \tag{4.46}$$

Then we expand *r* in terms of the basis functions, and solve the integrals on the reference triangle, resulting in

$$\int_0^1 d\rho \int_0^{(1-\rho)} d\kappa \left[r_0 + (r_1 - r_0)\kappa + (r_2 - r_0)\rho \right] u_i(\kappa,\rho) u_j(\kappa,\rho) \det(J)$$

$$= 2A_{side} \left\{ \begin{array}{l} r_0 \int_0^1 d\rho \int_0^{(1-\rho)} d\kappa \, u_i(\kappa,\rho) u_j(\kappa,\rho) \\ + (r_1 - r_0) \int_0^1 d\rho \int_0^{(1-\rho)} d\kappa \, \kappa u_i(\kappa,\rho) u_j(\kappa,\rho) \\ + (r_2 - r_0) \int_0^1 d\rho \int_0^{(1-\rho)} d\kappa \, \rho u_i(\kappa,\rho) u_j(\kappa,\rho) \end{array} \right\} \tag{4.47}$$

$$= A_{side} r_2 \left\{ \frac{\gamma_0}{12} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} + \frac{(\gamma_1 - \gamma_0)}{60} \begin{pmatrix} 2 & 2 & 1 \\ 2 & 6 & 2 \\ 1 & 2 & 2 \end{pmatrix} + \frac{(1-\gamma_0)}{60} \begin{pmatrix} 2 & 1 & 2 \\ 1 & 2 & 2 \\ 2 & 2 & 6 \end{pmatrix} \right\}.$$

The integrals over the surface of the cell are defined only for one edge in the triangle because each side has only one edge that corresponds to a cell edge. In matrix form, these integrals are

$$\oint_{\partial A} ds \left(\vec{n} \bullet \vec{\Omega}_{m,n} \right) r u_i(r,z) u_j(r,z)$$

$$= \oint_{\partial A} ds \left(\vec{n} \bullet \vec{\Omega}_{m,n} \right) \left(\sum_{k=0}^{N_{edge}} r_k u_k(r,z) \right) u_i(r,z) u_j(r,z)$$

$$= \oint_{\partial A} ds \left(\vec{n} \bullet \vec{\Omega}_{m,n} \right) \left(r_i u_i(r,z) + r_j u_j(r,z) \right) u_i(r,z) u_j(r,z)$$

$$= \vec{\Omega}_{m,n} \bullet \vec{l}_{cell\ edge} \left\{ \frac{r_0}{12} \begin{pmatrix} 3 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} + \frac{r_1}{12} \begin{pmatrix} 1 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 0 \end{pmatrix} \right\}$$

$$= \vec{\Omega}_{m,n} \bullet \vec{l}_{cell\ edge} r_2 \left\{ \frac{\gamma_0}{12} \begin{pmatrix} 3 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} + \frac{\gamma_1}{12} \begin{pmatrix} 1 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 0 \end{pmatrix} \right\}. \tag{4.48}$$

The cell edge corresponds to the side edge between vertex 0 and vertex 1 in Figure 4.5. The Galerkin LDFEM discretization on the triangular sides in RZ is given by

$$
\begin{bmatrix}
\mu_{m,n}\dfrac{l_{edge,r}r_2}{12}\left\{\gamma_0\begin{pmatrix}3&1&0\\1&1&0\\0&0&0\end{pmatrix}+\gamma_1\begin{pmatrix}1&1&0\\1&3&0\\0&0&0\end{pmatrix}\right\} \\
+\eta_{m,n}\dfrac{l_{edge,z}r_2}{12}\left\{\gamma_0\begin{pmatrix}3&1&0\\1&1&0\\0&0&0\end{pmatrix}+\gamma_1\begin{pmatrix}1&1&0\\1&3&0\\0&0&0\end{pmatrix}\right\}
\end{bmatrix}
\begin{pmatrix}\tilde{\psi}_{m,n,0}\\ \tilde{\psi}_{m,n,1}\\ \tilde{\psi}_{m,n,2}\end{pmatrix}
$$

$$
-\begin{bmatrix}
\mu_{m,n}\dfrac{l_{edge,r}r_2}{12}\left\{\gamma_0\begin{pmatrix}3&1&0\\1&1&0\\0&0&0\end{pmatrix}+\gamma_1\begin{pmatrix}1&1&0\\1&3&0\\0&0&0\end{pmatrix}\right\} \\
+\eta_{m,n}\dfrac{l_{edge,z}r_2}{12}\left\{\gamma_0\begin{pmatrix}3&1&0\\1&1&0\\0&0&0\end{pmatrix}+\gamma_1\begin{pmatrix}1&1&0\\1&3&0\\0&0&0\end{pmatrix}\right\}
\end{bmatrix}
\begin{pmatrix}\psi_{m,n,0}\\ \psi_{m,n,1}\\ \psi_{m,n,2}\end{pmatrix}
$$

$$
+\mu_{m,n}\left\{\frac{A_{side}}{12}\begin{pmatrix}2&1&1\\1&2&1\\1&1&2\end{pmatrix}-\frac{r_2}{24}\begin{pmatrix}(2\gamma_0+\gamma_1+1)l_{opp,0,r}&(2\gamma_0+\gamma_1+1)l_{opp,1,r}&(2\gamma_0+\gamma_1+1)l_{opp,2,r}\\(\gamma_0+2\gamma_1+1)l_{opp,0,r}&(\gamma_0+2\gamma_1+1)l_{opp,1,r}&(\gamma_0+2\gamma_1+1)l_{opp,2,r}\\(\gamma_0+\gamma_1+2)l_{opp,0,r}&(\gamma_0+\gamma_1+2)l_{opp,1,r}&(\gamma_0+\gamma_1+2)l_{opp,2,r}\end{pmatrix}\right\}\begin{pmatrix}\psi_{m,n,0}\\ \psi_{m,n,1}\\ \psi_{m,n,2}\end{pmatrix}
$$

$$
-\eta_{m,n}\frac{r_2}{24}\begin{pmatrix}(2\gamma_0+\gamma_1+1)l_{opp,0,z}&(2\gamma_0+\gamma_1+1)l_{opp,1,z}&(2\gamma_0+\gamma_1+1)l_{opp,2,z}\\(\gamma_0+2\gamma_1+1)l_{opp,0,z}&(\gamma_0+2\gamma_1+1)l_{opp,1,z}&(\gamma_0+2\gamma_1+1)l_{opp,2,z}\\(\gamma_0+\gamma_1+2)l_{opp,0,z}&(\gamma_0+\gamma_1+2)l_{opp,1,z}&(\gamma_0+\gamma_1+2)l_{opp,2,z}\end{pmatrix}\begin{pmatrix}\psi_{m,n,0}\\ \psi_{m,n,1}\\ \psi_{m,n,2}\end{pmatrix}
$$

$$
+\frac{\dfrac{1}{\tau_{m,n}}\alpha^n_{m+\frac{1}{2}}}{w_{m,n}}\frac{A_{side}}{12}\begin{pmatrix}2&1&1\\1&2&1\\1&1&2\end{pmatrix}\begin{pmatrix}\psi_{m,n,0}\\ \psi_{m,n,1}\\ \psi_{m,n,2}\end{pmatrix}
$$

$$
+\sigma_{side}A_{side}r_2\left\{\frac{\gamma_0}{12}\begin{bmatrix}2&1&1\\1&2&1\\1&1&2\end{bmatrix}+\frac{(\gamma_1-\gamma_0)}{60}\begin{pmatrix}2&2&1\\2&6&2\\1&2&2\end{pmatrix}+\frac{(1-\gamma_0)}{60}\begin{pmatrix}2&1&2\\1&2&2\\2&2&6\end{pmatrix}\right\}\begin{pmatrix}\psi_{m,n,0}\\ \psi_{m,n,1}\\ \psi_{m,n,2}\end{pmatrix}
$$

$$
=A_{side}r_2\left\{\frac{\gamma_0}{12}\begin{bmatrix}2&1&1\\1&2&1\\1&1&2\end{bmatrix}+\frac{(\gamma_1-\gamma_0)}{60}\begin{pmatrix}2&2&1\\2&6&2\\1&2&2\end{pmatrix}+\frac{(1-\gamma_0)}{60}\begin{pmatrix}2&1&2\\1&2&2\\2&2&6\end{pmatrix}\right\}\begin{pmatrix}S_0\\ S_1\\ S_2\end{pmatrix}
$$

$$
\frac{\left(\dfrac{1-\tau_{m,n}}{\tau_{m,n}}\alpha^n_{m+\frac{1}{2}}+\alpha^n_{m-\frac{1}{2}}\right)}{w_{m,n}}\frac{A_{side}}{12}\begin{pmatrix}2&1&1\\1&2&1\\1&1&2\end{pmatrix}\begin{pmatrix}\psi_{m-\frac{1}{2},n,0}\\ \psi_{m-\frac{1}{2},n,1}\\ \psi_{m-\frac{1}{2},n,2}\end{pmatrix}. \tag{4.49}
$$

The only difference between the LD definitions on the side subcell volume and LD on a triangular mesh element is that the surface matrices, given in the first two lines of Eq. (4.49), are not full. Also, again note that $\tilde{\psi}_{m,n}$ is determined by an upwinding condition and will contain within-cell unknowns or incoming intensities depending on that condition.

Using Eq. (4.49), we can easily write the starting direction spatial discretization by setting γ_0 and γ_1 equal to one, and eliminating the angular derivative terms because the particle is aimed along the axis. For the starting direction, the PWLD discretization for one side in a cell is

$$\begin{aligned}
&\mu_{\frac{1}{2},n}\frac{l_{\text{cell edge},r}}{6}\begin{pmatrix}2&1&0\\1&2&0\\0&0&0\end{pmatrix}\begin{pmatrix}\tilde{\psi}_{\frac{1}{2},n,0}\\ \tilde{\psi}_{\frac{1}{2},n,1}\\ \tilde{\psi}_{\frac{1}{2},n,2}\end{pmatrix}+\xi_{\frac{1}{2},n}\frac{l_{\text{cell edge},z}}{6}\begin{pmatrix}2&1&0\\1&2&0\\0&0&0\end{pmatrix}\begin{pmatrix}\tilde{\psi}_{\frac{1}{2},n,0}\\ \tilde{\psi}_{\frac{1}{2},n,1}\\ \tilde{\psi}_{\frac{1}{2},n,2}\end{pmatrix}\\
&-\mu_{\frac{1}{2},n}\frac{l_{\text{cell edge},r}}{6}\begin{pmatrix}2&1&0\\1&2&0\\0&0&0\end{pmatrix}\begin{pmatrix}\psi_{\frac{1}{2},n,0}\\ \psi_{\frac{1}{2},n,1}\\ \psi_{\frac{1}{2},n,2}\end{pmatrix}-\xi_{\frac{1}{2},n}\frac{l_{\text{cell edge},z}}{6}\begin{pmatrix}2&1&0\\1&2&0\\0&0&0\end{pmatrix}\begin{pmatrix}\psi_{\frac{1}{2},n,0}\\ \psi_{\frac{1}{2},n,1}\\ \psi_{\frac{1}{2},n,2}\end{pmatrix}\\
&-\mu_{\frac{1}{2},n}\frac{1}{6}\begin{pmatrix}l_{opp,0,r}&l_{opp,1,r}&l_{opp,2,r}\\ l_{opp,0,r}&l_{opp,1,r}&l_{opp,2,r}\\ l_{opp,0,r}&l_{opp,1,r}&l_{opp,2,r}\end{pmatrix}\begin{pmatrix}\psi_{\frac{1}{2},n,0}\\ \psi_{\frac{1}{2},n,1}\\ \psi_{\frac{1}{2},n,2}\end{pmatrix}-\eta_{\frac{1}{2},n}\frac{1}{6}\begin{pmatrix}l_{opp,0,z}&l_{opp,1,z}&l_{opp,2,z}\\ l_{opp,0,z}&l_{opp,1,z}&l_{opp,2,z}\\ l_{opp,0,z}&l_{opp,1,z}&l_{opp,2,z}\end{pmatrix}\begin{pmatrix}\psi_{\frac{1}{2},n,0}\\ \psi_{\frac{1}{2},n,1}\\ \psi_{\frac{1}{2},n,2}\end{pmatrix}\\
&+\sigma_{side}\frac{A_{side}}{12}\begin{pmatrix}2&1&1\\1&2&1\\1&1&2\end{pmatrix}\begin{pmatrix}\psi_{\frac{1}{2},n,0}\\ \psi_{\frac{1}{2},n,1}\\ \psi_{\frac{1}{2},n,2}\end{pmatrix}=\frac{A_{side}}{12}\begin{pmatrix}2&1&1\\1&2&1\\1&1&2\end{pmatrix}\begin{pmatrix}S_0\\ S_1\\ S_2\end{pmatrix}.
\end{aligned}\tag{4.50}$$

Implementation of PWLD in Capsaicin

SERRANO is a software package being developed at Los Alamos National Laboratory to simulate photon transport in moving materials. It supports transport solutions on unstructured XY and RZ meshes of triangles and quadrilaterals. The SERRANO software package is designed to be a component of larger radiation hydrodynamics codes. Capsaicin is a deterministic transport solver inside of SERRANO. Before the present effort, Capsaicin employed Linear Discontinuous and Bilinear Discontinuous Finite Element discretizations of the transport equation on triangular and quadrilateral cells, respectively. We added the PWLD method in XY and RZ to the Capsaicin code, implemented for general polygons [23].

Many features of the LD discretization were used to implement PWLD into Capsaicin. Because the PWLD matrices are really constructed by summing integrals of standard linear functions over the triangular side subcell volumes, we reuse the general form of the LD matrices, which are those same integrals of standard linear functions, mapped onto the side over which we are integrating. The zeroth row of the LD matrix is associated with the zeroth vertex in the side, the first row of the LD matrix is associated with the first vertex in the side, and the second row of the LD matrix is associated with the center point vertex in the side.

For example, we substitute the definition of the PWL basis functions given in Eq. (4.22) into the general discretized DFEM form of the transport equation given in Eq.(4.25), and assume our method is Galerkin.

$$\sum_{\substack{e=\\ edges\\ \in cell}} \int_e ds\vec{n}_e \bullet \vec{\Omega}_{m,n} r\left[t_i(r,z)+\beta_i t_c(r,z)\right]\left(\sum_{j=1}^{J}\tilde{\psi}_{m,n,j}\left[t_j(r,z)+\beta_j t_c(r,z)\right]\right)$$

$$-\sum_{\substack{e=\\ edges\\ \in cell}} \int_e ds\vec{n}_e \bullet \vec{\Omega}_{m,n} r\left[t_i(r,z)+\beta_i t_c(r,z)\right]\left(\sum_{j=1}^{J}\psi_{m,n,j}\left[t_j(r,z)+\beta_j t_c(r,z)\right]\right)$$

$$+\sum_{\substack{s=\\ side\\ \in cell}} \int_s dA\left[t_i(r,z)+\beta_i t_c(r,z)\right]\vec{\Omega}_{m,n}\bullet\vec{\nabla}\left[r\left(\sum_{j=1}^{J}\psi_{m,n,j}\left[t_j(r,z)+\beta_i t_c(r,z)\right]\right)\right]$$

$$+\frac{\alpha^n_{m+1/2}}{\tau_{m,n}w_{m,n}}\sum_{\substack{s=\\ side\\ \in cell}} \int_s dA\left[t_i(r,z)+\beta_i t_c(r,z)\right]\left(\sum_{j=1}^{J}\psi_{m,n,j}\left[t_j(r,z)+\beta_i t_c(r,z)\right]\right)$$

$$+\sum_{\substack{s=\\ side\\ \in cell}} \sigma\int_s dA\left[t_i(r,z)+\beta_i t_c(r,z)\right] r\left(\sum_{j=1}^{J}\psi_{m,n,j}\left[t_j(r,z)+\beta_i t_c(r,z)\right]\right)$$

$$=\sum_{\substack{s=\\ side\\ \in cell}} \int_s dAr\left[t_i(r,z)+\beta_i t_c(r,z)\right]\left(\sum_{j=1}^{J}S_j\left[t_j(r,z)+\beta_i t_c(r,z)\right]\right)$$

$$+\frac{1-\tau_{m,n}}{w_{m.n}\tau_{m,n}}\alpha^n_{m+1/2}\sum_{\substack{s=\\ side\\ \in cell}} \int_s dAr\left[t_i(r,z)+\beta_i t_c(r,z)\right]\left(\sum_{j=1}^{J}\psi_{m-1/2,n,j}\left[t_j(r,z)+\beta_i t_c(r,z)\right]\right) \quad (4.51)$$

$$+\frac{\alpha^n_{m-1/2}}{w_{m,n}}\sum_{\substack{s=\\ side\\ \in cell}} \int_s dAr\left[t_i(r,z)+\beta_i t_c(r,z)\right]\left(\sum_{j=1}^{J}\psi_{m-1/2,n,j}\left[t_j(r,z)+\beta_i t_c(r,z)\right]\right)$$

Eq. (4.51) suggests an algorithm for building the DFEM matrices by looping over sides. In each side, four distinct integrals must be performed. For the *(i,j)* element in the mass matrix, these integrals are

$$\int_s dA_s r u_i u_j = \int_s dA_s\left[r t_i t_j + r\beta_i t_c t_j + r t_i \beta_j t_c + r\beta_i t_c \beta_j t_c\right]. \quad (4.52)$$

For the *(i,j)* element in the within cell gradient matrix these integrals are

$$\int_s dA_s u_i \vec{\nabla} r u_j = \int_s dA_s \left[t_i \vec{\nabla} r t_j + \beta_i t_c \vec{\nabla} r t_j + t_i \beta_j \vec{\nabla} r t_c + \beta_i t_c \beta_j \vec{\nabla} r t_c \right]. \quad (4.53)$$

For the *(i,j)* element in the surface matrix these integrals are

$$\int_{s_{edge}} dA_s r u_i u_j = \int_{s_{edge}} ds \left[r t_i t_j + r \beta_i t_c t_j + r t_i \beta_j t_c + r \beta_i t_c \beta_j t_c \right]. \quad (4.54)$$

Note that in a given side not all *t* functions are non-zero. Only two values of *i* and *j* have non-zero components in a given side—the two *i* and *j* values that are equal to the vertex identifiers on the side. Also note that every t_c value is non-zero in every side, but these t_c functions are zero on side edges. Also, only two values of *i* and *j* have non-zero components on a given side edge. As a result, all side integrals can be calculated using the 3x3 LD matrices developed for triangles, and the elements of these LD matrices can be summed into the appropriate elements of the PWLD matrices. Below, we show how we build the mass matrix for the PWLD method using the LD matrices for the sides of a polygonal cell.

```
Loop over sides
    cs = cell global vertex 1 in the side;
    cn = cell global vertex 2 in the side;

    //Form 3x3 LD mass matrix on the triangle determined by cs, cn, and the
    //cell center

    PWLD_M[cs][cs] += LD_M[0][0];
    PWLD_M[cs][cn] += LD_M[0][1];
    PWLD_M[cn][cs] += LD_M[1][0];
    PWLD_M[cn][cn] += LD_M[1][1];

    for(jj=0; ii<num_sides; ++ii)
```

```
{
        PWLD_M[ii][cs] += LD_M[2][0]*beta[ii];
        PWLD_M[ii][cn] += LD_M[2][1]*beta[ii];

        PWLD_M[cs][ii] += LD_M[0][2]*beta[ii];
        PWLD_M[cn][ii] += LD_M[1][2]*beta[ii];

        for(jj=0; jj<num_sides; ++jj)
        {
            PWLD_M[ii][jj] += LD_M[2][2]*beta[ii]*beta[jj];
         }
}
```

End Loop over sides

All that is left to do is determine the 3x3 LD matrices on an arbitrary triangle, which we have shown previously. This method of mapping the LD matrices onto the triangular sides of an arbitrary polygonal cell is relatively simple and useful. It shows a way to easily build a PWLD method into a code that already contains an LD discretization, which is what we have done in Capsaicin. Furthermore, it also underscores the close relationship between PWLD and LD [23].

Asymptotic analysis and the thick diffusive limit

As explained previously, we are interested in how spatial discretizations of the radiation transport equation perform in the thick diffusive limit. Because the background material in many problems of interest tends to be optically thick to radiation, we encounter spatial cells that are many mean free paths thick and dominated by interactions in which photon energy is not lost but simply redistributed, which is the definition of a thick diffusive region. Any spatial discretization of the radiation transport equation must limit to a valid discretization of the diffusion equation to ensure that the method will not produce

unphysical results in such thick diffusive cells. The asymptotic analysis we perform follows Adams' general analysis for DFEMs [6] and Palmer's asymptotic analysis for general DFEMs in RZ geometry [21]. We start with Eqs. (4.25) and (4.26) in matrix form, assuming the source is due to isotropic scattering and a fixed source.

$$\sum_{\substack{e= \\ edges \in cell}} \left\{ \mu_{m,n} \underline{\underline{N}}_e^r \begin{bmatrix} \tilde{\psi}_{m,n,1,e} \\ \tilde{\psi}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}_{m,n,J,e} \end{bmatrix} + \xi_{m,n} \underline{\underline{N}}_e^z \begin{bmatrix} \tilde{\psi}_{m,n,1,e} \\ \tilde{\psi}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}_{m,n,J,e} \end{bmatrix} - \mu_{m,n} \underline{\underline{T}}_e^r \vec{\psi}_{m,n} - \xi_{m,n} \underline{\underline{T}}_e^z \vec{\psi}_{m,n} \right\}$$

$$+ \mu_{m,n} \underline{\underline{L}}^r \vec{\psi}_{m,n} + \xi_{m,n} \underline{\underline{L}}^z \vec{\psi}_{m,n} + \frac{\frac{1}{\tau_{m,n}} \alpha^n_{m+1/2}}{w_{m,n}} \underline{\underline{P}} \vec{\psi}_{m,n} + \sigma \underline{\underline{M}} \vec{\psi}_{m,n} \tag{4.55}$$

$$= \frac{1}{4\pi} \underline{\underline{M}} \left\{ \left(\sigma - \sigma_a \right) \vec{\phi} + \vec{S} \right\} + \frac{\left(\frac{1-\tau_{m,n}}{\tau_{m,n}} \alpha^n_{m+1/2} + \alpha^n_{m-1/2} \right)}{w_{m,n}} \underline{\underline{P}} \vec{\psi}_{m-1/2,n},$$

where

$$\underline{\underline{N}}_e^r \to n_{i,j,e}^r = n_{r,e}\int_e dsru_iu_j$$

$$\underline{\underline{N}}_e^z \to n_{i,j,e}^z = n_{z,e}\int_e dsru_iu_j$$

$$\underline{\underline{T}}_e^r \to t_{i,j,e}^r = n_{r,e}\int_e dsru_iu_j$$

$$\underline{\underline{T}}_e^z \to t_{i,j,e}^z = n_{z,e}\int_e dsru_iu_j$$

$$\underline{\underline{L}}^r \to l_{i,j}^r = \sum_{sides\in cell}\int_{side} dAu_i\frac{\partial\left(ru_j\right)}{\partial r}$$

$$\underline{\underline{L}}^z \to l_{i,j}^z = \sum_{sides\in cell}\int_{side} dAu_i\frac{\partial\left(ru_j\right)}{\partial z}$$

$$\underline{\underline{P}} \to p_{i,j} = \sum_{sides\in cell}\int_{side} dAu_iu_j$$

$$\underline{\underline{M}} \to m_{i,j} = \sum_{sides\in cell}\int_{side} dAru_iu_j, \tag{4.56}$$

and

$$\sum_{\substack{e=\\ edges\in cell}}\left\{\mu_{m,n}\underline{\underline{N}}_e^{r,start}\begin{bmatrix}\tilde{\psi}_{1/2,n,1,e}\\ \tilde{\psi}_{1/2,n,2,e}\\ \vdots\\ \tilde{\psi}_{1/2,n,J,e}\end{bmatrix}+\xi_{m,n}\underline{\underline{N}}_e^{z,start}\begin{bmatrix}\tilde{\psi}_{1/2,n,1,e}\\ \tilde{\psi}_{1/2,n,2,e}\\ \vdots\\ \tilde{\psi}_{1/2,n,J,e}\end{bmatrix}\right\}$$
$$-\sum_{\substack{e=\\ edges\in cell}}\left\{\mu_{1/2,n}\underline{\underline{T}}_e^{r,start}\vec{\psi}_{1/2,n}+\xi_{1/2,n}\underline{\underline{T}}_e^{z,start}\vec{\psi}_{1/2,n}\right\} \tag{4.57}$$
$$+\mu_{1/2,n}\underline{\underline{L}}^{r,start}\vec{\psi}_{1/2,n}+\xi_{1/2,n}\underline{\underline{L}}^{z,start}\vec{\psi}_{1/2,n}+\sigma\underline{\underline{M}}^{start}\vec{\psi}_{1/2,n}$$
$$=\frac{1}{4\pi}\underline{\underline{M}}^{start}\left\{\left(\sigma-\sigma_a\right)\vec{\phi}+\vec{S}\right\},$$

where

$$
\begin{aligned}
\underline{\underline{N}}_e^{r,start} &\rightarrow n_{i,j,e}^{r,start} = n_{r,e}\int_e ds u_i u_j \\
\underline{\underline{N}}_e^{z,start} &\rightarrow n_{i,j,e}^{z,start} = n_{z,e}\int_e ds u_i u_j \\
\underline{\underline{T}}_e^{r,start} &\rightarrow t_{i,j,e}^{r,start} = n_{r,e}\int_e ds u_i u_j \\
\underline{\underline{T}}_e^{z,start} &\rightarrow t_{i,j,e}^{z,start} = n_{z,e}\int_e ds u_i u_j \\
\underline{\underline{L}}^{r,start} &\rightarrow l_{i,j}^{r,start} = \sum_{sides\in cell}\int_{side} dA u_i \frac{\partial\left(u_j\right)}{\partial r} \\
\underline{\underline{L}}^{z,start} &\rightarrow l_{i,j}^{z,start} = \sum_{sides\in cell}\int_{side} dA u_i \frac{\partial\left(u_j\right)}{\partial z} \\
\underline{\underline{M}}^{start} &\rightarrow m_{i,j}^{start} = \sum_{sides\in cell}\int_{side} dA r u_i u_j .
\end{aligned}
\tag{4.58}
$$

In the definitions of the single cell matrices, in Eqs. (4.56) and (4.58), the *i,j* subscripts represent the *i,j* matrix elements. The total single cell matrix will be the collection of all *i,j* terms.

We require that the quadrature weights are normalized such that

$$
\sum_{m,n} w_{m,n} = 4\pi \,, \tag{4.59}
$$

and we define the scalar flux and current as

$$
\begin{aligned}
\phi &= \sum_{m,n} w_{m,n}\psi_{m,n}, \\
J_r\hat{e}_r + J_z\hat{e}_z &= \sum_{m,n} w_{m,n}\left(\mu_{m,n}\hat{e}_r + \xi_{m,n}\hat{e}_z\right)\psi_{m,n}.
\end{aligned}
\tag{4.60}
$$

Also, we note that the surface angular flux values, $\tilde{\psi}_{m,n,e}$ are determined by an upwinding condition.

$$\tilde{\psi}_{m,n,e} = \begin{cases} \psi_{m,n,cell} \;\; if \;\; \vec{n}_e \bullet \vec{\Omega}_{m,n} > 0 \\ \psi_{m,n,upwind\ cell,e} \;\; if \;\; \vec{n}_e \bullet \vec{\Omega}_{m,n} < 0 \end{cases} \tag{4.61}$$

The analysis begins by scaling the physical parameters in Eqs. (4.55) and (4.57) such that the problem becomes diffusive as the small parameter, ε, tends to zero.

$$\begin{aligned} \sigma &\to \frac{\sigma}{\varepsilon} \\ \sigma_a &\to \varepsilon \sigma_a \\ S &\to \varepsilon S \end{aligned} \tag{4.62}$$

This scaling is applied to these equations, resulting in

$$\begin{aligned} \sum_{\substack{e= \\ edges \in cell}} \left\{ \mu_{m,n} \underline{\underline{N}}_e^r \begin{bmatrix} \tilde{\psi}_{m,n,1,e} \\ \tilde{\psi}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}_{m,n,J,e} \end{bmatrix} + \xi_{m,n} \underline{\underline{N}}_e^z \begin{bmatrix} \tilde{\psi}_{m,n,1,e} \\ \tilde{\psi}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}_{m,n,J,e} \end{bmatrix} - \mu_{m,n} \underline{\underline{T}}_e^r \vec{\psi}_{m,n} - \xi_{m,n} \underline{\underline{T}}_e^z \vec{\psi}_{m,n} \right\} \\ + \mu_{m,n} \underline{\underline{L}}^r \vec{\psi}_{m,n} + \xi_{m,n} \underline{\underline{L}}^z \vec{\psi}_{m,n} + \frac{\frac{1}{\tau_{m,n}} \alpha^n_{m+1/2}}{w_{m,n}} \underline{\underline{P}} \vec{\psi}_{m,n} + \frac{\sigma}{\varepsilon} \underline{\underline{M}} \vec{\psi}_{m,n} \\ = \frac{1}{4\pi} \underline{\underline{M}} \left\{ \left(\frac{\sigma}{\varepsilon} - \varepsilon \sigma_a \right) \vec{\phi} + \varepsilon \vec{S} \right\} + \frac{\left(\frac{1-\tau_{m,n}}{\tau_{m,n}} \alpha^n_{m+1/2} + \alpha^n_{m-1/2} \right)}{w_{m,n}} \underline{\underline{P}} \vec{\psi}_{m-1/2,n}, \end{aligned} \tag{4.63}$$

and

$$\sum_{\substack{e= \\ edges \in cell}} \left\{ \mu_{1/2,n} \underline{\underline{N}}_e^{r,start} \begin{bmatrix} \tilde{\psi}_{1/2,n,1,e} \\ \tilde{\psi}_{1/2,n,2,e} \\ \vdots \\ \tilde{\psi}_{1/2,n,J,e} \end{bmatrix} + \xi_{1/2,n} \underline{\underline{N}}_e^{z,start} \begin{bmatrix} \tilde{\psi}_{1/2,n,1,e} \\ \tilde{\psi}_{1/2,n,2,e} \\ \vdots \\ \tilde{\psi}_{1/2,n,J,e} \end{bmatrix} \right\}$$

$$- \sum_{\substack{e= \\ edges \in cell}} \left\{ \mu_{1/2,n} \underline{\underline{T}}_e^{r,start} \vec{\psi}_{1/2,n} + \xi_{1/2,n} \underline{\underline{T}}_e^{z,start} \vec{\psi}_{1/2,n} \right\} \tag{4.64}$$

$$+ \mu_{1/2,n} \underline{\underline{L}}^{r,start} \vec{\psi}_{1/2,n} + \xi_{1/2,n} \underline{\underline{L}}^{z,start} \vec{\psi}_{1/2,n} + \frac{\sigma}{\varepsilon} \underline{\underline{M}}^{start} \vec{\psi}_{1/2,n}$$

$$= \frac{1}{4\pi} \underline{\underline{M}}^{start} \left\{ \left(\frac{\sigma}{\varepsilon} - \varepsilon \sigma_a \right) \vec{\phi} + \varepsilon \vec{S} \right\}.$$

We guess that the fluxes can be expanded in a power series in ε:

$$\begin{aligned} \psi &= \psi^{(0)} + \varepsilon \psi^{(1)} + \varepsilon^2 \psi^{(2)} \ldots \\ \phi &= \phi^{(0)} + \varepsilon \phi^{(1)} + \varepsilon^2 \phi^{(2)} \ldots \end{aligned} \tag{4.65}$$

These are substituted into Eqs. (4.63) and (4.64), resulting in

$$\sum_{\substack{e= \\ edges \in cell}} \left\{ \mu_{m,n} \underline{\underline{N}}_e^r \begin{bmatrix} \tilde{\psi}^{(0)}_{m,n,1,e} + \varepsilon \tilde{\psi}^{(1)}_{m,n,1,e} \cdots \\ \tilde{\psi}^{(0)}_{m,n,2,e} + \varepsilon \tilde{\psi}^{(1)}_{m,n,2,e} \cdots \\ \vdots \\ \tilde{\psi}^{(0)}_{m,n,J,e} + \varepsilon \tilde{\psi}^{(1)}_{m,n,J,e} \cdots \end{bmatrix} + \xi_{m,n} \underline{\underline{N}}_e^z \begin{bmatrix} \tilde{\psi}^{(0)}_{m,n,1,e} + \varepsilon \tilde{\psi}^{(1)}_{m,n,1,e} \cdots \\ \tilde{\psi}^{(0)}_{m,n,2,e} + \varepsilon \tilde{\psi}^{(1)}_{m,n,2,e} \cdots \\ \vdots \\ \tilde{\psi}^{(0)}_{m,n,J,e} + \varepsilon \tilde{\psi}^{(1)}_{m,n,J,e} \cdots \end{bmatrix} \right\}$$

$$- \sum_{\substack{e= \\ edges \in cell}} \left\{ \mu_{m,n} \underline{\underline{T}}_e^r \left(\vec{\psi}^{(0)}_{m,n} + \varepsilon \vec{\psi}^{(1)}_{m,n} \cdots \right) + \xi_{m,n} \underline{\underline{T}}_e^z \left(\vec{\psi}^{(0)}_{m,n} + \varepsilon \vec{\psi}^{(1)}_{m,n} \cdots \right) \right\}$$

$$+ \mu_{m,n} \underline{\underline{L}}^r \left(\vec{\psi}^{(0)}_{m,n} + \varepsilon \vec{\psi}^{(1)}_{m,n} \cdots \right) + \xi_{m,n} \underline{\underline{L}}^z \left(\vec{\psi}^{(0)}_{m,n} + \varepsilon \vec{\psi}^{(1)}_{m,n} \cdots \right)$$

$$+ \frac{\dfrac{1}{\tau_{m,n}} \alpha^n_{m+1/2}}{w_{m,n}} \underline{\underline{P}} \left(\vec{\psi}^{(0)}_{m,n} + \varepsilon \vec{\psi}^{(1)}_{m,n} \cdots \right) + \frac{\sigma}{\varepsilon} \underline{\underline{M}} \left(\vec{\psi}^{(0)}_{m,n} + \varepsilon \vec{\psi}^{(1)}_{m,n} \cdots \right)$$

$$= \frac{1}{4\pi} \underline{\underline{M}} \left\{ \left(\frac{\sigma}{\varepsilon} - \varepsilon \sigma_a \right) \left(\vec{\phi}^{(0)} + \varepsilon \vec{\phi}^{(1)} \cdots \right) + \varepsilon \vec{S} \right\}$$

$$+ \left(\frac{1 - \tau_{m,n}}{w_{m,n} \tau_{m,n}} \alpha^n_{m+1/2} + \frac{\alpha^n_{m-1/2}}{w_{m,n}} \right) \underline{\underline{P}} \left(\vec{\psi}^{(0)}_{m-1/2,n} + \varepsilon \vec{\psi}^{(1)}_{m-1/2,n} \cdots \right) \tag{4.66}$$

and

$$\sum_{\substack{e= \\ edges \\ \in cell}} \left\{ \mu_{1/2,n} \underline{\underline{N}}_e^{r,start} \begin{bmatrix} \tilde{\psi}^{(0)}_{1/2,n,1,e} + \varepsilon \tilde{\psi}^{(1)}_{1/2,n,1,e} \cdots \\ \tilde{\psi}^{(0)}_{1/2,n,2,e} + \varepsilon \tilde{\psi}^{(1)}_{1/2,n,2,e} \cdots \\ \vdots \\ \tilde{\psi}^{(0)}_{1/2,n,J,e} + \varepsilon \tilde{\psi}^{(1)}_{1/2,n,J,e} \cdots \end{bmatrix} + \xi_{1/2,n} \underline{\underline{N}}_e^{z,start} \begin{bmatrix} \tilde{\psi}^{(0)}_{1/2,n,1,e} + \varepsilon \tilde{\psi}^{(1)}_{1/2,n,1,e} \cdots \\ \tilde{\psi}^{(0)}_{1/2,n,2,e} + \varepsilon \tilde{\psi}^{(1)}_{1/2,n,2,e} \cdots \\ \vdots \\ \tilde{\psi}^{(0)}_{1/2,n,J,e} + \varepsilon \tilde{\psi}^{(1)}_{1/2,n,J,e} \cdots \end{bmatrix} \right\}$$

$$- \sum_{\substack{e= \\ edges \in cell}} \left\{ \mu_{1/2,n} \underline{\underline{T}}_e^{r,start} \left(\vec{\psi}^{(0)}_{1/2,n} + \varepsilon \vec{\psi}^{(1)}_{1/2,n} \cdots \right) + \xi_{1/2,n} \underline{\underline{T}}_e^{z,start} \left(\vec{\psi}^{(0)}_{1/2,n} + \varepsilon \vec{\psi}^{(1)}_{1/2,n} \cdots \right) \right\}$$

$$+ \mu_{1/2,n} \underline{\underline{L}}^{r,start} \left(\vec{\psi}^{(0)}_{1/2,n} + \varepsilon \vec{\psi}^{(1)}_{1/2,n} \cdots \right) + \xi_{m,n} \underline{\underline{L}}^{z,start} \left(\vec{\psi}^{(0)}_{1/2,n} + \varepsilon \vec{\psi}^{(1)}_{1/2,n} \cdots \right) \tag{4.67}$$

$$+ \frac{\sigma}{\varepsilon} \underline{\underline{M}}^{start} \left(\vec{\psi}^{(0)}_{1/2,n} + \varepsilon \vec{\psi}^{(1)}_{1/2,n} \cdots \right)$$

$$= \frac{1}{4\pi} \underline{\underline{M}}^{start} \left\{ \left(\frac{\sigma}{\varepsilon} - \varepsilon \sigma_a \right) \left(\vec{\phi}^{(0)} + \varepsilon \vec{\phi}^{(1)} \cdots \right) + \varepsilon \vec{S} \right\}.$$

We collect the *O(1/ε)* terms in Eqs. (4.66) and (4.67)

$$\begin{aligned}\sigma \underline{\underline{M}}\,\vec{\psi}^{(0)}_{m,n} &= \frac{1}{4\pi}\underline{\underline{M}}\,\sigma\vec{\phi}^{(0)},\\ \sigma \underline{\underline{M}}^{start}\vec{\psi}^{(0)}_{\frac{1}{2},n} &= \frac{1}{4\pi}\underline{\underline{M}}^{start}\sigma\vec{\phi}^{(0)}.\end{aligned} \tag{4.68}$$

Because the mass matrices are invertible, Eq. (4.68) yields the result that the leading order angular flux is isotropic for both the regular directions and the starting direction.

$$\vec{\psi}^{(0)}_{m,n} = \vec{\psi}^{(0)}_{\frac{1}{2},n} = \frac{1}{4\pi}\vec{\phi}^{(0)}. \tag{4.69}$$

For the remainder of the analysis we ignore the starting direction equation because its quadrature weight is zero.

Next, we collect the *O(1)* terms in Eqs. (4.66).

$$\begin{aligned}&\sum_{\substack{e=\\ edges\in cell}}\left\{\mu_{m,n}\underline{\underline{N}}^{r}_{e}\begin{bmatrix}\tilde{\psi}^{(0)}_{m,n,1,e}\\ \tilde{\psi}^{(0)}_{m,n,2,e}\\ \vdots\\ \tilde{\psi}^{(0)}_{m,n,J,e}\end{bmatrix}+\xi_{m,n}\underline{\underline{N}}^{z}_{e}\begin{bmatrix}\tilde{\psi}^{(0)}_{m,n,1,e}\\ \tilde{\psi}^{(0)}_{m,n,2,e}\\ \vdots\\ \tilde{\psi}^{(0)}_{m,n,J,e}\end{bmatrix}\right\}\\ &\quad-\sum_{\substack{e=\\ edges\in cell}}\left\{\mu_{m,n}\underline{\underline{T}}^{r}_{e}\vec{\psi}^{(0)}_{m,n}+\xi_{m,n}\underline{\underline{T}}^{z}_{e}\vec{\psi}^{(0)}_{m,n}\right\}\\ &+\mu_{m,n}\underline{\underline{L}}^{r}\vec{\psi}^{(0)}_{m,n}+\xi_{m,n}\underline{\underline{L}}^{z}\vec{\psi}^{(0)}_{m,n}+\frac{\frac{1}{\tau_{m,n}}\alpha^{n}_{m+\frac{1}{2}}}{w_{m,n}}\underline{\underline{P}}\vec{\psi}^{(0)}_{m,n}+\sigma\underline{\underline{M}}\,\vec{\psi}^{(1)}_{m,n}\\ &=\frac{1}{4\pi}\sigma\underline{\underline{M}}\,\vec{\phi}^{(1)}+\frac{\left(\frac{1-\tau_{m,n}}{\tau_{m,n}}\alpha^{n}_{m+\frac{1}{2}}+\alpha^{n}_{m-\frac{1}{2}}\right)}{w_{m,n}}\underline{\underline{P}}\vec{\psi}^{(0)}_{m-\frac{1}{2},n},\end{aligned} \tag{4.70}$$

We multiply Eq. (4.70) by $w_{m,n}$ and sum over all *m* and *n*. Eq. (4.70) becomes

$$\sum_{m,n} w_{m,n}\mu_{m,n} \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{N}}_e^r \begin{bmatrix} \tilde{\vec{\psi}}_{m,n,1,e}^{(0)} \\ \tilde{\vec{\psi}}_{m,n,2,e}^{(0)} \\ \vdots \\ \tilde{\vec{\psi}}_{m,n,J,e}^{(0)} \end{bmatrix} + \sum_{m,n} w_{m,n}\xi_{m,n} \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{N}}_e^z \begin{bmatrix} \tilde{\vec{\psi}}_{m,n,1,e}^{(0)} \\ \tilde{\vec{\psi}}_{m,n,2,e}^{(0)} \\ \vdots \\ \tilde{\vec{\psi}}_{m,n,J,e}^{(0)} \end{bmatrix}$$
$$-\sum_{m,n} w_{m,n}\mu_{m,n} \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{T}}_e^r \vec{\psi}_{m,n}^{(0)} - \sum_{m,n} w_{m,n}\xi_{m,n} \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{T}}_e^z \vec{\psi}_{m,n}^{(0)}$$
$$+\sum_{m,n} w_{m,n}\mu_{m,n} \underline{\underline{L}}^r \vec{\psi}_{m,n}^{(0)} + \sum_{m,n} w_{m,n}\xi_{m,n} \underline{\underline{L}}^z \vec{\psi}_{m,n}^{(0)} \tag{4.71}$$
$$+\sum_{m,n} w_{m,n} \left[\frac{\alpha_{m+1/2}^n}{w_{m,n}} \underline{\underline{P}} \vec{\psi}_{m+1/2,n}^{(0)} - \frac{\alpha_{m-1/2}^n}{w_{m,n}} \underline{\underline{P}}_{i,j} \vec{\psi}_{m-1/2,n}^{(0)} \right]$$
$$+\sum_{m,n} w_{m,n}\sigma \underline{\underline{M}} \vec{\psi}_{m,n}^{(1)} - \frac{1}{4\pi} \sum_{m,n} w_{m,n}\sigma \underline{\underline{M}} \vec{\phi}^{(1)} = 0,$$

where we have used the relationship in Eq. (4.11) to simplify Eq. (4.71).

Using the definition of scalar flux, and the fact that the leading order angular flux is isotropic, Eq. (4.71) simplifies to

$$\sum_{m,n} w_{m,n}\mu_{m,n} \sum_{\substack{e= \\ edges\in cell}} \underline{\underline{N}}_e^r \begin{bmatrix} \tilde{\psi}^{(0)}_{m,n,1,e} \\ \tilde{\psi}^{(0)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(0)}_{m,n,J,e} \end{bmatrix} + \sum_{m,n} w_{m,n}\xi_{m,n} \sum_{\substack{e= \\ edges\in cell}} \underline{\underline{N}}_e^z \begin{bmatrix} \tilde{\psi}^{(0)}_{m,n,1,e} \\ \tilde{\psi}^{(0)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(0)}_{m,n,J,e} \end{bmatrix}$$
$$-\frac{1}{4\pi} \sum_{\substack{e= \\ edges\in cell}} \underline{\underline{T}}_e^r \vec{\phi}^{(0)} \sum_{m,n} w_{m,n}\mu_{m,n} - \frac{1}{4\pi} \sum_{\substack{e= \\ edges\in cell}} \underline{\underline{T}}_e^z \vec{\phi}^{(0)} \sum_{m,n} w_{m,n}\xi_{m,n}$$
$$+\frac{1}{4\pi} \underline{\underline{L}}^r \vec{\phi}^{(0)} \sum_{m,n} w_{m,n}\mu_{m,n} + \frac{1}{4\pi} \underline{\underline{L}}^z \vec{\phi}^{(0)} \sum_{m,n} w_{m,n}\xi_{m,n} \qquad (4.72)$$
$$+\frac{1}{4\pi} \underline{\underline{P}} \vec{\phi}^{(0)} \sum_{m,n} w_{m,n} \left[\frac{\alpha^n_{m+1/2}}{w_{m,n}} - \frac{\alpha^n_{m-1/2}}{w_{m,n}} \right]$$
$$+\sigma \underline{\underline{M}} \vec{\phi}^{(1)} - \frac{1}{4\pi} \sigma \underline{\underline{M}} \vec{\phi}^{(1)} \sum_{m,n} w_{m,n} = 0.$$

Terms 3-6 are all equal to zero when the quadrature set correctly integrates linear functions of the cosines, which it does if it is symmetric. Term 7 equals zero due to the symmetry of the quadrature set and the definition of α given in Eq.(4.6). Finally, terms 8 and 9 cancel because the sum of the quadrature weights has been defined to be 4π. These simplifications result in

$$\sum_{m,n} w_{m,n}\mu_{m,n} \sum_{\substack{e= \\ edges\in cell}} \underline{\underline{N}}_e^r \begin{bmatrix} \tilde{\psi}^{(0)}_{m,n,1,e} \\ \tilde{\psi}^{(0)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(0)}_{m,n,J,e} \end{bmatrix} + \sum_{m,n} w_{m,n}\xi_{m,n} \sum_{\substack{e= \\ edges\in cell}} \underline{\underline{N}}_e^z \begin{bmatrix} \tilde{\psi}^{(0)}_{m,n,1,e} \\ \tilde{\psi}^{(0)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(0)}_{m,n,J,e} \end{bmatrix} = 0. \qquad (4.73)$$

We now look at Eq. (4.73) in more detail. We first substitute the *N* matrix definitions from Eq. (4.56) into the equation and write it for one weight function in cell *k*.

$$\sum_{\substack{e= \\ edges \\ \in cell\ k}} \vec{n}_{e,k} \bullet \sum_{m,n} w_{m,n}\vec{\Omega}_{m,n} \int_e dsru_i \sum_{j=1}^{J} \tilde{\psi}^{(0)}_{m,n,j,e} u_j = 0, \qquad (4.74)$$

where $\vec{n}_{e,k} = n_{r,e,k}\hat{e}_r + n_{z,e,k}\hat{e}_z$ and $\vec{\Omega}_{m,n} = \mu_{m,n}\hat{e}_r + \xi_{m,n}\hat{e}_z$. We break the quadrature sum into two parts—an incoming set of directions and an outgoing set of directions, and use the upwinding condition to determine $\tilde{\psi}^{(0)}_{m,n,j,e}$.

$$\sum_{\substack{e= \\ edges \\ \in cell\ k}} \int_e dsru_i \left\{ \begin{array}{l} \displaystyle\sum_{\substack{m,n: \\ \vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}>0}} w_{m,n}\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}\sum_{j=1}^{J}\psi^{(0)}_{m,n,j}u_j \\ + \displaystyle\sum_{\substack{m,n: \\ \vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0}} w_{m,n}\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}\sum_{j=1}^{J}\psi^{(0)}_{m,n,j,e,+}u_j \end{array} \right\} = 0, \qquad (4.75)$$

where $\psi^{(0)}_{m,n,j}$ is the angular flux unknown in cell *k*, and $\psi^{(0)}_{m,n,j,e,+}$ is the angular flux unknown that is used as the upwinding condition on edge *e* at vertex *j*. At this point we define a useful quantity that represents a weighted average of the half range $\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}$ values.

$$\rho_{e,k} = \frac{2\displaystyle\sum_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}>0} w_{m,n}\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}}{\displaystyle\sum_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}>0} w_{m,n}} = \frac{2\displaystyle\sum_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0} w_{m,n}\left|\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}\right|}{\displaystyle\sum_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0} w_{m,n}} \approx 1 \qquad (4.76)$$

We now focus on a cell that is in the problem interior and simplify Eq. (4.75) using the fact that the leading-order angular flux is isotropic.

$$\sum_{\substack{e= \\ edges \\ \in cell\ k}} \int_e dsru_i \left\{ \sum_{\substack{m,n: \\ \vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}>0}} w_{m,n}\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}\left(\sum_{j=1}^{J}\phi^{(0)}_j u_j - \sum_{j=1}^{J}\phi^{(0)}_{j,e,+}u_j\right)\right\} = 0. \qquad (4.77)$$

In Eq. (4.77) we have multiplied the equation by 4π and noted that

$$\sum_{\substack{m,n: \\ \vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}>0}} w_{m,n}\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n} = -\sum_{\substack{m,n: \\ \vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0}} w_{m,n}\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}. \qquad (4.78)$$

We now substitute Eq. (4.76) into Eq. (4.77), resulting in

$$\sum_{\substack{e= \\ edges \\ \in cell\ k}} \rho_{e,k} \int_e ds r u_i \left(\sum_{j=1}^{J} \phi_j^{(0)} u_j - \sum_{j=1}^{J} \phi_{j,e,+}^{(0)} u_j \right) = 0 . \tag{4.79}$$

Eq. (4.79) is true for all *i* such that the vertex *i* is in the interior of the problem. We can generalize Eq. (4.79) to include edges and vertices on the boundary of the problem domain by defining a boundary scalar flux in terms of the incident angular flux.

$$\phi_{j,e,+}^{(0)} = \phi_{bdy,j,e}^{(0)} = \frac{2}{\rho_{e,k}} \sum_{m,n:\, \vec{n}_{e,k} \bullet \vec{\Omega}_{m,n} < 0} w_{m,n} 2 \left| \vec{n}_{e,k} \bullet \vec{\Omega}_{m,n} \right| \psi_{inc,m,n,j,e}^{(0)} . \tag{4.80}$$

This relationship defines the boundary scalar intensity such that the incident particle flow rate would be the same if the incident angular flux were isotropic, i.e. if $\psi_{inc,m,n}^{(0)} = \frac{\phi_{bdy}^{(0)}}{4\pi}$.

Adams states that the meaning of Eq. (4.79) is that the leading-order scalar flux intensity is "continuous in a weighted-residual sense, but not necessarily continuous pointwise."[6] We can further simplify Eq. (4.79) by lumping the surface matrices. To lump a matrix in general, the integrals in that matrix become

$$\int dA u_i \sum_{j=1}^{J} \psi_{m,n,j}^{(0)} u_j \xrightarrow{lump} \psi_{m,n,i}^{(0)} \int_{edge} dA u_i . \tag{4.81}$$

Given surface-matrix lumping, Eq. (4.79) becomes

$$\sum_{\substack{e= \\ edges \\ \in cell\ k}} \rho_{e,k} \left(\phi_i^{(0)} - \phi_{i,e,+}^{(0)} \right) \int_e ds r u_i = 0 . \tag{4.82}$$

We now examine Eq. (4.82) for all cells surrounding a given vertex. Figure 4.6 shows an interior vertex surrounded by *K* cells.

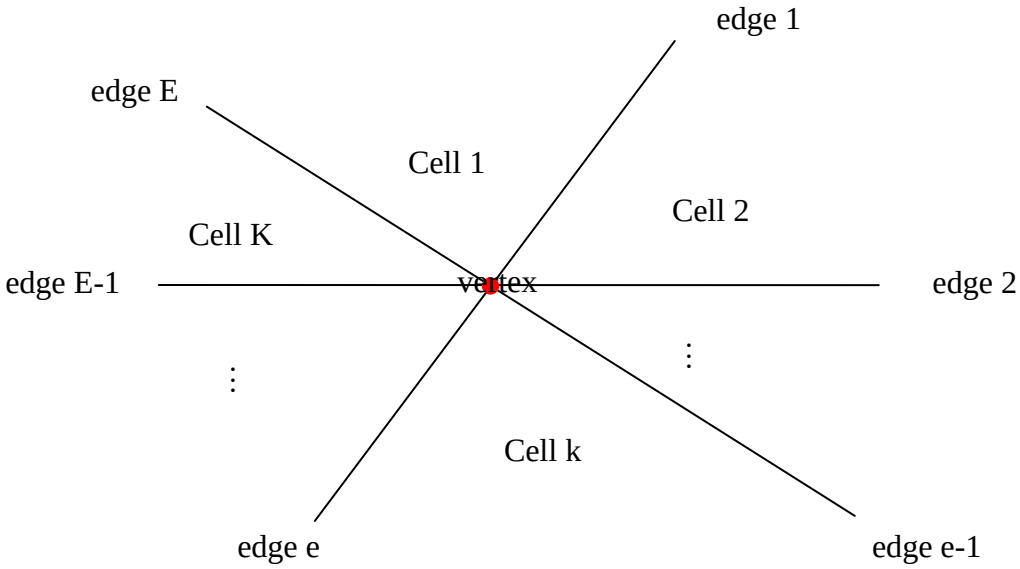

Figure 4.6: Cells surrounding an interior vertex in two dimensions

Using this figure, we can write Eq. (4.82) *K* times—one equation for each cell that contains the interior vertex. Each cell's equation has two terms because each cell has two edges that connect to that vertex. We write these *K* equations:

$$\begin{aligned}
\left(\phi^{(0)}_{i,cell\,1}-\phi^{(0)}_{i,cell\,2}\right)I_{edge\,1,\,cell\,1}+\left(\phi^{(0)}_{i,cell\,1}-\phi^{(0)}_{i,cell\,K}\right)I_{edge\,E,\,cell\,1}&=0\\
\left(\phi^{(0)}_{i,cell\,2}-\phi^{(0)}_{i,cell\,3}\right)I_{edge\,2,\,cell\,2}+\left(\phi^{(0)}_{i,cell\,2}-\phi^{(0)}_{i,cell\,1}\right)I_{edge\,1,\,cell\,2}&=0\\
\left(\phi^{(0)}_{i,cell\,3}-\phi^{(0)}_{i,cell\,4}\right)I_{edge\,3,\,cell\,3}+\left(\phi^{(0)}_{i,cell\,3}-\phi^{(0)}_{i,cell2}\right)I_{edge\,2,\,cell\,3}&=0\\
&\vdots\\
\left(\phi^{(0)}_{i,cell\,K}-\phi^{(0)}_{i,cell\,1}\right)I_{edge\,E,\,cell\,K}+\left(\phi^{(0)}_{i,cell\,K}-\phi^{(0)}_{i,cell\,K-1}\right)I_{edge\,E-1,\,cell\,K}&=0,
\end{aligned} \tag{4.83}$$

where we define

$$I_{edge\,e,\,cell\,k}=I^{k}_{e}=\rho_{e,k}\int\limits_{edge\,e} dsru_{i,\,cell\,k}\ . \tag{4.84}$$

The set of equations in Eq. (4.83) is *K* equations and *K* unknowns, and we write these equations in matrix form

$$\begin{bmatrix} I_1^1 + I_E^1 & -I_1^1 & 0 & \cdots & -I_E^1 \\ -I_1^2 & I_2^2 + I_1^2 & -I_2^2 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & -I_{E-2}^{K-1} & I_{E-1}^{K-1} + I_{E-2}^{K-1} & -I_{E-1}^{K-1} \\ I_E^K & 0 & \cdots & I_{E-1}^K & I_E^K + I_{E-1}^K \end{bmatrix} \begin{pmatrix} \phi_{i,cell1}^{(0)} \\ \phi_{i,cell2}^{(0)} \\ \vdots \\ \phi_{i,cellK-1}^{(0)} \\ \phi_{i,cellK}^{(0)} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}. \quad (4.85)$$

If this matrix system is invertible, then the leading-order scalar flux solution is completely determined by these surface matrix relationships that connect cells together through the upwinding condition. If this is true, our leading-order solution in the interior of the problem is dependent only on this coupling. This result means that the leading-order solution is not dependent on the source or cross-section inside of the cell. Furthermore, if the matrix is invertible and the boundary conditions of our problem are vacuum conditions, then the entire leading order-solution is zero no matter what source terms appear in the problem. Thus, the case where the matrix in Eq. (4.85) is invertible produces a ridiculous leading-order solution.

However, the matrix in Eq. (4.85) is not invertible if $I_{edge\,e,\,cell\,k} = I_{edge\,e,\,cell\,k+1}$. It is easy to see that the column and row sums of all columns and rows in this matrix will be zero if these integrals are equal. These integrals are equal, if the weight function on an edge in a cell exactly matches the weight function on same edge in the adjacent cell. Adams calls this property “surface matching.” [6] PWL weight and basis functions have the surface matching property on all edges in an arbitrary polygon. As a result, the surface lumped PWLD matrix from Eq. (4.85) will be singular for polygonal cells in general. This surface matching property also explains why the LD and BLD methods fail in the diffusion limit on arbitrary polygons. These methods’ weight functions cannot match on an edge between cells because they do not have enough degrees of freedom. As a result, if we write Eq. (4.85) for these methods on arbitrary polygonal cells, the system of

equations will be invertible for every cell type except triangles for LD and quadrilaterals for BLD.

We now must determine how many redundant equations are in the matrix in Eq. (4.85) to determine how many additional constraints we need to find to solve for the leading-order scalar flux. We find that if we reduce our matrix system by one equation and unknown, our system becomes invertible. To do this, we make the *K*th equation and *K*th unknown part of the right hand side of the system:

$$\begin{bmatrix} I_1^1 + I_E^1 & -I_1^1 & \cdots & 0 \\ -I_1^2 & I_2^2 + I_1^2 & -I_2^2 & \vdots \\ 0 & \ddots & \ddots & 0 \\ 0 & \cdots & -I_{E-2}^{K-1} & I_{E-1}^{K-1} + I_{E-2}^{K-1} \end{bmatrix} \begin{pmatrix} \phi_{i,cell\,1}^{(0)} \\ \phi_{i,cell\,2}^{(0)} \\ \vdots \\ \phi_{i,cell\,K-1}^{(0)} \end{pmatrix} = \begin{pmatrix} I_E^1 \\ 0 \\ \vdots \\ I_{E-1}^{K-1} \end{pmatrix} \phi_{i,cell\,K}^{(0)}. \quad (4.86)$$

This system of equations is invertible. Because the system of equations becomes invertible by eliminating only one equation and unknown, we need only one additional constraint to determine our leading-order scalar flux solution. Furthermore, this system of equations reveals some significant behavior of our surface lumped method. Because the system in Eq. (4.86) is invertible, we know it has a unique solution. That is, each of the scalar intensities on the left side is a unique multiple of the single scalar intensity on the right side. As a result, if we find a solution to this system of equations, it is the unique solution. It is easy to see that the solution that satisfies Eq. (4.86) is

$$\phi_{i,cell\,1}^{(0)} = \phi_{i,cell\,2}^{(0)} = \ldots = \phi_{i,cell\,K-1}^{(0)} = \phi_{i,cell\,K}^{(0)}, \quad (4.87)$$

meaning that the leading-order scalar flux is pointwise continuous for an arbitrary number of cells surround a vertex [6]. We remark that this simple result is obtained only with surface lumping. The unlumped equations do not produce a continuous leading-order solution [6]. We will assume surface lumping for the remainder of this analysis.

Next, we try to find the extra constraint to solve for the leading-order scalar flux solution in the interior of the problem. We multiply Eq. (4.70) by $w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)$ and sum over all *m,n*.

$$\begin{aligned}
&\sum_{m,n} w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)\sum_{\substack{e=\\ edges\in cell}}\left\{\mu_{m,n}\underline{\underline{N}}_e^r\begin{bmatrix}\tilde{\psi}^{(0)}_{m,n,1,e}\\ \tilde{\psi}^{(0)}_{m,n,2,e}\\ \vdots\\ \tilde{\psi}^{(0)}_{m,n,J,e}\end{bmatrix}+\xi_{m,n}\underline{\underline{N}}_e^z\begin{bmatrix}\tilde{\psi}^{(0)}_{m,n,1,e}\\ \tilde{\psi}^{(0)}_{m,n,2,e}\\ \vdots\\ \tilde{\psi}^{(0)}_{m,n,J,e}\end{bmatrix}\right\}\\
&-\sum_{m,n} w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)\sum_{\substack{e=\\ edges\in cell}}\left\{\mu_{m,n}\underline{\underline{T}}_e^r\vec{\psi}^{(0)}_{m,n}+\xi_{m,n}\underline{\underline{T}}_e^z\vec{\psi}^{(0)}_{m,n}\right\}\\
&+\sum_{m,n} w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)\left\{\mu_{m,n}\underline{\underline{L}}^r\vec{\psi}^{(0)}_{m,n}+\xi_{m,n}\underline{\underline{L}}^z\vec{\psi}^{(0)}_{m,n}\right\}\\
&+\sum_{m,n} w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)\left[\frac{\alpha^n_{m+1/2}}{w_{m,n}}\underline{\underline{P}}\vec{\psi}^{(0)}_{m+1/2,n}-\frac{\alpha^n_{m-1/2}}{w_{m,n}}\underline{\underline{P}}\vec{\psi}^{(0)}_{m-1/2,n}\right]\\
&+\sum_{m,n} w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)\sigma\underline{\underline{M}}\vec{\psi}^{(1)}_{m,n}-\frac{1}{4\pi}\sum_{m,n} w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)\sigma\underline{\underline{M}}\vec{\phi}^{(1)}=0.
\end{aligned}\tag{4.88}$$

Because we have found that in the interior, the leading-order angular flux is continuous due to surface lumping, we can say that

$$\begin{bmatrix}\tilde{\psi}^{(0)}_{m,n,1,e}\\ \tilde{\psi}^{(0)}_{m,n,2,e}\\ \vdots\\ \tilde{\psi}^{(0)}_{m,n,J,e}\end{bmatrix}=\vec{\psi}^{(0)}_{m,n},\tag{4.89}$$

for interior cells. Also note that because the leading-order angular flux is isotropic

$$\begin{aligned}
\sum_{m,n}\mu^2_{m,n}w_{m,n}\vec{\psi}^{(0)}_{m,n}=\sum_{m,n}\xi^2_{m,n}w_{m,n}\vec{\psi}^{(0)}_{m,n}=\frac{1}{3}\vec{\phi}^{(0)},\\
\sum_{m,n}\mu_{m,n}\xi_{m,n}w_{m,n}\vec{\psi}^{(0)}_{m,n}=0.
\end{aligned}\tag{4.90}$$

Here we have assumed that the quadrature set correctly integrates the square of the direction cosines. When the relationships in Eqs. (4.89) and (4.90) are substituted into Eq. (4.88), the result is

$$\begin{aligned}
&\frac{1}{3}\sum_{\substack{e= \\ edges\in cell}}\underline{\underline{N}}_e^r\vec{\phi}^{(0)}+\frac{1}{3}\sum_{\substack{e= \\ edges\in cell}}\underline{\underline{N}}_e^z\vec{\phi}^{(0)}-\frac{1}{3}\sum_{\substack{e= \\ edges\in cell}}\underline{\underline{T}}_e^r\vec{\phi}^{(0)}-\frac{1}{3}\sum_{\substack{e= \\ edges\in cell}}\underline{\underline{T}}_e^z\vec{\phi}^{(0)} \\
&+\frac{1}{3}\underline{\underline{L}}^r\vec{\phi}^{(0)}+\frac{1}{3}\underline{\underline{L}}^z\vec{\phi}^{(0)}+\sum_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)\left[\alpha^n_{m+\frac{1}{2}}\underline{\underline{P}}\vec{\psi}^{(0)}_{m+\frac{1}{2},n}-\alpha^n_{m-\frac{1}{2}}\underline{\underline{P}}\vec{\psi}^{(0)}_{m-\frac{1}{2},n}\right] \\
&+\sigma\underline{\underline{M}}\sum_{m,n}w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)\vec{\psi}^{(1)}_{m,n}-\frac{1}{4\pi}\sigma\underline{\underline{M}}\vec{\phi}^{(1)}\sum_{m,n}w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)=0.
\end{aligned} \tag{4.91}$$

Because the *N* and *T* matrices are identical, the *N* and *T* terms cancel. We can further simplify Eq. (4.91) using the definition of the current in Eq. (4.60) and by noting that

$$\begin{aligned}
\sum_{m,n}w_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)&=0, \\
\sum_{m,n}\left(\mu_m\hat{e}_r+\xi_m\hat{e}_z\right)\left[\alpha^n_{m+\frac{1}{2}}\vec{\psi}^{(0)}_{m+\frac{1}{2},n}-\alpha^n_{m-\frac{1}{2}}\vec{\psi}^{(0)}_{m-\frac{1}{2},n}\right]&=-\frac{1}{3}\vec{\phi}^{(0)}\hat{e}_r,
\end{aligned} \tag{4.92}$$

resulting in

$$\frac{1}{3}\underline{\underline{L}}^r\vec{\phi}^{(0)}\hat{e}_r+\frac{1}{3}\underline{\underline{L}}^z\vec{\phi}^{(0)}\hat{e}_z-\frac{1}{3}\underline{\underline{P}}\vec{\phi}^{(0)}\hat{e}_r+\sigma\underline{\underline{M}}\left(\vec{J}^{(1)}_r\hat{e}_r+\vec{J}^{(1)}_z\hat{e}_z\right)=0. \tag{4.93}$$

We introduce a compact notation for the current unknowns in the cell. We have a current, which is a vector quantity in *r* and *z*, at every vertex in the cell. Previously, we have denoted the list of unknowns at each vertex as a vector. As a result, our current unknown in the cell is a vector of vectors, which we denote

$$\vec{\vec{J}}=\left(\vec{J}_r\hat{e}_r+\vec{J}_z\hat{e}_z\right). \tag{4.94}$$

From Eq. (4.93) we can now solve for the *O(1)* current:

$$\vec{\vec{J}}^{(1)}=-\frac{1}{3\sigma}\underline{\underline{M}}^{-1}\left\{\left(\underline{\underline{L}}^r-\underline{\underline{P}}\right)\hat{e}_r+\underline{\underline{L}}^z\hat{e}_z\right\}\vec{\phi}^{(0)}. \tag{4.95}$$

Eq. (4.95) is a discretized version of Fick's law, where the stencil of the approximation is determined by the stencil of the individual matrices, M^{-1}, L, and P.

The next step in the analysis is to collect all $O(\varepsilon)$ terms in Eq. (4.66)

$$\sum_{\substack{e= \\ edges \in cell}} \left\{ \mu_{m,n} \underline{\underline{N}}_e^{r,lump} \begin{bmatrix} \tilde{\psi}^{(1)}_{m,n,1,e} \\ \tilde{\psi}^{(1)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(1)}_{m,n,J,e} \end{bmatrix} + \xi_{m,n} \underline{\underline{N}}_e^{z,lump} \begin{bmatrix} \tilde{\psi}^{(1)}_{m,n,1,e} \\ \tilde{\psi}^{(1)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(1)}_{m,n,J,e} \end{bmatrix} \right\}$$

$$- \sum_{\substack{e= \\ edges \in cell}} \left\{ \mu_{m,n} \underline{\underline{T}}_e^{r,lump} \vec{\psi}^{(1)}_{m,n} + \xi_{m,n} \underline{\underline{T}}_e^{z,lump} \vec{\psi}^{(1)}_{m,n} \right\} + \mu_{m,n} \underline{\underline{L}}^{r} \vec{\psi}^{(1)}_{m,n} + \xi_{m,n} \underline{\underline{L}}^{z} \vec{\psi}^{(1)}_{m,n} \qquad (4.96)$$

$$+ \underline{\underline{P}} \frac{\alpha^{n}_{m+\frac{1}{2}} \vec{\psi}^{(1)}_{m+\frac{1}{2},n} - \alpha^{n}_{m-\frac{1}{2}} \vec{\psi}^{(1)}_{m-\frac{1}{2},n}}{w_{m,n}} + \sigma \underline{\underline{M}} \vec{\psi}^{(2)}_{m,n} - \frac{1}{4\pi} \sigma \underline{\underline{M}} \vec{\phi}^{(2)} + \frac{1}{4\pi} \sigma_a \underline{\underline{M}} \vec{\phi}^{(0)} = \frac{1}{4\pi} \underline{\underline{M}} \vec{S}.$$

When Eq. (4.96) is multiplied by $w_{m,n}$ and summed over all (m,n), the result is

$$\sum_{m,n} w_{m,n} \mu_{m,n} \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{N}}_e^{r,lump} \begin{bmatrix} \tilde{\psi}^{(1)}_{m,n,1,e} \\ \tilde{\psi}^{(1)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(1)}_{m,n,J,e} \end{bmatrix} + \sum_{m,n} w_{m,n} \xi_{m,n} \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{N}}_e^{z,lump} \begin{bmatrix} \tilde{\psi}^{(1)}_{m,n,1,e} \\ \tilde{\psi}^{(1)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(1)}_{m,n,J,e} \end{bmatrix}$$

$$+ \left\{ \left(\underline{\underline{L}}^{r} - \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{T}}_e^{r,lump} \right) \vec{J}^{(1)}_{r} + \left(\underline{\underline{L}}^{z} - \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{T}}_e^{z,lump} \right) \vec{J}^{(1)}_{z} \right\} \qquad (4.97)$$

$$+ \underline{\underline{P}} \sum_{m,n} w_{m,n} \left[\frac{\alpha^{n}_{m+\frac{1}{2}} \vec{\psi}^{(1)}_{m+\frac{1}{2},n} - \alpha^{n}_{m-\frac{1}{2}} \vec{\psi}^{(1)}_{m-\frac{1}{2},n}}{w_{m,n}} \right] + \sigma_a \underline{\underline{M}} \vec{\phi}^{(0)} = \underline{\underline{M}} \vec{S}^{(0)}.$$

The angular derivative term in Eq. (4.97) equals zero because the summation is a telescoping sum, where its endpoints are zero because α_M and α_0 are zero by definition, resulting in

$$\sum_{m,n} w_{m,n}\mu_{m,n} \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{N}}_e^{r,lump} \begin{bmatrix} \tilde{\psi}^{(1)}_{m,n,1,e} \\ \tilde{\psi}^{(1)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(1)}_{m,n,J,e} \end{bmatrix} + \sum_{m,n} w_{m,n}\xi_{m,n} \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{N}}_e^{z,lump} \begin{bmatrix} \tilde{\psi}^{(1)}_{m,n,1,e} \\ \tilde{\psi}^{(1)}_{m,n,2,e} \\ \vdots \\ \tilde{\psi}^{(1)}_{m,n,J,e} \end{bmatrix} \tag{4.98}$$
$$+ \left\{ \left(\underline{\underline{L}}^r - \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{T}}_e^{r,lump} \right) \hat{e}_r + \left(\underline{\underline{L}}^z - \sum_{\substack{e= \\ edges \in cell}} \underline{\underline{T}}_e^{z,lump} \right) \hat{e}_z \right\} \bullet \vec{\vec{J}}^{(1)} + \sigma_a \underline{\underline{M}} \vec{\phi}^{(0)} = \underline{\underline{M}} \vec{S}^{(0)}.$$

We now consider the *i*th row of this matrix equation and rewrite this equation for this row using the definitions in Eq. (4.56) and assuming surface matrix lumping in both the *N* and the *T* matrices.

$$\sum_{m,n} w_{m,n}\mu_{m,n} \sum_{\substack{e= \\ edges \\ with\ i}} \tilde{\psi}^{(1)}_{m,n,i,e} n_{r,e} \int_e dsru_i + \sum_{m,n} w_{m,n}\xi_{m,n} \sum_{\substack{e= \\ edges \\ with\ i}} \tilde{\psi}^{(1)}_{m,n,i,e} n_{z,e} \int_e dsru_i$$
$$+ \left(\left(\sum_{j=1}^{J} J^{(1)}_{j,r} \sum_{\substack{sides \\ \in cell}} \int_{side} dAu_i \frac{\partial \left(ru_j \right)}{\partial r} \right) - J^{(1)}_{i,r} \sum_{\substack{e= \\ edges \\ with\ i}} \left(n_{r,e} \int_e dsru_i \right) \right)$$
$$+ \left(\left(\sum_{j=1}^{J} J^{(1)}_{j,z} \sum_{\substack{sides \\ \in cell}} \int_{side} dAu_i \frac{\partial \left(ru_j \right)}{\partial z} \right) - J^{(1)}_{i,z} \sum_{\substack{e= \\ edges \\ with\ i}} \left(n_{z,e} \int_e dsru_i \right) \right) \tag{4.99}$$
$$+ \sigma_a \sum_{sides \in cell} \int_{side} dAru_i \sum_{j=1}^{J} \phi^{(0)}_j u_j = \sum_{sides \in cell} \int_{side} dAru_i \sum_{j=1}^{J} S^{(0)}_j u_j$$

A given interior vertex is surrounded by *K* cells as shown in Figure 4.6. Eq. (4.99) can be written for all cells surrounding a vertex in the interior of the problem domain. When we sum the matrix rows for each cell corresponding to the weight functions centered at vertex *i*, the terms with the angular fluxes defined on cell surfaces cancel, because each appears with a positive sign in one cell and a negative sign in the neighboring cell. Adding the *i*th row of all cells' equations surrounding vertex *i* results in:

$$\sum_{\substack{k=cells\\ around\ i}}\left(\left(\sum_{j=1}^{J}J^{(1)}_{j,k,r}\sum_{\substack{sides\\ \in cell\ k}}\int_{side}dAu_{i,k}\frac{\partial\left(ru_{j,k}\right)}{\partial r}\right)-J^{(1)}_{i,k,r}\sum_{\substack{e_k=\\ edges\\ with\ i}}\left(n_{r,e,k}\int_{e}dsru_{i,k}\right)\right)$$
$$+\sum_{\substack{k=cells\\ around\ i}}\left(\left(\sum_{j=1}^{J}J^{(1)}_{j,k,z}\sum_{\substack{sides\\ \in cell\ k}}\int_{side}dAu_{i,k}\frac{\partial\left(ru_{j,k}\right)}{\partial z}\right)-J^{(1)}_{i,k,z}\sum_{\substack{e_k=\\ edges\\ with\ i}}\left(n_{z,e,k}\int_{e}dsru_{i,k}\right)\right) \qquad (4.100)$$
$$+\sum_{\substack{k=cells\\ around\ i}}\sigma_{a,k}\sum_{\substack{sides\\ \in cell\ k}}\int_{side}dAru_{i,k}\sum_{j=1}^{J}\phi^{(0)}_{j,k}u_{j,k}=\sum_{\substack{k=cells\\ around\ i}}\sum_{\substack{sides\\ \in cell\ k}}\int_{side}dAru_{i,k}\sum_{j=1}^{J}S^{(0)}_{j,k}u_{j,k}.$$

The system of equations, given by Eqs. (4.100) and (4.95) is a valid discretization of the diffusion equation. This discretization has a nine-point stencil on an orthogonal grid, much like a standard bilinear continuous finite element method applied to the diffusion equation. If we lump the mass matrix, *M*, in the same way that we lumped the surface matrices, we enhance the diagonal dominance of the resultant coefficient matrix and improve the robustness of the method in the diffusion limit.

We further define an *F* matrix actually a matrix of r,z vectors):

$$\underline{\underline{F}}_k\vec{\phi}^{(0)}=\vec{\vec{J}}^{(1)}=\underline{\underline{M}}_k^{-1}\left\{\left(\underline{\underline{L}}^r_k-\underline{\underline{P}}_k\right)\hat{e}_r+\underline{\underline{L}}^z_k\hat{e}_z\right\}\vec{\phi}^{(0)}, \qquad (4.101)$$

where, when we apply mass lumping, an element of this matrix is defined as

$$\underline{\underline{\vec{F}}}_k \rightarrow f_{i,j,k} = \frac{\int\limits_{cell\,k} dAru_i \left(\frac{\partial u_j}{\partial r}\hat{e}_r + \frac{\partial u_j}{\partial z}\hat{e}_z \right)}{\int\limits_{cell\,k} dAru_i}. \tag{4.102}$$

If we define the diffusion coefficient to be

$$D_k = \frac{1}{3\sigma_k}, \tag{4.103}$$

then the first order current becomes

$$\begin{aligned} J_{i,r,k}^{(1)}\hat{e}_r + J_{i,z,k}^{(1)}\hat{e}_z &= -D_k \sum_{j=1}^{J} f_{i,j,k}\phi_{j,k}^{(0)} \\ &= -D_k \sum_{j=1}^{J} \frac{\int\limits_{cell\,k} dAru_i \left(\frac{\partial u_j}{\partial r}\hat{e}_r + \frac{\partial u_j}{\partial z}\hat{e}_z \right)\phi_{j,k}^{(0)}}{\int\limits_{cell\,k} dAru_i}. \end{aligned} \tag{4.104}$$

Substituting this current into Eq. (4.100) results in

$$
-\sum_{k=1}^{K}\left[\begin{array}{l}
D_k\sum_{j=1}^{J}\dfrac{\left[\left(\int_{\text{cell } k} dAu_{i,k}\dfrac{\partial\left(ru_{j,k}\right)}{\partial r}-\int_{\text{cell } k} dAu_{i,k}u_{j,k}\right)\int_{\text{cell } k} dAu_{i,k}\dfrac{\partial\left(ru_{j,k}\right)}{\partial r}\right]}{\int_{\text{cell } k} dVu_{i,k}}\phi_{j,k}^{(0)} \\
-D_k\dfrac{\left[\left(\int_{\text{cell } k} dAu_{i,k}\dfrac{\partial\left(ru_{i,k}\right)}{\partial r}-\int_{\text{cell } k} dAu_{i,k}u_{i,k}\right)\right]}{\int_{\text{cell } k} dVu_{i,k}}\sum_{\substack{e_k=\\ \text{edges}\\ \text{with } i}}\left(n_{r,e,k}\int_e dsru_{i,k}\right)\phi_{i,k}^{(0)} \\
+D_k\sum_{j=1}^{J}\dfrac{\left[\left(\dfrac{\partial\left(ru_{j,k}\right)}{\partial z}\int_{\text{cell } k} dAu_{i,k}\right)\int_{\text{cell } k} dAu_{i,k}\dfrac{\partial\left(ru_{j,k}\right)}{\partial z}\right]}{\int_{\text{cell } k} dVu_{i,k}}\phi_{j,k}^{(0)} \\
-D_k\dfrac{\left[\left(\int_{\text{cell } k} dAu_{i,k}\dfrac{\partial\left(ru_{i,k}\right)}{\partial z}-\int_{\text{cell } k} dAu_{i,k}u_{i,k}\right)\right]}{\int_{\text{cell } k} dVu_{i,k}}\sum_{\substack{e_k=\\ \text{edges}\\ \text{with } i}}\left(n_{z,e,k}\int_e dsru_{i,k}\right)\phi_{i,k}^{(0)}
\end{array}\right] \quad (4.105)
$$

$$
+\sum_{k=1}^{K}\left[\sigma_{a,k}\phi_{i,k}^{(0)}\int_{\text{cell } k} dAru_{i,k}-S_{i,k}^{(0)}\int_{\text{cell } k} dAru_{i,k}\right]=0.
$$

Eq. (4.105) represents a valid discretization of the RZ diffusion equation. This discretization has a 9 point stencil on a quadrilateral grid. Coupling to non-neighbor vertices only occurs through the within-cell gradient integrals and the angular derivative integrals because we have lumped the surface and mass matrices. If we had not lumped the surface matrices the diffusion discretization would result in a symmetric positive definite (SPD) coefficient matrix, but this method will not produce an SPD matrix in general because we haven chosen to surface lump our method.

Diffusion limit analysis of the boundary cells

The previous diffusion limit analysis considers cells that are not on the boundary of the problem domain. Because the boundary conditions of the problem directly affect the cells on the boundary of the problem, we have to treat the analysis on these boundary cells separately. From the asymptotic diffusion limit analysis performed on the analytic transport equation (which is straightforward), we know that the boundary scalar intensity that yields the correct leading order solution in the interior is

$$\Phi^{(0)}\left(\vec{r}\right) = 2 \int_{\vec{n}\bullet\vec{\Omega}<0} d\Omega W\left(\left|\vec{n}\bullet\vec{\Omega}\right|\right)\psi_{inc}\left(\vec{r},\vec{\Omega}\right), \tag{4.106}$$

where W is defined in terms of Chandrasekhar's H-function [6].

$$W\left(\left|\vec{n}\bullet\vec{\Omega}\right|\right) \equiv \frac{\sqrt{3}}{2}\mu \mathrm{H}\left(\mu\right) \approx \mu + \frac{3}{2}\mu^2 \tag{4.107}$$

If our method's leading-order solution satisfies an accurate discrete diffusion equation subject to a Dirichlet scalar intensity on the boundary that is a good approximation to Eq.(4.106), we conclude that it will produce the correct leading order interior solution in the diffusion limit.

Following the example in the Adams paper [6] and assuming surface matrix lumping and mass matrix lumping, we take the first angular moment of the *i*th row of Eq.(4.70):

$$\begin{aligned}
&\sum_{e=1}^{e_k}\vec{n}_{e,k}\bullet\sum_{m,n}w_{m,n}\vec{\Omega}_{m,n}\vec{\Omega}_{m,n}\tilde{\psi}^{(0)}_{m,n,e,i}\int_e dsru_i\\
&-\sum_{e=1}^{e_k}\vec{n}_{e,k}\bullet\sum_{m,n}w_{m,n}\vec{\Omega}_{m,n}\vec{\Omega}_{m,n}\psi^{(0)}_{m,n,i}\int_e dsru_i\\
&+\sum_{m,n}w_{m,n}\vec{\Omega}_{m,n}\vec{\Omega}_{m,n}\bullet\sum_{sides}\int_{side}dAu_i\vec{\nabla}r\sum_{j=1}^{J}\psi^{(0)}_{m,n,j}u_j\\
&+\sum_{m,n}w_{m,n}\vec{\Omega}_{m,n}\sum_{side}\int_{side}dAu_i\frac{\alpha^n_{m+\frac{1}{2}}\sum_{j=1}^{J}\psi^{(0)}_{m,n,j}u_j-\alpha^n_{m-\frac{1}{2}}\sum_{j=1}^{J}\psi^{(0)}_{m,n,j}u_j}{w_{m,n}}\\
&+\sigma\sum_{m,n}w_{m,n}\vec{\Omega}_{m,n}\psi^{(1)}_{m,n,i}\sum_{sides}\int_{side}dAru_i=\frac{\sigma}{4\pi}\sum_{m,n}w_{m,n}\vec{\Omega}_{m,n}\phi^{(1)}_i\sum_{sides}\int_{side}dAru_i.
\end{aligned}\tag{4.108}$$

In Eq. (4.108), we have defined e_k to be the number of edges of cell k. Using previously defined quantities, it is easy to see that Eq. (4.108) simplifies to

$$\begin{aligned}
&\sum_{e=1}^{e_k}\vec{n}_{e,k}\bullet\sum_{m,n}w_{m,n}\vec{\Omega}_{m,n}\vec{\Omega}_{m,n}\tilde{\psi}^{(0)}_{m,n,e,i}\int_e dsru_i\\
&-\sum_{e=1}^{e_k}\vec{n}_{e,k}\bullet\sum_{m,n}w_{m,n}\vec{\Omega}_{m,n}\vec{\Omega}_{m,n}\frac{\phi^{(0)}_i}{4\pi}\int_e dsru_i\\
&+\frac{1}{4\pi}\sum_{m,n}w_{m,n}\vec{\Omega}_{m,n}\vec{\Omega}_{m,n}\bullet\sum_{sides}\int_{side}dAu_i\vec{\nabla}r\sum_{j=1}^{J}\phi^{(0)}_j u_j\\
&+\sum_{m,n}\vec{\Omega}_{m,n}\sum_{side}\int_{side}dAu_i\left(\alpha^n_{m+\frac{1}{2}}\sum_{j=1}^{J}\psi_{m+\frac{1}{2},n,j}u_j-\alpha^n_{m-\frac{1}{2}}\sum_{j=1}^{J}\psi_{m-\frac{1}{2},n,j}u_j\right)\\
&+\sigma\bar{J}^{(1)}_i\sum_{sides}\int_{side}dAru_i=0,
\end{aligned}\tag{4.109}$$

where $\bar{J}_i=J_{i,r}\hat{e}_r+J_{i,z}\hat{e}_z$. The current is not a double vector because we are looking at only one row of the matrix in a cell and the mass matrix is lumped, so there is only one vertex current in this equation. We continue to assume that our quadrature set integrates the quadratic functions of the direction cosines correctly:

$$\sum_{m,n} w_{m,n}\vec{\Omega}_{m,n}\vec{\Omega}_{m,n} = \frac{4\pi}{3}\underline{\underline{I}}, \tag{4.110}$$

and use the relationships in Eq. (4.92) to obtain

$$\begin{aligned} &\sum_{e=1}^{e_k}\vec{n}_{e,k}\bullet\sum_{m,n} w_{m,n}\vec{\Omega}_{m,n}\vec{\Omega}_{m,n}\tilde{\psi}^{(0)}_{m,n,e,i}\int_e dsru_i \\ -\frac{\phi_i^{(0)}}{3}\sum_{e=1}^{e_k}\vec{n}_{e,k}\int_e dsru_i &+\left(\frac{1}{3}\sum_{sides}\int_{side} dAu_i\sum_{j=1}^{J}\phi_j^{(0)}u_j\right)\hat{e}_r + \frac{1}{3}\sum_{sides}\int_{side} dAu_i\vec{\nabla}\left(\sum_{j=1}^{J}\phi_j^{(0)}u_j\right)+ \\ &-\left(\frac{1}{3}\sum_{side}\int_{side} dAu_i\sum_{j=1}^{J}\phi_j^{(0)}u_j\right)\hat{e}_r + \sigma\bar{J}_i^{(1)}\sum_{sides}\int_{side} dAru_i = 0. \end{aligned} \tag{4.111}$$

The first term can be divided into “incoming” and “outgoing” components.

$$\begin{aligned} &\frac{1}{3}\sum_{e=1}^{e_k}\left[\vec{n}_{e,k}\frac{\phi_i^{(0)}}{2} + \sum_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0} 3w_{m,n}\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}\vec{\Omega}_{m,n}\tilde{\psi}^{(0)}_{m,n,e,i,+}\right]\int_e dsru_i \\ &-\frac{\phi_i^{(0)}}{3}\sum_{e=1}^{e_k}\vec{n}_{e,k}\int_e dsru_i + \frac{1}{3}\sum_{sides}\int_{side} dAu_i\vec{\nabla}\sum_{j=1}^{J}\phi_j^{(0)}u_j \\ &+\sigma\bar{J}_i^{(1)}\sum_{sides}\int_{side} dAru_i = 0, \end{aligned} \tag{4.112}$$

where we have used the fact that the interior leading-order angular flux is isotropic and we have assumed that our quadrature set performs half range integrals correctly:

$$\vec{n}\bullet\sum_{m,n:\vec{n}\bullet\vec{\Omega}_{m,n}>0} w_{m,n}\vec{\Omega}_{m,n}\vec{\Omega}_{m,n} = \frac{2\pi}{3}\vec{n}\,. \tag{4.113}$$

Based on previous discussion, we can show that the leading order-scalar fluxes are pointwise continuous if the incident partial currents are continuous. Figure 4.7 shows a boundary case where three cells meet at the boundary.

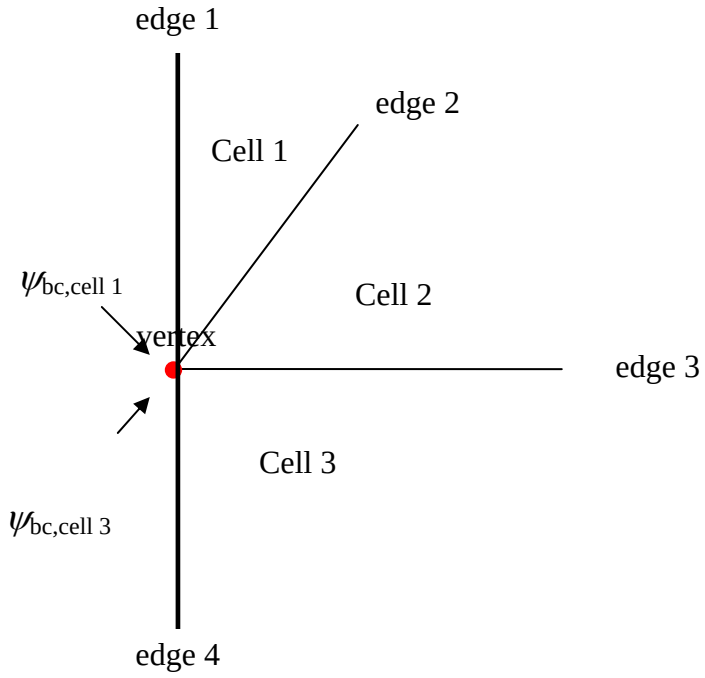

Figure 4.7: Cells on a boundary

We can write Eq. (4.82) for all cells and edges on the boundary if we state that at the boundary $\phi_{i,e,+}^{(0)} = \phi_{bdy,i,e}^{(0)}$, which was defined in Eq. (4.80). We then re-write Eq. (4.83) for this case:

$$\begin{aligned}
&\left(\phi_{i,cell\,1}^{(0)} - \phi_{i,cell\,2}^{(0)}\right) I_{edge\,2,\,cell\,1} + \left(\phi_{i,cell\,1}^{(0)} - \phi_{bdy,i,1}^{(0)}\right) I_{edge\,1,\,cell\,1} = 0 \\
&\left(\phi_{i,cell\,2}^{(0)} - \phi_{i,cell\,3}^{(0)}\right) I_{edge\,3,\,cell\,2} + \left(\phi_{i,cell\,2}^{(0)} - \phi_{i,cell\,1}^{(0)}\right) I_{edge\,2,\,cell\,2} = 0 \\
&\left(\phi_{i,cell\,3}^{(0)} - \phi_{bdy,i,3}^{(0)}\right) I_{edge\,4,\,cell\,3} + \left(\phi_{i,cell\,3}^{(0)} - \phi_{i,cell\,2}^{(0)}\right) I_{edge\,3,\,cell\,3} = 0.
\end{aligned} \tag{4.114}$$

We assume for now that the two defined boundary scalar intensities are equal. This is an assumption that the incident partial current is spatially continuous, which will be true for many (but not all) problems of interest. We write the equations for the three interior intensities in terms of this single boundary intensity.

$$\begin{aligned}
\left(\phi^{(0)}_{i,cell\,1}-\phi^{(0)}_{i,cell\,2}\right)I_{edge\,2,\,cell\,1}+\phi^{(0)}_{i,cell\,1}I_{edge\,1,\,cell\,1}&=\phi^{(0)}_{bdy,i}I_{edge\,1,\,cell\,1}\\
\left(\phi^{(0)}_{i,cell\,2}-\phi^{(0)}_{i,cell\,3}\right)I_{edge\,3,\,cell\,2}+\left(\phi^{(0)}_{i,cell\,2}-\phi^{(0)}_{i,cell\,1}\right)I_{edge\,2,\,cell\,2}&=0\\
\phi^{(0)}_{i,cell\,3}I_{edge\,4,\,cell\,3}+\left(\phi^{(0)}_{i,cell\,3}-\phi^{(0)}_{i,cell2}\right)I_{edge\,3,\,cell\,3}&=\phi^{(0)}_{bdy,i}I_{edge\,4,\,cell\,3}
\end{aligned}\qquad(4.115)$$

This system is analogous to the one in Eq.(4.86). As a result,

$$\phi^{(0)}_{i,cell\,1}=\phi^{(0)}_{i,cell\,2}=\phi^{(0)}_{i,cell\,3}=\phi^{(0)}_{bdy,i}\,.\qquad(4.116)$$

Again this result is only true for the surface lumped case and only if the incident partial currents are spatial continuous.

We are now able to find the boundary conditions satisfied by the leading-order interior solution of this method through the surface angular flux, $\psi^{(0)}_{inc,m,n,i,e}=\tilde{\psi}^{(0)}_{m,n,e,i,+}$, in Eq. (4.112). We see that, the boundary condition in the thick diffusion limit for our surface lumped DFEM involves the quantity

$$DFEM=\left[\frac{\phi^{(0)}_i}{2}\vec{n}_{e,k}+\sum_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0}3w_{m,n}\left(\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}\right)\vec{\Omega}_{m,n}\psi^{(0)}_{inc,m,n,i,e}\right]\int_e dsu_{i,k}\,.\qquad(4.117)$$

Ideally the surface scalar intensities would satisfy the same condition of the analytic diffusion limit. This ideal condition is given by

$$Ideal=2\sum_{m,n:\,\vec{n}\bullet\vec{\Omega}_{m,n}<0}w_{m,n}W\left(\left|\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}\right|\right)\psi_{inc,m,n,e,i}\int_{edge}dsu_{i,k}\left(\vec{r}\right).\qquad(4.118)$$

Because we have shown that the leading-order scalar flux on the boundary is pointwise continuous, we are able to write $\phi^{(0)}_i$ in terms of $\psi^{(0)}_{inc,m,n,i,e}$, because it is equal to $\phi^{(0)}_{bdy,i}$, which is defined in Eq. (4.80). We can now substitute Eq. (4.80) into Eq. (4.117) to discover our method's boundary term in the diffusion limit.

$$DFEM = \left[\begin{array}{l} \sum\limits_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0} w_{m,n} 2 \dfrac{\vec{n}_{e,k}\left|\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}\right|}{\rho_{e,k}} \psi^{(0)}_{inc,m,n,i,e} \\ + \sum\limits_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0} 3 w_{m,n} \vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}\vec{\Omega}_{m,n}\psi^{(0)}_{inc,m,n,i,e} \end{array} \right] \int\limits_{edge} dAu_i \qquad (4.119)$$

We now define $\mu_{m,n} = \vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}$, causing Eq. (4.119) to simplify to

$$DFEM = \left[\sum_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0} w_{m,n} \left(\vec{n}_{e,k} 2 \frac{\left|\mu_{m,n}\right|}{\rho_{e,k}} + 3\mu_{m,n}\vec{\Omega}_{m,n} \right) \psi^{(0)}_{inc,m,n,i,e} \right] \int\limits_{edge} dAu_i \,. \qquad (4.120)$$

Next, we separate the remaining $\vec{\Omega}_{m,n}$ term into a normal and tangential component:

$$\begin{array}{l} \vec{\Omega}_{m,n} = \mu_{m,n}\vec{n}_{e,k} + \vec{\omega}_{m,n}, \\ \vec{\omega}_{m,n} \equiv \text{tangential component} \end{array} \qquad (4.121)$$

and substitute it into Eq. (4.120):

$$DFEM = \left[\sum_{\substack{m,n: \\ \vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0}} w_{m,n} \left(\vec{n}_{e,k} 2 \frac{\left|\mu_{m,n}\right|}{\rho_{e,k}} + 3\mu_{m,n} \left(\mu_{m,n}\vec{n}_{e,k} + \vec{\omega}_{m,n} \right) \right) \psi^{(0)}_{inc,m,n,i,e} \right] \int\limits_{edge} dAu_i \,. \qquad (4.122)$$

Eq. (4.122) can be written in terms of a boundary term that is normal to the surface and one that is tangential to the surface.

$$\begin{array}{l} DFEM = \left[\vec{n}_{e,k}\Phi_{bdy,e,k,i} + \sum\limits_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0} 3 w_{m,n}\mu_{m,n}\vec{\omega}_{m,n}\psi^{(0)}_{inc,m,n,i,e} \right] \int\limits_{edge} dAu_i \\ \Phi_{bdy,e,k,i} = \sum\limits_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0} w_{m,n} \left[2\dfrac{\left|\mu_{m,n}\right|}{\rho_{e,k}} + 3\mu^2_{m,n} \right] \psi^{(0)}_{inc,m,n,i,e} \\ \approx \sum\limits_{m,n:\,\vec{n}_{e,k}\bullet\vec{\Omega}_{m,n}<0} w_{m,n} \left[2\left|\mu_{m,n}\right| + 3\mu^2_{m,n} \right] \psi^{(0)}_{inc,m,n,i,e} \end{array} \qquad (4.123)$$

Eq. (4.123) shows that the normal component of the boundary term for the surface lumped DFEM is a good approximation to the analytic limit, given in Eq. (4.107). The tangential term, however, is an erroneous contribution to the DFEM solution—there is no counterpart to this term in the analytic leading-order solution. Thus, the DFEM solution is flawed to the extent that the tangential boundary term affects it.

To determine the extent to which the tangential term affects the leading-order solution, we need to examine the result of applying our boundary condition to Eq. (4.112). This equation is used to develop an expression for the net current in the cell. As a result, the tangential boundary term, which contaminates the solution, only does so through this net current. The expression for the net current in a cell is given in Eq. (4.101). If we examine the matrix terms closely in this equation, we find that the leading-order scalar flux is being multiplied by integrals whose solutions involve outward-going normals of cell edges. The radial component of the current is the within-cell-gradient matrix in *r* minus the angular derivative matrix. On a side, the integrals for the within-cell-gradient terms are shown in Eq. (4.43), and the integrals for the angular derivative terms are shown in Eq. (4.36). When these two matrices are subtracted, all that is left are terms that contain the normal value of the edge. The z component of the current only contains the within-cell-gradient matrix in *z*. Every integral in this matrix, found in Eq. (4.43), contains an edge normal value. For orthogonal grids, these edge normals are zeros for directions tangential to the boundary. As a result, even though the boundary introduces a tangential term, this term will never propagate through the solution because it is multiplied by something that is zero in the problem interior.

The effect of the tangential boundary terms on solutions with non-orthogonal cells is more difficult to predict. These tangential boundary terms will propagate into the interior of the problem and act like fictitious sources to interior cells. However, some of these fictitious sources are positive and some are negative. Near the boundary we anticipate that these fictitious sources will impact the solution, creating negative fluxes,

but for cells sufficiently far from the boundary we predict these fictitious sources will cancel each other out. Adams, Wareing and Walters have demonstrated this property [13] for characteristic methods in the thick diffusion limit. We expect our method to behave similarly.

We remark that our boundary term analysis assumed surface lumping of the method. For a non-surface lumped method, the analysis is not as simple, but the basic conclusions remain the same [6].

Lumping in RZ geometry

The asymptotic analysis of the interior has suggested a few changes to the PWL method that will enhance its performance in the thick diffusive limit. The first change is to lump the surface integral terms (the *N* and *T* matrices in Eqs. (4.55) and (4.57)). This lumping ensures that the leading order angular flux is continuous at interior vertices. The second change is to lump the mass matrix. This lumping adds diagonal dominance to the approximation of Fick's law and to the absorption term. We have implemented a version of the PWL method that incorporates surface and mass matrix lumping in the Capsaicin code, and we call it surface-mass lumping.

Morel and Warsa have proposed another lumping scheme, which we refer to as generalized lumping, that can be applied to any DFEM discretization of the RZ transport equation [24]. To derive this lumping scheme, we begin with the RZ transport equation:

$$\frac{1}{r}\frac{\partial(r\psi)}{\partial r}-\frac{1}{r}\frac{\partial\eta\psi}{\partial\omega}+\xi\frac{\partial\psi}{\partial z}+\sigma\psi=\sigma_s\phi+q\,. \qquad (4.124)$$

We multiply Eq. (4.124) by a weight function, v_1, and integrate over a cell volume. Then we apply Gauss-Divergence to the streaming term integrals resulting in

$$\int_{\partial V} v_1 r\vec{\Omega}\bullet\vec{n}\psi dS - \int_V r\psi\vec{\Omega}\bullet\vec{\nabla}\left(v_1\right)dV - \int_V v_1\frac{\partial\eta\psi}{\partial\omega}dV + \sigma_t\int_V rv_1\psi dV = \sigma_s\int_V rv_1\phi dV + \int_V rv_1 q dV. \tag{4.125}$$

We apply Gauss-Divergence a second time to the second term in Eq. (4.125) resulting in

$$\int_{\partial V} v_1 r\vec{\Omega}\bullet\vec{n}\tilde{\psi}dS - \int_{\partial V} v_1 r\vec{\Omega}\bullet\vec{n}\psi dS + \int_V v_1\vec{\Omega}\bullet\vec{\nabla}\left(r\psi\right)dV - \int_V v_1\frac{\partial\eta\psi}{\partial\omega}dV + \sigma_t\int_V rv_1\psi dV = \sigma_s\int_V rv_1\phi dV + \int_V rv_1 q dV. \tag{4.126}$$

We now lump the surface, mass and angular derivative integrals, resulting in

$$\left(\tilde{\psi}_1 - \psi_1\right)\int_{\partial V} v_1 r\vec{\Omega}\bullet\vec{n}dS + \int_V v_1\vec{\Omega}\bullet\vec{\nabla}\left(r\psi\right)dV - \frac{\partial\eta\psi_1}{\partial\omega}\int_V v_1 dV + \sigma_t\psi_1\int_V rv_1 dV = \sigma_s\phi_1\int_V rv_1 dV + q_1\int_V rv_1 dV. \tag{4.127}$$

The gradient term can be expanded using the product rule of differentiation

$$\left(\tilde{\psi}_1 - \psi_1\right)\int_{\partial V} v_1 r\vec{\Omega}\bullet\vec{n}dS + \int_V rv_1\vec{\Omega}\bullet\vec{\nabla}\left(\psi\right)dV + \int_V \psi v_1\vec{\Omega}\bullet\vec{\nabla}\left(r\right)dV - \frac{\partial\eta\psi_1}{\partial\omega}\int_V v_1 dV + \sigma_t\psi_1\int_V rv_1 dV = \sigma_s\phi_1\int_V rv_1 dV + q_1\int_V rv_1 dV. \tag{4.128}$$

Note that the gradient of *r* reduces to $1\hat{i} + 0\,\hat{j}$, causing Eq. (4.128) to simplify to

$$\left(\tilde{\psi}_1 - \psi_1\right)\int_{\partial V} v_1 r\vec{\Omega}\bullet\vec{n}dS + \int_V rv_1\vec{\Omega}\bullet\vec{\nabla}\left(\psi\right)dV + \mu\int_V \psi v_1 dV - \frac{\partial\eta\psi_1}{\partial\omega}\int_V v_1 dV + \sigma_t\psi_1\int_V rv_1 dV = \sigma_s\phi_1\int_V rv_1 dV + q_1\int_V rv_1 dV. \tag{4.129}$$

We further lump the third term in Eq. (4.129) resulting in

$$\begin{aligned}&\left(\tilde{\psi}_1-\psi_1\right)\int_{\partial V} v_1 r\vec{\Omega}\bullet\vec{n}dS+\int_V rv_1\vec{\Omega}\bullet\vec{\nabla}\left(\psi\right)dV+\mu\psi_1\int_V v_1 dV\\&-\frac{\partial\eta\psi_1}{\partial\omega}\int_V v_1 dV+\sigma_t\psi_1\int_V rv_1 dV=\sigma_s\phi_1\int_V rv_1 dV+q_1\int_V rv_1 dV.\end{aligned}\tag{4.130}$$

Eq. (4.130) is the general lumped version of a DFEM applied to the RZ transport equation. We have also implemented this lumping method for PWL in Capsaicin.

Test problems for RZ geometry

We have tested the PWLD method with a variety of test problems. The first PWL property we tested is its solution on triangular grids. As noted previously, the PWL method on a triangular grid collapses to the LD method on a triangular grid. We ran multiple test problems that compared these two methods and found their solutions to be identical within the numerical precision of the computer. We also developed unit tests to test the method on a problem with a zero solution, a problem with a constant solution, and a variety of problems with manufactured solutions. An example of a manufactured solution problem is one with the solution of

$$\phi\left(r,z\right)=e^{\left(z-r^2\right)}.\tag{4.131}$$

These unit tests involved multi-group and parallel calculations. The unlumped and lumped versions of the PWL method pass all of these unit tests [25].

The majority of the test problems we have run have been to determine the truncation error of the PWL method. Truncation error test problems are used to determine the order by which the solution of the numerical method is converging to the actual solution. This order is *n* if the error in the solution decrease by a factor of to 2^n every time the mesh is refined by a factor of two in each dimension. For example, we say a method is second order if the error of the solution in some norm decreases by a factor of four every time we refine the mesh by a factor of two in every dimension. The first truncation error

problem we used to test the methods was a problem with a manufactured solution, which is a standard test problem for RZ methods in Capsaicin. The manufactured solution is

$$\phi(r,z) = \left[\sin(\pi r) + 1 - r\right]\sin(\pi z), \tag{4.132}$$

and we used a level symmetric quadrature set for this test problem [25].

We first tested the unlumped version of PWL on a rectangular grid and a randomized quadrilateral mesh. Figure 4.8 shows a plot of the convergence rate of PWL and BLD on these meshes. The reference line in this figure and all subsequent truncation error figures is an arbitrary line with a slope representative of second-order convergence behavior. The error for both methods is almost identical, and the error between the two meshes is extremely close. Furthermore, both methods exhibit second-order convergence behavior, which is expected.

We then tested the both the surface-mass lumped and the generalized lumped versions of PWL and BLD on the orthogonal and random meshes. The results of these test problems are shown in Figure 4.9. This plot shows that, for this test problem, the type of lumping makes no difference in the order of convergence or in the magnitude of the error. These results also suggest that the solutions on random grids are slightly less accurate than solutions on orthogonal grids.

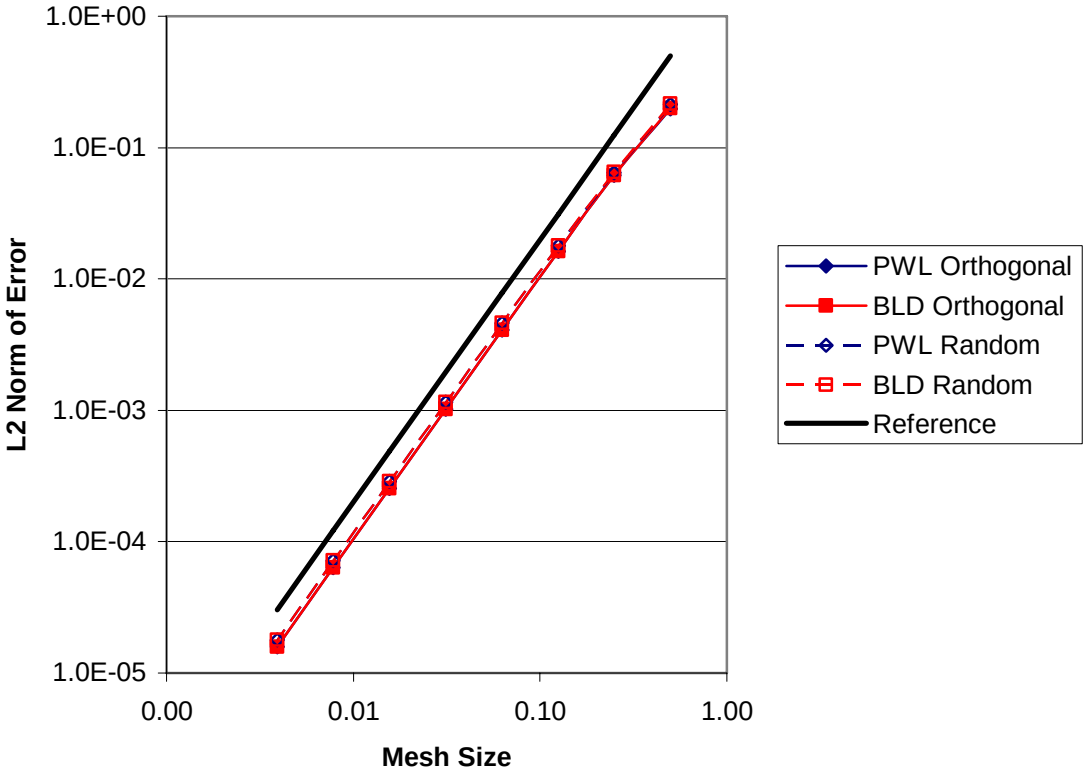

Figure 4.8: Truncation error of unlumped methods

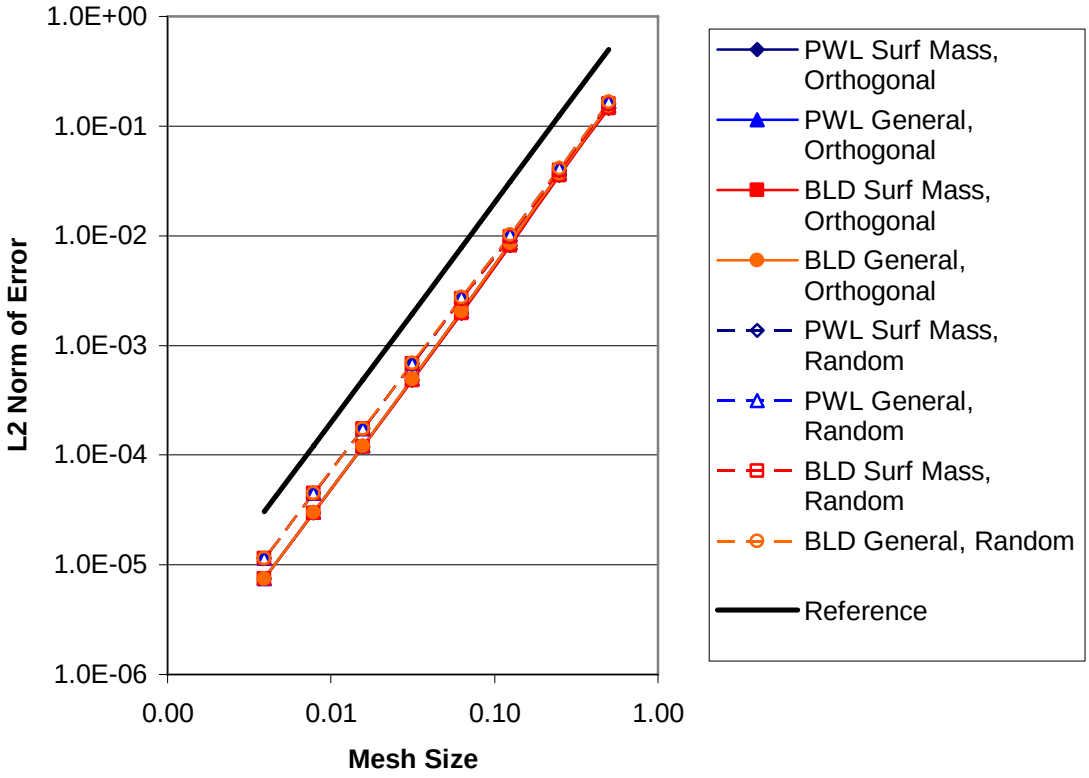

Figure 4.9: Truncation error of lumped methods

We also studied the truncation error of the lumped methods for a diffusive problem, where the problem is resolved in diffusion lengths, but unresolved in mean free paths. This problem was developed by Morel and Warsa to test their generalized lumping scheme [24]. It is a one dimensional problem that has an analytic solution of

$$\phi(z) = \frac{S}{D}\left\{\left[\frac{1-e^{1/L}}{e^{-1/L}-e^{1/L}}-1\right]e^{z/L}+\left[\frac{e^{1/L}-1}{e^{-1/L}-e^{1/L}}\right]e^{-z/L}+1\right\}+O\left(\varepsilon^2\right) \tag{4.133}$$

$$\varepsilon \approx \frac{1}{\sqrt{1-c}} \approx 10^{-3}$$

with a total cross section of 8192 cm^{-1} and a scattering ratio of 0.9999987. L in Eq. (4.133) is a diffusion length. We ran this problem on a random quadrilateral mesh of N cells in the z direction and $N/4$ cells in the r direction, and used a square Chebyshev quadrature set. The results of this convergence rate problem, shown in Figure 4.10, indicate that PWLD has the almost identical accuracy to BLD in the thick diffusive limit, and that all methods exhibit second-order convergence behavior. These results also indicate that both PWLD lumping methods have similar convergence properties for this problem.

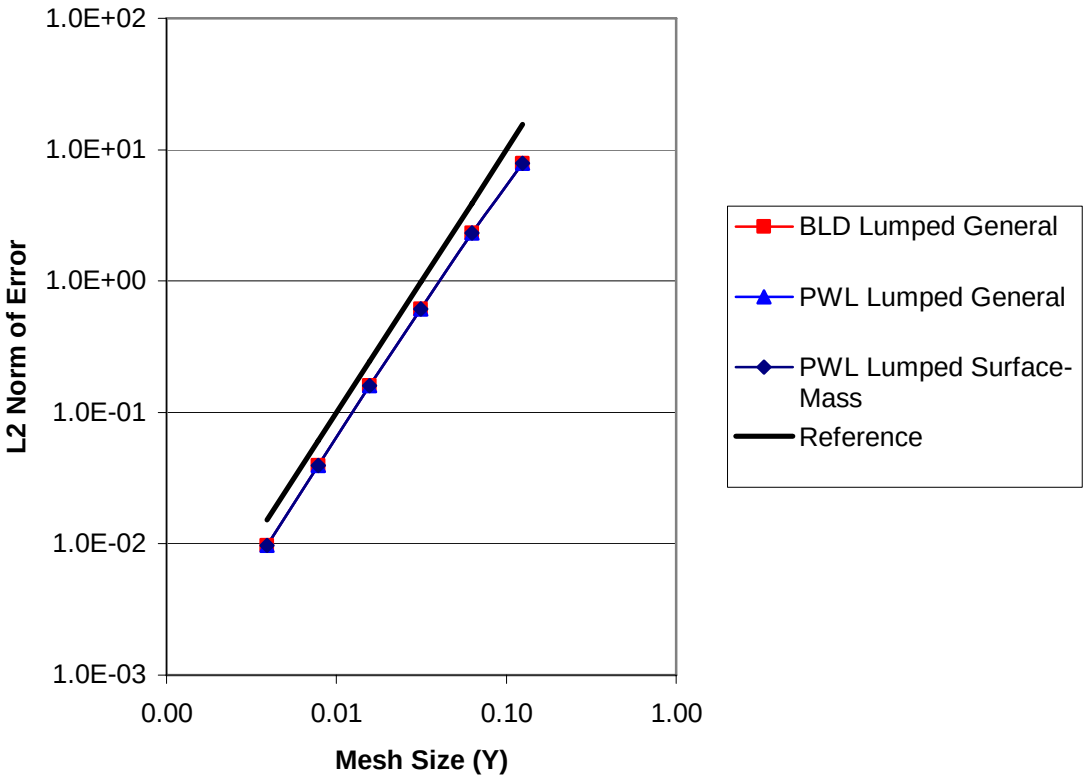

Figure 4.10: Truncation error of lumped methods on a diffusive test problem

This problem not only reinforces the results of our diffusion limit analysis, it also underscores the simplicity of adding the PWLD method to a mature code that contains the LD method. In order for a transport method to converge to a solution in the diffusion limit, it must be accelerated. When we implemented the PWLD method, we seamlessly utilized existing code. In this case, we were able to use the Diffusion Synthetic Acceleration (DSA) scheme that was implemented for BLD on our PWLD method. Recently it has been shown that using DSA as a preconditioner for a GMRES iterative solution of the within group scattering problem can produce rapid and robust convergence even without a "consistent" derivation [19, 20]. Our successful use of the existing diffusion discretization in Capsaicin adds further evidence to this effect. This is also further evidence of PWLD's close relationship to LD and BLD and its similar robustness and accuracy.

Summary

This chapter includes a variety of topics. We begin the chapter with a brief description of the unique features of the discrete ordinates angular discretization for the RZ transport equation. We then apply a general discontinuous finite element method to the RZ transport equation. Next, we derive our RZ piecewise linear basis functions and apply them to the DFEM formulation, noting the close relationship between PWLD and LD. At this point, we introduce the Capsaicin code which we implemented our PWLD method in.

The next section of the chapter is an asymptotic analysis of the RZ PWLD method. This analysis results in a valid diffusion discretization that is satisfied by the leading-order flux. The analysis of the boundary cells results in a normal term, which is a good approximation to the analytic boundary condition, and a tangential term which acts as a contamination term. If the boundary cell is orthogonal, this tangential term immediately disappears from the interior solution. As a result, the solution in the boundary cell is incorrect, but the solution in the interior of the problem is not affected. If cells are non-

orthogonal on the boundary, we anticipate that this contamination term will propagate into the interior, but will eventually cancel itself out. Adams, Wareing, and Walters have shown this for similar problems [13].

We conclude the chapter with a discussion of test problems we use to characterize the PWLD method. To determine the accuracy of PWLD, we run a series of truncation error test problems for the lumped and unlumped method and compare it to BLD on quadrilateral meshes, which we consider to be a standard by which to judge PWLD's accuracy and robustness. The result of these truncation error problems is that PWLD is a second-order accurate method with the same magnitude of error as BLD in problems in the thin limit and the thick diffusive limit. Furthermore, we note that PWLD is able to utilize the same acceleration scheme as BLD without modification indicating that PWLD is robust and easy to implement in existing code. We have proven through the development, analysis, and testing of the PWLD method in RZ that PWLD is an accurate and robust method that has great potential for radiative-transfer problems in RZ geometry.

CHAPTER V
PWLD IN XYZ GEOMETRY

In this chapter we derive the Piecewise Linear Discontinuous Finite Element method for the three-dimensional, Cartesian geometry transport equation. We develop a general discontinuous finite element discretization, and then describe the piecewise linear basis functions we will apply to the general method. After we have created the PWLD method for the three-dimensional system, we discuss how to solve the integrals in the method through standard finite element mapping methods. After we have fully described our unlumped method, we provide a brief description of how to lump the method, motivated by the diffusion-limit analysis of the RZ transport discretization. We describe a fully lumped discretization and a lumping-parameter discretization, noting benefits and detriments of each. We conclude the chapter with a thick diffusion limit analysis in which we show, in great detail, the conditions under which the PWLD method satisfies this limit. We include interior cells and boundary cells in our analysis.

The three-dimensional, XYZ transport equation

We begin our derivation of the PWLD method applied to the three-dimensional transport equation in Cartesian geometry by introducing the steady-state, discrete ordinates transport equation for one frequency group.

$$\begin{aligned}\mu_m \frac{\partial}{\partial x}\psi_m(x,y,z)+\eta_m \frac{\partial}{\partial y}\psi_m(x,y,z)+\xi_m \frac{\partial}{\partial z}\psi_m(x,y,z)+\sigma(x,y,z)\psi_m(x,y,z)\\ =\sum_{l=0}^{L}\sum_{k=-l}^{l}\frac{2l+1}{4\pi}\sigma_{s,l}(x,y,z)\phi_{l,k}(x,y,z)Y_l^k(\Omega)+\frac{1}{4\pi}S(x,y,z)\end{aligned} \tag{5.1}$$

where

$$\phi_{l,k}(x,y,z)=\sum_m w_m\psi_m(x,y,z)Y_{l,k}^*(\Omega_m). \tag{5.2}$$

In Eqs. (5.1) and (5.2) we define ψ as the unknown angular intensity, σ as the macroscopic total opacity, σ_s as the macroscopic scattering opacity, and $\vec{\Omega}$ as the direction in three dimensional space in which the particle is traveling, where

$$\begin{aligned} \mu &= \vec{\Omega} \bullet \vec{e}_x \\ \eta &= \vec{\Omega} \bullet \vec{e}_y \\ \xi &= \vec{\Omega} \bullet \vec{e}_z \end{aligned} \tag{5.3}$$

Unlike the RZ case, the 3D Cartesian equation does not contain angular derivative terms, so applying the discrete ordinates angular discretization is simpler here than in RZ. The discrete ordinates angular discretization is described in detail in Chapter III.

For simplicity, we rewrite Eq. (5.1) as

$$\vec{\Omega}_m \bullet \vec{\nabla} \psi_m (x, y, z) + \sigma (x, y, z) \psi_m (x, y, z) = Q_m (x, y, z), \tag{5.4}$$

where the quadrature direction is denoted by $\vec{\Omega}_m$, and the scattering and external source terms have been combined into $Q_m (x, y, z)$.

Now we apply a general discontinuous finite element method to Eq. (5.4). First, we multiply the equation by a weight function, and integrate over the cell in the mesh where the weight function is non-zero.

$$\int_{V_{cell}} v_i \left[\vec{\Omega}_m \bullet \vec{\nabla} \psi_m (x, y, z) \right] dV + \int_{V_{cell}} v_i \left[\sigma (x, y, z) \psi_m (x, y, z) \right] dV = \int_{V_{cell}} v_i Q_m (x, y, z)\, dV \tag{5.5}$$

We then apply the Gauss divergence theorem to the first term in Eq. (5.5):

$$\int_{\partial V_{cell}} \left(\vec{\Omega}_m \bullet \vec{n}\right) v_i \tilde{\psi}_m \left(x,y,z\right) ds - \int_{V_{cell}} \psi_m \left(x,y,z\right) \left[\vec{\Omega}_m \bullet \vec{\nabla} v_i\right] dV$$
$$+ \int_{V_{cell}} v_i \left[\sigma\left(x,y,z\right)\psi_m \left(x,y,z\right)\right] dV = \int_{V_{cell}} v_i Q_m \left(x,y,z\right) dV, \tag{5.6}$$

where $\tilde{\psi}_m$ denotes an angular intensity unknown on the boundary of the cell. These surface quantities are determined by an upwinding condition.

$$\tilde{\psi}_m = \begin{cases} \psi_{m,cell} \ \ if \ \ \vec{n} \bullet \vec{\Omega}_m > 0 \\ \psi_{m,upwind\ cell} \ \ if \ \ \vec{n} \bullet \vec{\Omega}_m < 0 \end{cases} \tag{5.7}$$

We apply Gauss divergence again to the second term in Eq. (5.6):

$$\int_{\partial V_{cell}} \left(\vec{\Omega}_m \bullet \vec{n}\right) v_i \left[\tilde{\psi}_m \left(x,y,z\right) - \psi_m \left(x,y,z\right)\right] ds + \int_{V_{cell}} v_i \left[\vec{\Omega}_m \bullet \vec{\nabla} \psi_m \left(x,y,z\right)\right] dV$$
$$+ \int_{V_{cell}} v_i \left[\sigma\left(x,y,z\right)\psi_m \left(x,y,z\right)\right] dV = \int_{V_{cell}} v_i Q_m \left(x,y,z\right) dV. \tag{5.8}$$

The solution and the source are approximated as linear combinations of specified basis functions, $\{u_j\}$:

$$\psi_m \left(x,y,z\right) = \sum_{j=1}^{J} \psi_{m,j} u_j \left(x,y,z\right),$$
$$\tilde{\psi}_{m,f} \left(x,y,z\right) = \sum_{j=1}^{J} \tilde{\psi}_{m,j,f} u_j \left(x,y,z\right), \tag{5.9}$$
$$Q_m \left(x,y,z\right) = \sum_{j=1}^{J} Q_{m,j} u_j \left(x,y,z\right),$$

which are substituted into Eq. (5.8).

$$\sum_{f\in cell}\int_{\partial V_f} dA_f\left(\vec{\Omega}_m\bullet\vec{n}_{,f}\right)v_i\left[\left(\sum_{j=1}^{J}\tilde{\psi}_{m,j,f}u_j\left(x,y,z\right)\right)-\left(\sum_{j=1}^{J}\psi_{m,j}u_j\left(x,y,z\right)\right)\right]$$
$$+\int_{V_{cell}} v_i\left[\vec{\Omega}_m\bullet\vec{\nabla}\left(\sum_{j=1}^{J}\psi_{m,j}u_j\left(x,y,z\right)\right)\right]dV \qquad (5.10)$$
$$+\int_{V_{cell}} v_i\left[\sigma\left(x,y,z\right)\sum_{j=1}^{J}\psi_{m,j}u_j\left(x,y,z\right)\right]dV=\int_{V_{cell}} v_i\sum_{j=1}^{J}Q_{m,j}u_j\left(x,y,z\right)dV$$

Eqs. (5.10) and (5.7) represent a general discontinuous finite element spatial discretization applied to the 3D Cartesian transport equation. We denote the terms in the first line of Eq. (5.10) as the surface-integral terms, the second as the within-cell gradient term, the first term of the third line as the collision term, and the right hand side of the equation as the source term. We solve the system of equations generated by multiplying Eq. (5.10) by J distinct weight functions $\{v_i\}$ to produce an approximate solution to the transport equation in the cell. Every cell in our mesh requires its own single-cell inversion, where coupling to other cells occurs through the upwinding condition in Eq. (5.7). In a way, this discontinuous finite element method is a series of one-cell continuous finite element methods whose boundary conditions are determined by the upwinding condition. In order to turn this general DFEM into the Piecewise Linear Discontinuous Finite Element Method (PWLD), we must define three-dimensional PWL basis functions.

PWL basis functions in three dimensions

The three-dimensional PWL basis functions are similar to the ones we built for the RZ discretization in that they build approximations over cells by dividing a cell into a set of subcell volumes called sides and using linear functions on each side. In 3D a side is a tetrahedron that is constructed from two adjacent vertices, a cell center point, and a face center point, where that face contains both vertices. Given this definition for sides, a tetrahedral cell would have 12 sides and a hexahedral cell would have 24 sides. Two examples of sides in hexahedral cells are shown in Figure 5.1 and Figure 5.2. In Figure

5.2, the top face of the hexahedral cell is defined by four non-co-planar vertices. To draw our sides in this cell, we determine a face center point, and facet the face around this face center point, so the face becomes the union of triangular surfaces, and each triangular surface becomes one face of a corresponding tetrahedral side.

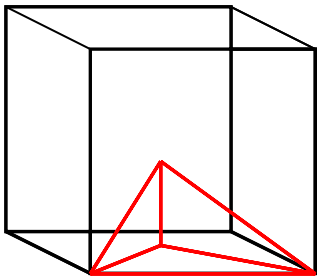

Figure 5.1: Side in an orthogonal hexahedral cell

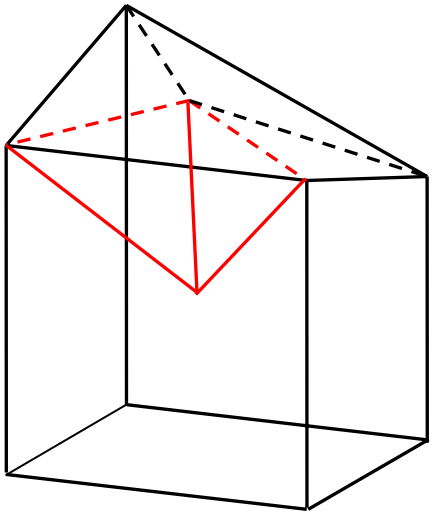

Figure 5.2: Side in a hexahedral cell with a faceted face

However, our treatment of these faceted faces is not yet complete. Recall that in Eq. (5.10), surface integrals are multiplied by the dot product of the quadrature direction,

$\vec{\Omega}_m$, and the outward going normal of a face, $\vec{n}_f$. Furthermore, the sign of this dot product is the quantity by which we determine our upwinding condition. When we facet a face, each facet has its own distinct normal, as shown in Figure 5.3. Non-planar faces can be re-entrant: a given direction could enter the face through one facet and leave through another facet. This behavior is also shown in Figure 5.3 as the dashed red arrow. To treat this situation, we assume an average normal value for the face. We use the face-average normal to determine the upwinding for all sides on the face, but use the individual facelet normals for all geometric calculations.

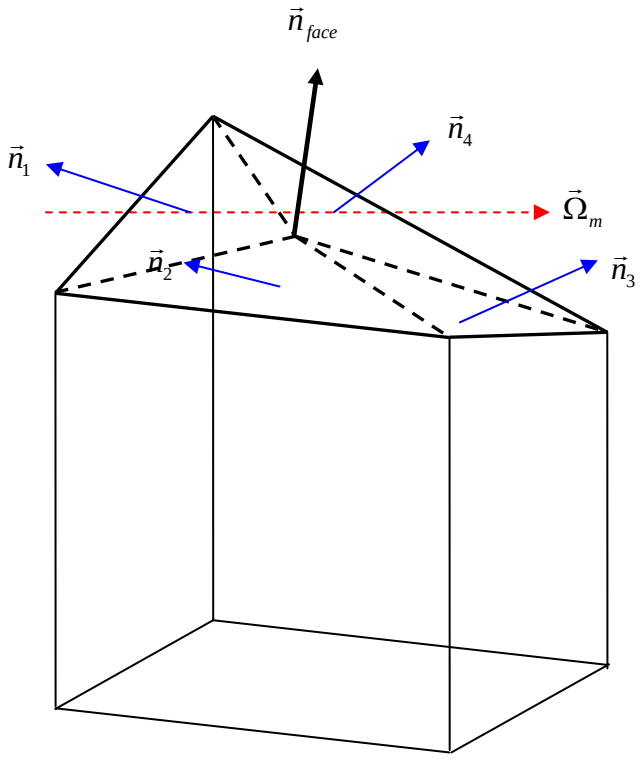

Figure 5.3: Normal vectors for a faceted face

We now define the basis functions as linear combinations of the standard linear functions on tetrahedral sides:

$$u_j(\vec{r}) \;=\; t_j(\vec{r}) \;+\; \sum_{faces\ at\ j} \beta_{f,j} t_f(\vec{r}) \;+\; \alpha_{c,j} t_c(\vec{r})\,. \tag{5.11}$$

where the *t* functions are standard linear functions defined tetrahedron by tetrahedron. For example, t_j equals 1 at the *j*-th vertex and decreases linearly to zero on all other vertices of each side that touches point *j*. t_c is unity at the cell midpoint and zero at each face midpoint and each cell vertex. t_f is unity at the face midpoint and zero at the cell midpoint and at each of the face's vertices. The α_c and β_f are weights that give the cell and face midpoints as weighted averages of their vertices:

$$\vec{r}_c \equiv \text{ cell midpoint} = \sum_{j@c} \alpha_{c,j} \vec{r}_j\,; \tag{5.12}$$

$$\vec{r}_f \equiv \text{ face midpoint} = \sum_{j@f} \beta_{f,j} \vec{r}_j\,. \tag{5.13}$$

For this research we assume that $\alpha_{c,j} = \dfrac{1}{J}$ and $\beta_{f,j} = \dfrac{1}{N_f}$, where *J* is the number of vertices in the cell, and N_f is the number of vertices in a face. As a result, α is the same for every basis function in the cell, and β_f is the same for every basis function on a face. We can now divide the integrals over cells in the general discontinuous finite element method in Eq. (5.10), into sums of integrals over side subcell volumes.

$$\begin{aligned}
&\sum_{\substack{f=face\\ @i}} \sum_{s\in f} \left(\vec{\Omega}_m \bullet \vec{n}\right)_s \int_{\partial V_s} v_i \left[\left(\sum_{j=1}^{J} \tilde{\psi}_{m,f,j} u_j(x,y,z) \right) - \left(\sum_{j=1}^{J} \psi_{m,j} u_j(x,y,z) \right) \right] ds \\
&\qquad + \sum_{s\in cell} \int_{V_s} v_i \left[\vec{\Omega}_m \bullet \vec{\nabla} \left(\sum_{j=1}^{J} \psi_{m,j} u_j(x,y,z) \right) \right] dV \\
&+ \sum_{s\in cell} \int_{V_s} v_i \left[\sigma(x,y,z) \sum_{j=1}^{J} \psi_{m,j} u_j(x,y,z) \right] dV = \sum_{s\in cell} \int_{V_s} v_i \sum_{j=1}^{J} Q_{m,j} u_j(x,y,z) dV
\end{aligned} \tag{5.14}$$

where

$$\tilde{\psi}_{m,f,i} = \begin{cases} \psi_{m,cell,i} \ \ if \ \ \vec{n}_f \bullet \vec{\Omega}_m > 0 \\ \psi_{m,f,upwind\ cell,i} \ \ if \ \ \vec{n}_f \bullet \vec{\Omega}_m < 0 \end{cases}. \tag{5.15}$$

As a result, Eq. (5.14) becomes

$$\begin{aligned} \sum_{\substack{f=face \\ @i}} \sum_{s \in cell} \left(\vec{\Omega}_m \bullet \vec{n}\right)_s \int_{\partial V_s} v_i \left[\begin{array}{l} \left(\sum_{j=1}^{J} \tilde{\psi}_{m,f,j} \left(t_j(\vec{r}) + \sum_{faces\ at\ j} \beta_{f,j} t_f(\vec{r}) + \alpha_{c,j} t_c(\vec{r}) \right) \right) \\ -\left(\sum_{j=1}^{J} \psi_{m,j} \left(t_j(\vec{r}) + \sum_{faces\ at\ j} \beta_{f,j} t_f(\vec{r}) + \alpha_{c,j} t_c(\vec{r}) \right) \right) \end{array} \right] ds \\ + \sum_{s \in cell} \int_{V_s} v_i \left[\vec{\Omega}_m \bullet \vec{\nabla} \left(\sum_{j=1}^{J} \psi_{m,j} \left(t_j(\vec{r}) + \sum_{faces\ at\ j} \beta_{f,j} t_f(\vec{r}) + \alpha_{c,j} t_c(\vec{r}) \right) \right) \right] dV \\ + \sum_{s \in cell} \int_{V_s} v_i \left[\sigma(x,y,z) \sum_{j=1}^{J} \psi_{m,j} \left(t_j(\vec{r}) + \sum_{faces\ at\ j} \beta_{f,j} t_f(\vec{r}) + \alpha_{c,j} t_c(\vec{r}) \right) \right] dV \\ = \sum_{s \in cell} \int_{V_s} v_i \sum_{j=1}^{J} Q_{m,j} \left(t_j(\vec{r}) + \sum_{faces\ at\ j} \beta_{f,j} t_f(\vec{r}) + \alpha_{c,j} t_c(\vec{r}) \right) dV. \end{aligned} \tag{5.16}$$

Integration of standard linear functions on tetrahedra

The integrals of the standard linear functions over side volumes can be solved using a reference side. This reference side, shown in Figure 5.4, is the tetrahedron defined by four points: the first vertex n_1 = (0,0,0), the second vertex n_2 = (2,0,0), the face center point n_f = (1,1,0), and the zone center point n_c = (1,1,1).

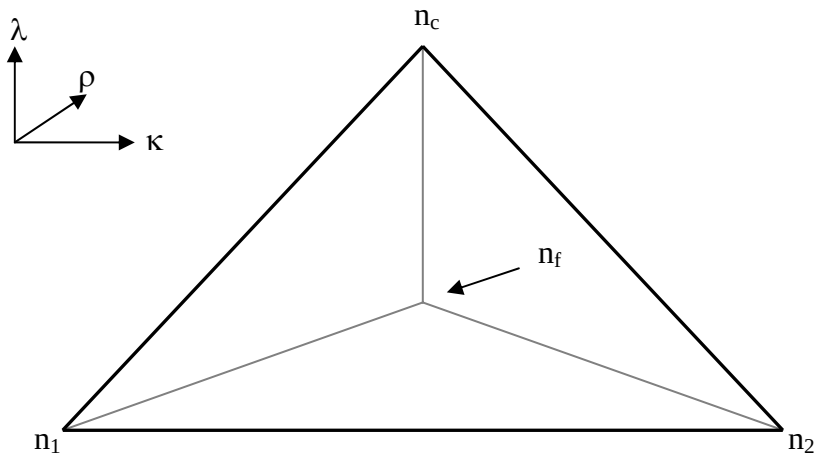

Figure 5.4: A reference tetrahedron

The basis functions defined on this tetrahedron at n_1, n_2, n_f, and n_c are defined as

$$\begin{aligned} t_1(\kappa,\rho,\lambda) &= 1-\frac{1}{2}\kappa-\frac{1}{2}\rho, \\ t_2(\kappa,\rho,\lambda) &= \frac{1}{2}\kappa-\frac{1}{2}\rho, \\ t_f(\kappa,\rho,\lambda) &= \rho-\lambda, \\ t_c(\kappa,\rho,\lambda) &= \lambda. \end{aligned} \tag{5.17}$$

We can now use the same mapping procedure we used in the two dimensional case to solve the integrals. We also assume here that this method is a Galerkin method, so $v_i = u_i$.

$$\begin{aligned} \int_{V_{side}} dV t_i(x,y,z)\,t_j(x,y,z) &= \int_0^{\Delta z} dz \int_{y_{\min}(z)}^{y_{\max}(z)} dy \int_{x_{\min}(y,z)}^{x_{\max}(y,z)} dx t_i(x,y,z)\,t_j(x,y,z) \\ &= \int_0^1 d\lambda \int_0^{1-\lambda} d\rho \int_{\rho+\lambda}^{2-\rho-\lambda} d\kappa t_i(\kappa,\rho,\lambda)\,t_j(\kappa,\rho,\lambda)\det(J) \end{aligned} \tag{5.18}$$

where

$$J = \begin{pmatrix} \frac{dx}{d\kappa} & \frac{dx}{d\rho} & \frac{dx}{d\lambda} \\ \frac{dy}{d\kappa} & \frac{dy}{d\rho} & \frac{dy}{d\lambda} \\ \frac{dz}{d\kappa} & \frac{dz}{d\rho} & \frac{dz}{d\lambda} \end{pmatrix}, \tag{5.19}$$

and

$$\begin{aligned} x(\kappa,\rho,\lambda) &= x_1 + \frac{1}{2}(x_2 - x_1)\kappa + \frac{1}{2}(2x_f - x_2 - x_1)\rho + (x_c - x_f)\lambda, \\ y(\kappa,\rho,\lambda) &= y_1 + \frac{1}{2}(y_2 - y_1)\kappa + \frac{1}{2}(2y_f - y_2 - y_1)\rho + (y_c - y_f)\lambda, \\ z(\kappa,\rho,\lambda) &= z_1 + \frac{1}{2}(z_2 - z_1)\kappa + \frac{1}{2}(2z_f - z_2 - z_1)\rho + (z_c - z_f)\lambda. \end{aligned} \tag{5.20}$$

We define the Jacobian as follows:

$$\begin{aligned} \det(J) &= \begin{pmatrix} \frac{1}{2}(x_2 - x_1) & \frac{1}{2}(2x_f - x_2 - x_1) & (x_c - x_f) \\ \frac{1}{2}(y_2 - y_1) & \frac{1}{2}(2y_f - y_2 - y_1) & (y_c - y_f) \\ \frac{1}{2}(z_2 - z_1) & \frac{1}{2}(2z_f - z_2 - z_1) & (z_c - z_f) \end{pmatrix} \\ &= \frac{1}{2}(x_2 - x_1)\left[\frac{1}{2}(2y_f - y_2 - y_1)(z_c - z_f) - (y_c - y_f)\frac{1}{2}(2z_f - z_2 - z_1)\right] \\ &\quad - \frac{1}{2}(2x_f - x_2 - x_1)\left[\frac{1}{2}(y_2 - y_1)(z_c - z_f) - (y_c - y_f)\frac{1}{2}(z_2 - z_1)\right] \\ &\quad + (x_c - x_f)\left[\frac{1}{2}(y_2 - y_1)\frac{1}{2}(2z_f - z_2 - z_1) - \frac{1}{2}(2y_f - y_2 - y_1)\frac{1}{2}(z_2 - z_1)\right] \\ &= 3V_{side}. \end{aligned} \tag{5.21}$$

The integrals we need to solve on a side are

$$\int_{\partial V_s} t_i(x,y,z)\,t_j(x,y,z)\,ds,$$

$$\int_{V_s} t_i(x,y,z)\left[\vec{\nabla} t_j(x,y,z)\right] dV \xrightarrow[\text{functions}]{\text{linear basis}} \vec{\nabla} t_j(x,y,z)\int_{V_s} t_i(x,y,z)\,dV, \qquad (5.22)$$

$$\int_{V_s} t_i(x,y,z)\,t_j(x,y,z)\,dV.$$

The first set of integrals in Eq. (5.22) are the same as the two dimensional Cartesian mass matrix integrals because the surface of the three dimensional sides are the two dimensional triangular planes of the faceted faces. In matrix form, these integrals are

$$\frac{\vec{A}_{facet}}{12}\begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}, \qquad (5.23)$$

where $\vec{A}_{facet}$ is the surface area normal given by

$$\vec{A}_{facet} = \vec{n}_{s_f} A_{facet}\,. \qquad (5.24)$$

The second and third integrals in Eq. (5.22) can be solved by mapping the weight and basis functions to the reference element.

$$\begin{aligned} \vec{\nabla} t_j(x,y,z)\int_{V_s} t_i(x,y,z)\,dV &= \vec{\nabla} t_j(\kappa,\rho,\lambda)\int_0^1 d\lambda \int_0^{1-\lambda} d\rho \int_{\rho+\lambda}^{2-\rho-\lambda} d\kappa\, t_i(\kappa,\rho,\lambda)\det(J) \\ \int_{V_s} t_i(x,y,z)\,t_j(x,y,z)\,dV &= \int_0^1 d\lambda \int_0^{1-\lambda} d\rho \int_{\rho+\lambda}^{2-\rho-\lambda} d\kappa\, t_i(\kappa,\rho,\lambda)\,t_j(\kappa,\rho,\lambda)\det(J) \end{aligned} \qquad (5.25)$$

We can now write the side based matrices for these integrals by noting that

$$\vec{\nabla} t_j(\kappa,\rho,\lambda)\int_0^1 d\lambda \int_0^{1-\lambda} d\rho \int_{\rho+\lambda}^{2-\rho-\lambda} d\kappa t_i(\kappa,\rho,\lambda)\det(J) = \frac{-\vec{A}_j}{3V_{side}}\left(\frac{1}{12}\right)3V_{side},$$

$$\int_0^1 d\lambda \int_0^{1-\lambda} d\rho \int_{\rho+\lambda}^{2-\rho-\lambda} d\kappa t_i(\kappa,\rho,\lambda) t_j(\kappa,\rho,\lambda)\det(J) = \begin{Bmatrix} \frac{1}{30}3V_{side} & i=j \\ \frac{1}{60}3V_{side} & i\neq j \end{Bmatrix}. \qquad (5.26)$$

These 4x4 matrices for integrals on a side become

$$-\frac{1}{12}\begin{bmatrix} \vec{A}_1 & \vec{A}_2 & \vec{A}_f & \vec{A}_c \\ \vec{A}_1 & \vec{A}_2 & \vec{A}_f & \vec{A}_c \\ \vec{A}_1 & \vec{A}_2 & \vec{A}_f & \vec{A}_c \\ \vec{A}_1 & \vec{A}_2 & \vec{A}_f & \vec{A}_c \end{bmatrix} \qquad (5.27)$$

for the within-cell-gradient term, and

$$\frac{1}{20}V_{side}\begin{bmatrix} 2 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 1 & 1 & 2 \end{bmatrix} \qquad (5.28)$$

for the collision term. The first row of these matrices corresponds to the integrals for weight functions containing t_1, the second row corresponds to weight functions containing t_2, the third row corresponds to weight functions containing t_f, and the fourth row corresponds to weight functions containing t_c. To form our single-cell matrix we loop over sides in a cell, form the individual side matrices for the surface integral, within-cell gradient, and collision terms, and map the elements of these side matrices to their appropriate global cell matrix position. If necessary, the elements are weighted by the appropriate α_c and β_f coefficients. This is analogous to how we developed single-cell matrix for our two-dimensional methods. We solve the resulting matrix equation using Gauss elimination. Based on this derivation of the PWLD method, we can say that it will never be as computationally efficient as a Tri-Linear Discontinuous Finite

Element Method (TRILD) on hexahedral cells because PWLD requires a loop over sides, and in three dimensions the number of sides in cells is not trivial. We further note that due to the matrix solution step in each cell, for each quadrature direction, we expect the PWLD method to be more computationally expensive for arbitrary polyhedral grids than the Upstream Corner Balance (UCB) method, which solves for the unknowns in a cell one at a time by sweeping through subcell volumes in the cell. Pre-storage of matrix elements for each cell has the potential to remove this computational premium, but we have not explored this option.

Lumping the three-dimensional PWLD method

Because the asymptotic analysis and diffusion-limit test problems of the two-dimensional RZ PWLD method motivated the use of lumped methods, we develop lumped methods for the three-dimensional discretization. Developing a fully lumped method is straightforward: we lump the surface integral terms and the collision terms. In order to lump the surface integral and collision terms, we assume that the expansion for the angular intensity unknown on the surface is lumped into one vertex, the vertex associated with the weight function with which we are weighting the transport equation.

$$\begin{aligned} \tilde{\psi}_{m,f}\left(x,y,z\right) &= \sum_{j=1}^{J} \tilde{\psi}_{m,f,j} u_j\left(x,y,z\right) \xrightarrow{lumped} \tilde{\psi}_{m,f,i}, \\ \psi_m\left(x,y,z\right) &= \sum_{j=1}^{J} \psi_{m,j} u_j\left(x,y,z\right) \xrightarrow{lumped} \psi_{m,i}. \end{aligned} \tag{5.29}$$

This approximation appears to be less accurate than the unlumped approximation, but we develop lumped methods because previous experience with lumped methods in the diffusion limit has indicated that they can be more accurate and less prone to unphysical oscillations.

We then substitute this new approximation for the intensity on the surface into the integral equation, Eq. (5.10) resulting in

$$\sum_{f\in cell}\int_{\partial V_f}\left(\vec{\Omega}_m\bullet\vec{n}_f\right)v_i\left(\tilde{\psi}_{m,f,i}-\psi_{m,i}\right)dA_f$$
$$+\int_{V_{cell}}v_i\left[\vec{\Omega}_m\bullet\vec{\nabla}\left(\sum_{j=1}^{J}\psi_{m,j}u_j\left(x,y,z\right)\right)\right]dV \qquad (5.30)$$
$$+\int_{V_{cell}}v_i\left[\sigma\left(x,y,z\right)\psi_{m,i}\right]dV=\int_{V_{cell}}v_iQ_{m,i}dV.$$

We then divide the integrals into sums of integrals on sides resulting in

$$\sum_{\substack{f=face\\ in\ cell}}\sum_{s\in cell}\left(\vec{\Omega}_m\bullet\vec{n}_f\right)_s\left(\tilde{\psi}_{m,f,i}-\psi_{m,i}\right)\int_{\partial V_s}v_idA_f$$
$$+\sum_{s\in cell}\int_{V_s}v_i\left[\vec{\Omega}_m\bullet\vec{\nabla}\left(\sum_{j=1}^{J}\psi_{m,j}u_j\left(x,y,z\right)\right)\right]dV \qquad (5.31)$$
$$+\sum_{s\in cell}\psi_{m,i}\int_{V_s}\sigma\left(x,y,z\right)v_idV=\sum_{s\in cell}^{S}Q_{m,i}\int_{V_s}v_idV.$$

As a result of this lumping, our lumped surface integral matrix for a side is

$$\frac{\vec{A}_{facet}}{3}\begin{pmatrix}1&0&0\\0&1&0\\0&0&1\end{pmatrix}, \qquad (5.32)$$

and our lumped collision term for a side becomes

$$\frac{1}{4}V_{side}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}. \qquad (5.33)$$

Lumping the collision term is often referred to mass matrix lumping. The process for forming the single-cell PWLD matrix is the same as before, but we use these new definitions for the surface integral matrix and the collision term matrix. Also, because our mass matrix is diagonal on each side, we do not have to do as much mapping from

the side matrices to the global matrices. As a result, the lumped method will be slightly faster than the unlumped method. However, we expect it to lack the accuracy of the unlumped method in the optically thin limit.

Because we have lumped the method in anticipation of more robust behavior in the diffusion limit, but we know that the lumping may add a source of error in general, we develop a method that becomes lumped in the thick diffusion limit and unlumped in the thin limit. This method is the lumping-parameter method. A traditional lumping parameter is defined as a factor that ranges between three for optically thin problems, and one for optically thick problem. We define a lumping parameter for the PWLD method on an edge-by-edge basis as suggested by Stone and Adams [4].

$$\theta_e = \frac{3+\left(\sigma l_e\right)^2}{1+\left(\sigma l_e\right)^2}, \tag{5.34}$$

where l_e is the length of edge e. This lumping parameter is defined for each edge in a cell, where an edge contains a vertex i and a vertex $i+1$. In a hexahedral cell, each vertex will be associated with three lumping parameters. The θ parameter ranges from 1 (optically thick) to 3 (optically thin). We find it more convenient to define a parameter that ranges from 0 (thick) to 1 (thin):

$$\lambda_e = \left[3\frac{\theta_e - 1}{2\theta_e}\right] = \frac{1}{1+\frac{1}{3}\left(\sigma l_e\right)^2}. \tag{5.35}$$

In order to introduce these lumping parameters, we begin with the unlumped mass matrix and manipulate it slightly.

$$\begin{aligned}
\int_{V_{cell}} dV v_i \sigma \left(\sum_{j=1}^{J} \psi_j u_j \right) &= \int_{V_{cell}} dV v_i \sigma \left(\sum_{j=1}^{J} \left(\psi_i + \psi_j - \psi_i \right) u_j \left(\vec{r} \right) \right) \\
&= \int_{V_{cell}} dV v_i \sigma \sum_{j=1}^{J} \psi_i u_j \left(\vec{r} \right) + \int_{V_{cell}} dV v_i \sigma \sum_{j=1}^{J} \left(\psi_j - \psi_i \right) u_j \left(\vec{r} \right) \\
&= \int_{V_{cell}} dV v_i \sigma \psi_i \sum_{j=1}^{J} u_j \left(\vec{r} \right) + \int_{V_{cell}} dV v_i \sigma \sum_{j=1}^{J} \left(\psi_j - \psi_i \right) u_j \left(\vec{r} \right) \\
&= \psi_i \int_{V_{cell}} dV \sigma v_i + \sum_{j=1}^{N} \left(\psi_j - \psi_i \right) \int_{V_{cell}} dV \sigma v_i u_j \left(\vec{r} \right)
\end{aligned} \tag{5.36}$$

The first term in the last line of Eq. (5.36) is simply the lumped definition for this term. We can now multiply the second term on the last line of Eq. (5.36) by lumping parameters so that the method will be lumped in the thick limit and unlumped in the thin limit:

$$\begin{aligned}
&\int_{V_{cell}} dV v_i \sigma \left(\sum_{j=1}^{J} \psi_j u_j \right) = \psi_i \int_{V_{cell}} dV \sigma v_i + \sum_{j=1}^{N} \left(\psi_j - \psi_i \right) \int_{V_{cell}} dV \sigma v_i u_j \\
&\xrightarrow[\text{parameter}]{\text{lumping}} \psi_i \int_{V_{cell}} dV \sigma v_i + \sum_{\substack{j=\text{vertex} \\ \text{adjacent to } i}} \lambda_{e_{i,j}} \left(\psi_j - \psi_i \right) \int_{V_{cell}} dV \sigma v_i u_j \\
&\quad + \left(\prod_{\substack{j=\text{vertex} \\ \text{adjacent to } i}} \lambda_{e_{i,j}} \right) \sum_{\substack{j=1, \\ j \neq i, \\ j \text{ not} \\ \text{adjacent to } i}}^{N} \left(\psi_j - \psi_i \right) \int_{V_{cell}} dV \sigma v_i u_j .
\end{aligned} \tag{5.37}$$

When the λ coefficients approach zero, the method becomes fully lumped. When the λ coefficients approach one, the method becomes unlumped. For the lumping parameters defined in Eq.(5.34), the method will approach a fully lumped state when an edge in a cell becomes large compared to a particle mean free path in that cell.

Eq. (5.37) suggests a simple algorithm for building the lumping parameter collision matrix. We first build the unlumped matrix and sum its rows to create the diagonal lumped matrix. We then define a new, lumping-parameter matrix and initialize it to be

the lumped matrix. Then we set the *i*,*j* element, for vertex *j* connected to vertex *i*, to $\lambda_{e_{i,j}} M_{i,j}^{unlumped}$ and subtract this quantity from the diagonal (*i*,*i*). If the *j*th vertex is not connected to vertex *i*, we set the element value to $\left[\prod_{\substack{e=edges\\ connected\\ to\ vertex\ i}} \lambda_e\right] M_{i,j}$ and subtract this quantity from the diagonal element of that row.

Lumping parameters can be applied to the surface integrals in the same way, except we develop an unlumped surface matrix for every face, and then apply the lumping parameter algorithm to each face-surface matrix. For these surface matrices, only two edges can be connected to any vertex making the first sum in Eq. (5.37) to be a sum of only two components.

From this discussion of the lumping parameter method, it is easy to see that we add computational expense with any implementation of the method because we must form the unlumped matrices first and then process them. However, these methods have the potential to provide the accuracy of the unlumped method for cells in the thin limit, the robustness of the lumped method for cells in the thick limit, and a physically dependent mixture of accuracy and robustness between these limits.

The asymptotic diffusion limit analysis for the XYZ method

As we have stated many times in this dissertation, to ensure the accuracy of our methods for radiative-transfer problems, it is necessary that our spatial discretization perform well in the thick diffusion limit. In this limit, the leading-order solution of the analytic transport equation satisfies a diffusion equation with known boundary conditions. As a result, if our method's leading-order solution satisfies an accurate discretization of that diffusion equation and its boundary conditions, our method is accurate in the same diffusion limit. We further note that the diffusion limit we analyze here is for a steady-

state, one-group, uncoupled transport problem. However, as we showed previously, this problem is a special case of the actual radiative-transfer problem, with small specific heat, large time step, and one frequency group.

We begin our analysis by rewriting our surface-lumped and mass-lumped discretization in matrix notation by writing Eq. (5.14) in terms of cell based matrices where these matrices are defined as

$$
\begin{aligned}
\underline{\underline{N}}_f^x &\rightarrow n_{i,j,f}^x = \delta_{i,j} n_{x,f} \sum_{\substack{sides \in \\ face\ f}} \int_{side} dAu_i \\
\underline{\underline{N}}_f^y &\rightarrow n_{i,j,f}^y = \delta_{i,j} n_{y,f} \sum_{\substack{sides \in \\ face\ f}} \int_{side} dAu_i \\
\underline{\underline{N}}_f^z &\rightarrow n_{i,j,f}^z = \delta_{i,j} n_{z,f} \sum_{\substack{sides \in \\ face\ f}} \int_{side} dAu_i \\
\underline{\underline{T}}_f^x &\rightarrow t_{i,j}^x = \delta_{i,j} n_{x,f} \sum_{\substack{sides \in \\ face\ f}} \int_{side} dAu_i \\
\underline{\underline{T}}_f^y &\rightarrow t_{i,j}^y = \delta_{i,j} n_{y,f} \sum_{\substack{sides \in \\ face\ f}} \int_{side} dAu_i \\
\underline{\underline{T}}_f^z &\rightarrow t_{i,j}^z = \delta_{i,j} n_{z,f} \sum_{\substack{sides \in \\ face\ f}} \int_{side} dAu_i \\
\underline{\underline{L}}^x &\rightarrow l_{i,j}^x = \sum_{sides \in cell} \int_{side} dVu_i \frac{\partial \left(u_j \right)}{\partial x} \\
\underline{\underline{L}}^y &\rightarrow l_{i,j}^y = \sum_{sides \in cell} \int_{side} dVu_i \frac{\partial \left(u_j \right)}{\partial y} \\
\underline{\underline{L}}^z &\rightarrow l_{i,j}^z = \sum_{sides \in cell} \int_{side} dVu_i \frac{\partial \left(u_j \right)}{\partial z} \\
\underline{\underline{M}} &\rightarrow m_{i,j} = \delta_{i,j} \sum_{sides \in cell} \int_{side} dVu_i .
\end{aligned}
\tag{5.38}
$$

Writing the integrals in terms of these single cell matrices causes Eq. (5.14)to become

$$\begin{aligned}\sum_{\substack{f=faces\\ \in cell}}\left\{\mu_m \underline{\underline{N}}_f^x\begin{bmatrix}\tilde{\psi}_{m,1,f}\\ \tilde{\psi}_{m,2,f}\\ \vdots\\ \tilde{\psi}_{m,J,f}\end{bmatrix}+\eta_m \underline{\underline{N}}_f^y\begin{bmatrix}\tilde{\psi}_{m,1,f}\\ \tilde{\psi}_{m,2,f}\\ \vdots\\ \tilde{\psi}_{m,J,f}\end{bmatrix}+\xi_m \underline{\underline{N}}_f^z\begin{bmatrix}\tilde{\psi}_{m,1,f}\\ \tilde{\psi}_{m,2,f}\\ \vdots\\ \tilde{\psi}_{m,J,f}\end{bmatrix}\right\}\\ -\sum_{\substack{f=faces\\ \in cell}}\left\{\mu_m \underline{\underline{T}}_f^x\vec{\psi}_m+\eta_m \underline{\underline{T}}_f^y\vec{\psi}_m+\xi_m \underline{\underline{T}}_f^z\vec{\psi}_m\right\}\\ +\mu_m \underline{\underline{L}}^x\vec{\psi}_m+\eta_m \underline{\underline{L}}^y\vec{\psi}_m+\xi_m \underline{\underline{L}}^z\vec{\psi}_m+\sigma\ \underline{\underline{M}}\ \vec{\psi}_m\\ =\frac{1}{4\pi}\underline{\underline{M}}\ \left\{\left(\sigma\ -\sigma_a\right)\vec{\phi}\ +\vec{S}\right\},\end{aligned} \tag{5.39}$$

where $\tilde{\psi}_{m,1,f}$ indicates an angular intensity on face f at the first support point. All other angular intensity unknowns live inside the cell volume. The angular intensity vector on the surface is determined by the upwinding condition given in Eq.(5.15).

Note that the *N* matrix equals the *T* matrix. We write these matrices separately because the *N* matrix is used for the angular intensities on the surface, and the *T* matrix, created when we apply Gauss divergence a second time, is used for the angular intensities inside the cell.

We have now defined our problem such that we can apply the asymptotic diffusion limit scaling for the interior cells in the discretization. To do this we follow procedures defined by Palmer [21] and Adams [6]. We scale Eq. (5.39) to be diffusive, where

$$\begin{aligned}\sigma &\to \frac{\sigma}{\varepsilon}\\ \sigma_a &\to \varepsilon\sigma_a\\ S &\to \varepsilon S.\end{aligned} \tag{5.40}$$

We substitute this scaling into Eq. (5.39) resulting in

$$\sum_{\substack{f=faces\\ \in cell}} \left\{ \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f} \\ \tilde{\psi}_{m,2,f} \\ \vdots \\ \tilde{\psi}_{m,J,f} \end{bmatrix} + \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f} \\ \tilde{\psi}_{m,2,f} \\ \vdots \\ \tilde{\psi}_{m,J,f} \end{bmatrix} + \xi_m \underline{\underline{N}}_f^z \begin{bmatrix} \tilde{\psi}_{m,1,f} \\ \tilde{\psi}_{m,2,f} \\ \vdots \\ \tilde{\psi}_{m,J,f} \end{bmatrix} \right\}$$
$$- \sum_{\substack{f=faces\\ \in cell}} \left\{ \mu_m \underline{\underline{T}}_f^x \vec{\psi}_m + \eta_m \underline{\underline{T}}_f^y \vec{\psi}_m + \xi_m \underline{\underline{T}}_f^z \vec{\psi}_m \right\} \tag{5.41}$$
$$+ \mu_m \underline{\underline{L}}^x \vec{\psi}_m + \eta_m \underline{\underline{L}}^y \vec{\psi}_m + \xi_m \underline{\underline{L}}^z \vec{\psi}_m + \frac{\sigma}{\varepsilon} \underline{\underline{M}} \vec{\psi}_m$$
$$= \frac{1}{4\pi} \underline{\underline{M}} \left\{ \left(\frac{\sigma}{\varepsilon} - \varepsilon\sigma_a \right) \vec{\phi} + \varepsilon \vec{S} \right\}.$$

We define the scalar intensity and current in terms of the quadrature set as

$$\begin{aligned} \phi &= \sum_m w_m \psi_m , \\ J_x \hat{e}_x + J_y \hat{e}_y + J_z \hat{e}_z &= \sum_m w_m \left(\mu_m \hat{e}_x + \eta_m \hat{e}_y + \xi_m \hat{e}_z \right) \psi_m , \end{aligned} \tag{5.42}$$

where

$$\sum_m w_m = 4\pi . \tag{5.43}$$

We guess that the intensities can be expanded in a power series of ε,

$$\begin{aligned} \psi &= \psi^{(0)} + \varepsilon\psi^{(1)} + \varepsilon^2\psi^{(2)} \ldots \\ \phi &= \phi^{(0)} + \varepsilon\phi^{(1)} + \varepsilon^2\phi^{(2)} \ldots \end{aligned} \tag{5.44}$$

and substituted into Eq. (5.41).

$$\sum_{\substack{f=faces\\ \in cell}} \left\{ \begin{array}{c} \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} + \varepsilon\tilde{\psi}_{m,1,f}^{(1)} \cdots \\ \tilde{\psi}_{m,2,f}^{(0)} + \varepsilon\tilde{\psi}_{m,2,f}^{(1)} \cdots \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} + \varepsilon\tilde{\psi}_{m,J,f}^{(1)} \cdots \end{bmatrix} + \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} + \varepsilon\tilde{\psi}_{m,1,f}^{(1)} \cdots \\ \tilde{\psi}_{m,2,f}^{(0)} + \varepsilon\tilde{\psi}_{m,2,f}^{(1)} \cdots \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} + \varepsilon\tilde{\psi}_{m,J,f}^{(1)} \cdots \end{bmatrix} \\ + \xi_m \underline{\underline{N}}_f \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} + \varepsilon\tilde{\psi}_{m,1,f}^{(1)} \cdots \\ \tilde{\psi}_{m,2,f}^{(0)} + \varepsilon\tilde{\psi}_{m,2,f}^{(1)} \cdots \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} + \varepsilon\tilde{\psi}_{m,J,f}^{(1)} \cdots \end{bmatrix} \end{array} \right\}$$

$$- \sum_{\substack{f=faces\\ \in cell}} \left\{ \mu_m \underline{\underline{T}}_f^x \left(\vec{\psi}_m^{(0)} + \varepsilon\vec{\psi}_m^{(1)} \cdots \right) + \eta_m \underline{\underline{T}}_f^y \left(\vec{\psi}_m^{(0)} + \varepsilon\vec{\psi}_m^{(1)} \cdots \right) + \xi_m \underline{\underline{T}}_f^z \left(\vec{\psi}_m^{(0)} + \varepsilon\vec{\psi}_m^{(1)} \cdots \right) \right\}$$

$$+ \mu_m \underline{\underline{L}}^x \left(\vec{\psi}_m^{(0)} + \varepsilon\vec{\psi}_m^{(1)} \cdots \right) + \eta_m \underline{\underline{L}}^y \left(\vec{\psi}_m^{(0)} + \varepsilon\vec{\psi}_m^{(1)} \cdots \right) + \xi_m \underline{\underline{L}}^z \left(\vec{\psi}_m^{(0)} + \varepsilon\vec{\psi}_m^{(1)} \cdots \right)$$

$$+ \frac{\sigma}{\varepsilon} \underline{\underline{M}} \left(\vec{\psi}_m^{(0)} + \varepsilon\vec{\psi}_m^{(1)} \cdots \right) = \frac{1}{4\pi} \underline{\underline{M}} \left\{ \left(\frac{\sigma}{\varepsilon} - \varepsilon\sigma_a \right) \left(\vec{\phi}^{(0)} + \varepsilon\vec{\phi}^{(1)} \cdots \right) + \varepsilon\vec{S} \right\} \tag{5.45}$$

We then collect the *O(1/ε)* terms in Eq. (5.45), resulting in

$$\sigma \underline{\underline{M}} \vec{\psi}_m^{(0)} = \frac{1}{4\pi} \underline{\underline{M}} \sigma \vec{\phi}^{(0)}. \tag{5.46}$$

Assuming that the mass matrix is invertible, Eq. (5.46) yields the result that the leading order angular intensity is isotropic.

$$\vec{\psi}_m^{(0)} = \frac{1}{4\pi} \vec{\phi}^{(0)} \tag{5.47}$$

Next, we collect the *O(1)* terms in Eq. (5.45):

$$\sum_{\substack{f=faces\\ \in cell}} \left\{ \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} + \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} + \xi_m \underline{\underline{N}}_f^z \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} \right\}$$
$$- \sum_{\substack{f=faces\\ \in cell}} \left\{ \mu_m \underline{\underline{T}}_f^x \vec{\psi}_m^{(0)} + \eta_m \underline{\underline{T}}_f^y \vec{\psi}_m^{(0)} + \xi_m \underline{\underline{T}}_f^z \vec{\psi}_m^{(0)} \right\} \tag{5.48}$$
$$+ \mu_m \underline{\underline{L}}^x \vec{\psi}_m^{(0)} + \eta_m \underline{\underline{L}}^y \vec{\psi}_m^{(0)} + \xi_m \underline{\underline{L}}^z \vec{\psi}_m^{(0)}$$
$$+ \sigma \underline{\underline{M}} \vec{\psi}_m^{(1)} = \frac{1}{4\pi} \underline{\underline{M}} \sigma \vec{\phi}^{(1)}.$$

We multiply Eq. (5.48) by w_m and sum over all m in the quadrature set.

$$\sum_{\substack{f=faces\\ \in cell}} \left\{ \sum_m w_m \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} + \sum_m w_m \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} + \sum_m w_m \xi_m \underline{\underline{N}}_f^z \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} \right\}$$
$$- \sum_{\substack{f=faces\\ \in cell}} \left\{ \sum_m w_m \mu_m \underline{\underline{T}}_f^x \vec{\psi}_m^{(0)} + \sum_m w_m \eta_m \underline{\underline{T}}_f^y \vec{\psi}_m^{(0)} + \sum_m w_m \xi_m \underline{\underline{T}}_f^z \vec{\psi}_m^{(0)} \right\} \tag{5.49}$$
$$+ \sum_m w_m \mu_m \underline{\underline{L}}^x \vec{\psi}_m^{(0)} + \sum_m w_m \eta_m \underline{\underline{L}}^y \vec{\psi}_m^{(0)} + \sum_m w_m \xi_m \underline{\underline{L}}^z \vec{\psi}_m^{(0)}$$
$$+ \sum_m w_m \sigma \underline{\underline{M}} \vec{\psi}_m^{(1)} = \frac{1}{4\pi} \sum_m w_m \underline{\underline{M}} \sigma \vec{\phi}^{(1)}$$

Using the fact that the leading order angular intensity inside the cell is isotropic and therefore not dependent on quadrature direction, Eq. (5.49) becomes

$$\sum_{\substack{f=faces\\ \in cell}} \left\{ \sum_m w_m \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} + \sum_m w_m \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} + \sum_m w_m \xi_m \underline{\underline{N}}_f^z \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} \right\}$$

$$- \sum_{\substack{f=faces\\ \in cell}} \left\{ \underline{\underline{T}}_f^x \frac{\vec{\phi}^{(0)}}{4\pi} \sum_m w_m \mu_m + \underline{\underline{T}}_f^y \frac{\vec{\phi}^{(0)}}{4\pi} \sum_m w_m \eta_m + \underline{\underline{T}}_f^z \frac{\vec{\phi}^{(0)}}{4\pi} \sum_m w_m \xi_m \right\} \tag{5.50}$$

$$+ \underline{\underline{L}}^x \frac{\vec{\phi}^{(0)}}{4\pi} \sum_m w_m \mu_m + \underline{\underline{L}}^y \frac{\vec{\phi}^{(0)}}{4\pi} \sum_m w_m \eta_m + \underline{\underline{L}}^z \frac{\vec{\phi}^{(0)}}{4\pi} \sum_m w_m \xi_m \vec{\psi}_m^{(0)}$$

$$+ \sigma \underline{\underline{M}} \sum_m w_m \vec{\psi}_m^{(1)} - \frac{\sigma}{4\pi} \underline{\underline{M}} \vec{\phi}^{(1)} \sum_m w_m = 0.$$

We can further simplify Eq. (5.50) by using the definition of scalar intensity found in Eq. (5.42) and the fact that the sum of the quadrature weights is 4π, causing last two terms in Eq. (5.50) to cancel. Also, we define our angular quadrature set to correctly integrate linear functions of the cosines, which it does if it is symmetric. As a result everything in Eq. (5.50) is zero except the first three terms.

$$\sum_{faces \in cell} \left\{ \sum_m w_m \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} + \sum_m w_m \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} + \sum_m w_m \xi_m \underline{\underline{N}}_f^z \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(0)} \\ \tilde{\psi}_{m,2,f}^{(0)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(0)} \end{bmatrix} \right\} = 0 \tag{5.51}$$

Eq. (5.51) contains a significant amount of information, and we examine it in detail. We first convert this equation from its matrix form to a form with integrals for one row of the matrix in cell *k*.

$$\sum_{\substack{f=\\ faces\\ \in cell\ k}} \vec{n}_{f,k} \bullet \sum_m w_m \vec{\Omega}_m \tilde{\psi}_{m,j,f}^{(0)} \int_f dA u_i = 0 \tag{5.52}$$

where $\vec{n}_{f,k} = n_{x,f,k}\hat{e}_x + n_{y,f,k}\hat{e}_y + n_{z,f,k}\hat{e}_z$ and $\vec{\Omega}_m = \mu_m\hat{e}_x + \eta_m\hat{e}_y + \xi_{m,n}\hat{e}_z$. We then divide the quadrature sum into a sum over incoming directions and a sum of outgoing directions. We also using the upwinding condition to determine the value of $\tilde{\psi}^{(0)}_{m,j,f}$.

$$\sum_{\substack{f=\\ faces\\ \in cell\ k}} \left\{ \sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m>0} w_m \vec{n}_{f,k}\bullet\vec{\Omega}_m \psi^{(0)}_{m,i} + \sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m \vec{n}_{f,k}\bullet\vec{\Omega}_m \psi^{(0)}_{m,i,f,+} \right\} \int_f dAu_i = 0 \qquad (5.53)$$

where $\psi^{(0)}_{m,i}$ is the angular intensity unknown in cell *k* at vertex *i*, and $\psi^{(0)}_{m,i,f,+}$ is the upwind angular intensity unknown from an adjacent cell on face *f* located at the support point of the weight function, vertex *i*. At this point we define a useful quantity that represents a weighted average of the half range $\vec{n}_{f,k}\bullet\vec{\Omega}_m$ values.

$$\rho_{f,k} = \frac{2\sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m>0} w_m \vec{n}_{f,k}\bullet\vec{\Omega}_m}{\sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m>0} w_m} = \frac{2\sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m \left|\vec{n}_{f,k}\bullet\vec{\Omega}_m\right|}{\sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m} \approx 1 \qquad (5.54)$$

For interior faces, we can use the previous result that the leading-order angular flux is isotropic and Eq. (4.76) to simplify Eq.(5.53):

$$\sum_{\substack{f=\\ faces\\ \in cell\ k}} \rho_{f,k}\left(\phi^{(0)}_i - \phi^{(0)}_{i,f,+}\right)\int_f dAu_i = 0\,. \qquad (5.55)$$

In Eq. (5.55) we have multiplied the equation by 4π and noted that

$$\sum_{\substack{m:\\ \vec{n}_{f,k}\bullet\vec{\Omega}_m>0}} w_m \vec{n}_{f,k}\bullet\vec{\Omega}_m = -\sum_{\substack{m:\\ \vec{n}_{f,k}\bullet\vec{\Omega}_m<0}} w_m \vec{n}_{f,k}\bullet\vec{\Omega}_m\,. \qquad (5.56)$$

Eq. (5.55) is true for all *i* such that the vertex *i* is in the interior of the problem. We can generalize Eq.(5.55) to include faces and vertices on the boundary of the problem domain by defining a boundary scalar flux in terms of the incident angular flux.

$$\phi_{i,e,+}^{(0)} = \phi_{bdy,i,e}^{(0)} = \frac{2}{\rho_{f,k}} \sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m 2\left|\vec{n}_{f,k}\bullet\vec{\Omega}_m\right|\psi_{inc,m,i,f}^{(0)} \ . \tag{5.57}$$

This relationship defines the boundary scalar intensity such that the incident particle flow rate would be the same if the incident angular flux were isotropic, i.e. if $\psi_{inc,m}^{(0)} = \frac{\phi_{bdy}^{(0)}}{4\pi}$.

As in the two-dimensional case, we write Eq. (5.55) for the all cells surrounding an interior vertex. For a hexahedral grid, each cell surrounding an interior vertex has three faces connected to that vertex. We show this hexahedral case in Figure 5.5.

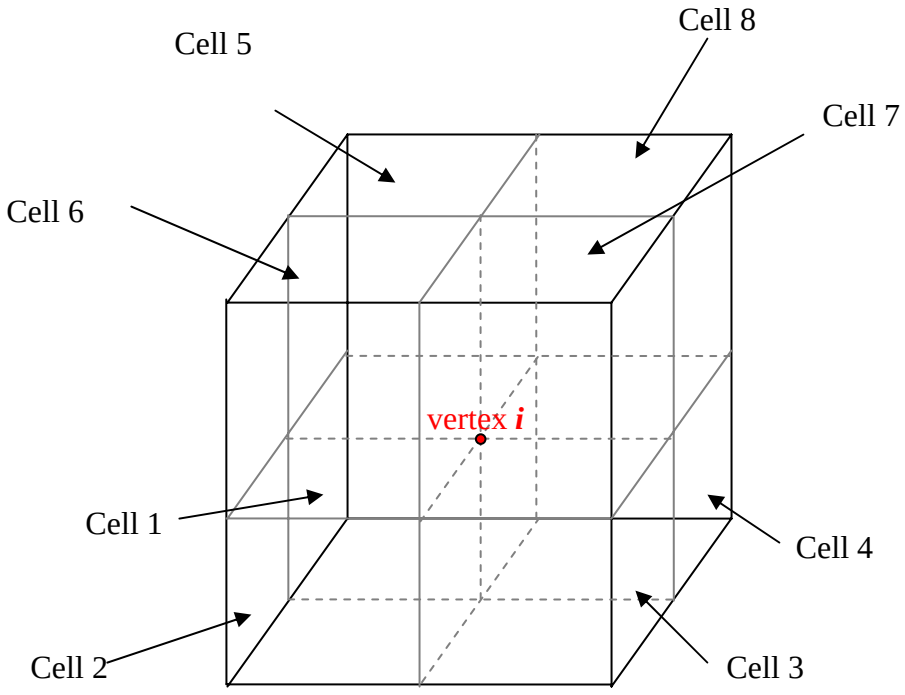

Figure 5.5: Cells surrounding an interior vertex for an orthogonal grid

For the hexahedral case, Eq. (5.55) has three terms for each cell. As a result, we have a system of eight equations and eight unknowns.

$$
\begin{aligned}
&\left(\phi^{(0)}_{i,cell1}-\phi^{(0)}_{i,cell2}\right)I_{face1-2,\,cell1}+\left(\phi^{(0)}_{i,cell1}-\phi^{(0)}_{i,cell3}\right)I_{face1-3,\,cell1}+\left(\phi^{(0)}_{i,cell1}-\phi^{(0)}_{i,cell5}\right)I_{face1-5,\,cell1}=0\\
&\left(\phi^{(0)}_{i,cell2}-\phi^{(0)}_{i,cell3}\right)I_{face2-3,\,cell2}+\left(\phi^{(0)}_{i,cell2}-\phi^{(0)}_{i,cell1}\right)I_{face2-1,\,cell2}+\left(\phi^{(0)}_{i,cell2}-\phi^{(0)}_{i,cell6}\right)I_{face2-6,\,cell2}=0\\
&\left(\phi^{(0)}_{i,cell3}-\phi^{(0)}_{i,cell4}\right)I_{face3-4,\,cell3}+\left(\phi^{(0)}_{i,cell3}-\phi^{(0)}_{i,cell2}\right)I_{face3-2,\,cell3}+\left(\phi^{(0)}_{i,cell3}-\phi^{(0)}_{i,cell7}\right)I_{face3-7,\,cell3}=0\\
&\left(\phi^{(0)}_{i,cell4}-\phi^{(0)}_{i,cell1}\right)I_{face4-1,\,cell4}+\left(\phi^{(0)}_{i,cell4}-\phi^{(0)}_{i,cell3}\right)I_{face4-3,\,cell4}+\left(\phi^{(0)}_{i,cell4}-\phi^{(0)}_{i,cell8}\right)I_{face4-8,\,cell4}=0\\
&\left(\phi^{(0)}_{i,cell5}-\phi^{(0)}_{i,cell6}\right)I_{face5-6,\,cell5}+\left(\phi^{(0)}_{i,cell5}-\phi^{(0)}_{i,cell8}\right)I_{face5-8,\,cell5}+\left(\phi^{(0)}_{i,cell5}-\phi^{(0)}_{i,cell1}\right)I_{face5-1,\,cell5}=0\\
&\left(\phi^{(0)}_{i,cell6}-\phi^{(0)}_{i,cell7}\right)I_{face6-7,\,cell6}+\left(\phi^{(0)}_{i,cell6}-\phi^{(0)}_{i,cell5}\right)I_{face6-5,\,cell6}+\left(\phi^{(0)}_{i,cell6}-\phi^{(0)}_{i,cell2}\right)I_{face6-2,\,cell6}=0\\
&\left(\phi^{(0)}_{i,cell7}-\phi^{(0)}_{i,cell8}\right)I_{face7-8,\,cell7}+\left(\phi^{(0)}_{i,cell7}-\phi^{(0)}_{i,cell6}\right)I_{face7-6,\,cell7}+\left(\phi^{(0)}_{i,cell7}-\phi^{(0)}_{i,cell3}\right)I_{face7-3,\,cell7}=0\\
&\left(\phi^{(0)}_{i,cell8}-\phi^{(0)}_{i,cell5}\right)I_{face8-5,\,cell8}+\left(\phi^{(0)}_{i,cell8}-\phi^{(0)}_{i,cell7}\right)I_{face8-7,\,cell8}+\left(\phi^{(0)}_{i,cell8}-\phi^{(0)}_{i,cell4}\right)I_{face8-4,\,cell8}=0
\end{aligned}
\tag{5.58}
$$

where

$$
I_{face\,k-l,\,cell\,k}=\rho_{face\,k-l,cell\,k}\int_{face\,1-2}dAu_{i,cell\,k} \tag{5.59}
$$

and *face k-l* equals *face l-k*. Like in the two-dimensional case, if this linear system has a unique solution, then the leading-order scalar flux in the interior is completely determined by this system. Again, this solution would be ridiculous because this leading-order scalar flux has no dependence on the sources and cross sections in the problem. We write Eq. (5.58) in matrix form, $Ax=b$, where the A matrix is

$$\begin{bmatrix}
\begin{pmatrix} I^1_{1-2}+I^1_{1-3} \\ +I^1_{1-5} \end{pmatrix} & -I^1_{1-2} & -I^1_{1-3} & 0 & -I^1_{1-5} & 0 & 0 & 0 \\
-I^2_{2-1} & \begin{pmatrix} I^2_{2-1}+I^2_{2-3} \\ +I^2_{2-6} \end{pmatrix} & -I^2_{2-3} & 0 & 0 & -I^2_{2-6} & 0 & 0 \\
0 & -I^3_{3-2} & \begin{pmatrix} I^3_{3-2}+I^3_{3-4} \\ +I^3_{3-7} \end{pmatrix} & -I^3_{3-4} & 0 & 0 & -I^3_{3-7} & 0 \\
-I^4_{4-1} & 0 & -I^4_{4-3} & \begin{pmatrix} I^4_{4-3}+I^4_{4-1} \\ +I^4_{4-8} \end{pmatrix} & 0 & 0 & 0 & -I^4_{4-8} \\
-I^5_{5-1} & 0 & 0 & 0 & \begin{pmatrix} I^5_{5-1}+I^5_{5-6} \\ +I^5_{5-8} \end{pmatrix} & -I^5_{5-6} & 0 & -I^5_{5-8} \\
0 & -I^6_{6-2} & 0 & 0 & -I^6_{6-5} & \begin{pmatrix} I^6_{6-2}+I^6_{6-5} \\ +I^6_{6-7} \end{pmatrix} & 0 & 0 \\
0 & 0 & I^7_{7-3} & 0 & 0 & -I^7_{7-6} & \begin{pmatrix} I^7_{7-3}+I^7_{7-6} \\ +I^7_{7-8} \end{pmatrix} & -I^7_{7-8} \\
0 & 0 & 0 & -I^8_{8-4} & -I^8_{8-5} & 0 & -I^8_{8-7} & \begin{pmatrix} I^8_{8-4}+I^8_{8-5} \\ +I^8_{8-7} \end{pmatrix}
\end{bmatrix} \quad (5.60)$$

and the *b* vector is zero. It is easy to see that this matrix is singular: all row and column sums are equal to zero. This is true as long as the weight functions in the integrals have the surface matching properties along faces that we previously discussed for the two-dimensional case. The PWL weight functions are designed to have this property in three dimensions. Furthermore, the PWL basis functions will have this property for cells with arbitrary numbers of faces. The Tri-linear discontinuous finite element (TRILD) does not have this surface matching property for a cell with an arbitrary number of faces. TRILD's surface matching properties are restricted to hexahedral cells [6].

The next part of the analysis is to determine how many redundant equations exist in the matrix in Eq. (5.60). We can show that there is only one redundant equation using the same logic as the two-dimensional case. We modify our *A* matrix by making the eighth unknown part of the *b* vector. As a result, system becomes

$$\underline{\underline{\tilde{A}}}\vec{\phi} = \vec{\tilde{b}}\phi^{(0)}_{i,cell\,8},$$
$$\vec{\tilde{b}} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ I^4_{4-8} \\ I^5_{5-8} \\ 0 \\ I^7_{7-8} \end{pmatrix}, \tag{5.61}$$

where $\underline{\underline{\tilde{A}}}$ is a 7x7 matrix and $\vec{\phi}$ and $\vec{\tilde{b}}$ are vectors with 7 elements. Also, the $\vec{\tilde{b}}$ vector is no longer zero. We also know that the 7x7 $\underline{\underline{\tilde{A}}}$ is invertible, so we know it has a unique solution. That is, each of the scalar intensities on the left side is a unique multiple of the single scalar intensity on the right side. Therefore, if we can find a solution to this system of equation, we have found the unique solution. This unique solution is

$$\phi^{(0)}_{i,cell\,1} = \phi^{(0)}_{i,cell\,2} = \cdots = \phi^{(0)}_{i,cell\,8}, \tag{5.62}$$

meaning that the leading-order scalar flux is pointwise continuous in three dimensions for a surface lumped method on a hexahedral grid. We remark that this simple result is obtained only with surface lumping, and that it is easy to show that it holds an arbitrary polyhedral grids. The unlumped equations do not produce a pointwise continuous leading-order solution [6]. We now know that we need one more constraint to completely determine our leading-order scalar flux solution.

We proceed with our analysis by multiplying Eq. (5.48) by $w_m\left(\mu_m\hat{e}_x + \eta_m\hat{e}_y + \xi_m\hat{e}_z\right)$ and summing over all points in the quadrature set.

$$\sum_m w_m\left(\mu_m\hat{e}_x+\eta_m\hat{e}_y+\xi_m\hat{e}_z\right)\sum_{\substack{faces\\ \in cell}}\left\{\mu_m\underline{\underline{N}}_f^x\begin{bmatrix}\tilde{\psi}_{m,1,f}^{(0)}\\ \tilde{\psi}_{m,2,f}^{(0)}\\ \vdots\\ \tilde{\psi}_{m,J,f}^{(0)}\end{bmatrix}+\eta_m\underline{\underline{N}}_f^y\begin{bmatrix}\tilde{\psi}_{m,1,f}^{(0)}\\ \tilde{\psi}_{m,2,f}^{(0)}\\ \vdots\\ \tilde{\psi}_{m,J,f}^{(0)}\end{bmatrix}+\xi_m\underline{\underline{N}}_f^z\begin{bmatrix}\tilde{\psi}_{m,1,f}^{(0)}\\ \tilde{\psi}_{m,2,f}^{(0)}\\ \vdots\\ \tilde{\psi}_{m,J,f}^{(0)}\end{bmatrix}\right\}$$

$$-\sum_m w_m\left(\mu_m\hat{e}_x+\eta_m\hat{e}_y+\xi_m\hat{e}_z\right)\sum_{\substack{faces\\ \in cell}}\left\{\mu_m\underline{\underline{T}}_f^x\vec{\psi}_m^{(0)}+\eta_m\underline{\underline{T}}_f^y\vec{\psi}_m^{(0)}+\xi_m\underline{\underline{T}}_f^z\vec{\psi}_m^{(0)}\right\} \tag{5.63}$$

$$+\sum_m w_m\left(\mu_m\hat{e}_x+\eta_m\hat{e}_y+\xi_m\hat{e}_z\right)\left\{\underline{\underline{L}}^x\vec{\psi}_m^{(0)}+\underline{\underline{L}}^y\vec{\psi}_m^{(0)}+\underline{\underline{L}}^z\vec{\psi}_m^{(0)}\right\}$$

$$+\sigma\underline{\underline{M}}\sum_m w_m\left(\mu_m\hat{e}_x+\eta_m\hat{e}_y+\xi_m\hat{e}_z\right)\vec{\psi}_m^{(1)}=\frac{1}{4\pi}\sigma\underline{\underline{M}}\sum_m w_m\left(\mu_m\hat{e}_x+\eta_m\hat{e}_y+\xi_m\hat{e}_z\right)\vec{\phi}^{(1)}$$

Because we have found that in the interior, the leading order angular intensity is continuous across cell boundaries (due to surface lumping) we can say that

$$\begin{bmatrix}\tilde{\psi}_{m,1,f}^{(0)}\\ \tilde{\psi}_{m,2,f}^{(0)}\\ \vdots\\ \tilde{\psi}_{m,J,f}^{(0)}\end{bmatrix}=\vec{\psi}_m^{(0)}, \tag{5.64}$$

for interior cells. Because the surface angular intensity is continuous, we can drop the sum over faces because all of the angular intensities defined at one vertex are equal. We further define

$$\begin{aligned}\underline{\underline{N}}_{fsum}&=\sum_{faces\in cell}\underline{\underline{N}}_f\\ \underline{\underline{T}}_{fsum}&=\sum_{faces\in cell}\underline{\underline{T}}_f.\end{aligned} \tag{5.65}$$

Also note that because the leading order angular intensity is isotropic

$$\sum_m \mu_m^2 w_m \vec{\psi}_m^{(0)} = \sum_m \eta_m^2 w_m \vec{\psi}_m^{(0)} = \sum_m \xi_m^2 w_m \vec{\psi}_m^{(0)} = \frac{4\pi}{3}\frac{\vec{\phi}^{(0)}}{4\pi}$$
$$\sum_m \mu_m^p \eta_m^q \xi_m^r w_m \vec{\psi}_m^{(0)} = 0, \ \begin{Bmatrix} p,q \neq 0 \\ p,r \neq 0 \\ q,r \neq 0 \end{Bmatrix} \tag{5.66}$$

When the relationships in Eqs. (5.64) - (5.66) are substituted into Eq. (5.63), the result is

$$\begin{aligned} &\frac{1}{3}\left\{ \underline{\underline{N}}^x_{fsum} \vec{\phi}^{(0)} \hat{e}_x + \underline{\underline{N}}^y_{fsum} \vec{\phi}^{(0)} \hat{e}_y + \underline{\underline{N}}^z_{fsum} \vec{\phi}^{(0)} \hat{e}_z \right\} \\ &-\frac{1}{3}\left\{ \underline{\underline{T}}^x_{fsum} \vec{\phi}^{(0)} \hat{e}_x - \underline{\underline{T}}^y_{fsum} \vec{\phi}^{(0)} \hat{e}_y - \underline{\underline{T}}^z_{fsum} \vec{\phi}^{(0)} \hat{e}_z \right\} \\ &+\frac{1}{3}\left\{ \underline{\underline{L}}^x \vec{\phi}^{(0)} \hat{e}_x + \underline{\underline{L}}^y \vec{\phi}^{(0)} \hat{e}_y + \underline{\underline{L}}^z \vec{\phi}^{(0)} \hat{e}_z \right\} \\ &+\sigma \underline{\underline{M}} \sum_m w_m \left(\mu_m \hat{e}_x + \eta_m \hat{e}_y + \xi_m \hat{e}_z \right) \left(\vec{\psi}_m^{(1)} \right) \\ &= \frac{1}{4\pi} \sigma \underline{\underline{M}} \vec{\phi}^{(1)} \sum_m w_m \left(\mu_m \hat{e}_x + \eta_m \hat{e}_y + \xi_m \hat{e}_z \right). \end{aligned} \tag{5.67}$$

We can further simplify Eq. (5.67) by using the definition of current in Eq. (5.42), observing that $\underline{\underline{N}}_{fsum} = \underline{\underline{T}}_{fsum}$, and noting that

$$\sum_m w_m \left(\mu_m \hat{e}_x + \eta_m \hat{e}_y + \xi_m \hat{e}_z \right) = 0 , \tag{5.68}$$

resulting in

$$\frac{1}{3}\left\{ \underline{\underline{L}}^x \vec{\phi}^{(0)} \hat{e}_x + \underline{\underline{L}}^y \vec{\phi}^{(0)} \hat{e}_y + \underline{\underline{L}}^z \vec{\phi}^{(0)} \hat{e}_z \right\} + \sigma \underline{\underline{M}} \left(\vec{J}_x^{(1)} \hat{e}_x + \vec{J}_y^{(1)} \hat{e}_y + \vec{J}_z^{(1)} \hat{e}_z \right) = 0 . \tag{5.69}$$

We introduce a compact notation for the current unknowns in the cell. We have a current, which is a vector quantity in *x, y* and *z*, at every vertex in the cell. Previously, we have denoted the list of unknowns at each vertex as a vector. As a result, our current unknown in the cell is a vector of vectors, which we denote

$$\vec{\vec{J}} = \left(\vec{J}_x^{(1)} \hat{e}_x + \vec{J}_y^{(1)} \hat{e}_y + \vec{J}_z^{(1)} \hat{e}_z \right). \tag{5.70}$$

Solving for the *O(1)* current in Eq. (5.69) results in

$$\vec{\vec{J}}^{(1)} = -\frac{1}{3\sigma} \underline{\underline{M}}^{-1} \left\{ \underline{\underline{L}}^x \vec{\phi}^{(0)} \hat{e}_x + \underline{\underline{L}}^y \vec{\phi}^{(0)} \hat{e}_y + \underline{\underline{L}}^z \vec{\phi}^{(0)} \hat{e}_z \right\} \vec{\phi}^{(0)} . \tag{5.71}$$

Eq. (5.71) is a discretized version of Fick's law, where the stencil of the approximation is determined by the stencil of the individual matrices $M^{-1}$and L.

We now collect all $O(\varepsilon)$ terms from Eq. (5.45):

$$\begin{aligned} \sum_{faces \in cell} \left\{ \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} + \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} + \xi_m \underline{\underline{N}}_f^z \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} \right\} \\ - \sum_{faces \in cell} \left\{ \mu_m \underline{\underline{T}}_f^x \vec{\psi}_m^{(1)} + \eta_m \underline{\underline{T}}_f^y \vec{\psi}_m^{(1)} + \xi_m \underline{\underline{T}}_f^z \vec{\psi}_m^{(1)} \right\} \\ + \mu_m \underline{\underline{L}}^x \vec{\psi}_m^{(1)} + \eta_m \underline{\underline{L}}^y \vec{\psi}_m^{(1)} + \xi_m \underline{\underline{L}}^z \vec{\psi}_m^{(1)} \\ + \sigma \underline{\underline{M}} \vec{\psi}_m^{(2)} = \frac{1}{4\pi} \underline{\underline{M}} \sigma \vec{\phi}^{(2)} - \frac{1}{4\pi} \underline{\underline{M}} \sigma_a \vec{\phi}^{(0)} + \frac{1}{4\pi} \underline{\underline{M}} \vec{S}^{(0)}. \end{aligned} \tag{5.72}$$

When Eq. (5.72) is multiplied by w_m and summed over all points in the quadrature set, the result is

$$\sum_{faces\in cell}\sum_{m} w_m \left\{ \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} + \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} + \xi_m \underline{\underline{N}}_f^z \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} \right\}$$
$$- \sum_{faces\in cell}\sum_{m} w_m \left\{ \mu_m \underline{\underline{T}}_f^x \vec{\psi}_m^{(1)} + \eta_m \underline{\underline{T}}_f^y \vec{\psi}_m^{(1)} + \xi_m \underline{\underline{T}}_f^z \vec{\psi}_m^{(1)} \right\}$$
$$+ \sum_{m} w_m \left\{ \mu_m \underline{\underline{L}}^x \vec{\psi}_m^{(1)} + \eta_m \underline{\underline{L}}^y \vec{\psi}_m^{(1)} + \xi_m \underline{\underline{L}}^z \vec{\psi}_m^{(1)} \right\} \tag{5.73}$$
$$+ \underline{\underline{M}} \sigma_a \vec{\phi}^{(0)} = \underline{\underline{M}} \vec{S}^{(0)}.$$

We have used the definition of scalar intensity and the fact that the sum of the quadrature weights is 4π to cancel the second order terms. We can further simplify Eq. (5.73) by substituting the definition for the *T* matrix given in Eq. (5.65).

$$\sum_{faces\in cell}\sum_{m} w_m \left\{ \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} + \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} + \xi_m \underline{\underline{N}}_f^z \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} \right\}$$
$$+ \sum_{m} w_m \left\{ \mu_m \left(\underline{\underline{L}}^x - \underline{\underline{T}}_{fsum}^x \right) + \eta_m \left(\underline{\underline{L}}^y - \underline{\underline{T}}_{fsum}^y \right) + \xi_m \left(\underline{\underline{L}}^z - \underline{\underline{T}}_{fsum}^z \right) \right\} \vec{\psi}_m^{(1)} \tag{5.74}$$
$$+ \underline{\underline{M}} \sigma_a \vec{\phi}^{(0)} = \underline{\underline{M}} \vec{S}^{(0)}$$

We can substitute the definition for current given in Eq. (5.42) into Eq. (5.74), causing it to simplify to

$$\sum_{faces\in cell}\sum_{m} w_m \left\{ \mu_m \underline{\underline{N}}_f^x \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} + \eta_m \underline{\underline{N}}_f^y \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} + \xi_m \underline{\underline{N}}_f^z \begin{bmatrix} \tilde{\psi}_{m,1,f}^{(1)} \\ \tilde{\psi}_{m,2,f}^{(1)} \\ \vdots \\ \tilde{\psi}_{m,J,f}^{(1)} \end{bmatrix} \right\} \tag{5.75}$$
$$- \left\{ \left(\underline{\underline{T}}_{fsum}^x - \underline{\underline{L}}^x \right) \hat{e}_x + \left(\underline{\underline{T}}_{fsum}^y - \underline{\underline{L}}^y \right) \hat{e}_y + \left(\underline{\underline{T}}_{fsum}^z - \underline{\underline{L}}^z \right) \hat{e}_z \right\} \vec{J}^{(1)} + \underline{\underline{M}} \sigma_a \vec{\phi}^{(0)} = \underline{\underline{M}} \vec{S}^{(0)}.$$

Eq. (5.75) is true for every interior cell surrounding a vertex in the mesh. Now we can write Eq. (5.75)for all cells surrounding a given vertex *i* as shown in Figure 5.5. When we sum the *i*th matrix row for each cell around that vertex, the terms with the angular fluxes defined on cell surfaces cancel because each appears with a positive sign in one cell and a negative sign in the neighboring cell resulting in:

$$\begin{aligned}
&\sum_{k=1}^{K}\left(\left(\sum_{j=1}^{J} J_{j,k,x}^{(1)} \sum_{\substack{sides\\ \in cell\ k}} \int_{side} dVu_{i,k} \frac{\partial\left(u_{j,k}\right)}{\partial x}\right) - J_{i,k,x}^{(1)} \sum_{\substack{f_k = \\ faces\\ with\ i}} \left(n_{x,f,k} \int_{f} dAu_{i,f,k}\right)\right) \\
&\sum_{k=1}^{K}\left(\left(\sum_{j=1}^{J} J_{j,k,y}^{(1)} \sum_{\substack{sides\\ \in cell\ k}} \int_{side} dVu_{i,k} \frac{\partial\left(u_{j,k}\right)}{\partial y}\right) - J_{i,k,y}^{(1)} \sum_{\substack{f_k = \\ faces\\ with\ i}} \left(n_{y,f,k} \int_{f} dAu_{i,f,k}\right)\right) \\
&\sum_{k=1}^{K}\left(\left(\sum_{j=1}^{J} J_{j,k,z}^{(1)} \sum_{\substack{sides\\ \in cell\ k}} \int_{side} dVu_{i,k} \frac{\partial\left(u_{j,k}\right)}{\partial z}\right) - J_{i,k,z}^{(1)} \sum_{\substack{f_k = \\ faces\\ with\ i}} \left(n_{z,f,k} \int_{f} dAu_{i,f,k}\right)\right) \\
&+\sum_{k}^{K} \sigma_{a,k} \phi_{i,k}^{(0)} \sum_{\substack{sides\\ \in cell\ k}} \int_{side} dVu_{i,k} = \sum_{k}^{K} S_{i,k}^{(0)} \sum_{\substack{sides\\ \in cell\ k}} \int_{side} dVu_{i,k}.
\end{aligned} \tag{5.76}$$

where the current, $J_{j,k,x}^{(1)}\hat{e}_x + J_{j,k,y}^{(1)}\hat{e}_y + J_{j,k,z}^{(1)}\hat{e}_z$, has previously been defined in Eq. (5.71).

This system of equations (Eqs. (5.76) and (5.71)) results in a fully coupled diffusion discretization for the leading-order intensity solution. As a result the surface lumped PWLD method applied to the 3D Cartesian geometry transport equation yields the correct leading-order scalar flux solution for problems in the thick diffusive limit.

We also note that

$$\underline{\underline{M}}_k^{-1}\left\{\underline{\underline{L}}_k^x \vec{\phi}^{(0)}\hat{e}_x + \underline{\underline{L}}_k^y \vec{\phi}^{(0)}\hat{e}_y + \underline{\underline{L}}_k^z \vec{\phi}^{(0)}\hat{e}_z\right\}\vec{\phi}_k^{(0)} = \underline{\underline{P}}_k \vec{\phi}_k^{(0)} = \vec{\vec{J}}_k^{(1)}, \tag{5.77}$$

where, an element of matrix *P* is defined as:

$$\underline{\underline{P}}_k \rightarrow p_{i,j,k} = \frac{\sum_{\substack{\text{sides in} \\ \text{cell } k}} \vec{\nabla} u_j \int_{\text{side}} dV u_i}{\sum_{\substack{\text{sides in} \\ \text{cell } k}} \int_{\text{side}} dV u_i}. \tag{5.78}$$

If we define a diffusion coefficient to be

$$D_k = \frac{1}{3\sigma_k}, \tag{5.79}$$

the current becomes

$$J_{i,x,k}^{(1)} \hat{e}_x + J_{i,y,k}^{(1)} \hat{e}_y + J_{i,z,k}^{(1)} \hat{e}_z = -D_k \sum_{j=1}^{J} p_{i,j,k} \phi_{j,k}^{(0)} = -D_k \sum_{j=1}^{J} \frac{\left[\sum_{\substack{\text{sides} \in \\ \text{cell} k}} \left(\vec{\nabla} u_j \right) \int_{\text{side}} dV u_i \right] \phi_{j,k}^{(0)}}{\left(\sum_{\substack{\text{sides} \in \\ \text{cell } k}} \int_{\text{side}} dV u_i \right)}. \tag{5.80}$$

Substituting this current into Eq. (5.76) results in

$$\sum_{k=1}^{K}\left(\left(-D_k\sum_{j=1}^{J_k}p_{i,j,k,x}\phi_{j,k}^{(0)}\sum_{\substack{sides\\ \in cell\ k}}\int_{side}dVu_{i,k}\frac{\partial\left(u_{j,k}\right)}{\partial x}\right)+D_k p_{i,i,k,x}\phi_{i,k}^{(0)}\sum_{\substack{f_k=\\ faces\\ with\ i}}\left(n_{x,f,k}\int_f dAu_{i,f,k}\right)\right)$$

$$\sum_{k=1}^{K}\left(\left(-D_k\sum_{j=1}^{J_k}p_{i,j,k,y}\phi_{j,k}^{(0)}\sum_{\substack{sides\\ \in cell\ k}}\int_{side}dVu_{i,k}\frac{\partial\left(u_{j,k}\right)}{\partial y}\right)+D_k p_{i,i,k,y}\phi_{i,k}^{(0)}\sum_{\substack{f_k=\\ faces\\ with\ i}}\left(n_{y,f,k}\int_f dAu_{i,f,k}\right)\right) \qquad (5.81)$$

$$\sum_{k=1}^{K}\left(\left(-D_k\sum_{j=1}^{J_k}p_{i,j,k,z}\phi_{j,k}^{(0)}\sum_{\substack{sides\\ \in cell\ k}}\int_{side}dVu_{i,k}\frac{\partial\left(u_{j,k}\right)}{\partial z}\right)+D_k p_{i,i,k,z}\phi_{i,k}^{(0)}\sum_{\substack{f_k=\\ faces\\ with\ i}}\left(n_{z,f,k}\int_f dAu_{i,f,k}\right)\right)$$

$$+\sum_{k}^{K}\sigma_{a,k}\phi_{i,k}^{(0)}\sum_{\substack{sides\\ \in cell\ k}}\int_{side}dVu_{i,k}=\sum_{k}^{K}S_{i,k}^{(0)}\sum_{\substack{sides\\ \in cell\ k}}\int_{side}dVu_{i,k}.$$

Eq. (5.81) represents a valid diffusion discretization of the leading-order scalar flux with a 27 point stencil on a hexahedral grid. Coupling to non-neighbor vertices occurs only through the within-cell gradient integrals. If we had not lumped the surface matrix, the resultant diffusion discretization from the asymptotic analysis would have resulted in a symmetric positive definite (SPD) coefficient matrix. This equality looks like a possibility if we can figure out a way to "undo" the Gauss-divergence step we performed to generate the within-cell gradient integrals and surface integrals; however, after this second Gauss-divergence step, we lumped the surface integrals so this equality will never be true in general. We also find that mass matrix lumping makes the resultant diffusion discretization more diagonally dominant, and therefore more robust.

Continuous PWL FEM and Palmer's method applied to diffusion

The diffusion equation can be written as

$$-\vec{\nabla}\bullet D\vec{\nabla}\phi+\sigma_a\phi=S\,. \qquad (5.82)$$

To apply the continuous PWL FEM to this equation, we multiply by a weight function, *v*, and integrate over the problem domain.

$$\int_{cells\,at\,i} d^3r\, v_i(r)\left[-\vec{\nabla}\bullet D\vec{\nabla}\phi(\vec{r})+\sigma_a\phi(\vec{r})-S(\vec{r})\right]=0 \tag{5.83}$$

For the *i*-th weight function the integral is non-zero only for cells that contain this weight function, which is why the integral in Eq. (5.83) is an integral over the cells at i. When the divergence theorem is applied to Eq. (5.83) the result is

$$\int_{\partial c\,at\,i} d^2r\, v_i(\vec{r})\vec{n}\bullet\vec{J}+\int_{cells\,at\,i} d^3r\left[D\vec{\nabla}\phi(\vec{r})\bullet\vec{\nabla}v_i(\vec{r})+v_i(\vec{r})\left(\sigma_a\phi(\vec{r})-S(\vec{r})\right)\right]=0\,. \tag{5.84}$$

The first term in Eq. (5.84) is zero for cells in the interior of the problem domain. We now expand the unknown scalar intensity and the source in terms of the basis functions *u*:

$$\begin{aligned}\phi(\vec{r})&=\sum_{all\,j}\phi_j u_j(\vec{r}),\\ S(\vec{r})&=\sum_{all\,j}S_j u_j(\vec{r}),\end{aligned} \tag{5.85}$$

which results in

$$\begin{aligned}\sum_j\phi_j\left\{\int_{cells\,at\,i} d^3r\left[D\vec{\nabla}u_j(\vec{r})\bullet\vec{\nabla}u_i(\vec{r})+\sigma u_j(\vec{r})u_i(\vec{r})\right]\right\}\\ =\sum_j S_j\int_{cells\,at\,i} d^3r\,u_i(\vec{r})u_j(\vec{r}).\end{aligned} \tag{5.86}$$

We have assumed a Galerkin method, which causes the weight and basis functions to be equal. We can now apply the same PWL basis functions, given in Eq.(5.11), to Eq. (5.86) as we used in the transport case [5]. In order to be able easily integrate these functions, we divide the integrals in Eq. (5.86) in sums of integrals over sides, as we did in the transport case.

$$\sum_j \phi_j \left\{ \sum_{\substack{cells \\ at\ i}} \int_{\substack{sides \\ in\ cell}} d^3r \left[D\vec{\nabla} u_j(\vec{r}) \bullet \vec{\nabla} u_i(\vec{r}) + \sigma u_j(\vec{r}) u_i(\vec{r}) \right] \right\} = \sum_j S_j \sum_{\substack{cells \\ at\ i}} \int_{\substack{sides \\ in\ cell}} d^3r\, u_i(\vec{r}) u_j(\vec{r}) \tag{5.87}$$

If we lump the mass matrix the same way we lumped the mass matrix in the transport case, the result is

$$\sum_j \phi_j \left\{ \sum_{\substack{cells \\ at\ i}} \int_{\substack{sides \\ in\ cell}} d^3r \left[D\vec{\nabla} u_j(\vec{r}) \bullet \vec{\nabla} u_i(\vec{r}) \right] \right\} + \phi_i \sum_{\substack{cells \\ at\ i}} \sigma_{a,cell} \int_{\substack{sides \\ in\ cell}} d^3r u_i = S_i \sum_{\substack{cells \\ at\ i}} \sigma_{a,cell} \int_{\substack{sides \\ in\ cell}} d^3r u_i \,. \tag{5.88}$$

Eq. (5.88) shows that the mass matrix part of the continuous PWL FEM applied to the diffusion equation is equivalent to the mass matrix part of the thick diffusive limit of the PWLD method applied to the transport equation. However, it is more difficult to show that the diffusion terms between the two methods are equivalent in any sense. In general, the diffusion term for the thick diffusive limit of the surface lumped PWLD transport method is not SPD, but this same diffusion term in the PWL method applied directly to the diffusion equation is SPD [5].

There is a specific case, however, where the two discretizations become equivalent. This case occurs when we do not lump the surface term in the transport method and the grid is restricted to tetrahedral elements. When we developed a diffusion discretization from the asymptotic analysis, we discretized a particle balance equation and a Fick's law equation separately. Fick's law is

$$\frac{1}{3}\vec{\nabla}\phi + \sigma\vec{J} = 0\,. \tag{5.89}$$

When Eq. (5.89) is discretized using the finite element method, the result is

$$\int d^3 r u_i \left\{ \frac{1}{3}\vec{\nabla}\phi(\vec{r}) + \sigma\vec{J}(\vec{r}) \right\} = 0\,. \tag{5.90}$$

We now expand the scalar intensity and current as

$$\begin{aligned} \vec{\nabla}\phi(\vec{r}) &= \sum_j \phi_j \vec{\nabla} u_j(\vec{r}), \\ \vec{J}(\vec{r}) &= \sum_j J_j u_j(\vec{r}). \end{aligned} \tag{5.91}$$

If $\left\{\vec{\nabla}u(\vec{r})\right\}$ lives in the space spanned by $\left\{u(\vec{r})\right\}$, then the result of the Fick's law discretization is that the current and the gradient of the scalar intensity are pointwise equal. As a result, when $\vec{\nabla}u_j(\vec{r})$ and $u_j(\vec{r})$ are substituted into the finite element discretized balance equation,

$$\int d^3 r \left\{ \vec{\nabla}\bullet\vec{J}(\vec{r}) + \sigma_a\phi(\vec{r}) - S(\vec{r}) \right\} = 0\,, \tag{5.92}$$

the diffusion term discretization becomes an SPD matrix, and this asymptotically developed discretization of the diffusion equation becomes equivalent to a direct discretization of the diffusion equation. The only case, for PWL basis functions, where $\left\{\vec{\nabla}u(\vec{r})\right\}$ lives in the space spanned by $\left\{u(\vec{r})\right\}$ is when the grid is composed of tetrahedral elements. Because PWL basis functions are forced to be linear on each tetrahedral side, for any cell shape other than tetrahedral, the gradients of these basis functions are discontinuous and are not in the space spanned by the basis functions.

Palmer's method can be cast as a Petrov-Galerkin finite element method with PWL basis functions [5, 26, 27]. The weight functions in this Petrov-Galerkin method are defined

to be unity over a dual cell. A dual cell is defined as the union of all wedges surrounding a given vertex. As a result, using Eq. (5.84) as a starting point, Palmer's method can be written as

$$\int_{\partial cell\ at\ i} dA\vec{n}\bullet\vec{J} + \int_{cell\ at\ i} d^3r\left[\left(\sigma\phi\left(\vec{r}\right)-S\left(\vec{r}\right)\right)\right]=0. \tag{5.93}$$

The second term in Eq. (5.84) cancels because the weight functions are constant, so the gradients of weight functions are zero. We can use the definition of the current, *J*, to simplify Eq. (5.93). Also, notice that the integral over the surface of the cells can be reduced to a summation.

$$-\sum_{\substack{Surfaces\\ \in Dual\ cell}} \vec{A}\bullet D\vec{\nabla}u_j\vec{\phi}_j + \int_{cell\ at\ i} d^2r\left[v_i\left(r\right)\left(\sigma\phi\left(r\right)-S\left(r\right)\right)\right]=0. \tag{5.94}$$

Palmer's method also lumps the mass matrix, but does this lumping differently than our lumping for the PWLD method. Palmer's complete lumped discretization is:

$$-\sum_{\substack{Surfaces\\ \in Dual\ Cell}} \vec{A}\bullet D\vec{\nabla}u_j\vec{\phi}_j + \sigma_{cell}\phi_i \sum_{\substack{wedges\\ \in Dual\ Cell}} V_w = S_i \sum_{\substack{wedges\\ \in Dual\ Cell}} V_w . \tag{5.95}$$

The difference between Palmer's method's mass matrix terms and the diffusive limit of PWLD applied to the transport equation is minor. The volume of a wedge equals one half the volume of a side, and the integral of a PWL basis function over a side is a fractional value of that side. Therefore, we can say these two types of mass matrix lumpings will behave similarly, but not exactly the same. Again the main difference between the two methods is found in the diffusion terms. The first property to note about the gradient term in Palmer's method is that it forms a seven point stencil, instead of a full 27-point stencil on an orthogonal brick grid. Furthermore, on a non-orthogonal grid, an SPD discretization is not guaranteed [27]. As a result, because the PWLD asymptotic diffusion discretization is equal to the direct PWL discretization of the

diffusion equation *on a tetrahedral grid*, and this discretization is SPD, the asymptotic PWL discretization is not equivalent to Palmer's discretization on a tetrahedral grid.

The diffusion discretization satisfied by the leading-order scalar flux is a well behaved discretization. This diffusion discretization involves a weighted-integral conservation equation that is identical to the Galerkin PWL continuous conservation equation, with one equation at each vertex. It also develops a relationship for a current in each cell that is a weighted-integral Fick's law relationship with PWL approximations. We consider the combination of these two weighted-integral equations to produce a reasonably accurate diffusion discretization. We will test our assessment of this diffusion discretization in the next chapter with a variety of test problems in the diffusion limit.

Boundary analysis

We know that boundary conditions will affect the interior of the solution, so we have to make sure a method treats incident intensities on the boundary consistently with the analytic behavior of the transport equation for thick diffusive systems. From the asymptotic diffusion limit analysis performed on the analytic transport equation (which is straightforward), we know that the correct boundary scalar intensity that yields the correct leading-order solution in the interior is

$$\Phi^{(0)}\left(\vec{r}\right) = 2 \int\limits_{\vec{n}\bullet\vec{\Omega}<0} d\Omega W\left(\left|\vec{n}\bullet\vec{\Omega}\right|\right)\psi_{inc}\left(\vec{r},\vec{\Omega}\right), \tag{5.96}$$

where, W is defined in terms of Chandrasekhar's H-function [6].

$$W\left(\left|\vec{n}\bullet\vec{\Omega}\right|\right) \equiv \frac{\sqrt{3}}{2}\mu \mathrm{H}\left(\mu\right) \approx \mu + \frac{3}{2}\mu^2 \tag{5.97}$$

If our method represents a Dirichlet scalar intensity on the boundary as a good approximation to Eq. (5.96), we consider the method to properly represent the leading-

order scalar intensity at the boundary such that it will produce the correct leading order interior solution in the diffusion limit.

Following the example in the Adams paper [6], and assuming surface and mass matrix lumping, we take the first angular moment of the *i*th row of Eq. (5.49).

$$\begin{aligned}
&\sum_{f=1}^{f_k}\vec{n}_{f,k}\bullet\sum_m w_m\vec{\Omega}_m\vec{\Omega}_m\tilde{\psi}^{(0)}_{m,f,i}\sum_{\substack{sides\in\\ face\ f_k}}\int_{side} dAu_i \\
&-\sum_{f=1}^{f_k}\vec{n}_{f,k}\bullet\sum_m w_m\vec{\Omega}_m\vec{\Omega}_m\psi^{(0)}_{m,f,i}\sum_{\substack{sides\in\\ face\ f_k}}\int_{side} dAu_i \\
&+\sum_m w_m\vec{\Omega}_m\vec{\Omega}_m\bullet\sum_{sides}\int_{side} dVu_i\vec{\nabla}\psi^{(0)}_m(\vec{r}) \\
&+\sigma\sum_m w_m\vec{\Omega}_m\psi^{(1)}_{m,f,i}\sum_{sides}\int_{side} dVu_i=\frac{\sigma}{4\pi}\sum_m w_m\vec{\Omega}_m\phi^{(1)}_{f,i}\sum_{sides}\int_{side} dVu_i
\end{aligned} \tag{5.98}$$

In Eq. (5.98), we have defined f_k to be the faces in cell k. Using previously defined quantities, it is easy to see that Eq. (5.98) simplifies to

$$\begin{aligned}
&\sum_{f=1}^{f_k}\vec{n}_{f,k}\bullet\sum_m w_m\vec{\Omega}_m\vec{\Omega}_m\tilde{\psi}^{(0)}_{m,f,i}\sum_{\substack{sides\in\\ face\ f_k}}\int_{side} dAu_i \\
&-\frac{1}{4\pi}\sum_{f=1}^{f_k}\vec{n}_{f,k}\sum_m w_m\vec{\Omega}_m\vec{\Omega}_m\phi^{(0)}_{f,i}\sum_{\substack{sides\in\\ face\ f_k}}\int_{side} dAu_i \\
&+\frac{1}{4\pi}\sum_m w_m\vec{\Omega}_m\vec{\Omega}_m\sum_{sides}\int_{side} dVu_i\vec{\nabla}\sum_{j=1}^{J}\phi^{(0)}_j u_j \\
&+\sigma\bar{J}^{(1)}_{f,i}\sum_{sides}\int_{side} dVu_i=0.
\end{aligned} \tag{5.99}$$

where $\bar{J}_i = J_{i,x}\hat{e}_x + J_{i,y}\hat{e}_y + J_{i,z}\hat{e}_z$. The current is not a double vector because we are looking at only one row of the matrix in a cell and the mass matrix is lumped, so there is

only one vertex current in this equation. We continue to assume that our quadrature set integrates the quadratic functions of the direction cosines correctly

$$\sum_m w_m \vec{\Omega}_m \vec{\Omega}_m = \frac{4\pi}{3} \underline{\underline{I}}, \tag{5.100}$$

and use this relationship to obtain

$$\begin{aligned} &\sum_{f=1}^{f_k} \vec{n}_{f,k} \bullet \sum_m w_m \vec{\Omega}_m \vec{\Omega}_m \tilde{\psi}^{(0)}_{m,f,i} \sum_{\substack{sides \in \\ face\ f_k}} \int_{side} dA u_i \\ &-\frac{1}{3} \sum_{f=1}^{f_k} \vec{n}_{f,k} \phi^{(0)}_{f,i} \sum_{\substack{sides \in \\ face\ f_k}} \int_{side} dA u_i \\ &+\frac{1}{3} \sum_{sides} \int_{side} dV u_i \vec{\nabla} \sum_{j=1}^{J} \phi^{(0)}_j u_j \\ &+\sigma \vec{J}^{(1)}_{f,i} \sum_{sides} \int_{side} dV u_i = 0. \end{aligned} \tag{5.101}$$

The first line of Eq. (5.101), which represents the surface terms, and therefore the boundary conditions, can be divided into "incoming" and "outgoing" components

$$\begin{aligned} &\frac{1}{3} \sum_{f=1}^{f_k} \vec{n}_{f,k} \bullet \left[\frac{\phi^{(0)}_{f,i}}{2} + \sum_{m:\vec{n}_{f,k} \bullet \vec{\Omega}_m < 0} 3 w_m \vec{\Omega}_m \vec{\Omega}_m \psi^{(0)}_{m,i,f,k,+} \right] \sum_{\substack{sides \in \\ face\ f_k}} \int_{side} dA u_i \\ &-\frac{1}{3} \sum_{f=1}^{f_k} \vec{n}_{f,k} \phi^{(0)}_{f,i} \sum_{\substack{sides \in \\ face\ f_k}} \int_{side} dA u_i + \frac{1}{3} \sum_{sides} \int_{side} dV u_i \vec{\nabla} \sum_{j=1}^{J} \phi^{(0)}_j u_j + \sigma \vec{J}^{(1)}_{f,i} \sum_{sides} \int_{side} dV u_i = 0, \end{aligned} \tag{5.102}$$

where we have used the fact that the interior leading-order angular flux is isotropic and we have assumed that our quadrature set performs half range integrals correctly:

$$\vec{n} \bullet \sum_{m,n:\vec{n} \bullet \vec{\Omega}_{m,n} > 0} w_{m,n} \vec{\Omega}_{m,n} \vec{\Omega}_{m,n} = \frac{2\pi}{3} \vec{n}\,. \tag{5.103}$$

Based on previous discussion, we can show that the leading order-scalar fluxes are pointwise continuous. In Chapter IV, we have also shown that the leading order-scalar fluxes at the boundary are pointwise continuous in the two-dimensional, RZ case. The same analysis used to prove this pointwise continuity can be directly applied to the three-dimensional boundary leading-order scalar fluxes. To force the leading-order scalar flux to be pointwise continuous on the boundary, we make the assumption that the incident partial currents are spatially continuous. This assumption is true for many, but not all problems of interest. We will omit this proof and state that the leading-order scalar flux is continuous at the boundary:

$$\phi_{i,cell\,1}^{(0)} = \phi_{i,cell\,2}^{(0)} = \phi_{i,cell\,3}^{(0)} = \phi_{i,cell\,4}^{(0)} = \phi_{bdy,i}^{(0)}. \tag{5.104}$$

We are restricting ourselves to hexahedral grids.

We are now able to apply boundary conditions to this method through the surface angular flux, $\psi_{inc,m,i,f,k}^{(0)} = \tilde{\psi}_{m,i,f,k,+}^{(0)}$, in Eq. (5.102). As a result, the boundary condition in the thick diffusion limit for our surface lumped DFEM is

$$DFEM = \left[\frac{\phi_i^{(0)}}{2} \vec{n}_{f,k} + \sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} 3w_m \left(\vec{n}_{f,k}\bullet\vec{\Omega}_m\right)\vec{\Omega}_m \psi_{inc,m,i,f,k}^{(0)} \right] \sum_{\substack{sides \in \\ face\ f_k}} \int_{side} dAu_{i,k}. \tag{5.105}$$

Ideally the surface scalar intensities would satisfy the same condition of the analytic diffusion limit. This ideal condition is given by

$$Ideal = 2 \sum_{m:\vec{n}\bullet\vec{\Omega}_m<0} w_m W\left(\left|\vec{n}_{f,k}\bullet\vec{\Omega}_m\right|\right) \psi_{inc,m,i,f,k} \sum_{\substack{side \in \\ face\ f_k}} \int_{side} dAu_{i,k}\left(\vec{r}\right). \tag{5.106}$$

Because we know that the leading-order scalar flux on the boundary is pointwise continuous, we are able to write $\phi_i^{(0)}$ in terms of $\psi_{inc,m,i,f,k}^{(0)}$, because it is equal to $\phi_{bdy,i}^{(0)}$,

which is defined in Eq. (4.80). We can now substitute Eq. (4.80) into Eq. (5.105) to discover our method's boundary term in the diffusion limit.

$$DFEM = \vec{n}_{f,k} \bullet \left[\begin{array}{l} \displaystyle\sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m 2 \frac{\left|\vec{n}_{f,k}\bullet\vec{\Omega}_m\right|}{\rho_{f,k}} \psi^{(0)}_{inc,m,i,f,k} \\ + \displaystyle\sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} 3 w_m \vec{\Omega}_m \vec{\Omega}_m \psi^{(0)}_{inc,m,i,f,k} \end{array} \right] \sum_{\substack{sides \in \\ face\ f_k}} \int_{side} dA u_i \qquad (5.107)$$

We now define $\mu_m = \vec{n}_{f,k} \bullet \vec{\Omega}_m$, causing Eq. (5.107) to simplify to

$$DFEM = \left[\sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m \left(\vec{n}_{f,k} 2 \frac{|\mu_m|}{\rho_{f,k}} + 3\mu_m \vec{\Omega}_m \right) \psi^{(0)}_{inc,m,i,f,k} \right] \sum_{\substack{sides \in \\ face\ f_k}} \int_{side} dA u_i \,. \qquad (5.108)$$

Next we separate the remaining $\vec{\Omega}_m$ term into a normal and tangential component:

$$\begin{aligned} &\vec{\Omega}_m = \mu_m \vec{n}_{f,k} + \vec{\omega}_m, \\ &\vec{\omega}_m \equiv \text{tangential component} \end{aligned} \qquad (5.109)$$

and substitute it into Eq. (5.108)

$$DFEM = \left[\sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m \left(\vec{n}_{f,k} 2 \frac{|\mu_m|}{\rho_{f,k}} + 3\mu_m \left(\mu_m \vec{n}_{f,k} + \omega_m \right) \right) \psi^{(0)}_{inc,m,i,f,k} \right] \sum_{\substack{sides \in \\ face\ f_k}} \int_{side} dA u_i \,. \qquad (5.110)$$

Eq. (5.110) can be written in terms of a boundary term that is normal to the surface and one that is tangential to the surface

$$\begin{aligned} DFEM &= \left[\vec{n}_{f,k} \Phi_{bdy,f,k,i} + \sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m 3\mu_m \vec{\omega}_m \psi^{(0)}_{inc,m,i,f,k} \right] \sum_{\substack{sides \in \\ face\ f_k}} \int_{side} dA u_i \\ \Phi_{bdy,f,k,i} &= \sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m \left[2\frac{|\mu_m|}{\rho_{f,k}} + 3\mu_m^2 \right] \psi^{(0)}_{inc,m,i,f,k} \\ &\approx \sum_{m:\vec{n}_{f,k}\bullet\vec{\Omega}_m<0} w_m \left[2|\mu_m| + 3\mu_m^2 \right] \psi^{(0)}_{inc,m,i,f,k}. \end{aligned} \qquad (5.111)$$

Eq. (5.111) shows that the normal component of the boundary term for the surface lumped DFEM is a good approximation to the analytical limit in three dimensions, given in Eq. (5.97). The tangential term, however, introduces an erroneous contribution to the leading-order DFEM solution. Thus, the DFEM solution is flawed to the extent that the tangential boundary term is able to propagate through the solution.

To determine the extent to which the tangential term affects the leading-order scalar flux solution, we need to examine the affect that this tangential term of the boundary condition has on the interior solution. The tangential term affects the net current in the boundary cell through Eq. (5.102). This current is coupled to an interior cell through the surface matrix coupling between cells, which we see in Eq. (5.76). As in the two-dimensional case, this current, which contains the contaminating tangential boundary terms is multiplied by quantities involving outgoing normals of faces. These normals come from the surface integral calculations and the within-cell gradient integrals. On an orthogonal grid, the tangential outgoing normal values will be zero. As a result, the tangential contamination terms from the boundary condition will disappear on an orthogonal grid in all cells in the interior of the problem, and the interior solution will see the correct boundary condition, and will produce the correct solution in the interior. For non-orthogonal boundary cells, we rely on the results from Adams, Wareing and Walters [13], which show that the cells near the boundary are affected by this tangential contamination term, but this contamination term cancels out deeper into the problem interior.

The overall result of this diffusion limit analysis for the surface and mass lumped, three-dimensional PWLD method is that it does produce a valid diffusion discretization satisfied by the leading-order scalar flux. This solution is pointwise continuous, which we ensure through surface lumping. The resultant diffusion discretization results in a 27-point stencil and we utilize mass matrix lumping to add diagonal dominance to this solution. Our boundary analysis shows that the surface lumped method approximates

the analytical limit reasonably well. The error introduced by the boundary condition in these problems does not significantly affect the interior solution for many problems of interest. As a result of this analysis we conclude that the fully lumped PWLD method applied to the three-dimensional Cartesian geometry transport equation satisfies the thick diffusion limit of the steady-state, one group, discrete ordinates transport equation.

Summary

In this chapter, we introduce the three-dimensional PWLD discretization of the Cartesian transport equation. We briefly describe how to develop the method, and how to analytically solve the integrals in the method using standard finite element mapping procedures. We then derive a lumped method and a lumping parameter. We design these methods to ensure that the PWLD method satisfies the thick diffusion limit. We end the chapter with a full asymptotic diffusion limit analysis, including the analysis for interior and boundary cells. We compare the resultant diffusion discretization with Palmer's method and a PWL continuous finite element method. We note that the diffusion discretization derived from the PWLD method through the asymptotic analysis is not equivalent in general to either Palmer's method or the PWL CFEM applied directly to the diffusion equation. The results of this analysis show that the fully lumped PWLD method is accurate and robust in the thick diffusion limit. As a result of this analysis, we believe that this method is an accurate discretization for radiation transport problems.

CHAPTER VI
IMPLEMENTATION AND TESTING OF THE XYZ PWLD DISCRETIZATION

This chapter is devoted to the test problems we developed to characterize our three-dimensional Piecewise Linear Discontinuous Finite Element discretization (PWLD) of the transport equation. We begin this chapter with a brief description of the Parallel Deterministic Transport (PDT) code being developed at Texas A&M University. We then turn our attention to a variety of test problems that we have developed in the PDT code to test spatial discretizations. The first test problem we have designed is a one-cell test problem we use to explore PWLD's robustness for problems with poorly shaped cells. We then derive a family of manufactured-solution test problems with quadratic solutions to test PWLD in optically thin and optically thick limits. To compare the accuracy of the PWLD method against an existing method that represents the state of the art, we also run these test problems on a Tri-Linear Discontinuous Finite Element Method (TRILD), and draw some conclusions about the accuracy and robustness of the unlumped, lumped, and lumping-parameter PWLD and TRILD methods.

The Texas A&M University Parallel Deterministic Transport code

The Parallel Deterministic Transport code (PDT), under development at Texas A&M University, is a research project that involves faculty and students from the Nuclear Engineering and Computer Science departments. This code, originally funded by the Advanced Strategic Computing Initiative (ASCI), is designed to be a methods test bed for parallel computing and all types of deterministic transport methods, ranging from time differencing schemes, to spatial discretizations, to new iterative methods. The PDT code is written in C++ making heavy use of the Standard Template Adaptive Parallel Library (STAPL), which has been developed in the Texas A&M University Computer Science department. PDT is a portable, massively parallel deterministic transport code. Furthermore, this code is designed to eventually be able to support multi-physics applications. From an academic perspective, this code is not only valuable as a research

tool, it is also valuable as an educational tool. Adding methods to the PDT code project exposes students to the challenge of working on a larger scale ongoing software development project and teaches them how to work on software development teams composed of people from a variety of technical backgrounds.

To implement a new spatial discretization in the PDT code, a physics developer need only write a single-cell solution module for the new spatial discretization. This single cell solution is called a "sweep chunk" routine. The code is designed to take this sweep chunk and apply to all cells in a mesh, for all quadrature directions, for all energy groups, at every time step. Parallelization and portability details are handled by STAPL. This library replaces the Standard Template Library (STL) in the PDT code and allows the code to be partitioned for massively parallel jobs. One advantage of building PDT on top of STAPL is to be able to quickly develop deterministic transport methods that can be tested for massively parallel applications [28].

One of the most difficult tasks of developing efficient algorithms, in particular efficient parallel algorithms, for deterministic transport methods is to develop a sweep order by which the cells in the problems are traversed for a given quadrature direction. The PDT code determines this sweep order using a directed, acyclic graph. This graph is formed at the beginning of the calculation and has been developed in a flexible way such that it can naturally deal with cycles in the grid and partitioning of the mesh for parallel calculations. Graph theory has proven to have significant potential as a computational method for handling sweep orders in deterministic calculations.

One of the significant benefits of the three-dimensional PWLD research described herein is to extend the capabilities of the PDT code and further test previously existing pieces of the code. Throughout this research we have added capabilities in the code to handle spatial discretization methods whose unknowns are stored in the corners of the cell. These types of discretizations include discontinuous finite element methods and corner

balance methods; thus, the work described here has enabled the implementation of more than just PWLD methods into the code. Furthermore, to test our discretization, we added features that allowed for linear and quadratic boundary conditions and sources. When testing all of the added features, we ensured that each new feature worked with existing parallelization algorithms and iterative methods for converging the scattering source. Finally, to complete our research, we added four corner-based spatial discretizations: the PWLD and BLD method in 2D cylindrical geometry and the PWLD and TRILD method in 3D Cartesian geometry. Also, all test problems we present here will be added to the standard test suite in the PDT code repository for future testing and future spatial discretization research. These unit test capabilities may be the most valuable contribution we will make to the PDT project throughout the course of this research because they will allow for more efficient method implementation and testing in the future. We cannot place enough emphasis on the importance of developing mathematically and physically significant unit tests for each new method that is added to the code.

The results presented in this chapter are all produced from the PDT code. We think these results are important because they help us characterize the PWLD method and also because they empirically demonstrate many of the capabilities of the PDT code.

Three-dimensional test problems for discontinuous finite element methods

Before we describe the actual test problems that we have run to test our method's accuracy and robustness, we describe the spatial grids that our code can generate. The first grid type is the orthogonal hexahedral mesh. The cells in this mesh are all hexahedra with flat faces. We will not have cycles in this mesh type, so the sweep order of each quadrature direction can be written as a directed, acyclic graph. We show an example of an 8x8x8 orthogonal, hexahedral grid in Figure 6.1.

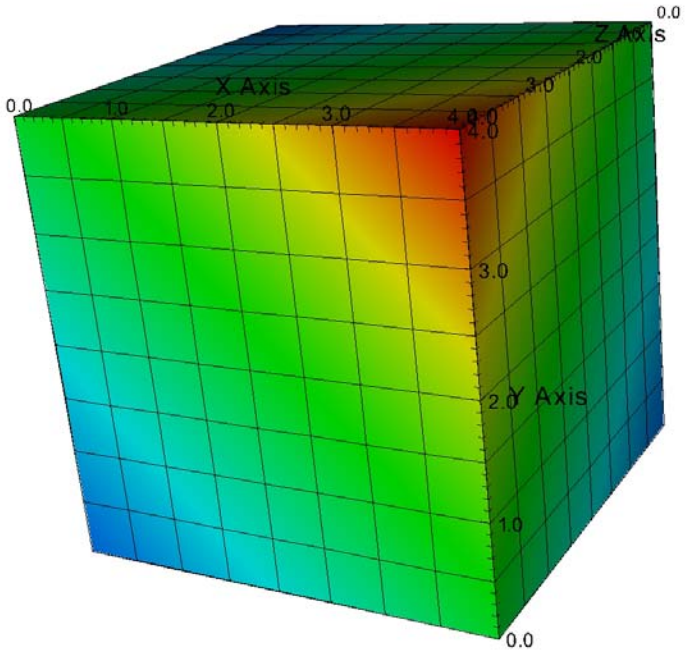

Figure 6.1: An 8x8x8 orthogonal grid

A second type of grid we utilize in our test problems is called a random grid. This grid is created by perturbing the vertices of the orthogonal grid by a random fraction of the distance to adjacent vertices, such that the vertices' positions change as given by Eq. (6.1).

$$\vec{v}\left(\tilde{x},\tilde{y},\tilde{z}\right)=\vec{v}\left(x,y,z\right)+\frac{1}{2}\left[P_x N_{rand,x}\Delta x\hat{i}+P_y N_{rand,y}\Delta y\hat{j}+P_z N_{rand,z}\Delta z\hat{k}\right], \quad N_{rand,x},N_{rand,y},N_{rand,z}\in\left[-1,1\right], \tag{6.1}$$

where P is the maximum fraction by which we randomize our vertices, and N_{rand} is a distinct random number generated by a standard C++ function. Eq. (6.1) ensures that a vertex will not cross over any other vertex in the same cell for any random fraction less than 1. As a result, we can ensure that we do not create cells with negative volumes, but we can create cells with faces that are non-planar and sweep graphs that have cycles due to the perturbation of the faces of cells. We show a few cells on the interior of and

8x8x8 cell random mesh in Figure 6.2. Notice that these cells have faces with non-co-planar vertices.

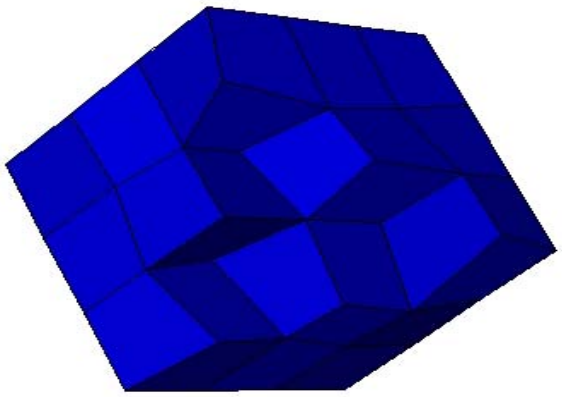

Figure 6.2: A few cells on the interior of a random mesh

Typically, a random mesh is defined to allow the vertices on the surface of the problem domain to move along that problem domain. We do not have that capability in generating our random meshes. As a result, the surfaces of our problems will have only vertices that are not perturbed from the original orthogonal grids.

The first set of test problems we consider have linear solutions. Since both PWLD and TRILD solution spaces contain all linear functions, these methods should exactly represent the linear-in-space solutions. A linear solution problem for the three-dimensional Cartesian transport equation can be derived as follows. The 3D Cartesian transport equation is

$$\mu\frac{\partial\psi}{\partial x}+\eta\frac{\partial\psi}{\partial y}+\xi\frac{\partial\psi}{\partial z}+\sigma_t\psi=\frac{\sigma_s}{4\pi}\phi+\frac{S}{4\pi}, \tag{6.2}$$

where

$$\phi=\int_{\Omega}d\Omega\,\psi. \tag{6.3}$$

A linear solution in space and angle for this equation is

$$\begin{aligned}\psi(x,y,z,\mu,\eta,\xi) &= ax+by+cz+d\mu+e\eta+f\xi+g \\ \phi(x,y,z) &= 4\pi(ax+by+cz+g).\end{aligned} \tag{6.4}$$

Our boundary conditions are uniquely determined by Eq. (6.4) and we plug this solution into Eqs. (6.2) and (6.3) to determine the source that will produce this linear solution. As a result,

$$S = 4\pi\left[a\mu+b\eta+c\xi+\sigma_t(d\mu+e\eta+f\xi)+(\sigma_t-\sigma_s)(ax+by+cz+g)\right]. \tag{6.5}$$

In order to have only a spatially and not directionally dependent source we can set

$$\begin{aligned}a &= -\sigma_t d, \\ b &= -\sigma_t e, \\ c &= -\sigma_t f.\end{aligned} \tag{6.6}$$

We have run multiple linear solution problems for both PWLD and TRILD on orthogonal and random grids, with and without scattering. Both methods exactly reproduce the linear solution to the level of convergence specified on all of these test problems. We show a three dimensional linear solution from a problem with an S_6 quadrature set and scattering on a random grid for PWLD and TRILD in Figure 6.3 and Figure 6.4, respectively.

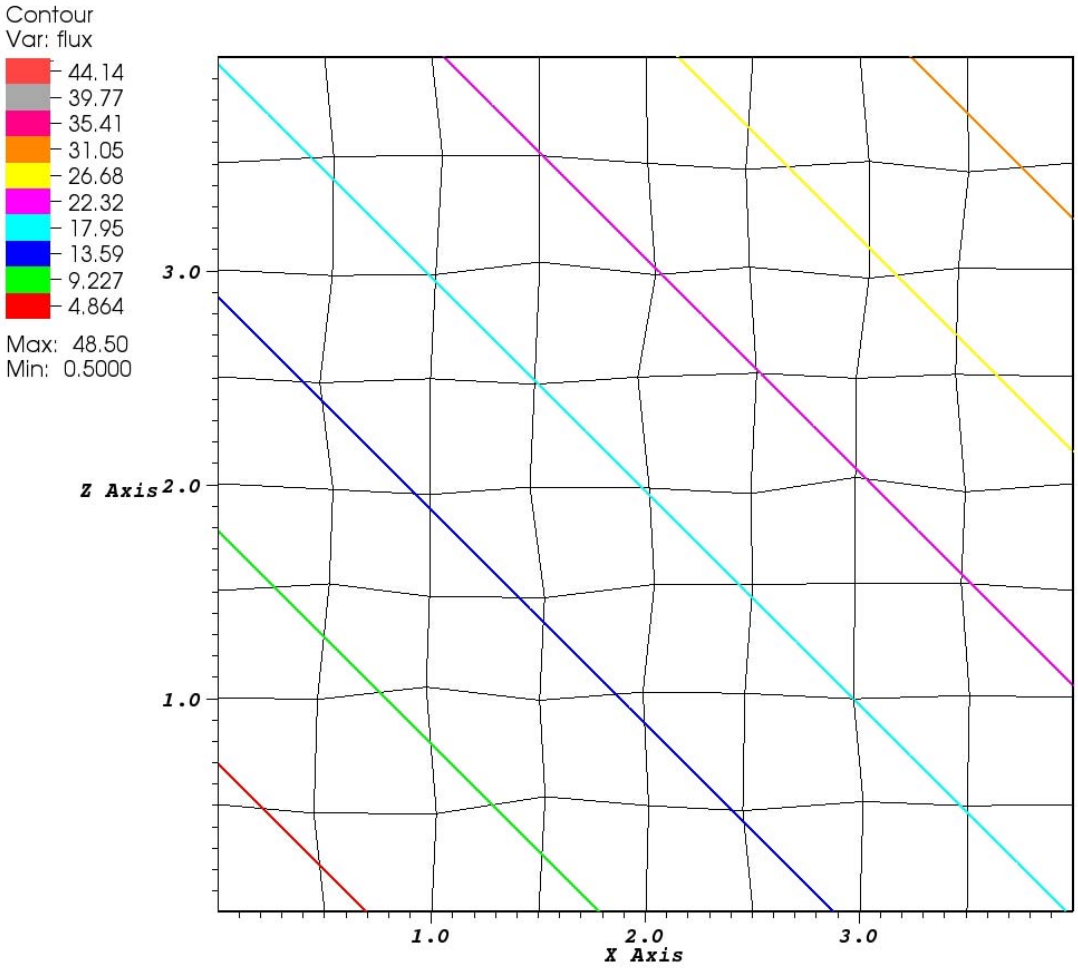

Figure 6.3: A contour plot of a PWLD linear solution on a random grid with a slice at y=0.4

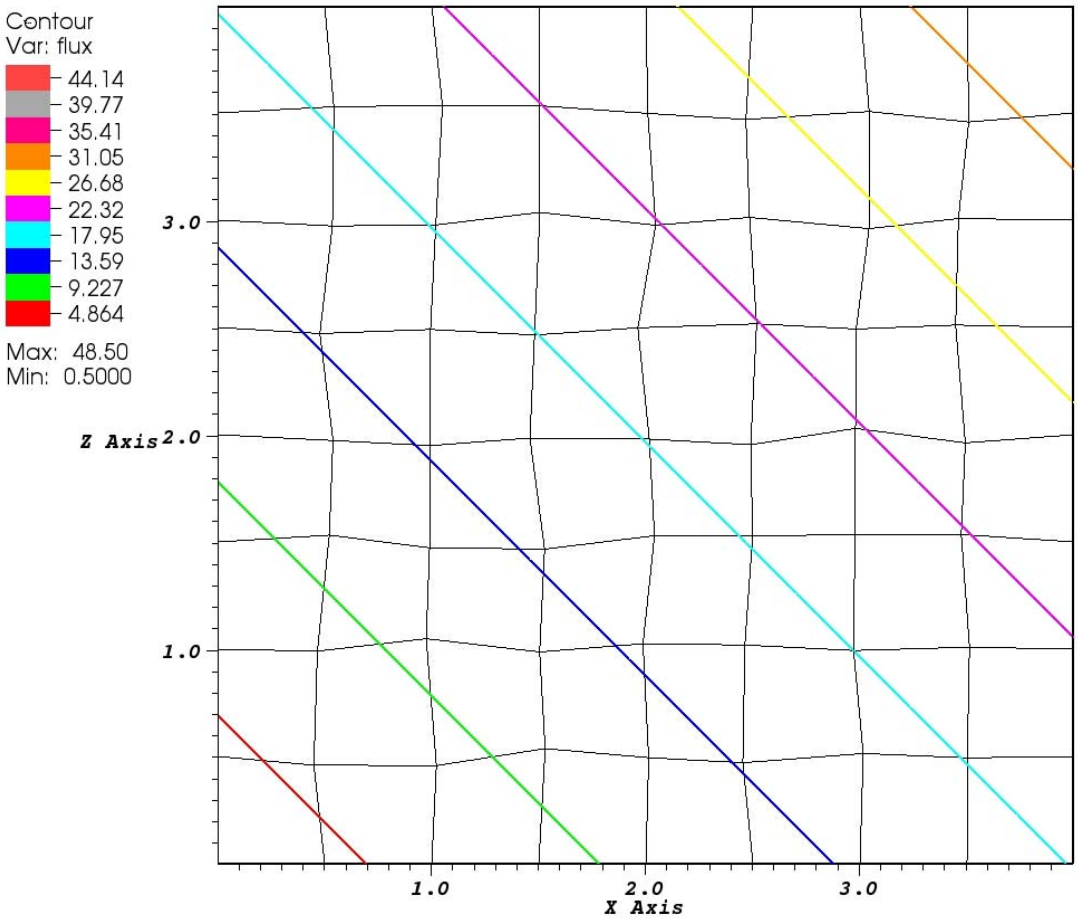

Figure 6.4: A contour plot of a TRILD linear solution on a random grid with a slice at y=0.4

These linear solution results give us confidence that our method is behaving as expected, and that we have implemented both methods correctly. As a result, the linear solution test problem is a particularly good unit test to add to the PDT repository.

The next test problem we developed we call the one-cell test problem. Before we describe this problem in detail we recall that in general DFEMs act like one-cell finite element problems whose boundary condition is determined by the upwinding condition

on the surface of the cell. Therefore, we can use one cell as a simple testbed, but we recognize that the results can be generalized to meshes with more than one cell. As a result of this property, a one-cell test problem, such as the problem presented here, is an extremely powerful diagnostic tool for discontinuous finite element methods in general.

In this test problem, we have a one cell within a problem domain of 4 cm x 4 cm x 4 cm, ranging from the point (0,0,0) to the point (4,4,4). We take the vertex at the origin, (0,0,0), of our domain and move it incrementally toward the vertex located at (4,4,4). That is, we begin with a cube and incrementally make it closer to concave, slightly concave, and ultimately dramatically concave. As we do this, some of the faces become significantly non-planar. Moreover, some of the "side" subcells take on negative volumes when the cell center point is outside the cell, which is true for some of tests. We have run the linear-solution test with an S_2 level-symmetric quadrature set and scattering on six of these one-cell problems where one vertex of the cell is set at six distinct points given in Table 6.1.

Table 6.1: List of one cell test cases

Test Case	Cell Origin	Cell center?	Re-entrant cell?
1	(0,0,0)	Inside cell	No
2	(1,1,1)	Inside cell	No
3	(2,2,2)	Inside cell	No
4	(2.2857, 2.2857, 2.2857)	On the surface of the cell	Yes
5	(3,3,3)	Outside cell	Yes
6	(4,4,4)	Outside cell	Yes

The coefficients for the linear-solution problem in Eq. (6.4) are

$$\begin{aligned} a = b = c &= 4.0, \\ d = e = f &= -0.5, \\ g &= 0.5. \end{aligned} \tag{6.7}$$

and our cross sections are $\sigma = 8.0$ and $\sigma_s = 4.0$.

We plot each of these six cases in Figure 6.5-Figure 6.10, using volumetric plots in which color represents solution magnitude. From these plots, it is easy to see that PWLD's ability to exactly reproduce the linear solution is extremely robust even on exceedingly distorted cells.

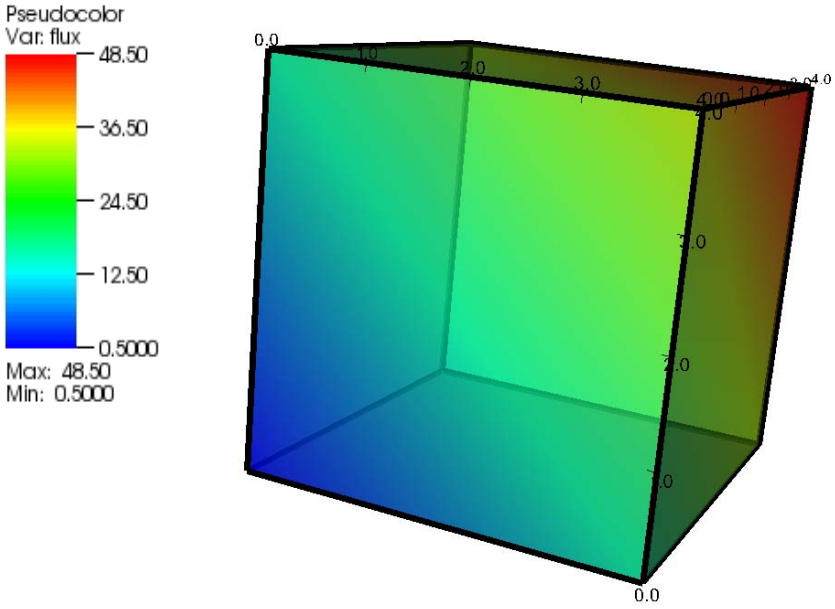

Figure 6.5: One cell, vertex at (0,0,0) (undistorted case)

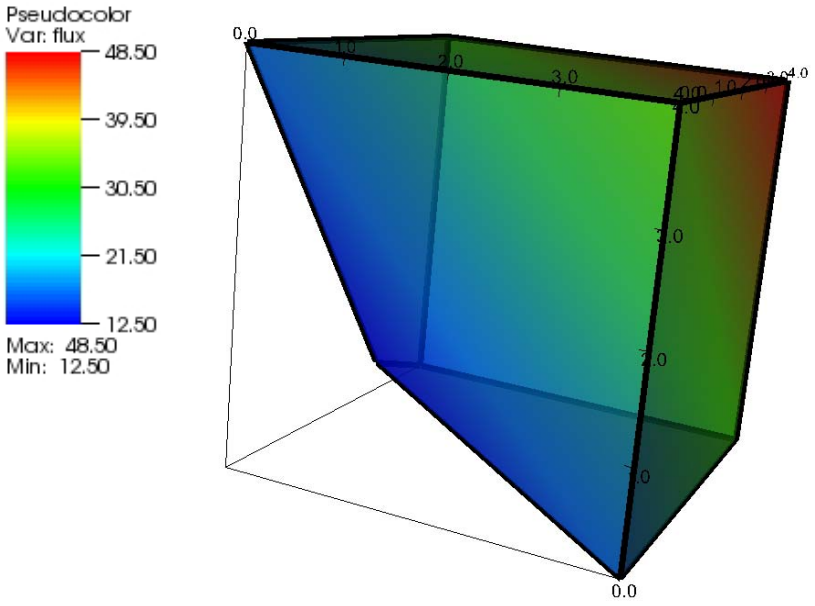

Figure 6.6: One cell origin at (1,1,1)

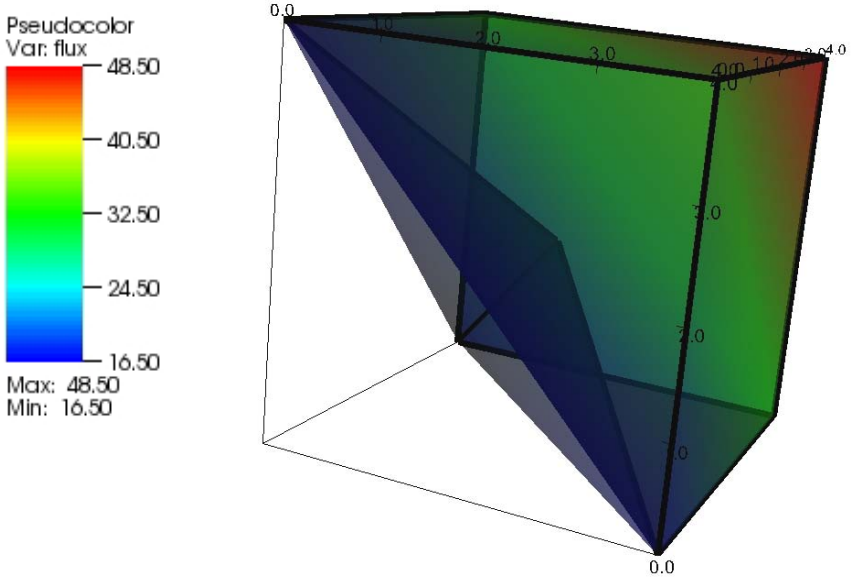

Figure 6.7: One cell origin at (2,2,2)

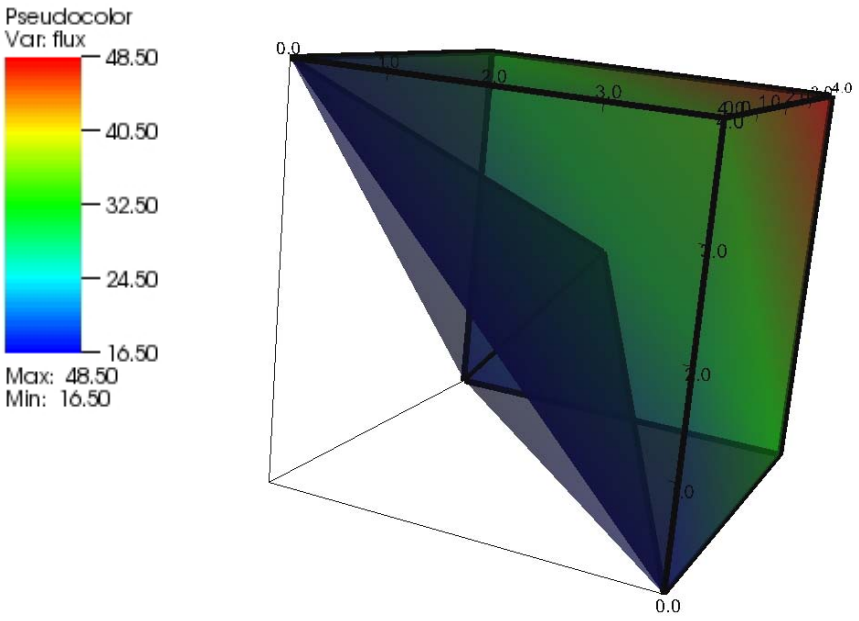

Figure 6.8: One cell origin at (2.2857, 2.2857, 2.2857)

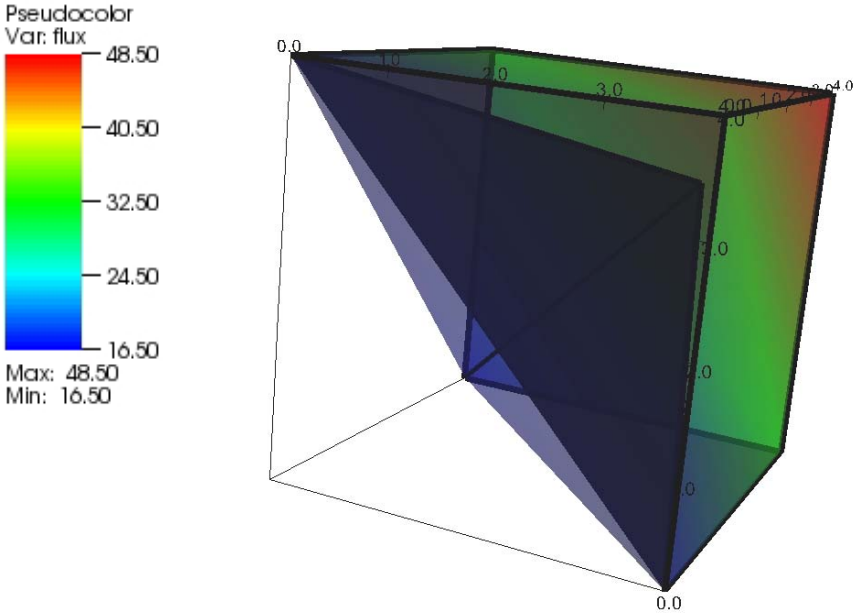

Figure 6.9: One cell origin at (3,3,3)

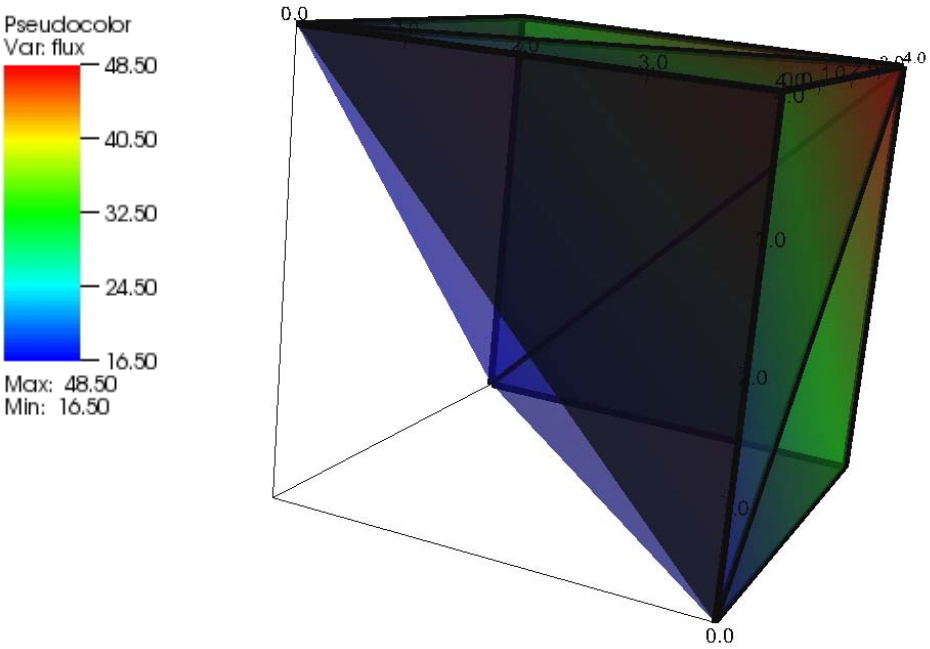

Figure 6.10: One cell origin at (4,4,4)

We also ran a test problem on the same series of meshes, whose solution was a quadratic function instead of a linear function. In general, the solution to this problem was also well-behaved on each of these one cell test cases.

The remainder of our test problems are truncation-order test problems. In truncation error test problems, we define a problem for which we know a solution, but whose solution is not in the solution space of the method we are testing. We then define a series of meshes on which we will solve this problem. Usually, these meshes start with a coarse mesh that is refined by a factor of two in every dimension until enough meshes are created to generate a truncation error trend. Since we know the solution to the problem we ran, we can define a norm to compare the error of the solution on each mesh. We use the L_2 norm for these calculations, where the L_2 norm is defined as

$$\|f\|_2 = \left\{ \int_V dV f^2 \right\}^{1/2}. \tag{6.8}$$

For our truncation error problems, we use an L_2 norm of the percent difference of the error weighted by a subcell corner volume, and divide this norm by the L_2 norm of the exact solution. Mathematically, we write this as

$$\|error\|_2 = \frac{\left\{ \sum_{corners \in domain} \left[\frac{|\phi_{exact} - \phi_{calculated}|}{\phi_{exact}} \right]^2 V_{corner} \right\}^{1/2}}{\left\{ \int_V dV \phi_{exact}^2 \right\}^{1/2}}. \tag{6.9}$$

where a corner is a subcell volume defined by a cell vertex, the mid-points of edges that touch the vertex, the center point of faces that touch the vertex, and a cell center. Truncation-error problems are useful for many purposes. First, empirical evidence leads us to expect that a discontinuous finite element spatial discretization of the transport equation that employs a linear approximation in space should be a second-order accurate method. Demonstrating this property is an excellent code verification tool, as well as a good test to compare the accuracy of a new method to existing methods.

Two major methods exist for finding the "known" solution for a truncation error study. One way to find a known solution is to run an arbitrary problem on an extremely fine mesh such that the solution to this problem will not change if the mesh is refined, proving that the solution on that mesh is a fine-mesh solution and in fact correct to the truncation error of the computer and the tolerance of the linear solver. A second method is to find a problem with a sufficiently complicated analytic solution using the method of manufactured solutions. We developed our own manufactured-solution test problem.

This test problem has a solution that varies quadratically in space and angle including quadratic cross term such as *x* times *y*. We start with the time-independent, one group, transport equation with isotropic scattering.

$$\mu\frac{\partial\psi(x,y,z,\mu,\eta,\xi)}{\partial x}+\eta\frac{\partial\psi(x,y,z,\mu,\eta,\xi)}{\partial y}+\xi\frac{\partial\psi(x,y,z,\mu,\eta,\xi)}{\partial z}$$
$$+\sigma\psi(x,y,z,\mu,\eta,\xi)=S(x,y,z,\mu,\eta,\xi)+\frac{\sigma_s}{4\pi}\phi(x,y,z) \tag{6.10}$$

We now write down a solution, and use this solution to determine the source and boundary conditions for the problem. The angular intensity solution we choose is

$$\begin{aligned}\psi(x,y,z,\mu,\eta,\xi) &= a+bx+c\mu+dx\mu+ex^2+f\mu^2\\ &+gy+h\eta+iy\eta+jy^2+k\eta^2\\ &+lz+m\xi+nz\xi+oz^2+p\xi^2\\ &+qxy+rxz+syz+t\mu\eta+u\mu\xi+v\eta\xi\\ &+w\mu y+A\eta x+B\xi x+C\mu z+D\eta z+E\xi y\end{aligned} \tag{6.11}$$

and the resultant scalar intensity is

$$\begin{aligned}\phi(x,y,z) &= \int_{4\pi} d\Omega\,\psi\\ &= 4\pi\left(a+bx+ex^2+gy+jy^2+lz+oz^2+qxy+rxz+syz\right)\\ &+\frac{4\pi}{3}f+\frac{4\pi}{3}k+\frac{4\pi}{3}p.\end{aligned} \tag{6.12}$$

When the solution in Eq. (6.11) is substituted into Eq. (6.10), the result is

$$
\begin{aligned}
&\mu\left(b+d\mu+2ex+qy+rz+A\eta+B\xi\right)\\
&+\eta\left(g+i\eta+2jy+qx+sz+w\mu+E\xi\right)\\
&+\xi\left(l+n\xi+2oz+rx+sy+C\mu+D\eta\right)\\
&+\sigma\begin{pmatrix}a+bx+c\mu+dx\mu+ex^2+f\mu^2\\+gy+h\eta+iy\eta+jy^2+k\eta^2\\+lz+m\xi+nz\xi+oz^2+p\xi^2\\+qxy+rxz+syz+t\mu\eta+u\mu\xi+v\eta\xi\\+w\mu y+A\eta x+B\xi x+C\mu z+D\eta z+E\xi y\end{pmatrix}\\
&=S\left(x,y,z,\mu,\eta,\xi\right)+\sigma_s\begin{bmatrix}a+bx+ex^2+gy+jy^2+lz+oz^2\\+qxy+rxz+syz+\frac{1}{3}f+\frac{1}{3}k+\frac{1}{3}p\end{bmatrix}.
\end{aligned}
\tag{6.13}
$$

Using Eq. (6.13), we now solve for the source needed to satisfy the manufactured solution.

$$
\begin{aligned}
&S\left(x,y,z,\mu,\eta,\xi\right)\\
&=\mu\left(b+d\mu+2ex+qy+rz+A\eta+B\xi\right)\\
&+\eta\left(g+i\eta+2jy+qx+sz+w\mu+E\xi\right)\\
&+\xi\left(l+n\xi+2oz+rx+sy+C\mu+D\eta\right)\\
&+\sigma\begin{pmatrix}c\mu+dx\mu+f\mu^2+h\eta+iy\eta+k\eta^2\\+m\xi+nz\xi+p\xi^2+t\mu\eta+u\mu\xi+v\eta\xi\\+w\mu y+A\eta x+B\xi x+C\mu z+D\eta z+E\xi y\end{pmatrix}\\
&+\left(\sigma-\sigma_s\right)\left(a+bx+ex^2+gy+jy^2+lz+oz^2+qxy+rxz+syz\right)\\
&-\sigma_s\left[\frac{1}{3}f+\frac{1}{3}k+\frac{1}{3}p\right]
\end{aligned}
\tag{6.14}
$$

This source, given in Eq. (6.14), can be simplified if we choose the coefficients of our solution carefully. For example, if we choose

$$
\begin{aligned}
b\mu &= -\sigma c\mu \\
d\mu^2 &= -\sigma f\mu^2 \\
2ex\mu &= -\sigma dx\mu \\
g\eta &= -\sigma h\eta \\
i\eta^2 &= -\sigma k\eta^2 \\
2jy\eta &= -\sigma iy\eta \\
l\xi &= -\sigma m\xi \\
n\xi^2 &= -\sigma p\xi \\
2oz\xi &= -\sigma nz\xi \\
qy\mu &= -\sigma wy\mu \\
qx\eta &= -\sigma Ax\eta \\
rx\xi &= -\sigma Bx\xi \\
rz\mu &= -\sigma Cz\mu \\
sz\eta &= -\sigma Dz\eta \\
sy\xi &= -\sigma Ey\xi \\
(A+w)\mu\eta &= -\sigma t\mu\eta \\
(B+C)\mu\xi &= -\sigma u\mu\xi \\
(E+D)\eta\xi &= -\sigma v\eta\xi
\end{aligned}
\tag{6.15}
$$

the source simplifies to

$$
\begin{aligned}
S(x,y,z,\mu,\eta,\xi) = (\sigma-\sigma_s)\left(a+bx+ex^2+gy+jy^2+lz+oz^2+qxy+rxz+syz\right) \\
-\frac{\sigma_s}{3}\left[f+k+p\right],
\end{aligned}
\tag{6.16}
$$

which is a simple expression based only on spatial quantities (i.e., it is isotropic). The boundary conditions for this manufactured solution are Dirichlet boundary conditions given by the solution itself in Eq. (6.11). However, because this solution is not linear, we cannot represent it exactly on the surface of the problem with our linear interpolating functions. For this reason, we choose to preserve integral quantities of the exact boundary condition when we set up our problem:

$$\int_{A_{cell}} dA u_i \sum_{j=1}^{N} \psi_j u_j = \int_{A_{cell}} dA u_i \psi_{boundary} , \tag{6.17}$$

where the u_j's in Eq. (6.17) are the same set of basis functions used in the spatial discretization. We also have to represent the source coefficients in the same way because the source is no longer spatially linear, and we cannot represent it exactly by expanding it in terms of our basis functions.

$$\int_{V_{cell}} dV v_i \sum_{j=1}^{N} S_j u_j = \int_{V_{cell}} dV v_i S_{exact} \tag{6.18}$$

To determine these coefficients in our code, we use quadrature integration for the right hand side of Eqs. (6.17) and (6.18). For the PWLD basis functions, we use a four-point quadrature set for triangles on the boundary condition faces, and a five-point quadrature set for the source for tetrahedral sides in the cells [22]. For the TRILD method we used a standard four-point Gauss quadrature set for the boundary condition faces, and a standard eight-point Gauss quadrature set for the source for the hexahedral cell shapes [22].

We note two properties of this manufactured solution. The logic we used to derive the quadratic manufactured solution can be extended to deriving higher order manufactured solutions, but in order to force the source to be angularly independent, the number of terms in the solution increases exponentially. Also, the manufactured solution we derived here favors the unlumped TRILD method because the spatial cross terms—the *xy*, *xz*, and *yz* terms—are in the vector space of the trilinear basis functions. However, when we lump the method, the lumping is equivalent to assuming that the solution gradient is constant, which it is not when the solution contains quadratic terms; therefore, it is unclear if the lumped versions of TRILD should be more accurate than the lumped versions of PWLD. For well behaved problems, we expect unlumped TRILD to

perform more accurately than PWLD. One measure of success of PWLD is how closely it can match TRILD on these problems that contain cross terms.

We first ran this quadratic truncation error problem on a series of seven different meshes for an optically thin case, where

$$
\begin{aligned}
&\sigma = 8 \\
&\sigma_s = 4 \\
&Domain = 4x4x4 \\
&Quadrature\ Set = S_6 \\
&Iterative\ Method = GMRES, \text{ Restart=10} \\
&a = 0 \\
&b = g = l = 4.0 \\
&c = h = m = -0.5 \\
&d = i = n = -2.0 \\
&e = j = o = 8.0 \\
&f = k = p = 0.25 \\
&q = r = s = 16 \\
&t = u = v = 0.5 \\
&w = A = B = C = D = E = -2.0.
\end{aligned}
\qquad . \qquad (6.19)
$$

We ran this problem for unlumped, lumped, and lumping-parameter PWLD and TRILD cases on an orthogonal grid, a random grid with a randomization factor of 0.25, and a random grid with a randomization factor of 0.33. The latter factor causes the meshes to have re-entrant faces. In the plots of these results, “U” before a method’s name refers to “unlumped”, “L” refers to “lumped”, and “LP” refers to “lumping parameter.”

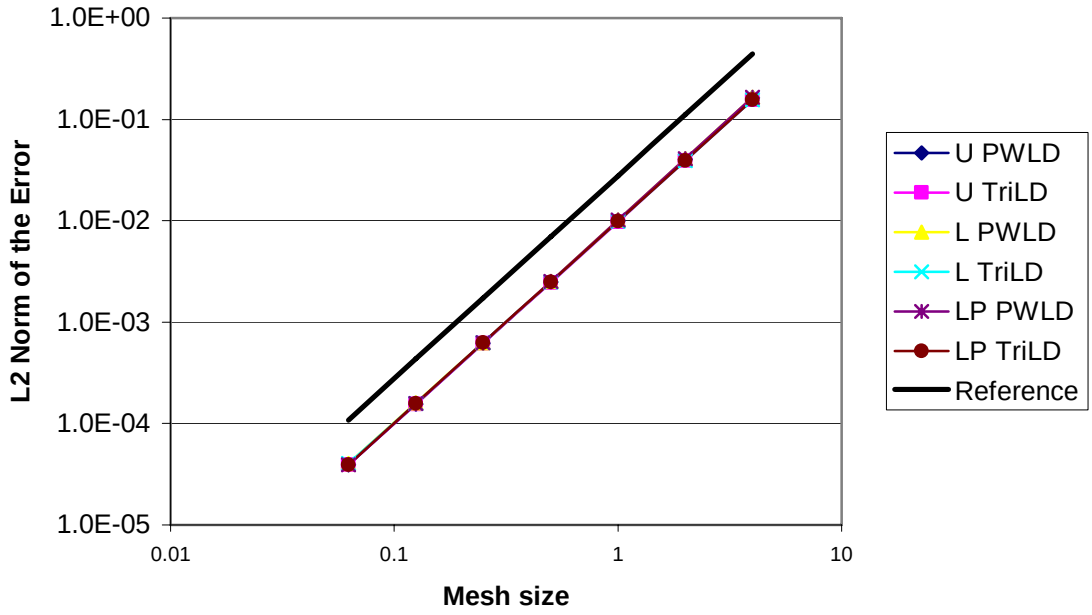

Figure 6.11: Truncation error of all methods on an orthogonal grid in the thin limit

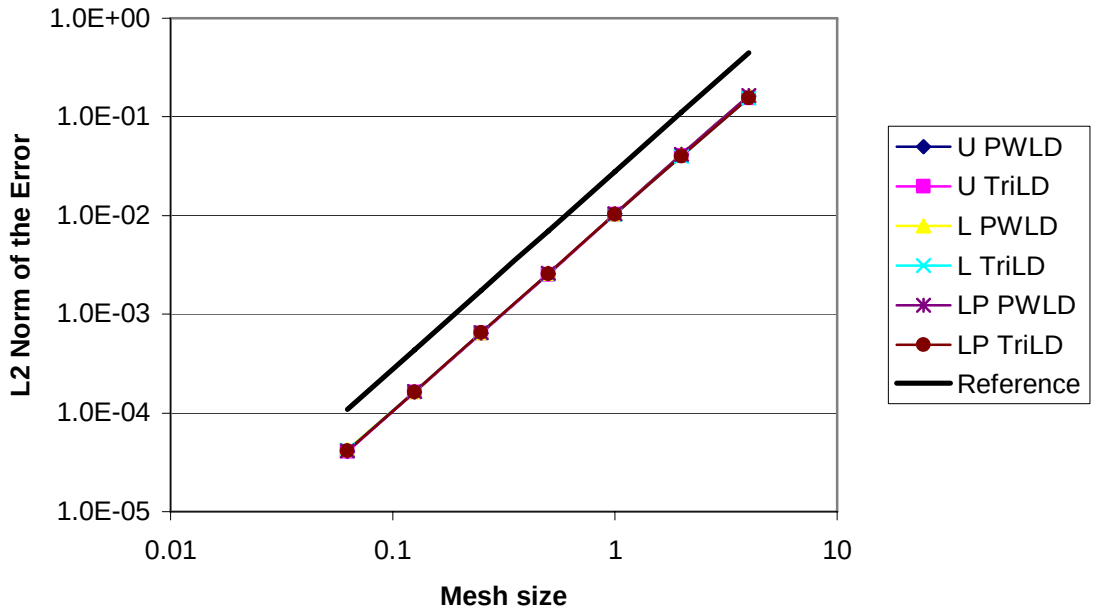

Figure 6.12: Truncation error of all methods on a random grid, vertices perturbed by up to 25% of the distance to the center of the edge, in the thin limit

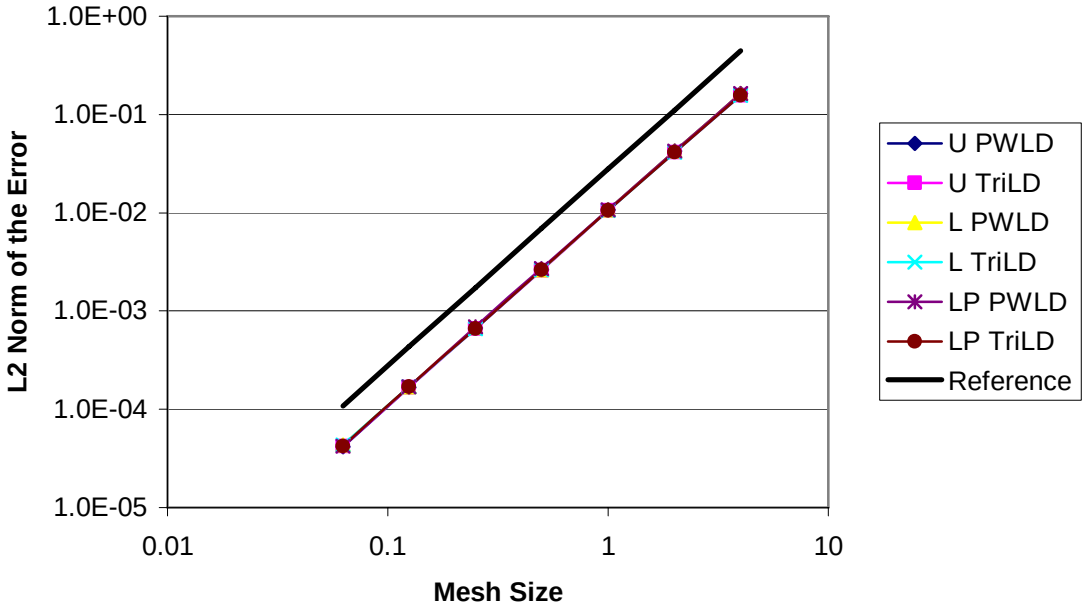

Figure 6.13: Truncation error of all methods on a random grid, vertices perturbed by up to 33% of the distance to the center of the edge in the thin limit using source iteration

For the orthogonal grid problems, shown in Figure 6.11, all methods easily produce a second-order truncation error rate. Surprisingly, the accuracy of the unlumped PWLD method is extremely close to the unlumped TRILD method, even though unlumped TRILD has a distinct advantage on this test problem. This trend remains true for all grid types, for all method types indicating that PWLD is just as accurate as TRILD on arbitrary hexahedral grids. We show the results of the two random grid problems in Figure 6.12 and Figure 6.13. In the thin limit, for all grids, the difference in accuracy between the unlumped, lumped and lumping parameter methods is insignificant for these test problems.

When we ran these problems with GMRES for the grids with re-entrant cells, we found that we began to lose accuracy. In fact, the 64x64x64 cell solution was less accurate

than the 32x32x32 cell solution for both PWLD and TRILD. This result proves nothing about the spatial discretization, but a great deal about how we have implemented the iterative method. When re-entrant cells exist in a mesh, causing "cycles" in the sweeping order, PDT breaks cycles by using surface intensities from the previous iteration. Thus, these surface intensity values become iterative unknowns along with the scattering source. However, the GMRES algorithm in PDT does not account for these additional unknowns, and as a result the code does not check whether they are converged. GMRES iterates on norms of the residual of the scalar intensity solution and stops when this quantity is convergence. Unfortunately, for fine highly randomized grids, the scalar intensity solution converges before the surface angular fluxes, and the code reports that the solution has converged when in reality it has not. As a result, we cannot draw any conclusion about this particular truncation error test when the problems are run with GMRES. We re-ran the problem using source iteration (Richardson's iteration), and we recovered our second-order behavior, which we show in our plots. However, source iteration does not rigorously add the surface angular intensity unknowns to its unknown vector, it just converges more slowly giving these surface unknowns more iterations in which to catch up to the rest of the solution. Also, source iteration tests not only a norm of the solution, but also the error in the pointwise solution, which is a much more stringent test of the solution's convergence.

The truncation-error test problem shows a vulnerability of iterative-method implementation in PDT. To rigorously check that the entire solution is converged, we would have to add the surface intensity unknowns to the unknown vector on which the iterative method is operating. Also, this would allow methods such as GMRES to accelerate the convergence of these surface intensities. A simpler way to mitigate the problem for GMRES would be to implement a pointwise error check instead of the global norm.

The overall result of the truncation error test problems in the thin limit is that we have considerable evidence that the PWLD method is both accurate and robust in the thin limit, and that it is coded correctly.

We now turn our attention to the thick limit. We have used this same quadratic manufactured solution to produce a problem that tests our methods in the thick diffusion limit. We design this problem such that the cells in the problem are more refined in diffusion lengths but extremely thick in mean free paths. A diffusion length is defined as

$$L = \sqrt{\frac{1}{3\sigma\left(\sigma - \sigma_s\right)}}, \tag{6.20}$$

and a mean free path is defined as

$$mfp = \frac{1}{\sigma}. \tag{6.21}$$

The physical parameters in this problem are

$$\begin{aligned} &\sigma = 128,000 \\ &\sigma_s = 127,999.9573333333 \\ &c = 0.999999666667 \\ &Domain = 4x4x4 \\ &Quadrature\ Set = S_6 \\ &Iterative\ Method = GMRES, \text{ Restart=10.} \end{aligned} \tag{6.22}$$

On the most refined grid, the ratio of the cell width to diffusion length of this problem is 8.01282, but the number of mean free paths in the most refined cell is 2000. For this problem, which is clearly in the thick diffusion limit for all meshes, we must use GMRES as our iterative method, otherwise this problem will consume very large amounts of computational assets. We again run this problem for unlumped, lumped, and lumping parameter PWLD and TRILD cases on an orthogonal grid and a random grid

with a randomization factor of 0.25. We cannot run this problem on a random grid with a randomization factor of 0.33 because the cells become re-entrant and GMRES is not able to converge the surface unknowns as we discussed previously.

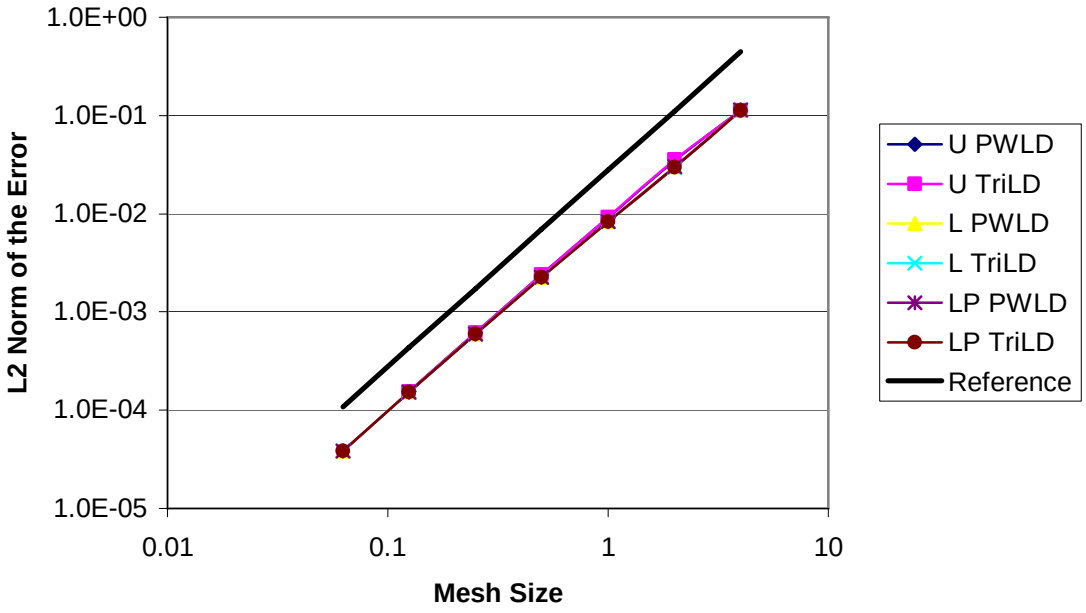

Figure 6.14: Truncation error for all methods on an orthogonal grid in the thick limit

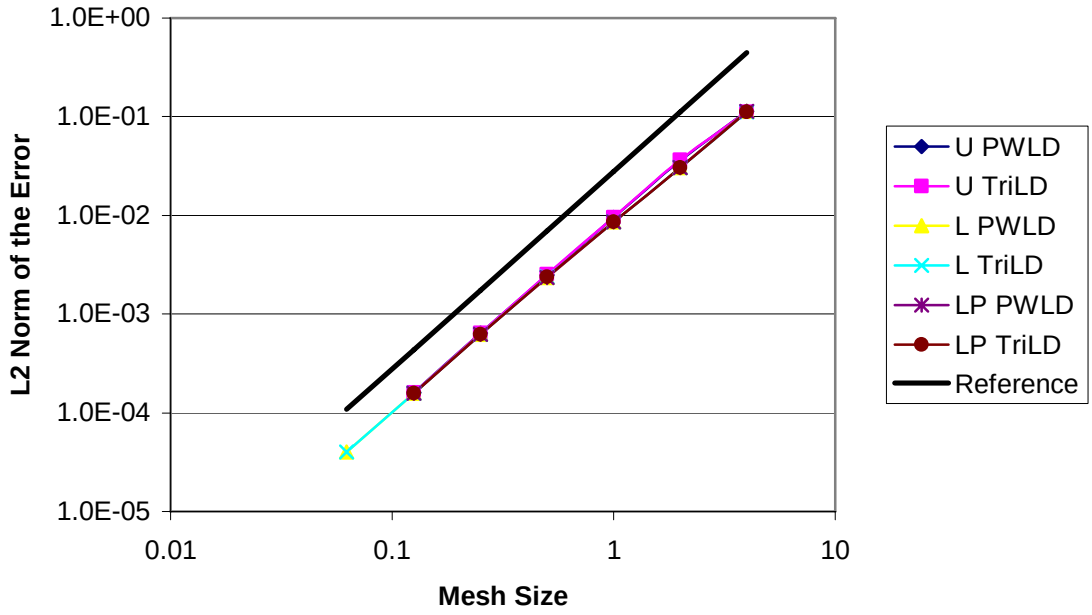

Figure 6.15: Truncation error for all methods on a random grid, vertices perturbed by up to 25% of the distance to the center of the edge, in the thick limit

The results of this series of thick diffusion test problems, shown in Figure 6.14 for an orthogonal grid and in Figure 6.15 for a random grid, indicate that our method does in fact retain its accuracy in this limit as predicted by the diffusion-limit analysis in the previous chapter. That is, the diffusion discretization satisfied by the leading-order solution in the diffusion limit appears to be a second-order approximation of the correct diffusion equation. The behavior does not lock solidly onto second order until the finest meshes that we ran, but this is understandable because the cells are not very fine compared to the diffusion length, L.

We see that the accuracy of PWLD in this limit is comparable to the accuracy of TRILD for both the orthogonal and random grid types for all discretization types. The importance of this result cannot be overstated. The goal of the PWL research is to develop a method that is as accurate as TRILD on hexahedral cells but that is extensible

to general polyhedral cells. The results shown here suggest that we have attained this goal.

The lumped methods, as predicted by the analysis, are more robust than the unlumped methods. The magnitude of lumped method error is slightly less than the unlumped method in general, and the number of iterations, shown in Table 6.2, required for convergence is significantly less for lumped methods compared to unlumped methods. Also, we note that the lumped and lumping-parameter methods have very similar behavior in the thick limit. This indicates that the lumping-parameter method behaves as we designed it.

Table 6.2: Iteration counts for thick limit problems on orthogonal grids

	Iterations: GMRES for Orthogonal/Random Grids, Thick Limit					
Cell Size	U PWLD	U TRILD	L PWLD	L TRILD	LP PWLD	LP TRILD
4	3/3	3/3	2/2	2/2	2/2	2/2
2	36/47	57/61	10/28	9/27	10/28	9/27
1	62/63	71/65	17/36	17/34	17/36	17/34
0.5	80/80	101/132	17/49	17/49	17/49	17/49
0.25	84/64	162/250	27/60	27/60	27/60	27/60
0.125	98/84	233/349	27/79	27/79	27/79	27/79
0.0625	122/123	NA/NA	37/97	36/97	37/NA	36/NA

When we compare the PWLD results to the TRILD results in the thick diffusion limit, we observe some interesting behavior. For the unlumped case, PWLD requires significantly fewer iterations to converge, and this effect is more pronounced as the number of cells in the meshes increases.

In general, the results of the truncation-error test problems for the optically thick limit are particularly good. We have shown that the PWLD method is just as accurate as the TRILD method for a problem on which TRILD appeared *a priori* to have a slight advantage. Furthermore, for the unlumped case, PWLD requires significantly fewer iterations than TRILD, and for the lumped version, the two methods require about the same number of iterations. These results indicate that the PWLD method is both robust and accurate in the thick diffusion limit. This strongly suggests that PWLD will be a successful spatial discretization of the transport equation on arbitrary polyhedral grids for radiative-transfer problems.

Also, we observe that the lumping-parameter methods behave as we have previously predicted. The purpose of the lumping parameter is to lump a method in the thick diffusion limit when it is necessary for robustness, and to unlump the method in the thin limit to retain the accuracy of the unlumped method in this limit. Our lumping parameter methods behaved as accurately and robustly as the lumped methods in the thick limit, and had comparable accuracy to the unlumped method in the thin limit, although the difference in error between all three method types was almost indistinguishable for our thin test problem. These results indicate that using lumping parameters has potential to retain accuracy in the thin limit and robustness in the thick limit.

Summary

The purpose of this chapter is to present a brief overview of our implementation of the PWLD method into the Texas A&M University PDT code and discuss the results of a variety of test problems developed to characterize the method and compare it to TRILD. These test problems serve the secondary purpose of uncovering strengths and weaknesses of the PDT code.

The results of our test problems are reassuring. We first show that PWLD and TRILD exactly reproduced a problem that has a spatially linear solution. This result is expected and we shall use it as a unit test for discretization implementation in the PDT code. We then develop a one-cell test problem that we use to show PWLD's robustness on greatly distorted cells.

The next series of test problems are truncation-error test problems for which we developed a family of manufactured quadratic solutions that are designed to be favorable to TRILD due to their inclusion of spatial cross terms. We run an optically thin version of this problem and notice that PWLD produced the same second-order behavior and produced the same magnitude of error as TRILD. We then run an optically thick and diffusive version and observe that for some problems PWLD performs slightly more accurately and often more robustly that TRILD in this limit. This result not only confirms our thick diffusion limit analysis from previous chapters, it also indicates that the PWLD method will be a particularly successful discretization of the radiation transport equation.

CHAPTER VII
CONCLUSION AND FUTURE WORK

In this dissertation, we have introduced the Piecewise Linear Discontinuous Finite Element spatial discretization of the RZ and XYZ transport equation for arbitrary polygonal and polyhedral grids, respectively. The target application of this research is radiative transfer problems, which are important for inertial confinement fusion systems, astrophysical studies, and laboratory experiments. A multi-physics approach is necessary to accurately model these systems. In this multi-physics approach, we model the movement of the background material using the hydrodynamics equations (of which the Euler equations are one model), and the movement of the radiation using the radiative transfer equations, where the radiative transfer equations consist of a radiation transport equation and an energy balance equation. In many applications of practical interest there are regions of these systems that are optically thick and diffusive. As a result, any radiation transport method developed simulating such systems must perform well in the thick diffusion limit if it is to accurately model the problem.

After describing the physical system, we briefly described standard time, energy, and angular discretizations of the transport equation that result in a steady-state, one group, spatially dependent transport equation along one quadrature direction. We applied our PWLD method to the RZ transport equation, performed a thick diffusion limit analysis of our method, and ran multiple test problems for our method, comparing it to both the Linear Discontinuous Finite Element method on triangular grids and the Bi-linear Discontinuous Finite Element method on quadrilateral grids. From our analysis and test problems, we found that the PWLD method produces the exact same solution as LD on triangles. This result agrees with our prediction, and also helps to verify our code. We then compared the PWLD RZ method to BLD for a variety of problems. In the thin limit, we found that PWLD and BLD both have second-order truncation error behavior. Furthermore, the magnitude of the error for both methods is similar. As a result, PWLD

produces essentially the same level of accuracy as BLD in the thin limit. Based on the results of the diffusion limit analysis, we tested lumped versions of the PWLD and BLD methods in the thick diffusion limit. Again, we found second-order accuracy behavior, where both methods' accuracy has the same order of magnitude. As a result of our analysis and test problems, we claim that the PWLD method applied to the RZ transport equation is an accurate and robust spatial discretization for radiative-transfer problems on quadrilateral grids and, based on the asymptotic analysis and the theory underlying the PWL basis functions, we expect this method to perform well on arbitrary polygonal grids.

The next section of the dissertation focuses on a PWLD discretization of the XYZ transport equation. We began this discussion by deriving the PWLD method for XYZ geometry, paying careful attention to how we handle faces with non-co-planar vertices. We then performed an asymptotic analysis, which predicts that a lumped version of the PWLD method will perform well in the thick diffusion limit. To fully test our method, we compared it against a Tri-linear Discontinuous Finite Element discretization (TRILD) on hexahedral cells. Most of our test problems involve a family of quadratic manufactured solution problems, which favors the TRILD method due to its inclusion of spatial cross terms, which TRILD solves exactly but PWLD does not. In the thin limit, we found that the PWLD and TRILD methods have similar behaviors for unlumped, lumped, and lumping-parameter versions on orthogonal, random, and re-entrant grids. In the thick limit, we found the same general behavior that PWLD compared well with TRILD on the quadratic problem. As a result of the thick diffusion limit analysis and the test problems, we conclude that the PWLD method is a robust and accurate spatial discretization of the XYZ transport equation for radiative-transfer problems on hexahedral grids and, based on the asymptotic analysis and the theory underlying the PWL basis functions, we expect these results to hold for arbitrary polyhedral grids.

Future work: timing studies in PDT

The scope of this research did not include an analysis of the computational efficiency of the PWLD method. As a result, a next step in this research is to study the run time difference between PWLD and TRILD and the run time of lumping-parameter methods compared to the unlumped and lumped methods. It is possible to precompute many pieces of the single cell matrix and combine these pieces in each cell for each quadrature direction and energy group when needed. Studying the computational impact of this precomputation is potentially valuable, for it has not been studied previously for transport solutions in the radiation-hydrodynamics setting. A study of the efficiency of the parallelization algorithms applied to our new spatial discretizations would also be of interest.

Future work: using LD as a preconditioner for solving the PWLD system

During the course of this research we have developed two promising new ideas to study. One of our goals in designing methods is computational efficiency. We note in the development of our spatial discretizations that PWLD will probably not be fast as the existing LD, BLD, or TRILD methods, but that it provides accuracy on polygonal and polyhedral grids by performing well in the thick diffusion limit. As a result, a method that could adapt itself to be LD when the problem is not diffusive and PWLD when it is diffusive could improve the overall computational efficiency of the PWLD method, while retaining the accuracy of the method. While thinking about this problem, we note that Stone and Adams have some success in formulating an approximate lumped PWLD method as a corner balance method using the LD solution to approximate certain quantities on corners [4]. This has led us to explore the idea of iteratively solving the PWLD equations within a cell using the LD equations for an initial guess, a preconditioner, or both. It is also possible to use the UCB equations as a preconditioner in an iterative solution of the PWLD equations within a cell. One advantage of these approaches is that they avoid solving the N×N PWLD matrix in the cell. Another is that

they could form the basis for an adaptive method that accepts the LD solution when it is accurate but iterates to obtain the PWLD solution when it is needed.

We begin with some definitions. The PWLD single cell matrix is *P*, an N×N matrix, where N is the number of unknowns or vertices in the cell. We also define a matrix *L* to be an N×N block matrix where the LD single cell matrix is the upper left block of *L*, the identity is the lower right block of *L*, and the rest of *L* is zero.

$$L = \begin{bmatrix} \Gamma_{LD} & 0 \\ 0 & I \end{bmatrix} \tag{7.1}$$

The difference between *P* and *L* is *B*. As a result

$$P = L + B\,. \tag{7.2}$$

We now write the linear system in the cell as we normally solve it.

$$P\psi = S \tag{7.3}$$

We note that the matrix *P* is in general a full N×N matrix and the solution of Eq. (7.3) is usually found with a full matrix inversion (or LU decomposition followed by forward and back substitution). We also note that the linear system generated by the LD method is the same cell is given by

$$\Gamma_{LD}\psi_{LD} = q_{LD}\,. \tag{7.4}$$

This system also generates a full matrix, but it is only a 3x3 matrix in two dimensions and a 4x4 system in three dimensions.

Using Eq. (7.2), we rewrite Eq. (7.3) as

$$\left(L + B\right)\psi = S\,. \tag{7.5}$$

We also rewrite the source as the sum of an LD-like source and a remainder,

$$S = S_{LD} + S_{\text{Rem}} \tag{7.6}$$

where

$$\begin{aligned} S_{LD} &= \begin{bmatrix} q_{LD} \\ 0 \end{bmatrix} \\ S_{\text{Rem}} &= S - S_{LD} \end{aligned} \tag{7.7}$$

We substitute this new source vector into Eq. (7.5), resulting in

$$\left(L + B\right)\psi = S_{LD} + S_{\text{Rem}}\,. \tag{7.8}$$

We now note that

$$\psi_{LD} = L^{-1} S_{LD} \tag{7.9}$$

and

$$S - P\psi_{LD} = r_{LD} \tag{7.10}$$

where r_{LD} is the residual generated by just the LD solution of the angular flux. The solution to Eq. (7.9) can be generated by solving a 3x3 system in 2D and a 4x4 system in 3D. For 2D, the result is a three-element vector that can be used to construct a solution plane over the cell. The actual angular flux solution vector for the cell, which has N elements, where N is the number of vertices in the cell, can then be determined by evaluating the planar solution at the necessary x,y points. In 3D, the angular flux unknowns can be determined similarly. Using the N-element LD solution vector, we can solve for the residual using the LD solution given by Eq. (7.10). All we need for this residual calculation is the *P* matrix, which is the original PWLD matrix. We can also solve for the LD residual by combining Eqs. (7.6), (7.9), and (7.10), resulting in

$$\begin{aligned} &S - P\left(L^{-1}S_{LD}\right) = r_{LD}, \\ &S_{\text{Rem}} + S_{LD} - \left(L + B\right)\left(L^{-1}S_{LD}\right) = r_{LD}, \\ &S_{\text{Rem}} + S_{LD} - LL^{-1}S_{LD} - BL^{-1}S_{LD} = r_{LD}, \\ &S_{\text{Rem}} - B\psi_{LD} = r_{LD}. \end{aligned} \tag{7.11}$$

Eq. (7.11) requires the formation of only the *B* matrix in the cell. As we have noted previously, it can be much less computationally expensive to generate the LD angular flux solution (especially in 3D on non-tetrahedral grids), and we can now determine if this solution is accurate enough by comparing the residual generated in either Eq. (7.10) or Eq. (7.11) to the source in the cell.

If we determine that the LD residual is not small enough, indicating the LD solution is not accurate enough, we can then apply known iterative methods with the LD solution as the first guess to find a more accurate estimate of the angular flux unknown vector. One possible iterative method would be to add and subtract a lower triangular matrix, *C*, from the LHS of Eq. (7.3).

$$\left(P + C - C\right)\psi = S \tag{7.12}$$

We now isolate the lower triangular matrix on the LHS of Eq. (7.12) so that we can solve the linear system by sweeping the cell.

$$C\psi = \left(C - P\right)\psi + S \tag{7.13}$$

We assume that the angular flux on the RHS of Eq. (7.13) is known from the previous iteration: as a result the iterative method is written as

$$C\psi^{(k)} = \left(C - P\right)\psi^{(k-1)} + S\,. \tag{7.14}$$

This is a Gauss-Seidel iteration. In our case, we assume that $\psi^{(0)}$ is actually ψ_{LD}, and from this initial guess, we generate an improved approximation to the PWLD solution with one iteration:

$$C\psi_{approx} = (C - P)\psi_{LD} + S . \tag{7.15}$$

C could correspond to UCB equations, because UCB generates a lower-triangular system that approximates the PWLD system. Thus, the equations above could correspond to an LD solution followed by a UCB solution. We emphasize that this approach uses UCB to generate a *correction* ($\psi_{approx} - \psi_{LD}$) to the LD solution, not the solution ψ_{approx} itself. If the LD solution is a reasonable approximation, which in most cases it should be, then this procedure should produce a more accurate solution than standard UCB.

By casting our PWLD solution methodology as a matrix iteration, we can see how to generate a better approximate solution using previously developed iterative methods. Above we used the LD solution to generate an initial guess for a Gauss-Seidel iteration step. We suggest that in many cases, one step in this iterative method will produce an "accurate enough" PWLD solution. This would mean we have obtained a PWLD solution via a combination of LD and UCB solutions. However, further iterations can be executed to obtain the PWLD solution even if one step is not sufficient. Another way to approach this problem is to use the *L* matrix as a preconditioner for a GMRES iterative method and the r_{LD} found in Eq. (7.11) as the initial residual for this iterative method. (This is equivalent to replacing *C* by *L*.) It is not clear, though, that applying GMRES in this context will lead to a more efficient matrix solution because GMRES involves some unpredictability in how the solution converges.

The overall result of solving the PWLD method this way is to allow for a simpler, cheaper LD approximation when it is valid. However, if it is not valid, we can still use this LD solution (and possibly UCB equations) to help us more quickly solve for the

PWLD solution. This idea provides us with a roadmap for developing more computationally efficient "piecewise" DFEMs for arbitrary polygonal and polyhedral cells.

Future work: new "piecewise" basis functions

A second suggested extension to this research is the development of Piecewise Bi-linear and Piecewise Tri-linear basis functions. These basis functions are built in the same way as the PWL basis functions, except the subcell volume we use to construct the approximation is the "corner" instead of the "side." In 2D a corner is defined as the quadrilateral area inside an arbitrary polygonal cell that is formed by a cell vertex, the two cell edge midpoints connected to that vertex, and the cell center. Figure 7.1 shows an example of a corner in 2D.

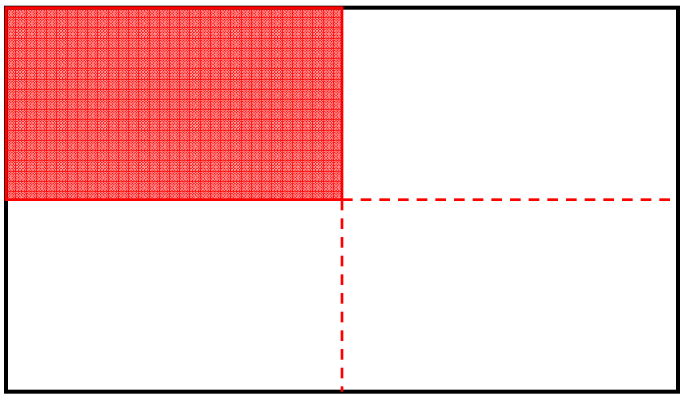

Figure 7.1: A corner subcell volume in two dimensions

We map this corner to a reference square and solve the integrals of the standard bi-linear equations over the corner. Then we add the contributions of the corner integrals to the whole cell matrix in a way that we preserve essential properties of the PWBLD basis functions (the basis functions have to be linearly independent and have to span the solution space). For example, in 2D, the PWBLD basis functions are

$$u_j(x,y) = b_j(x,y) + \sum_{\substack{e=edge \\ Edges\ touching\ j=2}} \Gamma_{e,j} b_e(x,y) + \beta_j b_c(x,y), \tag{7.16}$$

where Γ and β are interpolating coefficients that add up to 1.

$$\begin{aligned} \sum_{\substack{k=corner \\ in\ edge\ e}} \Gamma_{e,k} &= 1 \\ \sum_{k=corner} \beta_k &= 1 \end{aligned} \tag{7.17}$$

For previously developed codes with corner data structures, the PWBLD method should be relatively simple to implement, analogous to the implementation of the PWLD in Capsaicin.

We can potentially further extend this idea to three dimensions and develop Piecewise Tri-linear basis functions in the same way. However, we encounter some problems with this development because in three dimensions not all subcell corner volumes are hexahedral in shape. A corner in three dimensions is the volume created by a vertex, mid-points of all edges connected to that vertex, mid-points of all faces connected to that vertex, and a cell center point. Because more that three edges can be connected to a vertex in 3D cells, the corners are not guaranteed to be hexahedral shapes. In order to satisfy the diffusion limit, we believe that we need to map tri-linear functions to hexahedral shaped corners to produce the surface matching properties necessary for DFEMs in this limit. If a corner has more than six faces, most likely, the method will fail in the diffusion limit. Consequently, developing a PWTLD is still not well-defined. We do know that the corners that have more than six faces can easily be divided into hexahedral "subcorners" and we can then map the tri-linear functions to these subcorners. Again, in each cell we will only have an *N*x*N* matrix to invert, where *N* is the number of vertices in the cell, but building the single cell matrix can become expensive if the number of subcorners is large. If we find that matrix precomputation is

computationally efficient, the added expense of building the single cell matrix with subcorners will not be significant. As a result, we believe we can write a PWTLD method for the XYZ transport equation. An open question is how to define the subcorner volumes. The potential benefit of the PWBLD and PWTLD methods are additional accuracy due to more curvature in the approximation space of the basis functions. A further benefit in 3D is the potential for additional computational efficiency. In a hexahedral cell, the PWLD matrix is formed by looping over 24 sides. If we develop the PWTLD method, we loop only over eight corners to form the matrix. The number of computations per subcell unit is higher for PWTLD, but as long as we reduce this number to less than a factor of three, PWTLD has the potential to be more accurate less expensive than PWLD.

We make one final observation about this family of "Piecewise" methods. Obtaining acceptable performance in the diffusion limit on arbitrary polygonal and polyhedral grids has proven to be a difficult task with standard basis functions that existed previously. The theory developed by Adams [6] says that DFEMs based on the "Piecewise" functions studied here will perform well in the diffusion limit on such grids because the basis functions have the required localization and surface-matching properties. Furthermore, because these methods belong to the finite element family, they are supported by a great deal of pre-existing mathematical theory. Finally, because these methods involve simple functions, all necessary integrals are easily calculated analytically. To our knowledge, no other methods for arbitrary polygonal and polyhedral cells perform well in the diffusion limit and have this mixture of calculational simplicity and mathematical support.

REFERENCES

1. G.C. Pomraning, The Equations of Radiation Hydrodynamics, Dover Publications, Inc., Mineola, NY, 2005.

2. E.E. Lewis, W.F. Miller, Computational Methods of Neutron Transport, American Nuclear Society, La Grange Park, IL, 1993.

3. H.G. Stone, M.L. Adams, A Piecewise Linear Finite Element Basis with Application to Particle Transport, in: Proceedings American Nuclear Society Topical Meeting Nuclear Mathematical and Computational Sciences Meeting, April 6-11, 2003, Gatlinburg, TN, CD-ROM, 2003.

4. H.G. Stone, M.L. Adams, New Spatial Discretization Methods for Transport on Unstructured Grids, in: Proceedings American Nuclear Society Topical Meeting Mathematics and Computation, Supercomputing, Reactor Physics and Biological Applications, September 12-15, 2005, Avignon, France, CD-ROM, 2005.

5. T.S. Bailey, M.L. Adams, B. Yang, M.R. Zika, A Piecewise Linear Finite Element Discretization of the Diffusion Equation for Arbitrary Polyhedral Grids, in: Proceedings American Nuclear Society Topical Meeting Mathematics and Computation, Supercomputing, Reactor Physics and Biological Applications, September 12-15, 2005, Avignon, France, CD-ROM, 2005.

6. M.L. Adams, Discontinuous Finite Element Transport Solutions in Thick Diffusive Problems, Nucl. Sci. and Eng. 137 (2001) 298-333.

7. T.A. Wareing, J.M. McGhee, J.E. Morel, S.D.Pautz, Discontinuous Finite Element S_N Methods on Three-Dimensional Unstructured Grids, Nucl. Sci. and Eng. 138 (2001) 256-268.

8. E.L. Wachspress, A Rational Finite Element Basis, Academic Press, New York 1975.

9. G.G. Davidson, T.S. Palmer, Finite Element Transport Using Wachspress Rational Basis Functions on Quadrilaterals in Diffusive Regions, in: Proceedings American Nuclear Society Topical Meeting Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications, September 12-15, 2005, Avignon, France, CD-ROM 2005.

10. M.L. Adams, P.F. Nowak, Asymptotic Analysis of a Computational Method for Time- and Frequency- Dependent Radiative Transfer, J. Comput. Phys. 146 (1998) 366-403.

11. K. Thompson, M.L. Adams, A Spatial Discretization for Solving the Transport Equation on Unstructured Grids of Polyhedra, in: Proceedings American Nuclear Society Topical Meeting Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications, September, 1999, Madrid, Spain 1999.

12. M.L. Adams, Subcell Balance Methods for Radiative Transfer on Arbitrary Grids, Transport Theory Statist. Phys. 26 (1997) 385-431.

13. M.L. Adams, T.A. Wareing, W.F. Walters, Characteristics Methods in Thick Diffusive Problems, Nucl. Sci. and Eng. 130 (1998) 18-46.

14. J.E. Morel, NUEN 689 Special Topics: Radiation Hydrodynamics, Class Notes, Texas A&M University, 2006.

15. J.I. Castor, Radiation Hydrodynamics, Cambridge University Press, Cambridge, United Kingdom, 2004.

16. D. Mihalas, B. Weibel-Mihalas, Foundations of Radiation Hydrodynamics, Dover Publications, Inc., Mineola, NY, 1999.

17. E.W. Larsen, G.C. Pomraning, V.C. Badham, Asymptotic analysis of radiative transfer problems, J. Quant. Spectrosc. Radiat. Transfer 29 (1983) 285-310.

18. M.L. Adams, NUEN 630: Computational Methods for Particle-Transport Problems, Class Notes, Texas A&M University, 2004.

19. J.S. Warsa, T.A. Wareing, J.E. Morel, Fully Consistent Diffusion Synthetic Acceleration of Linear Discontinuous S_N Transport Discretizations on Unstructured Tetrahedral Meshes, Nucl. Sci. and Eng. 141 (2002) 236-251.

20. J.S. Warsa, T.A. Wareing, J.E. Morel, Krylov Iterative Methods and the Degraded Effectiveness of Diffusion Synthetic Acceleration for Multidimensional S_N Calculations in Problems with Material Discontinuities, Nucl. Sci. and Eng. 147 (2004) 218-248.

21. T.S. Palmer, Curvilinear Geometry Transport Discretizations in Thick Diffusive Regions, Ph.D. Dissertation, University of Michigan, Ann Arbor, MI, 1993.

22. A. Ern , J.L. Guermond, Theory and Practice of Finite Elements, Springer-Verlag, New York, 2004.

23. T.S. Bailey, K. Thompson, A Piecewise Linear Finite Element Discretization of the RZ Transport Equation, Los Alamos National Laboratory Research Note, CCS-4, 2006.

24. J.E. Morel, J.S. Warsa, A Lumped Bilinear Discontinuous S_n Spatial Discretization for R-Z Quadrilateral Meshes, Los Alamos National Laboratory Research Note, CCS-4.

25. J.H. Chang, Los Alamos National Laboratory, Private Communication, 2006-2007.

26. T.S. Palmer, Discretizing the diffusion equation on unstructured polygonal meshes in two dimensions, Annals of Nuclear Energy 28 (2000) 1851-1880.

27. T.S. Palmer, A Point-Centered Diffusion Differencing for Unstructured Meshes in 3-D, in: Proceedings International Conference on Mathematics and

Computations, Reactor Physics and Environmental Analyses, April 30-May 4, 1995, Portland, OR, 1995.

28. J.H. Chang, Efficient Algorithms for Discrete-Ordinates Transport Iterations in Massively Parallel Computers, Ph.D. Dissertation, Texas A&M University, College Station, TX, 2004.

VITA

Name:	Teresa S. Bailey
Address:	7000 East Avenue Lawrence Livermore National Laboratory Livermore, CA 94550
Email Address:	baileyte@tamu.edu, bailey42@llnl.gov
Education:	Ph.D., Nuclear Engineering, Texas A&M University, 2008 M.S., Nuclear Engineering, Texas A&M University, 2006 B.S., Nuclear Engineering, Oregon State University, 2002