RESOLUTION REQUIREMENTS FOR ACCURATE REACTOR CALCULATIONS USING THE LINEAR DISCONTINUOUS FINITE ELEMENT METHOD

A Dissertation

by

CAROLYN NICOLE MCGRAW

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

DOCTOR OF PHILOSOPHY

Chair of Committee,	Marvin Adams
Committee Members,	Jim Morel
	Jean Ragusa
	Nancy Amato
Head of Department,	Michael Nastasi

May 2020

Major Subject: Nuclear Engineering

ABSTRACT

This work seeks to ascertain the accuracy of the Linear Discontinuous Finite Element Method (LD) for the spatial variable in realistic neutron-transport calculations for nuclear reactors. Our results have shown that LD is an excellent alternative to other methods, such as the Method of Characteristics, for computing these types of problems.

We have implemented the Linear Discontinuous Finite Element Method (LD) in the massively parallel transport code PDT. We ran problems with a range of spatial, angular, and energy discretization choices. We then analyzed the spatial accuracy in these problems. From this analysis, we have determined spatial resolution requirements necessary when calculating accurate solutions using LD. For a typical 2D PWR reactor assembly, using 76 spatial cells per pin cell, we achieve an accuracy of 3 pcm in k-effective. Also we achieve an accuracy of 3 pcm in the pin power we see in the 4 fuel pins we looked at.

We have also developed an error model that accurately predicts quantities of interest at infinite resolution. This model treats both spatial and angular discretization error for a given energy discretization. The model is 'trained' using a series of training points - a range of spatial and angular discretization choices and the corresponding QOI. The model then uses a least squares approach to fit the QOIs as a function of our discretization choices. The model quantifies the error in our computations when the mesh is not infinitely refined. This is important for not only predicting solutions to large scale problems that cannot be run but also for quantifying the accuracy of solutions to problems that can.

We have also compared to the Method of Characteristics (MOC) for a range of problems in 2D and 3D. Through our collaboration with researchers from the University of Michigan and their MOC code MPACT, we have determined that for problems with geometric features or boundary layers that are extremely small relative to the problem domain, LD achieves higher accuracy with similar meshes when compared to MOC. This is due to the fact that the Method of Characteristics must use small track spacing to accurately resolve fine mesh attributes. This leads to large numbers of unknowns, compared to DFEMs, to compute solutions to problems with similar levels of accuracy.

Finally we have observed the phenomenon of unphysical oscillations that appear in LD solutions to k-eigenvalue problems in which both the problem domain and the spatial cells have high aspect ratios. We have explained why and for what types of meshes these oscillations occur. We can now predict, for simple problems, both the scalar flux shape and the k-eigenvalue for problems with and without unphysical oscillations.

DEDICATION

To Ruby.

ACKNOWLEDGMENTS

I would first like to thank my colleagues in the Nuclear Engineering Department for their wonderful collaboration and help in keeping me sane. You supported me greatly and were always willing to help me. In particular, I would like to thank Daryl Hawkins for serving as my personal PDT expert. You have answered so many of my panicked questions, and this work would not have been possible without your help.

I would also like to single out my chair, Marvin Adams. Dr. Adams, I want to thank you for your excellent guidance and for all of the opportunities I was given to conduct my research and further my dissertation at A&M.

I would also like to thank my committee members: Jean Ragusa, Jim Morel, and Nancy Amato. Your comments and expertise helped greatly in the guiding of this research.

Finally I would like to thank my friends in Los Alamos. You all helped me stay sane while finishing my research working from home. Specifically, thank you Simon and Madeline for lending me your puppy to watch. Her energy helped keep my spirits up on cold winter days of writing and waiting in the queues.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Dr. Marvin Adams [advisor] and Drs. Jean Ragusa and Jim Morel of the Department of Nuclear Engineering and Dr. Nancy Amato of the Department of Computer Science.

The analyses depicted in Chapter 5 were conducted in part by Brendan Kochunas of the University of Michigan. All other work conducted for the dissertation was completed by the student independently.

Funding Sources

This material is based upon work supported by the Department of Energy, National Nuclear Security Administration, under Award Number(s) DE-NA0002376. Established by Congress in 2000, NNSA is a semi-autonomous agency within the U.S. Department of Energy responsible for enhancing national security through the military application of nuclear science. NNSA maintains and enhances the safety, security, reliability and performance of the U.S. nuclear weapons stockpile without nuclear testing; works to reduce global danger from weapons of mass destruction; provides the U.S. Navy with safe and effective nuclear propulsion; and responds to nuclear and radiological emergencies in the U.S. and abroad.

NOMENCLATURE

1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional
BOL	Beginning of Life
CASL	Consortium for the Advanced Simulation of Light Water Reactors
DFEM	Discontinuous Finite Element Method
FEDS	Finite Element Discontinuous Support
IFBA	Integral Fuel Burnable Absorber
LD	Linear Discontinuous Finite Element Method
MG	Multi-Group
MOC	Method of Characteristics
MPACT	Michigan Parallel Characteristics Transport Code
MPI	Message Passing Interface
OECD	Organisation for Economic Co-operation and Development
OpenMP	Open Multi-Processing
RERTR	Reduced Enrichment for Research and Test Reactors
TAMU	Texas A&M University
TRIGA	Training, Research, Isotopes, General Atomics
S_N	Discrete Ordinates
STAPL	Standard Template Adaptive Parallel Library
VERA	Virtual Environment for Reactor Applications

TABLE OF CONTENTS

Page

ABSTRACT ii

DEDICATION iv

ACKNOWLEDGMENTS v

CONTRIBUTORS AND FUNDING SOURCES vi

NOMENCLATURE vii

TABLE OF CONTENTS viii

LIST OF FIGURES x

LIST OF TABLES xiii

1. INTRODUCTION AND LITERATURE REVIEW 1
    - 1.1 Introduction 1
    - 1.2 Literature Review 2
    - 1.3 Finite Element Methods 3

2. SIMPLIFIED REACTOR MODEL 11
    - 2.1 C5G7 Specification 11
    - 2.2 PDT Model of C5G7 Benchmark Problem 13
    - 2.3 Truncation Error Model 17
    - 2.4 Results 19
        - 2.4.1 2D 19
        - 2.4.2 3D 23
    - 2.5 Comparing to Spider Web Grids 29
    - 2.6 Conclusions 29

3. INTERACTION BETWEEN SPACE AND ANGLE 30
    - 3.1 Single Pin Cell Problems 30
        - 3.1.1 Orthogonal Mesh 30
        - 3.1.2 Spider Web Mesh 32
            - 3.1.2.1 Separating Fuel and Moderator 34
        - 3.1.3 Modeling Space and Angle 38

3.2 Assembly Problems 40
3.3 Conclusions 41

4. ENERGY DISCRETIZATION 43

4.1 Single Pin Cell 43
4.2 2D Assembly 51
4.3 Conclusions 54

5. COMPARISON TO THE METHOD OF CHARACTERISTICS 55

5.1 Geometry Comparison 55
5.1.1 DFEMs 56
5.1.2 Method of Characteristics 59
5.1.3 Results 61
5.2 Numerical Comparison 62
5.2.1 Pin Cell Calculations 62
5.2.1.1 51 Groups 63
5.2.1.2 252 Group - MultiGroup 66
5.2.2 2D Assembly Calculations 68
5.2.2.1 2D C5G7 68
5.3 Conclusions 70

6. OSCILLATIONS 71

6.1 LD Equations 71
6.2 Test Problem 72
6.3 Ansatz 80
6.3.1 Scalar Fluxes and Net Current Densities 81
6.3.2 Cell-Edge Quantities 85
6.3.3 X-Slope Terms 89
6.3.4 Zeroth and First Angular Moments 90
6.3.5 X-Slope Terms 92
6.3.6 Collect and Solve 93
6.4 Results 97
6.5 Conclusions 105

7. SUMMARY 106

REFERENCES 108

LIST OF FIGURES

FIGURE Page

1.1 Illustration of Linear Galerkin DFEM Solution 6

2.1 Assembly Layout of C5G7 Benchmark Problem 12

2.2 Top-Down and Two Side Layouts of the 3D C5G7 Benchmark Problem 12

2.3 Meshes Used for Pin Cells, Ranging from (3,1) to (10,5), Where (n,m) Means n Rings in Fuel and m in Coolant. 14

2.4 A Portion of the (3,1) Mesh in the 2D Study Illustrating Fine, Intermediate, and Coarse Regions of the Reflector. 15

2.5 A Portion of the (3,1) Mesh in the 3D Study Illustrating Fine, Intermediate, and Coarse Regions of the Reflector. 16

2.6 Flux Solution for Groups 0, 2, and 5 for the 2D C5G7 20

2.7 Flux Solution for Group 5 on the (10,5) Mesh (left) and (3,1) Mesh (right). 21

2.8 k-effective vs Resolution for 2D 23

2.9 Max Pin Power vs Resolution for 2D 24

2.10 Inner UO2 Assembly Power vs Resolution for 2D 24

2.11 k-effective vs Resolution for 3D 26

2.12 Max Pin Power vs Resolution for 3D 28

2.13 Inner UO2 Assembly Power vs Resolution for 3D 28

3.1 Meshes of Orgothonal Mesh Problem 31

3.2 Relative Difference (in pcm) in k-effective for Orthogonal Mesh Problem 31

3.3 Meshes of Spider Web Mesh Problem 32

3.4 Percent Difference in k-effective for Spider Web Mesh Problem 33

3.5 Spider Web Mesh with 8, Mixed, and 12 Azimuthal Segment per Ring 34

3.6 Percent Difference in k-effective for Azimuthal Segment Study 34

3.7 Meshes with Different Number of Cells in the Fuel 35

3.8 Meshes with Different Number of Cells in the Moderator 36

3.9 Relative Difference (pcm) in k-effective for Different Numbers of Cells in the Fuel (left) and Moderator (right) 36

3.10 Assembly Used in the Space Angle Study 40

4.1 Meshes Used for Energy Study 44

4.2 Meshes Used for Energy Study with 10% outer Fuel Cell 44

4.3 Comparison of (6,3) Mesh with 1 Cell (left) and 2 Cells (right) in Outer 10% 45

4.4 Scalar Flux Solution for 65 Energy Unknowns Without (left) and With (right) Cell in the Outer 10% of the Fuel 46

4.5 Scalar Flux Solution for 244 Energy Unknowns Without (left) and With (right) Outer 10% Cell for Each Mesh 47

4.6 Scalar Flux Solution for 65 Energy Unknowns for (3,2) Mesh Without Outer 10% Cell 47

4.7 Zoom of Scalar Flux solution for 65 Energy Unknowns Cross Sections (left) and 244 Energy Unknowns (right) for Meshes With a Cell in the Outer 10% of the Fuel . 48

4.8 Scalar Flux Solution with (Magenta) and Without (Green) the Outer 10% Cell for the (6,3) Mesh 48

4.9 Layout of the 2D Assembly Problem 51

5.1 Circle and Approximating Polygon used in DFEMs 56

5.2 Zoom in on One 'Side' 57

5.3 Track Spacing in MOC 60

5.4 Comparison of Geometry Error between DFEMs and MOC 61

5.5 Meshes for MOC (left) and DFEMs (right) for Problem 1B (modified) of the VERA Benchmark 63

5.6 Meshes for MOC (left) and DFEMs (right) for Problem 1E of the VERA Benchmark 64

5.7 MOC Low Res (left): 0.03 cm Spacing, and High Res (right): 0.001 cm Spacing Pin Cell Meshes 69

5.8 DFEM C5G7 Pin Cell Mesh 69
6.1 k-effective vs λ for Case 1 98
6.2 Flux Shape for Case 1 99
6.3 k-effective vs λ for Case 2 99
6.4 Flux Shape for Case 2 100
6.5 k-effective vs λ for Case 3 101
6.6 Flux Shape for Case 3 101
6.7 k-effective vs λ for Case 4 102
6.8 Flux Shape for Case 4 102
6.9 k-effective vs λ for Case 5 103
6.10 Flux Shape for Case 5 103
6.11 k-effective vs λ for Case 6 104
6.12 Flux Shape for Case 6 105

LIST OF TABLES

TABLE Page

2.1 2D QOIs 21

2.2 Percent Differences with Respect to MCNP Reference Solutions for 2D Problem 22

2.3 Percent Differences with Respect to PDT Reference Solutions for 2D Problem 22

2.4 Specifications of 2D Spatial Meshes 23

2.5 Parameters and Their Values in our Full Model of k-effective (2D C5G7) 25

2.6 3D QOIs 26

2.7 Percent Differences with Respect to MCNP Reference Solutions for 3D Problem 27

2.8 Percent Differences with Respect to PDT Reference Solutions for 3D Problem 27

2.9 Parameters and Their Values in our Full Model of k-effective (3D C5G7) 29

3.1 Parameters and Their Values in our Full Model of k-effective (Fuel and Moderator) . 37

3.2 Parameters and Their Values in our Full Model of k-effective (Space and Angle) 39

3.3 Parameters and Their Values in our Full Model of k-effective for Assembly Problems 41

4.1 Relative Difference (pcm), Compared to the Benchmark, in k-effective for 65 Energy Unknowns (left) and 244 Energy Unknowns (right) 49

4.2 Relative Difference (pcm), Compared to the Most Refined Case, in k-effective for 65 Energy Unknowns (left) and 244 Energy Unknowns (right) 49

4.3 Percent Absorption and Fission in the Outer 10% of the Fuel for 65 (left) and 244 (right) Energy Unknowns 50

4.4 Percent Relative Difference (in pcm) in Percent Absorption and Percent Fission in the Outer 10% of the Fuel for 65 (left) and 244 (right) Energy Unknowns 50

4.5 Relative Difference (in pcm) in k-effective for the 2D Assembly Problem 52

4.6 Relative Difference (in percent) in Pin Power for Pin A for the 2D Assembly Problem 53

4.7 Relative Difference (in percent) in Pin Power for Pin B for the 2D Assembly Problem 53

4.8 Relative Difference (in percent) in Pin Power for Pin C for the 2D Assembly Problem 53

4.9 Relative Difference (in percent) in Pin Power for Pin D for the 2D Assembly Problem 54

4.10 RMS for the 2D Assembly Problem 54

5.1 k-effective and Relative Difference (pcm) to Benchmark for Problem 1B with 51 Groups 64

5.2 k-effective and Relative Difference (pcm) to Benchmark for Problem 1E with 51 Groups 65

5.3 Absorption Rate for 1B (left) and 1E (right) and Relative Difference to the Benchmark Solution for 51 Groups 66

5.4 ν times Fission Rate for 1B (left) and 1E (right) and Relative Difference to the Benchmark Solution for 51 Groups 66

5.5 k-effective and Relative Difference (pcm) to Benchmark for Problem 1B with 252 Groups 66

5.6 k-effective and Relative Difference (pcm) to Benchmark for Problem 1E with 252 Groups 67

5.7 Absorption Rate for 1B (left) and 1E (right) and Relative Difference to the Benchmark Solution for 252 Groups 67

5.8 ν times the Fission Rate for 1B (left) and 1E (right) and Relative Difference to the Benchmark Solution for 252 Groups 67

5.9 QOI Comparison (Relative Difference in pcm) for the 2D C5G7 Problem between MOC and DFEMs 70

6.1 Details of Test Problems 97

6.2 k-effective Results 104

1. INTRODUCTION AND LITERATURE REVIEW

1.1 Introduction

Neutron transport is the study of the motions and interactions of neutrons with materials. Neutron transport methods fall into two categories: stochastic and deterministic. Stochastic methods use probablities and a random number generator to 'follow' a particle from birth to death and then characterize where that particle, and thousands to millions of other particles, ends up. These methods then take averages of these 'tracks' to determine the characteristics of the system, such as k-effective and reaction rates. Deterministic methods solve a system of mathematical equations to model these systems. Today, with massively parallel computers, deterministic transport methods, such as the Method of Characteristics (MOC) or Discontinous Finite Element Methods (DFEMs), are still under very active development in research institutions throughout the world. The neutron transport equation remains one of the most computationally challenging problems in the world since it depends on the variables of space (three-dimensions), time, direction (two-dimensions), and energy, with the variable of energy spanning vast ranges (from fractions of an eV to several MeV).

One of the primary applications of neutron transport is reactor problems. There exists a significant body of literature concerning the accuracy of the Method of Characteristics applied to nuclear reactors [1], including with meshes that accurately represent the complex heterogeneous geometry [2, 3]. However, little has been published about the performance of discontinuous finite element methods on such problems. The purpose of this work is to study the accuracy of the Linear Discontinuous (LD) Galerkin Finite Element Method on realistic reactor geometries as well as to provide resolution requirements for accurate solutions to these types of problems.

We implemented the Linear Discontinuous Finite Element Method in the PDT massively parallel transport code developed at Texas A&M University. We performed a series of resolution studies on several reactor geometries with varying complexities to determine the contributions that each of our discretization choices has to the error present in our models. Simultaneously, we made comparisons to MOC. Next we studied the phenomenon of unphysical oscillations that appear in certain types of k-eigenvalue problems. Finally we provide an understanding of resolution requirements for accurate reactor analysis, specifically when using the Linear Discontinuous Finite Element Method.

1.2 Literature Review

The thick diffusion limit is characterized by large scattering cross sections and small absorption cross sections. In this limit, the spatial cells are many mean free paths thick. In [4] and [5] Adams et. al. describe the properties of characteristic methods and DFEMs that ensure accurate solutions when cells are many mean free paths thick, namely: locality and mirror-image properties. These two properties are satisfied for LD on triangles in 2D and tetrahedra in 3D, but not for cells with more vertices. Therefore, if LD is applied to rectangles or hexahedra, it will not in general get accurate solutions when the material is highly scattering and the cells are many mean free paths thick. In [4], the authors also conclude that as spatial cells become optically thick and highly scattering, characteristic methods like MOC behave almost exactly like DFEMs.

Reactor problems do not have cells that are not many mean free paths thick for the cross sections of the materials present. This means that this known flaw in LD does not prohibit accurate solutions to reactor problems. The present work did uncover poor LD behavior, as described in Chapter 6, but the cause and phenomenology differ from the behavior exhibited when the cells are many mean free paths thick. This behavior only occurs for combinations of problem geometries and meshes that are unlikely to be encountered in reactor calculations.

In [6], Wang and Ragusa describe the convergence properties of Discontinuous Galerkin Finite Element Methods applied to the spatial variable in transport calculations. They note that the regularity of the solution limits the convergence order of the methods. The higher the regularity, for example in smooth transport solutions, the less the methods are limited. In the preasymptotic range, it is possible to see a convergence rate of $(p + 1)$ where p is the polynomial order of the spatial method. For many problems, the meshes are in the preasymptotic range, and therefore we see a convergence order of $(p+1)$.

Several deterministic codes, including MPACT [3], OpenMOC [7], and CASMO [8] use the Method of Characteristics to model the transport equation. In this method, the integral form of the transport equation is solved along discrete 'tracks' in the discrete ordinates (S_N) quadrature directions. Difficulty arises when computing solutions to problems in three dimensions, as the finite track spacing and increased S_N directions yield large numbers of unknowns to solve. The MPACT code has employed a 2D/1D scheme to overcome such challenges [9]. This scheme decomposes the 3D problem into a 1D axial stack of 2D planes, and links those 2D planes with axial leakages. Another solution, employed by the OpenMOC code, is to employ hierarchical parallelism with MPI and OpenMP [1] to brute force the increase in number of unknowns.

PDT, which is built on STAPL (Standard Template Adaptive Parallel Library) and uses the Finite Element Method to solve the differential form of the transport equation, is designed to scale well to high processor counts. PDT uses discrete-ordinates (S_N) in direction and multi-group (or related methods whose equations are structured like multi-group equestions) in energy to solve steady-state, k-eigenvalue, and time-depedent neutron and gamma transport problems. The STAPL library, developed at Texas A&M University (TAMU), handles the parallel communication, data structures, etc. [10, 11, 12, 13]. Currently PDT uses only across node parallelism, but will soon be able to employ nested, on-node parallelism to increase its parallel efficiency. At present PDT demonstrates 70% parallel efficiency on approximately 768k cores in a weak scaling study in 3D with orthogonal grids [14]. PDT has also demonstrated high parallel efficiency on high-fidelity reactor problems out to 768k cores.

1.3 Finite Element Methods

The principle equation used in nuclear reactor analysis is the neutron transport equation, which is a limit of the Boltzmann transport equation where the particles interact with the background but not with each other. The solution to this equation, the angular neutron flux ψ, lies in a seven dimensional phase space: position (x, y, z), direction ($\hat{\Omega}$ = [ξ, γ]), energy, and time. For the purposes of this work, the energy dimension will be discretized using the multi-group or FEDS (Finite

Element Discontinuous Support) method, and the angular dimensions will be discretized using the discrete-ordinates (S_N) method. The spatially dependent k-eigenvalue form of the neutron transport equation, which is not dependent on time, can then be written as:

$$\hat{\Omega}_m \cdot \vec{\nabla}\psi_m^g(\vec{r}) + \sigma_t^g(\vec{r})\psi_m^g(\vec{r}) = q_{tot}^g(\vec{r}, \hat{\Omega}_m). \tag{1.1}$$

Here $\hat{\Omega}_m$ is the m^{th} discrete direction in the quadrature set, $\psi_m^g(\vec{r})$ is the angular neutron flux in the m^{th} direction in energy group g, and σ_t is the total cross section. The total (scattering plus fission) source, $q_{tot}^g(\vec{r}, \hat{\Omega}_m)$, is defined as

$$q_{tot}^g(\vec{r}, \hat{\Omega}_m) \equiv \underbrace{\sum_{g\prime=1}^{G}\sum_{\ell=0}^{L}\frac{2\ell+1}{4\pi}\sigma_{s\ell}^{g\prime\to g}(\vec{r})\sum_{n=-\ell}^{\ell}\phi_{\ell,n}^{g\prime}(\vec{r})Y_{\ell,n}(\hat{\Omega}_m)}_{\text{scattering source}} + \underbrace{\frac{1}{k}\frac{\chi_g}{4\pi}\sum_{g\prime=1}^{G}\nu^{g\prime}\sigma_f^{g\prime}(\vec{r})\phi^{g\prime}(\vec{r})}_{\text{fission source}}. \tag{1.2}$$

The first term in Eq. 1.2 is the scattering source, which is approximated by a Legendre expansion using spherical harmonics and the second term is the fission source, where k is the eigenvalue of the equation. The angular-flux moments, $\phi_{\ell,n}^{g\prime}(\vec{r})$, are approximated in the discrete-ordinates method using a quadrature set:

$$\phi_{\ell,n}^{g\prime}(\vec{r}) \xrightarrow{\text{discrete-ordinates}} \sum_{m=1}^{M} Y_{\ell,n}^{*}(\hat{\Omega}_m)\psi^{g\prime}(\vec{r}, \hat{\Omega}_m)\omega_m. \tag{1.3}$$

If the scattering cross sections are isotropic, as in the C5G7 benchmark problem, only the 0^{th} moment of the scattering cross section is non-zero and the total source simplifies to:

$$q_{tot}^g(\vec{r}, \hat{\Omega}) \xrightarrow{\text{isotropic}} \sum_{g\prime=1}^{G}\frac{1}{4\pi}\sigma_s^{g\prime\to g}(\vec{r})\phi^{g\prime}(\vec{r}) + \frac{1}{k}\frac{\chi_g}{4\pi}\sum_{g\prime=1}^{G}\nu^{g\prime}\sigma_f^{g\prime}(\vec{r})\phi^{g\prime}(\vec{r}). \tag{1.4}$$

The spatial dimensions are discretized using the discontinous finite element method (DFEM). The standard DFEM spatial discretization [5] is applied to Equation (1.1) by multiplying by a test function, $p_i(\vec{r})$, and integrating over all space:

$$\int d^3r p_i(\vec{r}) \left[\hat{\Omega}_m \cdot \vec{\nabla}\psi^g_m(\vec{r}) + \sigma^g_t(\vec{r})\psi^g_m(\vec{r})\right] = \int d^3r p_i(\vec{r}) \left[q^g_{tot,m}(\vec{r})\right]. \tag{1.5}$$

To prevent the surface discontinuity from becoming a problem, we now integrate by parts:

$$\begin{aligned} -\int d^3r\psi^g_m(\vec{r})\hat{\Omega}_m \cdot \vec{\nabla}p_i(\vec{r}) + \int_{\delta K} d^2r(\hat{\Omega}_m \cdot \hat{n}_K(r))p_i(\vec{r})\psi^g_m(\vec{r}) + \int d^3r p_i(\vec{r})\sigma^g_t(\vec{r})\psi^g_m(\vec{r}) \\ = \int d^3r p_i(\vec{r}) \left[q^g_{tot,m}(\vec{r})\right]. \end{aligned} \tag{1.6}$$

Where δK represents the cell boundary and $\hat{n}_K(r)$ is the outward normal at the cell boundary K. The unknowns are then expanded in terms of finite element basis functions, $b_j(\vec{r})$, each of which goes discontinuously to zero just outside of the cell on which it is defined:

$$\psi^g_m(\vec{r}) = \sum_{j=1}^{N} \psi^g_{m,j} b_j(\vec{r}) \tag{1.7}$$

Plugging this in gives:

$$\begin{aligned} -\int d^3r\psi^g_m b_i(\vec{r})\hat{\Omega}_m \cdot \vec{\nabla}p_i(\vec{r}) + \int_{\delta K} d^2r(\hat{\Omega}_m \cdot n_K(r))p_i(\vec{r})\psi^g_m(\vec{r}) + \int d^3r p_i(\vec{r})\sigma^g_t(\vec{r})\psi^g_m b_i(\vec{r}) \\ = \int d^3r p_i(\vec{r}) \left[q^g_{tot,m} b_i(\vec{r})\right]. \end{aligned} \tag{1.8}$$

Notice that the limits of integration are only one cell. This is due to the fact that the basis functions are only non-zero on one cell. A Galerkin method is one in which the test and basis functions span the same space in every cell. The linear discontinuous basis functions can be defined on any 3D cell as:

$$b_0(\vec{r}) = 1, \qquad b_1(\vec{r}) = \frac{2(x - \overline{x})}{\Delta\overline{x}}, \qquad b_2(\vec{r}) = \frac{2(y - \overline{y})}{\Delta\overline{y}}, \qquad b_3(\vec{r}) = \frac{2(z - \overline{z})}{\Delta\overline{z}}. \tag{1.9}$$

Here $\overline{u}$ is the average value of u in the cell and $\Delta\overline{u}$ is a measure of the span of the cell in the direction u. These basis functions define a solution space that is planar in every cell, but can be discontinuous across cell boundaries. As an illustration, consider Figure 1.1. An isotropic source is incident on the left side of the problem and an S$_2$ quadrature set is used. The scalar flux solution is linear in every cell, but can be discontinuous in magnitude and slope across cell boundaries.

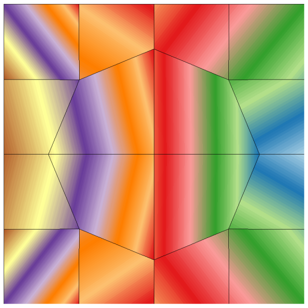

Figure 1.1: Illustration of Linear Galerkin DFEM Solution

For completeness, we will now derive the discretized equations from the continuous ones. For simplicity, we will only be looking at the 1 group 1D equations. Using the finite element method, we move everything over to the left hand side, multiply by a test function, and integrate over a cell:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} dx \left[p_i(x)(\mu\frac{\partial}{\partial x})\psi(x) + \sigma_t\psi(x)p_i(x) - Q(x)p_i(x) \right] = 0 \tag{1.10}$$

Since we're using the Galerkin method, our test function spans the same space as our basis functions:

$$p_{0i} = b_{0i} = 1, \qquad p_{1i} = b_{1i} = 2\frac{x - x_i}{\Delta\overline{x}} \tag{1.11}$$

Here, x_i is the cell center value, and $\Delta\overline{x}$ is defined as $x_{i+1/2} - x_{i-1/2}$. Again, these basis and test functions go discontinuously to zero at the cell edges. We will split up our integral into three terms: a streaming term, a collision term, and a source term. For our first equation, which defines the cell centered value of the angular flux in terms of cell edge values, we use the basis function b_0. Term 1, the streaming term, is then:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} dx \left[p_{0i}(x)(\mu \frac{\partial}{\partial x})\psi(x) \right] \tag{1.12}$$

Since $\psi(x)$ is undefined at the cell edges, we cannot take the derivative of it there. Instead, we will integrate by parts to pass the derivative onto the test function:

$$-\int_{x_{i-1/2}}^{x_{i+1/2}} dx(\mu \frac{\partial p_0(x)}{\partial x})\psi(x) + \mu p_0(x)\psi(x)|_{x_{i-1/2}}^{x_{i+1/2}} \tag{1.13}$$

The first term here goes to zero because the derivative of a constant is zero. The second term goes to:

$$\mu \left(\psi_{i+1/2} - \psi_{i-1/2} \right) \tag{1.14}$$

Here $\psi_{i+1/2}$ and $\psi_{i-1/2}$ are the cell edge values. Moving onto the second term in Eq. 1.10, which is the collision term:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} dx \left[\sigma_t \psi(x) p_{0i}(x) \right] = \sigma_t \int_{x_{i-1/2}}^{x_{i+1/2}} dx b_{0i} \left[\psi_i + 2\frac{x - x_i}{\Delta \overline{x}} \hat{\psi}_i \right] \tag{1.15}$$

We can now replace b_{0i} with 1. Using u-substitution:

$$u = 2\frac{x - x_i}{\Delta \overline{x}} \qquad du = \frac{2}{\Delta \overline{x}} dx$$

Plugging these in we get:

$$\frac{\Delta \overline{x} \sigma_t}{2} \int_{-1}^{1} du \left[\psi_i + u \hat{\psi}_i \right] = \frac{\Delta \overline{x} \sigma_t}{2} \left[\psi_i u|_{-1}^{1} + \frac{1}{2} u^2 \hat{\psi}_i|_{-1}^{1} \right] = \Delta \overline{x} \sigma_t \psi_i \tag{1.16}$$

Similar tricks can be played to get the third term. Putting everything together gives:

$$\boxed{\frac{\mu}{\Delta\overline{x}}\left(\psi_{i+1/2}-\psi_{i-1/2}\right)+\sigma_t\psi_i=Q_i} \tag{1.17}$$

This is the discretized transport equation for the cell centered neutron flux, ψ_i. For our second equation, which will be the discretized transport equation for the neutron flux slopes, we use b_1. Again, we will break up the equation into three terms: a streaming term, a collision term, and a source term. The first term, which is the streaming term, is then:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} dx(\mu\frac{\partial}{\partial x})\psi(x)b_1(x) \tag{1.18}$$

Using integration by parts we get:

$$-\int_{x_{i-1/2}}^{x_{i+1/2}} dx(\mu\frac{\partial b_1(x)}{\partial x})\psi(x)+\mu b_1(x)\psi(x)|_{x_{i-1/2}}^{x_{i+1/2}} \tag{1.19}$$

Splitting this up into two terms, and taking the first part, we get:

$$-\mu\int_{x_{i-1/2}}^{x_{i+1/2}} dx(\frac{2}{\Delta\overline{x}})\left[\psi_i+2\frac{(x-x_i)}{\Delta\overline{x}}\hat{\psi}_i\right] \tag{1.20}$$

Now we use u substitution with $u=2\frac{(x-x_i)}{\Delta\overline{x}}$:

$$-\mu\int_{-1}^{1} du\left[\psi_i+u\hat{\psi}_i\right]=-2\mu\psi_i \tag{1.21}$$

Now taking the second term:

$$\begin{aligned}\mu b_1(x)\psi(x)|_{x_{i-1/2}}^{x_{i+1/2}} &= \mu\left[b_{1,i+1/2}\psi_{i+1/2}-b_{1,i-1/2}\psi_{i-1/2}\right] \\ &= \mu\left[(1)\psi_{i+1/2}-(-1)\psi_{i-1/2}\right] \\ &= \mu\left[\psi_{i+1/2}+\psi_{i-1/2}\right]\end{aligned} \tag{1.22}$$

Here we've defined:

$$\begin{aligned}
\psi_{i+1/2} = \psi_i + \hat{\psi}_i \qquad & i = (i, I-1), \mu > 0 \\
\psi_{i-1/2} = \psi_i - \hat{\psi}_i \qquad & i = (1, I-1), \mu < 0 \\
\psi_{1/2} = \psi_{inc} \qquad & i = 1/2, \mu > 0 \\
\psi_{I+1/2} = \psi_{inc} \qquad & i = I + 1/2, \mu < 0 \qquad (1.23)
\end{aligned}$$

These are the 1D closure relationships for LD. Simply put, it defines the cell edge quantities in terms of cell centered fluxes and cell centered slopes. The last two equations in 1.23 are the boundary conditions at the problem edges. They define the incoming flux at each boundary of the problem. Putting these two together gives term 1:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} dx(\mu\frac{\partial}{\partial x})\psi(x)b_1(x) = \mu\left[\psi_{i+1/2} + \psi_{i-1/2} - 2\psi_i\right] \qquad (1.24)$$

Moving onto term 2, which is the collision term:

$$\begin{aligned}
\int_{x_{i-1/2}}^{x_{i+1/2}} \sigma_t\psi(x)b(x) &= \sigma_t \int_{-1}^{1} \frac{\Delta\overline{x}}{2} u du \left[\psi_i + \hat{\psi}_i\right] && (1.25) \\
&= \frac{\Delta\overline{x}\sigma_t}{2} \int_{-1}^{1} \left[u\psi_i + u^2\hat{\psi}_i\right] du && (1.26) \\
&= \frac{\Delta\overline{x}\sigma_t}{2} \left[\frac{1}{2}u^2\psi_i + \frac{1}{3}u^3\hat{\psi}_i\right] |_{-1}^{1} && (1.27) \\
&= \frac{\Delta\overline{x}\sigma_t}{2} \left[0 + \frac{2}{3}\hat{\psi}_i\right] && (1.28) \\
&= \frac{\Delta\overline{x}}{3}\sigma_t\hat{\psi}_i && (1.29)
\end{aligned}$$

Again, similar tricks can be played to get the source term:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} dxQ(x)b(x) = \frac{\Delta\overline{x}}{3}Q_i^1 \qquad (1.30)$$

Putting everything together, we get:

$$\boxed{\frac{3\mu}{\Delta \overline{x}}\left[\psi_{i+1/2}+\psi_{i-1/2}-2\psi_i\right]+\sigma_t\hat{\psi}_i=\hat{Q}_i} \tag{1.31}$$

Together with equation 1.17 and the closure relations 1.23, these make up the 1D discretized S_N equations in cell i. To solve these, the closure relations are substituted into the two main equations, for each cell, to get equations in terms of cell centered quantities. Due to the closure relations, each cell's solution depends on its upstream cell's solution – cell $(i-1)$ for $\mu > 0$ and cell $(i+1)$ for cell $\mu < 0$. Together, all the cell equations form a matrix, which can be solved with sweeping or other matrix solution techniques.

This is only an example of how to manipulate the analytic equations to derive the discretized ones. The 2D equations applied to rectangles are similarly derived and can be seen below:

$$\mu_m \frac{\psi_{m,i+1/2,j}-\psi_{m,i-1/2,j}}{\Delta x}+\eta_m \frac{\psi_{m,i,j+1/2}-\psi_{m,i,j-1/2}}{\Delta y}+\sigma_t\psi_{mij}=Q_{i,j}\ , \tag{1.32}$$

$$3\mu_m \frac{\psi_{m,i+1/2,j}+\psi_{m,i-1/2,j}-2\psi_{mij}}{\Delta x}+\eta_m \frac{\psi^x_{m,i,j+1/2}-\psi^x_{m,i,j-1/2}}{\Delta y}+\sigma_t\psi^x_{mij}=Q^x_{i,j}\ , \tag{1.33}$$

$$\mu_m \frac{\psi^y_{m,i+1/2,j}-\psi^y_{m,i-1/2,j}}{\Delta x}+3\eta_m \frac{\psi_{m,i,j+1/2}+\psi_{m,i,j-1/2}-2\psi_{mij}}{\Delta y}+\sigma_t\psi^y_{mij}=Q^y_{i,j}\ . \tag{1.34}$$

2. SIMPLIFIED REACTOR MODEL

To start our investigation into LD applied to reactor problems, we looked at a set of simplified benchmark problems: the C5G7 benchmark [15, 16]. This benchmark has 7 energy groups, with the macroscopic cross sections predefined. The cross sections being given simplifies the computation by cutting out a major complication when performing reactor calculations. The benchmark keeps the fuel and moderator unhomogenized, though it does not include a clad or gap. A more detailed description of this benchmark is seen below in Section 2.1.

We modeled the 2D and 3D version of this benchmark for our study. We performed a resolution study in space and angle to determine the accuracy of LD in moderately difficult problems. We also compared to benchmark solutions to show that LD can be used to compute highly accurate solutions.

2.1 C5G7 Specification

The C5G7 is a well-known benchmark problem developed by the Organisation for Economic Co-operation and Development's Nuclear Energy Agency (OECD/NEA). The problem has a configuration of 16 assemblies surrounded by a water reflector region, which can be seen in Figure 2.1. The problem is modeled using reflective boundary conditions on the $-x$ and $+y$ boundaries (and $-z$ for the 3D version) and vacuum boundary conditions on the $+x$ and $-y$ boundaries (and $+z$ for the 3D version). The top-left and bottom-right assemblies have UO_2 fuel while the top-right and bottom-left assemblies contain three different MOX fuel enrichments.

Two side views of the 3D version of the benchmark can be seen in Figure 2.2. The fuel assemblies are broken up into three axial regions. Three control rod configurations are defined in [16] where control rods are inserted from the top and bottom of the core due to the reflecting boundary conditions.

The materials in the benchmark are characterized using 7-group transport-corrected cross sections, generated using the collision probability code DRAGON (G. Marleau, *et al.*) and the WIMS-

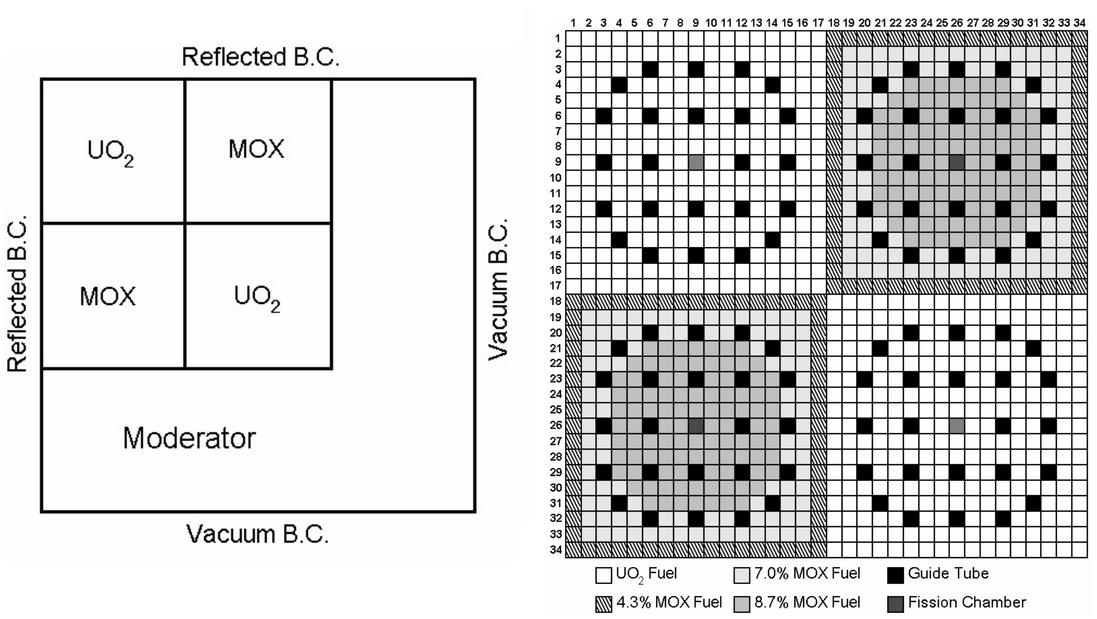

Figure 2.1: Assembly Layout of C5G7 Benchmark Problem

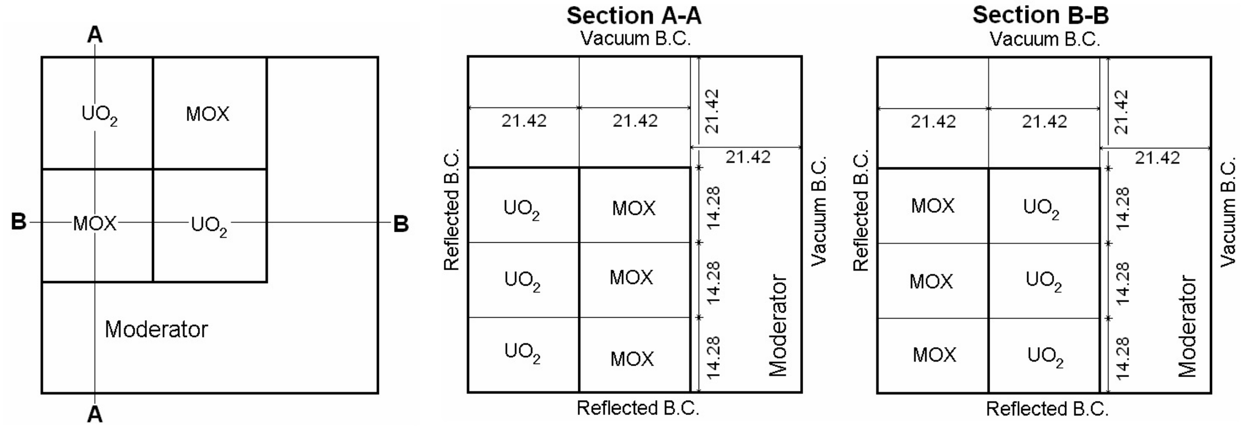

Figure 2.2: Top-Down and Two Side Layouts of the 3D C5G7 Benchmark Problem

AECL 69-group library. The scattering cross sections are isotropic. The fuel rods, fission chamber, guide tubes, and control rods are homogenized, but no fuel-moderator homogenization was performed.

The quantities of interest for the C5G7 are the individual fission rate densities in each fuel pin, the average fission rate densities in each assembly, and k_{eff}. For more details on this benchmark problem see [15] and [16].

2.2 PDT Model of C5G7 Benchmark Problem

Our computational models were developed in the massively parallel transport code PDT, being developed at Texas A&M University. PDT is designed to run on meshes with arbitrary polygonal (2D) or polyhedral (3D) cells. Such meshes can efficiently and accurately represent complex geometries and are well suited to modeling reactor lattices. Examples of meshes in a single pincell can be seen in Figure 2.3. The mesh is refined by adding “rings" to the different material regions. Our pin-cell meshes, shown in Figure 2.3, range from 3 rings in the fuel and 1 in the water, i.e. (3,1), to 10 rings in the fuel and 5 in the moderator, i.e (10,5). At the time of the testing reported here, our mesh-generation capability was limited to $N \times N$ quadrilateral spatial cells per pin cell, with vertices placed to best approximate the geometry. This is not ideal, because it forces the azimuthal refinement to equal the radial refinement. That is, if we want 8 radial rings in the fuel, we must have more than 16 cell edges along each pincell edge. We have since employed more efficient grids, with separately chosen radial and azimuthal resolution.

At the time of this study, our mesh generation capability also required vertices to connect across pin cell boundaries, i.e. no hanging nodes. This means that we must keep the mesh spacing in the water assemblies equal to that in the fuel region in the direction perpindicular to the core, e.g. in the y direction in the water assemblies to the right of the core and in the x direction in the water assemblies below the core. We do allow the mesh spacing to coarsen as we move away from the fuel region, e.g. in the x direction in the water assemblies to the right of the core and in the y direction in the water assemblies below the core. This allows us to coarsen the mesh in regions where the solution is smooth and not changing excessively.

For the 2D problem, we treated the water reflector region as if it were five assemblies of 17×17 pin cells, but we divided each of the resulting "water pin-cells" into simple rectangular mesh cells. We then allowed rows (or columns) of these water pin-cells to coarsen as we move away from the

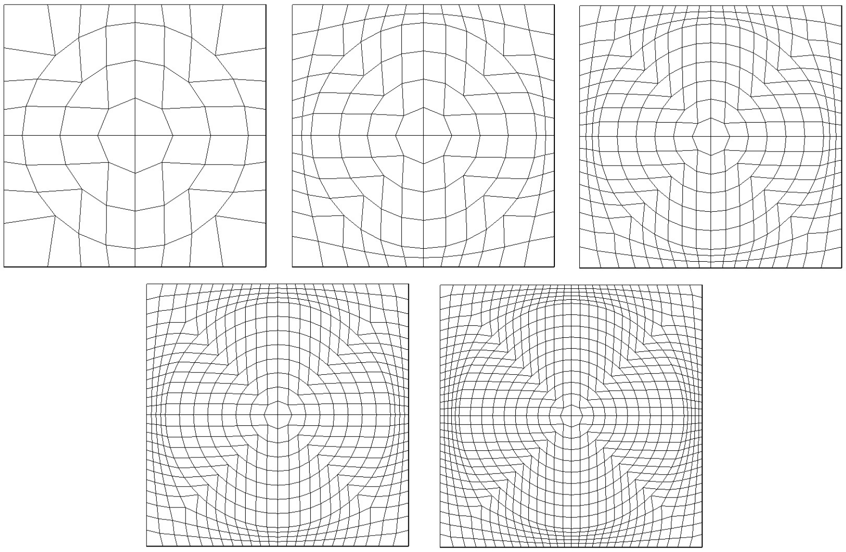

Figure 2.3: Meshes Used for Pin Cells, Ranging from (3,1) to (10,5), Where (n,m) Means n Rings in Fuel and m in Coolant.

fuel region. We defined the 5 to 7 rows (or columns) of water pin-cells next to the fuel region as our fine region, the 5 to 8 rows (or columns) farthest from the fuel region as our coarse region, and the rows (or columns) in the middle as our intermediate region. The fine region had a mesh spacing twice the average spacing in the fuel region and the coarse region had a spacing 4-6 times the average mesh spacing in the fuel. An example reflector mesh for the 2D study can be seen in Figure 2.4.

Our mesh generation capability improved between when we studied the 2D problem and when we studied the 3D problem. For the 2D version of the problem, each pin cell was required to have the same physical dimension, but the number of mesh cells per pin cell could vary pin to pin. For the 3D version, the code required the same number of mesh cells per pin cell, but the pin cells themselves could vary in physical dimension. This allows for more straight forward specification of the water assemblies.

Similarly to the 2D problem, we divided up the reflector for the 3D problem into 5 water assemblies. We then separated the assemblies into three equal regions between the core and each

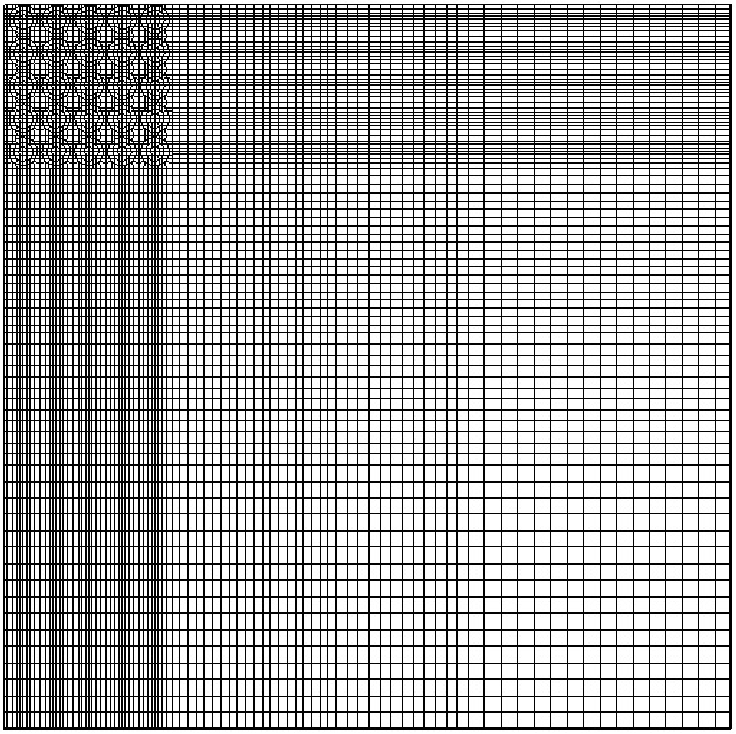

Figure 2.4: A Portion of the (3,1) Mesh in the 2D Study Illustrating Fine, Intermediate, and Coarse Regions of the Reflector.

boundary in the x-y plane. The region closest to the fuel was divided into 3 water-pin-cells, the region in the middle into 2 water-pin-cells, and the outer region into 1 water-pin-cell. This means that the assemblies to the $+x$ direction of the core had a 6×17 pin cell layout, the assemblies to the $-y$ direction of the core had a 17×6 pin cell layout, and the corner water assembly had a 6×6 pin cell layout. Each section then was discretized into simple rectangular mesh cells, but with the same number of cells per water-pin-cell as the fuel region. This results in the water being 3 times finer in resolution near the fuel region than in the periphery of the problem. An example reflector mesh can be seen in Figure 2.5. For both the 2D and 3D problems, the meshes were chosen such that the total problem had a pseudo 40×40 pin cell layout.

In order to make use of the optimal transport sweeping algorithm in PDT we must define sets of cells ("cell-sets") with regular rectangular shapes. Cell-sets, in this context, refer to mesh cells that are grouped together when performing parallel transport sweeps. In the spatial meshes used

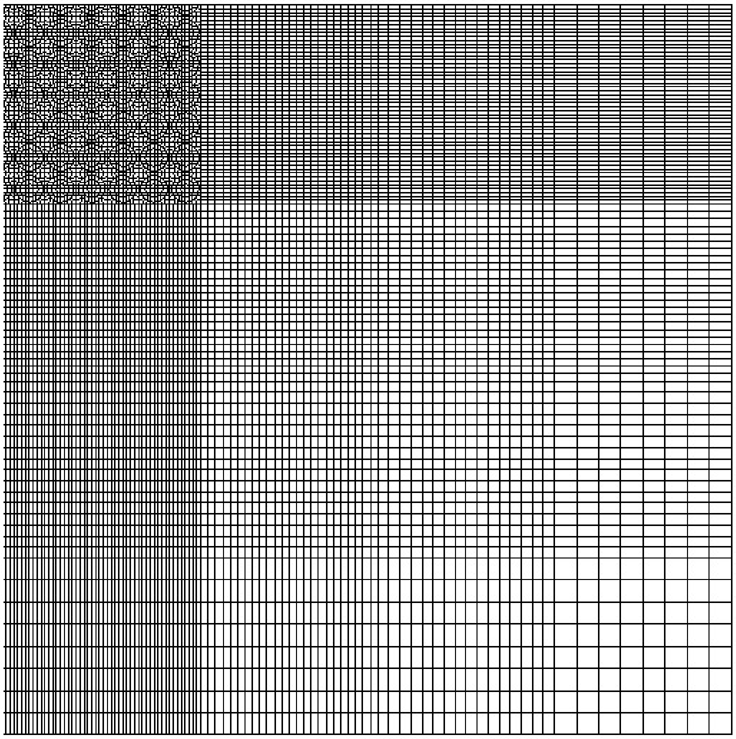

Figure 2.5: A Portion of the (3,1) Mesh in the 3D Study Illustrating Fine, Intermediate, and Coarse Regions of the Reflector.

here, the rectangular requirement means cell-sets cannot be smaller than quarter-pin-cells in the core region. Parallel load balancing is best if each cell-set has the same number of cells, so we impose this requirement as well. With the reflector meshes we have chosen, each of our meshes looks to PDT like an 80×80 array of quarter-pin-cells, which it can aggregate into cell-sets as is appropriate for best parallel performance.

For the axial mesh in the 3D benchmark, we divided the problem into 4 regions: three equally spaced layers in the fuel, as defined by [16], and one layer in the reflector region above the core. Each layer was discretized with an equal number of axial mesh cells ranging from 8 total axial cells to 80 total axial cells. We used a Product-Gauss-Legendre-Chebychev quadrature for all the problems. This means that the total number of angles in our quadrature set is equal to the number of polar angles multiplied by the number of azimuthal angles. The polar angles have a Gauss-Legendre distribution and the azimuthal angles are equally spaced around the x-y plane.

2.3 Truncation Error Model

To determine the truncation error contributed by each discretization choice, we refined each variable (radial spatial resolution, axial resolution, number of polar angles, number of azimuthal angles) individually and studied how the quantities of interest changed. The quantities of interest, or QOI's, for the truncation error analysis were the k-eigenvalue and the pin powers. We can describe each QOI computed as the sum of the exact solution and the errors contributed from our discretizations.

$$\begin{aligned}\text{QOI}(\Delta x, \Delta z, \Delta\xi, \Delta\gamma) = \text{QOI}^{\text{exact}} + C_x\left(\frac{\Delta x}{\Delta x_{max}}\right)^{p_x} + C_z\left(\frac{\Delta z}{\Delta z_{max}}\right)^{p_z} + \\ C_\xi\left(\frac{\Delta\xi}{\Delta\xi_{max}}\right)^{p_\xi} + C_\gamma\left(\frac{\Delta\gamma}{\Delta\gamma_{max}}\right)^{p_\gamma}\end{aligned} \tag{2.1}$$

where:

$$\Delta x = \frac{\text{Pitch}/2}{\text{Fuel Rings + Moderator Rings}}, \tag{2.2}$$

$$\Delta z = \frac{\text{Problem Height}}{\text{Number of } z \text{ planes}}, \tag{2.3}$$

$$\Delta\xi = \frac{1}{\text{Number of Polar Angles}}, \tag{2.4}$$

$$\Delta\gamma = \frac{\pi/2}{\text{Number of Azimuthal Angles per Quadrant}}, \tag{2.5}$$

the p's are the orders by which each error shrinks and the C's are constants for a given QOI. Each variable is normalized to its largest value to better see which variable has the greatest impact on our solution. To determine these p's and C's, we ran a series of test runs where we refined each variable one at a time. We then used all the series to perform a least squares analysis to determine the powers and coefficients. The error present due to a particular discretization choice, E_N, is defined as:

$$E_N = \text{QOI}^{\text{exact}} - \text{QOI}(\Delta x, \Delta z, \Delta\xi, \Delta\gamma) = -C_x\left(\frac{\Delta x}{\Delta x_{max}}\right)^{p_x} - C_z\left(\frac{\Delta z}{\Delta z_{max}}\right)^{p_z} - C_\xi\left(\frac{\Delta\xi}{\Delta\xi_{max}}\right)^{p_\xi} - C_\gamma\left(\frac{\Delta\gamma}{\Delta\gamma_{max}}\right)^{p_\gamma} \tag{2.6}$$

Here the $\text{QOI}^{\text{exact}}$ is computed using our Python script which computes the exact solution, as well as the powers and coefficients in our truncation error model using the data from our series of test runs. Note that the z component of the truncation error model is only present in our 3D results.

Several remarks are in order concerning our error model. First, transport problems are notorious for their lack of smoothness. If a solution has a discontinuous first derivative, then ultimately even a high-order method can achieve no better than first-order truncation error for most QOIs [17]. Even seemingly benign transport problems have solutions with discontinuous first derivatives in space and often discontinuous solutions in angle (across quadrant boundaries, for example). Nevertheless, until meshes get exceedingly fine, high-order methods can achieve much better than first-order error reductions. The problems we consider here are mostly in this limit, with relatively fine but not exceedingly fine meshes.

Second, it is known [18] that spatial truncation error is a function of angular discretization (i.e., a function of the quadrature set in discrete-ordinates methods), and that spatial error can increase when the quadrature set is refined. (As an example, consider a slab-geometry spatial discretization scheme whose accuracy degrades as $\sigma_t \Delta x/|\mu|$ increases, which happens if smaller values of $|\mu|$ are introduced.) In the 2D problems considered here, the analog is increasing chord lengths through cells for directions that approach vertical (along z). Refining the polar quadrature set introduces directions that are closer to vertical, thereby challenging a given spatial discretization. In our error model this translates to a C_x that depends on polar-angle resolution and in fact increases for a sufficiently large number of polar angles. It is possible for polar refinement to produce poorer results if the spatial mesh is not sufficiently resolved. We are aware of this and have studied this interaction in Chapter 3.

2.4 Results

This section is divided into two subsections: 2D and 3D results. In each subsection we first provide tables of the QOI solutions for each case in the resolution study. We then show the truncation error analysis for each of the discretization variables. For reference we have provided the MCNP solution for each of the QOI's, with the 2σ value reported in the benchmark, as well as our finest resolution result calculated using PDT. Our QOI's are the k-eigenvalue, maximum pin power, minimum pin power, and assembly powers for the Inner UO_2, Outer UO_2, and MOX assemblies. Finally we show the relative difference between those QOI's and the MCNP reference solution as well as between them and the finest resolution result from PDT.

2.4.1 2D

We have run a "high-resolution" model of the C5G7 benchmark problem using PDT to generate what we believe is a highly accurate reference solution, in addition to running a series of problems for resolution and error studies. The maximum and minimum pin powers, assembly powers, and the k-eigenvalue for this high-resolution model, as well as the models used in the error study, can be seen in Table 2.1. The high-resolution model used the (10,5) mesh (which has 1.44 million cells), 24 polar angles, and 256 azimuthal angles. All quantities of interest computed using this model are within 2 standard deviations of the MCNP reference results. The relative differences of the QOI's for all the PDT models, as compared to MCNP reference results, can be seen in Table 2.2. Those same QOI's compared to the high-resolution PDT results can be seen in Table 2.3. Earlier versions of Tables 2.1, 2.2, and 2.3 appeared in conference proceedings in [19].

Assuming that our errors can be described by Equation 2.1, the truncation error analysis estimates that our "high resolution" model of the 2D C5G7 has approximately 0.7 pcm of error in the k-eigenvalue and less than 0.004% error in the worst pin power. The pin powers computed by PDT were within 1σ of MCNP pin powers 86.6% of the time (914 pins), and all pin powers were within 2σ, or 160 pcm. The flux solutions computed using the fine resolution PDT model, for Groups 0, 2, and 5, can be seen in Figure 2.6, which were generated using Visit. The color scale for Figure

2.6 is linear.

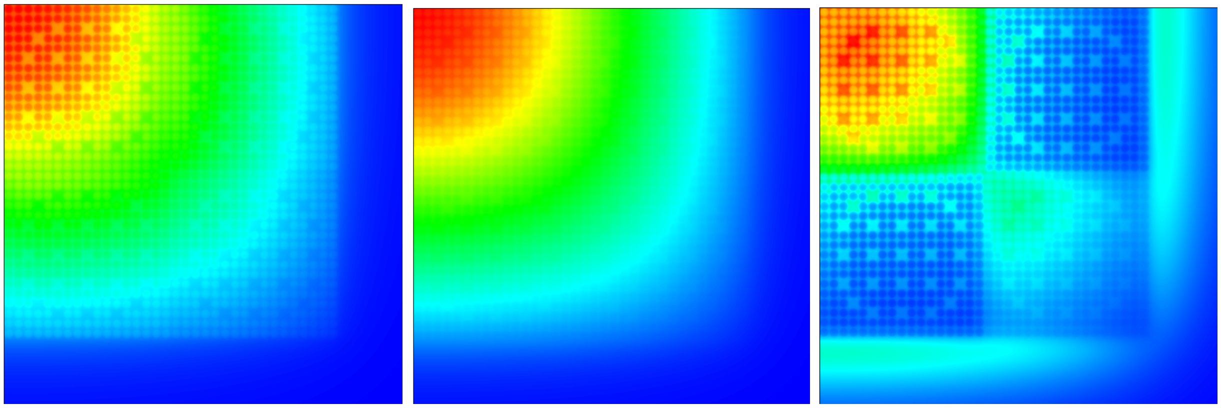

Figure 2.6: Flux Solution for Groups 0, 2, and 5 for the 2D C5G7

Figure 2.7 shows a side by side comparison of our Group 5 flux solutions for the high-resolution mesh (900 cells per pin cell) and a coarser mesh (64 cells per pin cell). The image shows the four corner pins where the four fuel assemblies meet. Close inspection reveals discretization artifacts in the coarse-mesh solution, but this solution is remarkably close to the ultra-fine-mesh solution, showing the robustness and accuracy of the LD method for this kind of problem.

The 2D resolution study included three series: radial mesh, polar angles and azimuthal angles. The radial mesh parameters used in the resolution study can be seen in Table 2.4. The four discretization levels for the polar angle series were 2, 4, 8, and 16 polar angles per quadrant and the five discretization levels of the azimuthal angle series were 8, 16, 32, 64, and 128 azimuthal angles per quadrant. For each series, the variables not being refined were held at the common point: (4,2) spatial mesh, 8 polar angles per quadrant, and 64 azimuthal angles per octant.

The results of our study, for 3 QOI's (k-effective, max pin power, and inner UO_2 assembly power) can be seen in Figures 2.8 - 2.10. The common point ((4,2) mesh, 8 polar angles, and 64 azimuthal angles per quadrant) can be seen where the curves interesect. In addition to the study results, the MCNP benchmark answer, together with its error bars, is presented in each figure. Both

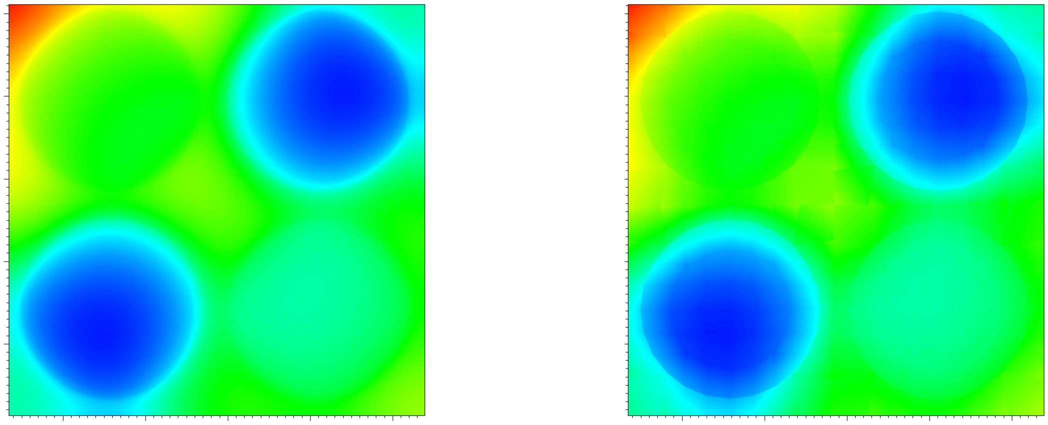

Figure 2.7: Flux Solution for Group 5 on the (10,5) Mesh (left) and (3,1) Mesh (right).

Table 2.1: 2D QOIs

	Mesh	Polar	Azim.	Max Pin	Min Pin	Inner UO2	MOX	Outer UO2	k
MCNP				2.49770	0.23150	492.847	211.695	139.764	1.186550
MCNP 2σ				0.16%	0.58%	0.10%	0.18%	0.20%	0.008%
PDT-fine resolution	(10,5)	24	64	2.49882	0.23173	492.992	211.680	139.647	1.186456
$\frac{\text{PDT-MCNP}}{\text{MCNP}}$				0.03%	-0.12%	0.04%	-0.01%	-0.11%	-0.005%
Spatial	(3,1)	8	64	2.49863	0.23179	492.968	211.689	139.654	1.186620
	(4,2)	8	64	2.49905	0.23174	493.022	211.666	139.646	1.186477
	(6,3)	8	64	2.49912	0.23173	493.032	211.662	139.644	1.186462
	(8,4)	8	64	2.49913	0.23173	493.034	211.661	139.644	1.186458
	(10,5)	8	64	2.49913	0.23173	493.034	211.661	139.643	1.186599
Polar	(4,2)	2	64	2.50539	0.23172	493.754	211.321	139.604	1.186392
	(4,2)	4	64	2.50053	0.23176	493.213	211.575	139.638	1.186394
	(4,2)	8	64	2.49905	0.23174	493.022	211.666	139.646	1.186477
	(4,2)	16	64	2.49878	0.23174	492.984	211.684	139.648	1.186508
Azimuthal	(4,2)	8	16	2.49936	0.23174	493.064	211.648	139.640	1.186376
	(4,2)	8	32	2.49909	0.23174	493.026	211.664	139.645	1.186465
	(4,2)	8	64	2.49905	0.23174	493.022	211.666	139.646	1.186477
	(4,2)	8	128	2.49905	0.23174	493.022	211.666	139.646	1.186478

Table 2.2: Percent Differences with Respect to MCNP Reference Solutions for 2D Problem

	Mesh	Polar	Azimuthal	Max Pin	Min Pin	Inner UO2	MOX	Outer UO2	k
Fine Resolution	(10,5)	24	64	0.033	-0.116	0.039	-0.009	-0.110	-0.005
Spatial	(3,1)	8	64	0.025	-0.093	0.034	-0.005	-0.105	0.006
	(4,2)	8	64	0.042	-0.111	0.045	-0.016	-0.110	-0.006
	(6,3)	8	64	0.045	-0.115	0.047	-0.018	-0.112	-0.007
	(8,4)	8	64	0.045	-0.116	0.047	-0.018	-0.112	-0.008
	(10,5)	8	64	0.045	-0.117	0.048	-0.018	-0.112	0.004
Polar	(4,2)	2	64	0.296	-0.120	0.193	-0.179	-0.140	-0.013
	(4,2)	4	64	0.101	-0.103	0.084	-0.059	-0.116	-0.013
	(4,2)	8	64	0.042	-0.111	0.045	-0.016	-0.110	-0.006
	(4,2)	16	64	0.031	-0.111	0.037	-0.008	-0.109	-0.004
Azimuthal	(4,2)	8	16	0.054	-0.114	0.054	-0.025	-0.114	-0.015
	(4,2)	8	32	0.043	-0.112	0.046	-0.017	-0.111	-0.007
	(4,2)	8	64	0.042	-0.111	0.045	-0.016	-0.110	-0.006
	(4,2)	8	128	0.042	-0.111	0.045	-0.016	-0.110	-0.006

Table 2.3: Percent Differences with Respect to PDT Reference Solutions for 2D Problem

	Mesh	Polar	Azimuthal	Max Pin	Min Pin	Inner UO2	MOX	Outer UO2	k
Spatial	(3,1)	8	64	-0.008	0.0231	-0.0049	0.0040	0.0052	0.0107
	(4,2)	8	64	0.009	0.0046	0.0060	-0.0068	-0.0007	-0.0013
	(6,3)	8	64	0.012	0.0002	0.0081	-0.0088	-0.0019	-0.0026
	(8,4)	8	64	0.012	-0.0006	0.0084	-0.0091	-0.0022	-0.0029
	(10,5)	8	64	0.012	-0.0008	0.0085	-0.0092	-0.0022	-0.0030
Polar	(4,2)	2	64	0.263	-0.0045	0.154	-0.170	-0.0306	-0.0085
	(4,2)	4	64	0.068	0.0123	0.045	-0.050	-0.0065	-0.0083
	(4,2)	8	64	0.009	0.0046	0.006	-0.007	-0.0007	-0.0013
	(4,2)	16	64	-0.002	0.0050	-0.002	0.002	0.0012	0.0013
Azimuthal	(4,2)	8	16	0.0214	0.0020	0.015	-0.016	-0.0044	-0.009
	(4,2)	8	32	0.0106	0.0041	0.007	-0.008	-0.0009	-0.0023
	(4,2)	8	64	0.0090	0.0046	0.006	-0.007	-0.0007	-0.0013
	(4,2)	8	128	0.0090	0.0050	0.006	-0.007	-0.0007	-0.0012

the common point and the most refined cases are within the error bars of the benchmark answer for all QOI's studied. It is unclear why the curve for spatial mesh increases between the second most refined case and the most refined case for k-effective, though the difference between the two is only 12 pcm. Finally, the infinitely refined point, as predicted by our model (discussed below) is plotted for each of the QOIs. They are all within the error bars of the MCNP answer.

The results of our least squares model can be seen in Table 2.5. The square root of the average

Table 2.4: Specifications of 2D Spatial Meshes

		Core Rings/pincell		Reflector # pincells × # cells/pincell		
		Fuel	Moderator	Fine	Medium	Coarse
Mesh	(3,1)	3	1	5 × 4	4 × 3	8 × 2
	(4,2)	4	2	7 × 6	3 × 3	7 × 3
	(6,3)	6	3	7 × 9	3 × 8	7 × 3
	(8,4)	8	4	7 × 12	4 × 9	6 × 4
	(10,5)	10	5	7 × 15	5 × 10	5 × 5

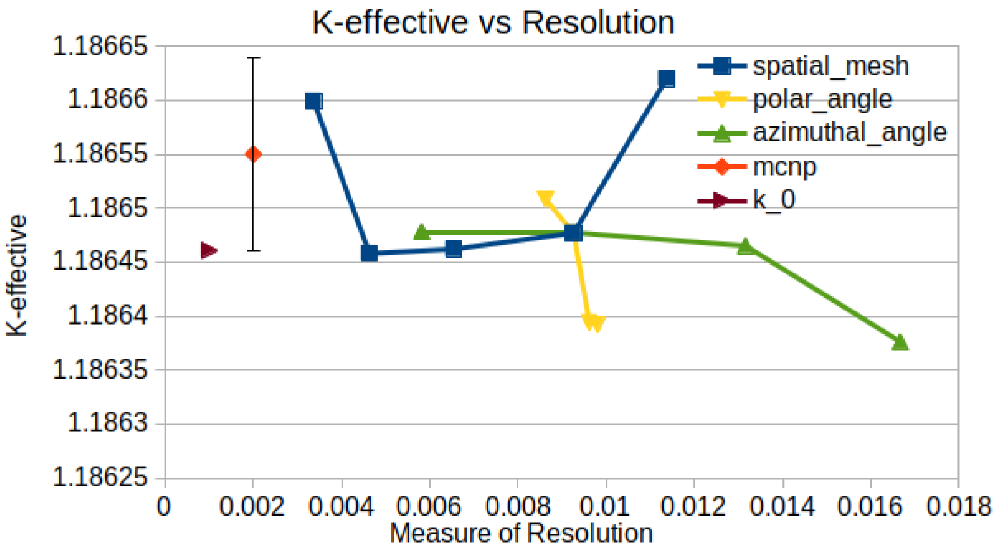

Figure 2.8: k-effective vs Resolution for 2D

of the square of the error summed over all training points between the model k-effective and the training data was 1.6 pcm. Also, the model predicted a k-effective for our most refined case of 1.186465. This is 0.8 pcm different than what PDT computed.

2.4.2 3D

We ran a “high-resolution" model of the C5G7 benchmark problem using PDT to generate what we believe is a highly accurate reference solution, in addition to the series of problems run

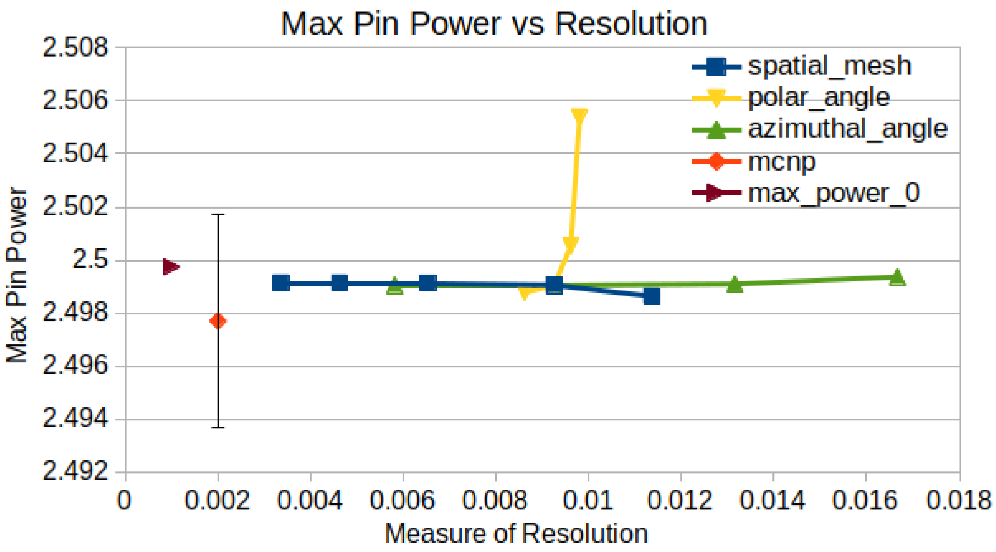

Figure 2.9: Max Pin Power vs Resolution for 2D

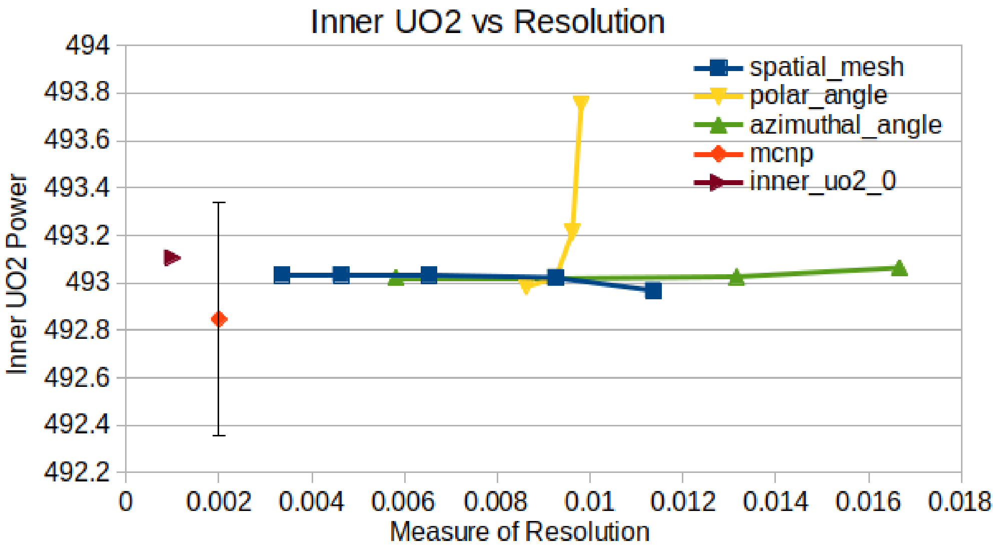

Figure 2.10: Inner UO2 Assembly Power vs Resolution for 2D

for our resolution and error studies. The maximum and minimum pin powers, assembly powers, and the k-eigenvalue for this high-resolution model, as well as the cases used in the error study, can be seen in Table 2.6. The high-resolution model used the (8,4) mesh with 60 z planes (over

Table 2.5: Parameters and Their Values in our Full Model of k-effective (2D C5G7)

Power	Value
p_x	2.0
p_ξ	2.0
p_γ	2.0

Coefficient	Value
k_0	1.1864607
C_x	5.53E-06
C_ξ	-1.85E-06
C_γ	-4.80E-09

55 million cells), 16 polar angles per hemisphere, and 192 total azimuthal angles. The relative differences of the QOI's for all the PDT models, as compared to MCNP reference results, can be seen in Table 2.7. Those same QOI's compared to the high-resolution PDT results can be seen in Table 2.8. Earlier versions of Tables 2.6, 2.7, and 2.8 appeared in conference proceedings in [20].

Again assuming our errors can be described using Equation 2.1, the truncation error analysis estimates that our "high-resolution" model of the C5G7 has approximately 5 pcm of error in the k-eigenvalue and less than 0.028% error in the worst pin power.

The 3D resolution study included 4 series: radial mesh, axial mesh, polar angles and azimuthal angles. The radial mesh study was performed on the (3,1), (4,2), (6,3) and (8,4) pin cell meshes as seen in Figure 2.3. The axial mesh study included runs with 8, 16, 20, 40, 60, and 80 z-planes. The polar angle series was done with 6, 8, 12, and 16 polar angles per octant and the azimuthal angle series was done with 8, 16, 32, and 48 azimuthal angles per octant. For each series, the variables not being refined were held at the common point: (4,2) spatial mesh, 40 z-planes, 8 polar angles per octant, and 32 azimuthal angles per octant.

The results of our study, for 3 QOI's (k-effective, max pin power, and inner UO_2 assembly power) can be seen in Figures 2.11 - 2.13. The common point ((4,2) mesh, 40 z-planes, 8 polar angles, and 32 azimuthal angles per quadrant) can be seen where the curves interesect. In addition to the study results, the MCNP benchmark answer, together with its error bars, is presented in each figure. Both the common point and the most refined cases are within the error bars of the benchmark answer for all QOI's studied. Finally we have also plotted our models prediction for each QOI at infinite resolution. All quantities of interest's predictions lie within the error bars for

Table 2.6: 3D QOIs

	Mesh	Z Planes	Polar	Azim.	Max Pin	Min Pin	Inner UO2	MOX	Outer UO2	k
MCNP					2.48126	0.23110	491.211	212.701	139.387	1.143080
MCNP 2σ					0.14%	0.38%	0.29%	0.21%	0.15%	0.006%
PDT	(8,4)	60	16	48	2.48070	0.23025	491.157	212.720	139.402	1.143081
$\frac{\text{PDT-MCNP}}{\text{MCNP}}$					-0.023%	-0.368%	-0.011%	0.009%	0.011%	0.000%
Pin Mesh	(3,1)	40	8	32	2.48030	0.23032	491.144	212.717	139.421	1.143257
	(4,2)	40	8	32	2.48063	0.23025	491.178	212.709	139.404	1.143118
	(6,3)	40	8	32	2.48061	0.23023	491.169	212.717	139.397	1.143103
	(8,4)	40	8	32	2.48059	0.23023	491.164	212.721	139.394	1.143101
Z Planes	(4,2)	8	8	32	2.47119	0.22971	490.316	213.192	139.300	1.143471
	(4,2)	16	8	32	2.47713	0.23004	490.875	212.883	139.358	1.143470
	(4,2)	20	8	32	2.47848	0.23013	490.995	212.815	139.375	1.143356
	(4,2)	40	8	32	2.48063	0.23025	491.178	212.709	139.404	1.143118
	(4,2)	60	8	32	2.48103	0.23028	491.210	212.690	139.410	1.143064
	(4,2)	80	8	32	2.48117	0.23028	491.221	212.684	139.411	1.143046
Polar	(4,2)	40	6	32	2.48100	0.23026	491.227	212.686	139.401	1.143090
	(4,2)	40	8	32	2.48063	0.23025	491.178	212.709	139.404	1.143118
	(4,2)	40	12	32	2.48041	0.23025	491.148	212.723	139.406	1.143139
	(4,2)	40	16	32	2.48035	0.23025	491.140	212.727	139.407	1.143146
Azim.	(4,2)	40	8	8	2.48219	0.23018	491.397	212.616	139.370	1.142812
	(4,2)	40	8	16	2.48089	0.23025	491.215	212.693	139.399	1.143041
	(4,2)	40	8	32	2.48063	0.23025	491.178	212.709	139.404	1.143118
	(4,2)	40	8	48	2.48060	0.23026	491.174	212.711	139.404	1.143126

the MCNP solution.

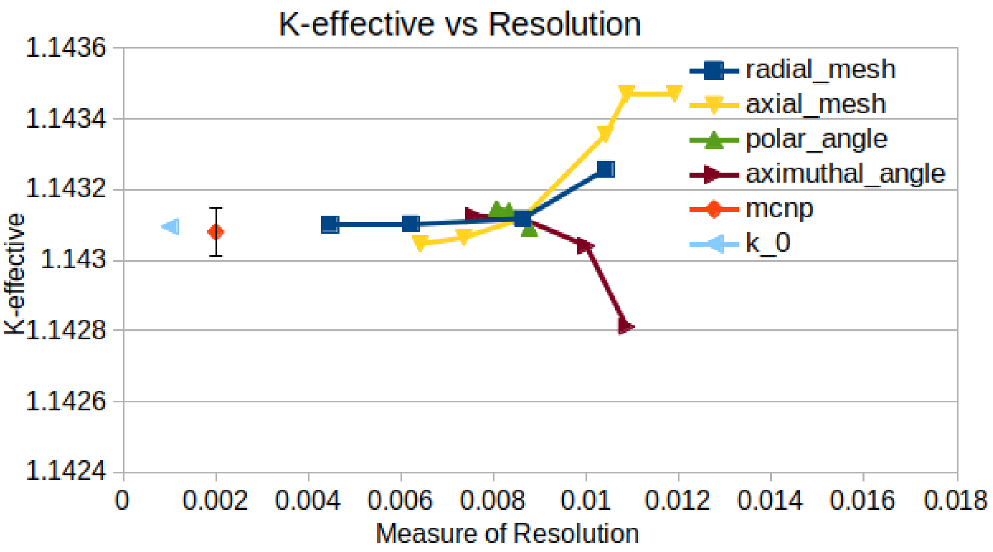

Figure 2.11: k-effective vs Resolution for 3D

Table 2.7: Percent Differences with Respect to MCNP Reference Solutions for 3D Problem

	Mesh	Z Planes	Polar	Azim.	Max Pin	Min Pin	Inner UO2	MOX	Outer UO2	k
PDT Fine Res.	(8,4)	60	16	48	-0.023	-0.368	-0.011	0.009	0.011	0.000
Pin Mesh	(3,1)	40	8	32	-0.039	-0.337	-0.014	0.008	0.024	0.015
	(4,2)	40	8	32	-0.026	-0.365	-0.007	0.004	0.012	0.003
	(6,3)	40	8	32	-0.026	-0.375	-0.009	0.008	0.007	0.002
	(8,4)	40	8	32	-0.027	-0.378	-0.010	0.009	0.005	0.002
Z Planes	(4,2)	8	8	32	-0.406	-0.600	-0.182	0.231	-0.062	0.034
	(4,2)	16	8	32	-0.167	-0.457	-0.068	0.086	-0.020	0.034
	(4,2)	20	8	32	-0.112	-0.421	-0.044	0.053	-0.008	0.024
	(4,2)	40	8	32	-0.026	-0.365	-0.007	0.004	0.012	0.003
	(4,2)	60	8	32	-0.009	-0.356	0.000	-0.005	0.017	-0.001
	(4,2)	80	8	32	-0.004	-0.353	0.002	-0.008	0.018	-0.003
Polar	(4,2)	40	6	32	-0.011	-0.364	0.003	-0.007	0.010	0.001
	(4,2)	40	8	32	-0.026	-0.365	-0.007	0.004	0.012	0.003
	(4,2)	40	12	32	-0.034	-0.365	-0.013	0.010	0.014	0.005
	(4,2)	40	16	32	-0.037	-0.365	-0.014	0.012	0.014	0.006
Azim.	(4,2)	40	8	8	0.037	-0.396	0.038	-0.040	-0.012	-0.023
	(4,2)	40	8	16	-0.015	-0.368	0.001	-0.004	0.009	-0.003
	(4,2)	40	8	32	-0.026	-0.365	-0.007	0.004	0.012	0.003
	(4,2)	40	8	48	-0.027	-0.365	-0.008	0.004	0.013	0.004

Table 2.8: Percent Differences with Respect to PDT Reference Solutions for 3D Problem

	Mesh	Z Planes	Polar	Azim.	Max Pin	Min Pin	Inner UO2	MOX	Outer UO2	k
Pin Mesh	(3,1)	40	8	32	-0.016	0.031	-0.003	-0.001	0.013	0.015
	(4,2)	40	8	32	-0.003	0.002	0.004	-0.005	0.001	0.003
	(6,3)	40	8	32	-0.004	-0.007	0.002	-0.001	-0.004	0.002
	(8,4)	40	8	32	-0.005	-0.010	0.001	0.000	-0.006	0.002
Z Planes	(4,2)	8	8	32	-0.383	-0.233	-0.171	0.222	-0.073	0.034
	(4,2)	16	8	32	-0.144	-0.090	-0.058	0.077	-0.031	0.034
	(4,2)	20	8	32	-0.089	-0.054	-0.033	0.045	-0.019	0.024
	(4,2)	40	8	32	-0.003	0.002	0.004	-0.005	0.001	0.003
	(4,2)	60	8	32	0.013	0.012	0.011	-0.014	0.005	-0.001
	(4,2)	80	8	32	0.019	0.015	0.013	-0.017	0.007	-0.003
Polar	(4,2)	40	6	32	0.012	0.004	0.014	-0.016	-0.001	0.001
	(4,2)	40	8	32	-0.003	0.002	0.004	-0.005	0.001	0.003
	(4,2)	40	12	32	-0.012	0.002	-0.002	0.001	0.003	0.005
	(4,2)	40	16	32	-0.014	0.003	-0.003	0.003	0.003	0.006
Azim.	(4,2)	40	8	8	0.060	-0.029	0.049	-0.049	-0.023	-0.024
	(4,2)	40	8	16	0.008	0.000	0.012	-0.013	-0.002	-0.003
	(4,2)	40	8	32	-0.003	0.002	0.004	-0.005	0.001	0.003
	(4,2)	40	8	48	-0.004	0.003	0.003	-0.004	0.001	0.004

The results of our least squares model can be seen in Table 2.9. The square root of the average of the square of the error summed over all training points between the model k-effective and the

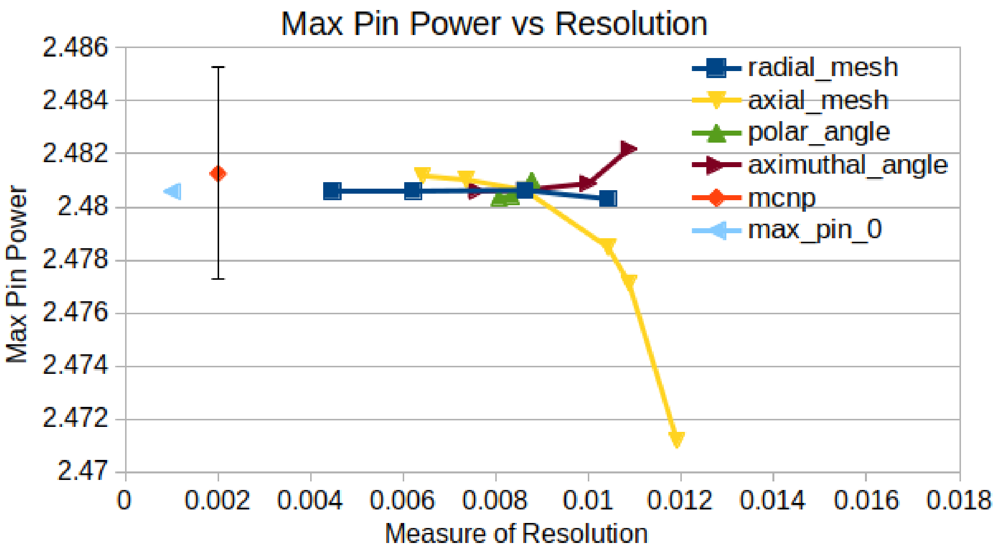

Figure 2.12: Max Pin Power vs Resolution for 3D

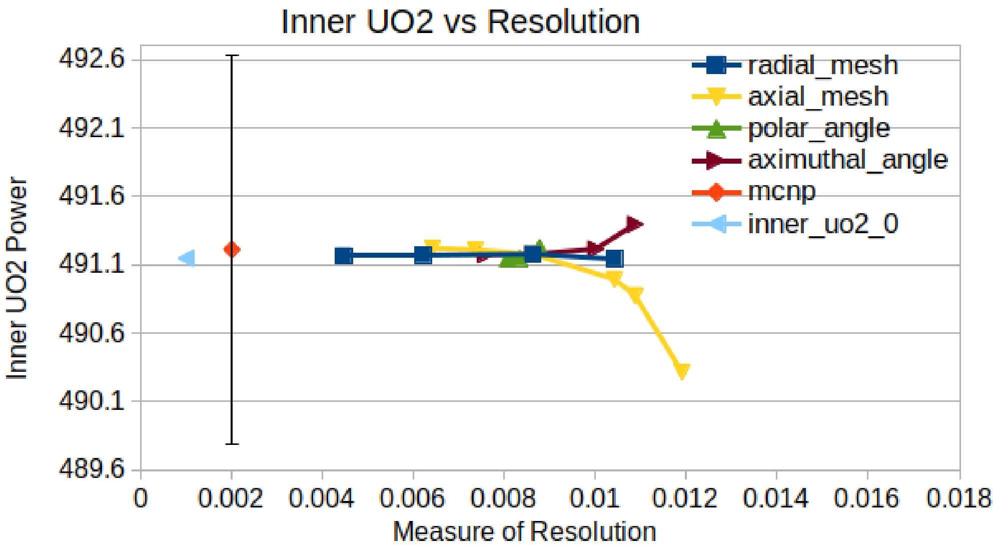

Figure 2.13: Inner UO2 Assembly Power vs Resolution for 3D

training data was 1.8 pcm. The prediction by the model for the most resolved case we ran with PDT was 2.1 pcm different than what we computed with PDT.

Table 2.9: Parameters and Their Values in our Full Model of k-effective (3D C5G7)

Power	Value
p_x	2.0
p_z	2.0
p_ξ	2.0
p_γ	2.0

Coefficient	Value
k_0	1.143095
C_x	2.01E-05
C_z	4.23E-06
C_ξ	-7.11E-06
C_γ	-9.94E-06

2.5 Comparing to Spider Web Grids

As a final note, we have compared our results using a morphed grid (results seen in this chapter) with results using our updated meshing capability: spiderweb grids. We ran the common point ((4,2) mesh, 8 polar angles and 64 azimuthal angles per quadrant) using both the morphed grid and the spiderweb grid capability. The difference in k-effective between the two types of meshes was 5 pcm. This gives us confidence that using spiderweb grids, while more efficient, gives us similar solutions to morphed grids.

2.6 Conclusions

We have modeled the 2D and 3D versions of the C5G7 benchmark problem. We have characterized the error in our computed solution using a least squares fit error model. We have determined from this error model that we need to take into account the interaction between our spatial and angular discretizations. Finally, we have computed what we believe to be highly resolved solutions to these benchmark problems using PDT.

3. INTERACTION BETWEEN SPACE AND ANGLE

In two dimensions we have determined that our polar discretization choice and our spatial mesh size interact in their effects on the transport solution. Pitkaranta and Scott observed this relationship in [18]. The reason for the interaction between space and angle in 2D is, as we increase the number of polar angles near the pole, the path length for a particle travelling through a given mesh cell increases. This interaction is limited in 3D as the vertical size of the of the spatial cells, and therefore the path-length of the particles within a cell, is limited by the discretization in the z-dimension.

3.1 Single Pin Cell Problems

To begin our study we started with a small model problem: a single 2D pin cell with a few energy groups. Specifically we are using cross sections from the C5G7 benchmark problem [15].

We studied the problem with two types of meshes: orthogonal and spider web. The orthogonal meshes had fuel and moderator with no cladding. The spider web meshes conform to the circular geometry and included a cladding. All results are comparing to the most finely resolved run of the set because we made up this problem, and therefore have no benchmark to compare to.

3.1.1 Orthogonal Mesh

Examples of the orthogonal meshes we ran with can be seen in Figure 3.1. The red in the figure is fuel and blue is moderator. The results of our study can be seen in Figure 3.2. The number of mesh cells is seen on the x-axis and the relative difference in k-effective between the most refined mesh and polar angle discretization and the current run is on the y-axis. The different curves are different number of polar angles per quadrant. We used 32 azimuthal angles per quadrant for this study to make sure the error associated with azimuthal angles was negligible compared to the spatial or polar error. We determined this number of azimuthal angles would be sufficient from our study of the C5G7 problem.

We achieve first order convergence - the error goes down by a factor of two when the mesh is refined by a factor of 2 - with the exception of the 4 polar angle per quadrant case. We believe that this is due to not having enough polar angles to resolve the problem, and therefore the error in the problem is predominantly from the polar discretization. Once we have enough polar angles to resolve the problem, our error is dominated by our spatial mesh. It takes almost 1000 mesh cells to achieve a relative difference in k-effective of 10 pcm. It should be noted that this plot, Figure 3.2, is a log-log plot.

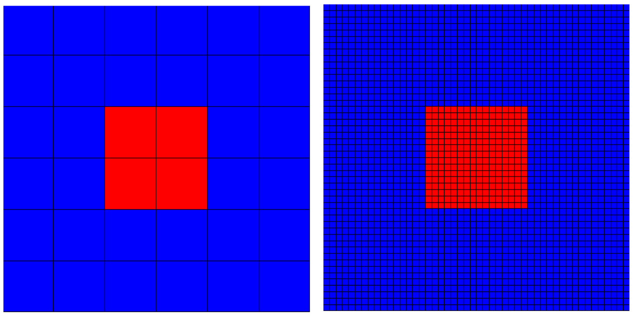

Figure 3.1: Meshes of Orgothonal Mesh Problem

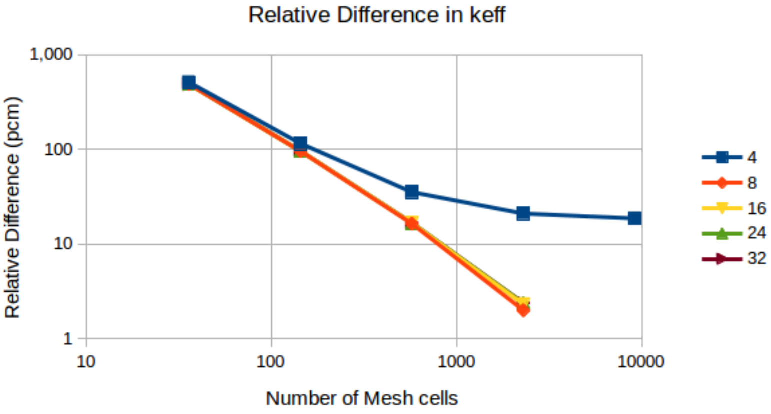

Figure 3.2: Relative Difference (in pcm) in k-effective for Orthogonal Mesh Problem

3.1.2 Spider Web Mesh

We next moved onto spider web meshes. Examples of the spider web meshes we ran with can be seen in Figure 3.3. The red in the figure is fuel, the blue is moderator, and the green is cladding. These meshes have 8 azimuthal segments in the fuel and 12 azimuthal segments in the cladding and moderator. The results of our study can be seen in Figure 3.4. Again the number of mesh cells is on the x-axis and the relative difference in k-effective between the finest resolution run and the current run is on the y-axis. The different curves are different numbers of polar angles per quadrant. Again we used 32 azimuthal angles per quadrant for all the runs.

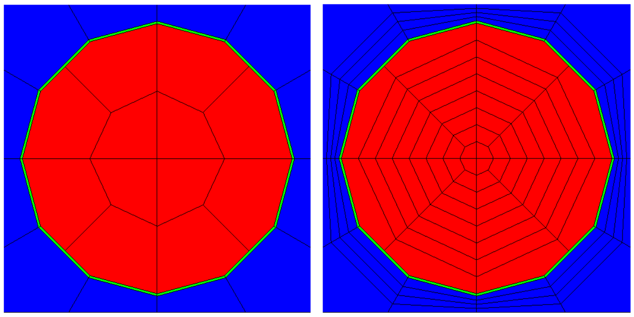

Figure 3.3: Meshes of Spider Web Mesh Problem

The solution in the coarsest mesh is within 10 pcm of the finest resolution mesh for each of the polar angle runs. There is a bigger difference between the 4 and 8 polar angle runs than between the 8 and 32 polar angle runs. This means that the solution is rapidly converging with polar angle refinement. There is also a bigger difference between the coarsest mesh and the second coarsest mesh than between the second coarsest mesh and the finest mesh. This means that our solution is also rapidly converging with spatial refinement.

In addition to refining the spatial mesh in the radial direction, the spider web mesh capability has the option of refining each radial section in the azimuthal direction separately. This means that a user can define the number of azimuthal segments in the fuel and the number of segments in the

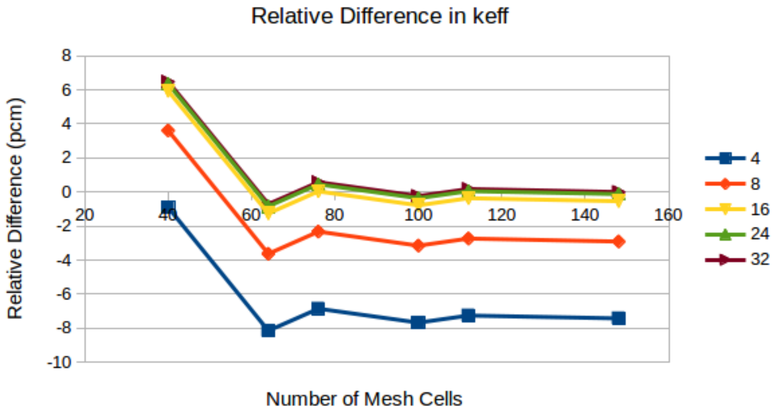

Figure 3.4: Percent Difference in k-effective for Spider Web Mesh Problem

moderator independently. We ran a study to see how the selection of azimuthal segments in each section affects the solution.

Figure 3.5 shows the different azimuthal discretization choice we made in the study. The left mesh has 8 azimuthal segments in each section of the problem. The right mesh has 12 azimuthal segments in each section of the problem. The middle mesh has 8 azimuthal segments in the fuel and 12 segments in the cladding and the moderator.

Figure 3.6 shows the results of this study. In this investigation, only the radial mesh was refined, leaving the number of azimuthal segments per radial section constant. On the x-axis is the number of mesh cells and on the y-axis is the relative difference in k-effective between the finest resolution run (12 azimuthal segments everywhere, 8 radial rings in the fuel and 4 in the moderator) and the current run. All the problems were run with 32 polar and 32 azimuthal angles per quadrant.

As expected, the different meshes converge to slightly different solutions. This is because the approximations made in each mesh are slightly different. All the solutions are within 10 pcm of the finest resolution case, even the coarsest mesh. Again, the difference between the coarsest mesh and the second coarsest mesh is smaller than the difference between the second coarsest mesh and the finest mesh. This means that our solution is rapidly converging with spatial refinement.

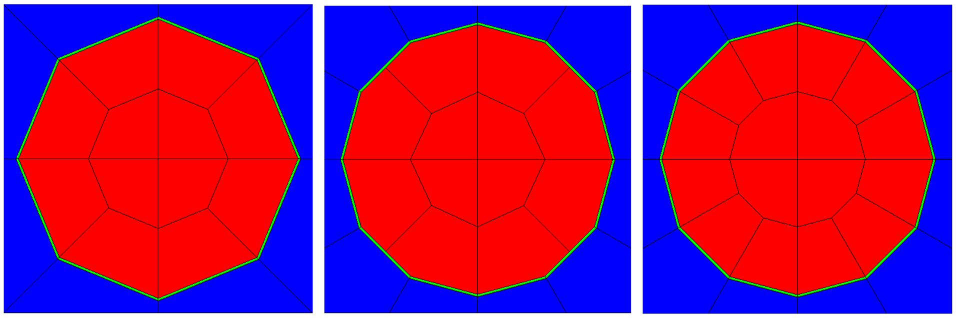

Figure 3.5: Spider Web Mesh with 8, Mixed, and 12 Azimuthal Segment per Ring

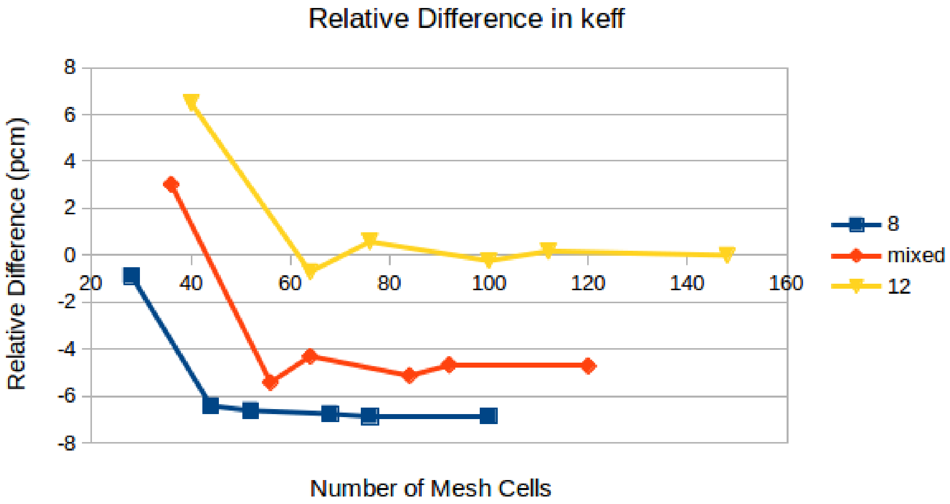

Figure 3.6: Percent Difference in k-effective for Azimuthal Segment Study

3.1.2.1 Separating Fuel and Moderator

We noticed in this study that our convergence rate was not monotonic, and wanted to discover why. To do this, we put together another run set, but this time with all the combinations of numbers of cells in the fuel and moderator. As an example of the meshes we ran, Figure 3.7 shows the different number of cells in the fuel, ranging from 2 to 8, for a single moderator radial cell count (4), and Figure 3.8 shows the different number of cells in the moderator, ranging from 1 to 5, for a single fuel radial cell count (4). We ran all the possibilities in the following set: number of radial cells in the fuel ranging from 2 to 8 and the number of radial cells in the moderator ranging from 1 to 5. For all these meshes, we used 8 azimuthal segments in the fuel and 12 segments in the cladding and moderator. We also used 32 polar angles and 32 azimuthal angles per quadrant as our quadrature to ensure that the error in our solution was mostly attributed to our spatial meshes.

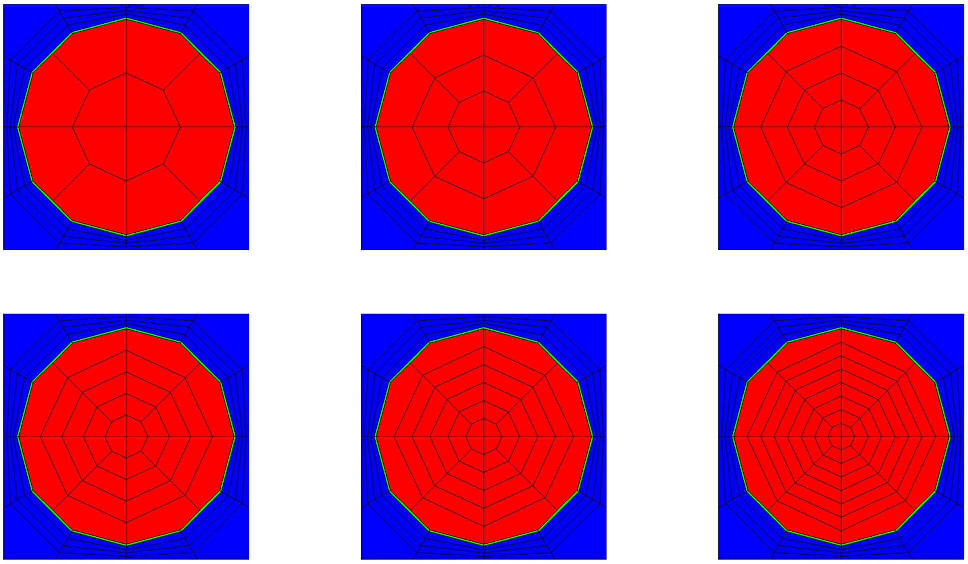

Figure 3.7: Meshes with Different Number of Cells in the Fuel

Figure 3.9 shows the results of our study. We looked at the relative difference in k-effective (in pcm) between each case and the most refined case: 8 cells in the fuel and 5 cells in the moderator. On the left, the curves each represent a single number of fuel cells. On the right, the curves each represent a single number of moderator cells. The data in both of these graphs are the same, just represented differently to show the different relationships. For a constant number of radial cells in the fuel, the curves are concave down, while for a constant number of radial cells in the moderator, the curves are concave up. This explains our non-monotonic relationships in our previous study, as we alternated adding cells in the fuel and the moderator.

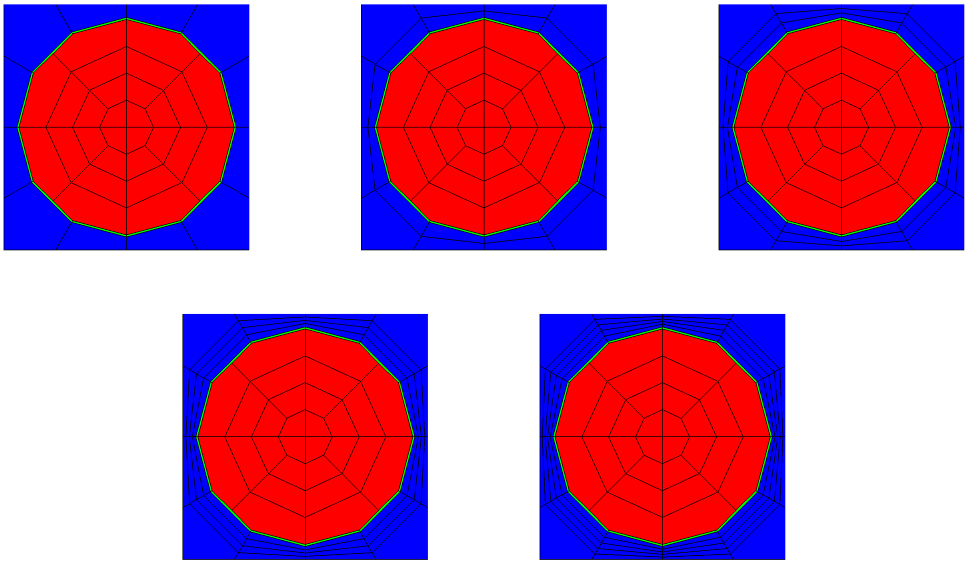

Figure 3.8: Meshes with Different Number of Cells in the Moderator

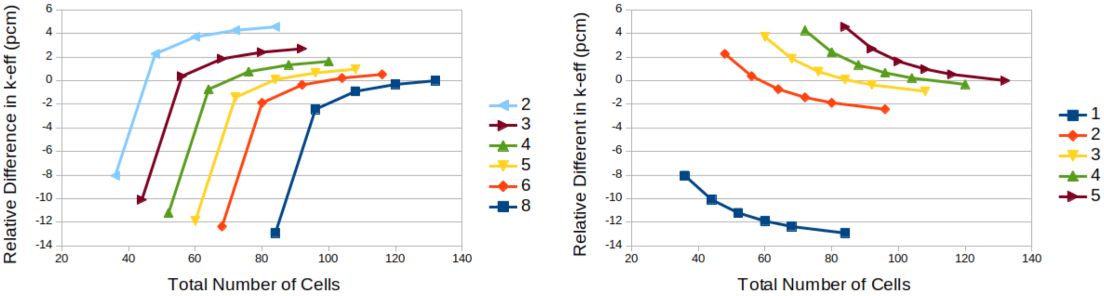

Figure 3.9: Relative Difference (pcm) in k-effective for Different Numbers of Cells in the Fuel (left) and Moderator (right)

In addition to our qualitative analysis, we also modeled this data using the following relationship:

$$k = k_0 + f_1 f^q + f_2 f^{2q} + m_1 m^r + m_2 m^{2r} + c_1 f^s m^t + c_2 f^{2s} m^t + c_3 f^s m^{2t} \tag{3.1}$$

Here, k_0 is the k-effective for an infinitely refined problem. Also, f and m are a measure of the cell areas in the fuel and moderator respectively, normalized to the highest resolution case.

They are computed by using the number of cells in the fuel (or moderator) and dividing by the number of cells in the fuel (or moderator) the highest resolution case has. This means that the highest resolution case has $1'$s for f and m, and all other cases have larger than 1 for f and m. This was to make the model non-dimensional and to better see which terms have the greatest impact. We used a python script with a series of powers, in multiples of 0.5 ranging from 0.5 to 2.0, for each of the $\{q, r, s, t\}$, and performed a least squares fit to determine the best coefficients $\{k_0, f_1, f_2, m_1, m_2, c_1, c_2, c_3\}$, that give the lowest absolute value of difference between our model and the results we calculated using PDT, when summed over all training points. Table 3.1 shows the powers and coefficients determined from this analysis. The term with the largest (in magnitude) coefficient, which therefore has the greatest effect, is the first term: f^q.

Table 3.1: Parameters and Their Values in our Full Model of k-effective (Fuel and Moderator)

Power	Value
q	1.5
r	1.0
s	1.5
t	1.0

Coefficient	Value
k_0	1.318101
f_1	-1.41E-05
f_2	6.25E-07
m_1	-8.96E-06
m_2	8.53E-06
c_1	3.37E-09
c_2	-1.35E-08
c_3	-1.88E-09

The L1 and L2 norm of the error between the predicted k-effective and the training data was smaller than the tolerance we used in our least squares fit. This means that our model is predicting our training data well. We left out the most refined case, the (8,5) mesh, from our training data. We then used our model to predict what k-effective that PDT would compute with that mesh. The difference between our model k-effective and the k-effective that PDT computed was 0.08 pcm. Also, the difference between the model k-effective and k_0 (the model's prediction of the k-effective at infinite resolution) was 1 pcm. This means that our model can extrapolate to higher resolution well and that the most refined case is converged, according to our model.

3.1.3 Modeling Space and Angle

We used the same process as the previous section to model both our spatial and angular resolution contributions to our k-effective. The model we used was:

$$\begin{aligned} k = k_0 + f_1 f^q + f_2 f^{2q} + m_1 m^r + m_2 m^{2r} + p_1 p^u + p_2 p^{2u} \\ + a_1 a^w + a_2 a^{2w} + c_1 f^s m^t p^v a^x + c_2 f^{2s} m^t p^v a^x \\ + c_3 f^s m^{2t} p^v a^x + c_4 f^s m^t p^{2v} a^x + c_5 f^s m^t p^v a^{2x} \end{aligned} \tag{3.2}$$

Here, again, k_0 is the model predition of k-effective at infinite resolution, f is the average area of the cells in the fuel, m is the average area of the cells in the moderator, p is π divided by the number of polar angles and a is 2π divided by the number of azimuthal angles. We then normalized all our variables (f, m, p, and a) to the most resolved case. This was again to make the model non-dimensional and to better see which terms have the greatest impact on our model. We ran cases that ranged from a (3,2) mesh to a (8,4) mesh, 12 polar angles to 32 polar angles per quadrant, and 8 azimuthal angles to 32 azimuthal angles per quadrant. We again used a python script to vary the powers {q,r,s,t,u,v,w,x} from 0.5 to 2.0 in increments of 0.5. We then performed a least squares analysis to find the best coefficients for each set of powers and computed the error in k-effective for all our test cases, where error is the absolute value of the difference between the model k-effective and the k-effective in our training data, when summed over all our training data. Finally, we picked the powers and coefficients that gave us the lowest error. Table 3.2 shows the powers and coefficients determined by our python script. The L1 norm of the error between the predicted k-effective and the training data was 0.1 pcm. The L2 norm was 0.1 pcm. Again, this means that our model is predicting our training data well. Similar to the previous study, the spatial term with the highest magnitude coefficient is f^q, while the term with the highest magnitude coefficient overall is a^w

To verify our model is accurate we ran an additional case to the cases used in building our model: a (10,5) mesh, 48 polar angles and 48 azimuthal angles per quadrant. This case is more

Table 3.2: Parameters and Their Values in our Full Model of k-effective (Space and Angle)

Power	Value
q	1.0
r	1.0
s	1.0
t	2.0
u	2.0
v	0.5
w	2.0
x	2.0

Coefficient	Value
k_0	1.318138
f_1	-1.99E-05
f_2	1.76E-09
m_1	8.71E-06
m_2	5.03E-06
p_1	-2.40E-06
p_2	-2.30E-08
a_1	-3.52E-05
a_2	1.94E-07
c_1	2.15E-07
c_2	-1.54E-08
c_3	-9.62E-09
c_4	-2.78E-08
c_5	-4.53E-09

refined in all four parameters. The model k-effective and the calculated k-effective disagree by 0.5 pcm, and the calculated k-effective is 2.1 pcm different than k_0. Also to guard against cancellation of errors, the sum of the absolute value of the error terms for this case was 2.7 pcm. This means that the most refined case is resolved, and the model can accurately extrapolate to more refined cases.

We also wanted to determine a ‘good enough’ resolution. The most resolved case we ran in the training data was an (8,4) mesh with 32 polar angles and 32 azimuthal angles per quadrant. The k-effective computed using this run was 4.3 pcm different than k_0 (7.1 pcm when summing the absolute value of the error terms rather than letting them cancel). This is the ‘error’ in the computation, according to our model. We then looked at a series of ‘mid range’ runs to determine if there was a level of resolution we could stop at and still get the same level of accuracy as the high resolution run. We found that a case with a (4,2) mesh and 24 polar angles and 32 azimuthal angles per quadrant had an error of 4.1 pcm (12.0 pcm when summing the absolute value of the error terms). This means that we do not necessarily need the highest resolution to get an accurate solution.

3.2 Assembly Problems

In addition to the single pin cell problems, we also ran a larger, more difficult assembly level problem. Also, in addition to using our model to determine our k-effective at an infinite resolution, we also want to use it to determine spatial mesh, polar, and azimuthal resolution requirements for an ‘acceptable’ level of error.

The test problem we used was a 17 by 17 layout of the pins we used in our single pin cell calculations, with 25 water holes interspersed throughout the assembly. Figure 3.10 shows the assembly. We ran cases with resolutions ranging from a (3,2) spatial mesh to an (8,4) spatial mesh, 12 to 32 polar angles per quadrant, and 8 to 32 azimuthal angles per quadrant. Table 3.3 shows the powers and coefficients our python script determined for our model for the assembly problems. The average absolute value of the error between the model k-effective and the training data was 0.03 pcm.

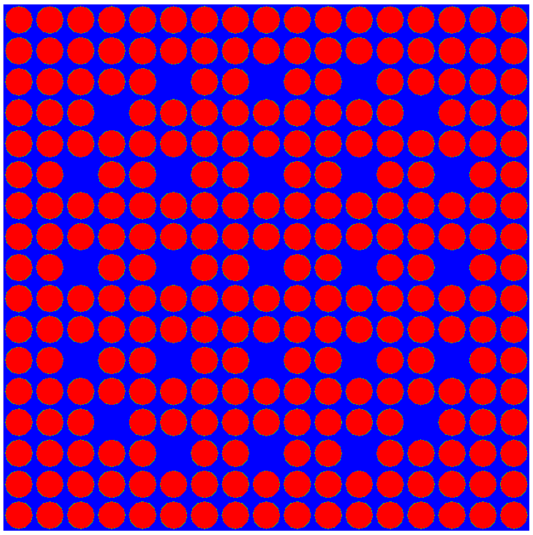

Figure 3.10: Assembly Used in the Space Angle Study

Table 3.3: Parameters and Their Values in our Full Model of k-effective for Assembly Problems

Power	Value
q	0.5
r	2.0
s	2.0
t	2.0
u	1.0
v	0.5
w	0.5
x	2.0

Coefficient	Value
k_0	1.334449
f_1	5.39E-06
f_2	-1.32E-05
m_1	5.37E-06
m_2	2.09E-08
p_1	9.17E-07
p_2	-2.24E-06
a_1	4.88E-04
a_2	-2.90E-04
c_1	1.01E-07
c_2	4.34E-11
c_3	-1.47E-08
c_4	-9.22E-09
c_5	-1.01E-09

We then ran a more refined problem than any of the training cases we used to build our model, in order to check the accuracy of that model. We got a disagreement in the model k-effective and the computed k-effective of 0.08 pcm. This means that the model can extrapolate accurately.

Finally we looked at the error we incur by using different resolution sizes. We did this by looking at the difference in the models ‘error’ terms, which are all the terms except k_0. Our fine resolution case, which was a (10,5) spatial mesh, 48 polar angles per quadrant, and 48 azumthal angles per quadrant, had an error of 15 pcm (21 pcm when summing the absolute value of the error terms). Our least resolved case used in training our model, which has a (3,2) spatial mesh, 12 polar angles per quadrant, and 8 azimuthal angles per quadrant, had an error of -17 pcm (22 pcm when summing the absolute value of the error terms). This means that we are effectively resolved using our least refined case.

3.3 Conclusions

We have studied the interaction between spatial and angular resolution and its effect k-effective for both pin cell and assembly level problems. We have performed a least squares analysis to determine the coefficients and powers in our model. This model is shown to be accurate both for the training data we used to create it, but also for extrapolatory problems where the resolution used is more refined than any training data point. This gives us confidence that the form of our model is accurate enough to predict k-effective solutions to problems we have not yet run.

4. ENERGY DISCRETIZATION

As the energy discretization in a problem gets more realistic, boundary layers become more prominent. This is due to large cross sections in the resonance region. We studied this phenomenon to determine the level of spatial resolution required to resolve these boundary layers using LD. We have used the spiderweb mesh capability in PDT for this study as these meshes are better able to conform to the geometry over orthogonal meshes.

We have used FEDS (Finite Element Discontinuous Support) cross sections for this investigation. This energy discretization method was developed by Dr. Andrew Till [21]. These cross sections have exceedingly large cross sections in the resolved resonance region due to how this method forms the ‘groups’, even for a small number of energy unknowns. This magnitude of cross section only arises with a large number (thousands) of energy groups using the multi-group (MG) method. The large cross sections cause boundary layers to form for much smaller number of unknowns in the FEDS method than in the MG method.

We have studied the formation of boundary layers using both small model problems (single pin cell) and more moderately sized problems (assembly level). The cross sections were created using the FEDS method from specifications found in the VERA benchmark from CASL [22, 23].

4.1 Single Pin Cell

The first problem chosen for studying the boundary layers that form with realistic energy discretization was a single pin cell with a pitch of 1.26cm. The meshes used can be seen in Figures 4.1 and 4.2. The red material is fuel, the green material is cladding, and the blue material is the moderator. The meshes range from 3 cells in the fuel and 2 cells in the moderator (with one cell in the cladding) to 6 cells in the fuel and 3 cells in the moderator (again with one cell in the cladding). The fuel has an outer radius of 0.4096 cm and the clad is a thickness of 0.0654 cm.

As a note, the quadrature used was a product quadrature with 24 polar angles and 32 azimuthal angles per quadrant. We know from previous studies that the azimuthal angle is converged at

32 angles per quadrant. This means that the error due to the azimuthal angle resolution is small enough that other errors overshadow it. Also, we compared our k-effective solutions to a (16,32) quadrature. The solutions were within 1 pcm for all quantities of interest studied. From this, we know that the polar angle is also converged.

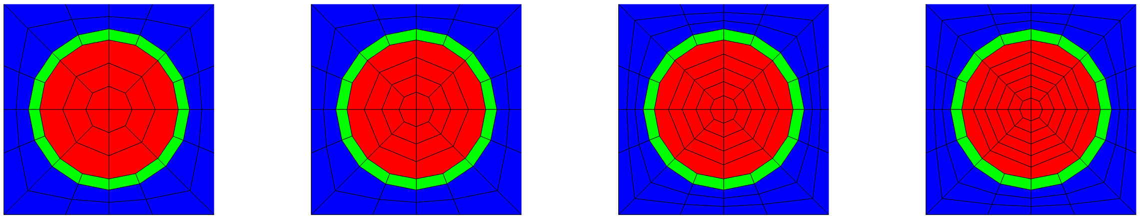

Figure 4.1: Meshes Used for Energy Study

In Figure 4.2, the meshes have a radial cell in the outer 10% of the fuel (by area), which is counted towards the radial cell count in the fuel. The cell is used to capture the steep gradient in the solution near the outer edge of the fuel. FEDS cross sections with 65 and 244 energy unknowns (with 30 and 108 unknowns in the resolved resonance region, respectively) were used for comparison.

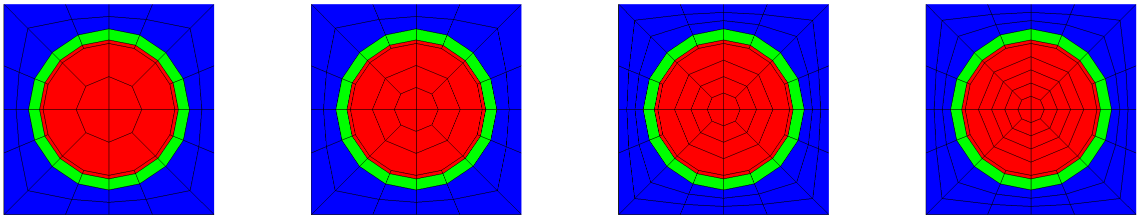

Figure 4.2: Meshes Used for Energy Study with 10% outer Fuel Cell

In addition to the meshes in Figure 4.2, another mesh was used as a reference. It is the same as the (6,3) mesh, but there are two cells in the outer 10% of the fuel. Figure 4.3 shows a comparison of the (6,3) mesh and what we are calling the (7,3) mesh.

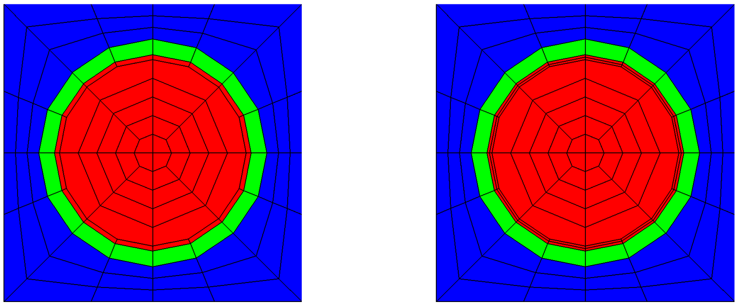

Figure 4.3: Comparison of (6,3) Mesh with 1 Cell (left) and 2 Cells (right) in Outer 10%

Figures 4.4 and 4.5 show the scalar flux for the right side of the problem for all meshes used, both without (left) and with (right) the cell in the outer 10% of the fuel for the 65 energy unknowns cross sections in ‘group’ 30 and for the 244 energy unknowns cross sections in ‘group’ 105. The solution is taken along a cutline of y = 0.63cm (the midpoint in y) from the center (in x) to the right side of the problem. These two energy ‘groups’ were chosen because they have the largest σ_t, which means the boundary layers will be most pronounced for these ‘groups’. The red curves are the (3,2) meshes, blue are the (4,2) meshes, magenta are the (5,3) meshes, and green are the (6,3) meshes. For the meshes with the cell in the outer 10%, the (7,3) mesh is also plotted for reference in black.

For the meshes without the cell in the outer 10% of the fuel there is a marked dip in the solution near the outer edge of the fuel, which occurs in slightly different places for the different resolutions of meshes used. The solution goes below zero in these dips. This is because the outer most cell in the fuel’s width is too large to accurately capture the solution gradient. As more cells are added in the fuel, and those cells get thinner, the dip gets thinner and closer to the outer edge and the negativities get smaller.

The right side of Figures 4.4 and 4.5 shows the scalar flux solution for meshes with the cell

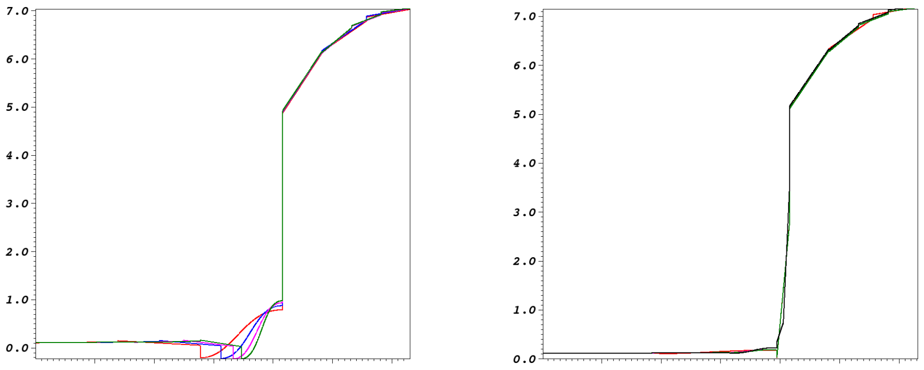

Figure 4.4: Scalar Flux Solution for 65 Energy Unknowns Without (left) and With (right) Cell in the Outer 10% of the Fuel

in the outer 10% of the fuel for 65 energy unknowns cross sections and 244 energy unknowns cross sections, respectively. The marked dip in the solution seen in the meshes without the cell is compressed to a discontinuity at the fuel edge. This is because the outermost cell is now thin enough to capture the solution gradient. The solution looks almost identical, at least in the fuel, for the (3,2) mesh and the (6,3) mesh. This can be seen in Figure 4.7, which is a zoom in of the solution at the fuel edge for both the 65 energy unknowns cross sections (left) and 244 energy unknowns cross sections (right). In addition to the meshes with 1 cell in the outer 10% of the fuel, the mesh with 2 cells in that region is also plotted. For the 65 energy unknowns plot, the dip in the solution disappears. For the 244 energy unknowns plot, the negativity disappears. This shows that adding an additional cell in the outer region of the fuel can reduce or remove negativities, or at the very least smooth out the flux profile.

We realize that the lineouts for the meshes without the outer 10% cell are curved in the cell where the flux goes negative. This is an artifact of the plotting software we used to plot these figures. Figure 4.6 is a lineout where we manually grabbed the fluxes from the output file for the (3,2) mesh. The reader can see that the flux does go negative in the outermost cell of the fuel, while the flux is not curved in that cell. Also the flux at the cell edges in this plot match that of Figure

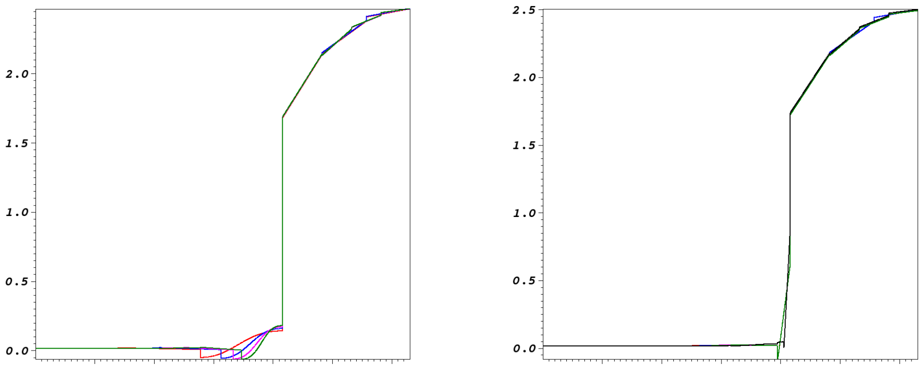

Figure 4.5: Scalar Flux Solution for 244 Energy Unknowns Without (left) and With (right) Outer 10% Cell for Each Mesh

4.4, giving us confidence that the curved line is just an artifact of the plotting software.

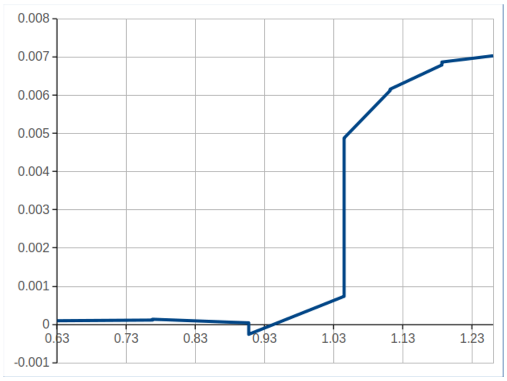

Figure 4.6: Scalar Flux Solution for 65 Energy Unknowns for (3,2) Mesh Without Outer 10% Cell

Figure 4.8 shows a comparison of the (6,3) mesh both with and without the cell in the outer 10% of the fuel for 65 energy unknowns.

Table 4.1 shows the relative difference in k-effective (in pcm) between cases run and the benchmark answer for cases without the 10% cell and with the 10% cell for 65 energy unknowns (left) and 244 energy unknowns (right). The meshes without the outer 10% cell have effectively the same

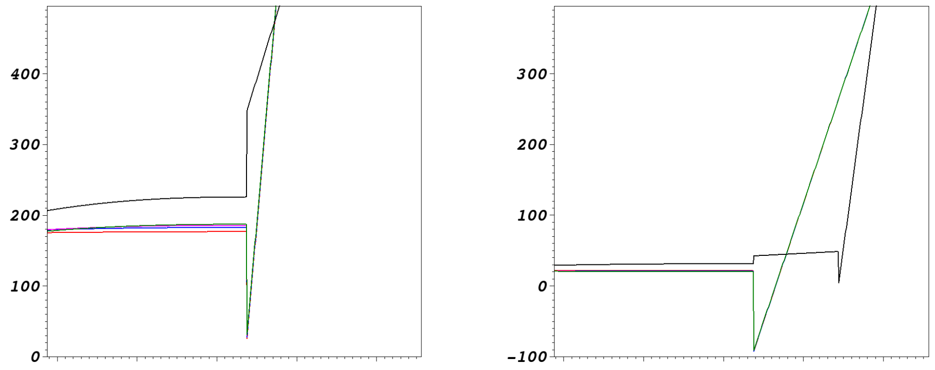

Figure 4.7: Zoom of Scalar Flux solution for 65 Energy Unknowns Cross Sections (left) and 244 Energy Unknowns (right) for Meshes With a Cell in the Outer 10% of the Fuel

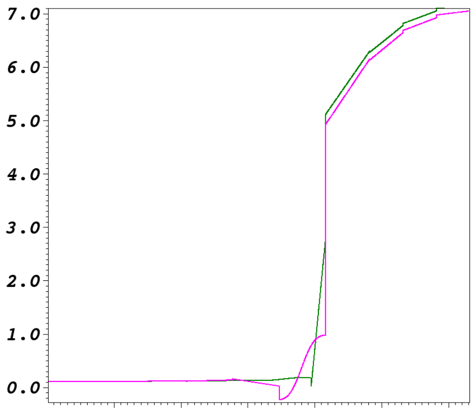

Figure 4.8: Scalar Flux Solution with (Magenta) and Without (Green) the Outer 10% Cell for the (6,3) Mesh

magnitude of error in k-effective as the meshes with those cells. The reason the cases with the outer 10% cell are so close to the cases without the cell in this comparison is because k-effective is a integrated quantity, not a pointwise one.

It is now informative to look at the relative difference in k-effective, compared to the most refined case we ran: the (7,3) mesh. This is to show the spatial error more clearly, as opposed to

Table 4.1: Relative Difference (pcm), Compared to the Benchmark, in k-effective for 65 Energy Unknowns (left) and 244 Energy Unknowns (right)

Mesh	Without Cell	With Cell
(3,2)	3986	3946
(4,2)	3980	3946
(5,3)	3981	3951
(6,3)	3979	3951
(7,3)		3946

Mesh	Without Cell	With Cell
(3,2)	272	226
(4,2)	264	224
(5,3)	266	230
(6,3)	263	230
(7,3)		225

the energy error which is much larger. Table 4.2 shows this comparison. The most refined case here is the (7,3) mesh for each set of cross sections. Here we can see that without the radial cell in the outer 10% of the fuel, the solution for k-effective is 35-50 pcm different than the most resolved case. With the cell in the outer 10% of the fuel, the error in k-effective is approximately 5 pcm, regardless of the spatial mesh or cross section set used. This shows that resolving the outer region of the fuel does not change k-effective much, as again it is an integrated quantity. The reaction rates show a different picture, as they are more pointwise quantities.

Table 4.2: Relative Difference (pcm), Compared to the Most Refined Case, in k-effective for 65 Energy Unknowns (left) and 244 Energy Unknowns (right)

Mesh	Without Cell	With Cell
(3,2)	41	0
(4,2)	35	-1
(5,3)	36	5
(6,3)	34	5
(7,3)		0

Mesh	Without Cell	With Cell
(3,2)	47	1
(4,2)	39	-1
(5,3)	42	6
(6,3)	38	6
(7,3)		0

Table 4.3 shows the percent reaction rate (absorption and fission) in the outer 10% of the fuel. Table 4.4 shows the relative difference (in pcm) of those quantities between the most resolved case ((7,3) mesh for each of the cross section sets) and each of the other cases. The most resolved case

was chosen as the comparison point because the benchmark does not have this quantity of interest in its collection of 'solutions', and the most resolved case is the best guess for the correct answer.

Table 4.3: Percent Absorption and Fission in the Outer 10% of the Fuel for 65 (left) and 244 (right) Energy Unknowns

Mesh	Absorption	Fission
(3,2)	12.540	10.351
(4,2)	12.544	10.353
(5,3)	12.545	10.353
(6,3)	12.546	10.354
(7,3)	12.470	10.352

Mesh	Absorption	Fission
(3,2)	13.245	10.397
(4,2)	13.249	10.399
(5,3)	13.251	10.400
(6,3)	13.252	10.401
(7,3)	13.145	10.399

The tables in Table 4.4 shows there is relatively little difference between the (3,2) case and the (6,3) case for each of the cross section sets, compared to the magnitude of the error overall. This is further proof that if there is a cell in the outer 10% of the fuel, a (3,2) mesh can achieve approximately the same spatial error as a (6,3) mesh. Though, at least for the absorption rate, it might be necessary to further resolve the outer region of the fuel with an extra cell, as seen in the magnitude of the error compared to the (7,3) mesh.

Table 4.4: Percent Relative Difference (in pcm) in Percent Absorption and Percent Fission in the Outer 10% of the Fuel for 65 (left) and 244 (right) Energy Unknowns

Mesh	Absorption	Fission
(3,2)	-561	17
(4,2)	-590	-4
(5,3)	-601	-8
(6,3)	-608	-14
(7,3)	0	0

Mesh	Absorption	Fission
(3,2)	-758	22
(4,2)	-793	-2
(5,3)	-809	-9
(6,3)	-818	-16
(7,3)	0	0

4.2 2D Assembly

The problem chosen for the assembly level comparison was taken from the VERA benchmark from CASL, specifically problem 2B from that benchmark. This problem is a 2D 17x17 assembly with 25 water holes, and 264 fuel pins that have a pitch of 1.26 cm. There is also a 4 mm water gap around all edges of the assembly. Figure 4.9 shows the layout of the assembly. The blue cells are water holes, the white cells are fuel pins, and the yellow cells are the 4 pins that have been chosen for further study. The yellow pins have the same composition as the rest of the fuel pins; their location in the assembly with respect to the water holes and the edge of the assembly makes them of interest for a closer look.

D	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
U	U	U	A	U	W	U	U	W	U	U	W	U	U	U	U	U
U	U	U	W	U	U	B	U	U	U	U	U	U	W	U	U	U
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
U	U	W	U	U	W	U	U	W	U	U	W	U	U	W	U	U
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
U	U	U	U	U	C	U	U	U	U	U	U	U	U	U	U	U
U	U	W	U	U	W	U	U	W	U	U	W	U	U	W	U	U
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
U	U	W	U	U	W	U	U	W	U	U	W	U	U	W	U	U
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
U	U	U	W	U	U	U	U	U	U	U	U	U	W	U	U	U
U	U	U	U	U	W	U	U	W	U	U	W	U	U	U	U	U
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U

Figure 4.9: Layout of the 2D Assembly Problem

Table 4.5 shows the relative difference (in pcm) in k-effective for the 2D assembly problem using the 65 energy unknowns cross sections (left) and the 244 energy unknowns cross sections (right) for fuel pins with a radial cell in the outer 10% of the fuel. These are compared to the benchmark solution for k-effective. The different meshes do not seem to make a difference in the solution – only the energy unknowns structure in the cross sections used. This suggests that the (3,2) mesh is resolved enough for this problem to get k-effective to within 3 pcm of the more refined (7,3) mesh. The cross sections dominate the error in the solution. It should be noted that this quantity of interest is an integrated quantity, which is why it is now important to look at the reaction rates and pin powers, which rely more on the details of the flux shape, in the four pin cells spelled out in Figure 4.9.

Table 4.5: Relative Difference (in pcm) in k-effective for the 2D Assembly Problem

Mesh	65 Energy Unknowns	244 Energy Unknowns
(3,2)	3498	168
(4,2)	3498	167
(5,3)	3501	171
(6,3)	3501	171
(7,3)	3497	166

Tables 4.6 - 4.9 show the relative difference, in percent, in pin powers compared to the benchmark for pins A-D, respectively. Again we see that we are essentially converged using the (3,2) mesh, and that the energy unknowns structure of the cross sections is the main provider of error in this simulation. We do get some cancellation of errors, which is why in some of the tables, the 65 energy unknowns cross sections get more accurate answers than the 244 energy unknowns cross sections. We also see that the errors are slightly larger for pin D using the 244 energy unknowns cross sections, which is at the edge of the assembly, rather than directly by a water hole. There is a 4 mm assembly gap around the assembly, made up of water, that contributes to this larger error. Still, all pin powers shown are within 1% of the benchmark solution, showing that with a radial cell in the outer 10% of the fuel, accurate solution can be computed even with the (3,2) mesh.

In addition to looking at individual pin powers, we looked at the pin powers over the whole assembly. We computed the RMS of the assembly for each mesh and cross section set. The RMS is computed by:

$$RMS = \sqrt{\frac{\sum_N e_n^2}{N}} \tag{4.1}$$

Table 4.6: Relative Difference (in percent) in Pin Power for Pin A for the 2D Assembly Problem

Mesh	65 Energy Unknowns	244 Energy Unknowns
(3,2)	0.332	0.186
(4,2)	0.332	0.187
(5,3)	0.335	0.189
(6,3)	0.335	0.189
(7,3)	0.335	0.189

Table 4.7: Relative Difference (in percent) in Pin Power for Pin B for the 2D Assembly Problem

Mesh	65 Energy Unknowns	244 Energy Unknowns
(3,2)	0.128	0.235
(4,2)	0.128	0.235
(5,3)	0.130	0.237
(6,3)	0.130	0.237
(7,3)	0.130	0.237

Table 4.8: Relative Difference (in percent) in Pin Power for Pin C for the 2D Assembly Problem

Mesh	65 Energy Unknowns	244 Energy Unknowns
(3,2)	0.004	0.335
(4,2)	0.005	0.335
(5,3)	0.007	0.332
(6,3)	0.007	0.332
(7,3)	0.007	0.332

where e_n is the error in pin n, or the percent difference between the computed pin power and the benchmark solution. Table 4.10 shows the RMS computed for each mesh and each cross section set. Similar to k-effective, the RMS is an integrated quantity. This is seen by the fact that the RMS is nearly identical for all the meshes used, and only changes with cross section set.

Table 4.9: Relative Difference (in percent) in Pin Power for Pin D for the 2D Assembly Problem

Mesh	65 Energy Unknowns	244 Energy Unknowns
(3,2)	0.227	0.795
(4,2)	0.227	0.795
(5,3)	0.227	0.796
(6,3)	0.227	0.796
(7,3)	0.227	0.796

Table 4.10: RMS for the 2D Assembly Problem

Mesh	65 Energy Unknowns	244 Energy Unknowns
(3,2)	0.18305	0.29173
(4,2)	0.18324	0.29159
(5,3)	0.18454	0.29181
(6,3)	0.18454	0.29179
(7,3)	0.18454	0.29173

4.3 Conclusions

As the energy discretization in a problem gets more realistic, i.e. more energy unknowns, the flux shape gets more complicated and harder to model. We studied this phenomenon for both pin cell problems and assembly level problems. We found that for both cross section sets we looked at, resolving the outer region of the fuel with a cell in the outer 10% is required to get accurate solutions for k-effective and the flux shape. For the 244 energy unknowns cross section set, it is sometimes required to have two cells in the outer 10% of the fuel to get non-negative flux shapes. We found for the assembly problem, once the outer region in the fuel is resolved, the resolution of the mesh does not need to be incredibly refined to compute pin powers to within 2 pcm of the most resolved mesh. The energy discretization is the main source of error in these simulations.

5. COMPARISON TO THE METHOD OF CHARACTERISTICS

The Method of Characteristics (MOC) has become the most widely used transport spatial discretization for solving reactor problems [3, 7, 8]. This work compares MOC and LD for reactor problems. A series of reactor calculations, ranging from simple to complex, have been performed to this end. Some of these calculations include geometric features or boundary layers that are extremely small relative to the problem domain, both of which require fine ray spacing for MOC to resolve, but for which LD is expected to perform well.

We first ran a series of single pin cell problems using the VERA benchmark from CASL - both with and without a thin annular ring of IFBA around the fuel. As a reference solution, we ran highly resolved versions of the problems using our DFEM code. We then ran the 2D C5G7 problem and compared several QOIs from that benchmark.

In addition to numerical comparisons, we have also performed analytic geometric comparisons. This was to show that the approximation that MOC codes use when using ‘tracks’ to compute transport calculations incurs comparable errors to DFEM codes that approximate cylindrical geometry with polygons.

5.1 Geometry Comparison

The purpose of this study is to calculate the geometry errors that DFEM and MOC codes incur when meshing problems of interest, particularly reactor problems. DFEM codes use polygons to approximate the cylindrical shape of fuel pins. MOC codes mesh the the problem exactly, but when computing transport calculations, use ‘tracks’ which are rectangular. The goal of this study is to show that while DFEM codes do not mesh the problem exactly, the errors they incur by meshing circles with polygons are comparable to those incurred by MOC codes when computing transport calculations with tracks.

5.1.1 DFEMs

Figure 5.1 shows the approximation DFEMs make when attempting to mesh a cylinder. We have a circle and an n sided polygon that share a center and an area. The areas of the parts of the polygon that lie inside the circle exactly equal the area of the parts of the polygon that lie outside the circle. We want to compute the total 'error' area, i.e. add up the areas of the polygon that are outside the circle and the areas of the circle that are outside the polygon without letting them cancel. This will give us as measure of the geometry error we incur by approximating a circle with an n sided polygon.

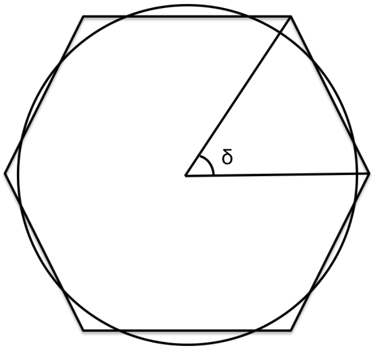

Figure 5.1: Circle and Approximating Polygon used in DFEMs

Because both the polygon and circle are radially symmetric, we can compute the 'error' areas for one side of the polygon and multiply by the number of sides, n. In Figure 5.2 we zoom in to one side of the polygon.

The radius of the circle is R and the sides of the triangle formed by the polygon are b, b, and p/n, where p is the perimeter of the polygon and n is the number of sides. The apothem of the polygon, which is the distance between the center and the middle of one side, is a. The angle subtended by both the polygon side and the circle arc is δ. The polygon intersects the circle two times. The angles subtended by these triangles/arcs are α and β, respectively. These two angles can be computed by finding the equation of the side of the polygon in radial coordinates and setting

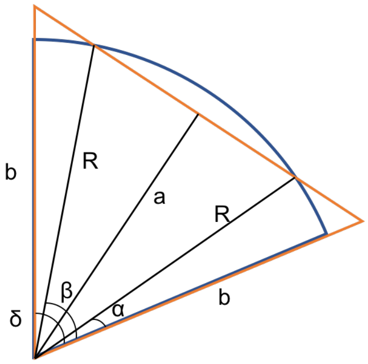

Figure 5.2: Zoom in on One 'Side'

the radius equal to R.

To do this, first let us define b in terms of R and n. To that end let us define the apothem in terms of the side length, s, and n:

$$a = \frac{1}{2} s \cot(\frac{\pi}{n}) \tag{5.1}$$

Now we know that the areas of the circle and polygon are equal:

$$\begin{aligned} \pi R^2 &= \frac{1}{2} nsa && (5.2) \\ &= \frac{1}{2} ns \left(\frac{1}{2} s \cot(\frac{\pi}{n}) \right) && (5.3) \\ &= \frac{1}{4} s^2 \cot(\frac{\pi}{n}) && (5.4) \\ s^2 &= \frac{4\pi R^2}{n \cot(\frac{\pi}{n})} && (5.5) \end{aligned}$$

Now we can plug this into the equation for b:

$$\begin{aligned} b &= \frac{1}{2} s \csc(\frac{\pi}{n}) && (5.6) \\ &= \sqrt{\frac{4\pi R^2}{2n \cot(\frac{\pi}{n})}} \csc(\frac{\pi}{n}) && (5.7) \\ &= \sqrt{\frac{2\pi R^2}{n}} \sqrt{\cot(\frac{\pi}{n})} \sec(\frac{\pi}{n}) && (5.8) \end{aligned}$$

The points that define the line segment of the side of the polygon are $(0, b)$ and $(b\cos(\pi - \delta), b\sin(\pi - \delta)$. The equation of the line is:

$$y = \left[\frac{\sin(\pi - \delta) - 1}{\cos(\pi - \delta)}\right] x + b \tag{5.9}$$

We can define a variable m as the term multiplying x in this equation. Putting this into radial coordinates gives:

$$r\sin(\theta) = mr\cos(\theta) + b \tag{5.10}$$

Now we need to solve this equation for r and set it equal to R:

$$r = \frac{b}{\sin(\theta) - m\cos(\theta)} = R \tag{5.11}$$

The roots of this equation are α and β as defined before:

$$\alpha = \tan^{-1}\left(\frac{-\sqrt{-b^2R^2 + m^2R^4 + R^4} - bmR}{m^2R^2 + R^2}, \frac{\frac{-mR\sqrt{-b^2R^2+m^2R^4+R^4}}{m^2R^2+R^2} - \frac{bm^2R^2}{m^2R^2+R^2} + b}{R}\right) \tag{5.12}$$

$$\beta = \tan^{-1}\left(\frac{-\sqrt{-b^2R^2 + m^2R^4 + R^4} - bmR}{m^2R^2 + R^2}, \frac{\frac{mR\sqrt{-b^2R^2+m^2R^4+R^4}}{m^2R^2+R^2} - \frac{bm^2R^2}{m^2R^2+R^2} + b}{R}\right) \tag{5.13}$$

Now the triangle/circle combination can be broken up into 3 pieces: the piece between the right edge and α, the piece between the two intersection points, and the piece between the β and the left edge. The error terms can be computed by taking the difference in areas between the circle and the polygon for each of these three sections and adding them up.

The pieces in between the edges and the intersection points are equal and therefore only the right side needs to be computed. Also, those two pieces added together equal the difference in area in between the intersection points. Therefore, 4 times the area in between the right edge and α is the total 'error' area.

$$\begin{aligned} E_\alpha &= A_{\text{tri}} - A_{\text{circ}} && (5.14) \\ &= \frac{1}{2}Rb\sin(\alpha) - \frac{1}{2}\alpha R^2 && (5.15) \\ E_{\text{total}} &= 4nE_\alpha && (5.16) \\ &= 4n\left(\frac{1}{2}Rb\sin(\alpha) - \frac{1}{2}\alpha R^2\right) && (5.17) \end{aligned}$$

Since we have expressed b in terms in R and n, the error can also be expressed solely in terms of R and n.

5.1.2 Method of Characteristics

A common misconception is that the method of characteristics makes no approximation when meshing circular geometries. This is not entirely true, particularly when computing transport solutions. The transport solutions are computed using what are called 'tracks'. These tracks intersect the problem geometry and are associated with a rectangular area. It is this rectangular area that produces the geometry error in MOC. Take, for example, Figure 5.3, which shows the track spacing for a vertically aligned direction. The red arrows are the tracks and the black rectangles are the areas associated with those tracks. There will be a picture like this for every azimuthal direction in the problem, but for simplicity we will only analyze this direction. We can do this because of the symmetry of the circular shape of the pin cell.

For simplicity, we will assume that there are an even number of tracks and that the tracks exactly span the circle. This is one of the more favorable cases for MOC, and is not always true. The x-location of the edges of the rectangles formed by the tracks are defined as x_i:

$$x_i = -R + i\frac{2R}{n} \tag{5.18}$$

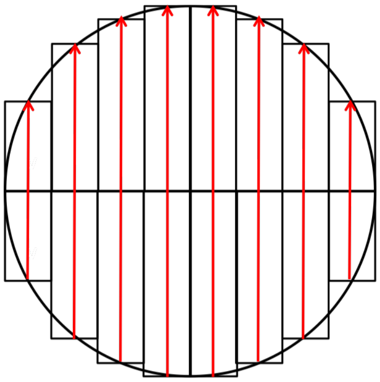

Figure 5.3: Track Spacing in MOC

The x-location of the centers of the tracks are defined as x_i':

$$x_i' = -R + \frac{2R}{n}\left(i + \frac{1}{2}\right) \tag{5.19}$$

Here n is the number of tracks within the circle and R is the radius of the circle. The center of the tracks are also the intersection points of the tracks with the circle. The equation of the circle is

$$y = \sqrt{R^2 - x^2}, \tag{5.20}$$

while the equation of the top of the track is

$$y = \sqrt{R^2 - x_i'^2}. \tag{5.21}$$

We can split the circle into two parts: the top and the bottom. We will only analyze the top of the circle and multiply by two due to symmetry. We will also break up the tracks into two pieces:

those left of center and those right of center. This will allow for easier computation of the error.

$$
\begin{aligned}
E &= 2\sum_{i=1}^{n/2}\left(\int_{x_i}^{x'_i}\sqrt{R^2-x_i'^2}-\sqrt{R^2-x^2}+\int_{x'_i}^{x_{i+1}}\sqrt{R^2-x^2}-\sqrt{R^2-x_i'^2}\right)\\
&+ 2\sum_{i=n/2+1}^{n-1}\left(\int_{x_i}^{x'_i}\sqrt{R^2-x^2}-\sqrt{R^2-x_i'^2}+\int_{x'_i}^{x_{i+1}}\sqrt{R^2-x_i'^2}-\sqrt{R^2-x^2}\right) \quad (5.22)
\end{aligned}
$$

5.1.3 Results

Figure 5.4 shows the results of the geometry error study, based on equations 5.17 and 5.22, setting the radius of the circle to one. The y axis, which represents the error, has a logarithmic scale. The x axis is the n, which is the number of sides to the polygon in the DFEM case, and the number of tracks in the MOC case. As can be seen in Figure 5.4, polygons (used in DFEMs) converge much faster than stair-stepping (used in MOC).

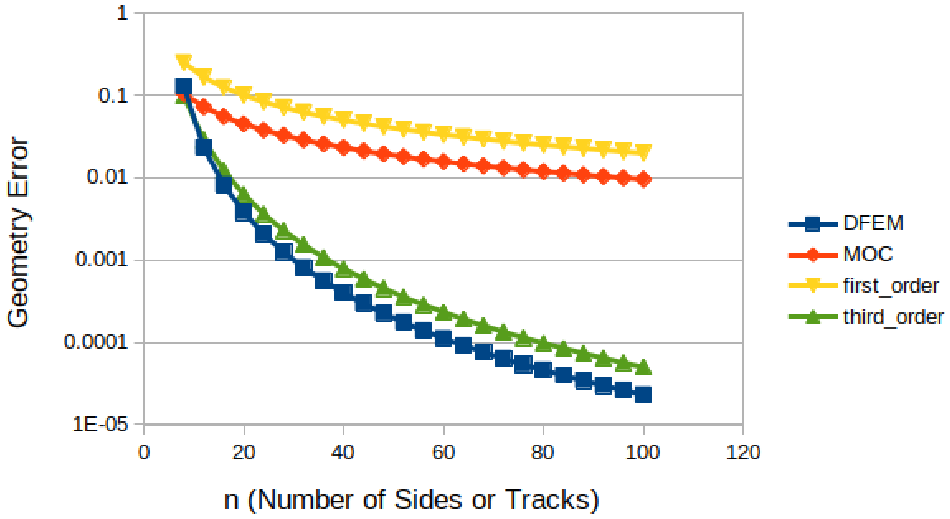

Figure 5.4: Comparison of Geometry Error between DFEMs and MOC

The shape of these two curves are as expected. MOC approximates a circle effectively with Riemann sums, while DFEMs approximate a circle with something similar to the trapezoidal rule.

The polygonal representation is third order accurate, meaning that if we increase the number of sides of the polygon by a factor of two, the error in the geometric approximation goes down by a factor of eight. Riemann sums are first order accurate, meaning if we increase the number of tracks by a factor of two, then the geometric approximation error goes down by a factor of two. We see this in Figure 5.4.

5.2 Numerical Comparison

Now we move on to the numerical study. The purpose of this study is to show for which type of problems each method, MOC and DFEMs, is more accurate, given a set level of spatial discretization. We have run a series of problems, both with and without a thin layer of IFBA, or integral fuel burnable absorber, around the fuel. We expect to show that for problems with large numbers of energy groups and with thin annular regions, LD will outperform MOC. To this purpose, we have run both pincell and 2D assembly problems and compared quantities of interest such as k-eff, reaction rates, and pin powers.

5.2.1 Pin Cell Calculations

To start, we ran a series of pin cell calculations taken from the VERA benchmark suite - specifically problems 1B and 1E. We modified problem 1B to match the fuel temperature of problem 1E, which has a layer IFBA around the fuel. This was done to determine the effect adding a layer of IFBA has on our solution metrics. This also means that we do not have a benchmark solution to our version of problem 1B.

Figure 5.5 shows the pin cell meshes used in problem 1B. The MOC code's we compared to, MPACT, mesh has three volumetrically equal radial rings in the fuel, one in the gap, one in the cladding, and two in the moderator. The DFEM mesh has three radial rings in the fuel (one being in the outer 10% - from our energy study), one ring in the gap, one in the cladding, and two rings in the moderator. These meshes were chosen because of their similar number of spatial cells. The mesh for MOC in the left image of Figure 5.5 shows more azimuthal segments than were run in the calculation (8). This is because the visualization software used to visualize these meshes does not allow curved mesh boundaries, and the increased number of azimuthal segments makes the circles look more circular. MPACT also used a 0.05 cm track spacing for problem 1B.

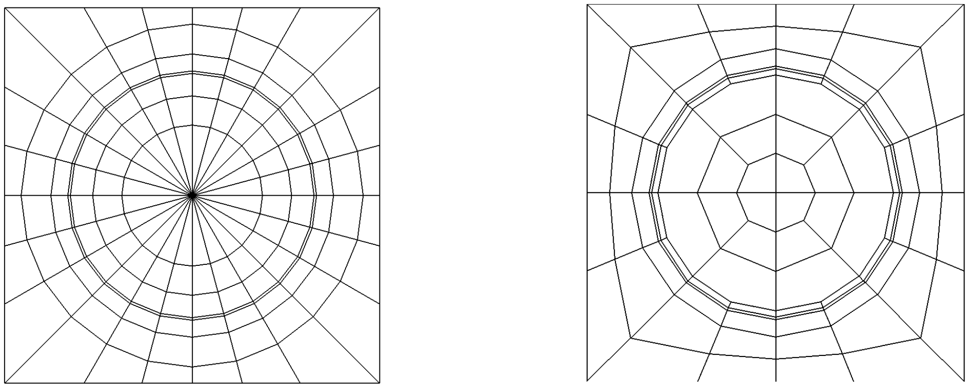

Figure 5.5: Meshes for MOC (left) and DFEMs (right) for Problem 1B (modified) of the VERA Benchmark

Figure 5.6 shows the meshes used by MOC and DFEMs for problem 1E. The meshes are similar to those in 1B, with the addition of a thin ring around the fuel for the IFBA. For the IFBA runs, MPACT used a track spacing of 0.005 cm and 8 azimuthal segments.

In addition to similar meshes, the same cross sections (multi-group) and quadrature were used between the two codes. Two cross section sets were used, a 51 group and a 252 group structure. The quadrature was a Tabuchi-Yamamoto-Chebychev product quadrature with 16 azimuthal angles and 3 polar angles per quadrant. This was taken from MPACT.

5.2.1.1 51 Groups

The first thing we looked at is k-effective between the two codes for the 51 group cross sections. Table 5.1 shows those results. We got a 157 pcm difference in k-effective between MOC and DFEMs for problem 1B using the 51 group cross section set. Because we modified problem 1B, we do not have a benchmark solution to compare to. To rectify this, we ran a highly resolved

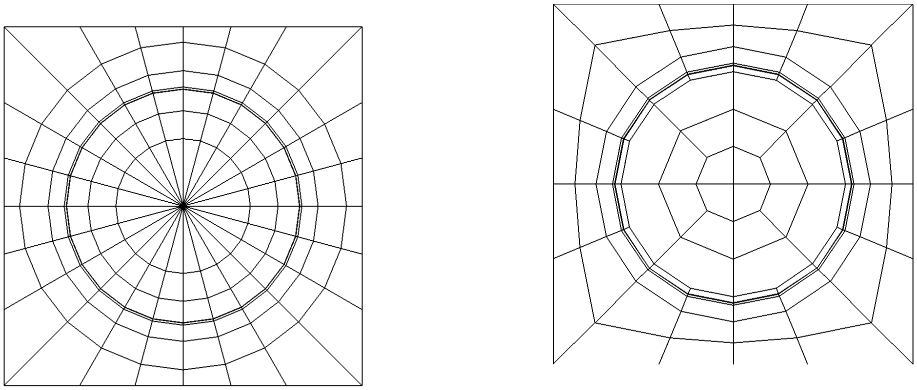

Figure 5.6: Meshes for MOC (left) and DFEMs (right) for Problem 1E of the VERA Benchmark

version of our modified problem using our DFEM code to serve as a benchmark answer. This highly resolved problem used a (10,5) mesh with 48 azimuthal and 48 polar angles per octant. From our studies in Chapter 3, we believe this problem has less than 1 pcm difference to the true k-effective. MOC had a 154 pcm difference to this benchmark solution and DFEMs had a -4 pcm difference.

Table 5.1: k-effective and Relative Difference (pcm) to Benchmark for Problem 1B with 51 Groups

	MOC	DFEM
k-effective	1.183006	1.1848683
pcm difference	154	-4

We then looked at problem 1E of the VERA benchmark suite. It is a single pin cell with a thin layer of IFBA (a burnable absorber) around the fuel. When comparing to the MCNP benchmark solution, MOC had a -250 pcm relative difference and PDT had a 305 pcm relative difference in k-effective. The magnitude of the error is mostly due to the energy discretization, as the benchmark solution was computed using continuous energy cross sections. As a better comparison, we again ran a highly resolved version of problem 1E (using the same mesh and quadrature as problem 1B's benchmark). When comparing to the DFEM benchmark solution, MOC had a -490 pcm relative difference and DFEMs had a 63 pcm relative difference in k-effective. The relative difference in pcm between each code and the DFEM benchmark is seen in Table 5.2.

Table 5.2: k-effective and Relative Difference (pcm) to Benchmark for Problem 1E with 51 Groups

	MOC	DFEM
k-effective	0.769760	0.7740459
pcm difference (benchmark)	-250	305
pcm difference (DFEM benchmark)	-490	63

Next we looked at reaction rates in the fuel pin, specifically the group-summed volume-averaged absorption rate and ν times the group-summed volume-averaged fission rate. Before we could compare them, though, we had to renormalize our solutions, due to the fact that the the codes normalize their flux using different metrics. We did this by taking the energy group-summed volume-averaged flux throughout the problem and using the ratio of this metric between the codes as our normalization factor.

First we compared the group-summed volume-averaged absorption rate in the fuel. Table 5.3 shows this quantity of interest for MOC and DFEMs, for both problems 1B (left) and 1E (right), and the relative difference (in pcm) between the codes and the benchmark solution. We used the same high resolution run as our benchmark solution as we did when comparing k-effective. DFEMs had better agreement to the benchmark for both problems 1B and 1E.

Next we compared the group-summed volume-averaged ν times the fission rate in the fuel. Table 5.4 shows this quantity of interest for MOC and DFEM, for both problems 1B (left) and 1E (right), and the relative difference (in pcm) between the codes and the benchmark solution.DFEMs and MOC have comparable differences to the benchmark solution for problem 1B, but DFEM is much closer to the benchmark for problem 1E.

Table 5.3: Absorption Rate for 1B (left) and 1E (right) and Relative Difference to the Benchmark Solution for 51 Groups

	MOC	DFEM
Abs. Rate	1.6458	1.6448
pcm difference	55	-5

	MOC	DFEM
Abs. Rate	1.1582	1.2097
pcm difference	-4247	9

Table 5.4: ν times Fission Rate for 1B (left) and 1E (right) and Relative Difference to the Benchmark Solution for 51 Groups

	MOC	DFEM
Fission Rate	2.2483	2.2480
pcm difference	16	3

	MOC	DFEM
Fission Rate	1.4540	1.4686
pcm difference	-929	63

5.2.1.2 252 Group - MultiGroup

For the 252 group cross section set, again we started by comparing k-effective for both codes against a reference solution. We again used a (10,5) mesh and 48 azimuthal angles and 48 polar angles per octant for our reference solution. Tables 5.5 and 5.6 show those comparisons for 1B and 1E, respectively. DFEM is more accurate for problem 1B and MOC is more accurate for problem 1E, at least when comparing k-effective.

Table 5.5: k-effective and Relative Difference (pcm) to Benchmark for Problem 1B with 252 Groups

	MOC	DFEM
k-effective	1.217555	1.216991
pcm difference	51	5

Next we looked at reaction rates - again the group-summed volume averaged absorption rate and ν times the fission rate. We used the same normalization as the 51 group set - the group-

Table 5.6: k-effective and Relative Difference (pcm) to Benchmark for Problem 1E with 252 Groups

	MOC	DFEM
k-effective	0.785407	0.786202
pcm difference	-28	73

summed volume-averaged flux. Tables 5.7 and 5.8 show the comparisons for the absorption rate and ν times the fission rate, respectively.

Table 5.7: Absorption Rate for 1B (left) and 1E (right) and Relative Difference to the Benchmark Solution for 252 Groups

	MOC	DFEM
Abs. Rate	1.63146	1.63234
pcm difference	-59	-5

	MOC	DFEM
Abs. Rate	1.1751	1.1764
pcm difference	-111	5

Table 5.8: ν times the Fission Rate for 1B (left) and 1E (right) and Relative Difference to the Benchmark Solution for 252 Groups

	MOC	DFEM
Abs. Rate	2.3055	2.30897
pcm difference	-145	5

	MOC	DFEM
Abs. Rate	1.4874	1.4916
pcm difference	-210	75

The relative differences in both these QOIs are larger for both problems for MOC than they are for DFEMs. This is due the solution being much more dependent on resolving spatial characteristics of the problem. We believe our DFEM code does this with the mesh we chose, while the MOC code does not.

5.2.2 2D Assembly Calculations

A significant portion of the errors we compute in our QOIs for our pin cell problems are due to the cross sections. We wish to explore spatial errors without the complication of energy discretization. To this end, we chose the 2D C5G7 problem to study next, which has pre-generated 7-group multi-group cross sections. This is the same problem we studied in our simplified reactor model in Chapter 2. In addition to the benchmark problem, we have also studied the 2D C5G7 problem with a gap and clad - a problem that is closer to realistic reactor geometries. We used a higher resoltuion quadrature than our pin cell calculations. We used a Tabuchi-Yamamoto-Chebychev product quadrature with 64 azimuthal angles and 3 polar angles per quadrant. This was to ensure the errors we incur with our quadrature approximation were small compared to our spatial errors.

5.2.2.1 2D C5G7

To begin, we ran the 2D C5G7 problem as it was specified in the benchmark [15]. Figure 5.7 shows the meshes the MOC code used in modelling this problem. On the left is their low resolution mesh and on the right is their high resolution mesh. In their low resolution case, they used a track spacing of 0.03 cm, and in their high resolution case they used a track spacing of 0.001 cm. The azimuthal segments used in computation (rather than visualization) are bolded in Figure 5.7. The low resolution mesh has five radial rings in the fuel, and three in the moderator. The high resolution mesh has ten radial rings in the fuel and six in the moderator. Both low and high resolution meshes use flat source regions.

Figure 5.8 shows the mesh used by DFEM to model the 2D C5G7 problem. There are four radial rings in the fuel and two in the moderator. A lower resolution mesh was also run with three rings in the fuel and one in the moderator, and the QOIs computed using this mesh were close enough to the (4,2) mesh to say that the (4,2) mesh is sufficiently resolved.

Table 5.9 shows the results of our comparisons for different QOIs between MOC and DFEMs. The results in this table are relative differences (in pcm) between each methods QOIs and the benchmark QOIs. The QOIs we chose to compare were the maximum and minimum pin power,

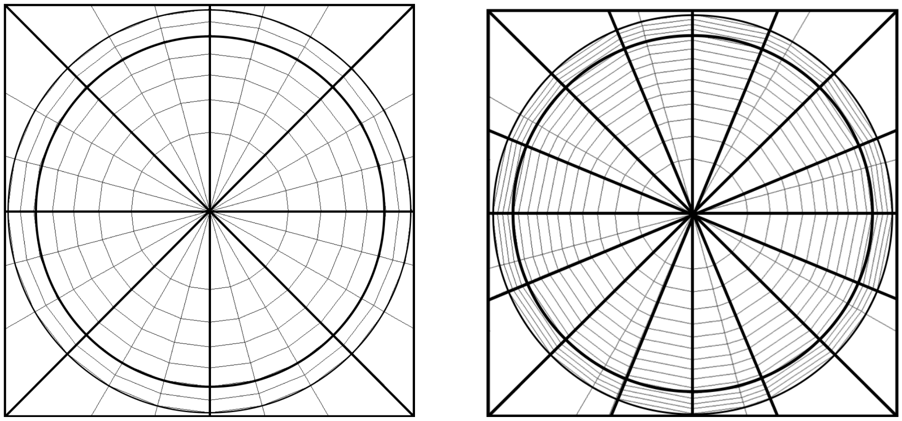

Figure 5.7: MOC Low Res (left): 0.03 cm Spacing, and High Res (right): 0.001 cm Spacing Pin Cell Meshes

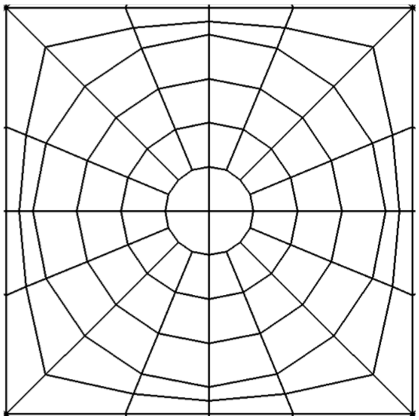

Figure 5.8: DFEM C5G7 Pin Cell Mesh

the assembly powers in the inner and outer UO_2 assembly and the MOX assembly, and k-effective. The first row is the DFEM results, while the last two rows are the low and high resolution MOC results, respectively.

First look at the last column: k-effective. Both spatial discretization methods compute k-effective to within 10 pcm of the benchmark solution. This is expected, as k-effective is a integral quantity and therefore is easier to compute accurately, due to cancellation of errors. The other columns - the pin powers and assembly powers - paint a different picture. These QOIs depend

Table 5.9: QOI Comparison (Relative Difference in pcm) for the 2D C5G7 Problem between MOC and DFEMs

	Max Pin	Min Pin	Inner UO2	MOX	Outer UO2	k
$\frac{\text{DFEM-MCNP}}{\text{MCNP}}$	51	-270	37	-41	-8	5
$\frac{\text{MOC}^{low}\text{-MCNP}}{\text{MCNP}}$	313	-1730	245	-207	-237	-10
$\frac{\text{MOC}^{high}\text{-MCNP}}{\text{MCNP}}$	-78	611	-62	76	-12	5

more heavily on the granular detail in space of the transport solution. The DFEM solutions to these QOIs are closer to the benchmark solution than both the low and high resolution MOC solutions for all of the QOIs, sometimes by an order of magnitude, even with fewer numbers of spatial unknowns. We believe this is due to MOC code's meshes not resolving the spatial characteristics of the problem well enough to compute the QOIs accurately.

5.3 Conclusions

We have compared an MOC code to a DFEM code for several problems - both pin cell level and assembly level. We have found that, in general, DFEMs compute more accurate solutions to a range of QOIs, for a given level of spatial resolution compared to MOC. We have also found that k-effective is easier to compute accurately, compared to more granular QOIs such as pin powers.

When going from 2D to 3D, the complexity of LD goes from computing 3x3 matrices to 4x4 matrices. MOC must add many more tracks per cell to achieve adequate accuracy in 3D versus 2D. Also, in general, the ray tracing in 3D becomes much more costly. Because of this, for 3D reactor analysis, LD looks especially promising.

6. OSCILLATIONS

We have observed unphysical oscillations in the solutions of simple k-eigenvalue problems, including problems as simple as homogeneous rectangles with vacuum boundaries and a single energy group. Here we analyze the relevant equations in an attempt to understand the oscillations and the conditions under which they can occur.

6.1 LD Equations

Consider a 2D Cartesian-geometry k-eigenvalue problem with one energy group, vacuum boundaries, and with a rectangular grid that has uniform cell spacing of $\Delta x \times \Delta y$. The LD discrete-ordinates equations for the i, j-th cell include the exact 0th- x-, and y-moment equations:

$$\mu_m \frac{\psi_{m,i+1/2,j} - \psi_{m,i-1/2,j}}{\Delta x} + \eta_m \frac{\psi_{m,i,j+1/2} - \psi_{m,i,j-1/2}}{\Delta y} + \sigma_t \psi_{mij} = \left(\frac{\nu \sigma_f}{k} + \sigma_s \right) \frac{\phi_{ij}}{4\pi} , \tag{6.1}$$

$$3\mu_m \frac{\psi_{m,i+1/2,j} + \psi_{m,i-1/2,j} - 2\psi_{mij}}{\Delta x} + \eta_m \frac{\psi^x_{m,i,j+1/2} - \psi^x_{m,i,j-1/2}}{\Delta y} + \sigma_t \psi^x_{mij} = \left(\frac{\nu \sigma_f}{k} + \sigma_s \right) \frac{\phi^x_{ij}}{4\pi} , \tag{6.2}$$

$$\mu_m \frac{\psi^y_{m,i+1/2,j} - \psi^y_{m,i-1/2,j}}{\Delta x} + 3\eta_m \frac{\psi_{m,i,j+1/2} + \psi_{m,i,j-1/2} - 2\psi_{mij}}{\Delta y} + \sigma_t \psi^y_{mij} = \left(\frac{\nu \sigma_f}{k} + \sigma_s \right) \frac{\phi^y_{ij}}{4\pi} , \tag{6.3}$$

along with closure equations:

$$\psi_{m,i\pm 1/2,j} = \psi_{mij} \pm \psi^x_{mij}, \quad \mu_m \gtrless 0 , \tag{6.4}$$

$$\psi_{m,i,j\pm 1/2} = \psi_{mij} \pm \psi^y_{mij}, \quad \eta_m \gtrless 0 , \tag{6.5}$$

$$\psi^y_{m,i\pm 1/2,j} = \psi^y_{mij}, \quad \mu_m \gtrless 0 , \tag{6.6}$$

$$\psi^x_{m,i,j\pm 1/2} = \psi^x_{mij}, \quad \eta_m \gtrless 0 , \tag{6.7}$$

and vacuum boundary conditions, which specify that the incoming angular flux is zero on problem boundaries.

The 0th angular moments[1] of the LD system are obtained by operating on Eqs. (6.1)-(6.3) with $\sum_m w_m(\cdot)$:

$$\frac{J_{\mu,i+1/2,j} - J_{\mu,i-1/2,j}}{\Delta x} + \frac{J_{\eta,i,j+1/2} - J_{\eta,i,j-1/2}}{\Delta y} = \left(\frac{\nu\sigma_f}{k} - \sigma_a\right)\phi_{ij} \ , \tag{6.8}$$

$$3\frac{J_{\mu,i+1/2,j} + J_{\mu,i-1/2,j} - 2J_{\mu ij}}{\Delta x} + \frac{J^x_{\eta,i,j+1/2} - J^x_{\eta,i,j-1/2}}{\Delta y} = \left(\frac{\nu\sigma_f}{k} - \sigma_a\right)\phi^x_{ij} \ , \tag{6.9}$$

$$\frac{J^y_{\mu,i+1/2,j} - J^y_{\mu,i-1/2,j}}{\Delta x} + 3\frac{J_{\eta,i,j+1/2} + J_{\eta,i,j-1/2} - 2J_{\eta ij}}{\Delta y} = \left(\frac{\nu\sigma_f}{k} - \sigma_a\right)\phi^y_{ij} \ . \tag{6.10}$$

At this point we have made no approximations to the LD equations.

6.2 Test Problem

We consider a problem that has only one cell in the x dimension, of width a. There are N_y cells in the y dimension, with $\Delta y = b/N_y$. The problem has vacuum boundaries. Because there is only one cell in x, we drop the i index. By symmetry we have several simplifying identities (still without approximation):

$$\phi^x_j = 0 \ , \tag{6.11}$$

$$\phi^{xy}_j = 0 \ . \tag{6.12}$$

Equation 6.11 is due to the x symmetry in the problem, and Equation 6.12 states that the LD formulation of the problem does not allow for the y-slope to depend on x and vice versa.

Because of the symmetry we can solve the problem for $\mu > 0$ and infer everything we need

---

[1] We use nonstandard notation for the components of the net current density, to avoid confusion between angular and spatial moments. The x and y components of the net current density are normally denoted J_x and J_y, but we use J_μ and J_η. Then we use x and y superscripts to denote first spatial moments in x and y.

from that. For $\mu > 0$ with one cell in x, Eqs. (6.1) - (6.3) become

$$\mu_m \frac{\psi_{m,3/2,j} - 0}{\Delta x} + \eta_m \frac{\psi_{m,j+1/2} - \psi_{m,j-1/2}}{\Delta y} + \sigma_t \psi_{mj} = \frac{Q_j}{4\pi} \equiv \left(\frac{\nu \sigma_f}{k} + \sigma_s \right) \frac{\phi_j}{4\pi} \ , \tag{6.13}$$

$$3 \frac{\psi_{m,3/2,j} + 0 - 2\psi_{mj}}{\Delta x} + \eta_m \frac{\psi^x_{m,j+1/2} - \psi^x_{m,j-1/2}}{\Delta y} + \sigma_t \psi^x_{mj} = 0 \ , \tag{6.14}$$

$$\mu_m \frac{\psi^y_{m,3/2,j} - 0}{\Delta x} + 3\eta_m \frac{\psi_{m,j+1/2} + \psi_{m,j-1/2} - 2\psi_{mj}}{\Delta y} + \sigma_t \psi^y_{mj} = \frac{Q^y_j}{4\pi} \equiv \left(\frac{\nu \sigma_f}{k} + \sigma_s \right) \frac{\phi^y_j}{4\pi} \ , \tag{6.15}$$

Define

$$\tau_x \equiv \sigma_t a \ , \tag{6.16}$$

$$\tau_y \equiv \sigma_t \Delta y \ , \tag{6.17}$$

$$\tilde{q}_j \equiv \frac{Q_j}{4\pi\sigma_t} - \eta_m \frac{\psi_{m,j+1/2} - \psi_{m,j-1/2}}{\tau_y} \ , \tag{6.18}$$

$$\tilde{q}^x_j \equiv -\eta_m \frac{\psi^x_{m,j+1/2} - \psi^x_{m,j-1/2}}{\tau_y} \ . \tag{6.19}$$

Recall that $\psi_{3/2,j} = \psi^x_j + \psi_j$ and $\psi^y_{3/2,j} = \psi^{xy}_j + \psi^x_j \xrightarrow{LD} \psi^y_j$. Then

$$\frac{\mu_m}{\tau_x} \left(\psi^x_j + \psi_j \right)_m + \psi_{mj} = \tilde{q}_j$$
$$\Rightarrow \psi_j = \frac{\tilde{q}_j \tau_x - \mu_m \psi^x_{mj}}{\tau_x + \mu_m} \ , \tag{6.20}$$

$$3 \frac{\mu_m}{\tau_x} \left(\psi^x_j - \psi_j \right)_m + \psi^x_{mj} = \tilde{q}^x_j$$
$$\Rightarrow \psi^x_j \left(\tau_x + 3\mu_m \right) = \tilde{q}^x_j \tau_x + 3\mu_m \psi_{mj} \ , \tag{6.21}$$

$$3\eta_m \frac{\psi_{m,j+1/2} + \psi_{m,j-1/2} - 2\psi_{mj}}{\sigma_t \Delta y} + \psi^y_{mj} \left(1 + \frac{\mu_m}{\tau_x} \right) = \frac{Q^y_j}{4\pi\sigma_t} \ , \tag{6.22}$$

Algebra on the average and x-moment equations yields:

$$\begin{aligned}
\psi_j^x\left(\tau_x + 3\mu_m\right) &= \tilde{q}_j^x \tau_x + 3\mu_m \frac{\tilde{q}_j \tau_x - \mu_m \psi_{mj}^x}{\tau_x + \mu_m} \ , \\
\Rightarrow \psi_j^x\left(\tau_x + 3\mu_m + 3\mu_m \frac{\mu_m}{\tau_x + \mu_m}\right) &= \tilde{q}_j^x \tau_x + \tilde{q}_j 3\mu_m \frac{\tau_x}{\tau_x + \mu_m} \ , \\
\Rightarrow \psi_j^x\left(\tau_x^2 + \tau_x \mu_m + 3\mu_m^2 + 3\mu_m \tau_x + 3\mu_m^2\right) &= \tilde{q}_j^x \tau_x(\tau_x + \mu_m) + \tilde{q}_j 3\mu_m \tau_x \ ,
\end{aligned}$$

$$\Rightarrow \psi_j^x\left(\frac{\tau_x^2}{\mu_m^2} + 4\frac{\tau_x}{\mu_m} + 6\right) = \tilde{q}_j^x \frac{\tau_x}{\mu_m}\left(\frac{\tau_x}{\mu_m} + 1\right) + \tilde{q}_j \frac{3\tau_x}{\mu_m} \ . \tag{6.23}$$

Define

$$d \equiv \frac{\tau_x^2}{\mu_m^2} + 4\frac{\tau_x}{\mu_m} + 6 \tag{6.24}$$

Return to Eq. (6.20) and manipulate:

$$\begin{aligned}
\left(\frac{\mu_m}{\tau_x} + 1\right)\psi_{jm} &= \tilde{q}_j - \frac{\mu_m}{\tau_x}\psi_j^x \\
&= \tilde{q}_j - \frac{\mu_m}{\tau_x}\frac{1}{d}\left[\tilde{q}_j^x \frac{\tau_x}{\mu_m}\left(\frac{\tau_x}{\mu_m} + 1\right) + \tilde{q}_j \frac{3\tau_x}{\mu_m}\right] \\
&= \frac{d-3}{d}\tilde{q}_j - \tilde{q}_j^x \frac{1}{d}\left(\frac{\tau_x}{\mu_m} + 1\right)
\end{aligned}$$

$$\Rightarrow \psi_{jm}\left(1 + \frac{\mu_m}{\tau_x}\right)\left(\frac{d}{d-3}\right) = \tilde{q}_j - \tilde{q}_j^x \frac{1 + \tau_x/\mu_m}{d-3} \tag{6.25}$$

Insert definition of $\tilde{q}_j$, summarize results, and also repeat the y-moment equation:

$$\frac{\eta_m}{\tau_y}\left(\psi_{m,j+1/2}-\psi_{m,j-1/2}\right)+\psi_{jm}\left(1+\frac{\mu_m}{\tau_x}\right)\left(\frac{d}{d-3}\right)=\frac{Q_j}{4\pi\sigma_t}-\tilde{q}_j^x\frac{1+\tau_x/\mu_m}{d-3}\ , \quad \mu_m>0, \tag{6.26}$$

$$\psi_j^x=\tilde{q}_j^x\frac{\tau_x}{\mu_m d}\left(\frac{\tau_x}{\mu_m}+1\right)+\tilde{q}_j\frac{3\tau_x}{\mu_m d}\ , \quad \mu_m>0\ , \tag{6.27}$$

$$3\frac{\eta_m}{\tau_y}\left(\psi_{m,j+1/2}+\psi_{m,j-1/2}-2\psi_{mj}\right)+\psi_{mj}^y\left(1+\frac{\mu_m}{\tau_x}\right)=\frac{Q_j^y}{4\pi\sigma_t} \quad \mu_m>0\ . \tag{6.28}$$

We recognize that for the S$_2$ equations there are four directions that we can characterize by $m=$ NE, NW, SE, and SW. We define:

$$\psi^N\equiv\pi\left(\psi^{NE}+\psi^{NW}\right), \tag{6.29}$$

$$\psi^S\equiv\pi\left(\psi^{SE}+\psi^{SW}\right), \tag{6.30}$$

$$\mu_0\equiv\frac{1}{\sqrt{3}}. \tag{6.31}$$

We also recognize by symmetry that

$$\psi_j^{x,NW}=-\psi_j^{x,NE}, \tag{6.32}$$

$$\tilde{q}_j^{x,NW}=-\tilde{q}_j^{x,NE}, \tag{6.33}$$

$$\psi_j^{x,SW}=-\psi_j^{x,SE}, \tag{6.34}$$

$$\tilde{q}_j^{x,SW}=-\tilde{q}_j^{x,SE} \tag{6.35}$$

We spell out the balance equations for the S$_2$ case:

$$\frac{\mu_0}{\tau_y}\left(\psi_{j+1/2}^{NE}-\psi_{j-1/2}^{NE}\right)+\psi_j^{NE}\left(1+\frac{\mu_0}{\tau_x}\right)\left(\frac{d}{d-3}\right)=\frac{Q_j}{4\pi\sigma_t}+\frac{1+\tau_x/\mu_0}{d-3}\frac{\mu_0}{\tau_y}\left(\psi_{j+1/2}^{x,NE}-\psi_{j-1/2}^{x,NE}\right)\ , \tag{6.36}$$

$$\frac{\mu_0}{\tau_y}\left(\psi_{j+1/2}^{NW}-\psi_{j-1/2}^{NW}\right)+\psi_j^{NW}\left(1+\frac{\mu_0}{\tau_x}\right)\left(\frac{d}{d-3}\right)=\frac{Q_j}{4\pi\sigma_t}-\frac{1+\tau_x/\mu_0}{d-3}\frac{\mu_0}{\tau_y}\left(\psi_{j+1/2}^{x,NW}-\psi_{j-1/2}^{x,NW}\right)\ , \tag{6.37}$$

$$-\frac{\mu_0}{\tau_y}\left(\psi^{SE}_{j+1/2}-\psi^{SE}_{j-1/2}\right)+\psi^{SE}_j\left(1+\frac{\mu_0}{\tau_x}\right)\left(\frac{d}{d-3}\right)=\frac{Q_j}{4\pi\sigma_t}-\frac{1+\tau_x/\mu_0}{d-3}\frac{\mu_0}{\tau_y}(\psi^{x,SE}_{j+1/2}-\psi^{x,SE}_{j-1/2})\ , \tag{6.38}$$

$$-\frac{\mu_0}{\tau_y}\left(\psi^{SW}_{j+1/2}-\psi^{SW}_{j-1/2}\right)+\psi^{SW}_j\left(1+\frac{\mu_0}{\tau_x}\right)\left(\frac{d}{d-3}\right)=\frac{Q_j}{4\pi\sigma_t}+\frac{1+\tau_x/\mu_0}{d-3}\frac{\mu_0}{\tau_y}(\psi^{x,SW}_{j+1/2}-\psi^{x,SW}_{j-1/2})\ . \tag{6.39}$$

Henceforth, d is understood to be evaluated with $\mu_m = \mu_0$. Recall that $\tilde{q}^{x,NW}_j = -\tilde{q}^{x,NE}_j$, with similar relation for the South directions. Add the two North equations and add the two South equations, after multiplying by 2π:

$$\frac{\mu_0}{\tau_y}\left(\psi^{N}_{j+1/2}-\psi^{N}_{j-1/2}\right)+\psi^{N}_j\left(1+\frac{\mu_0}{\tau_x}\right)\left(\frac{d}{d-3}\right)=\frac{Q_j}{2\sigma_t}+2\pi\frac{1+\tau_x/\mu_0}{d-3}\frac{\mu_0}{\tau_y}(\psi^{x,NE}_{j+1/2}-\psi^{x,NE}_{j-1/2})\ , \tag{6.40}$$

$$-\frac{\mu_0}{\tau_y}\left(\psi^{S}_{j+1/2}-\psi^{S}_{j-1/2}\right)+\psi^{S}_j\left(1+\frac{\mu_0}{\tau_x}\right)\left(\frac{d}{d-3}\right)=\frac{Q_j}{2\sigma_t}-2\pi\frac{1+\tau_x/\mu_0}{d-3}\frac{\mu_0}{\tau_y}(\psi^{x,SE}_{j+1/2}-\psi^{x,SE}_{j-1/2})\ . \tag{6.41}$$

Do the same for the y-moment equations, which do not have any complicating terms:

$$\frac{\mu_0}{\tau_y}\left(\psi^{N}_{j+1/2}+\psi^{N}_{j-1/2}-2\psi^{N}_j\right)+\frac{1}{3}\psi^{y,N}_j\left(1+\frac{\mu_0}{\tau_x}\right)=\frac{1}{3}\frac{Q^y_j}{2\sigma_t}\ , \tag{6.42}$$

$$-\frac{\mu_0}{\tau_y}\left(\psi^{S}_{j+1/2}+\psi^{S}_{j-1/2}-2\psi^{S}_j\right)+\frac{1}{3}\psi^{y,S}_j\left(1+\frac{\mu_0}{\tau_x}\right)=\frac{1}{3}\frac{Q^y_j}{2\sigma_t}\ ,\ . \tag{6.43}$$

Work on the x moment, recognizing that $\psi_j^{x,NE} \xrightarrow{LD} \psi_{j+1/2}^{x,NE}$:

$$\begin{aligned} 2\pi\psi_{j+1/2}^{x,NE} = 2\pi\psi_j^{x,NE} &= 2\pi\tilde{q}_j \tfrac{3\tau_x}{\mu_0 d} + 2\pi\tilde{q}_j^x \tfrac{\tau_x}{\mu_0 d}\left(\tfrac{\tau_x}{\mu_0}+1\right) \\ &= \tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t} - \tfrac{\mu_0}{\tau_y}2\pi(\psi_{j+1/2}^{NE} - \psi_{j-1/2}^{NE})\right] \\ &\quad + \tfrac{\tau_x}{\mu_0 d}\left(\tfrac{\tau_x}{\mu_0}+1\right)\left[-\tfrac{\mu_0}{\tau_y}2\pi(\psi_{j+1/2}^{x,NE} - \psi_{j-1/2}^{x,NE})\right] \end{aligned} \tag{6.44}$$

Recall that $\psi_{j+1/2}^{NE} = \psi_{j+1/2}^{NW} \Rightarrow 2\pi\psi_{j+1/2}^{NE} = \psi_{j+1/2}^{N}$. Define:

$$\xi \equiv \tfrac{\tau_x}{\mu_0 d}\left(\tfrac{\tau_x}{\mu_0}+1\right)\tfrac{\mu_0}{\tau_y} = \tfrac{1}{d}\tfrac{\tau_x}{\tau_y}\left(1+\tfrac{\tau_x}{\mu_0}\right). \tag{6.45}$$

Then:

$$2\pi\psi_{j+1/2}^{x,NE} = \tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t} - \tfrac{\mu_0}{\tau_y}(\psi_{j+1/2}^{N} - \psi_{j-1/2}^{N})\right] - \xi 2\pi(\psi_{j+1/2}^{x,NE} - \psi_{j-1/2}^{x,NE}) \tag{6.46}$$

and

$$2\pi\psi_{j+1/2}^{x,NE}(1+\xi) = \tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t} - \tfrac{\mu_0}{\tau_y}(\psi_{j+1/2}^{N} - \psi_{j-1/2}^{N})\right] + \xi 2\pi\psi_{j-1/2}^{x,NE} \tag{6.47}$$

or

$$2\pi(\psi_{j+1/2}^{x,NE} - \psi_{j-1/2}^{x,NE}) = (\tfrac{\xi}{1+\xi} - 1)2\pi\psi_{j-1/2}^{x,NE} + \tfrac{1}{1+\xi}\tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t} - \tfrac{\mu_0}{\tau_y}(\psi_{j+1/2}^{N} - \psi_{j-1/2}^{N})\right] \tag{6.48}$$

or

$$\boxed{2\pi(\psi_{j+1/2}^{x,NE} - \psi_{j-1/2}^{x,NE}) = -\tfrac{1}{1+\xi}2\pi\psi_{j-1/2}^{x,NE} + \tfrac{1}{1+\xi}\tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t} - \tfrac{\mu_0}{\tau_y}(\psi_{j+1/2}^{N} - \psi_{j-1/2}^{N})\right] \qquad (6.49)}$$

In words, this says that: (out-slope minus in-slope) = (negative #)times in-slope + positive number times source + negative number times (out-avg - in-avg). Write the similar expression for SE direction:

$$\boxed{-2\pi(\psi^{x,SE}_{j+1/2} - \psi^{x,SE}_{j-1/2}) = -\tfrac{1}{1+\xi}2\pi\psi^{x,SE}_{j+1/2} + \tfrac{1}{1+\xi}\tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t} + \tfrac{\mu_0}{\tau_y}(\psi^{S}_{j+1/2} - \psi^{S}_{j-1/2})\right]} \tag{6.50}$$

Insert these equations into the N and S balance equations:

$$\begin{aligned}\tfrac{\mu_0}{\tau_y}\left(\psi^N_{j+1/2} - \psi^N_{j-1/2}\right) + \psi^N_j\left(1+\tfrac{\mu_0}{\tau_x}\right)\left(\tfrac{d}{d-3}\right) &= \tfrac{Q_j}{2\sigma_t} \\ &+ \tfrac{1+\tau_x/\mu_0}{d-3}\tfrac{\mu_0}{\tau_y}\left(-\tfrac{1}{1+\xi}2\pi\psi^{x,NE}_{j-1/2} + \tfrac{1}{1+\xi}\tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t} - \tfrac{\mu_0}{\tau_y}(\psi^N_{j+1/2} - \psi^N_{j-1/2})\right]\right)\ ,\end{aligned} \tag{6.51}$$

$$\begin{aligned}-\tfrac{\mu_0}{\tau_y}\left(\psi^S_{j+1/2} - \psi^S_{j-1/2}\right) + \psi^S_j\left(1+\tfrac{\mu_0}{\tau_x}\right)\left(\tfrac{d}{d-3}\right) &= \tfrac{Q_j}{2\sigma_t} \\ &+ \tfrac{1+\tau_x/\mu_0}{d-3}\tfrac{\mu_0}{\tau_y}\left(-\tfrac{1}{1+\xi}2\pi\psi^{x,SE}_{j+1/2} + \tfrac{1}{1+\xi}\tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t} + \tfrac{\mu_0}{\tau_y}(\psi^S_{j+1/2} - \psi^S_{j-1/2})\right]\right)\ ,\end{aligned} \tag{6.52}$$

or

$$\begin{aligned}\tfrac{\mu_0}{\tau_y}\left(\psi^N_{j+1/2} - \psi^N_{j-1/2}\right)\left(1 + \tfrac{1+\tau_x/\mu_0}{d-3}\tfrac{\mu_0}{\tau_y}\tfrac{1}{1+\xi}\tfrac{3\tau_x}{\mu_0 d}\right) + \psi^N_j\left(1+\tfrac{\mu_0}{\tau_x}\right)\left(\tfrac{d}{d-3}\right) &= \tfrac{Q_j}{2\sigma_t}\left(1 + \tfrac{1+\tau_x/\mu_0}{d-3}\tfrac{\mu_0}{\tau_y}\tfrac{1}{1+\xi}\tfrac{3\tau_x}{\mu_0 d}\right) \\ &- \tfrac{1+\tau_x/\mu_0}{d-3}\tfrac{\mu_0}{\tau_y}\tfrac{1}{1+\xi}2\pi\psi^{x,NE}_{j-1/2}\ ,\end{aligned} \tag{6.53}$$

$$\begin{aligned}-\tfrac{\mu_0}{\tau_y}\left(\psi^S_{j+1/2} - \psi^S_{j-1/2}\right)\left(1 + \tfrac{1+\tau_x/\mu_0}{d-3}\tfrac{\mu_0}{\tau_y}\tfrac{1}{1+\xi}\tfrac{3\tau_x}{\mu_0 d}\right) + \psi^S_j\left(1+\tfrac{\mu_0}{\tau_x}\right)\left(\tfrac{d}{d-3}\right) &= \tfrac{Q_j}{2\sigma_t}\left(1 + \tfrac{1+\tau_x/\mu_0}{d-3}\tfrac{\mu_0}{\tau_y}\tfrac{1}{1+\xi}\tfrac{3\tau_x}{\mu_0 d}\right) \\ &- \tfrac{1+\tau_x/\mu_0}{d-3}\tfrac{\mu_0}{\tau_y}\tfrac{1}{1+\xi}2\pi\psi^{x,SE}_{j+1/2}\ ,\end{aligned} \tag{6.54}$$

We will multiply through by the factor:

$$\frac{1}{1+\frac{1+\tau_x/\mu_0}{d-3}\frac{\mu_0}{\tau_y}\frac{1}{1+\xi}\frac{3\tau_x}{\mu_0 d}} = \frac{d(1+\xi)\frac{\tau_y}{\tau_x}}{d(1+\xi)\frac{\tau_y}{\tau_x}+\left(1+\frac{\tau_x}{\mu_0}\right)\frac{3}{d-3}} \tag{6.55}$$

and define

$$R_x \equiv \left(1+\frac{\mu_0}{\tau_x}\right)\left(\frac{d}{d-3}\right)\frac{d(1+\xi)\frac{\tau_y}{\tau_x}}{d(1+\xi)\frac{\tau_y}{\tau_x}+\left(1+\frac{\tau_x}{\mu_0}\right)\frac{3}{d-3}} - 1\ , \tag{6.56}$$

$$\Lambda \equiv \frac{d(1+\xi)\frac{\tau_y}{\tau_x}}{d(1+\xi)\frac{\tau_y}{\tau_x}+\left(1+\frac{\tau_x}{\mu_0}\right)\frac{3}{d-3}}\frac{1+\frac{\tau_x}{\mu_0}}{d-3}\frac{\mu_0}{\tau_y}\frac{1}{1+\xi}$$

$$= \frac{\mu_0+\tau_x}{d(1+\xi)\tau_y+\left(1+\frac{\tau_x}{\mu_0}\right)\frac{3}{d-3}\tau_x}\frac{d}{d-3}\ . \tag{6.57}$$

Then our North and South equations become:

$$\tfrac{\mu_0}{\tau_y}\left(\psi^N_{j+1/2}-\psi^N_{j-1/2}\right)+\psi^N_j\left(1+R_x\right) = \tfrac{Q_j}{2\sigma_t} - \Lambda 2\pi\psi^{x,NE}_{j-1/2}\ , \tag{6.58}$$

$$-\tfrac{\mu_0}{\tau_y}\left(\psi^S_{j+1/2}-\psi^S_{j-1/2}\right)+\psi^S_j\left(1+R_x\right) = \tfrac{Q_j}{2\sigma_t} - \Lambda 2\pi\psi^{x,SE}_{j+1/2}\ , \tag{6.59}$$

$$\tfrac{\mu_0}{\tau_y}\left(\psi^N_{j+1/2}+\psi^N_{j-1/2}-2\psi^N_j\right)+\tfrac{1}{3}\psi^{y,N}_j\left(1+\tfrac{\mu_0}{\tau_x}\right) = \tfrac{1}{3}\tfrac{Q^y_j}{2\sigma_t}\ , \tag{6.60}$$

$$-\tfrac{\mu_0}{\tau_y}\left(\psi^S_{j+1/2}+\psi^S_{j-1/2}-2\psi^S_j\right)+\tfrac{1}{3}\psi^{y,S}_j\left(1+\tfrac{\mu_0}{\tau_x}\right) = \tfrac{1}{3}\tfrac{Q^y_j}{2\sigma_t} \tag{6.61}$$

$$2\pi(1+\xi)\left(\psi^{x,NE}_{j+1/2}-\psi^{x,NE}_{j-1/2}\right)+2\pi\psi^{x,NE}_{j+1/2} = \tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t}-\tfrac{\mu_0}{\tau_y}(\psi^N_{j+1/2}-\psi^N_{j-1/2})\right]\ , \tag{6.62}$$

$$-2\pi(1+\xi)\left(\psi^{x,SE}_{j+1/2}-\psi^{x,SE}_{j-1/2}\right)+2\pi\psi^{x,SE}_{j-1/2} = \tfrac{3\tau_x}{\mu_0 d}\left[\tfrac{Q_j}{2\sigma_t}+\tfrac{\mu_0}{\tau_y}(\psi^S_{j+1/2}-\psi^S_{j-1/2})\right] \tag{6.63}$$

At this point we have created what looks like a 1D problem in y, with the exception of the ψ^x

terms. We did this by solving for the x moment of the angular flux and modifying the average and y-moment equations to account for leakage out of the x surfaces.

At this point our only "approximation" has been to use the S_2 quadrature set to evaluate the x-leakage correction terms.

6.3 Ansatz

We rewrite our equations in terms of point values at edges instead of averages and slopes, because we believe we can more readily write down an ansatz for those variables that matches the observed behaviors. We continue to use North and South for direction of travel. We now introduce Top and Bottom to denote cell edges. We have:

$$\psi_j^{N/S} \equiv \tfrac{1}{2}(\psi_{jT}^{N/S} + \psi_{jB}^{N/S}) \tag{6.64}$$

$$\psi_j^{y,N/S} \equiv \tfrac{1}{2}(\psi_{jT}^{N/S} - \psi_{jB}^{N/S}) \tag{6.65}$$

$$\psi_{j+1/2}^{N} \equiv \psi_{jT}^{N} \tag{6.66}$$

$$\psi_{j-1/2}^{S} \equiv \psi_{jB}^{S} \tag{6.67}$$

We introduce the ansatz:

$$\begin{aligned} \psi_{j,T}^{N} &= A\left[e^{i\lambda(y_{j+1/2}+b/2)} - e^{-i\lambda(y_{j+1/2}+b/2)}\right] \\ &\equiv iD\left[e^{i\lambda(y_{j+1/2}+b/2)} - e^{-i\lambda(y_{j+1/2}+b/2)}\right] \end{aligned} \tag{6.68}$$

$$\psi_{j,B}^{N} = Ce^{i\lambda(y_{j-1/2}+b/2)} + \overline{C}e^{-i\lambda(y_{j-1/2}+b/2)} \tag{6.69}$$

$$\psi_{j,B}^{S} = iD\left[e^{-i\lambda(y_{j-1/2}-b/2)} - e^{i\lambda(y_{j-1/2}-b/2)}\right] \tag{6.70}$$

$$\psi_{j,T}^{S} = Ce^{-i\lambda(y_{j+1/2}-b/2)} + \overline{C}e^{i\lambda(y_{j+1/2}-b/2)} \tag{6.71}$$

$$2\pi\psi_{j+1/2}^{x,NE} = F\left[e^{i\lambda(y_{j+1/2}+b/2)} - e^{-i\lambda(y_{j+1/2}+b/2)}\right] \tag{6.72}$$

$$2\pi\psi_{j-1/2}^{x,SE} = F\left[e^{-i\lambda(y_{j-1/2}-b/2)} - e^{i\lambda(y_{j-1/2}-b/2)}\right] \tag{6.73}$$

Here we have recognized that since all angular fluxes must be real, the constant A must be purely imaginary (since it multiplies a purely imaginary number to produce a real number), and for convenience we have defined A to equal iD where D is real. Again, since all angular fluxes are real, the constant in front of the exponential in the second term of the expressions for $\psi^N_{j,B}$ and $\psi^S_{j,T}$ must be $\overline{C}$, which is the complex conjugate of C. Note that we have used the same constants (A, C, F) for the South angular fluxes as for the North ones. Examination of the ansatz shows that we are simply imposing symmetry, so that a North angular flux a certain distance above the center of the problem is the same as the South flux the same distance below center.

This ansatz holds both in the interior of the problem and at the boundaries—the constants in the ansatz have been chosen so that the expressions satisfy the vacuum boundary conditions.

6.3.1 Scalar Fluxes and Net Current Densities

In our eigenvalue problem the source is from fission, which depends only on the scalar flux, not on each angular flux individually. The eigenvector will therefore be determined by two quantities per cell, which we can represent either as $\{\phi_{jT}, \phi_{jB}\}$ or $\{\phi_j, \phi^y_j\}$. Let us see what the ansatz says about these quantities:

$$\begin{aligned}\phi_{jT} &\equiv \psi^N_{j,T} + \psi^S_{j,T} \\ &= iD\left[e^{i\lambda(y_{j+1/2}+b/2)} - e^{-i\lambda(y_{j+1/2}+b/2)}\right] + Ce^{-i\lambda(y_{j+1/2}-b/2)} + \overline{C}e^{i\lambda(y_{j+1/2}-b/2)} \\ &= e^{i\lambda y_{j+1/2}}\left[iDe^{i\lambda b/2} + \overline{C}e^{-i\lambda b/2}\right] + e^{-i\lambda y_{j+1/2}}\left[-iDe^{-i\lambda b/2} + Ce^{i\lambda b/2}\right] \\ &= e^{i\lambda y_j}\left[iDe^{i\lambda b/2}e^{i\theta/2} + \overline{C}e^{-i\lambda b/2}e^{i\theta/2}\right] + e^{-i\lambda y_j}\left[-iDe^{-i\lambda b/2}e^{-i\theta/2} + Ce^{i\lambda b/2}e^{-i\theta/2}\right] \ ,\end{aligned} \tag{6.74}$$

where

$$\theta \equiv \lambda\Delta y \ . \tag{6.75}$$

Define

$$G \equiv iDe^{i\lambda b/2}e^{i\theta/2} \tag{6.76}$$

$$H \equiv \overline{C}e^{-i\lambda b/2}e^{i\theta/2} \tag{6.77}$$

This yields

$$\phi_{jT} = (G+H)e^{i\lambda y_j} + (\overline{G}+\overline{H})e^{-i\lambda y_j}\ . \tag{6.78}$$

Similarly,

$$\begin{aligned}
\phi_{jB} &\equiv \psi^S_{j,B} + \psi^N_{j,B} \\
&= iD\left[e^{-i\lambda(y_{j-1/2}-b/2)} - e^{i\lambda(y_{j-1/2}-b/2)}\right] + Ce^{i\lambda(y_{j-1/2}+b/2)} + \overline{C}e^{-i\lambda(y_{j-1/2}+b/2)} \\
&= e^{i\lambda y_{j-1/2}}\left[-iDe^{-i\lambda b/2} + Ce^{i\lambda b/2}\right] + e^{-i\lambda y_{j-1/2}}\left[iDe^{i\lambda b/2} + \overline{C}e^{-i\lambda b/2}\right] \\
&= e^{i\lambda y_j}\left[-iDe^{-i\lambda b/2}e^{-i\theta/2} + Ce^{i\lambda b/2}e^{-i\theta/2}\right] + e^{-i\lambda y_j}\left[iDe^{i\lambda b/2}e^{i\theta/2} + \overline{C}e^{-i\lambda b/2}e^{i\theta/2}\right] \\
&= (\overline{G}+\overline{H})e^{i\lambda y_j} + (G+H)e^{-i\lambda y_j}
\end{aligned} \tag{6.79}$$

It follows that

$$\begin{aligned}
\phi_j &\equiv \tfrac{1}{2}\left(\phi_{jT} + \phi_{jB}\right) \\
&= \tfrac{1}{2}\left((G+H)e^{i\lambda y_j} + (\overline{G}+\overline{H})e^{-i\lambda y_j} + (\overline{G}+\overline{H})e^{i\lambda y_j} + (G+H)e^{-i\lambda y_j}\right) \\
&= \tfrac{1}{2}\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\left((G+H) + (\overline{G}+\overline{H})\right) && (6.80) \\
&= \left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\left(\Re\left[G\right] + \Re\left[H\right]\right)\ , && (6.81)
\end{aligned}$$

where $\Re[z] \equiv$ real part of the complex number z, and

$$\begin{aligned}
\phi^y_j &\equiv \tfrac{1}{2}\left(\phi_{jT} - \phi_{jB}\right) \\
&= \tfrac{1}{2}\left((G+H)e^{i\lambda y_j} + (\overline{G}+\overline{H})e^{-i\lambda y_j} - (\overline{G}+\overline{H})e^{i\lambda y_j} + (G+H)e^{-i\lambda y_j}\right) \\
&= \tfrac{1}{2}\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left((G+H) - (\overline{G}+\overline{H})\right) \qquad (6.82) \\
&= \left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(\Im\left[G\right] + \Im\left[H\right]\right) \ , \qquad (6.83)
\end{aligned}$$

where $\Im[z] \equiv$ imaginary part of the complex number z. Collect ϕ expressions:

$$\boxed{\begin{aligned}
\phi_{jT} &= (G+H)e^{i\lambda y_j} + (\overline{G}+\overline{H})e^{-i\lambda y_j} \ , \qquad (6.84) \\
\phi_{jB} &= (\overline{G}+\overline{H})e^{i\lambda y_j} + (G+H)e^{-i\lambda y_j} \ , \qquad (6.85) \\
\phi_j &= \left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\Re\left[G+H\right] \ , \qquad (6.86) \\
\phi^y_j &= \left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right) i\Im\left[G+H\right] \ . \qquad (6.87)
\end{aligned}}$$

It will be convenient to have similar expressions for the net current densities:

$$\begin{aligned}
J_{jT} &\equiv \mu_0\left(\psi^N_{j,T} - \psi^S_{j,T}\right) \\
\Rightarrow \tfrac{1}{\mu_0}J_{jT} &= iD\left[e^{i\lambda(y_{j+1/2}+b/2)} - e^{-i\lambda(y_{j+1/2}+b/2)}\right] - Ce^{-i\lambda(y_{j+1/2}-b/2)} - \overline{C}e^{i\lambda(y_{j+1/2}-b/2)} \\
&= e^{i\lambda y_j}e^{i\theta/2}\left[iDe^{i\lambda b/2} - \overline{C}e^{-i\lambda b/2}\right] + e^{-i\lambda y_j}e^{-i\theta/2}\left[-iDe^{-i\lambda b/2} - Ce^{i\lambda b/2}\right] \qquad (6.88)
\end{aligned}$$

or

$$J_{jT} = \mu_0\left((G-H)e^{i\lambda y_j} + (\overline{G}-\overline{H})e^{-i\lambda y_j}\right) \ . \qquad (6.89)$$

Similarly,

$$
\begin{aligned}
J_{jB} &\equiv \mu_0\left(\psi^N_{j,B} - \psi^S_{j,B}\right) \\
\Rightarrow \tfrac{1}{\mu_0} J_{jB} &= Ce^{i\lambda(y_{j-1/2}+b/2)} + \overline{C}e^{-i\lambda(y_{j-1/2}+b/2)} - iD\left[e^{-i\lambda(y_{j-1/2}-b/2)} - e^{i\lambda(y_{j-1/2}-b/2)}\right] \\
&= e^{i\lambda y_j}e^{-i\theta/2}\left[iDe^{-i\lambda b/2} + Ce^{i\lambda b/2}\right] + e^{-i\lambda y_j}e^{i\theta/2}\left[-iDe^{i\lambda b/2} + \overline{C}e^{-i\lambda b/2}\right] \\
&= e^{i\lambda y_j}\left[-(\overline{G} - \overline{H})\ \right] + e^{-i\lambda y_j}\left[-(G-H)\right] \\
&= -\left((\overline{G} - \overline{H})e^{i\lambda y_j} + (G-H)e^{-i\lambda y_j}\right) \qquad (6.90)
\end{aligned}
$$

It follows that

$$
\begin{aligned}
J_j &\equiv \tfrac{1}{2}\left(J_{jT} + J_{jB}\right) \\
&= \tfrac{\mu_0}{2}\left((G-H)e^{i\lambda y_j} + (\overline{G} - \overline{H})e^{-i\lambda y_j} - (\overline{G} - \overline{H})e^{i\lambda y_j} - (G-H)e^{-i\lambda y_j}\right) \\
&= \tfrac{\mu_0}{2}\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left((G-H) - (\overline{G} - \overline{H})\right) \\
&= \mu_0\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right) i\Im\left[G-H\right]\ , \qquad (6.91)
\end{aligned}
$$

and

$$
\begin{aligned}
J^y_j &\equiv \tfrac{1}{2}\left(J_{jT} - J_{jB}\right) \\
&= \tfrac{\mu_0}{2}\left((G-H)e^{i\lambda y_j} + (\overline{G} - \overline{H})e^{-i\lambda y_j} + (\overline{G} - \overline{H})e^{i\lambda y_j} + (G-H)e^{-i\lambda y_j}\right) \\
&= \tfrac{\mu_0}{2}\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\left((G-H) + (\overline{G} - \overline{H})\right) \\
&= \mu_0\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\Re\left[G-H\right]\ . \qquad (6.92)
\end{aligned}
$$

Collect J expressions:

$$J_{jT} = \mu_0 \left((G - H) e^{i\lambda y_j} + (\overline{G} - \overline{H}) e^{-i\lambda y_j} \right) \ , \tag{6.93}$$

$$J_{jB} = -\mu_0 \left((\overline{G} - \overline{H}) e^{i\lambda y_j} + (G - H) e^{-i\lambda y_j} \right) \ , \tag{6.94}$$

$$J_j = \mu_0 \left(e^{i\lambda y_j} - e^{-i\lambda y_j} \right) i \Im \left[G - H \right] \ , \tag{6.95}$$

$$J_j^y = \mu_0 \left(e^{i\lambda y_j} + e^{-i\lambda y_j} \right) \Re \left[G - H \right] \ . \tag{6.96}$$

6.3.2 Cell-Edge Quantities

Other terms will arise on cell edges as we proceed with our analysis. Let us study them:

$$\begin{aligned} J_{j+1/2} &\equiv \mu_0 \left(\psi^N_{j+1/2} - \psi^S_{j+1/2} \right) \equiv \mu_0 \left(\psi^N_{jT} - \psi^S_{j+1,B} \right) \\ \Rightarrow \tfrac{1}{\mu_0} J_{j+1/2} &= iD \left[e^{i\lambda(y_{j+1/2}+b/2)} - e^{-i\lambda(y_{j+1/2}+b/2)} \right] - iD \left[e^{-i\lambda(y_{j+1/2}-b/2)} - e^{i\lambda(y_{j+1/2}-b/2)} \right] \\ \tfrac{1}{\mu_0} J_{j+1/2} &= iD e^{i\lambda y_j} e^{i\theta/2} \left[e^{i\lambda b/2} + e^{-i\lambda b/2} \right] - iD e^{-i\lambda y_j} e^{-i\theta/2} \left[e^{-i\lambda b/2} + e^{i\lambda b/2} \right] \\ J_{j+1/2} &= \mu_0 iD \left(e^{i\lambda y_j} e^{i\theta/2} - e^{-i\lambda y_j} e^{-i\theta/2} \right) \left(e^{i\lambda b/2} + e^{-i\lambda b/2} \right) \ . \end{aligned} \tag{6.97}$$

Similarly,

$$\begin{aligned} J_{j-1/2} &\equiv \mu_0 \left(\psi^N_{j-1/2} - \psi^S_{j-1/2} \right) \equiv \mu_0 \left(\psi^N_{j-1,T} - \psi^S_{j,B} \right) \\ \Rightarrow \tfrac{1}{\mu_0} J_{j-1/2} &= iD \left[e^{i\lambda(y_{j-1/2}+b/2)} - e^{-i\lambda(y_{j-1/2}+b/2)} \right] - iD \left[e^{-i\lambda(y_{j-1/2}-b/2)} - e^{i\lambda(y_{j-1/2}-b/2)} \right] \\ \tfrac{1}{\mu_0} J_{j-1/2} &= iD e^{i\lambda y_j} e^{-i\theta/2} \left[e^{i\lambda b/2} + e^{-i\lambda b/2} \right] - iD e^{-i\lambda y_j} e^{i\theta/2} \left[e^{-i\lambda b/2} + e^{i\lambda b/2} \right] \\ J_{j-1/2} &= \mu_0 iD \left(e^{i\lambda y_j} e^{-i\theta/2} - e^{-i\lambda y_j} e^{i\theta/2} \right) \left(e^{i\lambda b/2} + e^{-i\lambda b/2} \right) \ . \end{aligned} \tag{6.98}$$

Continue with other edge terms:

$$\begin{aligned}
\phi_{j+1/2} &\equiv \left(\psi^N_{j+1/2} + \psi^S_{j+1/2}\right) \equiv \left(\psi^N_{jT} + \psi^S_{j+1,B}\right) \\
&= iD\left[e^{i\lambda(y_{j+1/2}+b/2)} - e^{-i\lambda(y_{j+1/2}+b/2)}\right] + iD\left[e^{-i\lambda(y_{j+1/2}-b/2)} - e^{i\lambda(y_{j+1/2}-b/2)}\right] \\
&= iDe^{i\lambda y_j}e^{i\theta/2}\left[e^{i\lambda b/2} - e^{-i\lambda b/2}\right] + iDe^{-i\lambda y_j}e^{-i\theta/2}\left[e^{i\lambda b/2} - e^{-i\lambda b/2}\right] \\
&= iD\left(e^{i\lambda y_j}e^{i\theta/2} + e^{-i\lambda y_j}e^{-i\theta/2}\right)\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right) \ . \qquad (6.99)
\end{aligned}$$

and

$$\begin{aligned}
\phi_{j-1/2} &\equiv \left(\psi^N_{j-1/2} + \psi^S_{j-1/2}\right) \equiv \left(\psi^N_{j-1,T} + \psi^S_{j,B}\right) \\
&= iD\left[e^{i\lambda(y_{j-1/2}+b/2)} - e^{-i\lambda(y_{j-1/2}+b/2)}\right] + iD\left[e^{-i\lambda(y_{j-1/2}-b/2)} - e^{i\lambda(y_{j-1/2}-b/2)}\right] \\
&= iDe^{i\lambda y_j}e^{-i\theta/2}\left[e^{i\lambda b/2} - e^{-i\lambda b/2}\right] + iDe^{-i\lambda y_j}e^{i\theta/2}\left[e^{i\lambda b/2} - e^{-i\lambda b/2}\right] \\
&= iD\left(e^{i\lambda y_j}e^{-i\theta/2} + e^{-i\lambda y_j}e^{i\theta/2}\right)\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right) \ . \qquad (6.100)
\end{aligned}$$

These terms appear as sums and differences in the equations we will be using:

$$\begin{aligned}
J_{j+1/2} - J_{j-1/2} &= \mu_0 iD\left(e^{i\lambda y_j}e^{i\theta/2} - e^{-i\lambda y_j}e^{-i\theta/2}\right)\left(e^{i\lambda b/2} + e^{-i\lambda b/2}\right) \\
&\quad - \mu_0 iD\left(e^{i\lambda y_j}e^{-i\theta/2} - e^{-i\lambda y_j}e^{i\theta/2}\right)\left(e^{i\lambda b/2} + e^{-i\lambda b/2}\right) \\
&= \mu_0 iD\left(e^{i\lambda b/2} + e^{-i\lambda b/2}\right)\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\left(e^{i\theta/2} - e^{-i\theta/2}\right) \ , \qquad (6.101) \\
J_{j+1/2} + J_{j-1/2} &= \mu_0 iD\left(e^{i\lambda y_j}e^{i\theta/2} - e^{-i\lambda y_j}e^{-i\theta/2}\right)\left(e^{i\lambda b/2} + e^{-i\lambda b/2}\right) \\
&\quad + \mu_0 iD\left(e^{i\lambda y_j}e^{-i\theta/2} - e^{-i\lambda y_j}e^{i\theta/2}\right)\left(e^{i\lambda b/2} + e^{-i\lambda b/2}\right) \\
&= \mu_0 iD\left(e^{i\lambda b/2} + e^{-i\lambda b/2}\right)\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(e^{i\theta/2} + e^{-i\theta/2}\right) \ . \qquad (6.102)
\end{aligned}$$

and

$$\begin{aligned}
\phi_{j+1/2} - \phi_{j-1/2} &= iD\left(e^{i\lambda y_j}e^{i\theta/2} + e^{-i\lambda y_j}e^{-i\theta/2}\right)\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right) \\
&\quad - iD\left(e^{i\lambda y_j}e^{-i\theta/2} + e^{-i\lambda y_j}e^{i\theta/2}\right)\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right) \\
&= iD\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right)\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(e^{i\theta/2} - e^{-i\theta/2}\right)\ , \qquad (6.103) \\
\phi_{j+1/2} + \phi_{j-1/2} &= iD\left(e^{i\lambda y_j}e^{i\theta/2} + e^{-i\lambda y_j}e^{-i\theta/2}\right)\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right) \\
&\quad + iD\left(e^{i\lambda y_j}e^{-i\theta/2} + e^{-i\lambda y_j}e^{i\theta/2}\right)\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right) \\
&= iD\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right)\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\left(e^{i\theta/2} + e^{-i\theta/2}\right)\ . \qquad (6.104)
\end{aligned}$$

There are also x-slope terms on cell edges:

$$\begin{aligned}
2\pi\left(\psi^{x,NE}_{j+1/2} + \psi^{x,SE}_{j+1/2}\right) &= F\left[e^{i\lambda(y_{j+1/2}+b/2)} - e^{-i\lambda(y_{j+1/2}+b/2)} + e^{-i\lambda(y_{j+1/2}-b/2)} - e^{i\lambda(y_{j+1/2}-b/2)}\right] \qquad (6.105) \\
&= F\left(e^{i\lambda y_j}e^{i\theta/2}(e^{i\lambda b/2} - e^{-i\lambda b/2}) - e^{-i\lambda y_j}e^{-i\theta/2}(e^{-i\lambda b/2} - e^{i\lambda b/2})\right) \\
&= F\left(e^{i\lambda y_j}e^{i\theta/2} + e^{-i\lambda y_j}e^{-i\theta/2}\right)\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right) \qquad (6.106)
\end{aligned}$$

and

$$\begin{aligned}
2\pi\left(\psi^{x,NE}_{j-1/2} + \psi^{x,SE}_{j-1/2}\right) &= F\left[e^{i\lambda(y_{j-1/2}+b/2)} - e^{-i\lambda(y_{j-1/2}+b/2)} + e^{-i\lambda(y_{j-1/2}-b/2)} - e^{i\lambda(y_{j-1/2}-b/2)}\right] \qquad (6.107) \\
&= F\left(e^{i\lambda y_j}e^{-i\theta/2}(e^{i\lambda b/2} - e^{-i\lambda b/2}) - e^{-i\lambda y_j}e^{i\theta/2}(e^{-i\lambda b/2} - e^{i\lambda b/2})\right) \\
&= F\left(e^{i\lambda y_j}e^{-i\theta/2} + e^{-i\lambda y_j}e^{i\theta/2}\right)\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right)\ . \qquad (6.108)
\end{aligned}$$

So:

$$\begin{aligned}2\pi\left(\psi^{x,NE}_{j+1/2}+\psi^{x,SE}_{j+1/2}\right)-2\pi\left(\psi^{x,NE}_{j-1/2}+\psi^{x,SE}_{j-1/2}\right)&\\ &=F\left(e^{i\lambda y_j}e^{i\theta/2}+e^{-i\lambda y_j}e^{-i\lambda/2}\right)\left(e^{i\lambda b/2}-e^{-i\lambda b/2}\right)\\ &\quad-F\left(e^{i\lambda y_j}e^{-i\theta/2}+e^{-i\lambda y_j}e^{i\theta/2}\right)\left(e^{i\lambda b/2}-e^{-i\lambda b/2}\right)\\ &=F\left(e^{i\lambda y_j}-e^{-i\lambda y_j}\right)\left(e^{i\theta/2}-e^{-i\theta/2}\right)\left(e^{i\lambda b/2}-e^{-i\lambda b/2}\right)\end{aligned}\tag{6.109}$$

Also:

$$2\pi\left(\psi^{x,NE}_{j+1/2}-\psi^{x,SE}_{j-1/2}\right)=F\left[e^{i\lambda(y_{j+1/2}+b/2)}-e^{-i\lambda(y_{j+1/2}+b/2)}-e^{-i\lambda(y_{j-1/2}-b/2)}+e^{i\lambda(y_{j-1/2}-b/2)}\right]\tag{6.110}$$

$$\begin{aligned}&=F\left(e^{i\lambda y_j}\left(e^{i(\lambda b+\theta)/2}+e^{-i(\lambda b+\theta)/2}\right)-e^{-i\lambda y_j}\left(e^{-i(\lambda b+\theta)/2}+e^{i(\lambda b+\theta)/2}\right)\right)\\ &=F\left(e^{i\lambda y_j}-e^{-i\lambda y_j}\right)\left(e^{i(\lambda b+\theta)/2}+e^{-i(\lambda b+\theta)/2}\right)\end{aligned}\tag{6.111}$$

Define the constants:

$$S_b\equiv\sin(\lambda b/2),\tag{6.112}$$

$$C_b\equiv\cos(\lambda b/2),\tag{6.113}$$

$$S_\theta\equiv\sin(\theta/2)=\sin(\lambda\Delta y/2),\tag{6.114}$$

$$C_\theta\equiv\cos(\theta/2)=\cos(\lambda\Delta y/2).\tag{6.115}$$

Collect edge-term expressions:

$$
\begin{aligned}
J_{j+1/2} - J_{j-1/2} &= \mu_0 iD \left(e^{i\lambda b/2} + e^{-i\lambda b/2}\right)\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\left(e^{i\theta/2} - e^{-i\theta/2}\right) \\
&= -4\mu_0 DC_b S_\theta \left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right) \ , \qquad (6.116) \\
J_{j+1/2} + J_{j-1/2} &= \mu_0 iD \left(e^{i\lambda b/2} + e^{-i\lambda b/2}\right)\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(e^{i\theta/2} + e^{-i\theta/2}\right) \\
&= 4\mu_0 iDC_b C_\theta \left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right) \ , \qquad (6.117) \\
\phi_{j+1/2} - \phi_{j-1/2} &= iD \left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right)\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(e^{i\theta/2} - e^{-i\theta/2}\right) \\
&= -4iDS_b S_\theta \left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right) \ , \qquad (6.118) \\
\phi_{j+1/2} + \phi_{j-1/2} &= iD \left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right)\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\left(e^{i\theta/2} + e^{-i\theta/2}\right) \\
&= -4DS_b C_\theta \left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right) \ , \qquad (6.119) \\
2\pi\left(\psi^{x,NE}_{j+1/2} + \psi^{x,SE}_{j+1/2}\right) &- 2\pi\left(\psi^{x,NE}_{j-1/2} + \psi^{x,SE}_{j-1/2}\right) \\
&= F\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(e^{i\theta/2} - e^{-i\theta/2}\right)\left(e^{i\lambda b/2} - e^{-i\lambda b/2}\right) \\
&= -4FS_b S_\theta \left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right) \ , \qquad (6.120) \\
2\pi\left(\psi^{x,NE}_{j+1/2} - \psi^{x,SE}_{j-1/2}\right) &= F\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(e^{i(\lambda b+\theta)/2} + e^{-i(\lambda b+\theta)/2}\right) \\
&= 2F\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(C_b C_\theta - S_b S_\theta\right) \ . \qquad (6.121)
\end{aligned}
$$

6.3.3 X-Slope Terms

When we add the N and S balance equations we obtain a sum of x-slope terms. Simplify the sum:

$$
\begin{aligned}
2\pi\psi^{x,NE}_{j-1/2} + 2\pi\psi^{x,SE}_{j+1/2} &= F\left[e^{i\lambda(y_{j-1/2}+b/2)} - e^{-i\lambda(y_{j-1/2}+b/2)}\right] + F\left[e^{-i\lambda(y_{j+1/2}-b/2)} - e^{i\lambda(y_{j+1/2}-b/2)}\right] \\
&= Fe^{i\lambda y_j}\left[e^{-i\theta/2}e^{i\lambda b/2} - e^{i\theta/2}e^{-i\lambda b/2}\right] + Fe^{-i\lambda y_j}\left[-e^{i\theta/2}e^{-i\lambda b/2} + e^{-i\theta/2}e^{i\lambda b/2}\right] \\
&= F\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\left(e^{i\lambda(b-\Delta y)/2} - e^{-i\lambda(b-\Delta y)/2}\right) \\
&= 2iF\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)\left(S_b C_\theta - C_b S_\theta\right) \ . \qquad (6.122)
\end{aligned}
$$

When we subtract the N and S balance equations we obtain a difference of x-slope terms. Simplify the difference:

$$\begin{aligned}
2\pi\psi^{x,NE}_{j-1/2} - 2\pi\psi^{x,SE}_{j+1/2} &= F\left[e^{i\lambda(y_{j-1/2}+b/2)} - e^{-i\lambda(y_{j-1/2}+b/2)}\right] - F\left[e^{-i\lambda(y_{j+1/2}-b/2)} - e^{i\lambda(y_{j+1/2}-b/2)}\right] \\
&= Fe^{i\lambda y_j}\left[e^{-i\theta/2}e^{i\lambda b/2} + e^{i\theta/2}e^{-i\lambda b/2}\right] + Fe^{-i\lambda y_j}\left[-e^{i\theta/2}e^{-i\lambda b/2} - e^{-i\theta/2}e^{i\lambda b/2}\right] \\
&= F\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(e^{i\lambda(b-\Delta y)/2} + e^{-i\lambda(b-\Delta y)/2}\right) \\
&= 2F\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)\left(C_bC_\theta + S_bS_\theta\right) \ . \qquad (6.123)
\end{aligned}$$

6.3.4 Zeroth and First Angular Moments

Now we begin to manipulate our equations and plug in our ansatz. First we take the zeroeth moment of our cell balance equations by adding Eq. 6.58 and 6.59:

$$\tfrac{1}{\tau_y}\left(J_{j+1/2} - J_{j-1/2}\right) + \left(1 + R_x\right)\phi_j = \left(1 + R_x + R_y\right)\phi_j - \Lambda 2\pi\left(\psi^{x,NE}_{j-1/2} + \psi^{x,SE}_{j+1/2}\right) \qquad (6.124)$$

or

$$\tfrac{1}{\tau_y}\left(J_{j+1/2} - J_{j-1/2}\right) = R_y\phi_j - \Lambda 2\pi\left(\psi^{x,NE}_{j-1/2} + \psi^{x,SE}_{j+1/2}\right) \qquad (6.125)$$

Here we have used the fact that:

$$\begin{aligned}
\frac{\nu\sigma_f}{k} + \sigma_s &= \left(\frac{\nu\sigma_f}{k} - \sigma_a\right) + \sigma_t \\
&= \sigma_t\left(R_x + R_y\right) + \sigma_t \\
&= \sigma_t\left(1 + R_x + R_y\right) \qquad (6.126)
\end{aligned}$$

Inserting results from the previous section and cancelling out $\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)$ from every term gives:

$$-4\tfrac{\mu_0}{\tau_y}DC_bS_\theta = R_y\Re\left[G+H\right] - \Lambda F2i\left(S_bC_\theta - C_bS_\theta\right)\ . \tag{6.127}$$

Now we take the first moment of the cell balance equations by subtracting Eq. (6.58) − (6.59), after multiplying by μ_0:

$$\tfrac{1}{3\tau_y}\left(\phi_{j+1/2} - \phi_{j-1/2}\right) + \left(1+R_x\right)J_j = -\mu_0\Lambda 2\pi\left(\psi^{x,NE}_{j-1/2} - \psi^{x,SE}_{j+1/2}\right)\ . \tag{6.128}$$

Inserting results from the previous section and cancelling out $\left(e^{i\lambda y_j} - e^{-i\lambda y_j}\right)$ from every term gives:

$$-4\tfrac{1}{3\tau_y}iDS_bS_\theta + \mu_0\left(1+R_x\right)i\,\Im\left[G-H\right] = -\mu_0\Lambda F2\left(C_bC_\theta + S_bS_\theta\right)\ . \tag{6.129}$$

Now we take the zeroth moment of the cell y-slope equations by adding Eqs. (6.60) and (6.61):

$$\tfrac{1}{\tau_y}\left(J_{j+1/2} + J_{j-1/2} - 2J_j\right) + \tfrac{1}{3}\left(1+\tfrac{\mu_0}{\tau_x}\right)\phi^y_j = \tfrac{1}{3}(1+R_x+R_y)\phi^y_j \tag{6.130}$$

or

$$\tfrac{1}{\tau_y}\left(J_{j+1/2} + J_{j-1/2} - 2J_j\right) = \tfrac{1}{3}(\rho+R_y)\phi^y_j\ , \tag{6.131}$$

where

$$\rho \equiv R_x - \tfrac{\mu_0}{\tau_x} \tag{6.132}$$

We insert results from the previous section and cancel out $i(e^{i\lambda y_j} - e^{-i\lambda y_j})$ from every term:

$$\boxed{\tfrac{1}{\tau_y}\left(4\mu_0 D C_b C_\theta - 2\mu_0 \Im\left[G - H\right]\right) = \tfrac{1}{3}(\rho + R_y) i \Im\left[G + H\right] \ . \qquad (6.133)}$$

Now we take the first moment of the cell y-slope equations by subtracting Eqs. (6.60)−(6.61), after multiplying by μ_0:

$$\tfrac{1}{3\tau_y}\left(\phi_{j+1/2} + \phi_{j-1/2} - 2\phi_j\right) + \tfrac{1}{3}\left(1 + \tfrac{\mu_0}{\tau_x}\right) J^y_j = 0 \ . \tag{6.134}$$

Inserting results from the previous section and cancelling $\left(e^{i\lambda y_j} + e^{-i\lambda y_j}\right)$ from every term gives:

$$\boxed{\tfrac{1}{3\tau_y}\left(-4DS_b C_\theta - 2\Re\left[G + H\right]\right) + \tfrac{1}{3}\mu_0\left(1 + \tfrac{\mu_0}{\tau_x}\right)\Re\left[G - H\right] = 0 \ . \qquad (6.135)}$$

6.3.5 X-Slope Terms

Next we subtract Eq. (6.63) from Eq. (6.62):

$$\begin{aligned}
2\pi(1+\xi)\left(\psi^{x,NE}_{j+1/2} - \psi^{x,NE}_{j-1/2}\right) + 2\pi\psi^{x,NE}_{j+1/2} + 2\pi(1+\xi)\left(\psi^{x,SE}_{j+1/2} - \psi^{x,SE}_{j-1/2}\right) - 2\pi\psi^{x,SE}_{j-1/2}& \\
= \tfrac{3\tau_x}{\mu_0 d}\left[-\tfrac{\mu_0}{\tau_y}(\psi^N_{j+1/2} - \psi^N_{j-1/2}) - \tfrac{\mu_0}{\tau_y}(\psi^S_{j+1/2} - \psi^S_{j-1/2})\right] \ ,& \\
= -\tfrac{3\tau_x}{d\tau_y}\left[\phi_{j+1/2} - \phi_{j-1/2}\right]&
\end{aligned} \tag{6.136}$$

or

$$2\pi(1+\xi)\left(\psi^{x,NE}_{j+1/2}+\psi^{x,SE}_{j+1/2}-\psi^{x,NE}_{j-1/2}-\psi^{x,SE}_{j-1/2}\right)+2\pi\left(\psi^{x,NE}_{j+1/2}-\psi^{x,SE}_{j-1/2}\right)$$
$$=-\tfrac{3\tau_x}{d\tau_y}\left[\phi_{j+1/2}-\phi_{j-1/2}\right] \tag{6.137}$$

Insert previous results and cancel the common $\left(e^{i\lambda y_j}-e^{-i\lambda y_j}\right)$ factor:

$$F(1+\xi)\left(e^{i\theta/2}-e^{-i\theta/2}\right)\left(e^{i\lambda b/2}-e^{-i\lambda b/2}\right)+F\left(e^{i(\lambda b+\theta)/2}+e^{-i(\lambda b+\theta)/2}\right)$$
$$=-\tfrac{3\tau_x}{d\tau_y}iD\left(e^{i\lambda b/2}-e^{-i\lambda b/2}\right)\left(e^{i\theta/2}-e^{-i\theta/2}\right) \tag{6.138}$$

or

$$-4F(1+\xi)S_bS_\theta+2F\left(C_bC_\theta-S_bS_\theta\right)=4\tfrac{3\tau_x}{d\tau_y}iDS_bS_\theta \tag{6.139}$$

or

$$F(3+2\xi)S_bS_\theta-FC_bC_\theta=-2\tfrac{3\tau_x}{d\tau_y}iDS_bS_\theta \tag{6.140}$$

or

$$\boxed{F=f_aiD \qquad \text{where} \qquad f_a\equiv-\frac{3\tau_x}{\tau_yd}\frac{2S_bS_\theta}{(3+2\xi)S_bS_\theta-C_bC_\theta}} \tag{6.141}$$

6.3.6 Collect and Solve

We adopt the definition of f_a and collect four previous equations, which we recall have the physical meaning of cell-avg 0^{th} moment, cell-avg 1^{st} moment, cell-y-slope 0^{th} moment, and cell-y-slope 1^{st} moment:

$$-4\tfrac{\mu_0}{\tau_y}DC_bS_\theta = R_y\Re\left[G+H\right] + 2\Lambda f_aD\left(S_bC_\theta - C_bS_\theta\right)\ , \tag{6.142}$$

$$-4\tfrac{1}{3\tau_y}DS_bS_\theta + \mu_0\left(1+R_x\right)\ \Im\left[G-H\right] = -2\mu_0\Lambda f_aD\left(C_bC_\theta + S_bS_\theta\right)\ , \tag{6.143}$$

$$\tfrac{1}{\tau_y}\left(4\mu_0DC_bC_\theta - 2\mu_0\Im\left[G-H\right]\right) = \tfrac{1}{3}(\rho+R_y)\Im\left[G+H\right]\ , \tag{6.144}$$

$$\tfrac{1}{3\tau_y}\left(-4DS_bC_\theta - 2\Re\left[G+H\right]\right) + \tfrac{1}{3}\mu_0\left(1+\tfrac{\mu_0}{\tau_x}\right)\Re\left[G-H\right] = 0\ . \tag{6.145}$$

Solve Eq. (6.145) for $\Re[H]$:

$$\tfrac{1}{3\tau_y}\left(-4DS_bC_\theta - 2\Re\left[G+H\right]\right) + \tfrac{1}{3}\mu_0\left(1+\tfrac{\mu_0}{\tau_x}\right)\Re\left[G-H\right] = 0$$

$$\tau_y\mu_0\left(1+\tfrac{\mu_0}{\tau_x}\right)\Re[G-H] = 4DS_bC_\theta + 2\Re[G+H]$$

$$\Rightarrow \Re[H]\left(2+\tau_y\mu_0\left(1+\tfrac{\mu_0}{\tau_x}\right)\right) = -4DS_bC_\theta - \Re[G]\left(2-\tau_y\mu_0\left(1+\tfrac{\mu_0}{\tau_x}\right)\right)$$

or

$$\Rightarrow \Re[H] = -\frac{4DS_bC_\theta + \Re[G]\left(2-\tau_y\mu_0\left(1+\tfrac{\mu_0}{\tau_x}\right)\right)}{2+\tau_y\mu_0\left(1+\tfrac{\mu_0}{\tau_x}\right)} \tag{6.146}$$

Solve Eq. (6.143) for $\Im[H]$:

$$-4\tfrac{1}{3\tau_y}DS_bS_\theta + \mu_0\left(1+R_x\right)\ \Im\left[G-H\right] = -2\mu_0\Lambda f_aD\left(C_bC_\theta + S_bS_\theta\right)$$

$$\Rightarrow -4DS_bS_\theta + 3\tau_y\mu_0(1+R_x)\Im[G-H] = -6\tau_y\mu_0\Lambda f_aD(C_bC_\theta + S_bS_\theta)$$

or

$$\Rightarrow \Im[G-H] = D\frac{4S_bS_\theta - 6\tau_y\mu_0\Lambda f_a\left(C_bC_\theta + S_bS_\theta\right)}{3\tau_y\mu_0\left(1+R_x\right)}$$

$$\Rightarrow \Im[H] = \Im[G] - D\frac{4S_bS_\theta - 6\tau_y\mu_0\Lambda f_a\left(C_bC_\theta + S_bS_\theta\right)}{3\tau_y\mu_0\left(1+R_x\right)} \tag{6.147}$$

Use these results in Eq. (6.142) and (6.144):

$$-4\tfrac{\mu_0}{\tau_y}DC_bS_\theta = R_y\Re\left[G+H\right] + 2\Lambda f_aD\left(S_bC_\theta - C_bS_\theta\right)$$

$$R_y = -D\frac{\left(4\frac{\mu_0}{\tau_y}C_bS_\theta + 2\Lambda f_a\left(S_bC_\theta - C_bS_\theta\right)\right)}{\Re[G+H]} \tag{6.148}$$

The only unknowns in this equation are λ (buried in the C and S terms) and R_y (which contains k). The D cancels out because $\Re[G+H]$ is proportional to D.

Continue with Eq. (6.144):

$$\tfrac{1}{\tau_y}\left(4\mu_0DC_bC_\theta - 2\mu_0\Im[G-H]\right) = \tfrac{1}{3}(\rho + R_y)\left(\Im[G+H]\right)$$

$$\Rightarrow R_y = \frac{\frac{3}{\tau_y}\left(4\mu_0DC_bC_\theta - 2\mu_0\Im[G-H]\right)}{\Im[G+H]} - \rho$$

$$\boxed{R_y = \frac{\frac{3}{\tau_y}\left(4\mu_0 D C_b C_\theta - 2\mu_0 \Im[G-H]\right) - \rho\Im[G+H]}{\Im[G+H]} \qquad (6.149)}$$

Again, the only unknowns in this equation are λ (buried in the C and S terms) and R_y (which contains k). Also, this equation does not depend on D because $\Im[G]$ and $\Im[H]$ both are proportional to D. Putting together the equations for R_y:

$$R_y = -D\frac{\left(4\frac{\mu_0}{\tau_y} C_b S_\theta + 2\Lambda f_a \left(S_b C_\theta - C_b S_\theta\right)\right)}{\Re[G+H]} \tag{6.150}$$

$$R_y = \frac{\frac{3}{\tau_y}\left(4\mu_0 D C_b C_\theta - 2\mu_0 \Im[G-H]\right) - \rho\Im[G+H]}{\Im[G+H]} \tag{6.151}$$

If both of these equations are to hold for the same value of R_y, they must be linearly dependent. That is, if they are expressed as $a_1 = b_1 R_y$ and $a_2 = b_2 R_y$, we recognize the necessary condition $a_1 b_2 - a_2 b_1 = 0$. This equation can be obtained several different ways, such as solving equation equation for R_y and equating them. But written this way, we have the best chance of avoiding numerical issues such as trying to evaluate 0/0.

Once we have a λ that satisfies our equation, we can determine R_y using either Equation 6.150 or 6.151. For numeric reasons, we choose to use Equation 6.150 to avoid dividing my zero in the small-Δy limit. When we determine R_y, this specifies k-effective. Also we can determine the scalar flux shape using Equations 6.84 and 6.85 along with:

$$\Re[H] = -\frac{4DS_bC_\theta + \Re[G]\left(2 - \tau_y\mu_0\left(1 + \frac{\mu_0}{\tau_x}\right)\right)}{2 + \tau_y\mu_0\left(1 + \frac{\mu_0}{\tau_x}\right)} \tag{6.152}$$

$$\Im[H] = \Im[G] - D\frac{4S_bS_\theta - 6\tau_y\mu_0\Lambda f_a\left(C_bC_\theta + S_bS_\theta\right)}{3\tau_y\mu_0\left(1 + R_x\right)} \tag{6.153}$$

$$\Re[G] = -D\left(C_bS_\theta + S_bC_\theta\right) \tag{6.154}$$

$$\Im[G] = D\left(C_bC_\theta - S_bS_\theta\right) \tag{6.155}$$

The D in these equations is a scaling factor, and can be chosen to be anything. Therefore we will chose it to match our test cases by integrating over the problem and taking the ratio of our test case data and the flux shape we get from this analysis.

6.4 Results

To test our analysis we developed a series of test problems, both with and without oscillations. Table 6.1 shows the details of these problems.

Table 6.1: Details of Test Problems

Case Number	Δy	b	a	Oscillations?
1	0.5	50	10,000	no
2	0.5	50	0.5	no
3	2.5	50	0.5	yes
4	2.5	50	1.0	yes
5	2.5	50	2.0	no
6	2.5	50	0.05	yes

We then developed a python script that executes the analysis we have derived here. As a sanity test, our first test case imitates a 1D problem. In Figure 6.1 we have plotted the k-effective computed where the function $a_1b_2 - a_2b_1$ crosses zero for test case 1. The python script then searches for the largest k-effective for λ between [0, $\frac{\pi}{\Delta y}$]. This range was chosen because, for the problems we tested, the function cycles on that interval. For the test problem used in Figure 6.1, the k-effective chosen as the largest was 1.19944, which is 3e-5 pcm different than the k-effective computed using PDT. Figure 6.2 shows the flux shape for both the test case and what was computed with our analysis. Our analysis gets very good agreement for the flux shape.

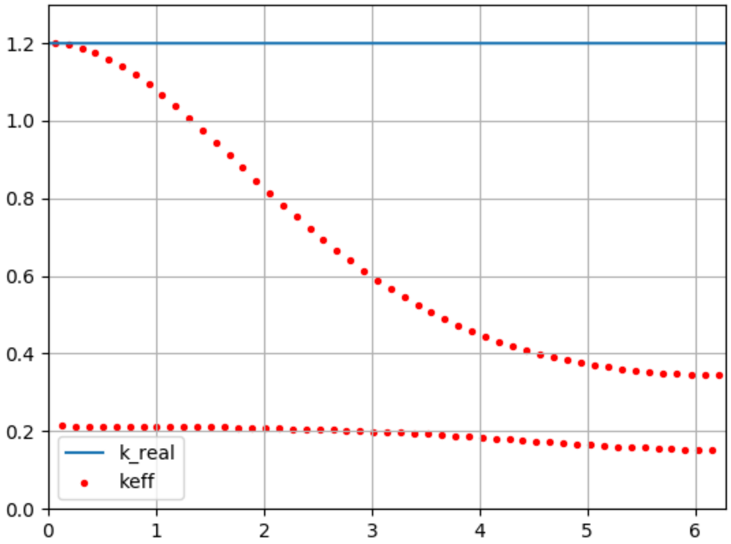

Figure 6.1: k-effective vs λ for Case 1

In Figure 6.3 we have plotted the k-effective computed where the function $a_1b_2 - a_2b_1$ crosses zero for test case 2. For the test problem used in Figure 6.3, the k-effective chosen as the largest was 0.55687, which is 4562 pcm different than the k-effective computed using PDT. All other test cases had a pcm difference much lower than this. The source of this large discrepency is on-going work.

Figure 6.4 shows the flux shape for both the test case and what was computed with our analysis.

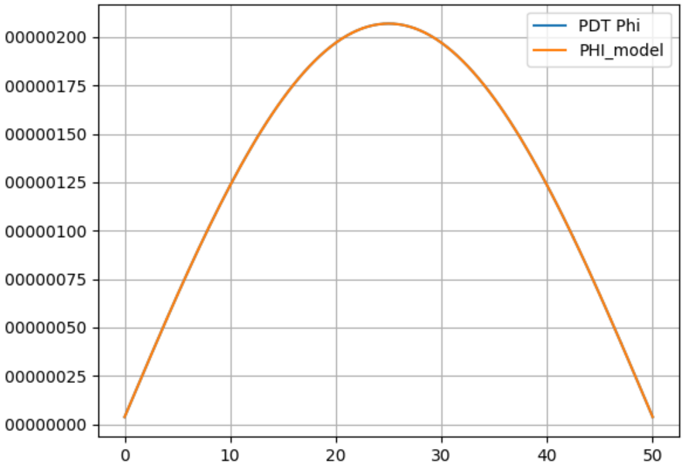

Figure 6.2: Flux Shape for Case 1

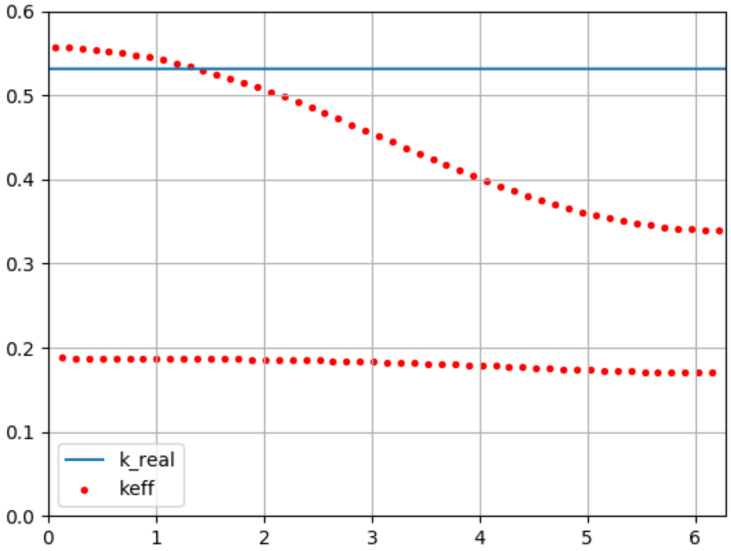

Figure 6.3: k-effective vs λ for Case 2

Our analysis gets very good agreement for the flux shape.

In Figure 6.5 we have plotted the k-effective computed where the function $a_1b_2 - a_2b_1$ crosses

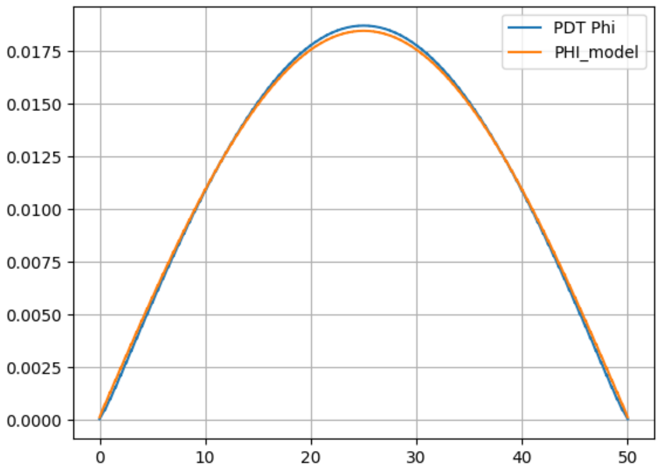

Figure 6.4: Flux Shape for Case 2

zero for test case 3. The only difference between test case 1 and 3 is Δy. The k-effective chosen by our analysis for test case 3 was 0.64117, which is 15 pcm different than the k-effective computed with PDT for this problem.

Figure 6.6 shows the flux shape for case 3. Again, we get very good agreement between our model and what PDT computes.

In Figure 6.7 we have plotted the k-effective computed where the function $a_1b_2 - a_2b_1$ crosses zero for test case 4. The only difference between test case 3 and 4 is a. The k-effective chosen by our analysis for test case 4 was 0.79198, which is 12 pcm different than the k-effective computed with PDT for this problem.

Figure 6.8 shows the flux shape for case 4. Again, we get very good agreement between our model and what PDT computes.

In Figure 6.9 we have plotted the k-effective computed where the function $a_1b_2 - a_2b_1$ crosses zero for test case 5. The only difference between test case 3 and 5 again is a. The k-effective chosen by our analysis for test case 5 was 0.96290, which is 361 pcm different than the k-effective computed with PDT for this problem.

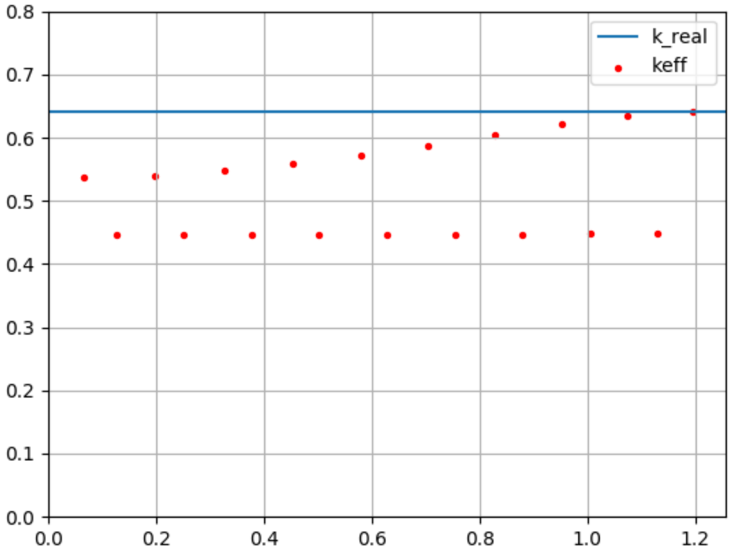

Figure 6.5: k-effective vs λ for Case 3

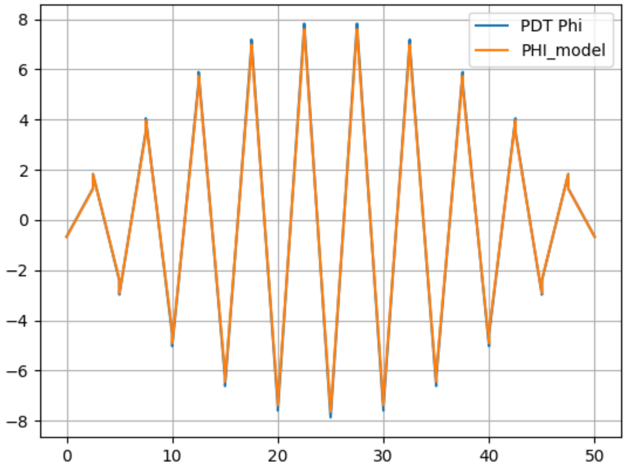

Figure 6.6: Flux Shape for Case 3

Figure 6.10 shows the flux shape for case 5. Again, we get very good agreement between our model and what PDT computes.

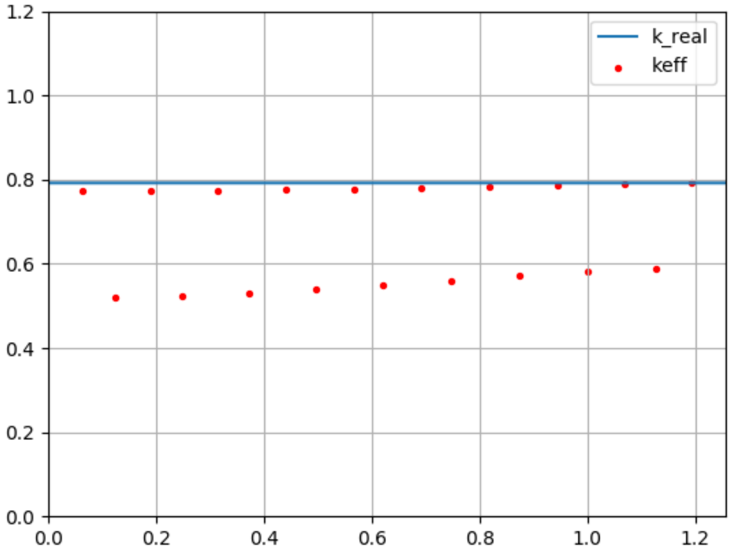

Figure 6.7: k-effective vs λ for Case 4

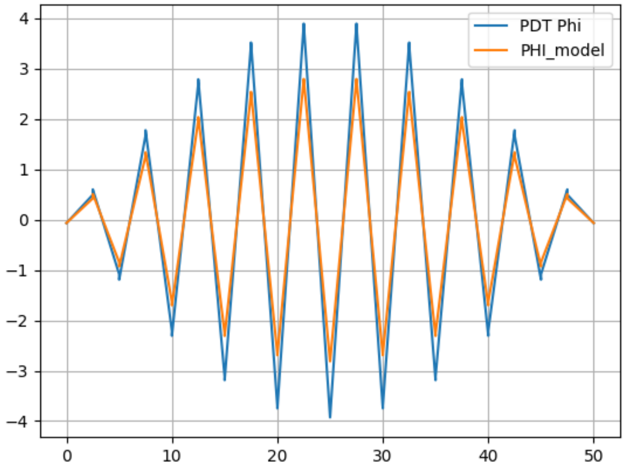

Figure 6.8: Flux Shape for Case 4

In Figure 6.11 we have plotted the k-effective computed where the function $a_1b_2 - a_2b_1$ crosses zero for test case 5. The only difference between test case 3 and 6 again is a. This case was chosen

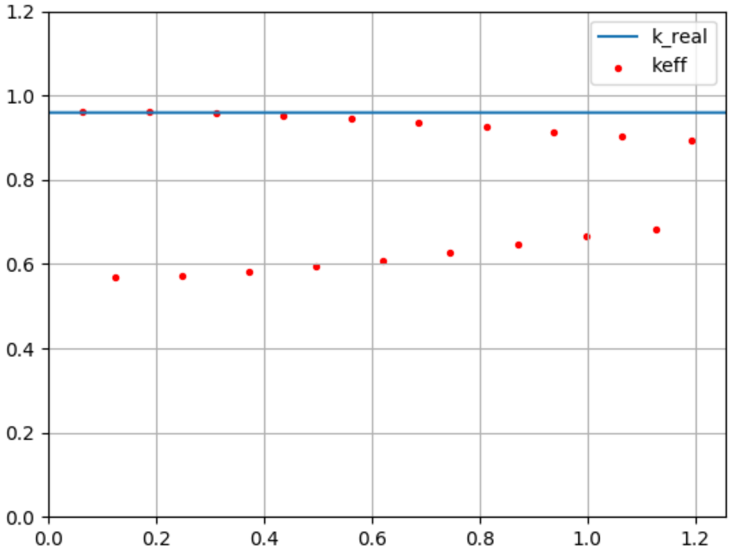

Figure 6.9: k-effective vs λ for Case 5

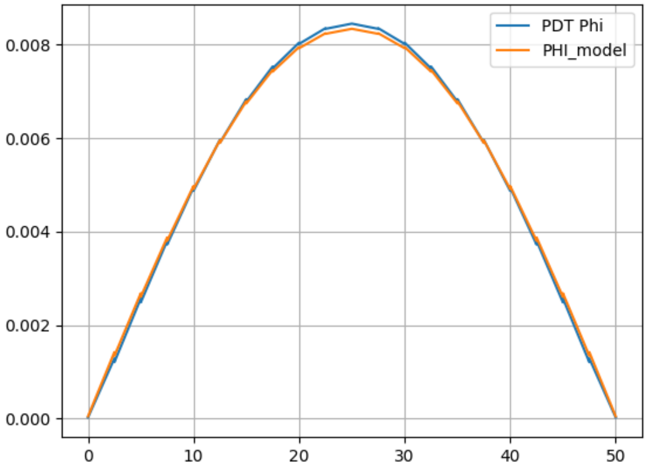

Figure 6.10: Flux Shape for Case 5

to see if our analysis holds for small a. The k-effective chosen by our analysis for test case 5 was 0.13761, which is 0.3 pcm different than the k-effective computed with PDT for this problem.

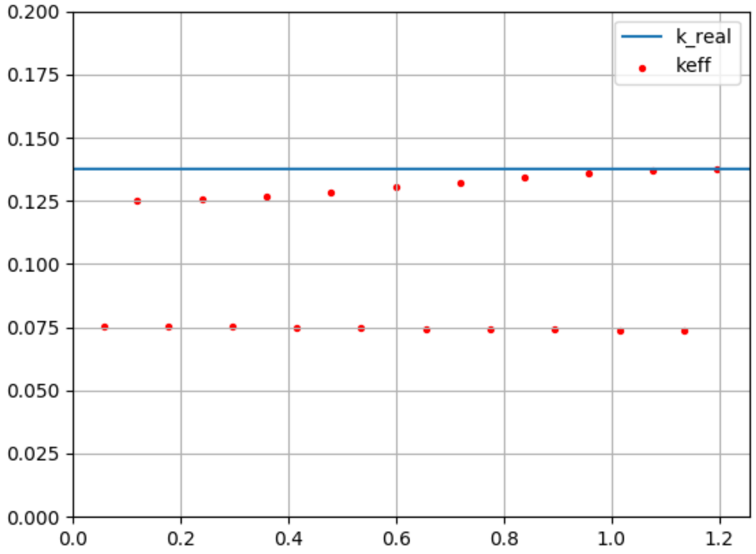

Figure 6.11: k-effective vs λ for Case 6

Figure 6.12 shows the flux shape for case 6. Again, we get very good agreement between our model and what PDT computes.

Table 6.2 summarizes the results we obtained for k-effective for each of our test cases.

Table 6.2: k-effective Results

Case Number	Model k	PDT k	pcm difference
1	1.19944	1.19944	3e-5
2	0.55687	0.53147	4562
3	0.64117	0.64107	15
4	0.79198	0.79188	12
5	0.96290	0.95942	361
6	0.13761	0.13760	0.3

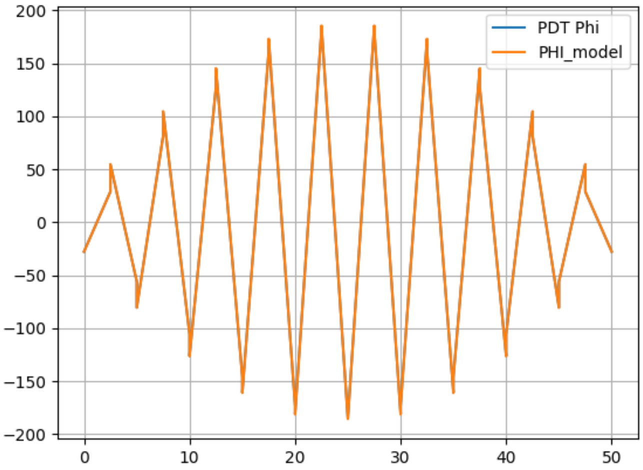

Figure 6.12: Flux Shape for Case 6

6.5 Conclusions

We have studied the oscillation phenomenon we see in k-eigenvalue problems and have determined when and why they occur. We see that they occur when Δy is large compared to Δx and when the total problem width is small compared to its height. We have demonstrated with a series of problems that our mathematical derivation is accurate in determining when oscillations occur, at least for these simple problems.

We do see a discrepency in the k-eigenvalue produced using our mathematical derivation and that computed using PDT for problems 2 and 5. The math derived here makes no approximations given the problems we study here - single cell in x, S_2 quadrature, homogeneous material. We believe that there is a sign error or coefficient wrong somewhere in the derivation but have yet been unable to find it.

7. SUMMARY

The Linear Discontinuous Finite Element Method has been implemented in the massively parallel transport code PDT and applied to reactor calculations. A series of problems featuring a range of spatial, angular, and energy discretization choices in 2D and 3D problems were run and the spatial accuracy was analyzed. We found that for a typical 2D PWR reactor assembly, using 76 spatial cells per pin cell, we achieve an accuracy of 3 pcm in k-effective. We also found that in those problems, the energy error was much greater than the spatial error.

We developed an error model which treats both spatial and angular discretization error for a given energy discretization. This error model, which uses a least squares approach, accurately predicts quantities of interest computed with both our training data and meshes more refined than our training data. This gives us confidence that our predictions for quantities of interest computed using infinitely refined meshes are also accurate. This allows to accurately predict QOIs for problems we do not yet have the means to run.

We have compared to the Method of Characteristics through a collaboration with researchers at the University of Michigan and their MOC code MPACT. We have concluded that for problems with geometric features or boundary layers that are extremely small relative to the problem domain, LD more accurately computes quantities of interest when using meshes similar to MOC. We also note that when going from 2D to 3D problems, the complexity of LD only raises from solving 3x3 matrices to solving 4x4 matrices while MOC must compute many more tracks and the ray spacing becomes much more computationally expensive. This gives us confidence that LD is a promising alternative to MOC for 3D reactor problems.

Finally unphysical oscillations in LD solutions to k-eigenvalue problems have been observed and explained using Fourier analysis. We have described the cause of these oscillations as well as what conditions are present when they occur. We have determined that the oscilllations appear when Δy is large compared to Δx and when the problem height is large compared to its width. We can now predict, for simple problems, both the scalar flux shape and the k-eigenvalue for problems with and without unphysical oscillations.

Future work includes finding the error in the oscillation math that produces discrepancies in k-effective for certain problems. Also, extending this analysis to more computationally complex 3D problems such as full reactor cores would benefit the community.

REFERENCES

[1] W. Boyd, K. Smith, and B. Forget, "Parallel performance results for the openmoc method of characteristics code on multi-core platforms," in *PHYSOR Conference*, (Kyoto, Japan), September 28-October 3 2014.

[2] M. Ryu, Y. Yung, H. Cho, and H. Joo, "Solution of the beavrs benchmark using the ntracer direct whole core transport code," in *PHYSOR Conference*, (Kyoto, Japan), September 28-October 3 2014.

[3] B. Collins, S. Stimpson, A. Godfrey, K. B., and T. Downar, "Assessment of the 2d/1d implementation in mpact," in *PHYSOR Conference*, (Kyoto, Japan), September 28-October 3 2014.

[4] M. Adams, T. A. Wareing, and W. F. Walters, "Characteristic methods in thick diffusive problems," *Nuclear Science and Engineering*, vol. 130, pp. 18–46, 1998.

[5] M. L. Adams, "Discontinuous finite element methods in thick diffusive problems," *Nucl. Sci. Eng.*, vol. 137, pp. 298–333, 2001.

[6] Y. Wang and J. C. Ragusa, "On the convergence of dgfem applied to the discrete ordinates transport equation for structured and unstructured triangular meshes," *Nuclear Science and Engineering*, vol. 163:1, pp. 56–72, 2009.

[7] W. Boyd, S. Shaner, L. Li, B. Forget, and K. Smith, "The openmoc method of characteristics neutral particle transport code," *Annals of Nucl. Energy*, 2013.

[8] R. Ferrer and J. Rhodes, "Extension of linear source moc methodology to anisotropic scattering in casmo5," in *PHYSOR Conference*, (Kyoto, Japan), September 28-October 3 2014.

[9] B. Collins, S. Stimpson, and T. Downar, "Axial transport solvers for the 2d/1d scheme in mpact," in *PHYSOR Conference*, (Kyoto, Japan), September 28-October 3 2014.

[10] A. Buss, Harshvardan, I. Papadopoulos, O. Pearce, T. Smith, G. Tanase, N. Thomas, X. Xu, M. Bianco, N. M. Amato, and L. Rauchwerger, “Stapl: Standard template adaptive parallel library,” in *SYSTOR*, (Haifa, Israel), June 4-6 2010.

[11] G. Tanase, A. Buss, A. Fidel, Harshvardan, I. Papadopoulos, O. Pearce, T. Smith, N. Thomas, X. Su, N. Mourad, M. Bianco, N. M. Amato, and L. Rauchwerger, “The stapl parallel container framework,” in *Proc. ACM SIGPLAN Symp. Prin. Prac. Par. Prog.*, 2010.

[12] A. Buss, A. Fidel, Harshvardan, T. Smith, G. Tanase, N. Thomas, X. Xu, M. Bianco, N. M. Amato, and L. Rauchwerger, “The stapl pview,” in *LCPC*, (Houston), October 7-9 2010.

[13] Harshvardan, A. Fidel, N. M. Amato, and L. Rauchwerger, “The stapl parallel graph library,” in *LCPC*, (Tokyo, Japan), September 2012.

[14] M. P. Adams, M. L. Adams, W. D. Hawkins, T. Smith, L. Rauchwerger, N. M. Amato, T. S. Bailey, and R. D. Falgout, “Provably optimal parallel transport sweeps on regular grids,” in *Proc. International Conference on Mathematics and Computational Methods applied to Nuclear Science and Engineering*, (Sun Valley, Idaho), May 5-9 2013.

[15] “Benchmark on deterministic transport calculations without spatial homogenisation,” *NEA/NSC/DOC*, vol. 16, 2003.

[16] M. A. Smith, E. E. Lewis, and B.-C. Na, “Benchmark on deterministic 3-d mox fuel assembly transport calculations without spatial homogenisation,” *Progress in Nuclear Energy*, vol. 48, 2006.

[17] N. Valette and M. L. Adams, “Accuracy of quasi-diffusion discretizations,” *Trans. Amer. Nucl. Soc.*, vol. 87, pp. 136–139, 2002.

[18] J. Pitkranta and L. R. Scott, “Error estimates for the combined spatial and angular approximations of the transport equation for slab geometry,” *SIAM J. Num. Anal.*, vol. 20, pp. 922–950, 1983.

[19] C. McGraw, M. Adams, D. Hawkins, M. Adams, and T. Smith, “Accuracy of the linear discontinuous galerkin method for reactor analysis with resolved fuel pins,” in *PHYSOR*, (Kyoto, Japan), September 28 - October 3 2014.

[20] C. McGraw, M. Adams, D. Hawkins, M. Adams, and T. Smith, “Accuracy of the linear discontinuous galerkin method for 3d reactor analysis with resolved fuel pins,” in *M and C*, (Nashville, Tennessee), April 19 - 23 2015.

[21] A. Till, *Finite Elements with Discontiguous Support for Energy Discretization in Particle Transport.* Ph.D. dissertation, Texas A&M University, 2015.

[22] A. T. Godfrey, “Vera core physics benchmark progression problem specifications.” `https://www.casl.gov/sites/default/files/docs/CASL-U-2012-0131-004.pdf`. Accessed: 2014-08-01.

[23] “Casl website.” `http://http://www.casl.gov/`. Accessed: 2017-02-03.