An Azimuthal, Fourier Moment-Based Axial S_N Solver for the 2D/1D Scheme

by

Shane Gray Stimpson

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Nuclear Engineering and Radiological Sciences and Scientific Computing)
in the University of Michigan
2015

Doctoral Committee:

Professor Thomas J. Downar, Co-Chair
Reactor Physics Nuclear Engineer Benjamin S. Collins, ORNL, Co-Chair
Professor Edward W. Larsen
Professor William R. Martin
Professor Divakar Viswanath

2015

ACKNOWLEDGMENTS

Most milestones in life are generally not reached alone, and I would like the thank everyone who has played a part in helping me reach this goal.

First, I wish to thank my co-advisors Prof. Thomas Downar and Dr. Benjamin Collins. Your guidance has been invaluable, and it has been a pleasure to work under your advisement. I would also like to thank the other members of my committee: Prof. Edward Larsen, Prof. William Martin, and Prof. Divakar Viswanath. I have learned a great deal from each of you, and I appreciate your time and effort in serving on my committee.

Next, I must express many thanks to all members of the MPACT team. I became part of the development team fairly early and have been fortunate to be a part of a project that has grown considerably in the past couple of years. Also, members of the Consortium for the Advanced Simulation of Light Water Reactors (CASL) [3] should be acknowledged as this work was funded through it under Department of Energy (DOE) contract number DE-AC05-00OR22725. I would also like to specifically thank Dr. Tara Pandya from Oak Ridge National Laboratory for her assistance in generating reference solutions with SHIFT.

Lastly and most importantly, I would like to thank my family for their endless support. In particular, I owe an extraordinary amount of appreciation to my wife, Rachel, for her inexhaustible encouragement and patience. I am very fortunate to have you in my life.

TABLE OF CONTENTS

Acknowledgments . . . . . **ii**

List of Figures . . . . . **vi**

List of Tables . . . . . **viii**

List of Appendices . . . . . **ix**

Abstract . . . . . **x**

Chapter 1 Introduction **1**

1.1 Motivation 1
1.2 History and Previous Work with "2D/1D" 2
1.3 Dissertation Layout 4

Chapter 2 Neutron Transport Methodology **6**

2.1 Transport Theory 6
2.1.1 The Boltzmann Transport Equation 6
2.1.2 Steady-State Transport Equation 9
2.1.3 The Multigroup Approximation. 10
2.1.4 Scattering Approximations 11
2.1.4.1 Anisotropic Scattering 11
2.1.4.2 Transport-Corrected Scattering 14
2.1.5 The Discrete Ordinates Approximation. 15
2.1.5.1 Quadrature Sets 15
2.1.6 The Diffusion Approximation 17
2.1.7 Constant Material Properties 19
2.1.8 Eigenvalue Problems 19
2.2 Fundamental Numerical Methods 20
2.2.1 Method of Characteristics (MOC) 20
2.2.1.1 Modularization 22
2.2.1.2 Other MOC Concepts 23
2.2.2 Coarse Mesh Finite Difference (CMFD) 23
2.2.3 Nodal Methods 25
2.2.3.1 Nodal Expansion Method (NEM) 25
2.2.3.2 Source Expansion Nodal Method (SENM) 26
2.2.3.3 Simplified P_N (SP_N) 27
2.2.3.4 One-Node vs. Two-Node 29

2.2.4 Monte Carlo Methods . . . 30
Chapter 3 The 2D/1D Framework . . . 32
3.1 Governing Equations . . . 32
3.1.1 Radial Equations . . . 32
3.1.1.1 Angular Dependence of the Axial Transverse Leakage. . . . 34
3.1.1.2 Polar Asymmetry . . . 34
3.1.2 Axial Equations . . . 35
3.1.2.1 Transport-Based . . . 35
3.1.2.2 Diffusion-Based . . . 38
3.1.2.3 Transverse Leakage Interpolation . . . 38
3.2 Transverse Leakage Splitting . . . 40
3.3 Iteration Strategy . . . 42
3.4 Relaxation . . . 43
3.5 Angular Distribution Scaling . . . 46
3.6 Sources of Error . . . 47
3.7 Parallel Decomposition Scheme . . . 51
3.7.1 Spatial Decomposition . . . 51
3.7.2 Angle Decomposition . . . 52
3.7.3 Ray Decomposition . . . 52
Chapter 4 The Axial S_N Sweeper . . . 53
4.1 Spatial Moments . . . 53
4.2 Azimuthal Moments . . . 57
4.2.1 Important Identities . . . 57
4.2.2 Axial Sweeper Derivation . . . 57
4.2.2.1 Zeroth-Moment . . . 60
4.2.2.2 Sine Moments . . . 61
4.2.2.3 Cosine Moments . . . 62
4.2.2.4 Boundary Conditions . . . 62
4.2.3 Transverse Leakage Coefficients . . . 62
4.2.4 Visualization . . . 63
4.3 Axial TL Summary and MOC Sweepers . . . 69
4.4 Parallelism . . . 73
4.4.1 Spatial Decomposition . . . 74
4.4.2 Azimuthal Angle/Moment Decomposition . . . 74
4.4.3 Polar Angle Decomposition . . . 75
4.5 Iteration Strategy . . . 75
4.6 Anisotropic Scattering . . . 77
4.7 Asymptotic Behavior . . . 79
4.7.1 Radial Solver . . . 79
4.7.2 Axial Solver . . . 80
4.7.3 Fourier Moment Expansions . . . 81
Chapter 5 Numerical Results . . . 82

5.1 Takeda LWR Rodded Benchmark . . . 83
5.1.1 Problem Description and Discretization . . . 83
5.1.2 Results . . . 84
5.2 C5G7 Benchmarks . . . 88
5.2.1 Geometry Description and Discretization . . . 88
5.2.2 3D, Heterogeneous Pin Results . . . 91
5.2.3 Discussion of Heterogeneous Pin Results . . . 96
5.2.4 3D, Homogeneous Pin Results . . . 97
5.2.5 Axial Mesh Refinement . . . 103
5.2.6 Cancellation of Error . . . 106
5.2.7 Memory Footprint and Timing Comparison . . . 107
5.2.8 Parallel Performance . . . 113

Chapter 6 Conclusions . . . 117

6.1 Summary of Work . . . 117
6.2 Proposed Future Research Topics . . . 119
6.2.1 Axial Transverse Leakage Spatial Distribution . . . 119
6.2.2 Spherical Harmonic Expansions for Angular Flux and Leakages . . . 120
6.2.3 Transverse Leakage Splitting . . . 120
6.2.4 Subplane Method . . . 121
6.2.5 Angle-Dependent Total Cross Sections . . . 121

Appendices . . . 122

Bibliography . . . 160

LIST OF FIGURES

1.1 2D/1D Concept 2

2.1 Definition of Azimuthal (α) and Polar (μ) Angles 7
2.2 Total Cross Section ($\Sigma_T(E)$) of U-235 10
2.3 Representative Level-Symmetric Quadrature 16
2.4 Representative Product Quadrature 17
2.5 Pin Cell Azimuthal Divisions 21
2.6 Rays at Different Azimuthal Angles 22
2.7 Ray Modularization 23
2.8 CMFD Pin Cell Homogenization 24
2.9 One-Node Kernel 29
2.10 Two-Node Kernel 29

3.1 Illustration of Polar Symmetry 35
3.2 TL Interpolation 39
3.3 Angle-Dependent TL Splitting 41
3.4 Spatial Source Shifting 41
3.5 2D/1D Calculation Flow 43
3.6 Group-Dependent Optimal Relaxation Factors 45
3.7 Convergence Rate of Various Relaxation Schemes 46

4.1 3x3 Pin Test Case Geometry 64
4.2 Radial TL, Rodded - Fast Group 65
4.3 Radial TL, Rodded - Fast Group (Bottom-Right Pin) 65
4.4 Radial TL, Rodded - Thermal Group 66
4.5 Radial TL, Rodded - Thermal Group (Bottom-Right Pin) 66
4.6 Radial TL, Rodded, Vacuum - Fast Group 67
4.7 Radial TL, Rodded, Vacuum - Fast Group (Bottom-Right Pin) 68
4.8 Radial TL, Rodded, Vacuum - Thermal Group 68
4.9 Radial TL, Rodded, Vacuum - Thermal Group (Bottom-Right Pin) 69
4.10 Axial Solvers Calculation Flow 76
4.11 Axial S_N Calculation Flow 77

5.1 Takeda Geometry - Top View 83
5.2 Takeda Geometry - Side View (South) 83
5.3 C5G7 - Radial Layout 89

5.4 C5G7 - Configuration UR . . . 89
5.5 C5G7 - Rodded Configuration A . . . 90
5.6 C5G7 - Rodded Configuration B . . . 90
5.7 C5G7 - Radial Layout . . . 97
5.8 C5G7-UR (hom.) - Pin Power Error (ISO TL) . . . 98
5.9 C5G7-UR (hom.) - Pin Power Error (EXP TL) . . . 99
5.10 C5G7-rA (hom.) - Pin Power Error (ISO TL) . . . 100
5.11 C5G7-rA (hom.) - Pin Power Error (EXP TL) . . . 101
5.12 C5G7-rB (hom.) - Pin Power Error (ISO TL) . . . 102
5.13 C5G7-rB (hom.) - Pin Power Error (EXP TL) . . . 103
5.14 C5G7-UR (hom.) - Pin Power Error (EXP TL) - 1.785 cm Axial Mesh . . . 104
5.15 C5G7-UR (hom.) - Pin Power Error (EXP TL) - 0.8925 cm Axial Mesh . . . 104
5.16 C5G7-rA (hom.) - Pin Power Error (EXP TL) - 1.785 cm Axial Mesh . . . 105
5.17 C5G7-rA (hom.) - Pin Power Error (EXP TL) - 0.8925 cm Axial Mesh . . . 105
5.18 C5G7-rB (hom.) - Pin Power Error (EXP TL) - 1.785 cm Axial Mesh . . . 106
5.19 C5G7-rB (hom.) - Pin Power Error (EXP TL) - 0.8925 cm Axial Mesh . . . 106
5.20 Explicit TL Memory Profile . . . 110
5.21 Moment-0 TL Memory Profile . . . 110
5.22 Moment-2 TL Memory Profile . . . 111
5.23 Moment-4 TL Memory Profile . . . 111
5.24 Planar Decomposition Strong Scaling . . . 114
5.25 Angular Decomposition Strong Scaling . . . 115
5.26 Moment Decomposition Weak Scaling . . . 116

B.1 SHIFT 2D-C5G7 Pin Powers . . . 148
B.2 SHIFT 2D-C5G7 Pin Power Errors . . . 148
B.3 SHIFT 2D-C5G7 Pin Power Symmetry Difference . . . 149
B.4 SHIFT 3D-C5G7-UR Pin Powers . . . 149
B.5 SHIFT 3D-C5G7-UR Pin Power Errors . . . 150
B.6 SHIFT 3D-C5G7-UR Pin Power Symmetry Difference . . . 150
B.7 SHIFT 3D-C5G7-rA Pin Powers . . . 150
B.8 SHIFT 3D-C5G7-rA Pin Power Errors . . . 151
B.9 SHIFT 3D-C5G7-rA Pin Power Symmetry Difference . . . 151
B.10 SHIFT 3D-C5G7-rB Pin Powers . . . 151
B.11 SHIFT 3D-C5G7-rB Pin Power Errors . . . 152
B.12 SHIFT 3D-C5G7-rB Pin Power Symmetry Difference . . . 152
B.13 SHIFT 3D-C5G7-UR (Hom. Pins) Pin Powers . . . 153
B.14 SHIFT 3D-C5G7-UR (Hom. Pins) Pin Power Errors . . . 153
B.15 SHIFT 3D-C5G7-UR (Hom. Pins) Pin Power Symmetry Difference . . . 154
B.16 SHIFT 3D-C5G7-rA (Hom. Pins) Pin Powers . . . 154
B.17 SHIFT 3D-C5G7-rA (Hom. Pins) Pin Power Errors . . . 154
B.18 SHIFT 3D-C5G7-rA (Hom. Pins) Pin Power Symmetry Difference . . . 155
B.19 SHIFT 3D-C5G7-rB (Hom. Pins) Pin Powers . . . 155
B.20 SHIFT 3D-C5G7-rB (Hom. Pins) Pin Power Errors . . . 155
B.21 SHIFT 3D-C5G7-rB (Hom. Pins) Pin Power Symmetry Difference . . . 156

LIST OF TABLES

2.1 Number of Flux Moments Stored for Anisotropic Scattering 13

5.1 Cross Section Data 84
5.2 Takeda Rodded Results 85
5.3 Takeda Rodded Flux Results 86
5.4 Takeda Rodded Results (Compared to 3D-MOC) 87
5.5 Takeda Rodded Results (Refined) 87
5.6 3D-C5G7 Unrodded (UR) Results 92
5.7 3D-C5G7 Unrodded (UR) Moment Results (Compared to EXP TL) 92
5.8 3D-C5G7 Rodded Config. A (rA) Results 93
5.9 3D-C5G7 Rodded Config. A (rA) Moment Results (Compared to EXP TL) . . 93
5.10 3D-C5G7 Rodded Config. B (rB) Results 95
5.11 3D-C5G7 Rodded Config. B (rB) Moment Results (Compared to EXP TL) . . 95
5.12 3D-C5G7 Comparison to 3D-MOC) 96
5.13 C5G7-UR (hom.) Results 98
5.14 C5G7-UR (hom.) Moment Results (Compared to EXP TL) 99
5.15 C5G7-rA (hom.) Results 100
5.16 C5G7-rA (hom.) Moment Results (Compared to EXP TL) 101
5.17 C5G7-rB (hom.) Results 102
5.18 C5G7-rB (hom.) Moment Results (Compared to EXP TL) 103
5.19 C5G7-UR (hom.) Mesh Refinement 104
5.20 C5G7-rA (hom.) Mesh Refinement 105
5.21 C5G7-rB (hom.) Mesh Refinement 106
5.22 C5G7-rB (hom.) Results 107
5.23 3D-C5G7 Rodded Configuration B (rB) Timing Results 108
5.24 3D-C5G7 Unrodded (UR) Total Memory (Fuel Plane) Results 109
5.25 3D-C5G7-UR Angular Construct Memory (Fuel Plane) Results 112

LIST OF APPENDICES

Chapter A Supplemental Derivations **122**

Chapter B C5G7 SHIFT Results **147**

Chapter C Homogenized-Pin 3D-C5G7 Cross Sections **157**

ABSTRACT

Despite the incredible advancements in computing power in recent decades, using explicitly three-dimensional neutron transport methods is still very computationally expensive. Several alternative methods have been developed that use various assumptions to make computation more tractable, while still maintaining a higher fidelity than has been the industry standard in reactor analysis. One such method is the "2D/1D" scheme, which decomposes three-dimensional geometries into an axial stack of radial planes. In this scheme, one common approximation is to assume that the radial and axial transverse leakages that couple the axial and radial solvers are isotropic, which means that all angular dependence of the leakage is neglected. For more complicated problems, such as those with control rods or mixed oxide (MOX) fuels, higher fidelity treatment of the axial and radial leakages is needed to better capture the relationship between the solvers.

The first objective of the work presented here investigates incorporating full angular dependence of both the azimuthal and polar angles into the transverse leakages. Fully explicit angular dependence is shown to be particularly burdensome, both in terms of memory and run time requirements. The second, more novel objective uses a Fourier series expansion to account for the azimuthal dependence, requiring the formulation of a new axial S_N solver to generate angular fluxes for the axial transverse leakage construction.

In several test cases analyzed, which include cases with both control rods and MOX fuels, noteworthy accuracy gains are observed by including the angular dependence of the leakages. The Fourier moment-based approach performs very well, accurately capturing the azimuthal dependence with only a few moments. Overall, the Fourier moment-based approach reduces the run time by roughly a factor of 1.5, the aggregate memory footprint by a factor of 3 to 4, and angle-dependent variables by an order of magnitude.

Other test problems highlight one of the remaining sources of error relating to the spatial distribution of the axial transverse leakage, which is introduced because the axial solver operates on a coarse radial grid. The results suggest that by including a more accurate angular representation, some cancellation of error between the spatial and angular treatments is removed, indicating that future work focusing on improving the spatial distribution should be pursued.

CHAPTER 1

Introduction

1.1 Motivation

In the past several decades, computational methods for nuclear reactor analysis have changed immensely. Historically, a two-step approach, in which assembly-level cross sections are generated using higher order transport solvers and used in three-dimensional nodal diffusion codes, has been the workhorse in the design and simulation of nuclear reactors. This approach has been effective because the computational burden of simulation is very small as nodal diffusion is a lightweight, low fidelity method, particularly with assembly- or quarter assembly-sized nodes. Additionally, these codes have the very practical ability of being easily and quickly run on desktop computers, which until recently were limited to single processor machines. However, as computational resources have become both faster and cheaper, multi-core machines are the new standard and leadership computing facilities have invested millions of dollars into large scale cluster machines with hundreds of thousands of cores. As a result, more attention is being given to massively parallelizable, high fidelity transport methods that make many fewer approximations both physically and geometrically. Despite the incredible advances in computing power, using explicitly three-dimensional methods is still very computationally expensive. There are several methods that have been developed that make various assumptions to ease this burden, while still maintaining a higher fidelity than has been the norm in reactor analysis.

One such method is the "2D/1D" scheme, which decomposes three-dimensional geometries into an axial stack of radial planes (Figure 1.1). In this scheme, higher fidelity transport methods, such as the method of characteristics (MOC), are typically used to explicitly capture the radial, more heterogeneous geometry characteristics (cylindrical fuel, clad, fuel-clad gap, core baffle/barrel, etc.). In general, lower fidelity methods, such as NEM or SP_N are used axially on a pin-homogenized basis, as the geometric changes and flux gradients in the axial direction tend to be less severe. With these solvers, one common

approximation is to assume that the radial and axial transverse leakages that couple the axial and radial solvers are isotropic, meaning that all angular dependence of the leakage is neglected. However, for more complicated problems, such as those with control rods or mixed oxide (MOX) fuels, higher fidelity treatment of the axial and radial leakages are needed to better capture the relationship between the radial and axial solvers.

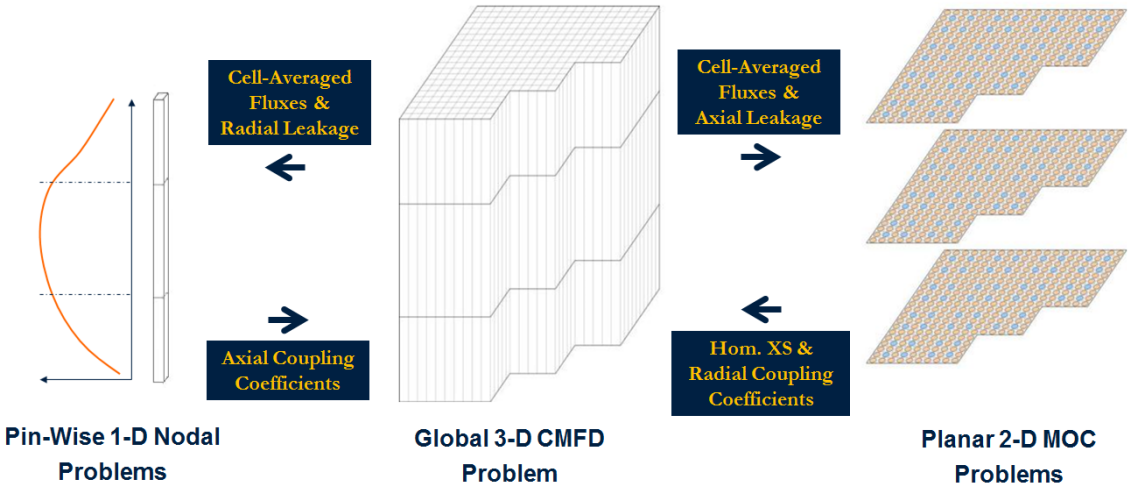

Figure 1.1: 2D/1D Concept

The work presented here has two primary goals. The first is to demonstrate the benefit of higher order angular treatment. Explicit angular representation, meaning full azimuthal and polar dependence, can be particularly burdensome in terms of both memory footprint and run time requirements. The second goal is to ease this burden by proposing a axial solver based on azimuthal, Fourier moments, rather than explicit azimuthal dependence. This can significantly reduce the memory requirements and run time while still maintaining a good representation of the azimuthal behavior.

All concepts covered in this work were implemented into the MPACT code, a reactor core simulator being developed collaboratively between the University of Michigan and Oak Ridge National Laboratory.

1.2 History and Previous Work with "2D/1D"

The "2D/1D" concept as used in this context originated through work performed by two groups in Korea. One of these groups, based out of the Korea Advanced Institute of Science and Technology (KAIST), developed what became known as the "2D/1D Fusion" method as implemented in the CRX code [7–9, 33]. The other group, based out of the

Korea Atomic Energy and Research Institute (KAERI), developed the somewhat more generally named "2D/1D" method in DeCART [4, 21, 24]. The primary difference is that "2D/1D Fusion" maintains full angular dependence of the transverse leakages using transport physics axially, while the "2D/1D" method neglects most if not all of the angular information and uses diffusion physics axially. Additionally, KAIST also developed a capability that performed axial solves on a subpin level basis instead of the homogenized pin basis [33]. Though the "2D/1D Fusion" method asymptotically approaches the 3D transport solution, more attention has been given to the method developed at KAERI because it has been found to be sufficiently accurate at a reasonable computational cost. However, there has been renewed interest in the "2D/1D Fusion" method, as well as methods that bridge the accuracy gap between them. While the work here focuses on improving the angular representation, it also sets up future development to approach the spatial leakage approximations being made.

The DeCART code, which began as part of a project through the International Nuclear Energy Research Initiative (INERI), included collaboration between KAERI, Argonne National Laboratory (ANL), and Purdue University. This effort was eventually split with one version owned by ANL [46] and another by the University of Michigan (UM). In the early versions of DeCART, nodal diffusion solvers (NEM and SENM) were primarily used [24]. Since then, using an axial SP_N solver has become more standard [49, 50, 54] since these methods provide improvements over diffusion-based kernels. However, work by Hursin et. al. [20] added an axial NEM-S_N solver that used azimuthally-integrated radial transverse leakages. This is the starting point for this work, which extends the radial and axial transverse leakages to include both polar and azimuthal dependence and is similar to the pin-homogenized "2D/1D Fusion" approach. While including azimuthal dependence is a great step, the novel part of this work is concentrated on representing the azimuthal dependence with a Fourier series, which can lead to the same order of accuracy, but at a fraction the computational cost.

Eventually, it was decided that the development of DeCART at UM would end, and a new 2D/1D implementation would be developed in MPACT [25, 26, 31, 54], which is now being developed collaboratively between the University of Michigan and Oak Ridge National Laboratory (ORNL). Work with DeCART at ANL also continued and similar concepts were extended to the PROTEUS-MOC code as well, which uses a finite element-based axial S_N solver [38]. Additionally, there are currently several other codes being developed internationally that use similar concepts. One being developed at Seoul National University (SNU), nTRACER [42, 49, 50], is very similar to MPACT, using an SP_N axial solver on pin-homogenized cross sections. Another noteworthy project, MICADO [13],

being developed at Électricité de France (EDF), is more in line with the "2D/1D Fusion" concept, performing axial solves on subpin level regions.

1.3 Dissertation Layout

Chapter 2 describes the basics of transport theory followed by background on the state-of-the-art numerical methods that are currently being employed in reactor analysis. The theory presented starts from the 3D linear Boltzmann transport equation and walks through several different approximations that are commonly made, particularly in deterministic transport methods. These approximations give way to discussions on the numerical methods, such the method of characteristics (MOC) and nodal methods (NEM, SENM, SP_N) that are being used within the 2D/1D framework. Also briefly mentioned are Monte Carlo (probabilistic) methods, which are the highest fidelity transport solvers available and are often used to generate reference solutions for code-to-code comparisons.

With these fundamentals in hand, Chapter 3 covers the 2D/1D framework and provides a better description of how the radial and axial solvers are connected through transverse leakages. This chapter also includes some details on the iteration strategy as well as the relaxation technique that has been an important component of the 2D/1D implementation thus far. To better understand some of the trends that are observed in the results, an overview of the sources of error is provided, which includes sources that are specific to the 2D/1D scheme, but also some that are present even in 2D-MOC cases. The parallel decomposition schemes currently implemented using MPI and OpenMP are described, concluding the chapter.

In Chapter 4, the higher fidelity axial S_N solver is presented, including both the spatial and angular expansions it employs. Additionally, several MOC sweepers have been implemented to account for the angular expression for the axial transverse leakage. This is particularly useful when Fourier moments are used to construct the axial transverse leakage, as now the storage for the radial solver can also be expressed as a Fourier series. In the most intuitive approach to using this, the expansion is evaluated every time it needs to be used. This is actually quite slow, and an alternative solver that requires slightly more storage is proposed. The rest of this chapter presents information on the parallelization and inner iteration strategy. Though none of the results presented here use it, a brief explanation of how anisotropic scattering can fit into this scheme is also shown. This chapter closes with an assessment of what should be expected asymptotically as parameters are refined, both in regards to the axial and radial mesh as well as the number of Fourier moments.

Chapter 5 presents the results of several benchmark cases including the Takeda Rodded

LWR case and the 3D-C5G7 extended cases, which have control rods and MOX fuel. A variant of the C5G7 cases has also been evaluated using homogenized pins, highlighting one of the remaining approximations being made: the shape of the axial transverse leakage. Additionally, the memory requirements and parallel performance are assessed. While providing more accurate results is very important, it is equally important that the accuracy gained comes at a reasonable price.

Lastly, Chapter 6 outlines the conclusions and lessons learned from this work and highlights several areas of future research.

CHAPTER 2

Neutron Transport Methodology

2.1 Transport Theory

This chapter is dedicated to covering the basic theory of neutron transport as well as the numerical methods used to simulate it, particularly the methods used in the existing 2D/1D scheme. Most of the fundamentals of theory are very well documented in various textbooks (such as the Nuclear Engineering Handbook [47]), and there are several good texts covering numerical methods [10, 17, 34], so the descriptions will be more condensed to provide a quick look at each. The linear Boltzmann transport equation will be the starting point for the theory and several commonly used approximations will be introduced, especially those used in deterministic transport methods.

2.1.1 The Boltzmann Transport Equation

The Boltzmann transport equation (Eq. 2.1a) can be used to describe the interaction of neutrons in a system, including production, scattering, streaming, and absorption. In this equation, there are seven independent variables: three spatial variables (x, y, z as in Eq. 2.1b), two angular variables (α, μ as in Eq. 2.1c), energy, and time:

$$\begin{aligned}\frac{1}{v}\frac{\partial\varphi}{\partial t}(\boldsymbol{x},\boldsymbol{\Omega},E,t)+\boldsymbol{\Omega}\cdot\nabla\varphi(\boldsymbol{x},\boldsymbol{\Omega},E,t)+\Sigma_t(\boldsymbol{x},E)\varphi(\boldsymbol{x},\boldsymbol{\Omega},E,t)=\\ \int_0^\infty\int_0^{4\pi}\Sigma_s(\boldsymbol{x},\boldsymbol{\Omega'}\cdot\boldsymbol{\Omega},E'\to E)\varphi(\boldsymbol{x},\boldsymbol{\Omega'},E',t)d\Omega' dE'\\ +\frac{\chi_p(\boldsymbol{x},E)}{4\pi}\int_0^\infty\int_0^{4\pi}[1-\beta(E')]\nu\Sigma_f(\boldsymbol{x},E')\varphi(\boldsymbol{x},\boldsymbol{\Omega'},E',t)d\Omega' dE'\\ +\sum_{j=1}^{N_{pg}}\frac{\chi_j(\boldsymbol{x},E)}{4\pi}\lambda_j C_j(\boldsymbol{x},t)+\frac{Q(\boldsymbol{x},E,t)}{4\pi},\end{aligned} \tag{2.1a}$$

$$\boldsymbol{x}=x\boldsymbol{i}+y\boldsymbol{j}+z\boldsymbol{k}, \tag{2.1b}$$

$$\begin{aligned}
\boldsymbol{\Omega} &= \Omega_x \boldsymbol{i} + \Omega_y \boldsymbol{j} + \Omega_z \boldsymbol{k}, \\
\Omega_x &= \sqrt{1-\mu^2}cos(\alpha), \\
\Omega_y &= \sqrt{1-\mu^2}sin(\alpha), \\
\Omega_z &= \mu.
\end{aligned} \tag{2.1c}$$

In Eq. 2.1c, α denotes the azimuthal angle, which is considered to be the angle of $\boldsymbol{\Omega}$ projected onto the $x-y$ plane with respect to the x-axis, and μ denotes cosine of the polar angle (θ), or the angle formed with respect to the z-axis (Fig. 2.1).

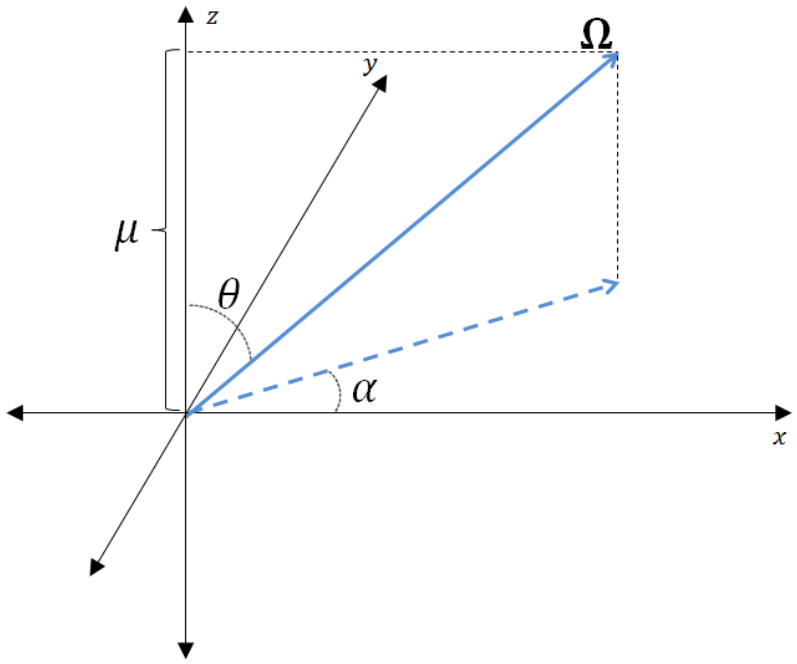

Figure 2.1: Definition of Azimuthal (α) and Polar (μ) Angles

Each term in Eq. 2.1a will briefly be explained, starting with the term that describes the rate of change of neutrons in an energy dE about E, angle $d\Omega$ about $\boldsymbol{\Omega}$, and dV about $\boldsymbol{x}$:

$$\frac{1}{v}\frac{\partial \varphi}{\partial t}(\boldsymbol{x}, \boldsymbol{\Omega}, E, t) dV d\Omega dE.$$

The streaming operator describes the rate neutrons travel into $d\Omega$ about $\boldsymbol{\Omega}$, dE about E, and dV about $\boldsymbol{x}$ at time t:

$$\boldsymbol{\Omega} \cdot \nabla \varphi(\boldsymbol{x}, \boldsymbol{\Omega}, E, t) dV d\Omega dE.$$

The collision term describes rate at which neutrons in $dVd\Omega dE$ about $(\boldsymbol{x}, \boldsymbol{\Omega}, E)$ undergo

collisions:

$$\Sigma_t(\boldsymbol{x}, E)\varphi(\boldsymbol{x}, \boldsymbol{\Omega}, E, t)dVd\Omega dE.$$

The scattering source describes the rate at which neutrons in dV about $\boldsymbol{x}$ scatter into $d\Omega$ about $\boldsymbol{\Omega}$ and dE about E at time t. In the basis concept of the scattering, one would consider a neutron in $\boldsymbol{\Omega'}$ scattering into $\boldsymbol{\Omega}$. However, because of rotational symmetry, this can be simplified as $\boldsymbol{\Omega'} \cdot \boldsymbol{\Omega}$:

$$\left(\int_0^\infty \int_0^{4\pi} \Sigma_s(\boldsymbol{x}, \boldsymbol{\Omega'} \cdot \boldsymbol{\Omega}, E' \to E)\varphi(\boldsymbol{x}, \boldsymbol{\Omega'}, E', t)d\Omega' dE'\right)dVd\Omega dE.$$

The prompt fission source describes the production rate of prompt neutrons via fission. Prompt neutrons denote neutrons that are immediately produced after a fission event, which is a vast majority. β defines the probability that a fission neutron is produced by a delayed neutron precursor and is typically less than 1%, but varies from isotope to isotope [44]:

$$\left(\frac{\chi_p(\boldsymbol{x}, E)}{4\pi} \int_0^\infty \int_0^{4\pi} [1 - \beta(E')]\nu\Sigma_f(\boldsymbol{x}, E')\varphi(\boldsymbol{x}, \boldsymbol{\Omega'}, E', t)d\Omega' dE'\right)dVd\Omega dE.$$

The delayed neutron source describes the rate at which delayed neutrons from precursors (C) are emitted. A commonly used number of precursor groups (N_{pg}) is six. Precursors are grouped based on the decay half-life, and an average decay constant for the group (λ_j) is used for all precursors in the group. Each group also has a different fission spectrum (χ_j):

$$\left(\sum_{j=1}^{N_{pg}} \frac{1}{4\pi}\chi_j(\boldsymbol{x}, E)\lambda_j C_j(\boldsymbol{x}, t)\right)dVd\Omega dE,$$

where the time-dependent precursor concentration (C_j) can be determined using Eq. 2.2:

$$\frac{\partial C_j}{\partial t}(\boldsymbol{x}, t) + \lambda_j C_j(\boldsymbol{x}, t) = \int_0^\infty \int_{4\pi} \beta_j(E')\nu\Sigma_f(E')\varphi(\boldsymbol{x}, \boldsymbol{\Omega'}, E', t)d\Omega' dE'. \tag{2.2}$$

Lastly, the external source describes the rate at which neutrons are born by any other means that does not fit into the streaming, scattering, or fission sources:

$$\frac{Q(\boldsymbol{x}, E, t)}{4\pi}.$$

Several approximations are made in formulating the linear Boltzmann transport equation-such as neglecting quantum physics properties (spin, polarization, etc.) and

neutron-neutron interactions (which is reasonable given typical flux levels in reactors). This list is not exhaustive, but it should be noted that some approximations are made, even with the most explicit, linear form that is used for reactor physics applications.

2.1.2 Steady-State Transport Equation

The class of problems assessed in this work has no time dependence. Thus, all of the time derivatives are zero, and the precursor concentration is static. Eq. 2.3 shows the steady-state eigenvalue form of the transport equation with the transient terms removed, meaning that Eq. 2.2 becomes:

$$\lambda_j C_j(\boldsymbol{x},t) = \int_0^\infty \int_{4\pi} \beta_j(E')\nu\Sigma_f(E')\varphi(\boldsymbol{x},\boldsymbol{\Omega}',E',t)d\Omega' dE'.$$

Substituting this back into Eq. 2.1a yields the first form of the steady-state transport equation under consideration:

$$\begin{aligned}
&\boldsymbol{\Omega}\cdot\nabla\varphi(\boldsymbol{x},\boldsymbol{\Omega},E)+\Sigma_t(\boldsymbol{x},E)\varphi(\boldsymbol{x},\boldsymbol{\Omega},E) = \\
&\int_0^\infty \int_0^{4\pi} \Sigma_s(\boldsymbol{x},\boldsymbol{\Omega}'\cdot\boldsymbol{\Omega},E'\rightarrow E)\varphi(\boldsymbol{x},\boldsymbol{\Omega}',E')d\Omega' dE' \\
&\quad + \frac{\chi_p(\boldsymbol{x},E)}{4\pi}\int_0^\infty \int_0^{4\pi} [1-\beta(E')]\nu\Sigma_f(\boldsymbol{x},E')\varphi(\boldsymbol{x},\boldsymbol{\Omega}',E')d\Omega' dE' \\
&\quad + \sum_{j=1}^{N_{pg}} \frac{\chi_j(\boldsymbol{x},E)}{4\pi}\int_0^\infty \int_{4\pi} \beta_j(E')\nu\Sigma_f(E')\varphi(\boldsymbol{x},\boldsymbol{\Omega}',E')d\Omega' dE' + \frac{Q(\boldsymbol{x},E)}{4\pi}.
\end{aligned} \tag{2.3}$$

However, it is a common approximation to neglect the fission spectrum dependence of the delayed neutron precursor groups, since the total delayed neutron production is typically less that 1% of the total. With this assumption, the equation can be further simplified (Eq. 2.4):

$$\begin{aligned}
&\boldsymbol{\Omega}\cdot\nabla\varphi(\boldsymbol{x},\boldsymbol{\Omega},E)+\Sigma_t(\boldsymbol{x},E)\varphi(\boldsymbol{x},\boldsymbol{\Omega},E) = \\
&\quad \int_0^\infty \int_0^{4\pi} \Sigma_s(\boldsymbol{x},\boldsymbol{\Omega}'\cdot\boldsymbol{\Omega},E'\rightarrow E)\varphi(\boldsymbol{x},\boldsymbol{\Omega}',E')d\Omega' dE' \\
&\qquad + \frac{\chi(\boldsymbol{x},E)}{4\pi}\int_0^\infty \int_0^{4\pi} \nu\Sigma_f(\boldsymbol{x},E')\varphi(\boldsymbol{x},\boldsymbol{\Omega}',E')d\Omega' dE' + \frac{Q(\boldsymbol{x},E)}{4\pi}.
\end{aligned} \tag{2.4}$$

2.1.3 The Multigroup Approximation

The next approximation to the transport equation that will be considered is the multigroup approximation, which is a means of discretizing the energy variable (E) into energy groups. Figure 2.2 shows the energy dependence of the total cross section for U-235 [28], a common isotope in nuclear fuel. As can be seen, there are extremely large variations in the cross section value, making the multigroup approximation particularly difficult.

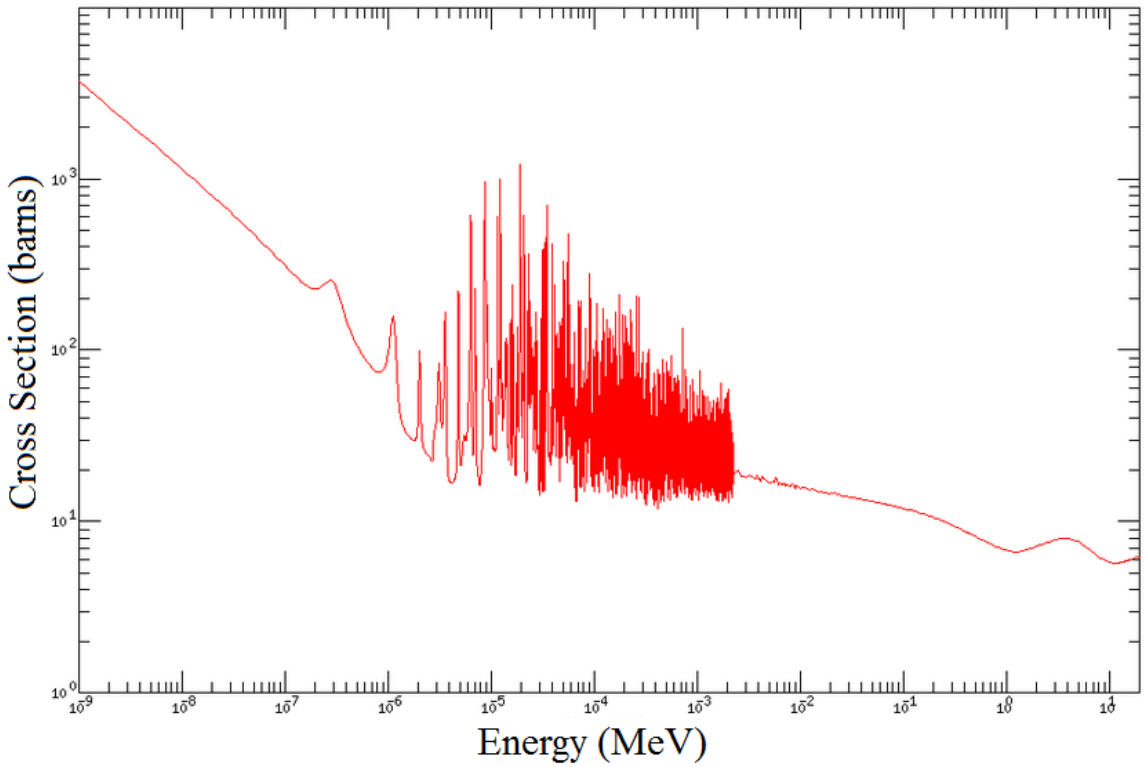

Figure 2.2: Total Cross Section ($\Sigma_T(E)$) of U-235

Though probabilistic (Monte Carlo) transport methods, which will be covered later in this chapter, can continuously represent this energy dependence, deterministic transport methods discretize all variables, including energy. Typically, the groups are indexed such that the indexing goes from highest/fastest energy ($g = 1$) to lowest/slowest energy ($g = N_{grp}$). The multigroup values are obtained by integrating energy over the bounds for each group (as in Eqs. 2.5):

$$\varphi_g(\boldsymbol{x},\boldsymbol{\Omega}) = \int_{E_g}^{E_{g-1}} \varphi(\boldsymbol{x},\boldsymbol{\Omega},E)dE, \tag{2.5a}$$

$$\chi_g(\boldsymbol{x},\boldsymbol{\Omega}) = \int_{E_g}^{E_{g-1}} \chi(\boldsymbol{x},\boldsymbol{\Omega},E)dE, \tag{2.5b}$$

$$Q_g(\boldsymbol{x},\boldsymbol{\Omega}) = \int_{E_g}^{E_{g-1}} Q(\boldsymbol{x},\boldsymbol{\Omega},E)dE, \tag{2.5c}$$

$$\Sigma_{x,g}(\boldsymbol{x},\boldsymbol{\Omega}) = \frac{\int_{E_g}^{E_{g-1}} \Sigma_x(\boldsymbol{x},E)\varphi(\boldsymbol{x},\boldsymbol{\Omega},E)dE}{\int_{E_g}^{E_{g-1}} \varphi(\boldsymbol{x},\boldsymbol{\Omega},E)dE}. \tag{2.5d}$$

However, cross sections typically do not retain angular dependence (as in Eq. 2.5d). A common approximation is to assume that the angular and energy variables are separable:

$$\varphi(\boldsymbol{x},\boldsymbol{\Omega},E) \approx \Phi(\boldsymbol{x},E)\Psi(\boldsymbol{x},\boldsymbol{\Omega}), \tag{2.6}$$

in which case the multigroup cross sections can be defined as angle-independent (except for the scattering cross section):

$$\Sigma_{x,g}(\boldsymbol{x}) \equiv \frac{\int_{E_g}^{E_{g-1}} \Sigma_x(\boldsymbol{x},E)\Phi(\boldsymbol{x},E)dE}{\int_{E_g}^{E_{g-1}} \Phi(\boldsymbol{x},E)dE}. \tag{2.7}$$

Eq. 2.8 shows the full steady-state transport equation with the multigroup approximation:

$$\begin{aligned}\boldsymbol{\Omega}\cdot\nabla\varphi_g(\boldsymbol{x},\boldsymbol{\Omega}) + \Sigma_{t,g}(\boldsymbol{x})\varphi_g(\boldsymbol{x},\boldsymbol{\Omega}) = &\sum_{g'=1}^{N_{grp}} \int_0^{4\pi} \Sigma_{s,g'\to g}(\boldsymbol{x},\boldsymbol{\Omega}'\cdot\boldsymbol{\Omega})\varphi_{g'}(\boldsymbol{x},\boldsymbol{\Omega}')d\Omega' \\ &+\frac{\chi_g(\boldsymbol{x})}{4\pi}\sum_{g'=1}^{N_{grp}} \int_0^{4\pi} \nu\Sigma_{f,g'}(\boldsymbol{x})\varphi_{g'}(\boldsymbol{x},\boldsymbol{\Omega}')d\Omega' + \frac{1}{4\pi}Q_g(\boldsymbol{x}).\end{aligned} \tag{2.8}$$

2.1.4 Scattering Approximations

Up to this point, the scattering sources have been considered to retain full angular dependence. In general, it is not good practice to store the angular fluxes, which are required when constructing this source. Commonly used scattering approximations include isotropic and anisotropic techniques, which use a truncated spherical harmonics expansion. Instead, the scalar flux and angular moments of the scalar flux can be used, allowing the memory burden to be reduced considerably.

2.1.4.1 Anisotropic Scattering

Starting from the multigroup scattering source (as in Eq. 2.8), the simplification can be made that $\boldsymbol{\Omega}'\cdot\boldsymbol{\Omega}$ can be represented by one angular value, which will be denoted as μ_s (Eq. 2.9):

$$\Sigma_{s,g'\to g}(\boldsymbol{x},\boldsymbol{\Omega'}\cdot\boldsymbol{\Omega}) = \Sigma_{s,g'\to g}(\boldsymbol{x},\mu_s), \tag{2.9}$$

where the scattering cross section can be expanded using Legendre polynomials (Eq. 2.10a):

$$\Sigma_{s,g'\to g}(\boldsymbol{x},\mu_s) = \sum_{n=0}^{\infty}\frac{2n+1}{4\pi}\Sigma_{sn}(\boldsymbol{x})P_n(\mu_s), \tag{2.10a}$$

$$\Sigma_{sn,g'\to g}(\boldsymbol{x}) = 2\pi\int_{-1}^{1}P_n(\mu')\Sigma_{s,g'\to g}(\boldsymbol{x},\mu')d\mu'. \tag{2.10b}$$

In reactor problems, this Legendre expansion can typically be truncated at about one to three moments, whereas shielding problems can require higher order representation. Two basic representations are isotropic scattering ($n = 0$, Eq. 2.11a) and linearly anisotropic scattering ($n = 1$, Eq. 2.11b):

$$\Sigma_{s,g'\to g}(\boldsymbol{x},\mu) = \frac{1}{4\pi}\Sigma_{s0,g'\to g}(\boldsymbol{x}), \tag{2.11a}$$

$$\Sigma_{s,g'\to g}(\boldsymbol{x},\mu) = \frac{1}{4\pi}\Sigma_{s0,g'\to g}(\boldsymbol{x}) + \frac{3\mu}{4\pi}\Sigma_{s1,g'\to g}(\boldsymbol{x}). \tag{2.11b}$$

Next, a spherical harmonic expansion for the angular flux term (Eq. 2.12) in the scattering source [18] will be considered:

$$\varphi(\boldsymbol{\Omega}) = \sum_{l=0}^{L}\frac{2l+1}{4\pi}\sum_{m=-1}^{l}\varphi_l^m R_l^m(\boldsymbol{\Omega}). \tag{2.12}$$

Here the real spherical harmonics (R_l^m in Eq. 2.13) can be expressed in terms of the azimuthal (α) and polar (μ) angles:

$$R_l^m(\boldsymbol{\Omega}) = \sqrt{(2-\delta_{m,0})\frac{(l-|m|)!}{(l+|m|)!}}P_l^{|m|}(\mu)T_m(\alpha), \tag{2.13a}$$

$$P_l^m(\mu) = (1-\mu^2)^{\frac{m}{2}}\frac{d^m}{d\mu^m}P_l(\mu), \tag{2.13b}$$

$$T_m(\alpha) = \begin{cases} cos(m\alpha)\ , & m > 0 \\ sin(|m|\alpha)\ , & other. \end{cases} \tag{2.13c}$$

For example, below are the spherical harmonic formulas for the 0^{th} (Eq. 2.14a), 1^{st} (Eq. 2.14b), and 2^{nd} (Eq. 2.14c) moments. Here it is important to note that $R_0^0(\boldsymbol{\Omega})$, $R_1^0(\boldsymbol{\Omega})$, and

$R_2^0(\mathbf{\Omega})$ correspond exactly to the 0^{th}, 1^{st}, and 2^{nd} Legendre Polynomials (P_{0-2}):

$$R_0^0(\mathbf{\Omega}) = 1, \tag{2.14a}$$

$$\begin{aligned} R_1^{-1}(\mathbf{\Omega}) &= \xi = \sqrt{1-\mu^2}cos(\alpha), \\ R_1^0(\mathbf{\Omega}) &= \mu, \\ R_1^1(\mathbf{\Omega}) &= \eta = \sqrt{1-\mu^2}sin(\alpha), \end{aligned} \tag{2.14b}$$

$$\begin{aligned} R_2^{-2}(\mathbf{\Omega}) &= \sqrt{\frac{3}{4}}(1-\mu^2)sin(2\alpha, \\ R_2^{-1}(\mathbf{\Omega}) &= \sqrt{3}\mu\sqrt{1-\mu^2}cos(\alpha), \\ R_2^0(\mathbf{\Omega}) &= \frac{1}{2}(3\mu^2-1), \\ R_2^1(\mathbf{\Omega}) &= \sqrt{3}\mu\sqrt{1-\mu^2}sin(\alpha), \\ R_2^2(\mathbf{\Omega}) &= -\sqrt{\frac{3}{4}}(1-\mu^2)cos(2\alpha). \end{aligned} \tag{2.14c}$$

This is realized in implementation by storing φ_l^m for each region, where φ_0^0 is the scalar flux (which is usually denoted as ϕ in this document) and $\varphi_0^0 = \phi = \int_0^{4\pi} \varphi(\mathbf{\Omega})d\Omega$. Table 2.1 shows the storage requirements for each scattering order as a function of the problem dimension. For purely 1D problems, the azimuthal dependence can be neglected, whereas for 2D problems, terms without azimuthal dependence can be omitted (i.e. the axial currents or flux moments are zero). However, 1D solvers that take in information pertaining to both the polar and azimuthal dependence are essentially pseudo-3D solvers, and require the same storage as a full 3D solver.

Table 2.1: Number of Flux Moments Stored for Anisotropic Scattering

Scattering Order	**Problem Dimension**		
	1D	**2D**	**3D**
0	1	1	1
1	2	3	4
2	3	7	9
3	4	13	16

2.1.4.2 Transport-Corrected Scattering

Another common approximation uses the isotropic scattering kernel, but modifies the self-scattering ($\Sigma_{s0,g\to g}$) in attempt to preserve some of the physics of linearly anisotropic scattering [60]. This class of approximations will be referred to as transport-corrected P0 scattering (TCP0).

The simplest of these is the out-scatter approximation, which subtracts the sum of the first order scattering data from the total cross section and zeroth order self-scattering (Eq. 2.15).

$$\Sigma_{tr,g} = \Sigma_{t,g} - \sum_{g'=1}^{N_{grp}} \Sigma_{s1,g\to g'}, \tag{2.15a}$$

$$\Sigma_{s0,g\to g} = \Sigma_{s0,g\to g} - \sum_{g'=1}^{N_{grp}} \Sigma_{s1,g\to g'}. \tag{2.15b}$$

In MPACT, this approximation is commonly used and also includes variants to help avoid negative cross sections. As can be seen in Eq. 2.15b, if the first order scattering data is large enough (as is the case with lighter elements, such as hydrogen), it is possible to yield negative self-scattering cross sections. This can pose an issue, particularly for moderator/reflector regions which may not have a fission source or sufficient in-scattering sources to maintain positive total sources.

Two variants to this have been considered. The first will be referred to as the limited transport correction (LTCP0), which removes the negative scattering portion in groups greater than 1 MeV. That is, it limits the amount of correction to maintain non-negative cross sections. The 1 MeV threshold is chosen under the assumption that below 1 MeV, there should be adequate in-scatter sources from higher groups to maintain positivity. The second variant is an extension of the limited transport correction but lowers the threshold to 0 MeV, so all groups are guaranteed to be positive. This variant is referred to as the fully limited transport correction (FLTCP0).

Both LTCP0 and FLTCP0 sacrifice preserving the maximum P_1 physics to gain positivity and stability. That being said, LTCP0 can generally be used without influencing power distributions too severely, but FLTCP0 is generally only used in situations where the assumptions of LTCP0 are not valid, as it can detrimentally change the power shape. FLTCP0 provides only slightly better accuracy than isotropic (P0) scattering. For larger problems, particularly those with large reflector regions (or axial reflectors, where entire planes are reflector material), transport-correction and its variants seem to struggle with

stability. For this reason, most full-core simulations in MPACT use explicit anisotropic scattering treatment.

2.1.5 The Discrete Ordinates Approximation

Similar to the multigroup approximation, the angular variables contained in $\mathbf{\Omega}$ can also be discretized to form what are known as discrete ordinates:

$$\begin{aligned}\mathbf{\Omega} &= \Omega_x\mathbf{i} + \Omega_y\mathbf{j} + \Omega_z\mathbf{k} \\ &= \cos(\alpha)\sqrt{1-\mu^2}\mathbf{i} + \sin(\alpha)\sqrt{1-\mu^2}\mathbf{j} + \mu\mathbf{k}.\end{aligned} \tag{2.16a}$$

In Eq. 2.16a, both the azimuthal (α) and polar angles μ can be discretized, as in Eq. 2.16b, where the angle index in indicated by l:

$$\mathbf{\Omega}_l = \cos(\alpha_l)\sqrt{1-\mu_l^2}\mathbf{i} + \sin(\alpha_l)\sqrt{1-\mu_l^2}\mathbf{j} + \mu_l\mathbf{k}. \tag{2.16b}$$

Eq. 2.17 shows the transport equation with both the azimuthal and polar angle discretized. Note the change in the angular flux variables, which now have the angular index as a subscript:

$$\begin{aligned}\mathbf{\Omega}_l \cdot \nabla\varphi_{g,l}(\boldsymbol{x}) + \Sigma_{t,g}(\boldsymbol{x})\varphi_{g,l}(\boldsymbol{x}) = &\sum_{g'=1}^{N_{grp}}\sum_{l'=1}^{N_{ang}} w_{l'}\Sigma_{s,g'\to g,l'\to l}(\boldsymbol{x})\varphi_{g',l'}(\boldsymbol{x}) \\ &+\frac{\chi_g(\boldsymbol{x})}{4\pi}\sum_{g'=1}^{N_{grp}}\sum_{l'=1}^{N_{ang}} w_{l'}\nu\Sigma_{f,g'}(\boldsymbol{x})\varphi_{g',l'}(\boldsymbol{x}) + \frac{1}{4\pi}Q_g(\boldsymbol{x}).\end{aligned} \tag{2.17}$$

Additionally, the scalar flux, which can be determined by integrating the angular flux, can now be obtained via summation (Eq. 2.18), where w_i is the weight of the angle and $\sum_{l=1}^{N_{ang}} w_l = 4\pi$:

$$\phi_g = \int_0^{4\pi}\varphi_g(\alpha,\mu)d\Omega = \sum_{l=1}^{N_{ang}} w_l\varphi_{g,l}. \tag{2.18}$$

2.1.5.1 Quadrature Sets

There are essentially two kinds of quadratures that are typically used to discretize the angular variables: level-symmetric and product quadratures. In this work, product quadratures are exclusively used, though both types will be briefly explained.

With the level-symmetric quadrature, the number of azimuthal angles corresponding to

each polar angle in the quadrature is allowed to vary. Additionally, the azimuthal angles are not guaranteed to be consistent with any of the other azimuthal angles at different polar indexes, but the quadrature is rotationally symmetric. Figure 2.3 shows an example octant from a level-symmetric quadrature, which can be seen to have four azimuthal angles on the largest polar angle and one azimuthal angle on the smallest polar angle. It should be noted that the figure is only intended to denote the angles, not the weights.

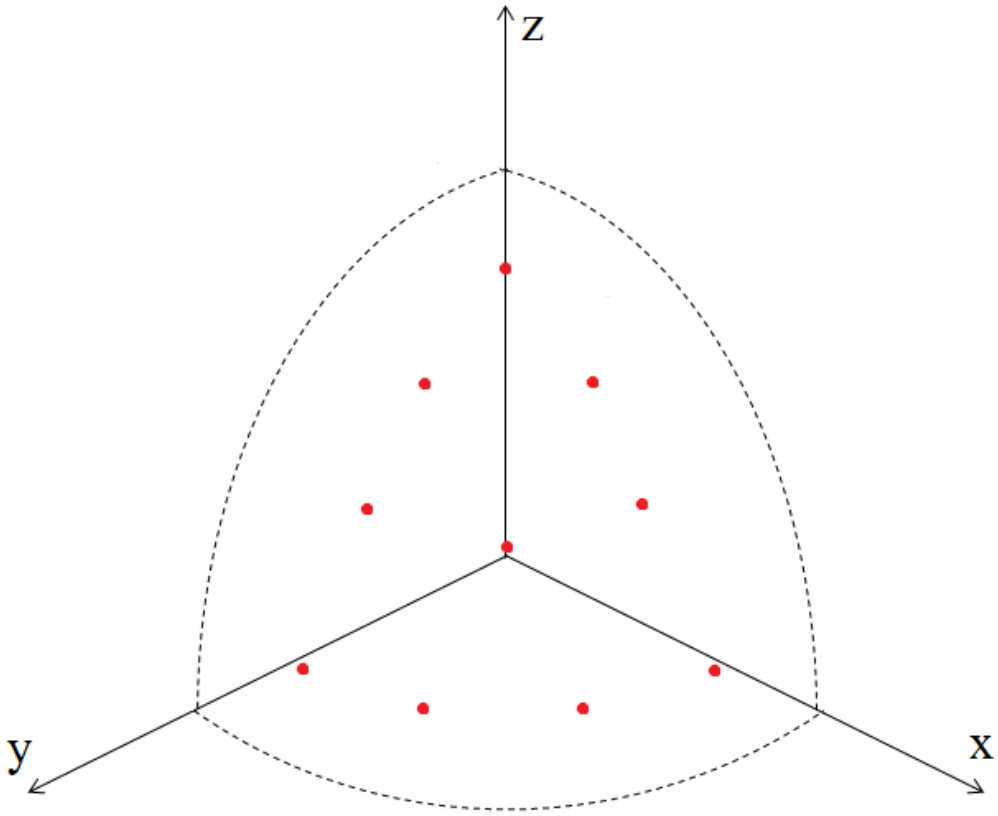

Figure 2.3: Representative Level-Symmetric Quadrature

In the product quadrature, each polar angle has exactly the same azimuthal angles. It is called product quadrature because the number of angles corresponds to the product of the number of azimuthal and polar angles. Figure 2.4 shows an example octant from a production quadrature. Note that each polar angle contains the same four azimuthal angles. For simplicity, when an angle at index l is considered, it is not just referring to azimuthal index l or polar index l, which in this cases there would both be four total. In this work, l denotes the combination of the azimuthal and polar angles in Ω_l, of which there are sixteen in the figure below. While some indexes will have the same polar or azimuthal angle, none will have both angles match.

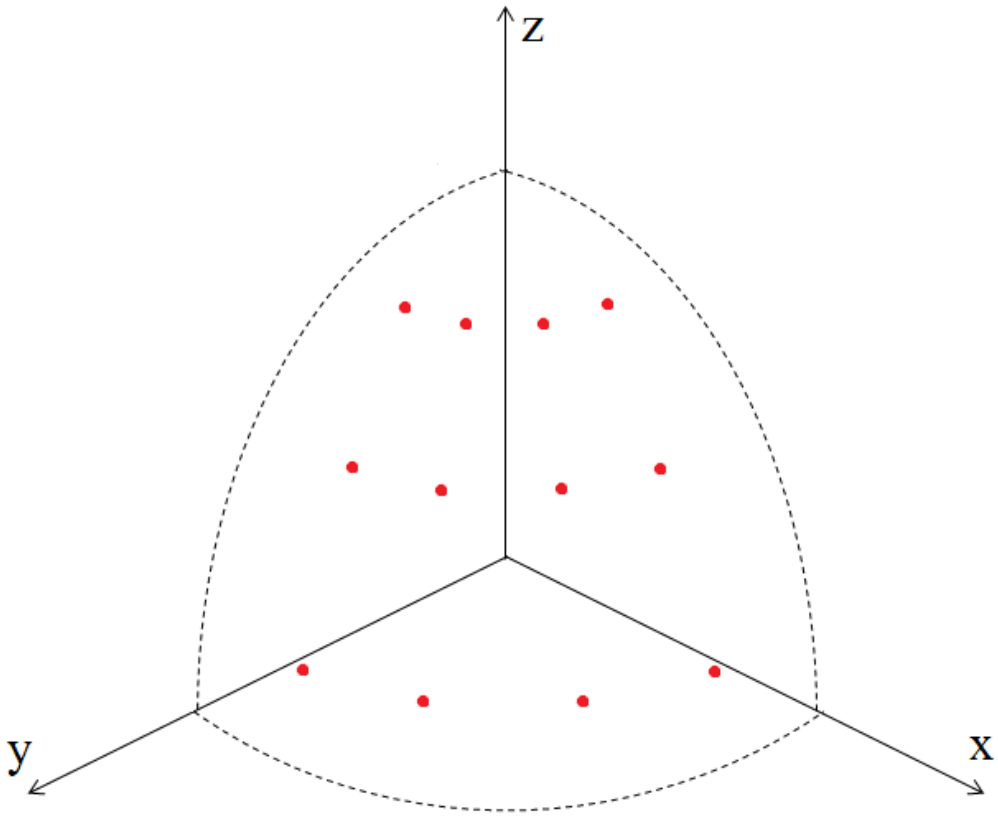

Figure 2.4: Representative Product Quadrature

Typically, the azimuthal angles, which range from 0 to 2π, are in a Chebyshev quadrature (i.e. equispaced). However, polar angle quadratures have a bit more complexity. Though they can also use a Chebyshev quadrature, generally a large number of polar angles are necessary for sufficient accuracy. To help reduce this, several other quadratures [32, 61] have been developed that reduce the number of polar angles necessary.

2.1.6 The Diffusion Approximation

The diffusion approximation makes use of a linear approximation of the angular flux, yielding Fick's Law to define the relation between the current and scalar flux. To quickly derive this, by starting from the steady-state, multigroup transport equation (Eq. 2.8), assuming linearly anisotropic scattering, and integrating over $\mathbf{\Omega}$, the zeroth moment of

the transport equation can be obtained (Eq. 2.19):

$$\frac{\partial}{\partial x}J_{x,g}(\boldsymbol{x})+\frac{\partial}{\partial y}J_{y,g}(\boldsymbol{x})+\frac{\partial}{\partial z}J_{z,g}(\boldsymbol{x})+\Sigma_{t,g}(\boldsymbol{x})\phi_g(\boldsymbol{x})= \sum_{g'=1}^{N_{grp}}\Sigma_{s0,g'\rightarrow g}(\boldsymbol{x})\phi_{g'}(\boldsymbol{x})+\chi_g(\boldsymbol{x})\sum_{g'=0}^{N_{grp}}\nu\Sigma_{f,g'}\phi_{g'}(\boldsymbol{x})+Q_g(\boldsymbol{x}). \tag{2.19}$$

Similarly, if Eq. 2.8 is multiplied by Ω_x, Ω_y, and Ω_z and integrated over $\boldsymbol{\Omega}$, the first moment equations are obtained (Eq. 2.20):

$$\frac{1}{3}\frac{\partial}{\partial x}\phi_g(\boldsymbol{x})+\Sigma_{t,g}(\boldsymbol{x})J_{x,g}(\boldsymbol{x})=\sum_{g'=1}^{N_{grp}}\Sigma_{s1,g'\rightarrow g}(\boldsymbol{x})J_{x,g'}(\boldsymbol{x}), \tag{2.20a}$$

$$\frac{1}{3}\frac{\partial}{\partial y}\phi_g(\boldsymbol{x})+\Sigma_{t,g}(\boldsymbol{x})J_{y,g}(\boldsymbol{x})=\sum_{g'=1}^{N_{grp}}\Sigma_{s1,g'\rightarrow g}(\boldsymbol{x})J_{y,g'}(\boldsymbol{x}), \tag{2.20b}$$

$$\frac{1}{3}\frac{\partial}{\partial z}\phi_g(\boldsymbol{x})+\Sigma_{t,g}(\boldsymbol{x})J_{z,g}(\boldsymbol{x})=\sum_{g'=1}^{N_{grp}}\Sigma_{s1,g'\rightarrow g}(\boldsymbol{x})J_{z,g'}(\boldsymbol{x}). \tag{2.20c}$$

In Eq. 2.20, it can be seen that the current still cannot be determined directly. However, by approximating the first order scattering matrix (Σ_{s1}) as a diagonal matrix by collapsing the cross sections ($\tilde{\Sigma}_{s1,g\rightarrow g}=\sum_{g'=1}^{N_{grp}}\Sigma_{s1,g'\rightarrow g}$), the currents can now be easily found:

$$\frac{1}{3}\frac{\partial}{\partial x}\phi_g(\boldsymbol{x})+\Sigma_{t,g}(\boldsymbol{x})J_{x,g}(\boldsymbol{x})=\tilde{\Sigma}_{s1,g\rightarrow g}(\boldsymbol{x})J_{x,g}(\boldsymbol{x}), \tag{2.21a}$$

$$\frac{1}{3}\frac{\partial}{\partial y}\phi_g(\boldsymbol{x})+\Sigma_{t,g}(\boldsymbol{x})J_{y,g}(\boldsymbol{x})=\tilde{\Sigma}_{s1,g\rightarrow g}(\boldsymbol{x})J_{y,g}(\boldsymbol{x}), \tag{2.21b}$$

$$\frac{1}{3}\frac{\partial}{\partial z}\phi_g(\boldsymbol{x})+\Sigma_{t,g}(\boldsymbol{x})J_{z,g}(\boldsymbol{x})=\tilde{\Sigma}_{s1,g\rightarrow g}(\boldsymbol{x})J_{z,g}(\boldsymbol{x}), \tag{2.21c}$$

Solving for the current in Eq. 2.20, the Fick's Law expression relating the current and scalar flux is found in Eq. 2.22:

$$\begin{aligned}\boldsymbol{J}_g(\boldsymbol{x}) &= -\frac{1}{3(\Sigma_{t,g}(\boldsymbol{x})-\tilde{\Sigma}_{s1,g\rightarrow g}(\boldsymbol{x}))}\boldsymbol{\nabla}\phi_g(\boldsymbol{x}) \\ &= -\frac{1}{3\Sigma_{tr,g}(\boldsymbol{x})}\boldsymbol{\nabla}\phi_g(\boldsymbol{x}) \\ &= -D_g(\boldsymbol{x})\boldsymbol{\nabla}\phi_g(\boldsymbol{x}).\end{aligned} \tag{2.22}$$

Substituting this into streaming term of the zeroth moment equation yields the

multigroup, steady-state diffusion equation (Eq. 2.23):

$$-\boldsymbol{\nabla}\cdot D_g(\boldsymbol{x})\boldsymbol{\nabla}\phi_g(\boldsymbol{x})+\Sigma_{t,g}\phi_g(\boldsymbol{x})=\sum_{g'=1}^{N_{grp}}\Sigma_{s0,g'\rightarrow g}(\boldsymbol{x})\phi'_g(\boldsymbol{x})+\chi_g(\boldsymbol{x})\sum_{g'=0}^{N_{grp}}\nu\Sigma_{f,g'}(\boldsymbol{x})\phi_g(\boldsymbol{x})+Q_g(\boldsymbol{x}). \tag{2.23}$$

2.1.7 Constant Material Properties

In the next section, the numerical methods used to solve the transport equation will be presented. With many of these methods, the transport equation describes behavior within a single mesh region, in which it is typical to assume that there is only one material and that the cross sections of this material are constant spatially. Eqs. 2.24 and 2.25 show the simplifications made to the steady state transport and diffusion equations, respectively:

$$\begin{aligned}\boldsymbol{\Omega}_l\cdot\nabla\varphi_{g,l}(\boldsymbol{x})+\Sigma_{t,g}\varphi_{g,l}(\boldsymbol{x})=&\sum_{g'=1}^{N_{grp}}\sum_{l'=1}^{N_{ang}}w_{l'}\Sigma_{s,g'\rightarrow g,l'\rightarrow l}\varphi_{g',l'}(\boldsymbol{x})\\&+\frac{\chi_g}{4\pi}\sum_{g'=1}^{N_{grp}}\sum_{l'=1}^{N_{ang}}w_{l'}\nu\Sigma_{f,g'}\varphi_{g',l'}(\boldsymbol{x})+\frac{1}{4\pi}Q_g(\boldsymbol{x}),\end{aligned} \tag{2.24}$$

$$-\boldsymbol{\nabla}\cdot D_g\boldsymbol{\nabla}\phi_g(\boldsymbol{x})+\Sigma_{t,g}\phi_g(\boldsymbol{x})=\sum_{g'=1}^{N_{grp}}\Sigma_{s0,g'\rightarrow g}\phi'_g(\boldsymbol{x})+\chi_g\sum_{g'=0}^{N_{grp}}\nu\Sigma_{f,g'}\phi_g(\boldsymbol{x})+Q_g(\boldsymbol{x}). \tag{2.25}$$

2.1.8 Eigenvalue Problems

For steady-state, reactor problems the external source term ($Q(\boldsymbol{x})$) is generally neglected and the fission source is multiplied by a factor of $\frac{1}{k_{\text{eff}}}$ where k_{eff} is the criticality of the system (also referred to as the eigenvalue). This is a very important value as it describes how the neutron population in the system is changing. If k_{eff} is less than 1, it means the system is subcritical and the fission source is increased to balance the production and loss (leakage and capture) in the system. If k_{eff} is greater than 1, then the system is supercritical and the fission source is decreased to provide balance. Eq. 2.26 shows the transport equation with constant material properties transformed into an eigenvalue problem.

$$\boldsymbol{\Omega}_l\cdot\nabla\varphi_{g,l}(\boldsymbol{x})+\Sigma_{t,g}\varphi_{g,l}(\boldsymbol{x})=\sum_{g'=1}^{N_{grp}}\sum_{l'=1}^{N_{ang}}w_{l'}\Sigma_{s,g'\rightarrow g,l'\rightarrow l}\varphi_{g',l'}(\boldsymbol{x})+\frac{\chi_g}{4\pi k_{\text{eff}}}\sum_{g'=1}^{N_{grp}}\sum_{l'=1}^{N_{ang}}w_{l'}\nu\Sigma_{f,g'}\varphi_{g',l'}(\boldsymbol{x}), \tag{2.26}$$

2.2 Fundamental Numerical Methods

2.2.1 Method of Characteristics (MOC)

Though the method of characteristics can be applied to both 1D and 3D problems [29, 30, 36, 58], this section focuses on MOC as it is used to solve 2D transport problems. Eq. 2.27a shows the 2D transport equation with the scattering, fission, and external sources lumped into a general source quantity (Q). Eq. 2.27b shows the same equation with the streaming operator expanded into x and y:

$$\mathbf{\Omega} \cdot \mathbf{\nabla} \varphi(x, y, \mathbf{\Omega}) + \Sigma_t(x, y) \varphi(x, y, \mathbf{\Omega}) = Q(x, y), \tag{2.27a}$$

$$\sqrt{1-\mu^2}\cos(\alpha)\frac{\partial \varphi(x, y, \alpha, \mu)}{\partial x} + \sqrt{1-\mu^2}\sin(\alpha)\frac{\partial \varphi(x, y, \alpha, \mu)}{\partial y} + \Sigma_t(x, y)\varphi(x, y, \alpha, \mu) = Q(x, y). \tag{2.27b}$$

Projecting this along a characteristic direction, the two dimensional partial differential equation becomes a one-dimensional ordinary differential equation (Eq. 2.28a) with an incoming angular flux boundary condition (Eq. 2.28b), which can be solved analytically for the flux at any point along the characteristic direction (Eq. 2.29). Here the constant material properties simplification has also been made:

$$\frac{\partial \varphi}{\partial s} + \Sigma_t \varphi(s) = Q, \tag{2.28a}$$

$$\varphi(0) = \varphi_{in}, \tag{2.28b}$$

$$\varphi(s) = \varphi_{in} e^{-\Sigma_t s} + \frac{Q}{\Sigma_t}\left(1 - e^{Sigma_t s}\right). \tag{2.29}$$

With Eq. 2.29, expressions for the outgoing (Eq. 2.30b) and average (Eq. 2.30c) angular fluxes can be found. The outgoing flux is determined by substituting the segment length (which is a function of polar angle, Eq. 2.30a) into Eq. 2.29, where t is the track length projected onto the X-Y plane:

$$l = \frac{t}{\sqrt{1-\mu^2}}, \tag{2.30a}$$

$$\varphi_{out} = \varphi(l) = \varphi_{in} e^{-\Sigma_t l} + \frac{Q}{\Sigma_t}\left(1 - e^{-\Sigma_t l}\right), \tag{2.30b}$$

The average angular flux is obtained by integrating Eq. 2.29 along the entire segment

length:

$$\begin{aligned}\tilde{\varphi} = \frac{1}{l}\int_0^l \varphi(s)ds &= \frac{Q}{\Sigma_t} + \frac{1}{\Sigma_t l}\left(\varphi_{in} - \frac{Q}{\Sigma_t}\right)\left(1 - e^{-\Sigma_t l}\right) \\ &= \frac{Q}{\Sigma_t} + \frac{\varphi_{in} - \varphi_{out}}{\Sigma_t l}.\end{aligned} \tag{2.30c}$$

The scalar flux 2.32 in each region can be calculated as the weighted sum of the average angular fluxes in each region (Eq. 2.31). Here it is important not to confuse the segment length (l_r) with the angular index (l):

$$\overline{\varphi}_l = \frac{\sum_{r=1}^{N_{ray,l}} \delta_r l_r \tilde{\varphi}_{l,r}}{\sum_{r=1}^{N_{ray,l}} \delta_r l_r}, \tag{2.31}$$

$$\overline{\phi_i} = \sum_{l=1}^{N_{ang}} w_l \overline{\varphi}_l. \tag{2.32}$$

To get an idea of how this looks in practice, consider a simple pin cell of fuel surrounded by moderator, which is discretized spatially with eight azimuthal divisions (Fig. 2.5), yielding sixteen fine mesh regions. With step characteristics, the source within each of these regions is flat, but higher order expansions permit the source shape to vary [12, 57]. Generally, additional radial divisions are used, but for simplicity these have been omitted. One of the major benefits of the MOC method is that is permits more accurate representation of curved or cylindrical material boundaries, as exist in fuel pins.

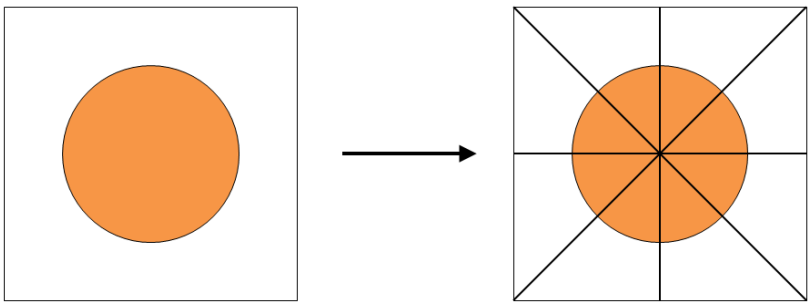

Figure 2.5: Pin Cell Azimuthal Divisions

Now also consider a quadrature with eight total azimuthal angles. Figure 2.6 shows the rays for these eight angles transcribed onto the pin cell.

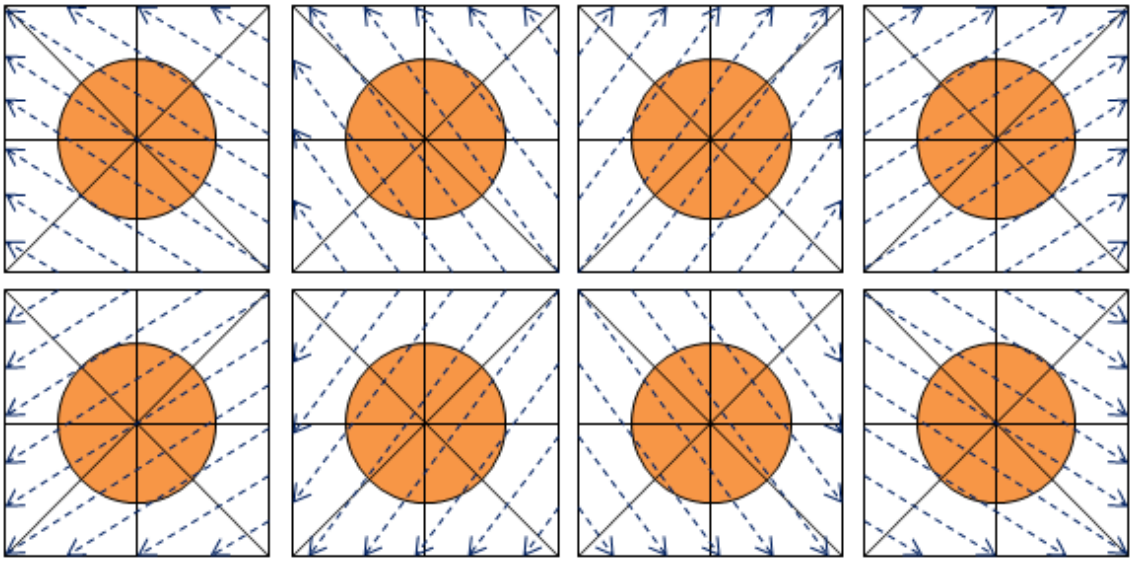

Figure 2.6: Rays at Different Azimuthal Angles

The angular fluxes for this problem can be determined by sweeping all of the rays in each angle and determining the average scalar flux in each region.

2.2.1.1 Modularization

Now consider a larger multipin problem in which the rays in each cell transmit data to neighboring cells. It is possible to trace rays throughout the entire spatial domain contiguously, but this would require a considerably larger memory overhead. To avoid this, ray segment data are only stored on the smallest repeatable structures. This could be on the assembly or quarter assembly basis (known as assembly modular ray tracing (AMRT)), but Figure 2.7 illustrates this on the pin cell basis (known as cell modular ray tracing (CMRT)). Taking one of the angles from Figure 2.6 and placing two pins side by side, it can be seen that the rays at the interface do not line up (shown below as blue). Though it is possible to carry on this calculation using interpolation to determine the incoming angular flux, the rays are usually "modularized" (shown below as red) to ensure that the rays line up, providing a more physical simulation of the neutron track. In this process the angles, angular weights, and ray spacing of the rays inside each module are adjusted [21, 40]. As the ray spacing is refined, fewer changes to these parameters are necessary to modularize the rays, minimizing the affect on accuracy.

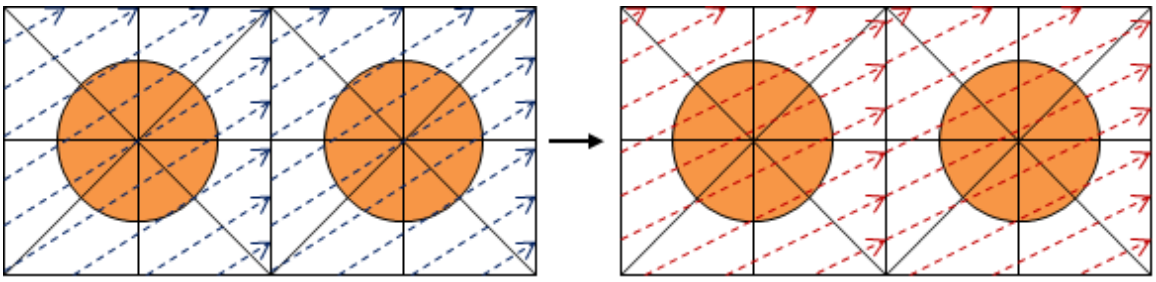

Figure 2.7: Ray Modularization

2.2.1.2 Other MOC Concepts

There are several other aspects of the MOC formulation that have been omitted (volume correction, cyclic rays, etc.) and will be deferred to the MPACT Theory Manual for more detail [40].

2.2.2 Coarse Mesh Finite Difference (CMFD)

Coarse Mesh Finite Difference (CMFD) is a commonly used acceleration method for solving transport problems. It builds on the principles of neutron conservation and finite difference methods, solving the diffusion equation on a mesh with larger cells than the transport sweeper uses. In MPACT, CMFD is performed on the pin-wise basis, enforcing a Cartesian grid over the entire system. The transport mesh, which is referred to as the "fine mesh" here, is homogenized (Eq. 2.33) to build the data for the "coarse mesh". In these equations, n is the index of the coarse mesh and i is the fine mesh index of regions contained inside each coarse mesh.

$$\overline{\Sigma}_{x,g}^{n} = \frac{\sum\limits_{i \in n} \Sigma_{x,g}^{i} V^{i} \phi_{g}^{i}}{\sum\limits_{i \in n} V^{i} \phi_{g}^{i}}, \tag{2.33a}$$

$$\overline{\phi}_{g}^{n} = \frac{\sum\limits_{i \in n} V^{i} \phi_{g}^{i}}{\sum\limits_{i \in n} V^{i}}. \tag{2.33b}$$

Figure 2.8 shows an illustration of the homogenization process on the pin-cell basis.

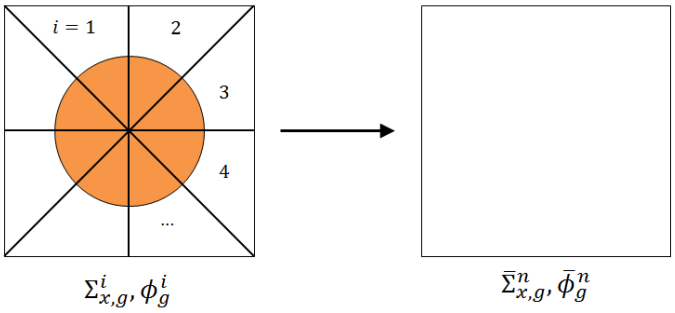

Figure 2.8: CMFD Pin Cell Homogenization

During the homogenization process, the projection ratios (Eq. 2.33c), which are used to transfer the final CMFD fluxes back to the transport mesh are also calculated. In this case i still represents the fine mesh and n, the coarse mesh. These are constructed using the same fluxes that are used to homogenize the cross section and flux data going into CMFD.

$$\bar{\xi}^i_g = \frac{\phi^i_g}{\bar{\phi}^n_g}, \tag{2.33c}$$

After the homogenization process is complete, the coupling coefficients are calculated. Eq. 2.34a shows the equation for the traditional finite difference coupling coefficient ($\tilde{D}$) for a coarse mesh cell n and neighbor $n(i)$, where D_g can be determined using the standard definition of the diffusion coefficient $\left(D_g = \frac{1}{3\Sigma_{tr}}\right)$ or with higher order methods [2]. In this equation, i denotes the directional index, as the neighbor index is dependent upon which direction (north, south, east, west, top, bottom) is being considered:

$$\tilde{D}_{g,n,i} = \frac{2D_{g,n}D_{g,n(i)}}{D_{g,n}h_{n(i)} + D_{g,n(i)}h_n}. \tag{2.34a}$$

Eqs. 2.34b and 2.34c show the expressions for the the current correction coupling coefficient, which uses the surface current at the boundary of the CMFD, which preserves the accuracy of the higher order method. With the 2D/1D scheme, currents from both the 2D-MOC and 1D-nodal solvers are used to construct the $\hat{D}$ values:

$$J^{transport}_{s,g,n,i} = -\tilde{D}_{g,n,i}(\bar{\phi}_n - \bar{\phi}_{n(i)}) + \hat{D}_{g,n,i}(\bar{\phi}_n + \bar{\phi}_{n(i)}), \tag{2.34b}$$

$$\hat{D}_{g,n,i} = \frac{J^{transport}_{s,g,n} + \tilde{D}_{g,n}(\bar{\phi}_n - \bar{\phi}_{n(i)})}{\bar{\phi}_n + \bar{\phi}_{n(i)}}. \tag{2.34c}$$

In addition to accelerating the eigenvalue and scalar flux distribution, the CMFD

solution can be used to accelerate the incoming angular fluxes on the system and parallel boundaries used in the MOC calculation. This is accomplished by formulating surface flux coefficients (Eq. 2.35) that are very similar to the current coupling coefficients above:

$$\tilde{s}_{g,n,i} = \frac{D_{g,n(s)} h_{n(i)}}{D_{g,n(i)} h_{n(i)} + D_{g,n} h_n}, \tag{2.35a}$$

$$\hat{s}_{g,n,i} = \frac{\phi^{MOC}_{s,g,n,i} - \tilde{s}_{g,n,i}\phi_{g,n} + (1 - \tilde{s}_{g,n,i})\phi_{g,n(i)}}{\phi_{g,n} + \phi_{g,n(i)}}. \tag{2.35b}$$

Once the surface fluxes and currents are found (per equations 2.34 and 2.35), the incoming angular fluxes can be updated, using a ratio of surface flux values on each surface before (pre) and after (post) the CMFD calculation (Eq. 2.36). This particular update isotropically scales the incoming angular flux of all angles (index l):

$$\varphi_{in,l,post} = \frac{\varphi_{s,post}}{\varphi_{s,pre}} \varphi_{in,l,pre}. \tag{2.36}$$

More complicated, angle-dependent updates were investigated [53], but without noteworthy improvement over the isotropic update.

2.2.3 Nodal Methods

The following sections discuss three methods that are used in the axial solvers in the 2D/1D scheme, so the focus will be on how these methods are used to solve 1D transport problems. Additionally, the Nodal Expansion Method and Source Expansion Nodal Method techniques, as implemented in MPACT, are used to discretize the 1D diffusion equation, and both use two-node kernel formulations. The Simplified P_N method, on the other hand, is not limited to the diffusion equation and uses a one-node kernel.

2.2.3.1 Nodal Expansion Method (NEM)

In the nodal expansion method, the source is expanded with quadratic Legendre polynomials and the flux is expanded with quartic polynomials [15], where ξ denotes the normalized spatial variable:

$$Q(\xi) = \sum_{i=0}^{2} q_i P_i(\xi), \tag{2.37a}$$

$$\phi(\xi) = \sum_{i=0}^{4} a_i P_i(\xi). \tag{2.37b}$$

The coefficients (a_i) are determined from the 0^{th} through 2^{nd} moment balance equations (Eq. 2.38a) in conjunction with the flux and current continuity enforcement (Eq. 2.38c and 2.38d):

$$\int_{-1}^{1} P_n(\xi)\left(-\Sigma_D \frac{d^2}{d\xi^2}\phi(\xi) + \Sigma_r\phi(\xi) - Q(\xi)\right)d\xi = 0, \tag{2.38a}$$

$$\Sigma_D = \frac{4D}{h^2}, \tag{2.38b}$$

$$\phi_1(1) = \phi_2(-1), \tag{2.38c}$$

$$J_1(1) = J_2(-1). \tag{2.38d}$$

These flux coefficients are then used to construct the source coefficients (q_i) for the next iteration. A full derivation of the NEM equations is presented in Appendix A.1. However, each internal node solves a linear system relating the information in the two-nodes being simulated (Eq. 2.39). Here 1 and 2 indicate the index of each of the two nodes:

$$\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 \\
-\frac{2D_1}{h_1} & -3\frac{2D_1}{h_1} & -6\frac{2D_1}{h_1} & -10\frac{2D_1}{h_1} & \frac{2D_2}{h_2} & -3\frac{2D_2}{h_2} & 6\frac{2D_2}{h_2} & -10\frac{2D_2}{h_2} \\
 & -3\Sigma_{D1} & & -10\Sigma_{D1} & & & & \\
\Sigma_{r1} & & -15\Sigma_{D1} & & & & & \\
 & \Sigma_{r1} & & -35\Sigma_{D1} & & & & \\
 & & & & & -3\Sigma_{D2} & & -10\Sigma_{D2} \\
 & & & & \Sigma_{r2} & & -15\Sigma_{D2} & \\
 & & & & & \Sigma_{r2} & & -35\Sigma_{D2}
\end{bmatrix}
\begin{bmatrix} \phi_{1,1} \\ \phi_{1,2} \\ \phi_{1,3} \\ \phi_{1,4} \\ \phi_{2,1} \\ \phi_{2,2} \\ \phi_{2,3} \\ \phi_{2,4} \end{bmatrix}
=
\begin{bmatrix} -\overline{\phi}_1 + \overline{\phi}_2 \\ 0 \\ q_{1,0} - \Sigma_{r1}\overline{\phi}_1 \\ q_{1,1} \\ q_{1,2} \\ q_{2,0} - \Sigma_{r2}\overline{\phi}_2 \\ q_{2,1} \\ q_{2,2} \end{bmatrix} \tag{2.39}$$

Boundary nodes are solved in a slightly different manner since one of the nodes would technically be outside of the system. In these cases, special one-node systems are formulated, taking into account the boundary conditions.

2.2.3.2 Source Expansion Nodal Method (SENM)

With SENM, both the source and the flux use a quartic Legendre expansion. However, the flux also has two additional hyperbolic terms [62]:

$$Q(\xi) = \sum_{i=0}^{4} q_i P_i(\xi), \tag{2.40a}$$

$$\phi(\xi) = Asinh(\kappa\xi) + Bcosh(\kappa\xi) + \sum_{i=0}^{4} a_i P_i(\xi), \tag{2.40b}$$

$$\kappa = \frac{h}{2}\sqrt{\frac{\Sigma_r}{D}}. \tag{2.40c}$$

In the previous set of equations, the flux has a homogeneous ($Asinh(\kappa\xi) + Bcosh(\kappa\xi)$) and particular solution $\left(\sum_{i=0}^{4} a_i P_i(\xi)\right)$. The particular solution coefficients can be determined by solving the 0^{th} through 4^{th} order moment balance equations. The homogeneous coefficient B for each node is given from the node-averaged flux and zeroth moment particular coefficient a_0:

$$B = \frac{\kappa}{sinh(\kappa)}\left(\overline{\phi} - a_0\right). \tag{2.41}$$

Having solved the particular coefficient (a_i) and homogeneous B coefficients, the homogeneous A coefficients for the two nodes (A_1 and A_2) can then be calculated by enforcing flux and current continuity at the interface of the two nodes (Eq. 2.38c and 2.38d). A full derivation of the SENM equations is presented in Appendix A.2.

2.2.3.3 Simplified P_N (SP_N)

In 1D, the simplified P_N and full P_N equations are identical. However, the term "simplified P_N" is commonly used to help emphasize that these methods are not as burdensome as the full, multidimensional P_N equations can be. With this method, the angular flux, which has been azimuthally integrated, is assumed to have a Legendre expansion in μ (Eq. 2.42) [39]:

$$\varphi_g(x,\mu) = \sum_{m=0}^{N_{mom}} \frac{2m+1}{2}\varphi_{m,g}(x)P_m(\mu). \tag{2.42}$$

By substituting Eq. 2.42 into the transport equation, multiplying by $P_n(\mu)$, and integrating over μ, Eq. 2.43a can be obtained which describes the relationship between the moments:

$$\frac{d}{dx}\left[\frac{n}{2n+1}\varphi_{n-1,g}(x) + \frac{n+1}{2n+1}\varphi_{n+1,g}(x)\right] + \Sigma_{t,g}(x)\varphi_n(x,g) = \Sigma_{sn,g\to g}\varphi_{n,g}(x) + Q_g(x)\delta_{n,0}. \tag{2.43a}$$

And it also yields the corresponding boundary conditions (Eq. 2.43b and 2.43c) for the left and right interfaces, respectively:

$$\sum_{m=0}^{N_{mom}} \frac{2m+1}{2}\left(\int_{-1}^{0} \mu P_n(\mu)P_m(\mu)d\mu\right)\varphi_{m,g}(0) = \int_0^1 \mu P_n(\mu)\varphi_g^b(\mu)d\mu. \tag{2.43b}$$

$$\sum_{m=0}^{N_{mom}} \frac{2m+1}{2}\left(\int_{-1}^{0} \mu P_n(\mu) P_m(\mu) d\mu\right)\varphi_{m,g}(X) = \int_{-1}^{0} \mu P_n(\mu)\varphi_g^b(\mu) d\mu. \tag{2.43c}$$

In MPACT, SP_3 and SP_5 have been implemented, which can be formulated to wrap the one-node NEM kernel by normalizing spatially (ξ) and rearranging the boundary conditions slightly. With this setup, SP_N handles the angular distribution and NEM handles the spatial distribution. The details of the derivation will be left out here (see Appendix A.3), but once the equations for the 0^{th} through 3^{rd} moments have been found, the first and third can be solved and substituted into the 0^{th} and 2^{nd} moment equations. These can be further simplified into Eqs. 2.44 and 2.45, which show the final equations for the SP_3 kernel:

Zeroth-Moment

$$-\frac{4D_0}{h^2}\frac{d^2}{d\xi^2}\Phi_0(\xi) + (\Sigma_t - \Sigma_{s0})\Phi_0(\xi) = Q(\xi) + 2(\Sigma_t - \Sigma_{s0})\Phi_2(\xi), \tag{2.44a}$$

$$-\frac{D_0}{h}\frac{d}{d\xi}[\Phi_0(-1)] + \frac{1}{4}\Phi_0(-1) = \int_0^1 \mu\varphi^b(\mu) d\mu + \frac{3}{16}\Phi_2(-1), \tag{2.44b}$$

$$\frac{D_0}{h}\frac{d}{d\xi}[\Phi_0(1)] + \frac{1}{4}\Phi_0(1) = \int_{-1}^{0} |\mu|\varphi_R^b(\mu) d\mu + \frac{3}{16}\Phi_2(1), \tag{2.44c}$$

Second-Moment

$$\begin{aligned} -\frac{4D_2}{h^2}\frac{d^2}{d\xi^2}[\Phi_2(\xi)] + \left(\frac{9}{5}\Sigma_t - \frac{4}{5}\Sigma_{s0} - \Sigma_{s2}\right)\Phi_2(\xi) \\ = -\frac{2}{5}\Big(Q(\xi) - (\Sigma_t - \Sigma_{s0})\Phi_0(\xi)\Big), \end{aligned} \tag{2.45a}$$

$$-\frac{D_2}{h}\frac{d}{d\xi}[\Phi_2(-1)] + \frac{1}{4}\Phi_2(-1) = \frac{3}{5}\int_0^1 P_3(\mu)\varphi^b(\mu) d\mu + \frac{3}{80}\Phi_0(-1) - \frac{1}{80}\Phi_2(-1), \tag{2.45b}$$

$$\frac{D_2}{h}\frac{d}{d\xi}[\Phi_2(1)] + \frac{1}{4}\Phi_2(1) = \frac{3}{5}\int_{-1}^{0} P_3(\mu)\varphi_R^b(\mu) d\mu + \frac{3}{80}\Phi_0(1) - \frac{1}{80}\Phi_2(1), \tag{2.45c}$$

where

$$\Phi_{0,g}(\xi) = \varphi_{0,g}(\xi) + 2\varphi_{2,g}(\xi), \tag{2.46a}$$

$$\Phi_{2,g}(\xi) = \varphi_{2,g}(\xi), \tag{2.46b}$$

$$D_{0,g} = \frac{1}{3\left(\Sigma_{t,g} - \sum_{g'=1}^{N_g} \Sigma_{s1,g'\to g}\right)}, \tag{2.46c}$$

$$D_{2,g} = \frac{9}{35\left(\Sigma_{t,g} - \sum_{g'=1}^{N_g} \Sigma_{s3,g'\rightarrow g}\right)}. \tag{2.46d}$$

In the current implementation, the higher order scattering cross sections are neglected. Future work will consider incorporating this data. Additionally, the SP_N equations derived in the appendix do not include the spatial normalization that is included in these equations.

2.2.3.4 One-Node vs. Two-Node

There are two types of nodal kernels that are primarily used in MPACT: one-node and two-node. In each of these, the underlying physics is the same and is determined by the flux and source expansions being used. However, the boundary conditions and constraints differ in important ways. In the one node formulation (Figure 2.9), the incoming partial currents (or angular fluxes for transport-based kernels) are the prescribed boundary conditions and the outgoing partial currents and flux distribution are output.

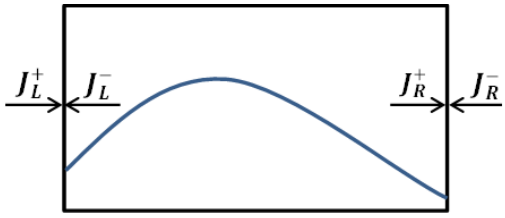

Figure 2.9: One-Node Kernel

With the two node kernels (Figure 2.10), the mesh averaged scalar fluxes are used as a constraint and the net current at the interface and the flux distributions in both nodes are output. Additionally, the two-node kernels own two instances of the intranode flux distribution (one for the left interface and one for the right interface). During the iteration process the two intranodal distributions will not necessarily agree, but they will at convergence.

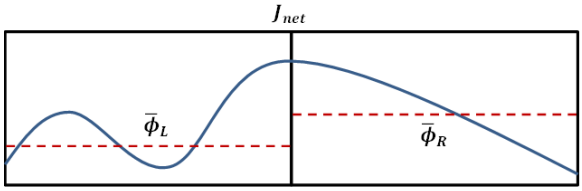

Figure 2.10: Two-Node Kernel

The fact that the two-node kernels do not allow the mesh average scalar flux (zeroth moment) to change is beneficial and considerably more stable. When incorporating the

transverse leakage terms, which will be discussed in more detail later, it is possible to encounter negative sources, which can drive the one-node kernels to non-physical negative fluxes, whereas the two-node kernels can handle these in stride more easily.

Unfortunately, two-node formulations are currently only available for the diffusion-based kernels (such as NEM and SENM). If such a formulation were applied to the transport-based kernels, mesh-averaged angular fluxes would need to be preserved, which are not available with the existing 2D/1D scheme. However, it is possible that a two-node transport kernel could be used in something such as the 2D/3D scheme, in which 2D-MOC and 3D-S_N are coupled [63].

While MPACT is limited to using one- and two-node kernels, other codes solve an entire axial pin in a single linear system [50]. This technique provides a tighter coupling between the nodes and the CMFD solve, and can require fewer iterations with the one-node kernels, but requires more parallel communication up front.

2.2.4 Monte Carlo Methods

The focus of the work in this document is on improving deterministic transport methods within the 2D/1D framework. A completely different set of methods exists within probabilistic transport or Monte Carlo methods. The goal of probabilistic methods is to simulate the transport of neutrons with as few approximations as possible by directly following the life of a neutron, from birth through fission (or several other mechanisms), through a number of scattering events, down to thermal energies where it is more likely to cause a subsequent fission event. Of course, not all neutrons follow this path, as there are a number of different possibilities, such as leaking from the system, being absorbed into a burnable poison pin, or causing a fast fission event. By simulating individual neutron "histories" and minimizing approximations (particularly in regards to space and energy discretization), Monte Carlo methods can provide the most exact solution.

Typically, histories are lumped in batches, where a batch is effectively an outer iteration. All of the histories within a batch are simulated using the same source distribution that governs where neutrons are produced. This source distribution becomes more accurate as more histories and batches have been run. Various parameters, such as the eigenvalue, can be calculated during each batch based on the history data within it. Each parameter will have an associated uncertainty because of the random nature of the simulations. Efficiently reducing the uncertainties of these parameters is a large field of research as it is one of the major hurdles of the method. With deterministic transport, an exact solution to an approximated transport problem is found, whereas with probabilistic transport

an approximate solution to the exact transport problem can be found. Reducing the uncertainties in the solution pose a considerable burden, particularly for larger, high-dominance problems.

Throughout the history of reactor analysis, many Monte Carlo codes have been written, perhaps the most prevalent being the MCNP code [59]. Several recent efforts, such as OpenMC [48] and SHIFT [11], have made great efforts to be massively parallel, a current limitation of MCNP. Later in this work, results will be compared to reference solutions obtained using SHIFT, which is being developed at the Oak Ridge National Laboratory.

CHAPTER 3

The 2D/1D Framework

With the 2D/1D scheme, spatially three-dimensional problems are decomposed into a 1D-axial stack of 2D-radial planes (as previously shown in Figure 1.1). Typically, the axial solver operates on pin-homogenized cross sections and uses a lower fidelity solver than in the radial direction. This is generally a valid approximation, as most of the geometric heterogeneity exists radially where the flux gradients can be more severe. Axially, however, there are fewer geometric changes and the flux gradients tend to be more diffusive.

In the following sections, the governing equations of the radial and axial sweepers is presented as well as detail of the transverse leakage splitting techniques, which are used to guarantee positive flux solutions. Additionally, the iteration flow and relaxation techniques are presented, as well as a more in depth consideration of the sources of error present in this method. Lastly, parallel decomposition schemes are discussed.

3.1 Governing Equations

3.1.1 Radial Equations

This section concerns the underlying equations for the radial sweepers. The three-dimensional, steady-state Boltzmann transport equation was covered previously in Chapter 2. For simplicity, the scattering is assumed to be isotropic:

$$\left(\sqrt{1-\mu_l^2}\Big(cos(\alpha_l)\frac{\partial}{\partial x}+sin(\alpha_l)\frac{\partial}{\partial y}\Big)+\mu_l\frac{\partial}{\partial z}\right)\varphi_{g,l}(x,y,z)+\Sigma_{t,g}(x,y,z)\varphi_{g,l}(x,y,z)=\overline{q}_{g,l}(x,y,z), \tag{3.1a}$$

$$\overline{q}_{g,l}(x,y,z)=\frac{\chi_g(x,y,z)}{4\pi k_{\text{eff}}}\sum_{g'=1}^{N_g}\nu\Sigma_{f,g'}(x,y,z)\phi_{g'}(x,y,z)+\frac{1}{4\pi}\sum_{g'=1,g'\neq g}^{N_g}(x,y,z)\Sigma_{s0,g'\rightarrow g}\phi_{g'}(x,y,z). \tag{3.1b}$$

Averaging Eq. 3.1 axially as in Eq. 3.2 and assuming separability between the cross

sections and the flux, Eq. 3.3 is obtained. Typically, the materials in each plane are axially constant, in which there is no approximation in averaging/homogenizing the cross sections:

$$\varphi^Z_{g,l}(x,y) = \frac{1}{h_z}\int_{z_B}^{z_T} \varphi_{g,l}(x,y,z)dz, \tag{3.2}$$

$$\sqrt{1-\mu_l^2}\left(cos(\alpha_l)\frac{\partial}{\partial x} + sin(\alpha_l)\frac{\partial}{\partial y}\right)\varphi^Z_{g,l}(x,y) + \Sigma^Z_{t,g}(x,y)\varphi^Z_{g,l}(x,y) = \tilde{q}^Z_{g,l}(x,y). \tag{3.3a}$$

where $\tilde{q}$ consists of the scattering and fission sources in $\overline{q}$ (Eq. 3.3c), and the axial component of the streaming operator becomes the axial transverse leakage, which is angle-dependent, using the angular flux at the top and bottom planar interfaces (Eq. 3.3d):

$$\tilde{q}^Z_{g,l}(x,y) = \overline{q}^Z_{g,l}(x,y) + TL^Z_{g,l}(x,y), \tag{3.3b}$$

$$\overline{q}^Z_{g,l}(x,y) = \frac{\chi^Z_g(x,y)}{4\pi k_{\text{eff}}}\sum_{g'=1}^{N_g} \nu\Sigma^Z_{f,g'}(x,y)\phi^Z_{g'}(x,y) + \frac{1}{4\pi}\sum_{g'=1,g'\neq g}^{N_g} \Sigma^Z_{s0,g'\rightarrow g}(x,y)\phi^Z_{g'}(x,y), \tag{3.3c}$$

$$TL^Z_{g,l}(x,y) = \frac{\mu_l}{h_z}\left(\varphi_{B,g,l}(x,y) - \varphi_{T,g,l}(x,y)\right). \tag{3.3d}$$

A common approximation is to assume that axial transverse leakage is handled isotropically. This leakage is obtained by averaging Eq. 3.3d over $\mathbf{\Omega}$:

$$TL^Z_{g,l}(x,y) = \frac{J_{B,g}(x,y) - J_{T,g}(x,y)}{4\pi h_z}. \tag{3.4}$$

This has several computational advantages that are discussed in the following sections.

Another approximation is to assume that the transverse leakage is spatially flat (Eq. 3.5), not only with the fine mesh region, but also across the entire pin, since the axial solver is generating surface flux moment quantities for each coarse mesh surface, which spans an entire pin. By averaging 3.3d radially, the equations for the flat approximation can be found:

$$TL^Z_{g,l} = \frac{\mu_l}{h_z}\left(\varphi^{XY}_{B,g,l} - \varphi^{XY}_{T,g,l}\right), \tag{3.5a}$$

$$TL^Z_{g,l} = \frac{J^{XY}_{B,g} - J^{XY}_{T,g}}{4\pi h_z}, \tag{3.5b}$$

where, for example, Eq. 3.6 shows the radially-averaged angular flux:

$$\varphi^{XY}_{g,l} = \frac{1}{A_{xy}}\int_{x_L}^{x_R}\int_{y_L}^{y_R} \varphi_{g,l}(x,y)dydx. \tag{3.6}$$

3.1.1.1 Angular Dependence of the Axial Transverse Leakage

In Equations 3.3 one can see that in its most accurate form the axial transverse leakage should be constructed using the angular fluxes at the axial boundaries of each plane. There are effectively two ways to do this: 1) using an approximation for the angular flux and 2) using the angular fluxes from the axial S_N sweeper (as in 3.5a).

In the first approach, one can use the surface flux and net current, assuming the angular flux varies linearly in angle (Eq. 3.7):

$$\varphi_l \approx \frac{1}{4\pi}[\phi_s + 3\mu_l J_z]. \tag{3.7}$$

Blake Kelley, a student under Prof. Edward Larsen, is working on alternative axial transverse leakage techniques that incorporate both angular moments and spatial distributions within each pin. In his work, he is using the axial surface flux and current to build approximations of the angular flux, as well as other approximations of the streaming operator. It is important to note that approximating the angular flux applicable to all axial solvers, even the ones that do not produce angular fluxes directly, such as the diffusion-based NEM and SENM solvers.

Using the angular fluxes directly from the axial S_N sweeper complements this work well, as it provides a more explicit representation of the leakage that he is attempting to approximate.

In either approach, however, it becomes necessary to break one of the assumptions used in most 2D-MOC sweepers: polar symmetry between the positive and negative half spaces.

3.1.1.2 Polar Asymmetry

When an isotropic axial transverse leakage is used, the MOC solution from the positive and negative half spaces being simulated is identical. The pre-existing 2D-MOC implementation takes advantage of this fact so only half of the polar angles are simulated at twice the weight. Figure 3.1 demonstrates this considering four polar angles (two per half space). The red rays denote symmetric polar angles that are usually neglected from the calculation. However, if polar or full angular dependence is added to the axial transverse leakage terms, the assumption of symmetry is no longer valid, and the full set of polar angles needs to be used. This also allows the radial transverse leakages to gain full polar dependence as well.

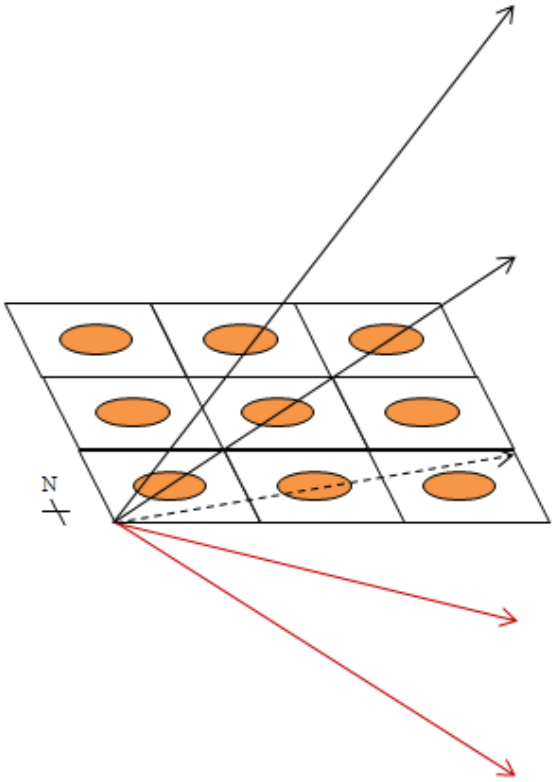

Figure 3.1: Illustration of Polar Symmetry

As is seen in the results in Chapter 5, while simulating both half-spaces of polar angles effectively doubles the amount of work that should be done, it does not translate to double the computational time. This is because part of the work performed by the sweeper, such as for setting up the long rays, does not need to be duplicated. In the end, there is about a 50% increase in the run time, which is considerably better than the 100% increase that might be more immediately expected.

Because there are a few different options for the axial transverse leakage, several "full polar" MOC sweepers have been implemented. The details of these are outlined in Chapter 4 after the Fourier moment-based transverse leakage has been introduced.

3.1.2 Axial Equations

3.1.2.1 Transport-Based

The axial equations can be derived in a similar manner by averaging the three-dimensional transport equation radially over both x and y, as in Eq. 3.8, yielding Eq. 3.9. This leaves only a dependence in z, but with a radial transverse leakage in the source 3.9d:

$$\varphi_{g,l}^{XY}(z,\alpha_l,\mu_l) = \frac{1}{A_{xy}} \int_{x_L}^{x_R} \int_{y_L}^{y_R} \varphi_{g,l}(x,y,z,\alpha_l,\mu_l) dy dx, \tag{3.8a}$$

$$\Sigma_{x,g}^{XY}(z) = \frac{\int_{x_L}^{x_R} \int_{y_L}^{y_R} \Sigma_{x,g}(x,y,z)\phi(x,y,z) dy dx}{\int_{x_L}^{x_R} \int_{y_L}^{y_R} \phi(x,y,z) dy dx}, \tag{3.8b}$$

$$\mu_l \frac{\partial}{\partial z} \varphi_{g,l}^{XY}(z,\alpha_l,\mu_l) + \Sigma_{t,g}^{XY}(z)\varphi_{g,l}^{XY}(z,\alpha_l,\mu_l) = \tilde{q}_{g,l}^{XY}(z,\alpha_l,\mu_l), \tag{3.9a}$$

$$\tilde{q}_{g,l}^{XY}(z,\alpha_l,\mu_l) = \overline{q}_{g,l}^{XY}(z,\alpha_l,\mu_l) + TL_{g,l}^{XY}(z,\alpha_l,\mu_l), \tag{3.9b}$$

$$\overline{q}_{g,l}^{XY}(z,\alpha_l,\mu_l) = \frac{\chi_g}{4\pi k_{\text{eff}}} \sum_{g'=1}^{N_g} \nu\Sigma_{f,g'}^{XY}(z)\phi_{g'}^{XY}(z) + \frac{1}{4\pi} \sum_{g'=1}^{N_g} \Sigma_{s,g'\to g}^{XY}(z)\phi_g^{XY}(z), \tag{3.9c}$$

$$TL_{g,l}^{XY}(z,\alpha_l,\mu_l) = -\frac{\sqrt{1-\mu_l^2}}{A_{xy}} \begin{pmatrix} cos(\alpha_l) \int_{y_L}^{y_R} \Big(\varphi_{g,l}(x_R,y,z,\alpha_l,\mu_l) - \varphi_{g,l}(x_L,y,z,\alpha_l,\mu_l)\Big) dy \\ + sin(\alpha_l) \int_{x_L}^{x_R} \Big(\varphi_{g,l}(x,y_R,z,\alpha_l,\mu_l) - \varphi_{g,l}(x,y_L,z,\alpha_l,\mu_l)\Big) dx \end{pmatrix}. \tag{3.9d}$$

Though the discrete ordinates approximation has already been made, α_l and μ_l are included in the angular dependence to make the angular integrations more clear. In reality, these integrals are replaced with summations over the discrete ordinates. Additionally, because of the separability assumed between the flux and cross sections, the homogenized cross sections are obtained using flux and area weighting, though the fission spectrum is homogenized using the fission source instead of the flux. For completeness, the total cross section should be angle-dependent and homogenized using the angular flux, but a common approximation is to also homogenize it using the scalar flux. This is done to save storage of both the angle-dependent cross sections and angular fluxes.

A variant of this is to integrate Eq. 3.9 azimuthally (Eq. 3.10), so the radial transverse leakage maintains polar dependence. This formulation was used by Hursin in previous work with DeCART [19, 20]:

$$\mu_l \frac{\partial}{\partial z} \varphi_{g,l}^{XY,\alpha}(z,\mu_l) + \Sigma_{t,g}^{XY}(z)\varphi_{g,l}^{XY,\alpha}(z,\mu_l) = \tilde{q}_{g,l}^{XY}(z,\mu_l), \tag{3.10a}$$

$$\tilde{q}_{g,l}^{XY,\alpha}(z,\mu_l) = \overline{q}_{g,l}^{XY,\alpha}(z,\mu_l) + TL_{g,l}^{XY,\alpha}(z,\mu_l), \tag{3.10b}$$

$$\overline{q}_{g,l}^{XY}(z,\mu_l) = \frac{\chi_g}{2k_{\text{eff}}} \sum_{g'=1}^{N_g} \nu\Sigma_{f,g'}^{XY}(z)\phi_{g'}^{XY}(z) + \frac{1}{2} \sum_{g'=1}^{N_g} \Sigma_{s,g'\to g}^{XY}(z)\phi_g^{XY}(z), \tag{3.10c}$$

$$TL_{g,l}^{XY,\alpha}(z,\mu_l) = -\frac{\sqrt{1-\mu_l^2}}{A}\int_0^{2\pi}\begin{pmatrix} cos(\alpha_l)\int_{y_L}^{y_R}\Big(\varphi_{g,l}(x_R,y,z,\alpha_l,\mu_l)-\varphi_{g,l}(x_L,y,z,\alpha_l,\mu_l)\Big)dy \\ +sin(\alpha_l)\int_{x_L}^{x_R}\Big(\varphi_{g,l}(x,y_R,z,\alpha_l,\mu_l)-\varphi_{g,l}(x,y_L,z,\alpha_l,\mu_l)\Big)dx \end{pmatrix}d\alpha, \tag{3.10d}$$

where

$$\varphi_{g,l}^{XY,\alpha}(z,\mu_l) = \int_0^{2\pi}\varphi_{g,l}^{XY}(z,\alpha_l,\mu_l)d\alpha. \tag{3.11}$$

Eq. 3.10d can be also be averaged polarly to remove angular dependence for use in the azimuthally-integrated equations 3.10. The next section, which covers the diffusion-based axial solver, uses isotropic radial transverse leakage, in which case Eq. 3.9d has no angular dependence, as in Eq. 3.12:

$$TL_{g,l}^{XY}(z) = -\frac{1}{2A}\int_{-1}^{1}\sqrt{1-\mu_l^2}\int_0^{2\pi}\begin{pmatrix} cos(\alpha_l)\int_{y_L}^{y_R}\Big(\varphi_{g,l}(x_R,y,z)-\varphi_{g,l}(x_L,y,z)\Big)dy \\ +sin(\alpha_l)\int_{x_L}^{x_R}\Big(\varphi_{g,l}(x,y_R,z)-\varphi_{g,l}(x,y_L,z)\Big)dx \end{pmatrix}d\alpha d\mu. \tag{3.12}$$

All three of the transverse leakage equations (Eqs. 3.9d, 3.10d, 3.12) can be averaged over z to use the angular fluxes from the radial solvers (Eq. 3.13):

$$TL_{g,l}^{XY}(z,\alpha_l,\mu_l) = -\frac{\sqrt{1-\mu_l^2}}{A}\begin{pmatrix} cos(\alpha_l)\int_{y_L}^{y_R}\Big(\varphi_{g,l}^{Z}(x_R,y,\alpha_l,\mu_l)-\varphi_{g,l}^{Z}(x_L,y,\alpha_l,\mu_l)\Big)dy \\ +sin(\alpha_l)\int_{x_L}^{x_R}\Big(\varphi_{g,l}^{Z}(x,y_R,\alpha_l,\mu_l)-\varphi_{g,l}^{Z}(x,y_L,\alpha_l,\mu_l)\Big)dx \end{pmatrix}, \tag{3.13a}$$

$$TL_{g,l}^{XY,\alpha}(z,\mu_l) = -\frac{\sqrt{1-\mu_l^2}}{A}\int_0^{2\pi}\begin{pmatrix} cos(\alpha_l)\int_{y_L}^{y_R}\Big(\varphi_{g,l}^{Z}(x_R,y,\alpha_l,\mu_l)-\varphi_{g,l}^{Z}(x_L,y,\alpha_l,\mu_l)\Big)dy \\ +sin(\alpha_l)\int_{x_L}^{x_R}\Big(\varphi_{g,l}^{Z}(x,y_R,\alpha_l,\mu_l)-\varphi_{g,l}^{Z}(x,y_L,\alpha_l,\mu_l)\Big)dx \end{pmatrix}d\alpha, \tag{3.13b}$$

$$TL_{g,l}^{XY,\alpha,\mu}(z) = -\frac{1}{2A}\int_{-1}^{1}\sqrt{1-\mu_l^2}\int_0^{2\pi}\begin{pmatrix} cos(\alpha_l)\int_{y_L}^{y_R}\Big(\varphi_{g,l}^{Z}(x_R,y,\alpha_l,\mu_l)-\varphi_{g,l}^{Z}(x_L,y,\alpha_l,\mu_l)\Big)dy \\ +sin(\alpha_l)\int_{x_L}^{x_R}\Big(\varphi_{g,l}^{Z}(x,y_R,\alpha_l,\mu_l)-\varphi_{g,l}^{Z}(x,y_L,\alpha_l,\mu_l)\Big)dx \end{pmatrix}d\alpha d\mu. \tag{3.13c}$$

3.1.2.2 Diffusion-Based

The diffusion based axial sweepers are formulated by radially-averaging the three-dimensional diffusion equation:

$$-D_g \frac{\partial^2 \phi_g(z)}{\partial z^2} + \Sigma_{r,g}(z)\phi_g(z) = \frac{\chi_g(z)}{k_{\text{eff}}} \sum_{g'=1}^{N_g} \nu\Sigma_{f,g'}(z)\phi_{g'}(z) + \frac{1}{4\pi} \sum_{g'=1,g'\neq g}^{N_g} \Sigma_{s,g'\to g}(z)\phi_{g'}(z) - TL_g^{XY}(z), \tag{3.14a}$$

$$TL_g^{XY}(z) = \frac{1}{h_x}\Big(J_{L,x,g}(z) - J_{R,x,g}(z)\Big) + \frac{1}{h_y}\Big(J_{L,y,g}(z) - J_{R,y,g}(z)\Big). \tag{3.14b}$$

Since the radial surfaces on which the currents exist span the entire plane, they effectively do not have an axial dependence, so the leakage is more consistent with Eq. 3.15:

$$TL_g^{XY}(z) = \frac{1}{h_x}\Big(J^Z_{L,x,g} - J^Z_{R,x,g}\Big) + \frac{1}{h_y}\Big(J^Z_{L,y,g} - J^Z_{R,y,g}\Big). \tag{3.15}$$

3.1.2.3 Transverse Leakage Interpolation

It is important to note that in equations 3.13 and 3.15 the angular fluxes and currents that come from the radial sweeper have no axial dependence. However, since most of the axial sweepers use some form of Legendre expansion of the spatial moments, they have the ability to incorporate higher order transverse leakage components into the source construction using the transverse leakages from the neighboring planes (Figure 3.2), higher order coefficients can be constructed to describe the shape of the transverse leakage in each axial node.

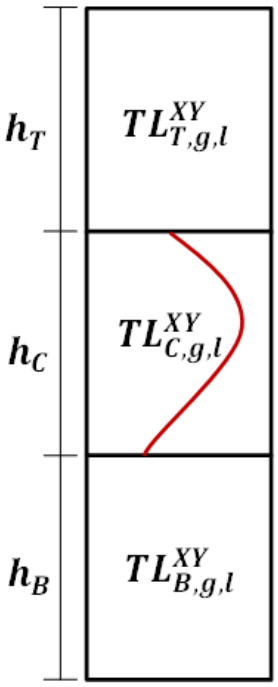

Figure 3.2: TL Interpolation

Eqs. 3.16 show the coefficients for a quadratic interpolation of the transverse leakage (Eq. 3.16a) using the values from the top and bottom neighboring planes, which can be obtained using Eqs. 3.13 and 3.15:

$$TL^{XY}_{g,l}(\xi) = \sum_{i=0}^{2} TL^{XY}_{g,l,i} P_i(\xi), \tag{3.16a}$$

$$G = 2(h_C + h_B)(h_C + h_T)(h_B + h_C + h_T), \tag{3.16b}$$

$$TL^{XY}_{g,l,0} = TL^{XY}_{C,g,l}, \tag{3.16c}$$

$$\begin{aligned} TL^{XY}_{g,l,1} = G^{-1} h^c [&(TL^{XY}_{T,g,l} - TL^{XY}_{C,g,l})(h_C + 2h_B)(h_C + h_B) \\ &- (TL^{XY}_{B,g,l} - TL^{XY}_{C,g,l})(h_C + 2h_T)(h_C + h_T)], \end{aligned} \tag{3.16d}$$

$$\begin{aligned} TL^{XY}_{g,l,2} = G^{-1} (h_C)^2 [&(TL^{XY}_{T,g,l} - TL^{XY}_{C,g,l})(h_C + h_B) \\ &+ (TL^{XY}_{B,g,l} - TL^{XY}_{C,g,l})(h_C + h_T)]. \end{aligned} \tag{3.16e}$$

It is feasible to use cubic or even quartic expansions, but this would require data from farther neighboring planes and potentially more data passing between processors.

3.2 Transverse Leakage Splitting

In both the radial and axial equations, difficulties can arise when the transverse leakage terms cause the total source (scattering plus fission plus transverse leakage) to become negative. One scheme to avoid this is known as transverse leakage splitting. This technique takes a negative transverse leakage (or some component of it) and adds it to the total cross section of the collision operator.

Consider the diffusion-based axial equation (Eq. 3.14a). If the transverse leakage were put into the left hand side, it would now look like Eq. 3.17:

$$-D_g \frac{\partial^2 \phi_g(z)}{\partial z^2} + \left(\Sigma_{r,g}^{XY}(z) + \frac{TL_g^{XY}(z)}{\phi_g(z)} \right) \phi_g(z) = \frac{\chi_g^{XY}(z)}{k_{\text{eff}}} \sum_{g'=1}^{N_g} \nu \Sigma_{f,g'}^{XY}(z) \phi_{g'}(z) + \sum_{g'=1, g' \neq g}^{N_g} \Sigma_{s,g' \to g}^{XY}(z) \phi_{g'}(z). \tag{3.17}$$

This guarantees positivity of the source and flux (so long as all other parameters are positive). There are several variants to this, pertaining to how the splitting is handled and how much is split [19]. One notable variant splits only the negative component of the total source, leaving a zero right hand side. This yields less influence on the final solution than splitting the entire transverse leakage. However, when spatial moments of the transverse leakage come into play (as is the case with the quadratic fit in the axial solvers), it is better to split only the zeroth moment of the leakage, leaving the scattering and fission sources as well as the higher order spatial moments of the leakage.

With the S_N sweeper, which can handle angle-dependent transverse leakages, effort is taken to ensure that the transverse leakage that is split is isotropic (i.e. that all angles see the same modified total cross section). This is done by looking at the sources for all angles and identifying the most negative source and using the transverse leakage for the angle to split (Fig. 3.3). This effectively leads to the transverse leakage being shifted uniformly to guarantee positive sources for all angles.

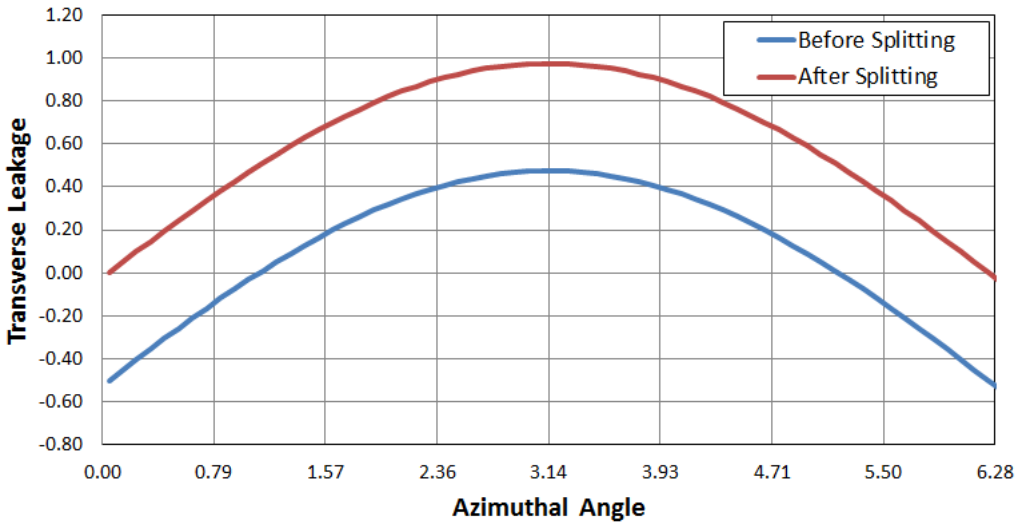

Figure 3.3: Angle-Dependent TL Splitting

The same can be done spatially by evaluating the sources at boundaries, searching for the most negative value on all angles (Fig. 3.4). An alternative approach is to limit the source coefficients to guarantee positivity [57], but this can also impact the accuracy of the solution.

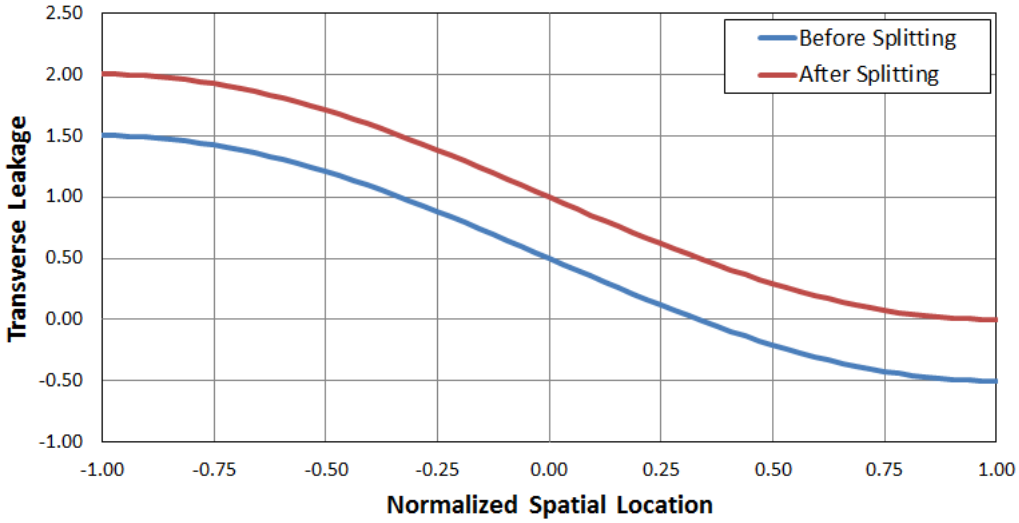

Figure 3.4: Spatial Source Shifting

It is important to note that these approaches to guaranteeing positivity are very aggressive in that they eliminate all possibility of negative fluxes. In fact, the production of negative fluxes is dependent on both the source and the incoming angular flux, so, in

practice, some source negativity can be allowed, as long as the incoming angular flux is large enough to offset it. During source construction, it is not clear how much negative source can be permitted, so an aggressive stance is taken.

It has been observed that these approaches to leakage splitting can be particularly detrimental to accuracy. Fortunately, all results analyzed in Chapter 5 were obtained without leakage splitting. Though these cases were able to be performed successfully, formulating a more consistent transverse leakage splitting technique could be necessary for other problems, particularly when it comes to mesh refinement, which can lead to problems with negative sources.

When the solution is guaranteed to be positive (as any physical solution is), any negative values encountered along the way are simply an issue with the path to the solution, in which splitting can be useful in making it through the potentially troublesome iterations. However, it should be noted that various approximations to the method can yield negative final solutions.

3.3 Iteration Strategy

Figure 3.5 shows the calculation flow for the 2D/1D scheme. In versions of DeCART (the predecessor of MPACT), 2D/1D could be used without CMFD, but in MPACT, all 2D/1D methods use CMFD as an accelerator. With this setup, CMFD updates the eigenvalue and the coarse mesh fluxes and currents, allowing for updated axial and radial transverse leakages to be constructed. Then, the axial and radial sweepers generate new surface currents that are used to build the $\hat{D}$ coupling coefficients in the next outer iteration. This loop process is continued until the eigenvalue and fine mesh fission source distribution are converged within the user-specified criteria.

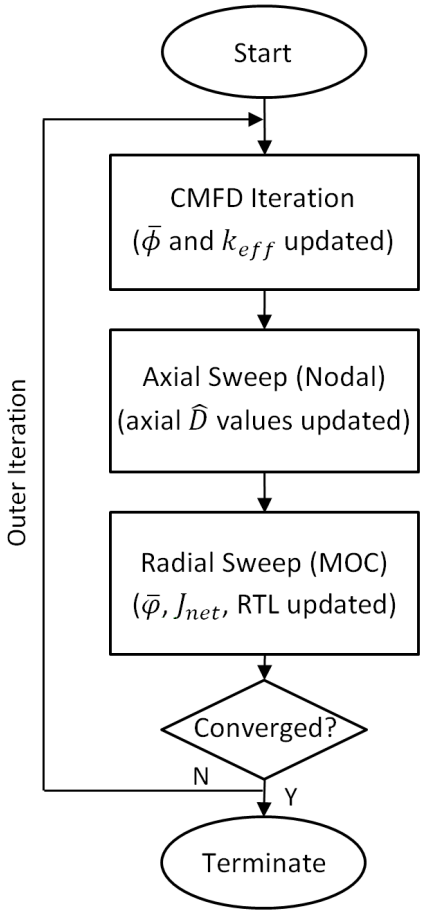

Figure 3.5: 2D/1D Calculation Flow

3.4 Relaxation

In previous work with DeCART, it was observed that the 2D/1D iteration scheme was inherently unstable in models containing thin planes, such as arise when resolving spacer grids and other structural components. The convergence was improved by incorporating a 3D-CMFD solve instead of the planar 2D-CMFD accelerators [55]. However, it was still observed that the scheme was not unconditionally stable. Further work by Kelley and Larsen [25] performed Fourier analyses to determine a group-dependent relaxation factor (θ_g in Eq. 3.18) to be applied to the iteration scheme, which is determined from the optical thickness of each plane ($\tau_g = \Sigma_{t,g}\Delta z$) and scattering ratio ($c_g = \Sigma_{s,g\to g}/\Sigma_{t,g}$). These values are used to relax the radial transport solution and can be seen in Fig. 3.5 from the previous section. It is important to note here that the equation for λ was empirically obtained through curve fitting.

$$\theta_g = \frac{2}{2 - \left(w_{min,g} + w_{max,g}\right)} \tag{3.18a}$$

$$w_{min,g} = \min\left(0, \frac{\left(1 + \frac{\lambda_{min,g}^2}{3}\right)\zeta_{min,g}\frac{tan^{-1}(\lambda_{min,g})}{\lambda_{min,g}} - \zeta_{min,g}}{\left(1 + \frac{\lambda_{min,g}^2}{3}\right) - \zeta_{min,g}}\right) \tag{3.18b}$$

$$w_{max,g} = \max\left(0, \frac{\left(1 + \frac{\lambda_{max,g}^2}{3}\right)\zeta_{max,g}\frac{tan^{-1}(\lambda_{max,g})}{\lambda_{max,g}} - \zeta_{max,g}}{\left(1 + \frac{\lambda_{max,g}^2}{3}\right) - \zeta_{max,g}}\right)$$

$$\begin{aligned} \zeta_{min,g} &= c_g \\ \zeta_{max,g} &= c_g - \frac{4}{3\tau_g^2} \end{aligned} \tag{3.18c}$$

$$\begin{aligned} \lambda_{min,g} &= a_1 + a_2(a_3 - \zeta_{min,g})^{a_4} \\ \lambda_{max,g} &= a_1 + a_2(a_3 - \zeta_{max,g})^{a_4} \\ a_1 &= 1.93801895412889 \\ a_2 &= 1.88037759461481 \\ a_3 &= 1.07821249297909 \\ a_4 &= 0.48797583713968 \end{aligned} \tag{3.18d}$$

Using these equations, relaxation factors are calculated for each pin on each plane. For each group, the minimum value for any particular pin is used to relaxation the entire solution. Figure 3.6 shows the minimum optimal, energy-dependent relaxation factors, which are a function of the axial optical thickness (τ_g) of each pin and the scattering ratio (c_g), for a toy problem. In this figure, two different sets of relaxation factors are shown, the first for a larger axial mesh of 7.6 cm and the second for a more refined mesh at 1.0 cm. However, only the optical thickness for the 1 cm mesh is shown. In this case, transport-corrected scattering was used, so the self-scattering cross section and scattering ratio can be negative. Additionally, note that the optical thickness increases considerably at lower energies, since the total cross section is larger, mostly due to the larger fission cross section at thermal energies. Given what is known about the instabilities occur because of optically thin regions from previous work, it makes sense that more relaxation is necessary in fast energy groups, where the optical thickness is lower. As expected, the relaxation factors for the 1 cm mesh are significantly lower than with the 7.6 cm mesh. It is also notable that the theory suggests that some energy groups can be overrelaxed (i.e. $\theta_g > 1$). In the current

implementation, the relaxation factors are restricted to be less than or equal to one, so no overrelaxation is permitted.

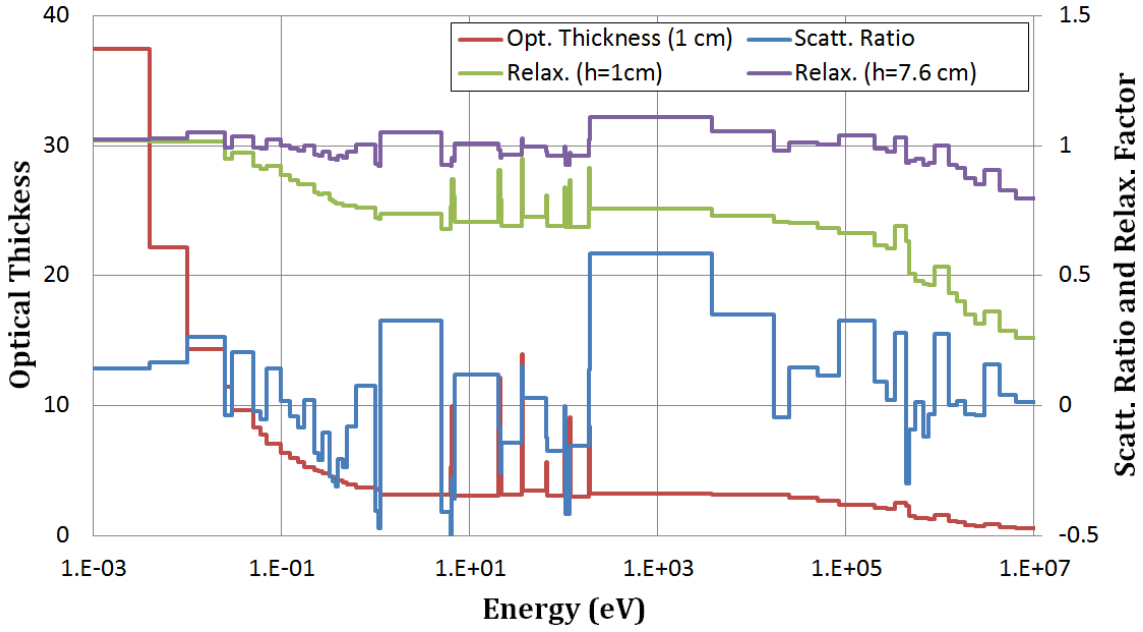

Figure 3.6: Group-Dependent Optimal Relaxation Factors

Figure 3.7 shows the 2-norm residual of the flux distribution as a function of iteration index. The light blue line shows the trend without any relaxation ($\theta = 1$). As can be seen, the residual approaches $1\text{x}10^{-5}$, then begins to diverge. The dark blue line shows the trend using a group-independent relaxation based on the minimum of all of the groups. The initial attempts at relaxation that occurred before the Fourier analyses focused on applying a constant, user-specified relaxation factor to all of the groups. In this case, similar behavior is observed where the flux residual monotonically decreases, but it takes a significant number of iterations. Lastly, the red line shows the residual with the group-dependent, optimal relaxation factor. A significant reduction in the number of iterations is seen compared to using a group-independent value.

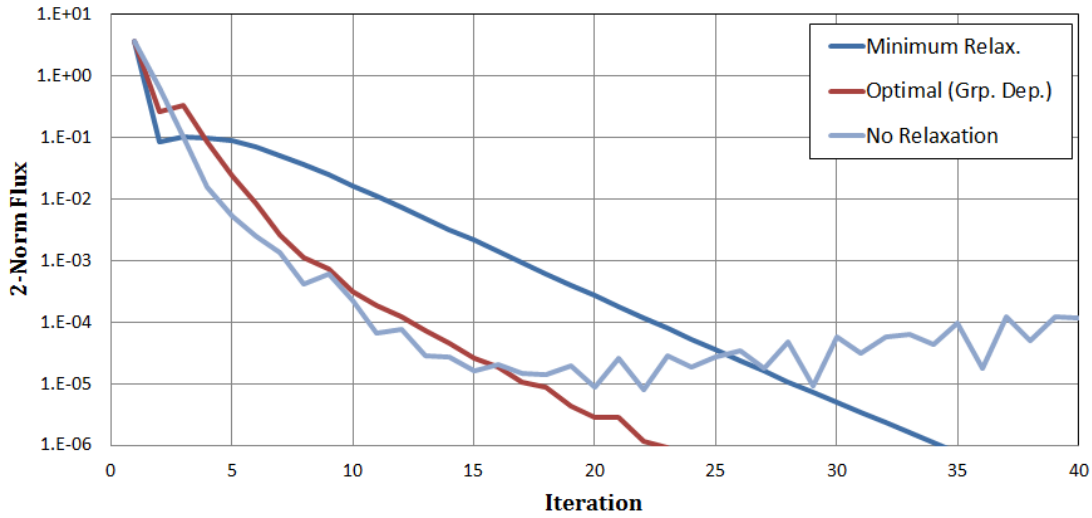

Figure 3.7: Convergence Rate of Various Relaxation Schemes

The relaxation techniques covered in this section have become an extremely important component of the 2D/1D implementation in MPACT, allowing for explicit axial representation of spacer grids and structural components [16], including Intermediate Flow Mixing (IFM) grids, which can be even smaller than typical spacer grids [14]. However, in the cases analyzed in Chapter 5, these relaxation techniques were turned off and do not seem to be necessary, as some cases were refined to below a 1 cm axial mesh without any convergence issue.

The Fourier analysis that yielded these relaxation factors were performed exclusively for a system using a finite difference axial solver without axial sweeping. Further analysis was performed for an NEM axial solver, which yielded consistently higher values. It was decided to keep the relaxation factors as were determined for finite difference as they would still be conservative in the case of NEM [27]. It is not clear that relaxation is necessary for the SP_N or S_N solvers because several axial sweeps are performed to help resolve the boundary conditions for each of the axial nodes. Additional analysis is necessary to confirm this, but there is currently no theoretical reason why relaxation should be applied to the methods in this work.

3.5 Angular Distribution Scaling

With isotropic transverse leakages, the surface currents that are generated directly by the 3D-CMFD solver are used to construct the axial and radial transverse leakages. However, CMFD has no sense of the angular distribution of the axial and radial leakages which are

calculated by the radial and axial solvers, respectively, after the CMFD solve. If the surface angular fluxes that construct the angular dependence of the leakages are integrated over all angles, one will find that the currents these generate can be different that the currents calculated by CMFD. This can cause problems with convergence since the systems now do not agree on what the neutron balance should be. To remedy this, the angle-dependent leakage distributions are scaled in such a way to preserve value of the isotropic transverse leakages that would be constructed using the currents. Here $XY||Z$ indicates that the transverse leakage could be either the axial or radial leakage.

$$TL_{k,g)}^{XY||Z} = TL_{k,g}^{XY||Z} + \left(TL_{iso}^{XY||Z} - \sum_{k=1}^{N_{ang}} w_k TL_{k,g}^{XY||Z} \right) \tag{3.19}$$

As the problem converges, this scaling factor approaches zero as CMFD and the transport solvers become more in agreement.

3.6 Sources of Error

The 2D/1D method has a number of sources of error that come into play in any particular problem, sometimes making it difficult to truly assess the improvement that individual components provide. Generally, if one of the largest sources of error is eliminated, improvement is observed, but when eliminating some of the smaller error components, there could be a reduction in the cancellation of error, yielding worse results, despite the improvement to the method. The first few sources are related to all 2D cases and the remaining few are specific to components of 2D/1D.

1. Radial Discretization

The radial discretization encompasses anything related to the spatial mesh used for each plane, including the number of fuel rings, the number of azimuthal divisions, etc. There are also situations where it is advantageous to use a different meshing scheme, such as a Cartesian grid, for reflector regions. Especially since the radial 2D-MOC solver uses step-characteristics (which means the source and flux within each fine mesh region are flat), the radial mesh can introduce non-negligible error unless a sufficiently refined mesh is used.

2. Angular Quadrature

There are two main components of the angular quadrature, including the number of azimuthal and polar angles used as well as the optimized quadrature sets and weights used in the configuration (Bickley, Yamamoto, Gauss, Chebyshev). Additionally, any changes introduced through modularization (see Section 2.2.1.1) will introduce some error.

3. Ray Spacing

The distance between the rays used by MOC, which is also slightly modified during modularization, can be a significant source of error, particularly considering the volume correction, which modifies the track length to ensure consistent volumetric representation. More refined ray spacings not only provide a more accurate simulation of the geometry, but minimizes the volume correction that would be necessary. In some scenarios, volume correction, though intended to be more consistent, can actually introduce more error.

4. Cross Section Library

For realistic problems, the cross section library can be one of the largest sources of error. In Section 2.1.3, the multigroup approximation was introduced and as was seen in Figure 2.2, the cross sections can be highly energy-dependent. Approximating this dependence with typical tens of groups (40-60 groups are common) can introduce considerable errors if not done carefully. On top of the multigroup approximation, there are generally several other approximations used in generating a cross section library. In recent years, more attention has been given to using Monte Carlo to generate libraries for deterministic codes [37], which help reduce these approximations. Fortunately, the results presented in this work are based on predefined benchmark cross sections and are compared to Monte Carlo results also using the same cross sections, so there is no error introduced here.

5. Scattering Order

In Chapter 2, the fundamentals of anisotropic and transport-corrected scattering were introduced. These inherently introduce some error, either from truncating the expansion order with anisotropic scattering or because of the limitations of the approximation, as with transport-corrected scattering.

6. Self-Shielding

Though the details of the resonance absorption methods was not covered in previous chapters, when using realistic cross section libraries, a self-shielding calculation is necessary to accurately estimate the equivalence cross section. In MPACT, there are two methods that handle this: the subgroup method and the embedded self-shielding method (ESSM) [35]. All of the parameters covered in points 1-4 affect the accuracy of the self-shielding method. Again, since this work uses benchmark cross sections, a self-shielding calculation is unnecessary and no error is introduced here.

This concludes the first set of sources of error, which is present even in 2D cases. The rest of the list focuses on errors that are specific to 2D/1D.

7. Axial Discretization

Just as the radial mesh discretization is an important factor, the axial mesh discretization is significant. In general, since nodal methods are typically used, which use higher order spatial expansions, the errors introduced with the axial mesh are considerably less than would be introduced with a radial mesh of the same size. It is also possible to minimize the error from the axial mesh by incorporating the subplane method, which allows for multiple axial mesh nodes per radial plane. This method has not been implemented in this work, but is planned for future development and has been successfully used in other instances of the 2D/1D method [6, 19, 20].

8. Axial Transverse Leakage Spatial Dependence

As previously discussed, one of the major approximations in the current 2D/1D implementation is that the axial transverse leakage shape is considered to be spatially flat and uniform across a each pin, meaning that all of the fine mesh regions in each pin are given the same axial leakage value. This is typical, since the axial solver is using pin-homogenized values and outputs surface flux moment data on the axial surface, which encompass complete pins. Accurate representation of the leakage shape for heterogeneous pins would likely need to have a different leakage in each of the distinct material regions. Encompassed within this is the subtlety of the differences between the "fine" and "coarse" grids. If the fine and coarse meshes are identical, then there is no additional error introduced here.

In the results in Chapter 5, it is observed that the flat axial leakage distribution, which is one of the only remaining sources of error is very significant and should be addressed in future research.

9. Axial Transverse Leakage Angular Dependence

The angular distribution of the axial transverse leakage is one of the major approximations under investigation in this work. Typically, the leakage is assumed to be isotropic and can be constructed using the net currents at the top and bottom surfaces of each pin (see Eq. 3.4). While this generally performs reasonably well, more complicated problems such as those with control rods or mixed oxide (MOX) fuels require higher order angular representation to achieve accurate results.

10. Radial Transverse Leakage Spatial Dependence

Unlike the axial transverse leakage, which is assumed to be flat, the radial transverse leakage is constructed using a quadratic Legendre expansion from the radial transverse leakage values on the neighboring planes (see Section 3.1.2.3). While this should be more accurate than using a flat distribution, this can introduce error in situations where a quadratic expansion is not sufficient or where a non-physical leakage distribution is produced [5].

11. Radial Transverse Leakage Angular Dependence

Similar to the angular dependence of the axial transverse leakage, it is typical to assume the leakage is isotropic. This is problematic for more complicated problems. In this work, the angular dependence of both the axial and radial transverse leakages are effectively analyzed in tandem. Initial work focused on just increasing the fidelity of the radial transverse leakage, but it can be found that the azimuthal dependence is essentially integrated out unless the axial transverse leakage also includes higher fidelity angular dependence. However, it is important to note that the error introduced should be considered from a separate source than the axial transverse leakage angular distribution.

12. Radial/Axial Transverse Leakage Splitting

In Section 3.2, the importance of and techniques for transverse leakage splitting were presented. When splitting isotropic leakages, the impact on accuracy is not very severe. However, when splitting angle-dependent leakages, the technique of splitting a constant, isotropic component that is to be applied to all angles can be detrimental to accuracy.

In the results presented later, transverse leakage splitting was fortunately not necessary to obtain results with angle-dependent transverse leakages. However, this would likely not be true for all cases. Several other isotropic transverse leakage splitting techniques were evaluated by Hursin [19], but more work is necessary to develop accurate techniques that are computationally feasible. The most explicit option would be to have a splitting value that is different for each angle, but would require storing the angular fluxes for each angle. However, this is not a practical solution.

13. Anisotropic Total Cross Section Dependence

It has been stated several times that the axial solve is performed on the pin cell-homogenized basis. With this approach, data from the radial solver is homogenized to formulate the cross sections by using flux-volume homogenization using the scalar flux (similar to Eq. 2.33a, but repeated here):

$$\overline{\Sigma}^{n}_{x,g} = \frac{\sum\limits_{i \in n} \Sigma^{i}_{x,g} V^{i} \phi^{i}_{g}}{\sum\limits_{i \in n} V^{i} \phi^{i}_{g}}. \tag{3.20}$$

However, a more accurate homogenization of the total cross section can be performed by homogenizing with the angular fluxes instead of the scalar flux, yielding an angle-dependent total cross section:

$$\overline{\Sigma}^{n}_{t,k,g} = \frac{\sum\limits_{i \in n} \Sigma^{i}_{t,g} V^{i} \varphi^{i}_{k,g}}{\sum\limits_{i \in n} V^{i} \varphi^{i}_{k,g}}. \tag{3.21}$$

The effect of this simplification was investigated in a modified OECD/NEA benchmark by KAIST in the 2D/1D Fusion code CRX [9]. In their results, the eigenvalue difference observed was less than 10 pcm and the difference in the maximum power were negligble. Additionally, a similar correction was investigated by Young [63] within the context of 2D/3D. The correction is important in reproducing the 2D-MOC solution on a coarser grid S_N solver. When applied to the 3D-C5G7 extended benchmarks, a 20-60 pcm difference was observed as a result of the correction, but with only a marginal affect on the pin power distribution. With respect to 2D/1D, one would expect this error to be lower, since it only affects the 1D axial solver, whereas in the 2D/3D work, the correction impacts the entire 3D-S_N solve. In cases with homogenized pins, some of which are analyzed in this work, the simplification is not necessary and no error is introduced as a result.

3.7 Parallel Decomposition Scheme

This section covers the various parallel decomposition schemes available in MPACT, which take advantage of both MPI [41] and OpenMP [45], as well as third party linear solvers such as PETSc [1].

3.7.1 Spatial Decomposition

Spatial decomposition covers any partitioning of the geometry, which includes both planar and radial partitioning. In MPACT, the initial preference is to partition planarly, as this requires the least overhead for the radial sweepers, which can be solved independently on a plane-by-plane basis. However, planar decomposition requires data passing for the axial solvers. This includes the net current, coarse mesh flux, radial transverse leakage, and cross section data for the two-node sweepers and radial transverse leakage and partial currents or angular fluxes for the one-node sweepers. Additionally, if finite difference is being used axially where CMFD is driving the axial transport, no additional data passing is necessary for the axial solver. Radial decomposition does not burden the axial solvers, but the radial sweepers must pass angular flux data along the radial decomposition interfaces. All communication with spatial decomposition is handled through MPI interfaces.

Both planar and radial decomposition affect the CMFD solver, which will pass coarse mesh fluxes, currents, and cross sections between neighboring processors to build the

coupling coefficients for the linear system. PETSc (Portable, Extensible Toolkit for Scientific Computing) is used to handle the setup and solution of the parallel linear system governed by CMFD.

3.7.2 Angle Decomposition

Angle decomposition processors are predominantly used to decompose the azimuthal angles for the MOC sweeper. There are several different partitioning schemes that aim to divide the work up more evenly, which can be an issue since each angle has a different number of ray segments, and thus, a different amount of work. Angle decomposition can be used for all transport sweepers, including the axial S_N sweeper, though the parallelization of this is covered more in Chapter 4. All communication for angle decomposition is handled with MPI, as well.

The CMFD solver can also take advantage of the angle decomposition processors, despite the fact that it does not include any angular variables. All of the angle processors belonging to a particular spatial decomposition domain can be used to further decompose the problem spatially. This is especially beneficial when solving larger systems where spatial domains might include multiple assemblies.

3.7.3 Ray Decomposition

Ray decomposition is the only parallel scheme in MPACT that uses OpenMP and is used to thread the long rays used in the MOC sweep. Ray assignment is handled dynamically as each thread finishes tracing the ray. This scheme can be particularly used for memory-bound cases where adding angular decomposition processors will be problematic as all threads shared a large majority of the data (with only small overhead for thread-specific constructs as necessary).

CHAPTER 4

The Axial S_N Sweeper

In Chapter 2, many of the fundamentals of several other axial solvers (NEM, SENM, SP_N) were covered and in Chapter 3, the governing equations for the axial solvers and corresponding transverse leakages were presented. This chapter is devoted to the axial S_N sweeper, which uses cubic characteristics to handle the spatial distribution and several different options for the angular dependence of the transverse leakages. Additionally, a Fourier moment-based approximation for the azimuthal dependence of the angular fluxes and transverse leakages is proposed and a new axial solver using these expansions is derived.

4.1 Spatial Moments

The derivation of the spatial moments for the axial S_N sweeper starts by considering the one-dimensional transport equation on a normalized spatial basis ($s = [0,1]$ because $l = \frac{h}{\mu}$) with all sources lumped into one source term $Q(s)$ (Eq. 4.1a), substituting cubic Legendre expansions for the flux and source (Eqs. 4.1b and 4.1c). The derivation process here is similar to what was shown in Section 2.2.1 for step characteristic MOC, in which can the expansions here would only have a zeroth order, flat spatial expansion:

$$\frac{1}{l}\frac{\partial \varphi}{\partial s} + \Sigma_t \varphi(s) = Q(s), \tag{4.1a}$$

$$\varphi(s) = \varphi_0 P_0(2s-1) + \varphi_1 P_1(2s-1) + \varphi_2 P_2(2s-1) + \varphi_3 P_3(2s-1), \tag{4.1b}$$

$$Q(s) = q_0 P_0(2s-1) + q_1 P_1(2s-1) + q_2 P_2(2s-1) + q_3 P_3(2s-1). \tag{4.1c}$$

Solving this equation gives us the angular flux at any point s (Eq. 4.2):

$$
\begin{aligned}
\varphi(s) &= \varphi_{in} e^{-\tau} \\
&+ q_0 \frac{1 - e^{-\tau}}{\Sigma_t} \\
&+ q_1 \frac{\left(\tau((2s-1) + e^{-\tau}) + 2e^{-\tau} - 2\right)}{\tau \Sigma_t} \\
&+ q_2 \frac{\left(12(1 - e^{-s\tau}) - 6\tau((2s-1) + e^{-s\tau}) + \tau^2(6s^2 - 6s + 1) - e^{-s\tau}\right)}{\tau^2 \Sigma_t} \\
&+ q_3 \frac{\left(-120(1 - e^{-s\tau}) + 60\tau((2s-1) + e^{-s\tau}) - 12\tau^2((5s^2 - 5s + 1) - e^{-s\tau})\right)}{\tau^3 \Sigma_t} \\
&+ q_3 \frac{\tau^3((20s^3 - 30s^2 + 12s - 1) + e^{-s\tau})}{\tau^3 \Sigma_t},
\end{aligned}
\tag{4.2}
$$

In this and the following equations in this section, τ is defined as the optical thickness of the characteristic ray segment ($\tau = l\Sigma_t$). To determine the outgoing angular flux, Eq. 4.2 simply needs to be evaluated at $s = 1$:

$$
\begin{aligned}
\varphi_{out} &= \varphi(1) = \varphi_{in} e^{-\tau} \\
&+ q_0 \frac{1 - e^{-\tau}}{\Sigma_t} \\
&+ q_1 \frac{\left(\tau(1 + e^{-\tau}) + 2e^{-\tau} - 2\right)}{\tau \Sigma_t} \\
&+ q_2 \frac{\left(12(1 - e^{-\tau}) - 6\tau(e^{-\tau}) + \tau^2(-6) - e^{-\tau}\right)}{\tau^2 \Sigma_t} \\
&+ q_3 \frac{\left(-120(1 - e^{-\tau}) + 60\tau(e^{-\tau}) - 12\tau^2((-5) - e^{-\tau}) + \tau^3((-10) + e^{-\tau}\right)}{\tau^3 \Sigma_t}.
\end{aligned}
\tag{4.3}
$$

With step characteristics, only one average angular flux value along each segment is necessary. However, with a higher order expansion, an average flux for each moment needs to be calculated. If Eq. 4.2 is multiplied by the corresponding Legendre polynomial and integrated from $s = 0$ to 1 (as in Eq. 4.4), the j^{th} moment angular flux can be found:

$$
\overline{\varphi}_j = \frac{\int_0^1 P_j(2s-1)\varphi_(s)ds}{\int_0^1 P_j(2s-1)P_j(2s-1)ds}.
\tag{4.4}
$$

Eqs. 4.5 show the final solutions of the angular flux for each moment:

$$\overline{\varphi}_0 = \frac{\int_0^1 P_0(2s-1)\varphi_(s)ds}{\int_0^1 P_0(2s-1)P_0(2s-1)ds}$$

$$\begin{aligned}
&= \varphi_{in}\frac{1-e^{-\tau}}{\tau} \\
&+ q_0\frac{(e^{-\tau}-1)+\tau}{\tau\Sigma_t} \\
&+ q_1\frac{\left(2-2e^{-\tau}-\tau(1+e^{-\tau})\right)}{\tau^2\Sigma_t} \\
&+ q_2\frac{-12(1-e^{-\tau})+6\tau(1+e^{-\tau})-\tau^2(1-e^{-\tau})}{\tau^3\Sigma_t} \\
&+ q_3\frac{3(\tau+2)(-120(1-e^{-\tau})+60\tau(1+e^{-\tau})-12\tau^2(1-e^{-\tau})+\tau^3(1+e^{-\tau})}{\tau^5\Sigma_t},
\end{aligned} \tag{4.5a}$$

$$\overline{\varphi}_1 = \frac{\int_0^1 P_1(2s-1)\varphi_(s)ds}{\int_0^1 P_1(2s-1)P_1(2s-1)ds}$$

$$\begin{aligned}
&= \varphi_{in}\frac{6-6e^{-\tau}-\tau(3e^{-\tau}+3)}{\tau^2} \\
&+ q_0\frac{3\left(2e^{-\tau}-2+\tau(1+e^{-\tau})\right)}{\tau^2\Sigma_t} \\
&+ q_1\frac{\left(12(1-e^{-\tau})-12\tau e^{-\tau}-3(1+e^{-\tau})\tau^2+\tau^3\right)}{\tau^3\Sigma_t} \\
&- q_2\frac{3\left(\tau+2)(12(1-e^{-\tau})-6\tau(1+e^{-\tau})+(1-e^{-\tau})\tau^2\right)}{\tau^4\Sigma_t} \\
&- q_3\frac{3\left(\tau+2)(-120(1-e^{-\tau})+60\tau(1+e^{-\tau})-12(1-e^{-\tau})\tau^2)+(1+e^{-\tau})\tau^3\right)}{\tau^5\Sigma_t},
\end{aligned} \tag{4.5b}$$

$$\overline{\varphi}_2 = \frac{\int_0^1 P_2(2s-1)\varphi_(s)ds}{\int_0^1 P_2(2s-1)P_2(2s-1)ds}$$

$$\begin{aligned}
&= \varphi_{in}\frac{5(12(1-e^{-\tau})-6\tau(1+e^{-\tau})+(1-e^{-\tau})\tau^2)}{\tau^3} \\
&+ q_0\frac{5(-12(1-e^{-\tau})+6\tau(1+e^{-\tau})-(1-e^{-\tau})\tau^2)}{\tau^3\Sigma_t} \\
&+ q_1\frac{5(\tau+2)(12(1-e^{-\tau})-6\tau(1+e^{-\tau})+(1-e^{-\tau})\tau^2)}{\tau^4\Sigma_t} \\
&+ q_2\frac{(-720(1-e^{-\tau})+720\tau e^{-\tau}+60(1+5e^{-\tau})\tau^2+60\tau^3e^{-\tau}-5(1-e^{-\tau})\tau^4+\tau^5))}{\tau^5\Sigma_t} \\
&- q_3\frac{5(12+6\tau+\tau^2)(-120(1-e^{-\tau})+60\tau(1+e^{-\tau})-12(1-e^{-\tau})\tau^2+(1+e^{-\tau})\tau^3)}{\tau^6\Sigma_t},
\end{aligned} \tag{4.5c}$$

$$\overline{\varphi}_3 = \frac{\int_0^1 P_3(2s-1)\varphi_(s)ds}{\int_0^1 P_3(2s-1)P_3(2s-1)ds}$$

$$\begin{aligned}
&= -\varphi_{in}\frac{7(-120(1-e^{-\tau})+60\tau(1+e^{-\tau})-12(1-e^{-\tau})\tau^2+(1+e^{-\tau})\tau^3)}{\tau^4} \\
&+ q_0\frac{7(-120(1-e^{-\tau})+60\tau(1+e^{-\tau})-12(1-e^{-\tau})\tau^2+(1+e^{-\tau})\tau^3)}{\tau^4\Sigma_t} \\
&- q_1\frac{7(\tau+2)(-120(1-e^{-\tau})+60\tau(1+e^{-\tau})-12(1-e^{-\tau})\tau^2+(1+e^{-\tau})\tau^3}{\tau^5\Sigma_t} \\
&+ q_2\frac{7(12+6\tau+\tau^2)(-120(1-e^{-\tau})+60\tau(1+e^{-\tau})-12(1+e^{-\tau})\tau^2)+(1+e^{-\tau})\tau^3)}{\tau^6\Sigma_t} \\
&+ q_3\frac{(100,800(1-e^{-\tau}-\tau e^{-\tau})-5,040(1+9e^{-\tau})\tau^2-11,760\tau^3e^{-\tau}}{\tau^7\Sigma_t} \\
&+ q_3\frac{168(1-11e^{-\tau})\tau^4-168\tau^5e^{-\tau}-7(1+e^{-\tau})\tau^6+\tau^7)}{\tau^7\Sigma_t}.
\end{aligned} \tag{4.5d}$$

For simplicity, the outgoing and average angular flux equations can be rewritten in terms of condensed in terms of functions:

$$\varphi_{out} = \varphi_{in}e^{-\tau} + \sum_{i=0}^{N_{mom}} q_i f_{out,i}(h,\mu,\Sigma_t), \tag{4.6a}$$

$$\overline{\varphi}_j = \varphi_{in} f_{j,in}(h,\mu,\Sigma_t) + \sum_{i=0}^{N_{mom}} q_i f_{j,i}(h,\mu,\Sigma_t). \tag{4.6b}$$

4.2 Azimuthal Moments

4.2.1 Important Identities

The derivation of the azimuthal, Fourier moment-based quantities makes use of several important trigonometric identities that are presented here:

$$\int_0^{2\pi} \sin(n\alpha) d\alpha = 0, \tag{4.7a}$$

$$\int_0^{2\pi} \cos(n\alpha) d\alpha = 0, \tag{4.7b}$$

$$\int_0^{2\pi} \sin(m\alpha)\cos(n\alpha) d\alpha = 0, \text{for any } m \text{ or } n, \tag{4.7c}$$

$$\int_0^{2\pi} \sin(m\alpha)\sin(n\alpha) d\alpha = \begin{cases} 0, m \neq n \\ \pi, m = n \end{cases}, \tag{4.7d}$$

$$\int_0^{2\pi} \cos(m\alpha)\cos(n\alpha) d\alpha = \begin{cases} 0, m \neq n \\ \pi, m = n \end{cases}. \tag{4.7e}$$

They will prove useful in eliminating various terms as necessary.

4.2.2 Axial Sweeper Derivation

The derivation begins with the multigroup transport equation, continuous in both space and angle with isotropic scattering and fission sources lumped into the source term (Q_g) and constant material properties:

$$\begin{aligned} \mu \frac{d\varphi_g(\boldsymbol{x},\alpha,\mu)}{dz} + \Sigma_{t,g}\varphi_g(\boldsymbol{x},\alpha,\mu) &= \frac{Q_g(\boldsymbol{x})}{4\pi} \\ &+ \Big(\sqrt{1-\mu^2}\cos(\alpha)\frac{d\varphi_g(\boldsymbol{x},\alpha,\mu)}{dx} + \sqrt{1-\mu^2}\sin(\alpha)\frac{d\varphi_g(\boldsymbol{x},\alpha,\mu)}{dy}\Big), \end{aligned} \tag{4.8a}$$

$$Q_g(\boldsymbol{x}) = \sum_{g'=1}^{N_{grp}} \Sigma_{s,g'\to g}\phi_g(\boldsymbol{x}) + \frac{\chi_g}{k_{\text{eff}}} \sum_{g'=1}^{N_{grp}} \nu\Sigma_{f,g'}\phi_g(\boldsymbol{x}). \tag{4.8b}$$

To eliminate the radial components and generate the equations for the axial solver, Eq. 4.8a will be averaged radially over defined bounds ($y = [y_L, y_R]$ and $x = [x_L, x_R]$). This is similar to the process in Section 3.1.2.1:

$$\begin{aligned}\frac{1}{A_{xy}}\int_{y_l}^{y_r}\int_{x_l}^{x_r}\left(\mu\frac{\partial\varphi_g(\boldsymbol{x},\alpha,\mu)}{\partial z}+\Sigma_{t,g}\varphi_g(\boldsymbol{x},\alpha,\mu)-\frac{Q_g(\boldsymbol{x})}{4\pi}\right)dxdy = \\ +\frac{\sqrt{1-\mu^2}}{A_{xy}}\int_{y_l}^{y_r}\int_{x_l}^{x_r}\left(cos(\alpha)\frac{\partial}{\partial x}\varphi_g(\boldsymbol{x},\alpha,\mu)\right)dxdy \\ +\frac{\sqrt{1-\mu^2}}{A_{xy}}\int_{y_l}^{y_r}\int_{x_l}^{x_r}\left(sin(\alpha)\frac{\partial}{\partial y}\varphi_g(\boldsymbol{x},\alpha,\mu)\right)dxdy.\end{aligned} \tag{4.9}$$

For clarity, the radially-integrated flux and source quantities are defined as the following:

$$\varphi_g^{XY}(z,\alpha,\mu) = \frac{1}{A_{xy}}\int_{y_l}^{y_r}\int_{x_l}^{x_r}\varphi_g(\boldsymbol{x},\alpha,\mu)dxdy,$$

$$Q_g^{XY}(z) = \frac{1}{A_{xy}}\int_{y_l}^{y_r}\int_{x_l}^{x_r}Q_g(\boldsymbol{x})dxdy.$$

Substituting these expressions and rearranging the transverse leakage terms yields Eq. 4.10:

$$\begin{aligned}\mu\frac{\partial\varphi_g^{XY}(z,\alpha,\mu)}{\partial z}+\Sigma_{t,g}\varphi_g^{XY}(z,\alpha,\mu)-\frac{Q_g^{XY}(z)}{4\pi} = \\ +\frac{\sqrt{1-\mu^2}}{A_{xy}}cos(\alpha)\int_{y_l}^{y_r}\int_{x_l}^{x_r}\frac{\partial}{\partial x}\varphi_g(\boldsymbol{x},\alpha,\mu)dxdy \\ +\frac{\sqrt{1-\mu^2}}{A_{xy}}sin(\alpha)\int_{y_l}^{y_r}\int_{x_l}^{x_r}\frac{\partial}{\partial y}\varphi_g(\boldsymbol{x},\alpha,\mu)dxdy.\end{aligned} \tag{4.10}$$

One of the integrals in each transverse leakage term can be evaluated:

$$\mu\frac{\partial\varphi_g^{XY}(z,\alpha,\mu)}{\partial z}+\Sigma_{t,g}\varphi_g^{XY}(z,\alpha,\mu) = \frac{Q_g^{XY}(z)}{4\pi}+TL^{XY}(z,\alpha,\mu), \tag{4.11a}$$

$$\begin{aligned}TL_g^{XY}(z,\alpha,\mu) = \frac{\sqrt{1-\mu^2}}{A_{xy}}\Bigg(\int_{y_l}^{y_r}cos(\alpha)\Big(\varphi_g(x_R,y,z,\alpha,\mu)-\varphi_g(x_L,y,z,\alpha,\mu)\Big)dy \\ +\int_{x_l}^{x_r}sin(\alpha)\Big(\varphi_g(x,y_R,z,\alpha,\mu)-\varphi_g(x,y_L,z,\alpha,\mu)\Big)dx\Bigg).\end{aligned} \tag{4.11b}$$

Eqs. 4.11a and 4.11b are consistent with the derivation for the axial sweeper with

explicit radial transverse leakage (both azimuthal and polar dependence) done by Hursin et. al. [20], but now things can get a bit more interesting. To consider a Fourier moment expansion, both the angular flux and radial transverse leakage terms need to be expanded, since they are governed by different sweepers:

$$\varphi_g^{XY}(z,\alpha,\mu) = \frac{1}{2\pi}\varphi_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\Big(\varphi_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + \varphi_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\Big), \tag{4.12a}$$

$$TL_g^{XY}(z,\alpha,\mu) = \frac{1}{2\pi}TL_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\Big(TL_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + TL_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\Big) \tag{4.12b}$$

A similar concept was employed by the DIT code [22, 23], which used a first-order Fourier expansion to resolve the spatial distribution of the scalar flux with the collision probabilities solver. Substituting these expansions yields Eq. 4.13, which will be the starting point for deriving the equations for each moment specifically:

$$\begin{aligned}
&\mu\frac{d}{dz}\left(\frac{1}{2\pi}\varphi_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\Big(\varphi_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + \varphi_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\Big)\right)\\
&+\Sigma_{t,g}\left(\frac{1}{2\pi}\varphi_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\Big(\varphi_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + \varphi_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\Big)\right) = \frac{Q_g^{XY}(z)}{4\pi}\\
&+\left(\frac{1}{2\pi}TL_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\Big(TL_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + TL_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\Big)\right).
\end{aligned} \tag{4.13}$$

The equation for each moment component can be obtained by multiplying 4.13 by the corresponding Fourier moment basis and integrating azimuthally. In the following subsections, the equations for the zeroth, sine, and cosine components will be evaluated.

4.2.2.1 Zeroth-Moment

To formulate the equation for the zeroth-moment, Eq. 4.13 simply needs to be integrated azimuthally, as in Eq. 4.14:

$$\int_0^{2\pi} \mu \frac{d}{dz}\left(\frac{1}{2\pi}\varphi_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\left(\varphi_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + \varphi_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\right)\right)d\alpha +$$
$$\int_0^{2\pi} \Sigma_{t,g}\left(\frac{1}{2\pi}\varphi_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\left(\varphi_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + \varphi_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\right)\right)d\alpha = \int_0^{2\pi}\frac{Q_g^{XY}(z)}{4\pi}d\alpha$$
$$-\int_0^{2\pi}\left(\frac{1}{2\pi}TL_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\left(TL_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + TL_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\right)\right)d\alpha. \tag{4.14}$$

Removing the terms that integrate to zero by using Eq. 4.7a-4.7e:

$$\int_0^{2\pi}\frac{\mu}{2\pi}\frac{d}{dz}\varphi_{g,0}^{XY}(z,\mu)d\alpha + \int_0^{2\pi}\frac{\Sigma_{t,g}}{2\pi}\varphi_{g,0}^{XY}(z,\mu)d\alpha = \int_0^{2\pi}\frac{Q_g^{XY}(z)}{4\pi}d\alpha + \int_0^{2\pi}\frac{1}{2\pi}TL_{g,0}^{XY}(z,\mu)d\alpha. \tag{4.15}$$

Evaluating the integrals, again using the identities in the previous section, the final form of the zeroth-moment equation is obtained:

$$\mu\frac{d}{dz}\varphi_{g,0}^{XY}(z,\mu) + \Sigma_{t,g}\varphi_{g,0}^{XY}(z,\mu) = \frac{Q_g^{XY}(z)}{2} + TL_{g,0}^{XY}(z,\mu). \tag{4.16}$$

4.2.2.2 Sine Moments

Multiplying Eq. 4.13 by $\sin(n\alpha)$ and integrating over α will yield the equations for the sine moments:

$$\int_0^{2\pi} \sin(n\alpha)\mu\frac{d}{dz}\left(\frac{1}{2\pi}\varphi_{g,0}^{XY}(z,\mu)+\frac{1}{\pi}\sum_{m=1}^{N_{mom}}\left(\varphi_{g,sm}^{XY}(z,\mu)\sin(m\alpha)+\varphi_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\right)\right)d\alpha+$$
$$\int_0^{2\pi} \sin(n\alpha)\Sigma_{t,g}\left(\frac{1}{2\pi}\varphi_{g,0}^{XY}(z,\mu)+\frac{1}{\pi}\sum_{m=1}^{N_{mom}}\left(\varphi_{g,sm}^{XY}(z,\mu)\sin(m\alpha)+\varphi_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\right)\right)d\alpha =$$
$$\int_0^{2\pi} \sin(n\alpha)\frac{Q_g^{XY}(z)}{4\pi}d\alpha$$
$$+\int_0^{2\pi} \sin(n\alpha)\left(\frac{1}{2\pi}TL_{g,0}^{XY}(z,\mu)+\frac{1}{\pi}\sum_{m=1}^{N_{mom}}\left(TL_{g,sm}^{XY}(z,\mu)\sin(m\alpha)+TL_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\right)\right)d\alpha. \tag{4.17}$$

Removing terms that integrate to zero:

$$\int_0^{2\pi} \sin(n\alpha)\mu\frac{d}{dz}\left(\frac{1}{\pi}\varphi_{g,sn}^{XY}(z,\mu)\sin(n\alpha)\right)d\alpha+\int_0^{2\pi} \sin(n\alpha)\Sigma_{t,g}\left(\frac{1}{\pi}\varphi_{g,sn}^{XY}(z,\mu)\sin(n\alpha)\right)d\alpha =$$
$$+\int_0^{2\pi} \sin(n\alpha)\left(\frac{1}{\pi}TL_{g,sn}^{XY}(z,\mu)\sin(n\alpha)\right)d\alpha, \tag{4.18}$$

and rearranging:

$$\mu\frac{d}{dz}\left(\frac{1}{\pi}\varphi_{g,sn}^{XY}(z,\mu)\int_0^{2\pi}\sin(n\alpha)\sin(n\alpha)d\alpha\right)+\Sigma_{t,g}\left(\frac{1}{\pi}\varphi_{g,sn}^{XY}(z,\mu)\int_0^{2\pi}\sin(n\alpha)\sin(n\alpha)d\alpha\right) =$$
$$+\left(\frac{1}{\pi}TL_{g,sn}^{XY}(z,\mu)\int_0^{2\pi}\sin(n\alpha)\sin(n\alpha)d\alpha\right). \tag{4.19}$$

Evaluating these integrals yields the final form of the equations for the sine moments (Eq 4.20):

$$\mu\frac{d}{dz}\varphi_{g,sn}^{XY}(z,\mu)+\Sigma_{t,g}\varphi_{g,sn}^{XY}(z,\mu) = TL_{g,sn}^{XY}(z,\mu). \tag{4.20}$$

It should be noted that the source term $Q_g^{XY}(z)$ integrates out of the equation. This is entirely expected, since both the fission and scattering sources in this derivation are isotropic. As will be discussed later, if anisotropic scattering is used, each moment will have some

contribution from the scattering source as well.

4.2.2.3 Cosine Moments

Using a similar technique of multiplying Eq. 4.13 by $cos(n\alpha)$ instead of $sin(n\alpha)$ to obtain the equations for the cosine moments:

$$\mu \frac{d}{dz}\varphi^{XY}_{g,cn}(z,\mu) + \Sigma_{t,g}\varphi^{XY}_{g,cn}(z,\mu) = TL^{XY}_{g,cn}(z,\mu). \tag{4.21}$$

It is important to note that this derivation yields moment equations that are completely independent of one another. This is different from the SP_N formulation, in which each moment is dependent on information from the neighboring moments (see Section 2.2.3.3). However, in the SP_N formulation, an expansion of μ is assumed, and when substituting into the axial streaming term, which also has a factor of μ, the subsequent integration produces set of equations that are coupled. Since the Fourier expansion is only dependent on the azimuthal angle (α), the additional μ has no impact on the integration.

4.2.2.4 Boundary Conditions

Since the only boundary conditions that will be considered are vacuum and reflective, the standard treatment of angular flux boundary conditions should hold for all moment equations. This means that at reflective boundaries, the value of the incoming angular flux moment is the same as the outgoing value (Eq. 4.22a) and on vacuum boundaries, the incoming angular flux moment is zero (Eq. 4.22b).

$$\varphi_{in,g,l} = \varphi_{out,g,l} \tag{4.22a}$$

$$\varphi_{in,g,l} = 0 \tag{4.22b}$$

4.2.3 Transverse Leakage Coefficients

In the previous section, the radial transverse leakage (TL^{XY}_g) was defined to be an expansion using a Fourier series:

$$TL^{XY}_g(z,\alpha,\mu) = \frac{1}{2\pi}TL^{XY}_{g,0}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\Big(TL^{XY}_{g,sm}(z,\mu)\sin(m\alpha) + TL^{XY}_{g,cm}(z,\mu)\cos(m\alpha)\Big). \tag{4.23}$$

$TL^{XY}_{g,0}$ is simply given by the azimuthally-integrated radial transverse leakage equation (Eq.

3.13b). To determine the $TL^{XY}_{g,sm}$ and $TL^{XY}_{g,cm}$ coefficients, Eq. 4.23 is multiplied by $\sin(n\alpha)$ and $\cos(n\alpha)$, respectively, and integrated azimuthally:

$$\int_0^{2\pi} \sin(n\alpha) TL^{XY}_g(z,\alpha,\mu)d\alpha = \sum_{l=1}^{N_{azi}} \sin(n\alpha_l) w_l TL^{XY}_{g,l}(z,\alpha_l,\mu)$$
$$= \int_0^{2\pi} \sin(n\alpha)\left(\frac{1}{2\pi} TL^{XY}_{g,0}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}} \left(TL^{XY}_{g,sm}(z,\mu)\sin(m\alpha) + TL^{XY}_{g,cm}(z,\mu)\cos(m\alpha)\right)\right)d\alpha, \tag{4.24a}$$

$$\int_0^{2\pi} \cos(n\alpha) TL^{XY}_g(z,\alpha,\mu)d\alpha = \sum_{l=1}^{N_{azi}} \cos(n\alpha_l) w_l TL^{XY}_{g,l}(z,\alpha_l,\mu)$$
$$= \int_0^{2\pi} \cos(n\alpha)\left(\frac{1}{2\pi} TL^{XY}_{g,0}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}} \left(TL^{XY}_{g,sm}(z,\mu)\sin(m\alpha) + TL^{XY}_{g,cm}(z,\mu)\cos(m\alpha)\right)\right)d\alpha. \tag{4.24b}$$

Removing the terms that integrate out yields Eq. 4.25:

$$\int_0^{2\pi} \sin(n\alpha) TL^{XY}(z,\alpha,\mu)d\alpha = \frac{1}{\pi}\int_0^{2\pi} TL^{XY}_{g,sn}\sin(n\alpha)\sin(n\alpha)d\alpha = TL^{XY}_{g,sn}(z,\mu), \tag{4.25a}$$

$$\int_0^{2\pi} \cos(n\alpha) TL^{XY}(z,\alpha,\mu)d\alpha = \frac{1}{\pi}\int_0^{2\pi} TL^{XY}_{g,cn}\cos(n\alpha)\cos(n\alpha)d\alpha = TL^{XY}_{g,cn}(z,\mu), \tag{4.25b}$$

which yields the final formulas for the coefficients:

$$TL^{XY}_{g,0}(z,\mu) = \int_0^{2\pi} TL^{XY}_g(z,\alpha,\mu)d\alpha = \sum_{l=1}^{N_{azi}} w_l TL^{XY}_{g,l}(z,\alpha_l,\mu), \tag{4.26a}$$

$$TL^{XY}_{g,sn}(z,\mu) = \int_0^{2\pi} \sin(n\alpha) TL^{XY}_g(z,\alpha,\mu)d\alpha = \sum_{l=1}^{N_{azi}} \sin(n\alpha_l) w_l TL^{XY}_{g,l}(z,\alpha_l,\mu), \tag{4.26b}$$

$$TL^{XY}_{g,cn}(z,\mu) = \int_0^{2\pi} \cos(n\alpha) TL^{XY}_g(z,\alpha,\mu)d\alpha = \sum_{l=1}^{N_{azi}} \cos(n\alpha_l) w_l TL^{XY}_{g,l}(z,\alpha_l,\mu). \tag{4.26c}$$

4.2.4 Visualization

To demonstrate what the radial transverse leakage tallies might look like when using Eqs. 4.26, consider a simple 3x3 pin case of a control rod pin surrounded by fuel pins (Fig. 4.1), using the seven group cross sections specified in the C5G7 problem [51]. In

this figure, the moderator is represented in green, the fuel in red, and a control rod in blue. Unless otherwise specified, each side uses a reflective boundary condition. All of these visualizations used 16 azimuthal angles per octant.

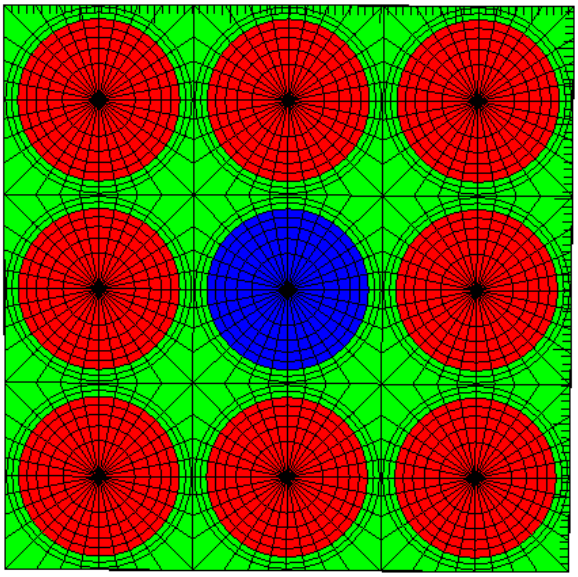

Figure 4.1: 3x3 Pin Test Case Geometry

Figures 4.2 and 4.4 show the radial transverse leakage angular dependence with reflective boundary conditions on all sides for a selected polar angle (distributions at other polar angles look similar). The solid blue line denotes the transverse leakage distribution with the explicit formulation and each dotted line represents a different order Fourier expansion. Figures 4.3 and 4.5 show the bottom right pin in both the fast and thermal figures to provide a better view of the expanded distributions. In these figures, the fast group distributions can exhibit fairly severe changes, and while all of the Fourier expansions do a reasonable job capturing the general shape, it takes quite a few moments to more accurately capture the finer details more explicitly. On the contrary, the distributions for the thermal group seem much smoother and more well-behaved. This is because the leakages for the thermal group are affected more locally since the mean free path is smaller, whereas the fast group distributions are affected by all pins in the problems because the mean free path is larger.

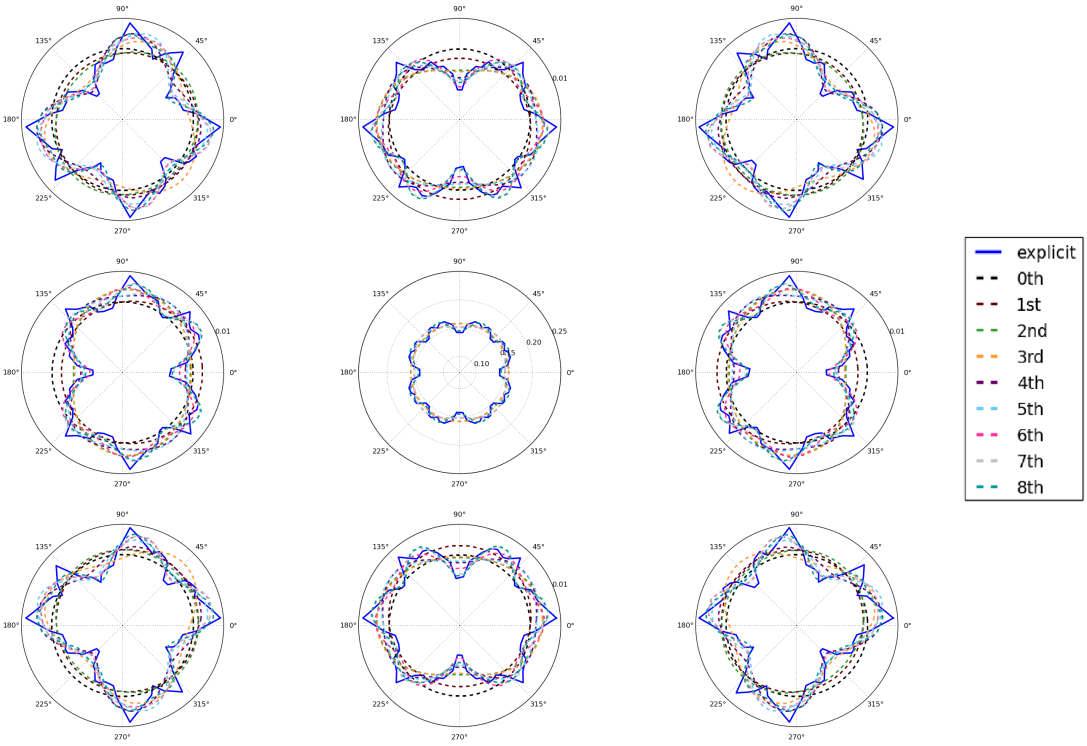

Figure 4.2: Radial TL, Rodded - Fast Group

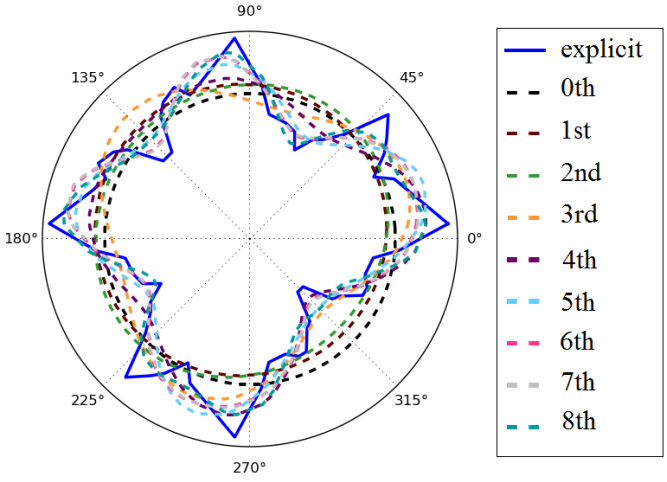

Figure 4.3: Radial TL, Rodded - Fast Group (Bottom-Right Pin)

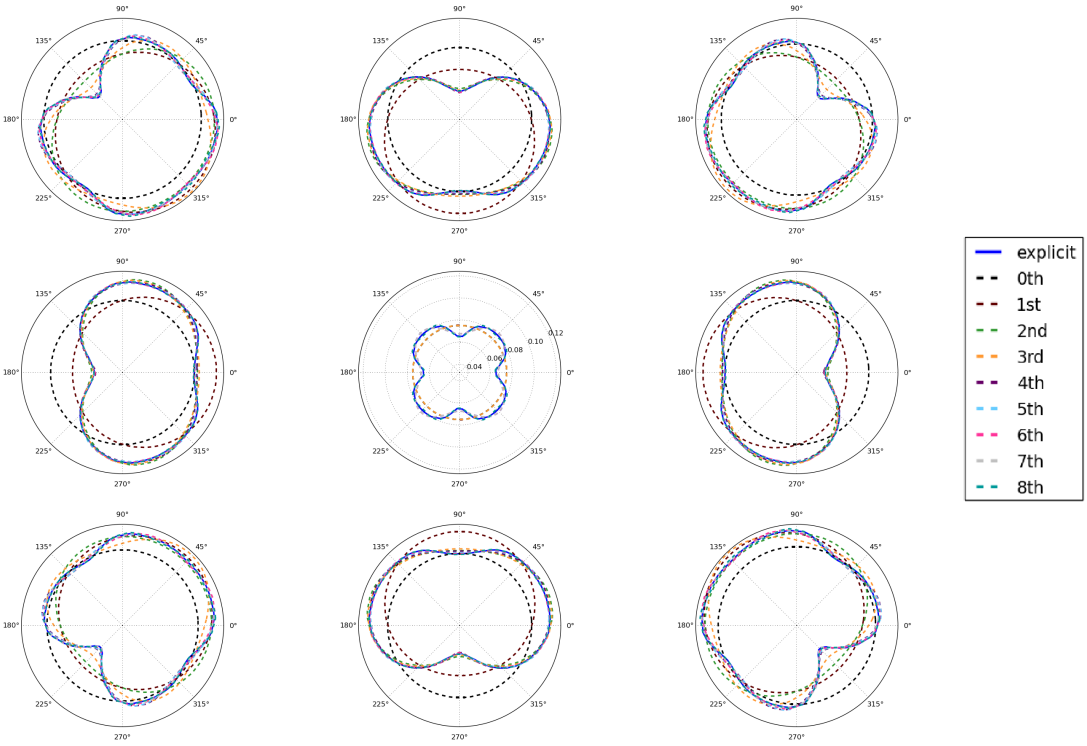

Figure 4.4: Radial TL, Rodded - Thermal Group

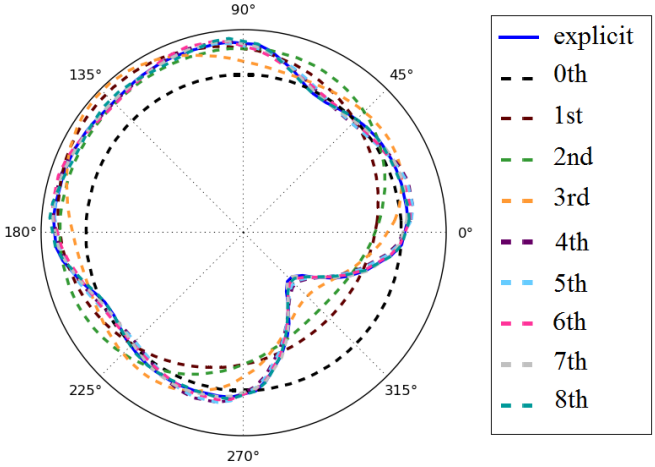

Figure 4.5: Radial TL, Rodded - Thermal Group (Bottom-Right Pin)

Figures 4.6 through 4.9 show the distributions with vacuum boundary conditions applied to the south and east faces. With this description, one would expect the distributions to be symmetric about the top-left to bottom-right diagonal, which they are. A notable difference is seen with the vacuum boundary conditions, as now the transverse leakages are

more directionally dominant towards the south and east faces. Additionally, it can be seen that the 0^{th} moment distributions actually do a poor job representing the general behavior, unlike the case with reflective boundary conditions. When incorporating the higher order expansions, the linear (1^{st} order) expansion more accurately captures the behavior, but everything past the 2^{nd} moment nearly matches the explicit angular dependence. For the thermal distributions, even the linear expansion performs well. The suggestion that only one or two Fourier moments is necessary is supported in the benchmark problems shown in Chapter 5.

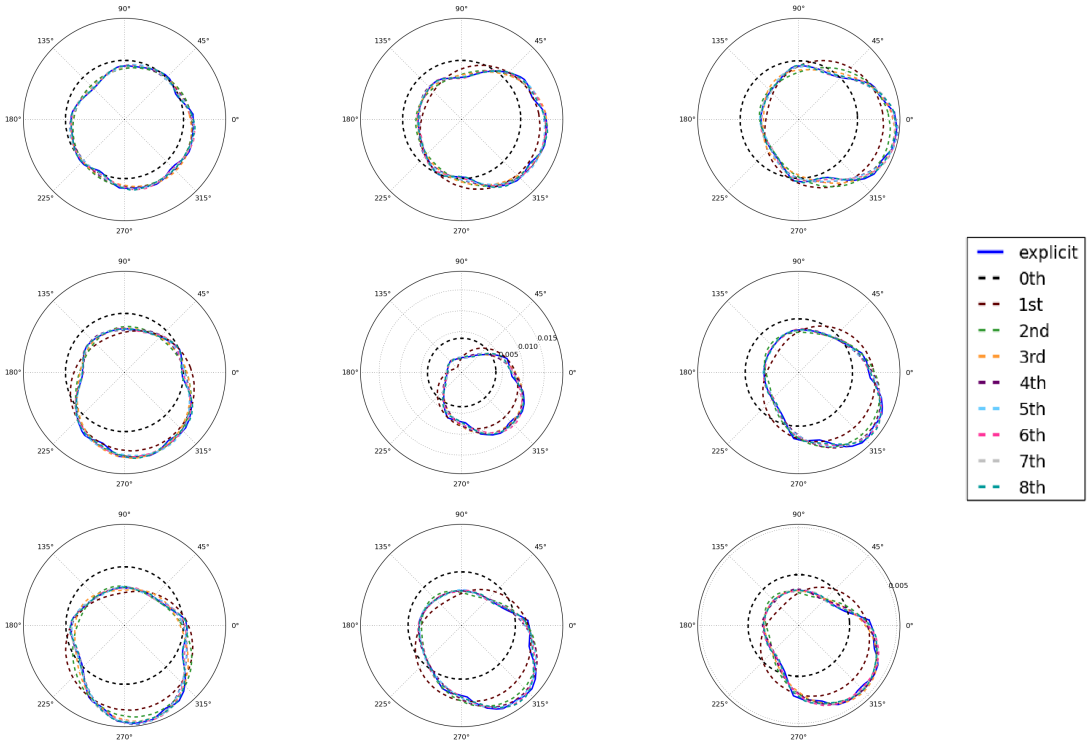

Figure 4.6: Radial TL, Rodded, Vacuum - Fast Group

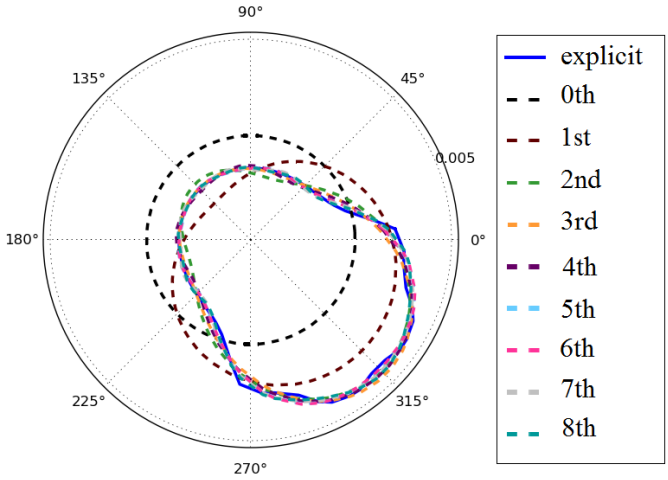

Figure 4.7: Radial TL, Rodded, Vacuum - Fast Group (Bottom-Right Pin)

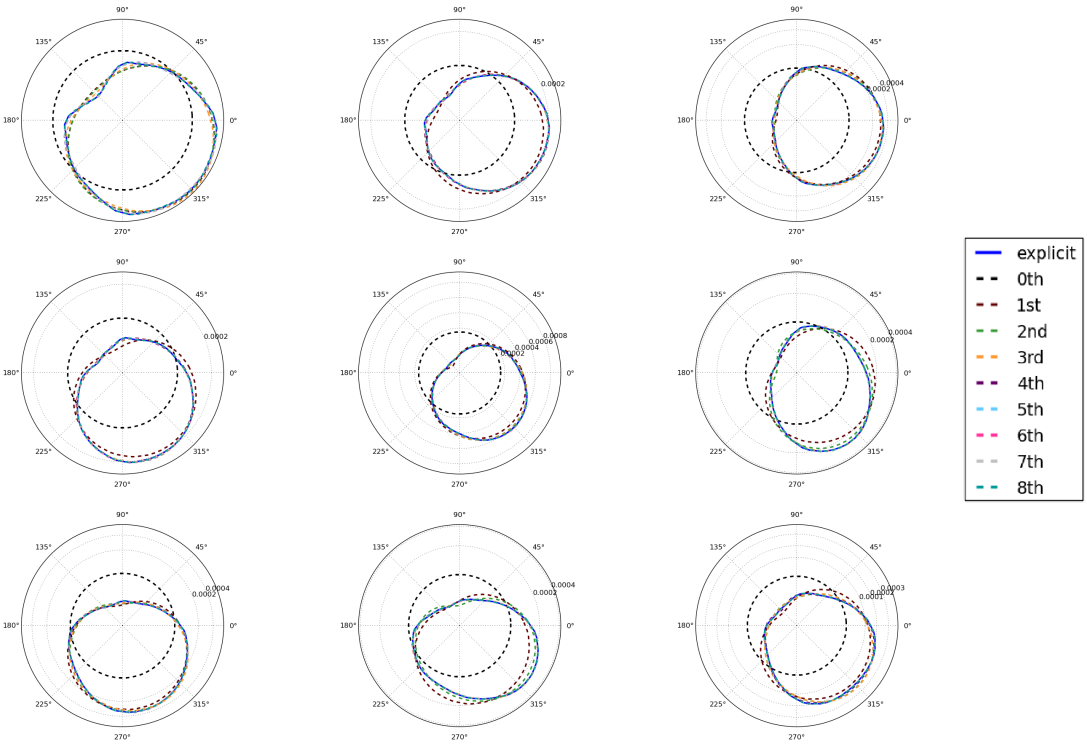

Figure 4.8: Radial TL, Rodded, Vacuum - Thermal Group

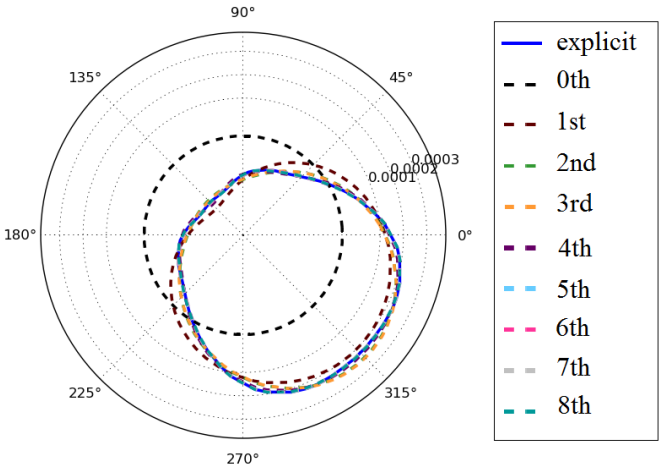

Figure 4.9: Radial TL, Rodded, Vacuum - Thermal Group (Bottom-Right Pin)

4.3 Axial TL Summary and MOC Sweepers

Now that the theory behind the moment-based transverse leakages has been presented, the various axial transverse leakage options will be summarized and the details of the MOC sweepers that correspond to each of them will be described. These MOC sweepers exist solely to support the functionality added by the new axial S_N solver.

First is the isotropic (ISO) transverse leakage, which is consistent with the methods that have been used in 2D/1D for some time (as in Eq. 3.5b). This transverse leakage option uses the pre-existing 2D-MOC sweeper, which simulates only one of the polar half-spaces.

Next is the azimuthally-integrated (AZI) transverse leakage, which only has a dependence on the polar angle, and the explicit (EXP) transverse leakage, which has both polar and azimuthal dependence (as in Eq. 3.5a). The MOC sweepers for these options simulate both the positive and negative polar half-spaces and store the sources directly. The azimuthally-integrated leakage stores a source value for each polar angle for each fine mesh region, and the explicit leakage stores one for each polar angle and each azimuthal angle for each fine mesh region. As one might expect, the storage for these (and particularly the explicit leakage), can turn out to be considerably more than the storage necessary for the isotropic leakage. Typically 3 or 4 polar angles and 12 to 16 azimuthal angles are used in each 3D octant, translating to 6 to 8 polar angles and 48 to 64 azimuthal angles in total. This gives roughly 6 to 8 times the storage for the azimuthally-integrated leakage compared to isotropic and 288 to 512 times the storage for the source with explicit angular dependence. Ongoing work is looking at other ways to reduce this storage without approximation, though it will likely require changing the design or responsibilities of the

MOC kernel.

Last is the Fourier moment-based source which was is similar to the radial transverse leakage source presented in the previous section in Eq. 4.23:

$$TL_g^Z(\alpha,\mu) = \frac{1}{2\pi} TL_{g,0}^Z(\mu) + \frac{1}{\pi} \sum_{m=1}^{N_{mom}} \Big(TL_{g,sm}^Z(\mu)\sin(m\alpha) + TL_{g,cm}^Z(z,\mu)\cos(m\alpha) \Big), \qquad (4.27)$$

but is constructed using the Fourier moment angular fluxes at the neighboring planes.

As with the azimuthally-integrated and explicit leakages, these sweepers simulate both polar half-spaces, but the storage is considerably reduced when compared to explicit representation. If the number of Fourier moments is zero, so there is only a flat component for each polar angle, the storage requirements and solution are identical to the azimuthally-integrated approximation. As the number of Fourier moments is incremented, a sine and cosine component are stored for each polar angle. For example, if 2 Fourier moments are used, the storage requirement is 1 flat component, 2 sine components, and 2 cosine components, so 5 values per polar angle, totaling 30-40 times the storage compared to isotropic and roughly 10x less storage than for explicit storage. Additionally, the number of azimuthal angles can be increased arbitrarily without increasing the cost of the moment-based storage. When the 3D-C5G7 results are presented in Chapter 5, memory profiling results demonstrate the memory savings.

Below is an example pseudo-code, with actual code in the kernel to illustrate floating point operations (FLOPs), that shows how the explicit (EXP) sweeper works. The azimuthally-integrated (AZI) sweeper is very similar, just without an azimuthal index for the source variable, qbar_ang. The way the MOC sweeper is set up, it sweeps both the forward and backward directions along a ray simultaneously in both the positive and negative half spaces, but only the code for the forward direction in the positive half-space is shown. As can be seen, it takes 5 FLOPs to evaluate the outgoing (phio1) and segment-averaged angular fluxes (phibar) for each segment, which totals to 20 flops for all directions.

```
LOOP over azimuthal angles
  ...
  LOOP over polar angles
    ...
    LOOP over segments along a ray

        !from positive half space
        !Evaluate forward direction
        ireg1=irg_seg(iseg1)
        phid1=phio1(iseg1-1)-thisTS%qbar_ang(ireg1,ipol,iang)
        phid1=phid1*exparg(iseg1,ipol)
        phio1(iseg1)=phio1(iseg1-1)-phid1
        phibar(ireg1)=phibar(ireg1)+phid1*wtang(ipol)

        !from positive half space
        !Evaluate backward direction

        !from negative half space
        !Evaluate forward direction

        !from negative half space
        !Evaluate backward direction

    ...
    END LOOP
  ...
  END LOOP
...
END LOOP
```

Next is the pseudo-code for the "Slow" moment MOC sweeper. With this sweeper, the source is evaluated on-the-fly, so each angle and segment has to evaluate the source for each fine mesh region as it encounters it during the sweep. This amounts to 4 FLOPs per moment. As one can predict, this can severely increase the number of FLOPs being performed and yields many duplicate evaluations. Even just one Fourier moment nearly doubles the FLOPs.

```
LOOP over azimuthal angles
  ...
  LOOP over polar angles
    ...
    LOOP over segments along a ray

      !from positive half space
      !Evaluate forward direction
      ireg1=irg_seg(iseg1)
      tmpqbar=thisTS%qbar_ang(ireg1,ipol,1) !isotropic component
      DO imom=1,nMom
        tmpqbar=tmpqbar+ &
          sinp(imom,iang)*thisTS%qbar_ang(ireg1,ipol,2*(imom-1)+2)+ &
          cosp(imom,iang)*thisTS%qbar_ang(ireg1,ipol,2*(imom-1)+3)
      ENDDO
      phid1=phio1(iseg1-1)-tmpqbar
      phid1=phid1*exparg(iseg1,ipol)
      phio1(iseg1)=phio1(iseg1-1)-phid1
      phibar(ireg1)=phibar(ireg1)+phid1*wtang(ipol)

      !from positive half space
      !Evaluate backward direction

      !from negative half space
      !Evaluate forward direction

      !from negative half space
      !Evaluate backward direction

    ...
    END LOOP
  ...
  END LOOP
...
END LOOP
```

Last is what is termed the "Fast" moment sweeper. Instead of evaluating the source on-the-fly for each angle in every fine mesh as it is reached during the sweep, the evaluated source is calculated and stored in a temporary array for the forward and backward azimuthal angles. This eliminates the duplicate evaluations, but requires an additional storage requirement equal to one Fourier moment (i.e. one sine and one cosine moment).

This provides much better speed than the "slow" moment sweeper, while still allowing considerable memory savings over the explicit source. It is important to note that the speedup compared to the "slow" sweeper increases as the ray spacing is refined.

```
LOOP over azimuthal angles
  ...
  Populate temporary thisTS%qbarAngle
  ...
  LOOP over polar angles
    ...
    LOOP over segments along a ray

      !from positive half space
      !Evaluate forward direction
      ireg1=irg_seg(iseg1)
      phid1=phio1(iseg1-1)-thisTS%qbarAngle(1,ipol,ireg1)
      phid1=phid1*exparg(iseg1,ipol)
      phio1(iseg1)=phio1(iseg1-1)-phid1
      phibar(ireg1)=phibar(ireg1)+phid1*wtang(ipol)

      !from positive half space
      !Evaluate backward direction

      !from negative half space
      !Evaluate forward direction

      !from negative half space
      !Evaluate backward direction

    ...
    END LOOP
  ...
  END LOOP
...
END LOOP
```

4.4 Parallelism

In Section 3.7, the parallel decomposition schemes available were described. Both spatial and angular decomposition use MPI communication, whereas the ray decomposition

uses OpenMP. In this work, both spatial and angular decomposition have been incorporated into the axial S_N solver, though the potential use of the ray decomposition processors to partition the polar angles will briefly be discussed. Scaling results for the spatial and angular decomposition schemes are shown in Chapter 5.

4.4.1 Spatial Decomposition

Spatial decomposition first exhausts the planar decomposition options and then moves to radial decomposition. Planar decomposition is generally used first since the radial 2D-MOC, which accounts for a majority of the run time, scales ideally as there is no communication necessary between the planes. However, one would expect the axial solvers to scale ideally when using radial decomposition, for similar reasons. This is still acceptable since the axial solver, even with explicit transverse leakages, is significantly less expensive than the MOC solver. For planar decomposition to work for the axial S_N solver, the angular fluxes at the planar decomposition boundaries need to be passed to the neighboring planes using MPI.

4.4.2 Azimuthal Angle/Moment Decomposition

Azimuthal decomposition is a bit more nuanced and the scheme is a little different with the explicit solver compared to the moment-based solver. With the explicit solver, the azimuthal angles are decomposed in a similar manner as they are for the 2D-MOC solver. However, it is important to note that the partitioning schemes are different. The axial solver partitions contiguously, at least in such a way the the angles on a domain are generally traveling in a similar direction, whereas the radial solver is responsible for the angle in the opposite direction as well. During the sweep, each angle tallies its contribution to the scalar flux moments and current. After each inner iteration, the scalar flux moments need to be reduced to update the sources for the next iteration. Once all sweeping is complete, the currents and angular fluxes, which are used to construct the axial transverse leakage for the radial solver, are reduced. This reduction for the angular fluxes can be particularly costly.

With the moment-based solver, each moment component (sine or cosine) can be partitioned. For example, an expansion with two Fourier moments has five moment components (one flat, two sine, and two cosine) and up to five processors can be used to decompose it. Unfortunately, an even number of processors is typically used for azimuthal decomposition, which effectively guarantees that the moment decomposition will be imbalanced. However, the zeroth moment (flat component) has slightly more responsibility that the sine/cosine moments in that it is solely responsible for determining

the scalar flux moments and currents. Unlike in the explicit solver where these needed to be reduced over angle, the processor with the zeroth moment can broadcast it to the other processors, which saves some overhead compared to the reduction. Additionally, the moments can be partitioned in such a way to guarantee that the zeroth moment is always located on a processor that has the least number of moment components, allowing some of the efficiency to be recovered since the processor will be idle for less time.

4.4.3 Polar Angle Decomposition

During the sweep over each azimuthal angle, the radial solver can use OpenMP to partition the rays. Since there are not multiple "rays" for each axial solve, this approach cannot be used. Alternatively, OpenMP could be used to partition the polar angle sweep. As the focus of this work is on the azimuthal dependence, the polar angle decomposition has not been implemented, though it should be very straight-forward.

4.5 Iteration Strategy

Figure 4.10 shows the calculation flow for the several different axial solver options that are available. If planar decomposition is used with the two-node solvers (NEM and SENM), the source data needs to be passed to the upper neighbor, which is responsible for solving the problem. Once solved, the flux data is passed back to the lower neighbor. With the one-node solvers (SP_N and S_N), the intranodal source and flux data do not need to be sent, but the surface flux moments and angular fluxes do.

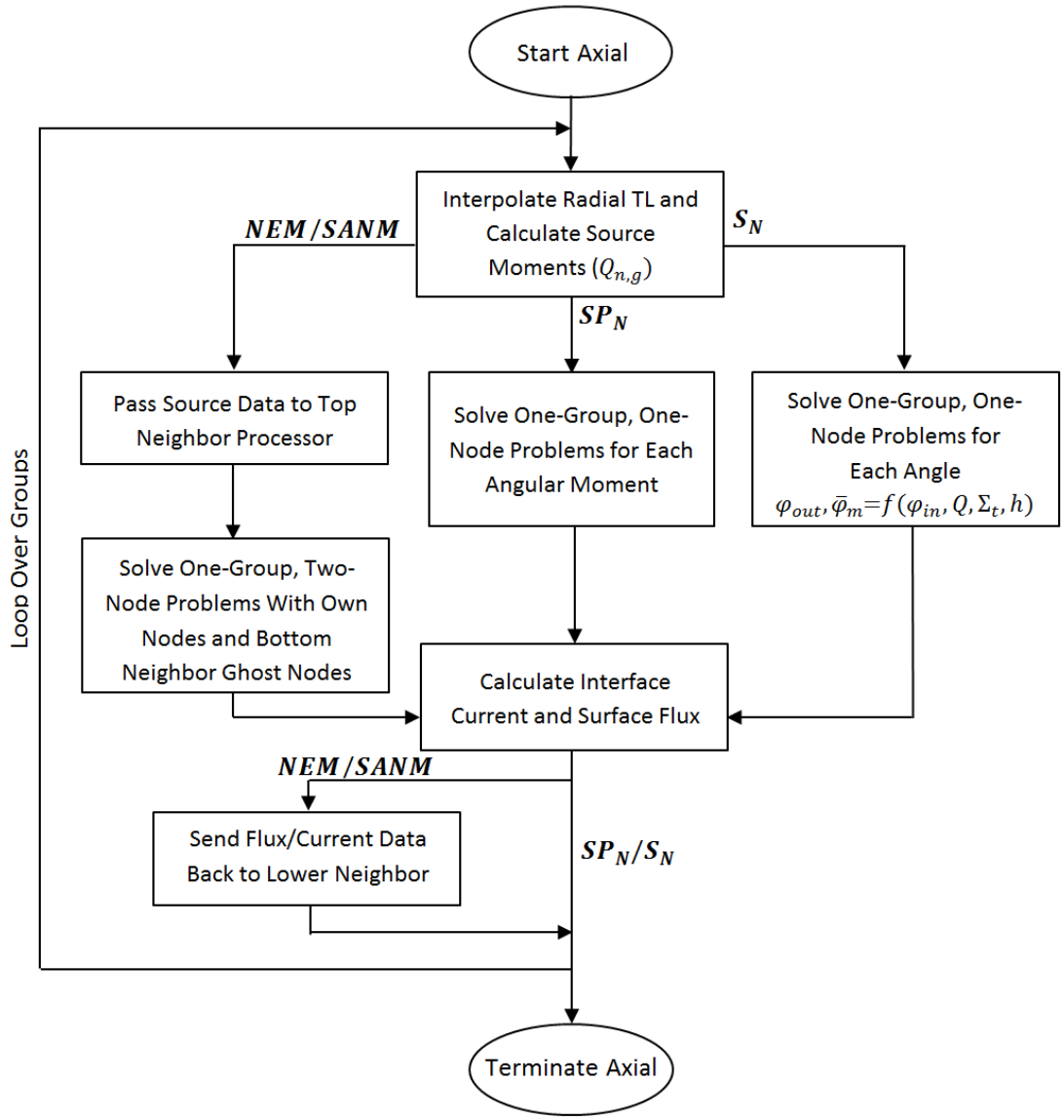

Figure 4.10: Axial Solvers Calculation Flow

Figure 4.11 shows more detail of the the calculation flow for the axial S_N sweeper when sweeping over a particular group, in particular to looping structures over the azimuthal and polar angles. It is important to note that there are multiple sweeps over each group to resolve the boundary conditions for each node. The number of inner iterations is currently hard-coded at five sweeps, but can be revisited once adequate modifications to the CMFD solver (i.e. $\hat{D}$ relaxation) have been implemented. The same applies to the MOC radial sweeper which also performs multiple sweeps.

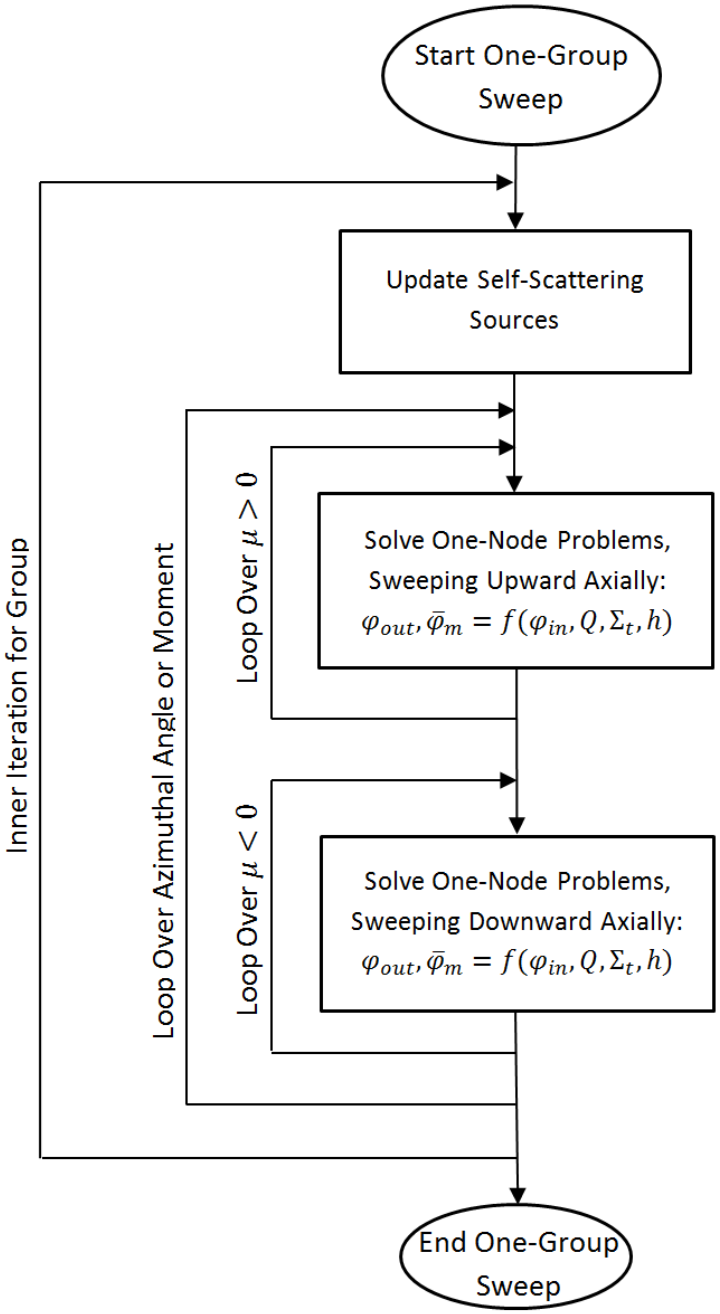

Figure 4.11: Axial S_N Calculation Flow

4.6 Anisotropic Scattering

Though anisotropic scattering is not a factor in any of the problems analyzed in this work, it is important to address how it fits in with the Fourier expansion axial solver. In Section 2.1.4.1, the fundamentals of anisotropic scattering were covered, and the spherical

harmonic functions were presented (as in Eq. 4.28):

$$R_0^0(\mathbf{\Omega}) = 1 \tag{4.28a}$$

$$\begin{aligned} R_1^{-1}(\mathbf{\Omega}) &= \xi = \sqrt{1-\mu^2}cos(\alpha) \\ R_1^0(\mathbf{\Omega}) &= \mu \\ R_1^1(\mathbf{\Omega}) &= \eta = \sqrt{1-\mu^2}sin(\alpha) \end{aligned} \tag{4.28b}$$

$$\begin{aligned} R_2^{-2}(\mathbf{\Omega}) &= \sqrt{\frac{3}{4}}(1-\mu^2)sin(2\alpha) \\ R_2^{-1}(\mathbf{\Omega}) &= \sqrt{3}\mu\sqrt{1-\mu^2}cos(\alpha) \\ R_2^0(\mathbf{\Omega}) &= \frac{1}{2}(3\mu^2-1) \\ R_2^1(\mathbf{\Omega}) &= \sqrt{3}\mu\sqrt{1-\mu^2}sin(\alpha) \\ R_2^2(\mathbf{\Omega}) &= -\sqrt{\frac{3}{4}}(1-\mu^2)cos(2\alpha) \end{aligned} \tag{4.28c}$$

In the way this is typically implemented, the flux moment for each corresponding part of the spherical harmonics expansion is tallied and stored. Then, these flux moments are used to construct the scattering source. During the transport sweep, the expanded source is evaluated for each angle as it is needed. However, because each of the spherical harmonics terms contains an azimuthal component that corresponds to the factor in the Fourier series, there does not need to be full azimuthal and polar angle coupling in the scattering term. In other words, there only needs to be polar angle coupling and the azimuthal moments can be treated independently.

Take, for example, anisotropic scattering in one-dimension, where the azimuthal dependence is neglected. In this situation, only the zeroth order spherical harmonics are necessary (R_0^0, R_1^0, R_2^0, etc.). A similar principle is true when the angular flux is expanded using a Fourier series. For P_2 scattering, there would be zeroth moment coupling between R_0^0, R_1^0, and R_2^0, first sine moment coupling between R_1^1 and R_2^1, and so on. Effectively, it requires the same amount of additional storage, since the information is collapsed to still use an expansion for the polar angle, but the source evaluation will be less expansion since the azimuthal behavior is represented in a similar expansion. Additionally, the order of the scattering governs how many Fourier moments receive an anisotropic scattering component. On the other hand, it will be shown in the results that usually one or two Fourier moments is adequate, which could be more restrictive of the scattering allowed. Typically, P_2 scattering is used for the radial solver, which would not be matched in the axial solver

if only using a first order Fourier expansion. This would not necessarily be detrimentally restrictive, though, as additional moments could be used to provide consistent scattering treatments if desired.

4.7 Asymptotic Behavior

One important aspect of 2D/1D is the consideration of the asymptotic solution as the axial and radial mesh are refined. In other words, this section investigates what is expected when using explicit and moment-based transverse leakages as these mesh are refined.

4.7.1 Radial Solver

To start this investigation, consider the equations for the radial solver, which were derived in Section 3.1.1, where the 3D transport equation was averaged axially, per Eq. 4.29:

$$\varphi^Z_{g,l}(x,y) = \frac{1}{h_z}\int_{z_B}^{z_T} \varphi_{g,l}(x,y,z)dz, \tag{4.29}$$

$$\sqrt{1-\mu_l^2}\Big(cos(\alpha_l)\frac{\partial}{\partial x} + sin(\alpha_l)\frac{\partial}{\partial y}\Big)\varphi^Z_{g,l}(x,y) + \Sigma^Z_{t,g}(x,y)\varphi^Z_{g,l}(x,y) = \tilde{q}^Z_{g,l}(x,y), \tag{4.30a}$$

$$\tilde{q}^Z_{g,l}(x,y) = \overline{q}^Z_{g,l}(x,y) + TL^Z_{g,l}(x,y), \tag{4.30b}$$

$$\overline{q}^Z_{g,l}(x,y) = \frac{\chi^Z_g(x,y)}{4\pi k_{\text{eff}}}\sum_{g'=1}^{N_g} \nu\Sigma^Z_{f,g'}(x,y)\phi^Z_{g'}(x,y) + \frac{1}{4\pi}\sum_{g'=1,g'\neq g}^{N_g} \Sigma^Z_{s0,g'\rightarrow g}(x,y)\phi^Z_{g'}(x,y), \tag{4.30c}$$

$$TL^Z_{g,l}(x,y) = \frac{\mu_l}{h_z}(\varphi_{B,g,l}(x,y) - \varphi_{T,g,l}(x,y)). \tag{4.30d}$$

As the axial mesh is refined ($h_z \rightarrow 0$), Eq. 4.29 approaches the actual value of the angular flux at an arbitrary point z, since it just becomes an average over a smaller range:

$$\lim_{h_z\rightarrow 0}\frac{1}{h_z}\int_{z_B}^{z_T} \varphi_{g,l}(x,y,z)dz = \varphi_{g,l}(x,y,z). \tag{4.31}$$

Additionally, the axial transverse leakage term limits to the derivative of the angular flux times μ, which is the original axial streaming operator:

$$\lim_{h_z\rightarrow 0}\frac{\mu_l}{h_z}\Big(\varphi_{B,g,l}(x,y) - \varphi_{T,g,l}(x,y)\Big) = \mu\frac{\partial}{\partial z}\varphi_{g,l}(x,y,z). \tag{4.32}$$

Written in terms of the angular fluxes from the axial solver, which are integrated radially

over x and y, we have Eq. 4.33. As will be seen when the axial solver is assessed, $\varphi^{XY}_{g,l}(z)$ will limit to $\varphi_{g,l}(x,y,z)$ as the radial/coarse mesh is refined, so it will be consistent with Eq. 4.32:

$$\lim_{h_z \to 0} \frac{\mu_l}{h_z} \left(\varphi^{XY}_{B,g,l} - \varphi^{XY}_{T,g,l} \right) = \mu \frac{\partial}{\partial z} \varphi^{XY}_{g,l}(z). \tag{4.33}$$

None of these results should be very surprising, since these equations are well founded from the 3D transport equation, but it is reassuring to see this. The equations for the radial solver then become:

$$\sqrt{1-\mu_l^2}\Big(cos(\alpha_l)\frac{\partial}{\partial x} + sin(\alpha_l)\frac{\partial}{\partial y}\Big)\varphi_{g,l}(x,y,z) + \Sigma_{t,g}(x,y,z)\varphi_{g,l}(x,y,z) = \tilde{q}_{g,l}(x,y,z), \tag{4.34a}$$

$$\tilde{q}_{g,l}(x,y,z) = \overline{q}_{g,l}(x,y,z) + TL_{g,l}(x,y,z), \tag{4.34b}$$

$$\overline{q}_{g,l}(x,y,z) = \frac{\chi_g(x,y,z)}{4\pi k_{\text{eff}}} \sum_{g'=1}^{N_g} \nu\Sigma_{f,g'}(x,y,z)\phi_{g'}(x,y,z) + \frac{1}{4\pi} \sum_{g'=1, g' \neq g}^{N_g} \Sigma_{s0,g' \to g}(x,y,z)\phi_{g'}(x,y,z), \tag{4.34c}$$

$$TL_{g,l}(x,y,z) = \mu \frac{\partial}{\partial z} \varphi_{g,l}(x,y,z), \tag{4.34d}$$

which are consistent with the 3D transport equation, just with the axial streaming on the right hand side.

4.7.2 Axial Solver

Now to consider the axial solver equations, which average the 3D transport equation over x and y using Eq. 4.35:

$$\varphi^{XY}_{g,l}(z) = \frac{1}{A_{xy}} \int_{x_L}^{x_R} \int_{y_L}^{y_R} \varphi_{g,l}(x,y,z) dy dx, \tag{4.35}$$

$$\mu_l \frac{\partial}{\partial z} \varphi^{XY}_{g,l}(z) + \Sigma^{XY}_{t,g}(z)\varphi^{XY}_{g,l}(z) = \tilde{q}^{XY}_{g,l}(z), \tag{4.36a}$$

$$\tilde{q}^{XY}_{g,l}(z) = \overline{q}^{XY}_{g,l}(z) + TL^{XY}_{g,l}(z), \tag{4.36b}$$

$$\overline{q}^{XY}_{g,l}(z) = \frac{\chi^{XY}_g(z)}{4\pi k_{\text{eff}}} \sum_{g'=1}^{N_g} \nu\Sigma^{XY}_{f,g'}(z)\phi^{XY}_{g'}(z) + \frac{1}{4\pi} \sum_{g'=1}^{N_g} \Sigma^{XY}_{s,g' \to g}(z)\phi^{XY}_{g}(z), \tag{4.36c}$$

$$TL^{XY}_{g,l}(z) = -\frac{\sqrt{1-\mu_l^2}}{A_{xy}} \begin{pmatrix} cos(\alpha_l) \int_{y_L}^{y_R} \Big(\varphi_{g,l}(x_R,y,z) - \varphi_{g,l}(x_L,y,z)\Big) dy \\ + sin(\alpha_l) \int_{x_L}^{x_R} \Big(\varphi_{g,l}(x,y_R,z) - \varphi_{g,l}(x,y_L,z)\Big) dx \end{pmatrix}. \tag{4.36d}$$

As the radial mesh is refined ($A_{xy} \rightarrow 0$), Eq. 4.35 also limits to the angular flux at the specified point:

$$\lim_{A_{xy}\rightarrow 0} \frac{1}{A_{xy}} \int_{x_L}^{x_R} \int_{y_L}^{y_R} \varphi_{g,l}(x,y,z)dydx = \varphi_{g,l}(x,y,z). \tag{4.37}$$

And just as in the axial transverse leakage, the radial transverse leakage also limits to the associated streaming component:

$$\lim_{A_{xy}\rightarrow 0} TL_{g,l}^{XY}(z) = -\sqrt{1-\mu_l^2}\left(cos(\alpha_l)\frac{\partial}{\partial x}\varphi_{g,l}(x,y,z) + sin(\alpha_l)\frac{\partial}{\partial y}\varphi_{g,l}(x,y,z)\right). \tag{4.38}$$

Substituting these back into the axial equations would similarly yield the 3D transport equation.

4.7.3 Fourier Moment Expansions

In the previous two sections, it was shown that in explicit form, the radial and axial equations limit to the 3D transport solution as the mesh is refined, though this is also true with the Fourier moment-based expansions for the angular fluxes and transverse leakages. However, for this to be completely true, the number of Fourier moments also needs to asymptotically approach infinity:

$$\lim_{N_{mom}\rightarrow\infty}\left(\frac{1}{2\pi}\varphi_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\left(\varphi_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + \varphi_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\right)\right) = \varphi_g^{XY}(z,\alpha,\mu). \tag{4.39}$$

The error that is introduced from the Fourier moment expansion is the difference between the infinite series and the truncated expansion. Results that will be covered in the next chapter suggest that only one or two moments adequately capture most of the azimuthal behavior. Substituting this back into the the axial transverse leakage:

$$TL_{g,l}^{XY}(x,y,z) = \mu\frac{\partial}{\partial z}\left(\frac{1}{2\pi}\varphi_{g,0}^{XY}(z,\mu) + \frac{1}{\pi}\sum_{m=1}^{N_{mom}}\left(\varphi_{g,sm}^{XY}(z,\mu)\sin(m\alpha) + \varphi_{g,cm}^{XY}(z,\mu)\cos(m\alpha)\right)\right), \tag{4.40}$$

which, as the number of moments and axial and radial mesh are refined, will asymptotically approach the explicit axial streaming term in the transport equation.

CHAPTER 5

Numerical Results

This chapter is devoted to presenting numerical results for several commonly used benchmark cases. The first case that will be assessed is the rodded Takeda LWR problem [56], which uses a two group cross section set and consists of a homogeneous cube core surrounded by moderator with a control rod placed immediately next to the core. While not a very physical problem, particularly since light water reactors do not have homogeneous fuel or rectangular control rods, it is a good first problem to consider because of its geometric simplicity.

The next cases under consideration are the extended 3D-C5G7 problems [52], which have a more realistic geometric configuration and a seven group cross section library. While still not very physical, it does capture the heterogeneity between the fuel and moderator within each pin cell, though the cladding and fuel-clad gap are neglected. In analyzing the three different configurations, which pertain to the position of the control rod, some interesting trends are observed, suggesting that cancellation of error is an important factor. To narrow in on these sources of error, an alternative set of cases using volume-homogenized cross sections are shown, helping quantify the cancellation of error as well as the remaining sources of errors.

In all of these cases, a Monte Carlo code (either MCNP or SHIFT) was used to obtain the reference solution. For the Takeda problem, the reference solution is specified in the benchmark [56]. However, for the C5G7 problems, it was found that the power distributions available from the benchmark reference (with MCNP) may not be very tightly converged, so new reference distributions were obtained by using SHIFT. Results for these new reference solutions are also included in Appendix B.

5.1 Takeda LWR Rodded Benchmark

5.1.1 Problem Description and Discretization

Figures 5.1 and 5.2 show the top and side views of the problem, respectively. The model is a 25 cm cube with a 15 cm cubic core surrounded by a reflector region. Immediately next to the core is a 5 cm by 5 cm by 25 cm control rod.

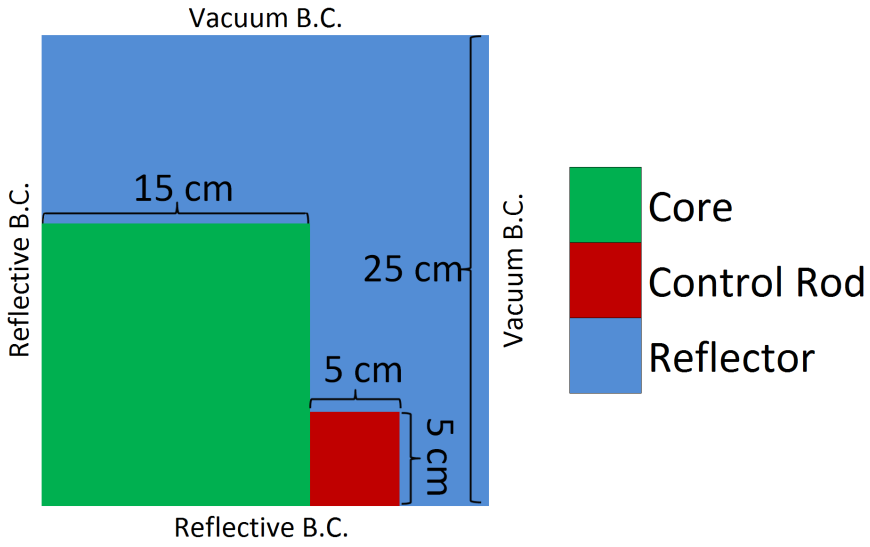

Figure 5.1: Takeda Geometry - Top View

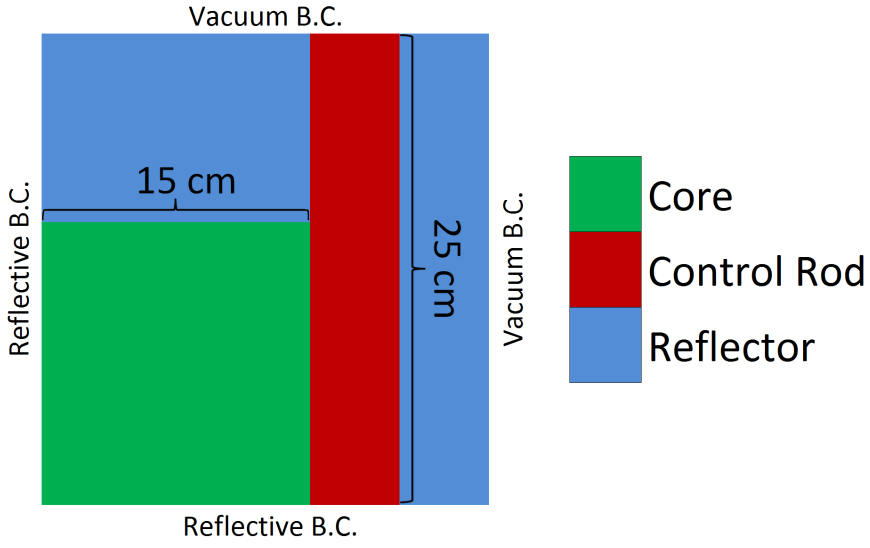

Figure 5.2: Takeda Geometry - Side View (South)

Table 5.1 shows the two group cross section data for the three different material types in the problem. Shown are the transport cross section ($\Sigma_{tr,g}$), nu-fission cross section ($\nu\Sigma_{f,g}$), fission spectrum (χ_g), and scattering matrix values ($\Sigma_{s0,1\rightarrow g}$ and $\Sigma_{s0,2\rightarrow g}$).

Table 5.1: Cross Section Data

Region	Group	$\Sigma_{tr,g}$	$\nu\Sigma_{f,g}$	χ_g	$\Sigma_{s0,1\rightarrow g}$	$\Sigma_{s0,2\rightarrow g}$
Core	1	2.23775E-01	9.09319E-03	1.0	1.92423E-01	0.00000E+00
	2	1.03864E+00	2.90183E-01	0.0	2.28253E-02	8.80439E-01
Control Rod	1	8.52325E-02	0.0000E+00	0.0	6.77241E-02	0.0000E+00
	2	2.17460E-01	0.0000E+00	0.0	6.45461E-05	3.52358E-02
Reflector	1	2.50367E-01	0.0000E+00	0.0	1.93446E-01	0.00000E+00
	2	1.64482E+00	0.0000E+00	0.0	5.65042E-02	1.62452E+00

In MPACT, the radial discretization consists of 1 cm by 1 cm pins using a 15 by 15 Cartesian grid submeshing. Axially, a 5 cm mesh was used yielding 5 total planes in the problem. The angular domain was discretized using 16 azimuthal angles per octant in a Chebyshev quadrature and 4 polar angles per half space in a Gauss quadrature set. Additionally, a 0.01 cm ray spacing was used. Lastly, all cases were run on Titan [43] using 500 processors (125 spatial decomposition domains and 4 angular decomposition processors).

5.1.2 Results

Table 5.2 shows the eigenvalue (k_{eff}) results as well as the number of outer iterations and timing data. In this table, the S_N results have an additional label to indicate the radial and axial transverse leakage approximations. With this description, the radial transverse leakage is listed first, then the axial transverse leakage after the hyphen (e.g. AZI-ISO indicates an azimuthally-integrated radial transverse leakage with an isotropic axial transverse leakage). For all other notations with only one description listed, the radial and axial transverse leakages are the same. Also, the moment descriptions include an integer indicating how many Fourier moments were used.

When considering results like these, it is often useful to lump several solvers together that should perform comparably well. For example, both the NEM and SENM solvers are resolving the diffusion equation axially, and only the expansions for the source and flux are different. As the axial mesh is refined, one would expect these solvers to be consistent. Not surprisingly, they do perform very similarly, at only 3 pcm difference. Next,

consider the SP_N solvers and the S_N solver with isotropic transverse leakages. Since in 1D, the SP_N equations are consistent with the P_N equations, which asymptotically approach the transport equation as N approaches infinity, one should expect these solutions to be fairly consistent with the S_N results since it also approaches the transport solution as N approaches infinity. That is not to say that the solutions should be identical, as N is limited it both solvers and they use slightly different spatial expansions, but they should be reasonably close. Additionally, introducing an azimuthally-integrated radial transverse leakage with an isotropic axial transverse leakage does not seem to significantly impact the result. Lastly, the S_N moment-based and explicit transverse leakages should exhibit the same behavior, particularly as the number of Fourier moments is increased. These results suggest that only one or two Fourier moments is necessary to accurately recreate the solution with explicit angular dependence. It is important to note that the eigenvalue uncertainty in this case is roughly 60 pcm [56], a range in which all of the moment-based solvers with at least one sine and cosine moment are contained.

Table 5.2: Takeda Rodded Results

	Eig.	Diff. (pcm)	Iters.	Time (sec)
Reference	0.96240	± 60	—	—
NEM	0.95002	-1238	17	5.54
SENM	0.95005	-1235	17	5.55
SP_3	0.95504	-736	18	5.61
SP_5	0.95520	-720	18	5.65
S_N ISO	0.95519	-721	22	6.41
S_N AZI-ISO	0.95336	-904	22	6.19
S_N MOM-0	0.95447	-793	22	8.21
S_N MOM-1	0.96282	42	22	9.48
S_N MOM-2	0.96236	-4	22	10.90
S_N MOM-3	0.96240	0	22	12.20
S_N MOM-4	0.96239	-1	22	13.69
S_N EXP	0.96240	0	22	17.38

Table 5.3 shows the relative error in the region-averaged scalar flux values for each group. For the reference data, the actual region-averaged fluxes are stated, as well as the uncertainty in parentheses. To tabulate these, the scalar fluxes are normalized in accordance

with Eq. 5.1:

$$\sum_{g=1}^{2}\left[\int \nu\Sigma_{f,g}(\boldsymbol{x})\phi_g(\boldsymbol{x})dV\right] = \sum_{g=1}^{2}\left[\sum_{i=1}^{N_{reg}} \nu\Sigma_{f,g,i}\phi_{g,i}V_i\right] = 1. \tag{5.1}$$

Table 5.3: Takeda Rodded Flux Results

	Group	Fuel	Reflector	Control Rod
Reference	1	4.9125E-03 (0.10%)	5.9109E-04 (0.21%)	1.2247E-03 (0.48%)
	2	8.6921E-04 (0.13%)	8.7897E-04 (0.23%)	2.4604E-04 (0.72%)
NEM	1	-1.52%	4.71%	-0.02%
	2	0.03%	4.65%	1.44%
SENM	1	-1.53%	4.71%	-0.03%
	2	0.03%	4.64%	1.44%
SP_3	1	-0.82%	1.79%	-0.24%
	2	-0.09%	2.06%	0.37%
SP_5	1	-0.68%	1.52%	-0.28%
	2	-0.12%	1.88%	0.26%
S_N ISO	1	-0.72%	1.58%	-0.31%
	2	-0.11%	1.86%	0.23%
S_N AZI-ISO	1	-0.82%	2.06%	-0.07%
	2	-0.09%	2.34%	0.83%
S_N MOM-0	1	-0.81%	1.93%	0.29%
	2	-0.10%	2.18%	1.16%
S_N MOM-1	1	-0.36%	-0.26%	-0.03%
	2	-0.18%	0.03%	-0.40%
S_N MOM-2	1	-0.42%	-0.05%	0.06%
	2	-0.17%	0.22%	-0.18%
S_N MOM-3	1	-0.41%	-0.08%	0.04%
	2	-0.17%	0.20%	-0.22%
S_N MOM-4	1	-0.41%	-0.07%	0.05%
	2	-0.17%	0.20%	-0.21%
S_N EXP	1	-0.41%	-0.07%	0.05%
	2	-0.17%	0.20%	-0.21%

As with the eigenvalue results, considerable errors are seen with the diffusion-based axial solvers (NEM and SENM), particular in the reflector. Moving into the SP_N and S_N

ISO results, definite improvements are observed, but still with appreciable errors in the reflector values. Incorporating azimuthal dependence with the moment-based and explicit transverse leakages yields terrifically improved results, particularly in the reflector.

Lastly, Table 5.4 shows the a comparison between the S_N solver with explicit transverse leakages and 3D-MOC results obtained from MPACT [29], which used 0.5 cm^3 regions with an S_8 quadrature and a 0.05 cm ray spacing.

It is also important to note the 3D-MOC results with MPACT were obtained using 1,000 processors on Titan, taking 39.86 seconds in total (roughly 11.07 core-hours). With 2D/1D, the runtime is significantly reduced, even with the explicit transverse leakages, taking only 17.38 seconds on 500 processors (roughly 2.41 core-hours). With one and two Fourier moments for azimuthal dependence, this is reduced to approximately 1.32 and 1.51 core-hours, respectively. In this case, the 2D/1D iteration scheme allows for more results to be obtained with more refined parameters in less time.

Table 5.4: Takeda Rodded Results (Compared to 3D-MOC)

	Eig.	Diff. (pcm)	Group	Fuel	Reflector	Control Rod
S_N EXP	0.96240	0	1	-0.41%	-0.07%	0.05%
			2	-0.17%	0.20%	-0.21%
MPACT 3D-MOC	0.96253	13	1	-0.69%	0.11%	0.31%
			2	-0.12%	-3.27%	2.21%

Table 5.5 shows the results with a considerably more refined set of parameters: 0.001 cm ray spacing, 32 azimuthal angles per octant, 8 polar angles per half-space. Two refined axial meshes (2.5 cm and 1.0 cm) were considered.

Table 5.5: Takeda Rodded Results (Refined)

	Eig.	Diff. (pcm)	Group	Fuel	Reflector	Control Rod
S_N EXP	0.96240	0	1	-0.41%	-0.07%	-0.05%
			2	0.17%	0.20%	-0.21%
Refined (2.5 cm)	0.96249	9	1	-0.32%	-0.17%	0.06%
			2	-0.18%	0.20%	-0.17%
Refined (1.0 cm)	0.96251	11	1	-0.31%	-0.19%	0.06%
			2	-0.19%	0.19%	-0.13%

Only minor improvements are observed in the average flux values, and though the eigenvalue difference is large, it is still well within the eigenvalue uncertainty specified by

the benchmark. These refinement results suggest that coarser parameters (even compared to what was initially used) might possibly yield sufficiently accurate answers.

5.2 C5G7 Benchmarks

The next problems that will be analyzed are from the C5G7 benchmark specification [51, 52], specifically the extended 3D-C5G7 problems, which consist of three different rod configurations. One would expect the angular dependence (particularly the azimuthal component) of the transverse leakages to be important in the more rodded cases, which is observed. Unfortunately, as will be seen, the angle-dependent leakages do not perform particularly well with the standard benchmark specification, which contains heterogeneous pins with resolved fuel and moderator regions. The source of this behavior is believed to be cancellation of error between the spatial and angular treatment of the transverse leakages, which is removed as the angular dependence is improved. To test this, alternative problems are presented that use pin-homogenized cross sections, which bring validity to the flat spatial approximation of the axial transverse leakage. These cases perform considerably better and support the hypothesis that cancellation of error is playing a role.

5.2.1 Geometry Description and Discretization

Radially, the core consists of a 2x2 assembly layout of UO_2 and MOX assemblies (17x17 pin) surrounded by moderator (Figure 5.3), with reflective boundary conditons on the north and west faces. Each pin cell in these assemblies has a radius of 0.54 cm arranged with a 1.26 cm pitch. In the figure below, the red pins are 3.3% enriched UO_2 fuel; the green pins, 4.3% enriched MOX; the orange pins, 7.0% enriched MOX; the cyan pin, 8.7% enriched MOX. The other pins are non-fuel: the yellow pins are guide tubes (which can be replaced with rodded material); the magenta pins are instrument tubes; and the blue pins (and material surrounding each pin) are moderator.

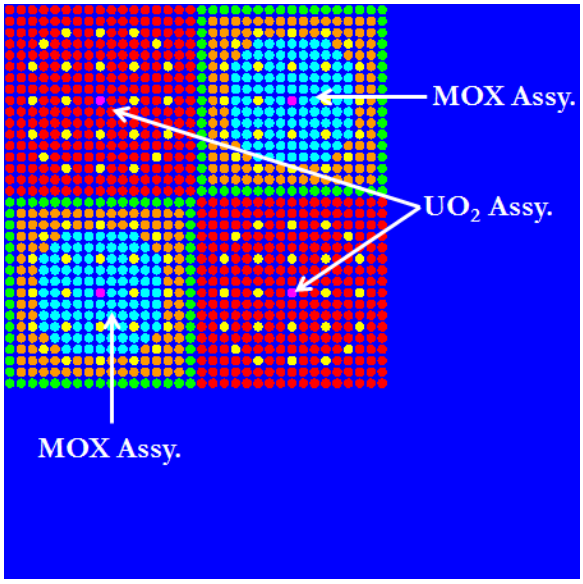

Figure 5.3: C5G7 - Radial Layout

Axially, there is 42.84 cm of active fuel with 21.42 cm of moderator at the top, with reflective boundary conditions on the bottom. There are three different configurations considered here, all of which pertain to the position of the control rod. The first is designated as the unrodded configuration (UR), in which case the control rod is only present in the axial reflector above the assemblies (Figure 5.4).

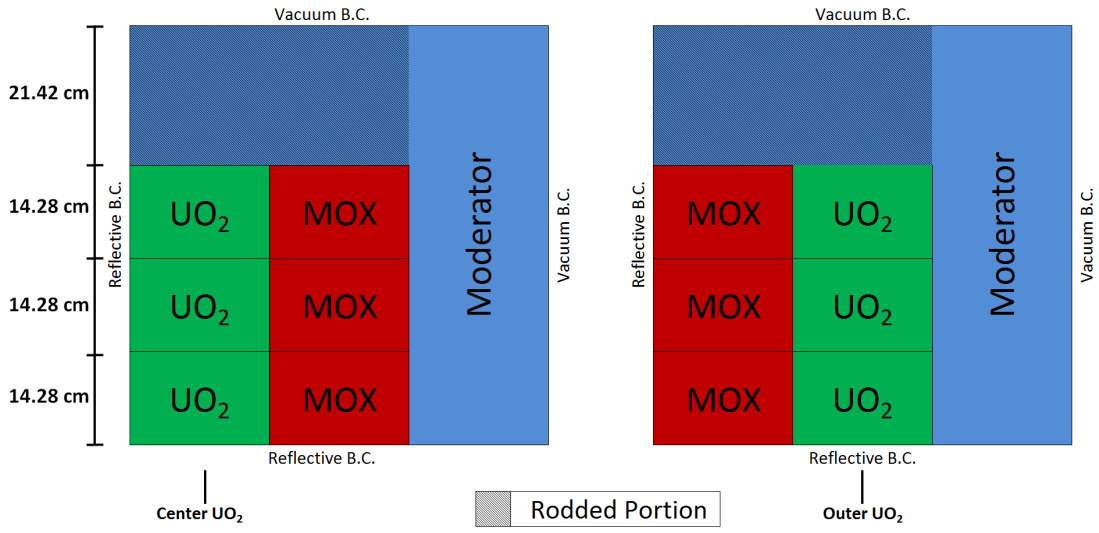

Figure 5.4: C5G7 - Configuration UR

The next, referred to as rodded configuration A (rA), has the control rod inserted 14.28 cm into the center UO_2 assembly, leaving the rod in the axial reflector for the other assemblies

(Figure 5.5).

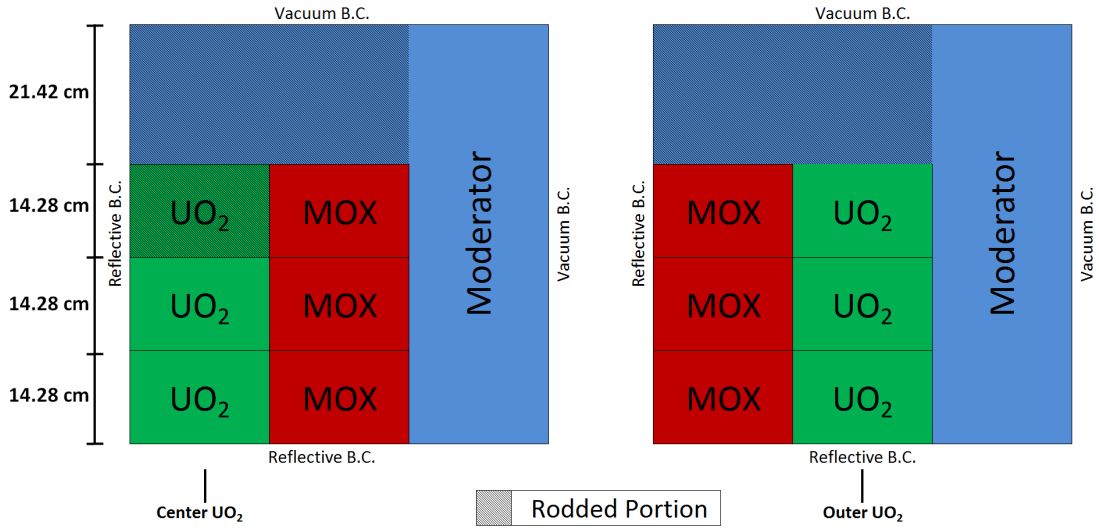

Figure 5.5: C5G7 - Rodded Configuration A

The last, rodded configuration B (rB) has the rod inserted 28.56 cm into the center UO_2 assembly and 14.28 cm into both MOX assemblies (Figure 5.6).

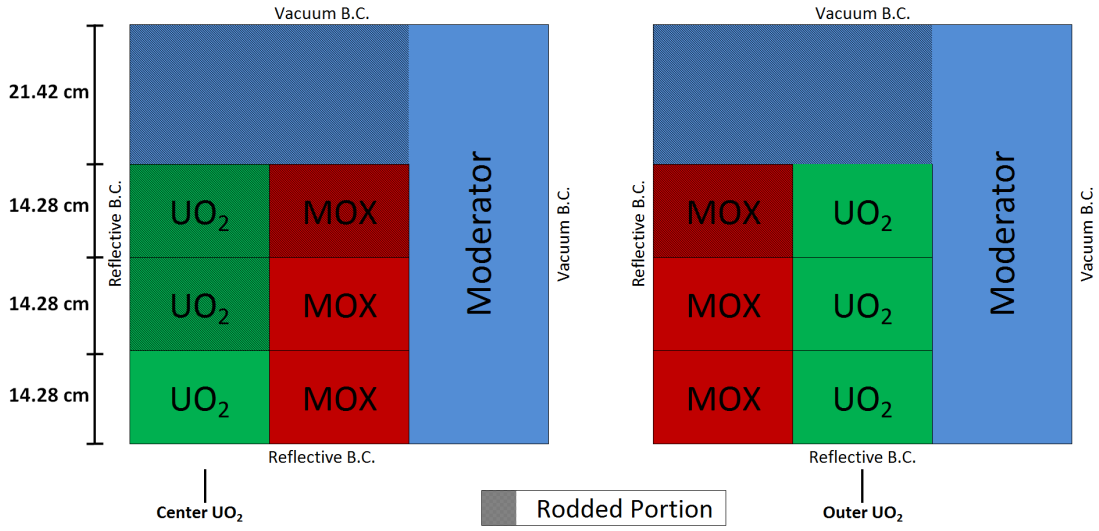

Figure 5.6: C5G7 - Rodded Configuration B

The cross sections are from a seven group library generated specifically for the benchmark [51]. Each of the fuel and control rod pins were discretized using 5 radial rings surrounded by 3 rings in the moderator, all with 16 azimuthal divisions. The reflector pins were discretized using a 15x15 Cartesian submeshing. Axially, a 3.57 cm axial mesh was

used, yielding 18 planes. A Chebyshev azimuthal quadrature and a Gauss polar quadrature were used with 16 azimuthal angles per octant and 4 polar angles per half-space and a 0.01 cm ray spacing. All cases were run on Titan using 648 processors (162 processes for spatial decomposition and 4 threads for ray decomposition).

5.2.2 3D, Heterogeneous Pin Results

Table 5.6 shows the results for the unrodded configuration (UR) for a variety of axial solvers. These results include the eigenvalue, difference (in pcm) from the reference eigenvalue obtained with SHIFT, the root mean squared (RMS (%)) and maximum (MAX(%)) absolute differences from the reference power distribution, as well as the number of iterations and time to completion. As expected, the diffusion-based solvers (NEM and SENM) perform comparably and yield the least accurate eigenvalue result, though overall the results are reasonably accurate. Introducing a higher order polar dependence with the SP_N and S_N solvers provides a significant improvement in the results. However, incorporating the azimuthal dependence into the leakages does not seem to have a considerable impact, actually yielding slightly worse results, at least with regard to the power distribution. While this is a bit troubling, a full explanation of the phenomenon is explained following the results from the rodded configurations.

Table 5.6: 3D-C5G7 Unrodded (UR) Results

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
MCNP	1.14308	—	—	—	—	—
SHIFT	1.14304	—	—	—	—	—
NEM	1.14182	-122	0.440	1.309	8	5.63
SENM	1.14182	-122	0.441	1.323	8	5.48
SP_3	1.14244	-60	0.257	0.798	8	5.41
SP_5	1.14246	-58	0.236	0.738	7	4.93
S_N ISO	1.14248	-56	0.214	0.635	6	5.21
S_N AZI-ISO	1.14233	-71	0.233	0.621	6	4.45
S_N MOM-0	1.14245	-59	0.244	0.714	14	13.44
S_N MOM-1	1.14314	10	0.275	0.856	13	13.05
S_N MOM-2	1.14312	8	0.276	0.858	14	15.11
S_N MOM-3	1.14312	8	0.274	0.855	13	15.03
S_N MOM-4	1.14312	8	0.275	0.857	13	16.02
S_N EXP	1.14312	8	0.275	0.857	13	20.19

Despite these unexpected results, the Fourier moment approach seems to perform well compared to the results with explicit angular dependence (Table 5.7), with even just one moment doing an admirable job of capturing the azimuthal dependence and in a fraction of the time. One would expect increasing the number of Fourier moments to approach the explicit angular representation, which is exactly what is observed.

Table 5.7: 3D-C5G7 Unrodded (UR) Moment Results (Compared to EXP TL)

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
S_N MOM-0	1.14245	-68	0.187	0.783	14	13.44
S_N MOM-1	1.14314	2	0.004	0.017	13	13.05
S_N MOM-2	1.14312	-0.17	0.002	0.004	14	15.11
S_N MOM-3	1.14312	0.02	0.001	0.003	13	15.03
S_N MOM-4	1.14312	0	0.000	0.001	13	16.02
S_N EXP	1.14312	—	—	—	13	20.19

Table 5.8 shows the results for the rodded configuration A (rA). Again, the SP_N and

S_N solvers provide notable improvements over the diffusion solvers. Incorporating the azimuthal dependence yields worse power distribution results (even a bit worse than what was observed in the unrodded configuration), though again the eigenvalue is improved.

Table 5.8: 3D-C5G7 Rodded Config. A (rA) Results

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
MCNP	1.12806	—	—	—	—	—
SHIFT	1.12810	—	—	—	—	—
NEM	1.12688	-122	0.248	0.928	6	4.44
SENM	1.12687	-123	0.251	0.940	6	4.49
SP_3	1.12752	-58	0.268	0.690	6	5.41
SP_5	1.12754	-56	0.264	0.680	7	4.93
S_N ISO	1.12753	-57	0.258	0.684	12	7.36
S_N AZI-ISO	1.12735	-75	0.284	0.838	12	7.40
S_N MOM-0	1.12753	-57	0.274	0.732	12	11.39
S_N MOM-1	1.12835	25	0.360	1.091	12	12.18
S_N MOM-2	1.12832	22	0.364	1.082	12	13.16
S_N MOM-3	1.12832	22	0.362	1.073	12	13.97
S_N MOM-4	1.12832	22	0.363	1.078	12	14.89
S_N EXP	1.12832	22	0.363	1.078	12	18.80

As before, only one Fourier moment seems necessary to adequately represent the azimuthal behavior, again in roughly half the time compared to the explicit azimuthal dependence (Table 5.9).

Table 5.9: 3D-C5G7 Rodded Config. A (rA) Moment Results (Compared to EXP TL)

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
S_N MOM-0	1.12753	-80	0.303	1.361	12	11.39
S_N MOM-1	1.12835	3	0.010	0.040	12	12.18
S_N MOM-2	1.12832	-0.22	0.001	0.008	12	13.16
S_N MOM-3	1.12832	0.21	0.002	0.008	12	13.97
S_N MOM-4	1.12832	0	0.000	0.001	12	14.89
S_N EXP	1.12832	—	—	—	12	18.80

Lastly, Table 5.10 shows the results for the rodded B configuration (rB). Because it has a considerably larger rodded portion than in other cases, the diffusion solvers perform significantly worse, and the SP_N and S_N solvers provide an excellent improvement. However, instead of yielding only slightly worse results, as before, including azimuthal dependence yields considerably larger error (roughly twice as large in the pin power error). This is very discouraging, despite the observed improvement in the eigenvalue. It should be kept in mind that the eigenvalue is the most integral quantity being assessed, and improvement in eigenvalue alone means very little.

It is also important to consider the discrepancy in the number of iterations necessary to converge with each of the solvers. In general, one can expect the two-node diffusion solvers to require the least number of iterations because by enforcing the zeroth moment of the scalar flux as a constraint, there are fewer moving parts and the flux being projected back to the radial solver will be the same as if the axial solver were not present, though the axial transverse leakage will be informed by the axial solver. When using a one-node formulation, as with the SP_N and S_N solvers, the scalar flux can change and understandably can take more iterations to resolve. With these problems, the eigenvalue residual seems to rapidly approach $1x10^{-5}$, but struggles a bit more to make it to $1x10^{-6}$. Additionally, incorporating an angle-dependent transverse leakage also adds some variability that could take extra iterations to resolve. In general, only one or two more iterations seem necessary.

Table 5.10: 3D-C5G7 Rodded Config. B (rB) Results

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
MCNP	1.07777	—	—	—	—	—
SHIFT	1.07772	—	—	—	—	—
NEM	1.07594	-178	0.770	3.545	7	5.02
SENM	1.07593	-179	0.781	3.608	7	4.94
SP_3	1.07701	-71	0.291	0.953	9	5.96
SP_5	1.07705	-67	0.292	0.939	6	4.51
S_N ISO	1.07701	-71	0.286	0.948	10	6.37
S_N AZI-ISO	1.07678	-94	0.316	1.093	10	6.46
S_N MOM-0	1.07700	-72	0.311	1.022	10	9.74
S_N MOM-1	1.07805	33	0.537	2.226	6	6.98
S_N MOM-2	1.07799	27	0.535	2.203	6	7.43
S_N MOM-3	1.07800	28	0.534	2.208	6	7.75
S_N MOM-4	1.07799	27	0.535	2.208	6	8.30
S_N EXP	1.07799	27	0.535	2.206	6	10.17

Yet again, the Fourier moment-based solvers and leakages do well in capturing the azimuthal behavior in reproducing results similar to the explicit representation (Table 5.11).

Table 5.11: 3D-C5G7 Rodded Config. B (rB) Moment Results (Compared to EXP TL)

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
S_N MOM-0	1.07700	-99	0.471	2.311	11	10.55
S_N MOM-1	1.07805	5	0.016	0.085	11	11.29
S_N MOM-2	1.07799	-0.52	0.002	0.013	11	12.40
S_N MOM-3	1.07800	0.6	0.002	0.011	11	12.94
S_N MOM-4	1.07799	0	0.000	0.002	11	13.90
S_N EXP	1.07798	—	—	—	11	17.32

Table 5.12 shows the results for the three different configurations compared to 3D-MOC. In the tables above, the absolute differences in the power distribution were reported as this emphasizes the power errors in high power regions. In the 3D-MOC results by Kochunas [29], relative pin power differences were reported. For consistency, the 2D/1D results

shown below are also relative, so there will be some differences from the previous results. Also included in this table are the core-hours required for each run. As can be seen, the 2D/1D runs require significantly less time (roughly two orders of magnitude difference) than 3D-MOC for comparable accuracy.

Table 5.12: 3D-C5G7 Comparison to 3D-MOC)

		Diff. (pcm)	RMS (%)	Max. (%)	Core-Hours
2D/1D S_N ISO	**UR**	-56	0.291	1.129	56.3
	rA	-57	0.419	1.260	79.5
	rB	-71	0.355	1.302	68.8
2D/1D S_N EXP	**UR**	8	0.324	0.777	218.0
	rA	22	0.500	1.536	203.0
	rB	26	0.633	1.483	187.1
3D-MOC	**UR**	-41	0.547	1.330	11805.1
	rA	-71	0.400	1.321	10827.4
	rB	153	0.325	1.182	10085.8

5.2.3 Discussion of Heterogeneous Pin Results

The results shown in Tables 5.6, 5.8, and 5.10 are very unsettling in that increasing the angular dependence of the transverse leakages yield worse results, particular in the case of rodded configuration B. In Section 3.6, the many sources of error in the 2D/1D framework were presented. In an effort to minimize the errors as much as possible, reasonably tight constraints were placed on the mesh (axial and radial) and quadrature. However, there are still a few sources of error that need to be considered. In Chapter 3, when the equations for the radial solvers was presented, two approximations to the axial transverse leakage were described, namely the isotropic and spatially flat approximations. Based on these results, it seems reasonable that perhaps there is cancellation of error between these two sources, yielding improved results for solvers using isotropic transverse leakages, and worse results for angle-dependent leakages, which will not benefit from the cancellation as one of the sources is being removed. To assess the truthfulness of this statement, alternative problems have been run using pin-homogenized cross sections (Figure 5.7), which are available in Appendix C.

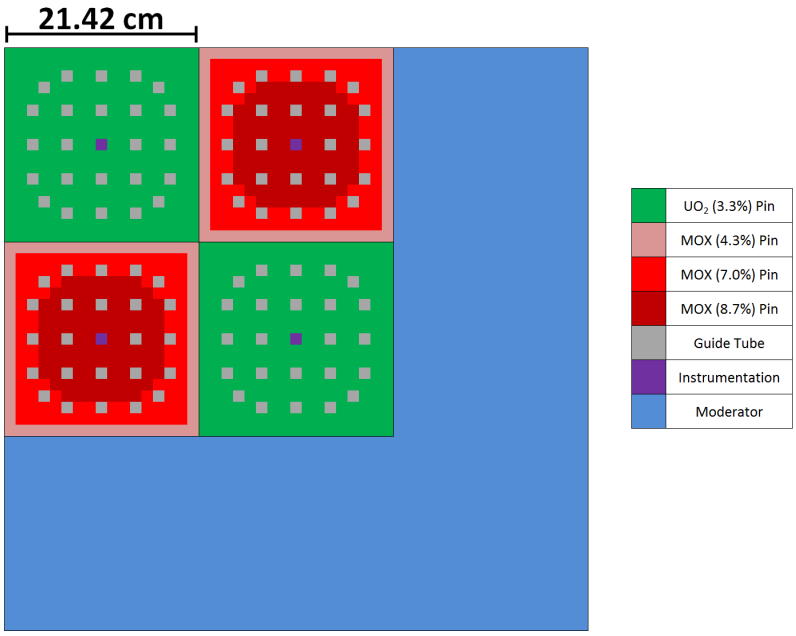

Figure 5.7: C5G7 - Radial Layout

5.2.4 3D, Homogeneous Pin Results

Table 5.13 shows the results for the unrodded configuration with homogenized pins. In general, all of the results are good, but an extreme improvement in both the eigenvalue and power distribution are observed by including the azimuthal dependence of the leakages. It should be noted that by using homogenized pins, the flat axial transverse leakage distribution approximation become more valid, but it is likely still not exact, so there could still be some error introduced from it, though it should be much lower that with the heterogeneous pins. Additionally, though the other parameters in these runs were well refined, there are still likely some small errors from ray spacing, quadrature, and mesh discretization. All of the discretization parameters in these cases are identical to those used in the heterogeneous pin problems, except for the radial mesh for the fuel and control rod pins, which was switched from a cylindrical mesh to a Cartesian mesh but with a 15x15 submeshing. By homogenizing the fuel and moderator within each pin, fuel is moved out to the corners of the pins, which have a larger mesh size. This is typically acceptable with heterogeneous pins as the fuel is meshed separately, but switching to a Cartesian submeshing allows for all sections of each pin to be well discretized.

Table 5.13: C5G7-UR (hom.) Results

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
SHIFT	1.14554	—	—	—	—	—
NEM	1.14427	-127	0.219	0.634	8	5.39
SENM	1.14427	-127	0.221	0.635	8	5.41
SP_3	1.14488	-66	0.202	0.817	7	4.93
SP_5	1.14491	-63	0.220	0.882	6	4.44
S_N ISO	1.14491	-63	0.218	0.925	13	7.82
S_N AZI-ISO	1.14475	-79	0.259	1.100	13	8.03
S_N MOM-0	1.14491	-65	0.230	0.974	13	11.95
S_N MOM-1	1.14559	5	0.038	0.145	13	13.12
S_N MOM-2	1.14558	4	0.040	0.143	13	14.17
S_N MOM-3	1.14558	4	0.041	0.145	13	15.05
S_N MOM-4	1.14558	4	0.041	0.144	13	16.07
S_N EXP	1.14558	4	0.041	0.144	13	20.30

Figure 5.8 and 5.9 show the errors in the pin power distribution with respect to the SHIFT reference data. Because an absolute difference was used here, it is not surprising that the pin power error peaks in the highest power (bottom) plane. It is likely that the angular dependence of the leakages at the UO_2-MOX assembly interfaces is poorly represented with an isotropic distribution. The errors are considerably reduced when including the full angular dependence.

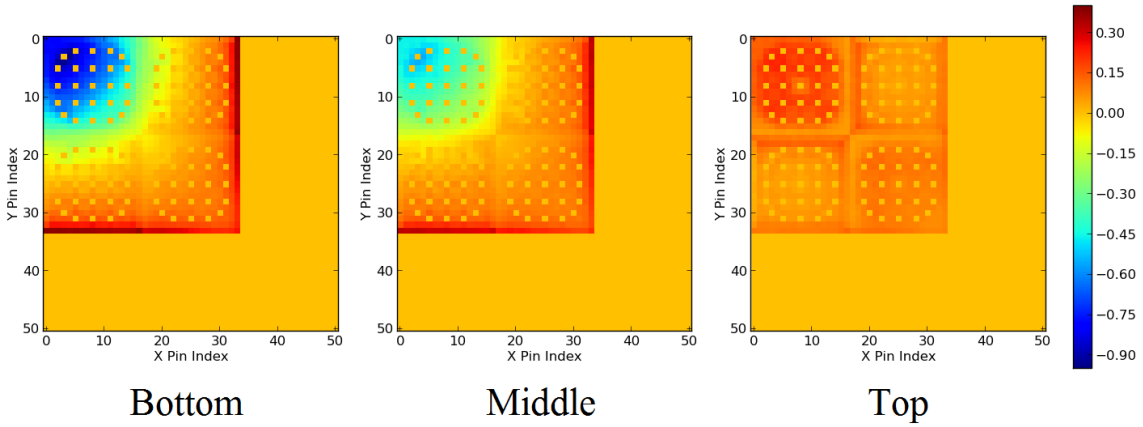

Figure 5.8: C5G7-UR (hom.) - Pin Power Error (ISO TL)

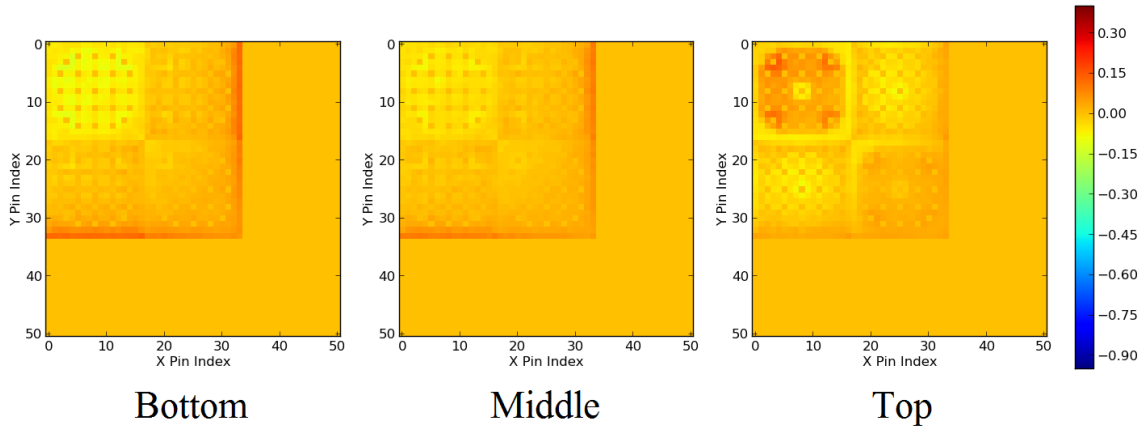

Figure 5.9: C5G7-UR (hom.) - Pin Power Error (EXP TL)

As with the heterogeneous pin problems, excellent performance using Fourier moments is observed as in Table 5.14.

Table 5.14: C5G7-UR (hom.) Moment Results (Compared to EXP TL)

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
S_N MOM-0	1.14491	-69	0.201	0.861	13	11.95
S_N MOM-1	1.14559	2	0.005	0.019	13	13.12
S_N MOM-2	1.14558	-0.1	0.001	0.002	13	14.17
S_N MOM-3	1.14558	0.01	0.001	0.002	13	15.05
S_N MOM-4	1.14558	0	0.000	0.000	13	16.07
S_N EXP	1.14558	—	—	—	13	20.30

Table 5.15 shows the results for the rodded configuration A. Again, a huge reduction in the error is observed with the root mean square and maximum pin power errors reducing by a factor of 3-4 and the eigenvalue error almost completely removed.

Table 5.15: C5G7-rA (hom.) Results

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
SHIFT	1.12934	—	—	—	—	—
NEM	1.12786	-148	0.483	1.917	7	4.99
SENM	1.12786	-148	0.486	1.916	7	4.90
SP_3	1.12854	-80	0.291	1.286	6	4.49
SP_5	1.12856	-78	0.294	1.319	6	4.71
S_N ISO	1.12854	-80	0.300	1.436	6	4.57
S_N AZI-ISO	1.12835	-99	0.366	1.752	6	4.49
S_N MOM-0	1.12853	-81	0.314	1.507	12	11.23
S_N MOM-1	1.12939	5	0.084	0.387	12	12.40
S_N MOM-2	1.12936	2	0.088	0.405	12	13.83
S_N MOM-3	1.12937	3	0.086	0.399	12	14.55
S_N MOM-4	1.12936	2	0.087	0.406	12	15.89
S_N EXP	1.12936	2	0.087	0.406	12	23.94

Figures 5.10 and 5.11 show the pin power distribution errors. These are similar but slightly larger than the errors observed in the unrodded case. This is not surprising, however, since one would expect the control rod to introduce more importance to the azimuthal dependence of the leakages. The remaining error in the explicit results could be from either the axial transverse leakage shape approximation, which is not guaranteed to be exactly flat even with homogeneous pins, or mesh discretization.

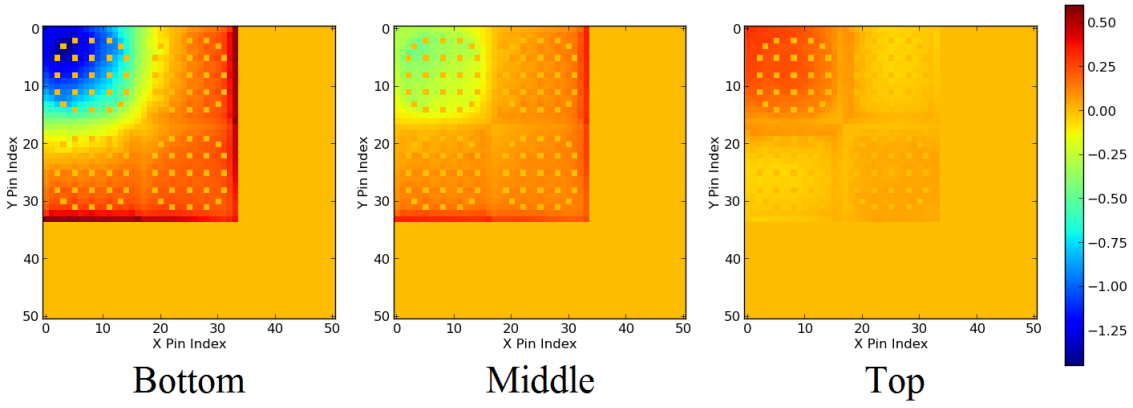

Figure 5.10: C5G7-rA (hom.) - Pin Power Error (ISO TL)

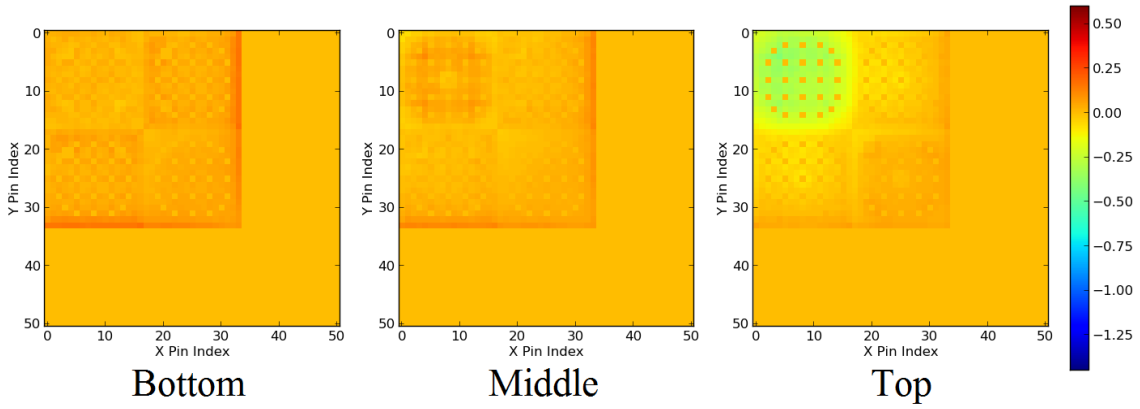

Figure 5.11: C5G7-rA (hom.) - Pin Power Error (EXP TL)

Table 5.16 shows the results of the moment solver compared to explicit representation

Table 5.16: C5G7-rA (hom.) Moment Results (Compared to EXP TL)

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
S_N MOM-0	1.12853	-83	0.327	1.512	12	11.19
S_N MOM-1	1.12939	3	0.011	0.038	12	12.22
S_N MOM-2	1.12936	-0.28	0.002	0.010	12	13.21
S_N MOM-3	1.12937	0.17	0.002	0.007	12	14.00
S_N MOM-4	1.12936	-0.01	0.000	0.001	12	14.97
S_N EXP	1.12936	—	—	—	12	18.83

Table 5.17 show the results for the rodded B configuration. As with the heterogeneous pin results, the diffusion solvers yield the worst results, and while the SP_N and S_N solvers with isotropic transverse leakages provide improvement, the moment and explicit transverse leakages again reduce the errors considerably.

Table 5.17: C5G7-rB (hom.) Results

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
SHIFT	1.07468	—	—	—	—	—
NEM	1.07241	-227	1.341	6.247	7	5.03
SENM	1.07239	-229	1.353	6.324	7	4.97
SP_3	1.07360	-108	0.468	2.241	8	5.45
SP_5	1.07364	-104	0.444	2.097	8	5.51
S_N ISO	1.07360	-108	0.488	2.365	7	4.89
S_N AZI-ISO	1.07336	-132	0.590	2.886	7	5.16
S_N MOM-0	1.07359	-109	0.481	2.354	10	9.63
S_N MOM-1	1.07470	2	0.151	0.690	11	11.67
S_N MOM-2	1.07464	-4	0.152	0.723	10	11.26
S_N MOM-3	1.07465	-3	0.150	0.714	10	11.94
S_N MOM-4	1.07465	-3	0.151	0.724	10	12.74
S_N EXP	1.07465	-3	0.151	0.724	10	15.83

Similar behavior is observed in Figures 5.12 and 5.13 and Table 5.18.

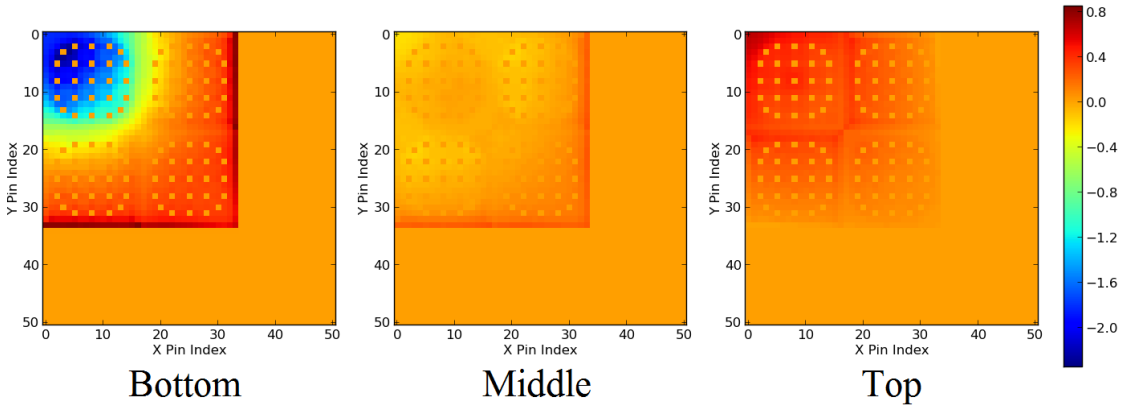

Figure 5.12: C5G7-rB (hom.) - Pin Power Error (ISO TL)

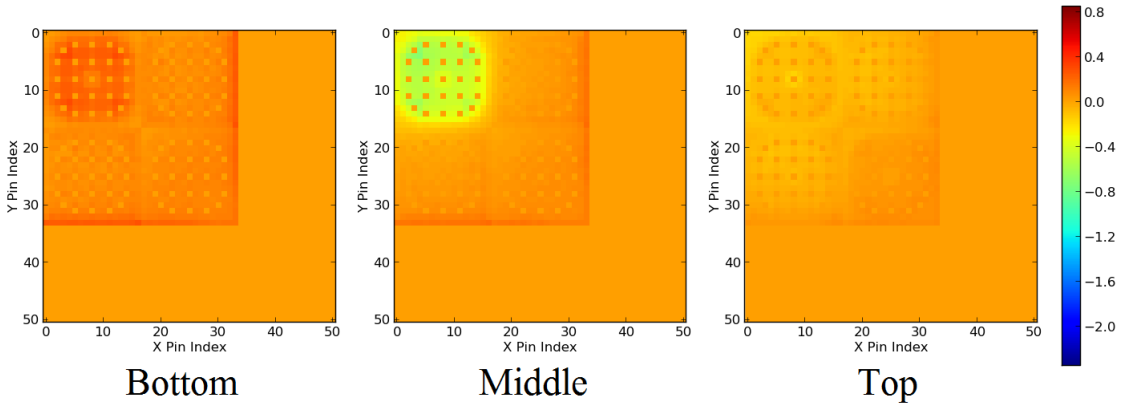

Figure 5.13: C5G7-rB (hom.) - Pin Power Error (EXP TL)

Table 5.18: C5G7-rB (hom.) Moment Results (Compared to EXP TL)

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
S_N MOM-0	1.07359	-106	0.522	2.654	10	9.63
S_N MOM-1	1.07470	5	0.018	0.090	11	11.67
S_N MOM-2	1.07464	-0.7	0.002	0.015	10	11.26
S_N MOM-3	1.07465	0.51	0.002	0.011	10	11.94
S_N MOM-4	1.07465	-0.01	0.000	0.002	10	12.74
S_N EXP	1.07465	—	—	—	10	15.83

5.2.5 Axial Mesh Refinement

While the results in the previous section look very good with explicit leakages, there are still some localized errors near the rod tips, as shown in Figures 5.11 and 5.13. To reduce these errors, the axial mesh was reduced from 3.57 cm to 1.785 cm and 0.8925 cm. Though the unrodded configuration results are fairly unaffected (as in Table 5.19 and Figures 5.14 and 5.15), both rodded configurations see noteworthy reductions in the error distribution (Tables 5.20 and 5.21; Figures 5.16 through 5.19).

Table 5.19: C5G7-UR (hom.) Mesh Refinement

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
S_N EXP (3.5700 cm)	1.14558	4	0.041	0.144	13	20.30
S_N EXP (1.7850 cm)	1.14557	3	0.039	0.143	11	26.17
S_N EXP (0.8925 cm)	1.14557	3	0.039	0.134	16	66.90

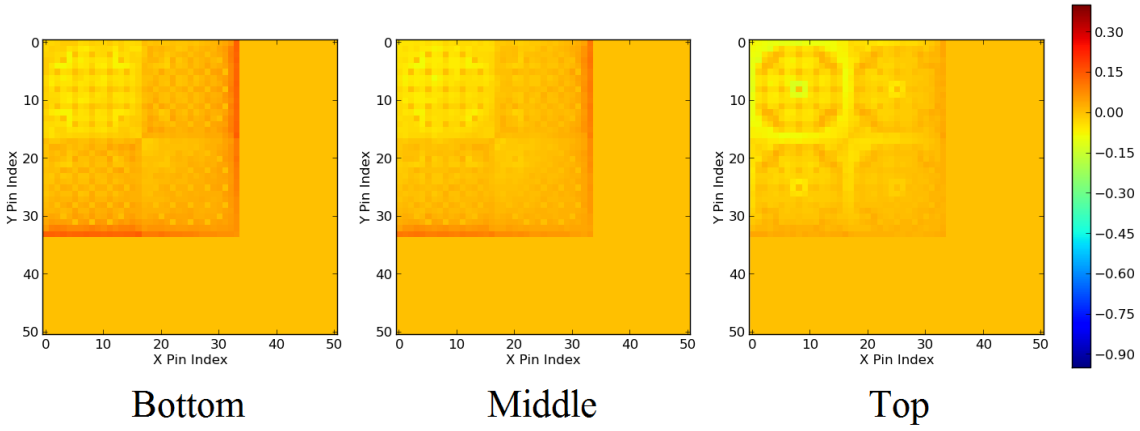

Figure 5.14: C5G7-UR (hom.) - Pin Power Error (EXP TL) - 1.785 cm Axial Mesh

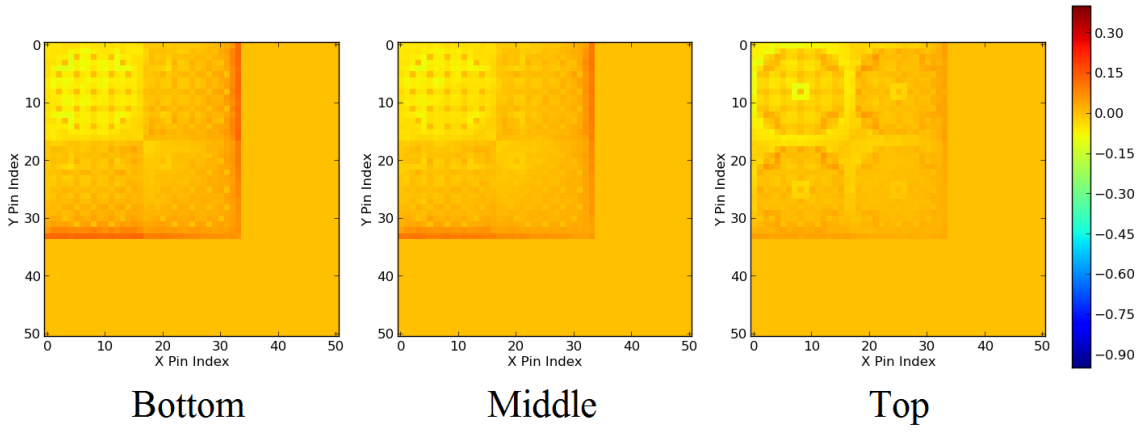

Figure 5.15: C5G7-UR (hom.) - Pin Power Error (EXP TL) - 0.8925 cm Axial Mesh

Table 5.20: C5G7-rA (hom.) Mesh Refinement

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
S_N EXP (3.5700 cm)	1.12936	2	0.086	0.407	12	18.83
S_N EXP (1.7850 cm)	1.12935	1	0.060	0.246	13	20.49
S_N EXP (0.8925 cm)	1.12935	1	0.053	0.173	16	67.05

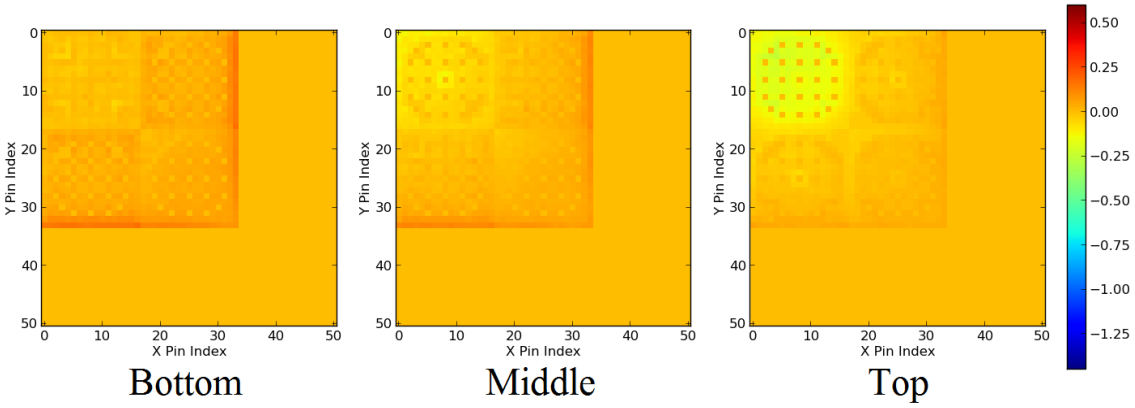

Figure 5.16: C5G7-rA (hom.) - Pin Power Error (EXP TL) - 1.785 cm Axial Mesh

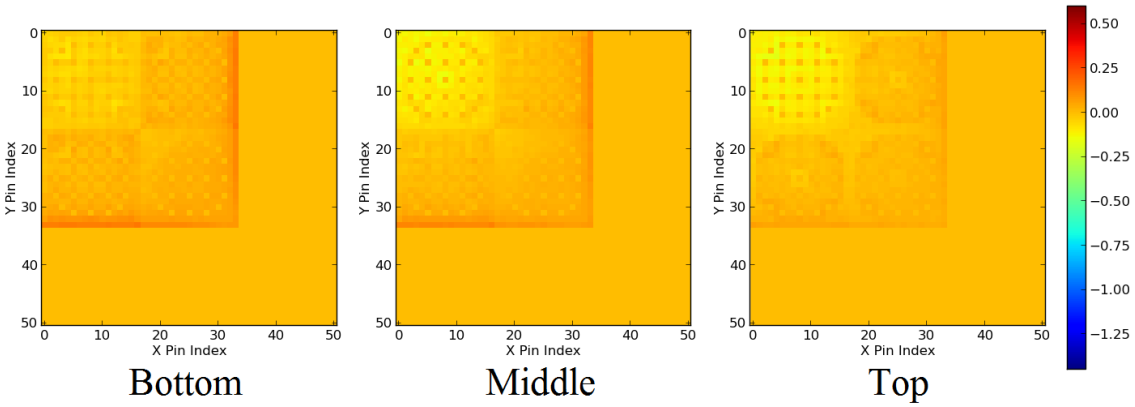

Figure 5.17: C5G7-rA (hom.) - Pin Power Error (EXP TL) - 0.8925 cm Axial Mesh

Table 5.21: C5G7-rB (hom.) Mesh Refinement

	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
S_N EXP (3.5700 cm)	1.07465	-3	0.151	0.724	10	15.83
S_N EXP (1.7850 cm)	1.07464	-4	0.093	0.422	14	31.56
S_N EXP (0.8925 cm)	1.07466	-2	0.077	0.319	21	87.05

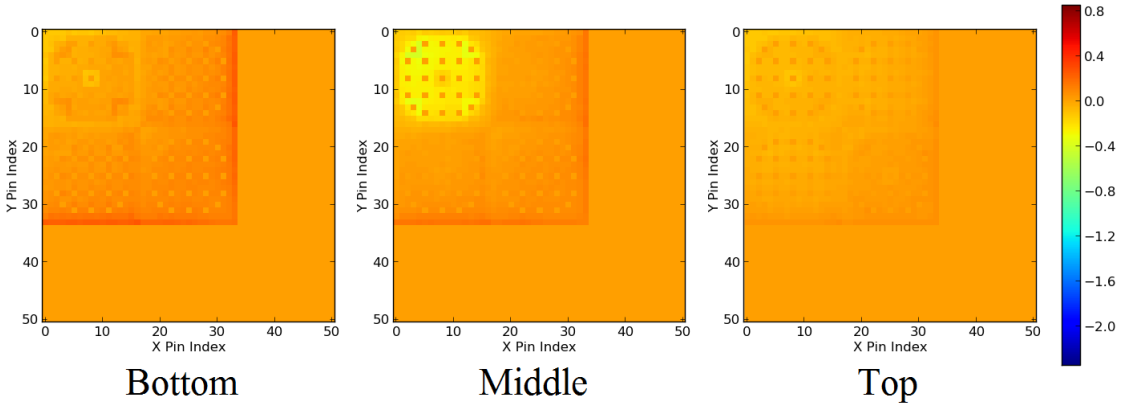

Figure 5.18: C5G7-rB (hom.) - Pin Power Error (EXP TL) - 1.785 cm Axial Mesh

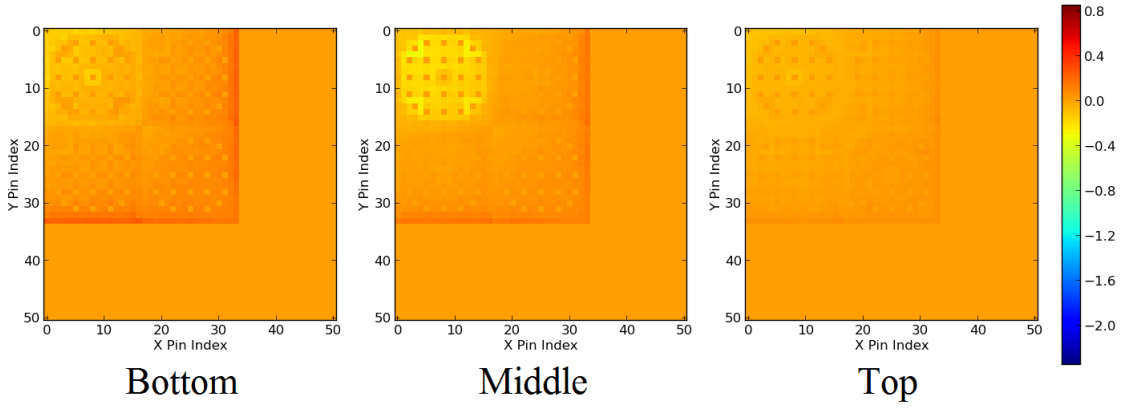

Figure 5.19: C5G7-rB (hom.) - Pin Power Error (EXP TL) - 0.8925 cm Axial Mesh

5.2.6 Cancellation of Error

In the previous subsection, both the heterogeneous and homogeneous pin results were shown. To more easily illustrate the claim that cancellation of error is a factor in the heterogeneous pin results, the rodded configuration B results for the solvers with isotropic

transverse leakages have been repeated in Table 5.22. It can be seen that the heterogeneous pin results are consistently better than with homogeneous pins, despite the fact that the homogeneous pin case should be easier to resolve correctly. In fact, the pin power errors for the SP_N and S_N solvers nearly doubles. This is highly suggestive that cancellation of error plays a significant result in the heterogeneous pin results and highlights obtaining a more accurate spatial distribution of the axial transverse leakage for future research.

Table 5.22: C5G7-rB (hom.) Results

	Axial Solver	Eig.	Diff. (pcm)	RMS (%)	Max. (%)	Iters.	Time (min)
Het. Pins	**NEM**	1.07594	-178	0.770	3.545	7	5.02
	SENM	1.07593	-179	0.781	3.608	7	4.94
	SP_3	1.07701	-71	0.291	0.953	9	5.96
	SP_5	1.07705	-67	0.292	0.939	6	4.51
	S_N ISO	1.07701	-71	0.286	0.948	10	6.37
Hom. Pins	**NEM**	1.07241	-227	1.341	6.247	7	5.03
	SENM	1.07239	-229	1.353	6.324	7	4.97
	SP_3	1.07360	-108	0.468	2.241	8	5.45
	SP_5	1.07364	-104	0.444	2.097	8	5.51
	S_N ISO	1.07360	-108	0.488	2.365	7	4.89

5.2.7 Memory Footprint and Timing Comparison

In the previous sections, most of the emphasis in comparing the moment-based solvers was placed on the accuracy and timing. In this section, a bit more attention will be given to comparing the timing for the two moment-based MOC sweepers that were covered in Section 4.3, as well as provide some insight into the memory savings that can be achieved from the moment-based approach.

Table 5.23 shows the timing results for the moment-based sweepers for the C5G7-rB case with heterogeneous pins. The first column shows the timing using an explicit axial transverse leakage. In this case, the axial sweeper uses the Fourier moments to formulate the axial transverse leakage. The primary difference in the run times for this column are because of the axial solver, which needs to do considerably less work than the explicit solver. The next column indicates the timing for the "Slow" moment-based MOC sweeper which evaluates the moment-based leakage source on-the-fly during the MOC sweep. Later in this section, the memory savings of this approach will be seen, but the timing results

demonstrate that this is not an ideal approach at all, costing as much time as the explicit solver after just three Fourier moments. The last column shows the results for the "Fast" sweeper which was used in all previous results. This approach requires temporarily storing the expanded source for two azimuthal angles at a time, instead of evaluating the source for each fine mesh region as it is encountered. The timing results show excellent improvement over its "Slow" counterpart, and later it will be seen that the increased memory burden is not very considerable (amounts to roughly one more Fourier moment).

Table 5.23: 3D-C5G7 Rodded Configuration B (rB) Timing Results

	Total Time (min)	
	Axial TL/MOC Sweeper	
Radial TL (# of Moments)	**"Slow" Moment**	**"Fast" Moment**
0	11.86	10.55
1	17.52	11.29
2	21.32	12.40
3	29.85	12.94
4	38.46	13.90
Explicit	17.32	

Table 5.24 shows the total memory footprint for one plane for each of the cases. The results for the 0^{th} moment and explicit cases were obtained using the Massif heap memory profile within Valgrind using only planar decomposition; the other moment results were interpolated, scaling the appropriate constructs as necessary. This is particularly simple since the anticipated size relationships are well known. In general, the memory footprint would be similar if using radial decomposition, with the primary difference being the angular flux storage on the parallel radial domains, which is not included with only planar decomposition. This difference is negligible compared to the other variables. If only one Fourier moment were used, for example, the memory burden would be reduced from 6.226 GB per plane to 1.648 GB with the "slow" sweeper and 1.707 GB with the "Fast" sweeper. While a 3-4x reduction in memory is very significant, a possible thought is that it should be more. In this case, the refined ray spacing of 0.01 cm limits the reduction possible, as the ray tracing data is a significant portion of the memory footprint (as illustrated in Figure 5.20). Essentially, the reduction is limited by the memory requirements of variables unrelated to the Fourier moment expansion (i.e. geometric data, ray tracing, etc.).

Table 5.24: 3D-C5G7 Unrodded (UR) Total Memory (Fuel Plane) Results

Radial TL (# of Moments)	Total Memory (GB)	
	Axial TL/MOC Sweeper	
	"Slow" Moment	"Fast" Moment
0	1.498	1.557
1	1.648	1.707
2	1.798	1.857
3	1.948	2.007
4	2.098	2.157
Explicit	6.226	

Figure 5.20 shows the memory requirements of several key components for the explicit transverse leakages. In this case, a majority of the memory is occupied by the sources used for the 2D-MOC sweepers, which now have full angular dependence. It is important to note that there are two source variables (qi1g and qbar); qi1g stores the incoming scattering and fission sources and qbar stores qi1g plus the self-scattering source. This is done to alleviate the burden of setting up the source multiple times during inner iterations, where the self-scattering source should be updated. Ongoing work is looking to eliminate the inner iterations entirely, which would eliminate the need for two source variables. However, it is should also be noted that nearly 17% of the memory is taken up by the rays for the MOC solver. These cases were run using assembly modular ray tracing (AMRT) to reduce the time spent during initialization as cell modular ray tracing (CMRT) is currently burdened by very large initialization times. Though it would be possible to run with CMRT to reduce this component, most realistic cases require the resolution of an assembly gap, in which case AMRT is necessary. It is just important to consider that both the "MOC Rays" and "Other" components in this chart are not impacted by using a Fourier expansion for the azimuthal behavior. Only the "Axial Solver" and "MOC Source" components will be reduced.

Figures 5.21 through 5.23 show the memory distribution when using zero, two, and four Fourier moments, respectively. As one would expect, the "MOC Rays" become the dominant portion of the burden, though the "MOC Source" and "Axial Solver" components are reduced dramatically.

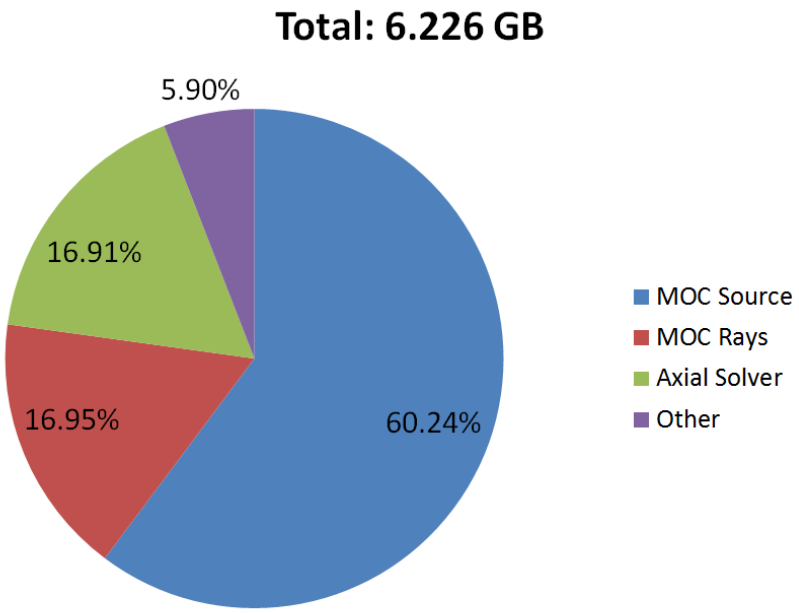

Figure 5.20: Explicit TL Memory Profile

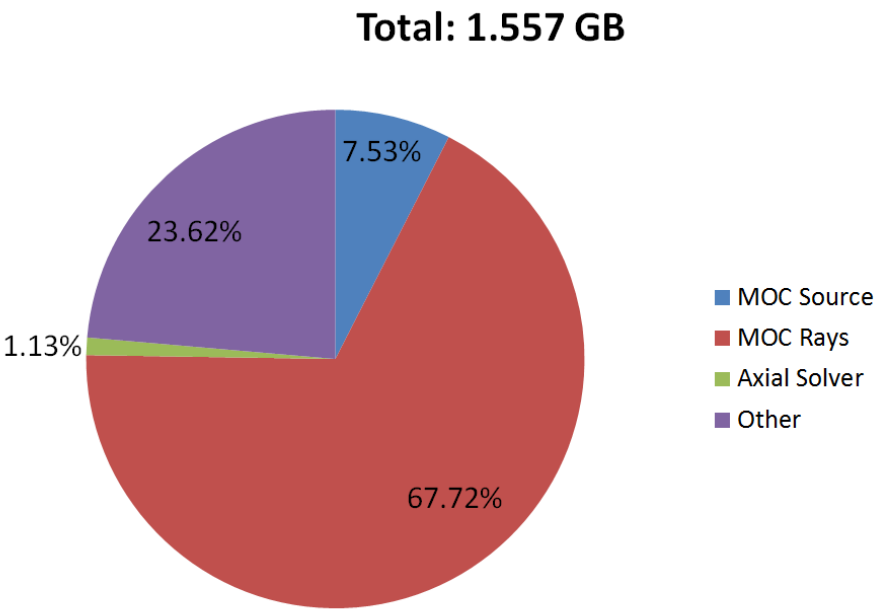

Figure 5.21: Moment-0 TL Memory Profile

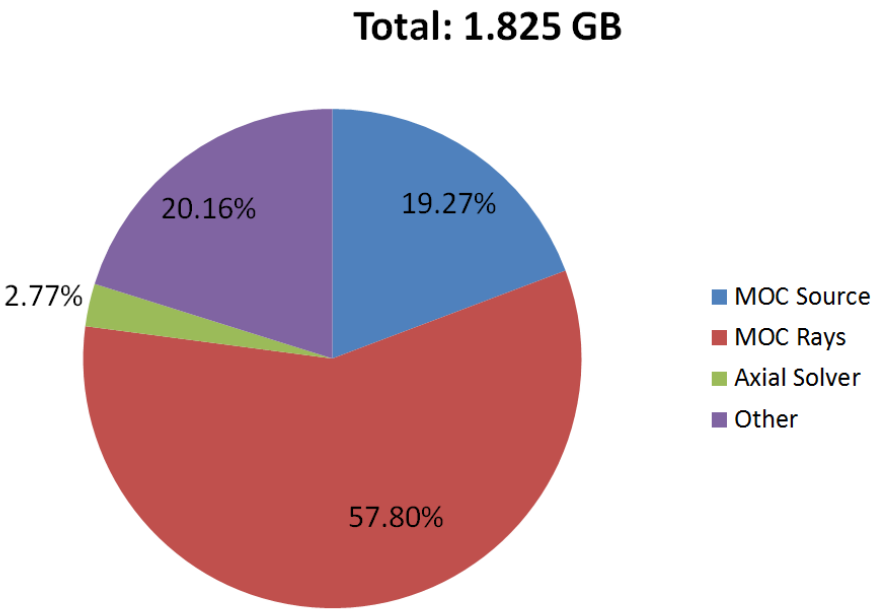

Figure 5.22: Moment-2 TL Memory Profile

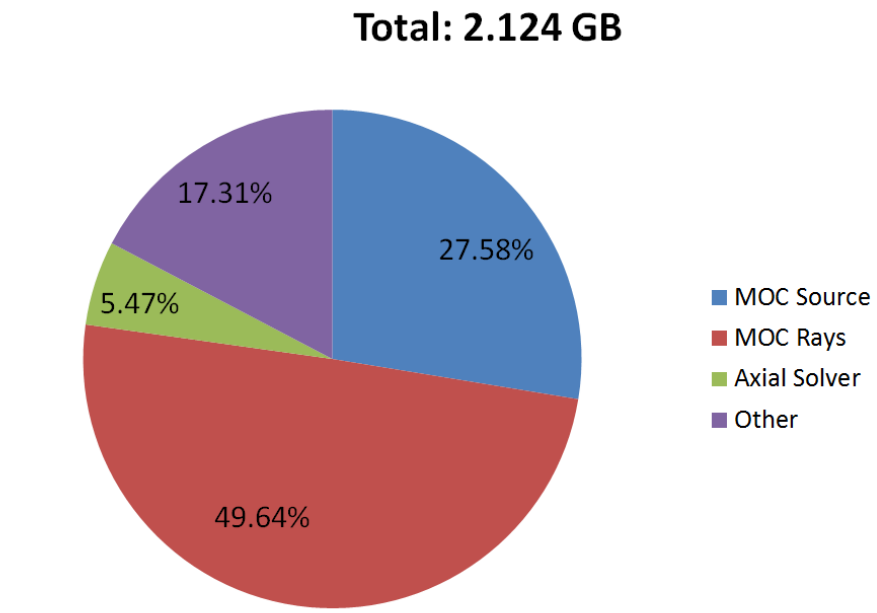

Figure 5.23: Moment-4 TL Memory Profile

To give a more intuitive and satisfying result, Table 5.25 shows the memory comparison for only the "MOC Source" and "Axial Solver" components, which can be approximately

related using the following equations for the memory savings for the "Slow" (Eq. 5.2) and "Fast" (5.3), where ξ_{moc} and ξ_{axial} denote the fraction of the memory that belongs to the MOC source of the axial solver ($\xi_{moc} + \xi_{axial} = 1$):

$$Savings\ Factor \approx \frac{N_{azimuthal}}{(1 + 2N_{mom})}, \tag{5.2}$$

$$Savings\ Factor \approx \frac{N_{azimuthal}}{\xi_{moc}(2 + 2N_{mom}) + \xi_{axial}(1 + 2N_{mom})}. \tag{5.3}$$

Neither of these equations are exactly correct as there is one component in the axial solver storage that also stores the isotropic radial transverse leakage. This component scales exactly as in Eq. 5.4, and is typically only a small portion of the memory problem, which is why Eq. 5.2 and 5.3 are typically very accurate:

$$Savings\ Factor \approx \frac{N_{azimuthal} + 1}{1 + 2N_{mom} + \frac{1}{N_{pol}}}. \tag{5.4}$$

It should be noted that the savings factor increases as the number of azimuthal angles increases, since the moment-based storage requirements are static. This makes quadrature refinement much more tractable, as well.

Table 5.25: 3D-C5G7-UR Angular Construct Memory (Fuel Plane) Results

	Memory (MB)	
	Axial TL/MOC Sweeper	
Radial TL (# of Moments)	**"Slow" Moment**	**"Fast" Moment**
0	77.00	137.01
1	230.70	290.71
2	384.41	444.42
3	538.11	598.12
4	691.82	751.83
Explicit	4,919	

As expected, the memory burden of these components scales much more in accordance with what one would predict just based on the theory behind the Fourier moments. It is also important to note that with the current MOC iteration strategy, the source from only one group is stored at a time. Ongoing work is considering changing the iteration strategy such that the source for multiple groups will need to be stored at a time. This could be

devastating to the memory footprint when using angle-dependent sources for each region. While moving to a moment-based approach will again make this more manageable, special considerations will need to be made to ensure that the memory requirements are tractable for each problem.

5.2.8 Parallel Performance

To demonstrate the parallel performance, a smaller problem using 3x3 pin assemblies instead the full 17x17 was built. Additionally, the number of axial planes was reduced from 18 to 16 to yield more factors for building the parallel trends. In Section 4.4, the spatial and angular decomposition schemes for the explicit and Fourier moment-based sweepers were described. All results in this section are the average of five, independent cases.

Figures 5.24 and 5.25 show the strong scaling results for the planar and angular decomposition, respectively. 2, 4, 8, and 16 processors were used, with 16 being full decomposed with one plane per process. The planar decomposition results look very good, even despite being slightly superlinear. It is hypothesized that the serial case has more issues with cache efficiency than the other cases, which leads to the serial timing being slightly longer, causing the other results to be faster than ideal. Regardless, the scaling is excellent, though some improvements could be made the reduce the superlinear behavior. Only planar decomposition was shown as including more spatial domains leads to radial decomposition as well. Because there is no communication necessary radially, this would be expected to scale ideally (just as the planar MOC scales ideally when considering planar decomposition).

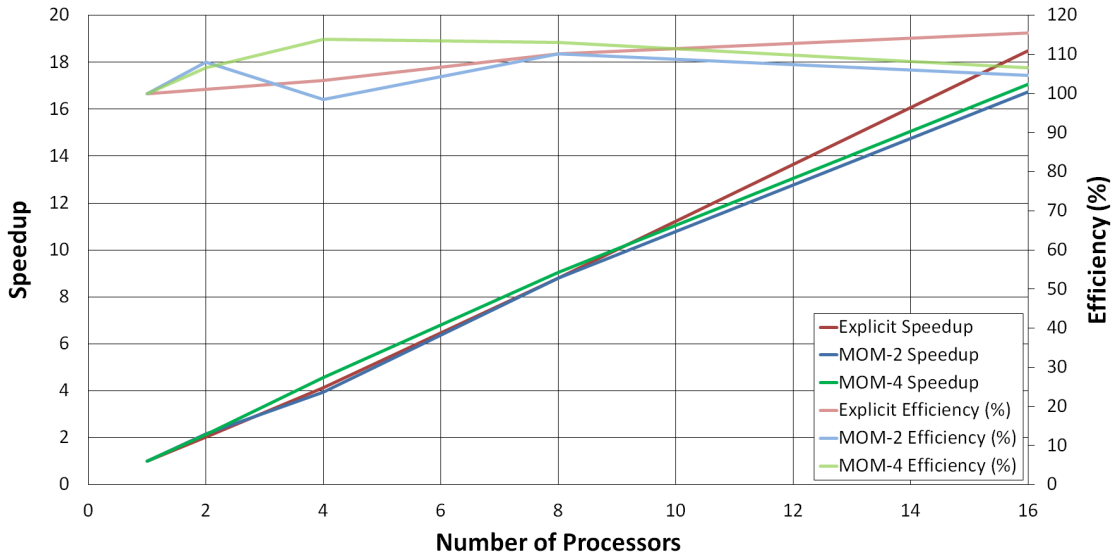

Figure 5.24: Planar Decomposition Strong Scaling

The angular decomposition results are a bit more interesting. First consider the scaling for the explicit solver where the efficiency drops off fairly quickly. While there is not an imbalance in the partitioning of the explicit azimuthal angles, there is a reduction for the angular fluxes (to be used in construction the axial transverse leakage) on each axial surface, which becomes particularly costly as the number of processors increases. This reduction could be more restricted, but at this point in time, the azimuthal angle partitioning on the axial sweeper is different than the partitioning for the radial sweeper, so a reduction of all angles is necessary. The scalar flux moments and axial currents are also reduced. Additionally, there are some components of the sweep, particularly the routine to calculate and setup the source moments, which are more limited in parallelization.

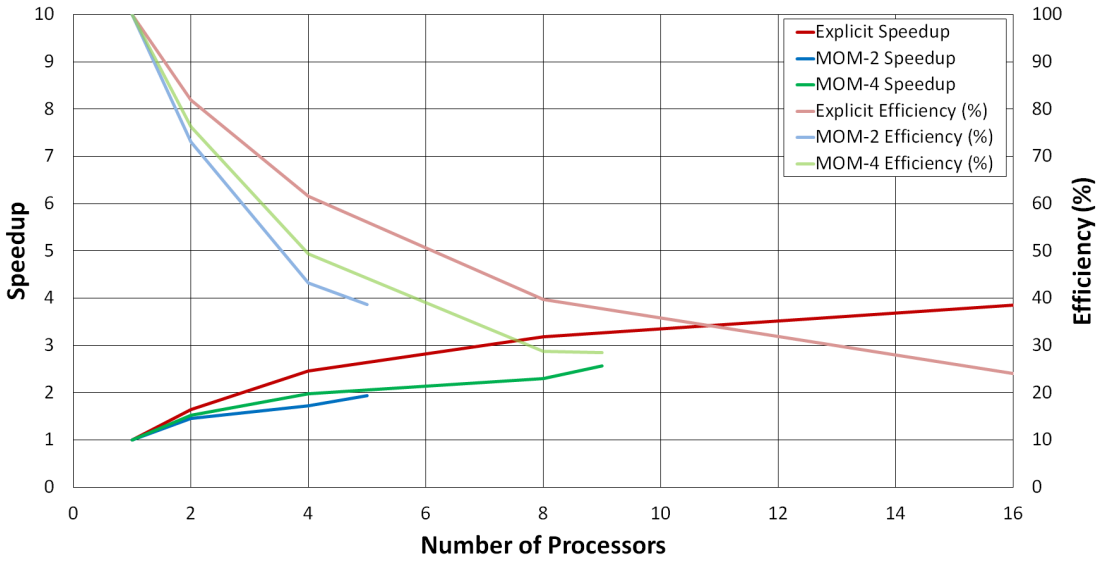

Figure 5.25: Angular Decomposition Strong Scaling

Similar behavior is observed with the moment-based solver, though it is important to note that only in serial or with full decomposition is the partitioning perfectly balanced, which is why the efficiency trend flattens off for the last data point. Not surprisingly, the efficiency for the moment-based solvers is slightly less than the explicit solver since the moment-based solvers are more decomposed on a similar number processors. There is also a reduction on the angular fluxes, as with the explicit sweeper, but this one is non-negotiable and is essential for constructing the axial transverse leakage, regardless of partitioning. Also significant is that the zeroth moment is responsible for tallying the scalar flux moments and currents, which instead of being reduced over all angular decomposition processors, can now be broadcast from the zeroth moment to the rest, saving the overhead of a reduction. Unfortunately, since the zeroth moment has more work than the other moments, in full decomposition, the efficiency is worse because the other moments are waiting for the zeroth to finish. This can be mitigated when using an imbalanced partition scheme by ensuring that the zeroth moment is located on the decomposition domain with the fewest number of moments.

Lastly, Figure 5.26 shows the weak scaling for the moment-based sweeper. With weak scaling the goal is for the amount of work on each processor to remain constant as the number of processors is increased, so these results increments the number of moments in the expansion, fully decomposing in each case. Ideally, the amount of time to execute would remain constant. However, the communication between the processors (through reduction or broadcast) significantly reduces the efficiency as the number of moments

increases.

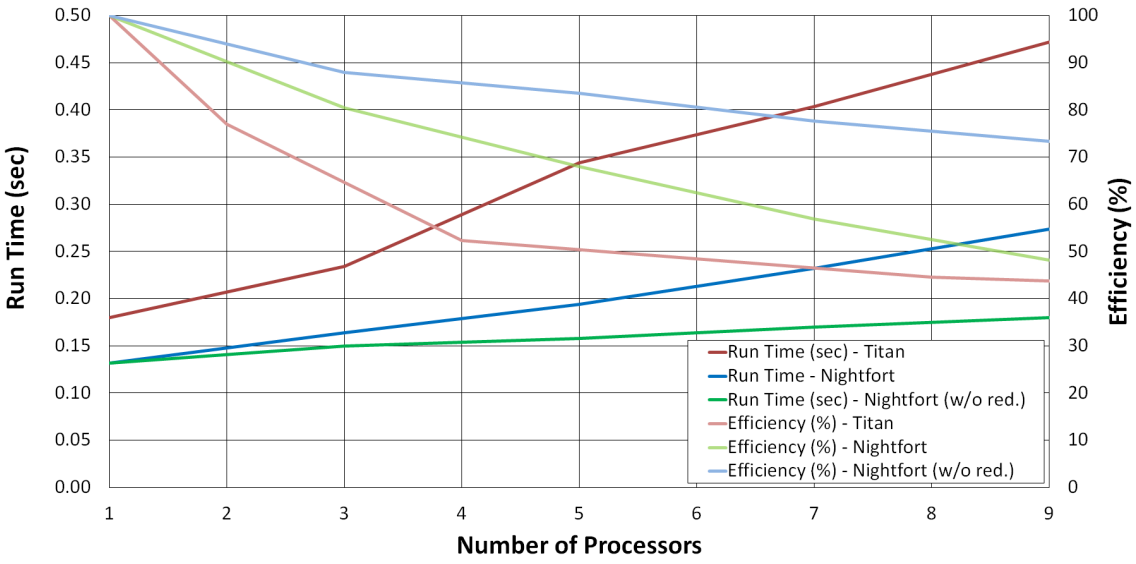

Figure 5.26: Moment Decomposition Weak Scaling

CHAPTER 6

Conclusions

6.1 Summary of Work

To build up the motivation for the novel azimuthal, Fourier moment-based axial S_N sweeper proposed here, the first few chapters were dedicated to presenting the theory behind neutron transport as well as common numerical methods used to simulate it, serving as a good foundation for presenting the 2D/1D framework. It was also shown how the solvers encompassed by the 2D/1D concept, which decomposes 3D problems in separate axial and radial solvers, are coupled together using radial and axial transverse leakages, respectively.

A typical approximation is to assume that transverse leakages are isotropic. Work by Hursin [19, 20] incorporated the polar angle dependence into these leakages. The work presented here builds off of that idea by including the azimuthal dependence. Because the number of azimuthal angles tends to be considerably larger than the number of polar angles, adding the dependence greatly increases the computational burden, particularly with regards to the memory requirements. In explicitly simulating the azimuthal dependence, this work effectively acts as a bridge between the "2D/1D" concept and the "2D/1D Fusion" concept, which models full angular dependence and performs axial solves on subpin level regions instead of pin-homogenized domains. To reduce the burden associated with explicit azimuthal dependence, a Fourier series expansion was proposed for the angular fluxes in the axial solver, as well as for the transverse leakages in both solvers.

A new axial solver using these expansions was derived and designed to wrap a cubic characteristics-based S_N kernel to handle the spatial distribution. Results based on initial visualizations of the radial transverse leakages for test problems, suggested that only a small number of Fourier moments would be necessary (roughly two moments, which include a flat component and two sine and cosine components). This notion was verified in all benchmark cases evaluated, in which one Fourier moment typically yielded errors on the

order of tens of pcm. For the Takeda benchmark, the azimuthally-integrated and isotropic transverse leakages yield several hundred pcm difference from full angular dependence. This error was reduced to 42 pcm with one Fourier moment and 2 pcm with two moments. For the C5G7 problem, even just one Fourier moment yielded less than 5 pcm difference from explicit representation. However, there were some interesting trends observed in the C5G7 problems.

In the Takeda benchmark, where the core is simulated as a homogeneous cube, explicit representation compares very well to the Monte Carlo reference solution. But in the 3D-C5G7 benchmark cases, the fuel and moderator within each pin are heterogeneous, as is the case in more realistic problems. With these problems, the results tended to be worse than with isotropic or azimuthally-integrated transverse leakages. One of the remaining sources of error, even after incorporating full angular dependence, is the spatial distribution of the axial transverse leakage within each pin. These results suggest that cancellation of error may be a major factor in allowing the isotropic leakages to yield more accurate results.

To test this, alternative C5G7 benchmarks were formulated using pin-homogenized cross sections in which the flat axial transverse leakage approximation should be more valid. It is not likely to be exact, as there could still be some shape within the pin, but it should be considerably more accurate than with the standard, heterogeneous pins. Reference solutions for these (and the standard C5G7) problems were obtained using the Monte Carlo code, SHIFT, developed at the Oak Ridge National Laboratory. SHIFT was initially used to generate more tightly converged reference solutions to the heterogeneous pin cases, which made it very simple to generate solutions for the cases with homogeneous pins. As anticipated, the results for the homogeneous pin cases were in excellent agreement with the explicit and moment-based results, proving that cancellation of error between the angular and spatial representation of the transverse leakages is a factor.

With results in hand to advocate the higher order angular representation and the Fourier moment approach, additional performance characteristics were evaluated such as memory usage and parallel scaling. Significant reductions in memory usage were observed, particularly in the radial source and axial storage constructs, which were very heavy with explicit representation; the requirements for these variables were reduced by an order of magnitude. Because the storage for other components, such as MOC ray segment length data, were unaffected, the overall memory reduction was roughly a factor of three. This is still very significant, particularly when paired with the run time reduction of close to a factor of 1.5.

Both spatial and angular decomposition were evaluated for the explicit and moment-based sweepers. Spatially, the solvers scaled very well, as communication is only necessary

between the neighboring planes. Angularly, the scalability was a bit poorer. This is mostly attributable to the reductions necessary for the angular fluxes at the axial surfaces, as they are used to construct the axial transverse leakage for the radial solver. Since even number of processors is generally used for angular decomposition and there are always an odd number of moment components, the partitioning is always imbalanced. Because of the memory duplication, angular decomposition is generally not used, and these results suggest that spatial decomposition should be exhausted before moving to angular decomposition.

With all results considered, several ideas can be concluded:

- The anisotropic behavior of the transverse leakages can be particularly important in certain cases, especially those with control rods. While assuming these are isotropic may be suitably accurate for some cases, including the polar and azimuthal dependence can provide significant improvements in accuracy.
- The Fourier moment approach reduces the memory and run time requirements compared to explicit representation. Though still more costly than isotropic or azimuthally-integrated transverse leakages, the accuracy gained for certain problems is worth the additional effort.
- The shape of the spatial distribution for the axial transverse leakage is a significant next step and will be necessary before pursuing more realistic problems, particularly those with heterogeneous pins.
- It is likely that applying this approach to a method similar to the "2D/1D Fusion" method where the axial solver does not use pin-homogenized domains but subpin level regions would also be worthwhile. Such methods are known to be particularly burdensome and the Fourier moment approach could alleviate this and not incur the same issues with the approximation to the axial transverse leakage shape.

6.2 Proposed Future Research Topics

6.2.1 Axial Transverse Leakage Spatial Distribution

As was shown in Chapter 5, there seems to be cancellation of error between the angular and spatial representation of the axial transverse leakage. This work explored improvements to the angular distribution, which in some cases led to higher errors, particularly those with heterogeneous fuel pins, where the flat spatial distribution is less accurate. The next step with this work would be to address the flat leakage approximation

being made and formulate more accurate shape functions to reduce this error. However, it is likely more complicated than just changing how much leakage is applied to each fine mesh region within each pin, as the spatial and angular dependence are coupled. With the current work, pin-averaged angular distributions were calculated. Quarter pin or smaller domains would possibly be necessary, unless the shape functions also described information on redistributing the angular dependence as well.

6.2.2 Spherical Harmonic Expansions for Angular Flux and Leakages

The work here focused on approximating the azimuthal dependence while maintaining explicit polar dependence. The results for the SP_N solver suggest that reasonable results can be obtained with a small number of moments in using a Legendre expansion for the polar dependence. In the discussions on anisotropic scattering, it was shown how the spherical harmonic functions match up well with the Fourier expansion. The next logical step for the angular dependence would be to formulate a solver based on the spherical harmonics functions, which yield simultaneous expansions for the polar and azimuthal angles. If successful, it could reduce the memory requirements of the angle-dependent variables even further. For example, if a second order spherical harmonics expansion is used, there would be 9 terms compared to the roughly 40 with a second order Fourier expansion with 8 polar angles in total. That being said, approximating the azimuthal dependence yields the most savings, but additional benefits could be gained by approximating both.

6.2.3 Transverse Leakage Splitting

Though transverse leakage splitting was not necessary to obtain results shown in this work, it is likely to be important for more complex problems, particularly when considering more realistic cross sections. Transverse leakage splitting has proven to be a important part in assuring convergence of cases with isotropic transverse leakages. When angular dependence is incorporated, it is even easier to have negative angular fluxes result. While some negativity can be accommodated, it is unphysical and could potentially cause problems.

The splitting techniques covered in this work use an isotropic splitting operator, in which the most negative source component (over all angles) is identified and used to adjust the total cross section. It is isotropic in the sense that all angles have the total cross section modified by the same factor. A more accurate representation of this would be to have an angle-dependent adjustment, though this would also require dividing by the angular flux instead of the scalar flux. In general, storing the angular fluxes is avoided at all costs as

this is particularly burdensome. It is possible, however, that an alternative method could be formulated that is less costly.

6.2.4 Subplane Method

In the 2D/1D framework presented in this work, it is assumed that there is only one axial node per plane. The subplane method has been proven to be very useful in mitigating the computational burden of 2D/1D, by allowing multiple axial nodes to exist for each radial MOC plane [6,19,20]. This allows for axial mesh refinement without requiring more MOC planes, which accounts for a majority of the computational time.

With the 2D/1D scheme in general, it is assumed that the angular distribution of the axial transverse leakage will be constant across the entire plane. In this work, axial mesh refinement means that each plane can have a different distribution. But with the subplane method, this approximation could become more severe as the thickness of the plane increases. It is not clear if this would necessarily cause accuracy problems, but it is something that should be investigated.

6.2.5 Angle-Dependent Total Cross Sections

In the discussion of the sources of error in 2D/1D (Section 3.6), that to be completely consistent, the axial S_N solver should use an angle-dependent total cross section. It is a common approach to homogenize this with the scalar flux, yielding an angle-independent value, because the angular fluxes are typically not stored, especially on the fine mesh basis. While it is possible that some of the error in the heterogeneous pin cases comes from this approximation, it is not suspected to be a major factor since cancellation of error claims seem supported even in the diffusion based solvers, where the total cross section homogenization is not a factor. Additionally, other work investigating the approximation suggests the effect is minimal [33, 63], though it may be worth pursuing once errors from other approximations have been resolved.

APPENDIX A

Supplemental Derivations

A.1 NEM

A.1.1 Interior Kernel

To derive the NEM kernel equations, the 1D-diffusion equation on the normal basis serves as the starting point.

$$-\Sigma_D \frac{d}{d\xi^2}\phi(\xi) + \Sigma_r \phi(\xi) - Q(\xi) = 0 \tag{A.1}$$

where

$$-\Sigma_D = \frac{4D}{h^2} \tag{A.2}$$

A quartic Legendre expansion is assumed for the flux (Eq. A.3a) and a quadratic expansion for the source (Eq. A.3b).

$$\phi(\xi) = a_0 P_0(\xi) + a_1 P_1(\xi) + a_2 P_2(\xi) + a_3 P_3(\xi) + a_4 P_4(\xi) \tag{A.3a}$$

$$Q(\xi) = q_0 P_0(\xi) + q_1 P_1(\xi) + q_2 P_2(\xi) \tag{A.3b}$$

The first step is to evaluate the 0^{th}, 1^{st}, and 2^{nd} moment balance equations, which can be obtained by multiplying Eq. A.1 by $P_n(\xi)$ and integrating over ξ.

A.1.1.1 0^{th} Moment Balance Equation

$$\int_{-1}^{1} P_0(\xi)\left(-\Sigma_D \frac{d}{d\xi^2}\phi(\xi) + \Sigma_r \phi(\xi) - Q(\xi)\right) d\xi = 0 \tag{A.4}$$

Substituting the Legendre expansions for the source and flux yields Eq. A.5.

$$\int_{-1}^{1} P_0(\xi)\Big(-\Sigma_D \frac{d}{d\xi^2}\Big(a_0P_0(\xi)+a_1P_1(\xi)+a_2P_2(\xi)+a_3P_3(\xi)+a_4P_4(\xi)\Big)+ \Sigma_r\Big(a_0P_0(\xi)+a_1P_1(\xi)+a_2P_2(\xi)+a_3P_3(\xi)+a_4P_4(\xi)\Big)- \Big(q_0P_0(\xi)+q_1P_1(\xi)+q_2P_2(\xi)\Big)\Big)d\xi = 0 \tag{A.5}$$

First, the second order derivative term can be evaluated:

$$\frac{d}{d\xi^2}\Big(a_0P_0(\xi)+a_1P_1(\xi)+a_2P_2(\xi)+a_3P_3(\xi)+a_4P_4(\xi)\Big) \tag{A.6}$$

For simplicity, the Legendre polynomials can easily be written out explicitly (Eq. A.7)

$$\frac{d}{d\xi^2}\left(a_0+a_1\xi+a_2\left(\frac{3}{2}\xi^2-\frac{1}{2}\right)+a_3\left(\frac{5}{2}\xi^3-\frac{3}{2}\xi\right)+a_4\left(\frac{35}{8}\xi^4-\frac{30}{8}\xi^2+\frac{3}{8}\xi\right)\right) \tag{A.7}$$

Evaluating the second derivative of each Legendre polynomial leaves the following solution:

$$3a_2+15\xi a_3+a_4\left(\frac{105}{2}\xi^2-\frac{30}{4}\right) \tag{A.8}$$

Substituting this back into Eq. A.5 gives:

$$\int_{-1}^{1} P_0(\xi)\Big(-\Sigma_D\Big(3a_2+15\xi a_3+a_4\Big(\frac{105}{2}\xi^2-\frac{30}{4}\Big)\Big) + \Sigma_r(a_0P_0(\xi)+a_1P_1(\xi)+a_2P_2(\xi)+a_3P_3(\xi)+a_4P_4(\xi)) - (q_0P_0(\xi)+q_1P_1(\xi)+q_2P_2(\xi))\Big)d\xi = 0 \tag{A.9}$$

Knowing that Legendre polynomials are orthonormal (i.e. $\int_{-1}^{1} P_m(x)P_n(x)dx = 0$ for m $\neq n$), it can be found that many of the terms integrate out of the equation.

$$\int_{-1}^{1} P_0(\xi)\Big(-\Sigma_D\Big(3a_2+15\xi a_3+a_4\Big(\frac{105}{2}\xi^2-\frac{30}{4}\Big)\Big)+\Sigma_r a_0P_0(\xi)-q_0P_0(\xi)\Big)d\xi = 0 \tag{A.10}$$

Evaluating the integral yields the final solution for the 0^{th} moment balance equation (Eq. A.11).

$$\boxed{-\Sigma_D\Big(3a_2+10a_4\Big)+\Sigma_r a_0 = q_0} \tag{A.11}$$

A.1.1.2 1^{st} Moment Balance Equation

A similar procedure can be carried out for the first moment balance equation:

$$\begin{aligned}\int_{-1}^{1} P_1(\xi)\Big(-\Sigma_D\Big(3a_2+15\xi a_3+a_4\Big(\frac{105}{2}\xi^2-\frac{30}{4}\Big)\Big)\\ +\Sigma_r(a_0P_0(\xi)+a_1P_1(\xi)+a_2P_2(\xi)+a_3P_3(\xi)+a_4P_4(\xi))\\ -(q_0P_0(\xi)+q_1P_1(\xi)+q_2P_2(\xi))\Big)d\xi=0\end{aligned} \tag{A.12}$$

Removing the terms that would integrate out:

$$\int_{-1}^{1} P_1(\xi)\Big(-\Sigma_D\Big(3a_2+15\xi a_3+a_4\Big(\frac{105}{2}\xi^2-\frac{30}{4}\Big)\Big)+\Sigma_r(a_1P_1(\xi))-(q_1P_1(\xi))\Big)d\xi=0 \tag{A.13}$$

And simplifying:

$$\int_{-1}^{1}\Big(-\Sigma_D\Big(3\xi a_2+15\xi^2a_3+a_4\Big(\frac{105}{2}\xi^3-\frac{30}{4}\xi\Big)\Big)+\Sigma_r(a_1\xi^2)-(q_1\xi^2)\Big)d\xi=0 \tag{A.14}$$

Evaluating the integrals yields the final first moment balance equation:

$$-10\Sigma_Da_3+\frac{2}{3}\Sigma_ra_1-\frac{2}{3}q_1=0 \tag{A.15}$$

$$\boxed{-15\Sigma_Da_3+\Sigma_ra_1-q_1=0} \tag{A.16}$$

A.1.1.3 2^{nd} Moment Balance Equation

Repeating for the second moment balance equation:

$$\begin{aligned}\int_{-1}^{1} P_2(\xi)\Big(-\Sigma_D\Big(3a_2+15\xi a_3+a_4\Big(\frac{105}{2}\xi^2-\frac{30}{4}\Big)\Big)\\ +\Sigma_r(a_0P_0(\xi)+a_1P_1(\xi)+a_2P_2(\xi)+a_3P_3(\xi)+a_4P_4(\xi))\\ -(q_0P_0(\xi)+q_1P_1(\xi)+q_2P_2(\xi))\Big)d\xi=0\end{aligned} \tag{A.17}$$

Eliminating appropriate terms:

$$\begin{aligned}\int_{-1}^{1} P_2(\xi)\Big(-\Sigma_D\Big(3a_2+15\xi a_3+a_4\Big(\frac{105}{2}\xi^2-\frac{30}{4}\Big)\Big)\\ +\Sigma_r(a_2P_2(\xi))-(q_2P_2(\xi))\Big)d\xi=0\end{aligned} \tag{A.18}$$

Simplifying:

$$\int_{-1}^{1}\Big(-\Sigma_D\Big(3a_2\Big(\frac{3}{2}\xi^2-\frac{1}{2}\Big)+15\xi a_3\Big(\frac{3}{2}\xi^2-\frac{1}{2}\Big)+a_4\Big(\frac{105}{2}\xi^2-\frac{30}{4}\Big)\Big(\frac{3}{2}\xi^2-\frac{1}{2}\Big)\Big) + \Sigma_r a_2\Big(\frac{3}{2}\xi^2-\frac{1}{2}\Big)^2 - q_2\Big(\frac{3}{2}\xi^2-\frac{1}{2}\Big)^2\Big)d\xi = 0 \tag{A.19}$$

Evaluating:

$$-14\Sigma_D a_4+\frac{2}{5}\Sigma_r a_2-\frac{2}{5}q_2=0 \tag{A.20}$$

$$-35\Sigma_D a_4+\Sigma_r a_2-q_2=0 \tag{A.21}$$

A.1.1.4 Interface Conditions

At the interface between the two nodes, it will be enforced that the scalar flux and current are continuous. In nodal core simulators, discontinuity factors are typically used to define the relation between these quantities at the interface, allowing them to be discontinuous based on values from the lattice physics code used to generated the cross sections.

Scalar Flux Continuity:

The scalar flux continuity can be enforced by ensuring that the scalar flux of the left node (ϕ_1) evaluated at the right-most boundary ($\xi = 1$) is equivalent to the scalar flux of the right node (ϕ_2) evaluated at the left-most boundary ($\xi = -1$), as in Eq. A.22.

$$\phi_1(1)=\phi_2(-1) \tag{A.22}$$

Substituting the quartic flux expansions yields:

$$a_{1,0}P_0(1)+a_{1,1}P_1(1)+a_{1,2}P_2(1)+a_{1,3}P_3(1)+a_{1,4}P_4(1) = a_{2,0}P_0(-1)+a_{2,1}P_1(-1)+a_{2,2}P_2(-1)+a_{2,3}P_3(-1)+a_{2,4}P_4(-1) \tag{A.23}$$

Since $P_n(1) = 1$ and $P_n(-1) = 1$ for even moments and $P_n(-1) = -1$ for odd moments, this above equation can be simplified.

$$a_{1,0}+a_{1,1}+a_{1,2}+a_{1,3}+a_{1,4}=a_{2,0}-a_{2,1}+a_{2,2}-a_{2,3}+a_{2,4} \tag{A.24}$$

As will be made clear when the details of the linear system are given, it will be useful to move the zeroth moment values to the right hand side:

$$\boxed{a_{1,1}+a_{1,2}+a_{1,3}+a_{1,4}+a_{2,1}-a_{2,2}+a_{2,3}-a_{2,4}=-a_{1,0}+a_{2,0}} \tag{A.25}$$

Current Continuity:

Similarly, the current at the interface can be enforced:

$$J_1(1) = J_2(-1) \tag{A.26}$$

The current can be found by using Fick's Law (see Section 2.1.6), evaluating for the derivative in the scalar flux at the interface.

$$-\frac{2D_1}{h}\frac{d\phi_1(\xi)}{d\xi}\bigg|_{x=1} = -\frac{2D_2}{h}\frac{d\phi_2(\xi)}{d\xi}\bigg|_{\xi=-1} \tag{A.27}$$

Substituting in the flux expansion and Legendre polynomials yields Eqs. A.28 and A.29

$$\begin{aligned}-\frac{2D_1}{h_1}\frac{d}{d\xi}\Big(a_{1,0}P_0(\xi)+a_{1,1}P_1(\xi)+a_{1,2}P_2(\xi)+a_{1,3}P_3(\xi)+a_{1,4}P_4(\xi)\Big)\bigg|_{\xi=1} = \\ -\frac{2D_2}{h_2}\frac{d}{d\xi}\Big(a_{2,0}P_0(\xi)+a_{2,1}P_1(\xi)+a_{2,2}P_2(\xi)+a_{2,3}P_3(\xi)+a_{2,4}P_4(\xi)\Big)\bigg|_{\xi=-1}\end{aligned} \tag{A.28}$$

$$\begin{aligned}-\frac{2D_1}{h_1}\Big(a_{1,1}+a_{1,2}(3\xi)+a_{1,3}(\frac{1}{2}(15\xi^2-3)+a_{1,4}(\frac{1}{8}(140\xi^3-60\xi))\Big)\bigg|_{\xi=1} = \\ -\frac{2D_2}{h_2}\Big(a_{2,1}+a_{2,2}(3\xi)+a_{2,3}(\frac{1}{2}(15\xi^2-3)+a_{2,4}(\frac{1}{8}(140\xi^3-60\xi))\Big)\bigg|_{\xi=-1}\end{aligned} \tag{A.29}$$

Evaluating at $\xi = 1$ yields the final equation:

$$-\frac{2D_1}{h_1}\Big(a_{1,1}+3a_{1,2}+6a_{1,3}+10a_{1,4}\Big) = -\frac{2D_2}{h_2}\Big(a_{2,1}-3a_{2,2}+6a_{2,3}-10a_{2,4}\Big) \tag{A.30}$$

$$\boxed{-\frac{2D_1}{h_1}\Big(a_{1,1}+3a_{1,2}+6a_{1,3}+10a_{1,4}\Big)+\frac{2D_2}{h_2}\Big(a_{2,1}-3a_{2,2}+6a_{2,3}-10a_{2,4}\Big)=0} \tag{A.31}$$

A.1.1.5 Node-Averaged Flux Preservation

The node-averaged flux can be preserved by forcing the zeroth moment value in the flux expansion to be the scalar flux, as in Eq. A.32.

$$a_{1,0} = \overline{\phi}_1 \tag{A.32a}$$

$$a_{2,0} = \overline{\phi}_2 \tag{A.32b}$$

A.1.1.6 Linear System Structure

With the 0^{th} through 2^{nd} moment equations, interface conditions, and flux preservation, an 8x8 linear system can be constructed to solve for each of the higher order spatial moments of the flux in two nodes.

$$\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 \\
-\frac{2D_1}{h_1} & -3\frac{2D_1}{h_1} & -6\frac{2D_1}{h_1} & -10\frac{2D_1}{h_1} & \frac{2D_2}{h_2} & -3\frac{2D_2}{h_2} & 6\frac{2D_2}{h_2} & -10\frac{2D_2}{h_2} \\
 & -3\Sigma_{D1} & & -10\Sigma_{D1} & & & & \\
\Sigma_{r1} & & -15\Sigma_{D1} & & & & & \\
 & \Sigma_{r1} & & -35\Sigma_{D1} & & & & \\
 & & & & & -3\Sigma_{D2} & & -10\Sigma_{D2} \\
 & & & & \Sigma_{r2} & & -15\Sigma_{D2} & \\
 & & & & & \Sigma_{r2} & & -35\Sigma_{D2}
\end{bmatrix}
\begin{bmatrix} a_{1,1} \\ a_{1,2} \\ a_{1,3} \\ a_{1,4} \\ a_{2,1} \\ a_{2,2} \\ a_{2,3} \\ a_{2,4} \end{bmatrix}
=
\begin{bmatrix} -\overline{\phi}_1 + \overline{\phi}_2 \\ 0 \\ q_{1,0} - \Sigma_{r1}\overline{\phi}_1 \\ q_{1,1} \\ q_{1,2} \\ q_{2,0} - \Sigma_{r2}\overline{\phi}_2 \\ q_{2,1} \\ q_{2,2} \end{bmatrix} \tag{A.33}$$

A.1.2 Boundary Kernel

To derive the system for the boundary kernel, the balance equations for the 0^{th}, 1^{st}, and 2^{nd} moments, which are repeated in Eq. A.34, can be reused.

$$\Sigma_D(-3a_2 - 10a_4) + \Sigma_r a_0 = q_0 \tag{A.34a}$$

$$-15\Sigma_D a_3 + \Sigma_r a_1 = q_1 \tag{A.34b}$$

$$-35a_4\Sigma_D + \Sigma_r a_2 = q_2 \tag{A.34c}$$

A.1.2.1 Interface Current

For the boundary kernel, the boundary condition α is used to define the relationship between the surface flux and current ($\alpha = \frac{J_s}{\phi_s}$).

$$J_1(1) = \alpha\phi_(1) \tag{A.35}$$

$$-\frac{D}{h}\frac{d\phi(\xi)}{d\xi}\bigg|_{\xi=1} = \alpha\phi(1) \tag{A.36}$$

$$-\frac{D}{h}\frac{d}{d\xi}\left(a_0 + a_1P_1(\xi) + a_2P_2(\xi) + a_3P_3(\xi) + a_4P_4(\xi)\right)\Big|_{\xi=1} \tag{A.37}$$
$$= \alpha\left(a_0 + a_1P_1(1) + a_2P_2(1) + a_3P_3(1) + a_4P_4(1)\right)$$

$$-\frac{D}{h}\left(a_1 + 3a_2 + 6a_3 + 10a_4\right) = \alpha\left(a_0 + a_1 + a_2 + a_3 + a_4\right) \tag{A.38}$$

$$\left(\frac{D}{h}+\alpha\right)a_1 + \left(3\frac{D}{h}+\alpha\right)a_2 + \left(6\frac{D}{h}+\alpha\right)a_3 + \left(10\frac{D}{h}+\alpha\right)a_4 = -\alpha a_0 \tag{A.39}$$

A.1.2.2 Node-Averaged Flux Preservation

Again, the node-averaged scalar flux is preserved:

$$a_0 = \overline{\phi} \tag{A.40}$$

A.1.2.3 Linear System Structure

Because only one node is involved with the boundary kernel, the linear system size is reduced from an 8x8 to a 4x4.

$$\begin{bmatrix} \left(\frac{D}{h}+\alpha\right) & \left(3\frac{D}{h}+\alpha\right) & \left(6\frac{D}{h}+\alpha\right) & \left(10\frac{D}{h}+\alpha\right) \\ & -3\Sigma_D & & -10\Sigma_D \\ \Sigma_r & & -15\Sigma_D & \\ & \Sigma_r & & -35\Sigma_D \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} -\overline{\phi}_1 + \overline{\phi}_2 \\ q_0 - \Sigma_{r1}\overline{\phi}_1 \\ q_1 \\ q_2 \end{bmatrix} \tag{A.41}$$

A.2 SENM

A.2.1 Interior Kernel

With the source expansion nodal method (SENM), the flux and source correspond to the following equations:

$$\phi(\xi) = Asinh(\kappa\xi) + Bcosh(\kappa\xi) + \sum_{i=0}^{4} a_i P_i(\xi) \tag{A.42a}$$

$$Q(\xi) = \sum_{i=0}^{4} q_i P_i(\xi) \tag{A.42b}$$

where

$$\kappa = \frac{h}{2}\sqrt{\frac{\Sigma_r}{D}}$$

The particular solution coefficients (a_i) can be found by making five moment balance equations and the hyperbolic coefficients can be found by enforcing continuity of the flux and current at the interface between the two-nodes and by enforcing preservation of the node-averaged scalar flux. First, the particular coefficient will be derived.

A.2.1.1 0^{th} Moment Balance Equation

As with the moment balance equations used to derive the NEM equations, the moment balance equations here multiply the 1D diffusion equation (on a normalized basis) by an n^{th} order Legendre polynomial, integrating over the entire node:

$$\int_{-1}^{1} P_0(\xi)\left(-\Sigma_D \frac{d}{d\xi^2}\phi_(\xi) + \Sigma_r\phi(\xi) - Q(\xi)\right)d\xi = 0 \tag{A.43}$$

where $\Sigma_D = \frac{4D}{h^2} = \frac{\Sigma_r}{\kappa^2}$

Substituting the flux expansion (Eq. A.42a) yields:

$$\int_{-1}^{1} P_0(\xi)\Bigg(-\Sigma_D \frac{d}{d\xi^2}(Asinh(\kappa\xi) + Bcosh(\kappa\xi) + \sum_{i=0}^{4} a_i P_i(\xi))$$
$$+\Sigma_r(Asinh(\kappa\xi) + Bcosh(\kappa\xi) + \sum_{i=0}^{4} a_i P_i(\xi)) - \sum_{i=0}^{4} q_i P_i(\xi)\Bigg)d\xi = 0$$

Rearranging to pull the *sinh* and *cosh* terms into one integral:

$$\int_{-1}^{1} P_0(\xi)\left(-\Sigma_D \frac{d}{d\xi^2}(A sinh(\kappa\xi) + B cosh(\kappa\xi) + \Sigma_r(A sinh(\kappa\xi) + B cosh(\kappa\xi)))\right)d\xi + \int_{-1}^{1}\left(-\Sigma_D \frac{d}{d\xi^2}\sum_{i=0}^{4} a_i P_i(\xi) + \Sigma_r \sum_{i=0}^{4} a_i P_i(\xi) - \sum_{i=0}^{4} q_i P_i(\xi)\right)d\xi = 0 \tag{A.44}$$

It can be found that the first integral will evaluate to zero by substituting in the previously found expression for Σ_D in terms of Σ_r and κ:

$$\int_{-1}^{1} P_0(\xi)\left(-\frac{\Sigma_r}{\kappa^2}(A\kappa^2 sinh(\kappa\xi) + B\kappa^2 cosh(\kappa\xi) + \Sigma_r(A sinh(\kappa\xi) + B cosh(\kappa\xi)))\right)d\xi$$

This is true for all of the moment equations, so it will be neglected for the rest of the derivation.

Returning to Eq. A.44:

$$\int_{-1}^{1} P_0(\xi)\left(-\Sigma_D \frac{d}{d\xi^2}\sum_{i=0}^{4} a_i P_i(\xi) + \Sigma_r \sum_{i=0}^{4} a_i P_i(\xi) - \sum_{i=0}^{4} q_i P_i(\xi))\right)d\xi = 0$$

The next step is to evaluate the second derivative:

$$\frac{d}{d\xi^2}\sum_{i=0}^{4} a_i P_i(\xi)$$

$$= \frac{d}{d\xi^2}\left(a_0 + a_1\xi + a_2(\frac{3}{2}\xi^2\frac{1}{2}) + a_3(\frac{5}{2}\xi^3 - \frac{3}{2}\xi) + a_4(\frac{35}{8}\xi^4 - \frac{30}{8}\xi^2 + \frac{3}{8})\right)$$

$$= 3a_2 + 15\xi a_3 + a_4\left(\frac{105}{2}\xi^2 - \frac{15}{2}\right)$$

Substituting this back in yields:

$$\int_{-1}^{1} P_0(\xi)\left(-\Sigma_D\left(3a_2 + 15\xi a_3 + a_4\left(\frac{105}{2}\xi^2 - \frac{15}{2}\right)\right) + \Sigma_r \sum_{i=0}^{4} a_i P_i(\xi) - \sum_{i=0}^{4} q_i P_i(\xi))\right)d\xi = 0$$

Because of orthogonality, only one term in each the Σ_r and Q terms will remain:

$$+ \int_{-1}^{1} P_0(\xi)\left(-\Sigma_D\left(3a_2 + 15\xi a_3 + a_4\left(\frac{105}{2}\xi^2 - \frac{15}{2}\right)\right) + \Sigma_r a_0 P_0(\xi) - q_0 P_0(\xi))\right)d\xi = 0$$

Evaluating the integral yields the following:

$$\Sigma_D(-3a_2 - 10a_4) + \Sigma_r a_0 = q_0$$

Rearranging and substituting for Σ_D:

$$\boxed{a_0 = \frac{1}{\Sigma_r}\left(q_0 + \frac{\Sigma_r}{\kappa}(3a_2 + 10a_4)\right)} \tag{A.45}$$

A.2.1.2 1^{st} Moment Balance Equation

Skipping to after the second derivative is evaluated:

$$\int_{-1}^{1} P_1(\xi)\left(-\Sigma_D\left(3a_2 + 15\xi a_3 + a_4\left(\frac{105}{2}\xi^2 - \frac{15}{2}\right)\right) + \Sigma_r \sum_{i=0}^{4} a_i P_i(\xi) - \sum_{i=0}^{4} q_i P_i(\xi)\right) d\xi = 0 \tag{A.46}$$

Because of orthogonality, only one term in each the Σ_r and Q terms will remain:

$$\int_{-1}^{1} P_1(\xi)\left(-\Sigma_D\left(3a_2 + 15\xi a_3 + a_4\left(\frac{105}{2}\xi^2 - \frac{15}{2}\right)\right) + \Sigma_r a_1 P_1(\xi) - q_1 P_1(\xi)\right) d\xi = 0$$

Evaluating the integral yields the following:

$$-\Sigma_D(15a_3) + \Sigma_r(a_1) = q_1$$

Rearranging and substituting for Σ_D:

$$\boxed{a_1 = \frac{1}{\Sigma_r}\left(q_1 + \frac{\Sigma_r}{\kappa^2}(15a_3)\right)} \tag{A.47}$$

A.2.1.3 2^{nd} Moment Balance Equation

Skipping to after the second derivative is evaluated

$$\int_{-1}^{1} P_2(\xi)\left(-\Sigma_D\left(3a_2 + 15\xi a_3 + a_4\left(\frac{105}{2}\xi^2 - \frac{15}{2}\right)\right) + \Sigma_r \sum_{i=0}^{4} a_i P_i(\xi) - \sum_{i=0}^{4} q_i P_i(\xi)\right) d\xi = 0 \tag{A.48}$$

Because of orthogonality, only one term in each the Σ_r and Q terms will remain:

$$\int_{-1}^{1} P_2(\xi)\left(-\Sigma_D\left(3a_2 + 15\xi a_3 + a_4\left(\frac{105}{2}\xi^2 - \frac{15}{2}\right)\right) + \Sigma_r a_2 P_2(\xi) - q_2 P_2(\xi)\right) d\xi = 0$$

Evaluating the integral yields the following:

$$-\Sigma_D(35a_4) + \Sigma_r(a_2) = q_2$$

Rearranging and substituting for Σ_D:

$$\boxed{a_2 = \frac{1}{\Sigma_r}\left(q_2 + \frac{\Sigma_r}{\kappa^2}(35a_4)\right)} \tag{A.49}$$

A.2.1.4 3^{rd} Moment Balance Equation

Skipping to after the second derivative is evaluated

$$\int_{-1}^{1} P_3(\xi)\left(-\Sigma_D\left(3a_2 + 15\xi a_3 + a_4\left(\frac{105}{2}\xi^2 - \frac{15}{2}\right)\right) + \Sigma_r \sum_{i=0}^{4} a_i P_i(\xi) - \sum_{i=0}^{4} q_i P_i(\xi)\right) d\xi = 0 \tag{A.50}$$

Because of orthogonality, only one term in each the Σ_r and Q terms will remain:

$$\int_{-1}^{1} P_3(\xi)\left(-\Sigma_D\left(3a_2 + 15\xi a_3 + a_4\left(\frac{105}{2}\xi^2 - \frac{15}{2}\right)\right) + \Sigma_r a_3 P_3(\xi) - q_3 P_3(\xi)\right) d\xi = 0$$

Evaluating the integral yields the following:

$$\Sigma_r(a_3) = q_3$$

Rearranging and substituting for Σ_D:

$$\boxed{a_3 = \frac{q_3}{\Sigma_r}} \tag{A.51}$$

A.2.1.5 4^{th} Moment Balance Equation

Skipping to after the second derivative is evaluated

$$\int_{-1}^{1} P_4(\xi)\left(-\Sigma_D\left(3a_2+15\xi a_3+a_4\left(\frac{105}{2}\xi^2-\frac{15}{2}\right)\right)+\Sigma_r\sum_{i=0}^{4}a_iP_i(\xi)-\sum_{i=0}^{4}q_iP_i(\xi)\right)d\xi=0 \tag{A.52}$$

Because of orthogonality, only one term in each the Σ_r and Q terms will remain:

$$\int_{-1}^{1} P_4(\xi)\left(-\Sigma_D\left(3a_2+15\xi a_3+a_4\left(\frac{105}{2}\xi^2-\frac{15}{2}\right)\right)+\Sigma_r a_4P_4(\xi)-q_4P_4(\xi)\right)d\xi=0 \tag{A.53}$$

Evaluating the integral yields the following:

$$\Sigma_r(a_4)=q_4 \tag{A.54}$$

Rearranging and substituting for Σ_D:

$$\boxed{a_4=\frac{q_4}{\Sigma_r}} \tag{A.55}$$

A.2.1.6 Summary

Substituting these back into one another as necessary yields:

$$a_0=\frac{1}{\Sigma_r}\left(q_0+\frac{1}{\kappa^2}(3q_2+10q_4)+\frac{1}{\kappa^2}105q_4\right) \tag{A.56a}$$

$$a_1=\frac{1}{\Sigma_r}\left(q_1+\frac{15q_3}{\kappa^2}\right) \tag{A.56b}$$

$$a_2=\frac{1}{\Sigma_r}\left(q_2+\frac{35q_4}{\kappa^2}\right) \tag{A.56c}$$

$$a_3=\frac{q_3}{\Sigma_r} \tag{A.56d}$$

$$a_4=\frac{q_4}{\Sigma_r} \tag{A.56e}$$

A.2.1.7 Interface Conditions

The flux and current continuity will be enforced to solve for the sinh coefficients of the flux expansion.

Flux Continuity:

$$\phi_1(1) = \phi_2(-1)$$

$$A_1 sinh(\kappa_1) + B_1 cosh(\kappa_1) + \sum_{i=0}^{4} a_{1,i} P_i(1) = A_2 sinh(-\kappa_2) + B_2 cosh(-\kappa) + \sum_{i=0}^{4} a_{2,i} P_i(-1)$$

$$A_1 sinh(\kappa_1) + B_1 cosh(\kappa_1) + a_{1,0} + a_{1,1} + a_{1,2} + a_{1,3} + a_{1,4} =$$
$$A_2 sinh(-\kappa_2) + B_2 cosh(-\kappa) + a_{2,0} - a_{2,1} + a_{2,2} - a_{2,3} + a_{2,4}$$

$$A_1 sinh(\kappa_1) + A_2 sinh(\kappa_2) = B_2 cosh(-\kappa) - B_1 cosh(\kappa_1) + \sum_{i=0}^{4} \left((-1)^i c_{2,i} - c_{1,i} \right)$$

Solving for A_2 yields:

$$A_2 = -A_1 \frac{sinh(\kappa_1)}{sinh(\kappa_2)} B_2 \frac{cosh(-\kappa)}{sinh(\kappa_2)} - B_1 \frac{cosh(\kappa_1)}{sinh(\kappa_2)} + \frac{1}{sinh(\kappa_2)} \sum_{i=0}^{4} \left((-1)^i c_{2,i} - c_{1,i} \right) \qquad \text{(A.57)}$$

Current Continuity:

$$J_1(1) = J_2(-1)$$
$$-\frac{2D_1}{h_1} \frac{d\phi_1(\xi)}{d\xi}\bigg|_{\xi=1} = -\frac{2D_2}{h_2} \frac{d\phi_2(\xi)}{d\xi}\bigg|_{\xi=-1}$$

For simplicity, $\beta_1 = \frac{2D_1}{h_1}$ and $\beta_2 = \frac{2D_2}{h_2}$ are defined.

$$-\beta_1 \frac{d}{d\xi} A_1 sinh(\kappa_1 \xi) + B_1 cosh(\kappa_1 \xi) + a_{1,0} P_0(\xi) + a_{1,1} P_1(\xi) + a_{1,2} P_2(\xi) + a_{1,3} P_3(\xi) + a_{1,4} P_4(\xi)\Big|_{\xi=1} =$$
$$-\beta_2 \frac{d}{d\xi} A_2 sinh(\kappa_2 \xi) + B_2 cosh(\kappa_2 \xi) + a_{2,0} P_0(\xi) + a_{2,1} P_1(\xi) + a_{2,2} P_2(\xi) + a_{2,3} P_3(\xi) + a_{2,4} P_4(\xi)\Big|_{\xi=-1}$$

Evaluating at $\xi = 1$ and $\xi = -1$ gives:

$$-\beta_1 A_1 \kappa_1 sinh(\kappa_1) + B_1 \kappa_1 cosh(\kappa_1) + a_{1,1} + 3a_{1,2} + 6a_{1,3} + 10a_{1,4} =$$
$$-\beta_2 A_2 \kappa_2 sinh(\kappa_2) + B_2 \kappa_2 cosh(\kappa_2) + a_{2,1} - 3a_{2,2} + 6a_{2,3} + 10a_{2,4}$$

Solving for A_1 yields:

$$\begin{aligned} A_1 =& A_2 \frac{\beta_2}{\beta_1} \frac{\kappa_2 cosh(\kappa_2)}{\kappa_1 cosh(\kappa_1)} \\ &- \frac{1}{\kappa_1 cosh(\kappa_1)} \frac{\beta_2}{\beta_1} \frac{\beta_2}{\beta_1} (B_2 \kappa_2 \sinh(\kappa_2) - a_{2,1} + 3a_{2,2} - 6a_{2,3} + 10a_{2,4}) \\ &- \frac{1}{\kappa_1 cosh(\kappa_1)} (B_1 \kappa_1 sinh(\kappa_1) + a_{1,1} + 3a_{1,2} + 6a_{1,3} + 10a_{1,4})) \end{aligned} \tag{A.58}$$

It is important to note that both A.57 and A.58 are both written in terms of A_1 and A_2. Combining these two equations, a new expression for A_1 can be found that has no dependence on A_2:

$$\begin{aligned} & A_1 \left(1 + \frac{\beta_2 \kappa_2 cosh(\kappa_2) sinh(\kappa_2)}{\beta_1 \kappa_1 cosh(\kappa_1) sinh(\kappa_1)}\right) \\ &= \frac{1}{sinh(\kappa_2)} \frac{\beta_2 \kappa_2 cosh(\kappa_2) sinh(\kappa_2)}{\beta_1 \kappa_1 cosh(\kappa_1) sinh(\kappa_1)} \left(B_2 cosh(\kappa_2) - B_1 cosh(\kappa_1) + \sum_{i=0}^{4} \left((-1)^i a_(2,i) - a(1,i)\right)\right) \\ &- \frac{1}{\kappa_1 \cosh(\kappa_1)} \frac{\beta_2}{\beta_1} (B_2 \kappa_2 sinh(\kappa_2) - a_{2,1} + 3a_{2,2} - 6a_{2,3} + 10a_{2,4}) \\ &- \frac{1}{\kappa_1 cosh(\kappa_1)} (B_1 \kappa_1 sinh(\kappa_1) + a_{1,1} + 3a_{1,2} + 6a_{1,3} + 10a_{1,4}) \end{aligned} \tag{A.59}$$

A.2.1.8 Node-Averaged Flux Preservaton

The node-averaged flux preservation can be used to determine the remaining coefficients (B), by averaging the flux expansion over ξ, which is known to be the node-averaged scalar flux ($\overline{\phi}$).

$$\begin{aligned} \frac{1}{2} \int_{-1}^{1} \phi(\xi) d\xi &= \overline{\phi} \\ = \int_{-1}^{1} \left(A sinh(\kappa\xi) + B cosh(\kappa\xi) + \sum_{i=0}^{4} a_0(\xi)\right) d\xi & \\ = B \frac{cosh(\kappa\xi)}{\kappa} + a_0 & \end{aligned} \tag{A.60}$$

Solving for B:

$$\boxed{B = \frac{\kappa(\overline{\phi} - a_0)}{sinh(\kappa)}} \tag{A.61}$$

A.2.1.9 Calculation Sequence

Unlike with the NEM formulation, the SENM solver does not setup up a linear system to determine the unknown flux coefficients. Instead, the particular coefficients are solved first since they only dependent on the source moments (q_i). Then, the *cosh* coefficients (B) are found and both a_i and are used to determine the *sinh* coefficients (A). It is also noteworthy to point out that the flux expansion is projected down onto a quartic Legendre expansion (without hyperbolic coefficients) when formulating the source moment values.

A.2.2 Boundary Kernel

To formulate the boundary kernel equations, only the interface current continuity equation needs to be revisited. That is, the particular solution coefficients (a_i) and cosh coefficients (B) are the same (and repeated below).
Substituting these back into one another as necessary yields:

$$a_0 = \frac{1}{\Sigma_r}\left(q_0 + \frac{1}{\kappa^2}(3q_2 + 10q_4) + \frac{1}{\kappa^2}105q_4\right) \tag{A.62a}$$

$$a_1 = \frac{1}{\Sigma_r}\left(q_1 + \frac{15q_3}{\kappa^2}\right) \tag{A.62b}$$

$$a_2 = \frac{1}{\Sigma_r}\left(q_2 + \frac{35q_4}{\kappa^2}\right) \tag{A.62c}$$

$$a_3 = \frac{q_3}{\Sigma_r} \tag{A.62d}$$

$$a_4 = \frac{q_4}{\Sigma_r} \tag{A.62e}$$

$$B = \frac{\kappa(\overline{\phi} - a_0)}{sinh(\kappa)} \tag{A.63}$$

A.2.2.1 Interface Current Continuity

$$J_1(1) = \alpha\phi_s$$
$$-\frac{2D}{h}\frac{d\phi(\xi)}{d\xi}\bigg|_{\xi=1} = \alpha\phi(x)\Big|_{\xi=1}$$

For simplicity, $\beta = \frac{2D_1}{h_1}$ is defined:

$$-\beta(A\kappa cosh(\kappa) + B\kappa sinh(\kappa) + a_1 + 3a_2 + 6a_3 + 10a_4 =$$
$$\alpha(Asinh(\kappa) + Bsinh(\kappa)) + a_0 + a_1 + a_2 + a_3 + a_4)$$

$$\boxed{A = -\frac{B(\beta\kappa sinh(\kappa) + \alpha cosh(\kappa)) + \alpha a_0 + (\beta + \alpha)a_1) + (3\beta + \alpha)a_2 + (6\beta + \alpha)a_3 + (10\beta + \alpha)a_4}{\alpha sinh(\kappa) + \beta\kappa cosh(\kappa)}} \quad \text{(A.64)}$$

A.3 SP_N

In this chapter, the SP_5 equations will be derived, from these, the SP_1 and SP_3 equations can be inferred, but similar techniques with additional moments would be necessary for higher order SP_n approximations.
The basis behind these methods is that the polar dependence of the angular flux is assumed to be represented by an N^{th} order Legendre expansion (Eq. A.65).

$$\varphi_g(x,\mu) = \sum_{n=0}^{N} \frac{2n+1}{2} \varphi_{n,g}(x) P_n(\mu) \tag{A.65}$$

A.3.1 Intranodal Equations

The equations for each moment (Eq. A.66) can be found my substituting the flux expansion into the 1D transport equation, multiplying by $P_n(\mu)$, and integrating over μ.

$$\frac{d}{dx}\left[\frac{n}{2n+1}\varphi_{n-1}(x) + \frac{n+1}{2n+1}\varphi_{n+1}(x)\right] + \Sigma_t \varphi_n(x) = \Sigma_{sn}\varphi_n(x) + Q(x)\delta_{n,0} \tag{A.66}$$

Using Eq. A.66, an equation for each moment can be formulated, as below:

n=0:

$$\frac{d}{dx}[\varphi_1(x)] + \Sigma_t \varphi_0(x) = \Sigma_{s0}\varphi_0(x) + Q(x)$$

$$\frac{d}{dx}\varphi_1(x) = Q(x) - (\Sigma_t - \Sigma_{s0})\varphi_0(x) \tag{A.67}$$

n=1:

$$\frac{d}{dx}\left[\frac{1}{3}\varphi_0(x) + \frac{2}{3}\varphi_2\right] + \Sigma_t \varphi_1(x) = \Sigma_{s1}\varphi_1(x)$$

$$\varphi_1(x) = -\frac{1}{3(\Sigma_t - \Sigma_{s1})}\frac{d}{dx}[\varphi_0(x) + 2\varphi_2(x)] = -D_0\frac{d}{dx}[\varphi_0(x) + 2\varphi_2(x)] \tag{A.68}$$

where

$$D_0 = \frac{1}{3(\Sigma_t - \Sigma_{s1})} \tag{A.69}$$

n=2:

$$\frac{d}{dx}\left[\frac{2}{5}\varphi_1(x) + \frac{3}{5}\varphi_3(x) = \Sigma_{s2}\varphi_2(x)\right]$$

$$\frac{d}{dx}\left[\frac{3}{5}\varphi_3(x)\right] + \Sigma_t\varphi_2(x) = \Sigma_{s2}\varphi_2(x) - \frac{d}{dx}\left[\frac{2}{5}\varphi_1(x)\right]$$

Substituting in from Eq. A.67:

$$\frac{d}{dx}\left[\frac{3}{5}\varphi_3(x)\right]+\Sigma_t\varphi_2(x)=\Sigma_{s2}\varphi_2(x)-\frac{2}{5}(Q(x)-(\Sigma_t-\Sigma_{s0})\varphi_0(x)) \tag{A.70}$$

$$\frac{d}{dx}\left[\varphi_3(x)\right]=-\frac{5}{3}(\Sigma_t-\Sigma_{s2})\varphi_2(x)-\frac{2}{3}(Q(x)-(\Sigma_t-\Sigma_{s0})\varphi_0(x))$$

n=3:

$$\frac{d}{dx}\left[\frac{3}{7}\varphi_2(x)-\frac{4}{7}\varphi_4(x)\right]+\Sigma_t\varphi_3(x)=\Sigma_{s3}(x)\varphi_3(x)$$

$$\varphi_3(x)=-\frac{1}{7(\Sigma_t-\Sigma_{s3})}\frac{d}{dx}[3\varphi_2(x)+4\varphi_4(x)]=-\frac{5}{9}D_2\frac{d}{dx}[3\varphi_2(x)+4\varphi_4(x)] \tag{A.71}$$

where

$$D_2=\frac{9}{35(\Sigma_t-\Sigma_{s3})} \tag{A.72}$$

n=4:

$$\frac{d}{dx}\left[\frac{4}{9}\varphi_3(x)+\frac{5}{9}\varphi_5(x)\right]+\Sigma_t\varphi_4(x)=\Sigma_{s4}\varphi_4(x)$$

and using Eq. A.70:

$$\frac{5}{9}\frac{d}{dx}[\varphi_5(x)]+(\Sigma_t-\Sigma_{s4})\varphi_4=\frac{4}{9}\left[\frac{5}{3}(\Sigma_t-\Sigma_{s2})\varphi_2(x)+\frac{2}{3}(Q(x)-(\Sigma_t-\Sigma_{s0})\varphi_0(x))\right] \tag{A.73}$$

n=5:

$$\frac{d}{dx}\left[\frac{5}{11}\varphi_4(x)\right]+\Sigma_t\varphi_5(x)=\Sigma_{s5}\varphi_5(x)$$

$$\varphi_5(x)=-\frac{5}{11(\Sigma_t-\Sigma_{s5})}\frac{d}{dx}[\varphi_4(x)]=-\frac{9}{5}D_4\frac{d}{dx}[\varphi_4(x)] \tag{A.74}$$

where

$$D_4=\frac{25}{99(\Sigma_t-\Sigma_{s5})} \tag{A.75}$$

To continue from here, the following definitions will be made:

$$\begin{aligned}\Phi_0(x)&=\varphi_0(x)+2\varphi_2(x)\\ \Phi_2(x)&=\varphi_2(x)+\frac{4}{3}\varphi_4(x)\\ \Phi_4(x)&=\varphi_4(x)\end{aligned} \tag{A.76}$$

Solving the definitions for $\varphi_0(x)$ and $\varphi_2(x)$ yields:

$$\begin{aligned}\varphi_0(x) &= \Phi_0(x) - 2\varphi_2(x)\\ \varphi_2(x) &= \Phi_2(x) - \frac{4}{3}\varphi_4 = \Phi_2(x) - \frac{4}{3}\Phi_4(x)\\ \varphi_0(x) &= \Phi_0(x) - 2\Phi_2(x) + \frac{8}{3}\Phi_4(x)\end{aligned} \tag{A.77}$$

Substituting these into the final equation for each moment yields Eq. A.78. Eqs. A.78b, A.78d, and A.78f have been substituted into A.78a, A.78c, and A.78e, respectively.

n=0:

$$-D_0\frac{d^2}{dx^2}\Phi_0 + (\Sigma_t - \Sigma_{s0})\Phi_0(x) = Q(x) + 2(\Sigma_t - \Sigma_{s0})\left(\Phi_2(x) - \frac{4}{3}\Phi_4(x)\right) \tag{A.78a}$$

n=1:

$$\varphi_1 = -D_0\frac{d}{dx}\Phi_0(x) \tag{A.78b}$$

n=2:

$$\begin{aligned}&-D_2\frac{d^2}{dx^2}[\Phi_2(x)] + \left(\frac{9}{5}\Sigma_t - \frac{4}{5}\Sigma_{s0} - \Sigma_{s2}\right)\Phi_2(x)\\ &= -\frac{2}{5}\left(Q(x) - (\Sigma_t - \Sigma_{s0})\Phi_0(x)\right) + \left(\frac{28}{15}\Sigma_t - \frac{16}{15}\Sigma_{s0} - \frac{4}{5}\Sigma_{s2}\right)\Phi_4\end{aligned} \tag{A.78c}$$

n=3:

$$\varphi_3 = -\frac{5}{9}D_2\frac{d}{dx}[3\Phi_2(x)] \tag{A.78d}$$

n=4:

$$\begin{aligned}&-D_4\frac{d^2}{dx^2}[\Phi_4(x)] + \left(\frac{225}{81}\Sigma_t + \frac{80}{81}\Sigma_{s2} - \frac{64}{81}\Sigma_{s0} - \Sigma_{s4}\right)\Phi_4(x) =\\ &\left[\frac{8}{27}(Q(x) - (\Sigma_t - \Sigma_{s0})\Phi_0(x)) + \left(\frac{36}{27}\Sigma_t - \frac{16}{27}\Sigma_{s0} - \frac{20}{27}\Sigma_{s2}\right)\Phi_2(x)\right]\end{aligned} \tag{A.78e}$$

n=5:

$$\varphi_5(x) = -\frac{9}{5}D_4\frac{d}{dx}\Phi_4(x) \tag{A.78f}$$

Note that all of the equations above depend only on $\Phi_0(x)$, $\Phi_2(x)$, and $\Phi_4(x)$, and this reduced set of equations can be solved for once the appropriate boundary conditions have been defined.

A.3.2 Boundary Conditions

A.3.2.1 Left Boundary

At the left boundary, the incoming surface flux moment (Eq. A.79) serves as the boundary condition for each moment equation.

$$\int_0^1 \mu P_n(\mu)\varphi_L^b(\mu)d\mu \tag{A.79}$$

where the boundary flux ($\varphi_L^b(\mu)$) is simply the the angular flux evaluated at the left boundary (Eq. A.80).

$$\begin{aligned}\varphi_L^b(\mu) = \varphi(\mu,0) =& \frac{1}{2}P_0(\mu)\varphi_0(0) + \frac{3}{2}P_1(\mu)\varphi_1(0) + \frac{5}{2}P_2(\mu)\varphi_2(0) \\ &+ \frac{7}{2}P_3(\mu)\varphi_3(0) + \frac{9}{2}P_4(\mu)\varphi_4(0) + \frac{11}{2}P_5(\mu)\varphi_5(0)\end{aligned} \tag{A.80}$$

The equation for the boundary condition of each moment is shown in Eq. A.81.

n=0:

$$\int_0^1 \mu\varphi_L^b(\mu)d\mu = \frac{1}{4}\varphi_0(0) + \frac{1}{2}\varphi_1(0) + \frac{5}{16}\varphi_2(0) - \frac{3}{32}\varphi_4(0) \tag{A.81a}$$

n=2:

$$\int_0^1 \mu P_2(\mu)\varphi_L^b(\mu)d\mu = \frac{1}{16}\varphi_0(0) + \frac{1}{5}\varphi_1(0) + \frac{5}{16}\varphi_2(0) + \frac{3}{10}\varphi_3(0) + \frac{39}{256}\varphi_4(0) \tag{A.81b}$$

n=4:

$$\int_0^1 \mu P_4(\mu)\varphi_L^b(\mu)d\mu = -\frac{1}{96}\varphi_0(0) + \frac{65}{768}\varphi_2(0) + \frac{2}{9}\varphi_3(0) + \frac{81}{256}\varphi_4(0) + \frac{5}{18}\varphi_5(0) \tag{A.81c}$$

Now the instances of the odd moment will be eliminated, starting by eliminating φ_1 from Eq. A.81b. This can be done by multiplying Eq. A.81a by 2/5 and subtracting from Eq. A.81b.

$$\int_0^1 \mu P_2(\mu)\varphi_L^b(\mu)d\mu = \frac{1}{16}\varphi(0) + \frac{1}{5}\varphi_1(0) + \frac{5}{16}\varphi_2(0) + \frac{3}{10}\varphi_3(0) + \frac{39}{256}\varphi_4(0)$$

$$-\frac{2}{5}\left(\int_0^1 \mu\varphi_L^b(\mu)d\mu\right) = \frac{2}{5}\left(\frac{1}{4}\varphi_0(0) + \frac{1}{2}\varphi_1(0) + \frac{5}{16}\varphi_2(0) - \frac{3}{32}\varphi_4(0)\right)$$

=

$$\int_0^1 \mu P_n(\mu)\varphi_L^b(\mu)d\mu + \frac{2}{5}\left(\int_0^1 \mu\varphi_L^b(\mu)d\mu\right) = \left(\frac{1}{16} - \frac{1}{10}\right)\varphi_0(0) + \left(\frac{1}{5} - \frac{1}{5}\right)\varphi_1(0)$$

$$+\left(\frac{5}{16} - \frac{1}{8}\right)\varphi_2(0) + \left(\frac{3}{10}\right)\varphi_3(0) + \left(\frac{39}{256} - \frac{3}{80}\right)\varphi_4(0)$$

=

$$\frac{3}{5}\int_0^1 P_3(\mu)\varphi_L^b(\mu)d\mu = -\frac{3}{80}\varphi_0(0) + \frac{3}{16}\varphi_2(0) + \frac{3}{10}\varphi_3(0) + \frac{243}{1280}\varphi_4(0)$$

Yielding Eq. A.82:

$$\frac{6}{5}\int_0^1 P_3(\mu)\varphi_L^b(\mu)d\mu = -\frac{3}{40}\varphi_0(0) + \frac{3}{8}\varphi_2(0) + \frac{3}{5}\varphi_3(0) + \frac{243}{640}\varphi_4(0) \tag{A.82}$$

Now, moving to Eq. A.81c, where φ_3 needs to be eliminated:

$$\int_0^1 \mu P_4(\mu)\varphi_L^b(\mu)d\mu = -\frac{1}{96}\varphi_0(0) + \frac{65}{768}\varphi_2(0) + \frac{2}{9}\varphi_3(0) + \frac{81}{256}\varphi_4(0) + \frac{5}{18}\varphi_5(0)$$

$$\frac{10}{27}\int_0^1 P_3(\mu)\varphi_L^b = \frac{10}{27}\left(-\frac{3}{40}\varphi_0(0) + \frac{3}{8}\varphi_2(0) + \frac{3}{5}\varphi_3(0) + \frac{243}{640}\varphi_4(0)\right)$$

=

$$\frac{5}{9}\int_0^1 P_5(\mu)\varphi_L^b(\mu)d\mu = \frac{5}{288}\varphi_0(0) - \frac{125}{2304}\varphi_2(0) + \frac{45}{256}\varphi_4(0) + \frac{5}{18}\varphi_5(0) \tag{A.83}$$

Now that the terms have been removed, Eq. A.77 can be substituted in to simplify the equations. This step includes substituting φ_1 and φ_3 per equations A.78b and A.78d. It is also VERY important to note that the boundary conditions are rewritten to match the equation for the partial current since the SP_N solvers wrap a one-node NEM kernel to handle the spatial expansion. The boundary conditions need to be consistent with the expected input to that kernel.

<u>n=0:</u>

$$\int_0^1 \mu\varphi_L^b(\mu)d\mu = \frac{1}{4}\varphi_0 + \frac{1}{2}\varphi_1 + \frac{5}{16}\varphi_2 - \frac{3}{32}\varphi_4$$

$$\int_0^1 \mu\varphi_L^b(\mu)d\mu = \frac{1}{4}(\Phi_0(0) - 2\Phi_2(0) + \frac{8}{3}\Phi_4(0))$$
$$-\frac{1}{2}D_0\frac{d}{dx}[\Phi_0(0)] + \frac{5}{16}(\Phi_2(0) - \frac{4}{3}\Phi_4(0)) - \frac{3}{32}\varphi_4(0)$$

$$\boxed{-\frac{1}{2}D_0\frac{d}{dx}[\Phi_0(0)] + \frac{1}{4}\Phi_0(0) = \int_0^1 \mu\varphi^b(\mu)d\mu + \frac{3}{16}\Phi_2(0) - \frac{5}{32}\Phi_4(0)} \quad \text{(A.84)}$$

n=2:

$$\frac{6}{5}\int_0^1 P_3(\mu)\varphi_L^b(\mu)d\mu = -\frac{3}{40}\varphi_0 + \frac{3}{8}\varphi_2 + \frac{3}{5}\varphi_3 + \frac{243}{640}\varphi_4 \quad \text{(A.85)}$$

$$\frac{6}{5}\int_0^1 P_3(\mu)\varphi_L^b(\mu)d\mu = -\frac{3}{40}(\Phi_0(0) - 2\Phi_2(0) + \frac{8}{3}\Phi_4(0))$$
$$+\frac{3}{8}(\Phi_2(0) - \frac{4}{3}\Phi_4(0))\frac{3}{5}\frac{5}{3}D_2\frac{d}{dx}[\Phi_2(0)] + \frac{243}{640}\Phi_4(0)$$

$$-D_2\frac{d}{dx}[\Phi_2(0)] + \frac{21}{40}\Phi_2(0) = \frac{6}{5}\int_0^1 P_3(\mu)\varphi^b(\mu)d\mu + \frac{3}{40}\Phi_0(0) + \frac{41}{128}\Phi_4(0)$$

$$\boxed{-\frac{1}{2}D_2\frac{d}{dx}[\Phi_2(0)] + \frac{1}{4}\Phi_2(0) = \frac{3}{5}\int_0^1 P_3(\mu)\varphi^b(\mu)d\mu + \frac{3}{80}\Phi_0(0) - \frac{1}{80}\Phi_2(0) + \frac{41}{256}\Phi_4(0)} \quad \text{(A.86)}$$

n=4:

$$\frac{5}{9}\int_0^1 P_5(\mu)\varphi_L^b(\mu)d\mu = \frac{5}{288}\varphi_0 - \frac{125}{2304}\varphi_2 + \frac{45}{256}\varphi_4 + \frac{5}{18}\varphi_5$$

$$\frac{5}{9}\int_0^1 P_5(\mu)\varphi_L^b(\mu)d\mu = \frac{5}{288}(\Phi_0(0) - 2\Phi_2(0) + \frac{8}{3}\Phi_4(0))$$
$$-\frac{125}{2304}(\Phi_2(0) - \frac{4}{3}\Phi_4(0)) + \frac{45}{256}\Phi_4(0) - \frac{5}{18}\frac{9}{5}D_4\frac{d}{dx}[\Phi_4(0)]$$

$$-\frac{1}{2}D_4\frac{d}{dx}[\Phi_4(0)] + \frac{2035}{6912}\Phi_4(0) = \frac{5}{9}\int_0^1 P_5(\mu)\varphi_L^b d\mu - \frac{5}{288}\Phi_0(0) + \frac{205}{2304}\Phi_2(0)$$

$$\boxed{-\frac{1}{2}D_4\frac{d}{dx}[\Phi_4(0)] + \frac{1}{4}\Phi_4(0) = \frac{5}{9}\int_0^1 P_5(\mu)\varphi_L^b(\mu)d\mu - \frac{5}{288}\Phi_0(0) + \frac{205}{2304}\Phi_2(0) - \frac{307}{6912}\Phi_4(0)} \quad \text{(A.87)}$$

In each of these equations the integral term is stored as it is output from the kernel and sent in as input during the next iteration. It can take several inner iterations to solve the boundary conditions, as with all one-node kernel based sweepers.

A.3.2.2 Right Boundary

A similar procedure as for the left boundary can be performed, but the details of which will be neglected here. Essentially, instead of integrationg from μ from 0 to 1 as was done on the left boundary, now the integration is over the incoming angles of the right boundary, which are from -1 to 0 (Eq. A.88).

$$\int_{-1}^{0} P_n(\mu)\varphi_R^b(\mu)d\mu \tag{A.88}$$

This yields the following equations for the right boundary conditions:
n=0:

$$\boxed{\frac{1}{2}D_0\frac{d}{dx}[\Phi_0(X)] + \frac{1}{4}\Phi_0(X) = \int_{-1}^{0}|\mu|\varphi_R^b(\mu)d\mu + \frac{3}{16}\Phi_2(X) - \frac{5}{32}\Phi_4(X)} \tag{A.89}$$

n=2:

$$\boxed{\frac{1}{2}D_2\frac{d}{dx}[\Phi_2(X)] + \frac{1}{4}\Phi_2(X) = \frac{3}{5}\int_{-1}^{0}P_3(\mu)\varphi_R^b(\mu)d\mu + \frac{3}{80}\Phi_0(X) - \frac{1}{80}\Phi_2(X) + \frac{41}{256}\Phi_4(X)} \tag{A.90}$$

n=4:

$$\boxed{\frac{1}{2}D_4\frac{d}{dx}[\Phi_4(X)] + \frac{1}{4}\Phi_4(X) = -\frac{5}{9}\int_{-1}^{0}P_5(\mu)\varphi_R^b(\mu)d\mu - \frac{5}{288}\Phi_0(X) + \frac{205}{2304}\Phi_2(X) - \frac{307}{6912}\Phi_4(X)} \tag{A.91}$$

A.3.3 Summary/SP$_5$

In summary, the SP$_5$ equations are shown below (Eq. A.92 through A.3.3).

n=0:

$$-D_0\frac{d^2}{dx^2}\Phi_0 + (\Sigma_t - \Sigma_{s0})\Phi_0(x) = Q(x) + 2(\Sigma_t - \Sigma_{s0})\left(\Phi_2(x) - \frac{4}{3}\Phi_4(x)\right) \tag{A.92a}$$

$$-\frac{1}{2}D_0\frac{d}{dx}[\Phi_0(0)] + \frac{1}{4}\Phi_0(0) = \int_0^1 \mu\varphi^b(\mu)d\mu + \frac{3}{16}\Phi_2(0) - \frac{5}{32}\Phi_4(0) \tag{A.92b}$$

$$\frac{1}{2}D_0\frac{d}{dx}[\Phi_0(X)] + \frac{1}{4}\Phi_0(X) = \int_{-1}^{0}|\mu|\varphi_R^b(\mu)d\mu + \frac{3}{16}\Phi_2(X) - \frac{5}{32}\Phi_4(X) \tag{A.92c}$$

n=2:

$$-D_2\frac{d^2}{dx^2}[\Phi_2(x)]+\left(\frac{9}{5}\Sigma_t-\frac{4}{5}\Sigma_{s0}-\Sigma_{s2}\right)\Phi_2(x) = -\frac{2}{5}\left(Q(x)-(\Sigma_t-\Sigma_{s0})\Phi_0(x)\right)+\left(\frac{28}{15}\Sigma_t-\frac{16}{15}\Sigma_{s0}-\frac{4}{5}\Sigma_{s2}\right)\Phi_4 \tag{A.93a}$$

$$-\frac{1}{2}D_2\frac{d}{dx}[\Phi_2(0)]+\frac{1}{4}\Phi_2(0)=\frac{3}{5}\int_0^1 P_3(\mu)\varphi^b(\mu)d\mu+\frac{3}{80}\Phi_0(0)-\frac{1}{80}\Phi_2(0)+\frac{41}{256}\Phi_4(0) \tag{A.93b}$$

$$\frac{1}{2}D_2\frac{d}{dx}[\Phi_2(X)]+\frac{1}{4}\Phi_2(X)=\frac{3}{5}\int_{-1}^0 P_3(\mu)\varphi_R^b(\mu)d\mu+\frac{3}{80}\Phi_0(X)-\frac{1}{80}\Phi_2(X)+\frac{41}{256}\Phi_4(X) \tag{A.93c}$$

n=4:

$$-D_4\frac{d^2}{dx^2}[\Phi_4(x)]+\left(\frac{225}{81}\Sigma_t+\frac{80}{81}\Sigma_{s2}-\frac{64}{81}\Sigma_{s0}-\Sigma_{s4}\right)\Phi_4(x) = \left[\frac{8}{27}(Q(x)-(\Sigma_t-\Sigma_{s0})\Phi_0(x))+\left(\frac{36}{27}\Sigma_t-\frac{16}{27}\Sigma_{s0}-\frac{20}{27}\Sigma_{s2}\right)\Phi_2(x)\right] \tag{A.94a}$$

$$-\frac{1}{2}D_4\frac{d}{dx}[\Phi_4(0)]+\frac{1}{4}\Phi_4(0)=\frac{5}{9}\int_0^1 P_5(\mu)\varphi_L^b(\mu)d\mu-\frac{5}{288}\Phi_0(0)+\frac{205}{2304}\Phi_2(0)-\frac{307}{6912}\Phi_4(0) \tag{A.94b}$$

$$\frac{1}{2}D_4\frac{d}{dx}[\Phi_4(X)]+\frac{1}{4}\Phi_4(X)=-\frac{5}{9}\int_{-1}^0 P_5(\mu)\varphi_R^b(\mu)d\mu-\frac{5}{288}\Phi_0(X)+\frac{205}{2304}\Phi_2(X)-\frac{307}{6912}\Phi_4(X) \tag{A.94c}$$

A.3.4 SP_3

The equations for the SP_3 solver can be determined by neglecting the fourth moment equation entirely and setting $\Phi_4(x) = 0$ in the zeroth and second moments:

n=0:

$$-D_0\frac{d^2}{dx^2}\Phi_0+(\Sigma_t-\Sigma_{s0})\Phi_0(x)=Q(x)+2(\Sigma_t-\Sigma_{s0})\Phi_2(x) \tag{A.95a}$$

$$-\frac{1}{2}D_0\frac{d}{dx}[\Phi_0(0)]+\frac{1}{4}\Phi_0(0)=\int_0^1\mu\varphi^b(\mu)d\mu+\frac{3}{16}\Phi_2(0) \tag{A.95b}$$

$$\frac{1}{2}D_0\frac{d}{dx}[\Phi_0(X)]+\frac{1}{4}\Phi_0(X)=\int_{-1}^0|\mu|\varphi_R^b(\mu)d\mu+\frac{3}{16}\Phi_2(X) \tag{A.95c}$$

n=2:

$$-D_2\frac{d^2}{dx^2}[\Phi_2(x)]+\left(\frac{9}{5}\Sigma_t-\frac{4}{5}\Sigma_{s0}-\Sigma_{s2}\right)\Phi_2(x) = -\frac{2}{5}\left(Q(x)-(\Sigma_t-\Sigma_{s0})\Phi_0(x)\right) \tag{A.96a}$$

$$-\frac{1}{2}D_2\frac{d}{dx}[\Phi_2(0)]+\frac{1}{4}\Phi_2(0)=\frac{3}{5}\int_0^1 P_3(\mu)\varphi^b(\mu)d\mu+\frac{3}{80}\Phi_0(0)-\frac{1}{80}\Phi_2(0) \tag{A.96b}$$

$$\frac{1}{2}D_2\frac{d}{dx}[\Phi_2(X)]+\frac{1}{4}\Phi_2(X)=\frac{3}{5}\int_{-1}^0 P_3(\mu)\varphi_R^b(\mu)d\mu+\frac{3}{80}\Phi_0(X)-\frac{1}{80}\Phi_2(X) \tag{A.96c}$$

A.3.5 SP_1

Lastly, the SP_1 equations can be found in a similar manner by taking the SP_3 equations, neglecting the second moment equations and setting $\Phi_0 = 0$ in the zeroth moment equations:

$$-D_0\frac{d^2}{dx^2}\Phi_0+(\Sigma_t-\Sigma_{s0})\Phi_0(x)=Q(x) \tag{A.97a}$$

$$-\frac{1}{2}D_0\frac{d}{dx}[\Phi_0(0)]+\frac{1}{4}\Phi_0(0)=\int_0^1 \mu\varphi^b(\mu)d\mu \tag{A.97b}$$

$$\frac{1}{2}D_0\frac{d}{dx}[\Phi_0(X)]+\frac{1}{4}\Phi_0(X)=\int_{-1}^0 |\mu|\varphi_R^b(\mu)d\mu \tag{A.97c}$$

APPENDIX B

C5G7 SHIFT Results

This appendix is dedicated to encompassing anything related to the generation of C5G7 results with SHIFT [11], which were used as the primary reference solution in this work. This was primarily done to obtain more convergence pin power distributions for the standard C5G7-3D extended cases [52]. In the end, this proved particularly useful when generating reference results for the alternative problems using homogenized pin geometries as there was no pre-existing reference for these cases.

For each of the problems presented here, the eigenvalue and associated uncertainty will be specified as well as information on the pin power (fission rate) distributions (value, error, and asymmetry). In these plots, the fission rate and errors (standard deviation) have been adjusted to normalize the fission rates. Quarter symmetry was used in instead of eighth symmetry to provide a comparison to the error values reported by SHIFT, but the final reference results were averaged in accordance with symmetry (so no asymmetry in the reference solution was allowed). All cases were run using 2M histories per cycle with 120,000 active cycles (500 inactive).

B.1 2D Results

The first problem under consideration is the 2D-C5G7 problem which SHFT calculated to have an eigenvalue of 1.186487 with an standard deviation of 0.586 pcm. Figures B.1 through B.3 show the pin power results and standard deviation errors.

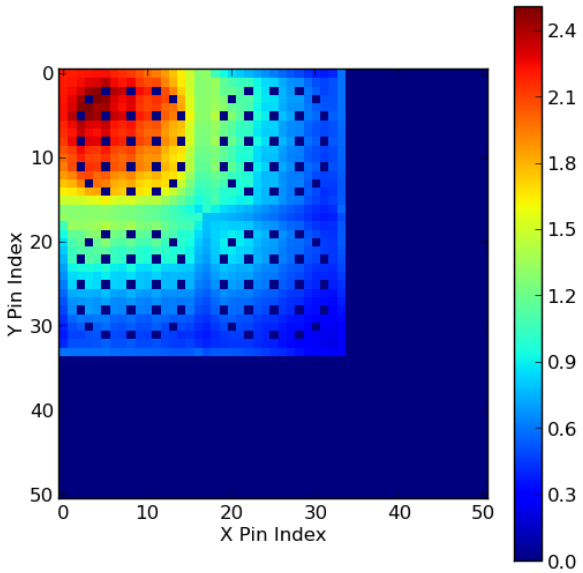

Figure B.1: SHIFT 2D-C5G7 Pin Powers

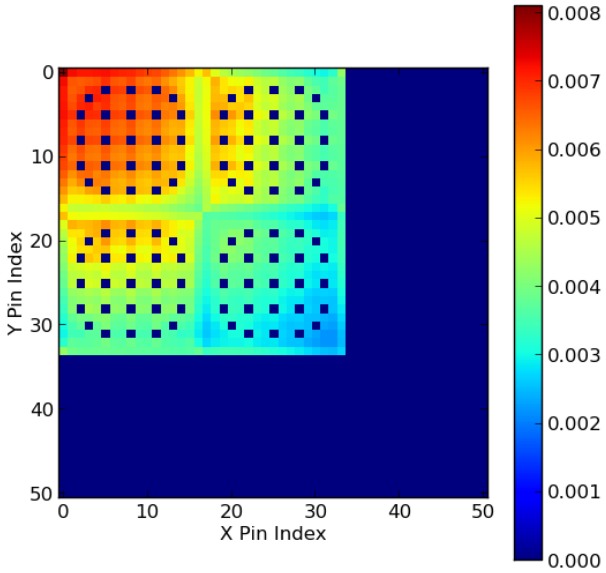

Figure B.2: SHIFT 2D-C5G7 Pin Power Errors

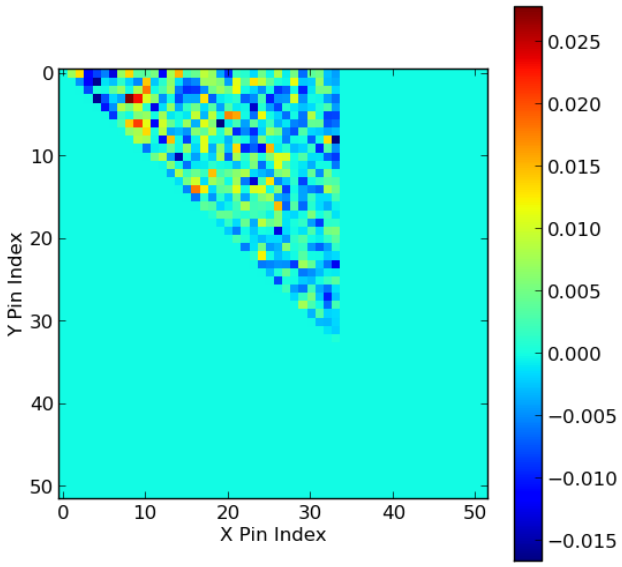

Figure B.3: SHIFT 2D-C5G7 Pin Power Symmetry Difference

B.2 3D Extended (Het. Pins) Results

B.2.1 Unrodded Configuration

SHIFT yielded an eigenvalue of 1.143037 with a standard deviation of 0.592 pcm. Figures B.4 through B.6 show the pin power distribution results. In all of these results, the pin powers were tallied on three planes. The bottom-most plane is on the left and top-most on the right in all of these figures.

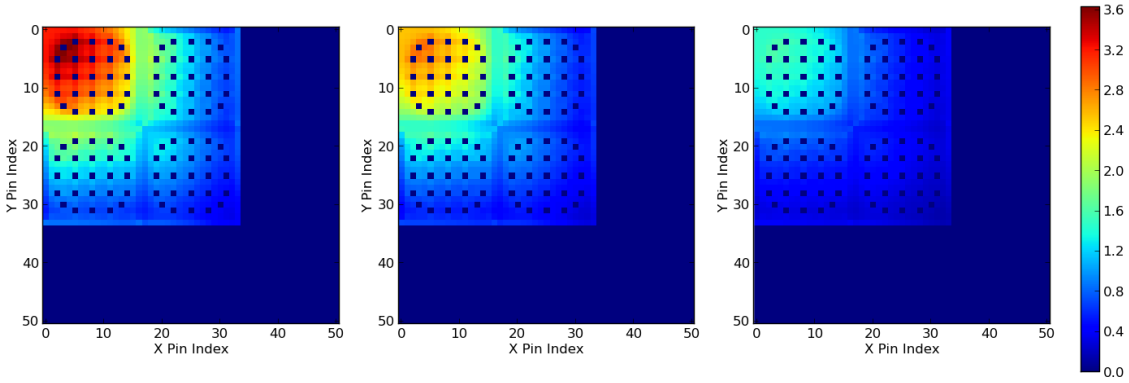

Figure B.4: SHIFT 3D-C5G7-UR Pin Powers

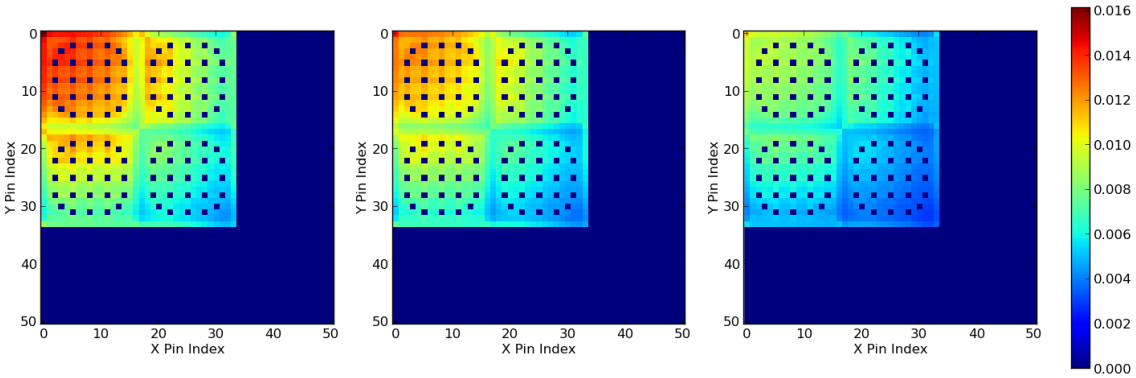

Figure B.5: SHIFT 3D-C5G7-UR Pin Power Errors

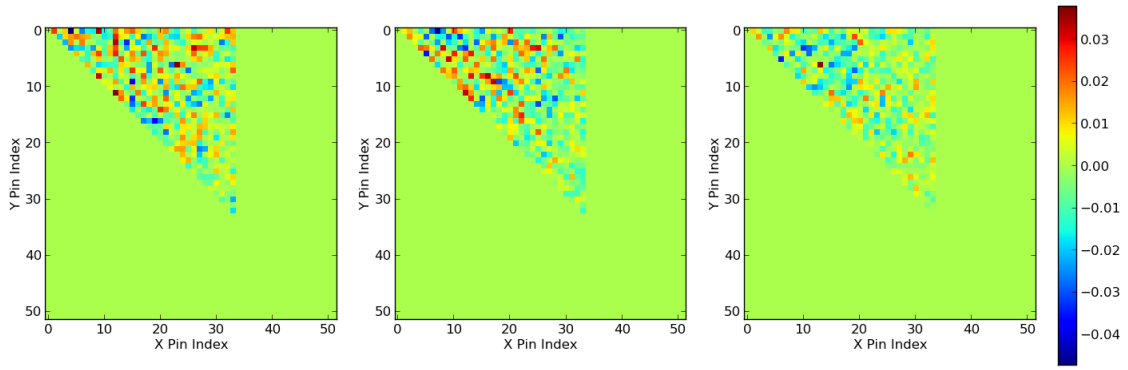

Figure B.6: SHIFT 3D-C5G7-UR Pin Power Symmetry Difference

B.2.2 Rodded Configuration A

SHIFT yielded an eigenvalue of 1.128097 with a standard deviation of 0.591 pcm.

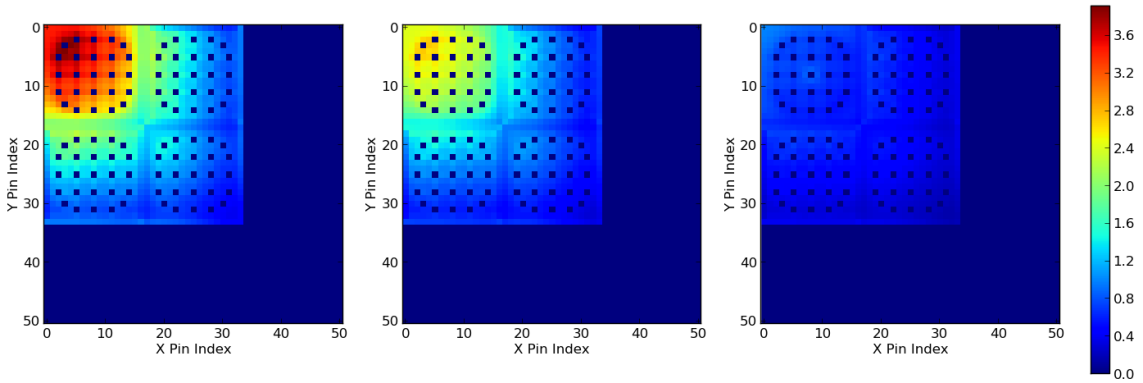

Figure B.7: SHIFT 3D-C5G7-rA Pin Powers

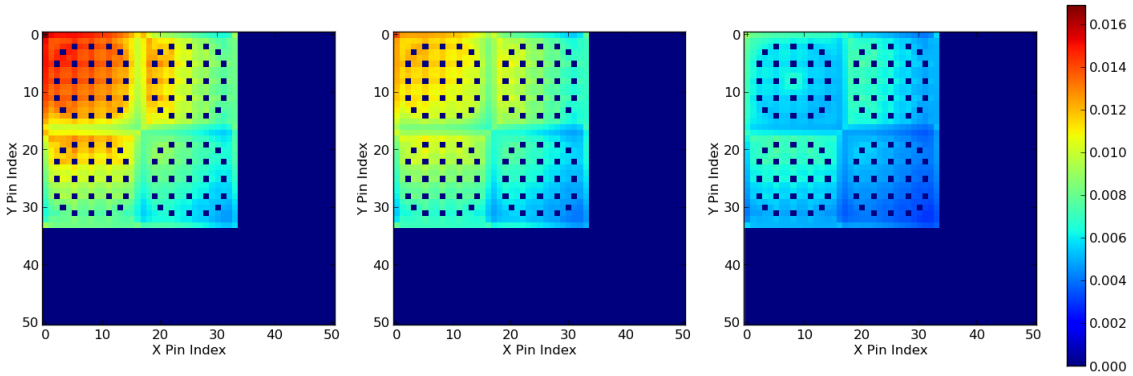

Figure B.8: SHIFT 3D-C5G7-rA Pin Power Errors

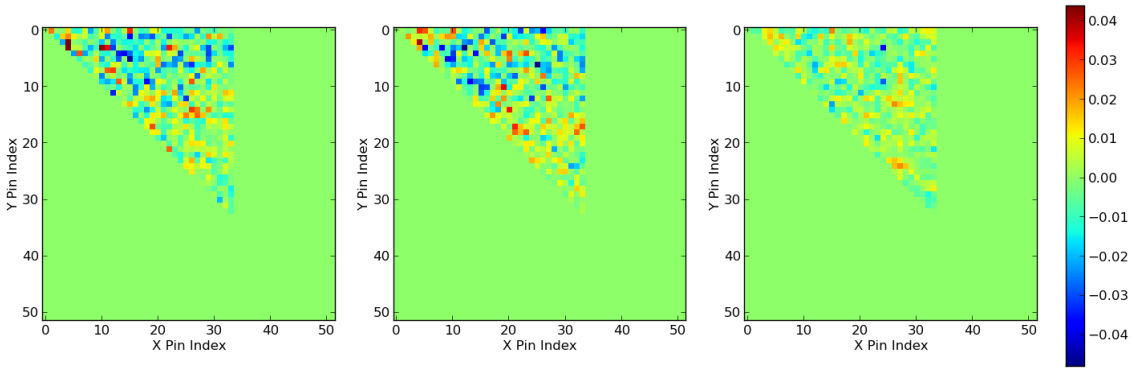

Figure B.9: SHIFT 3D-C5G7-rA Pin Power Symmetry Difference

B.2.3 Rodded Configuration B

SHIFT yielded an eigenvalue of 1.077716 with a standard deviation of 0.579 pcm.

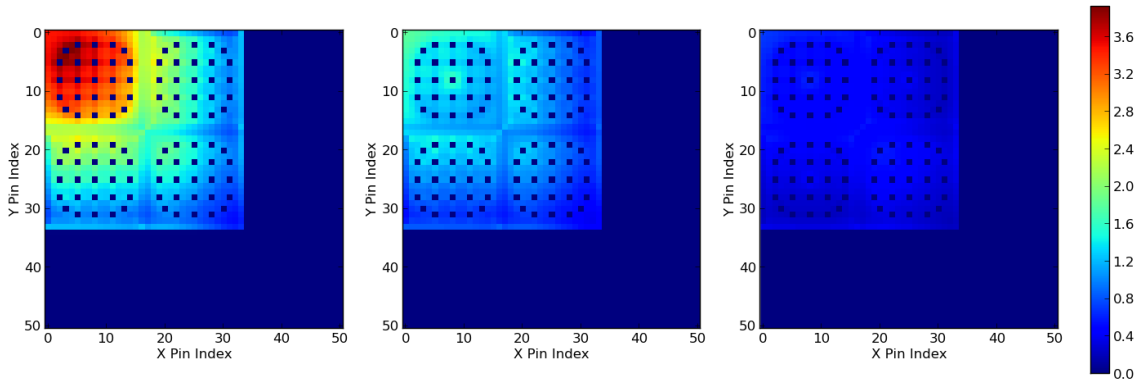

Figure B.10: SHIFT 3D-C5G7-rB Pin Powers

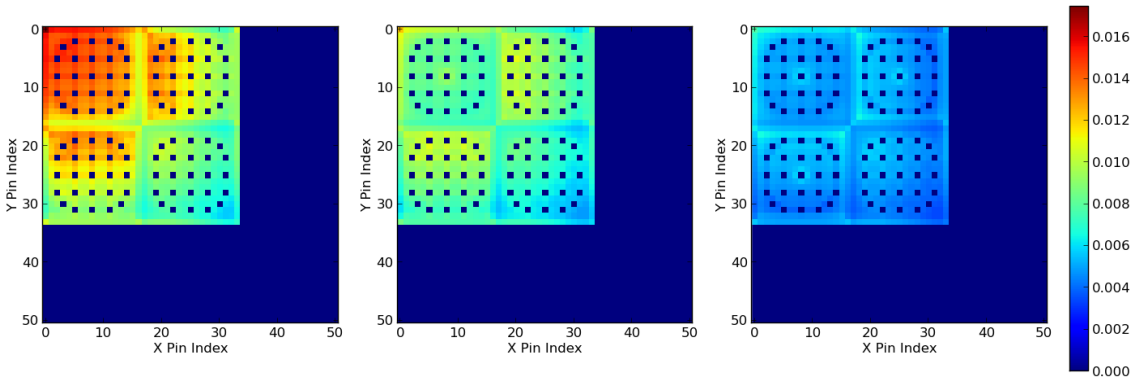

Figure B.11: SHIFT 3D-C5G7-rB Pin Power Errors

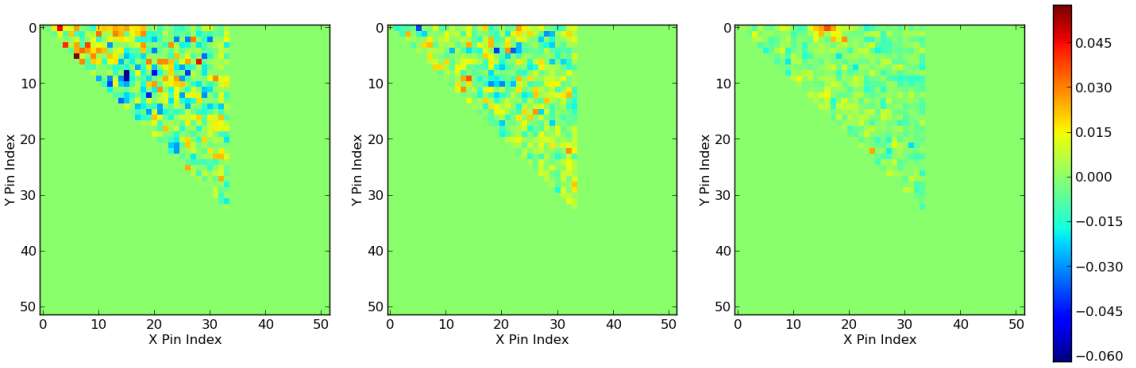

Figure B.12: SHIFT 3D-C5G7-rB Pin Power Symmetry Difference

B.3 3D Extended (Hom. Pins) Results

B.3.1 Unrodded Configuration

SHIFT yielded an eigenvalue of 1.145539 with a standard deviation of 0.488 pcm.

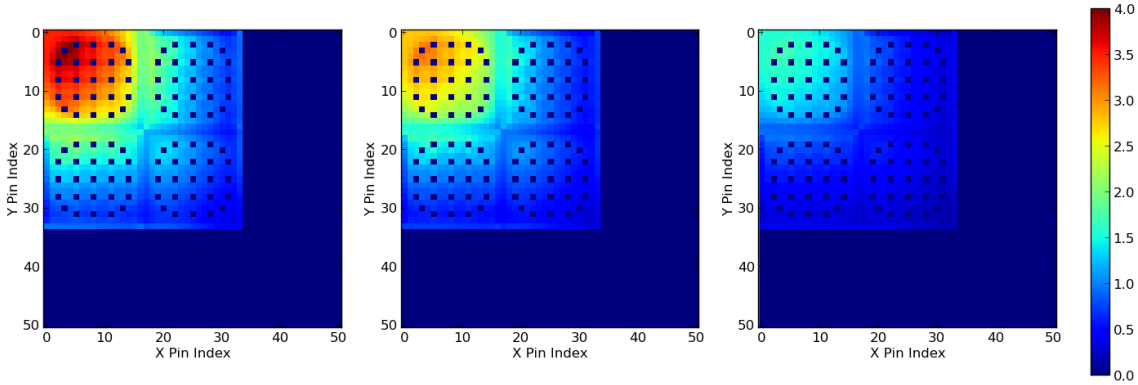

Figure B.13: SHIFT 3D-C5G7-UR (Hom. Pins) Pin Powers

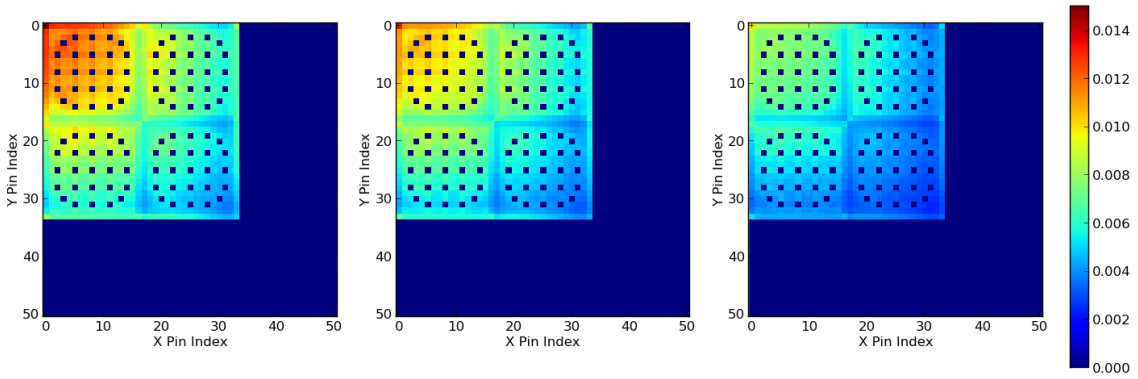

Figure B.14: SHIFT 3D-C5G7-UR (Hom. Pins) Pin Power Errors

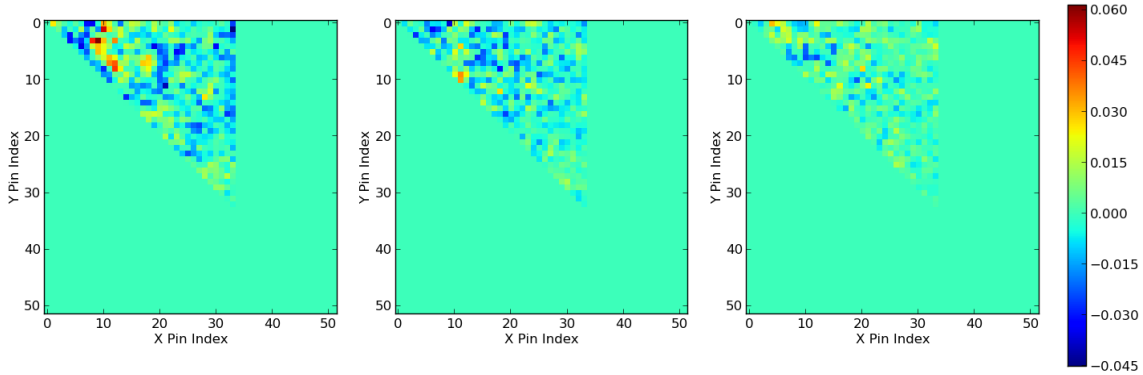

Figure B.15: SHIFT 3D-C5G7-UR (Hom. Pins) Pin Power Symmetry Difference

B.3.2 Rodded Configuration A

SHIFT yielded an eigenvalue of 1.129344 with a standard deviation of 0.488 pcm.

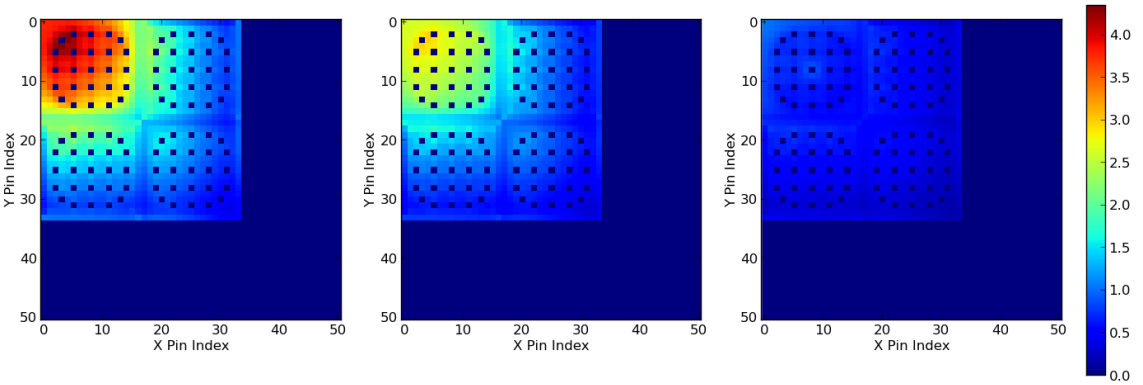

Figure B.16: SHIFT 3D-C5G7-rA (Hom. Pins) Pin Powers

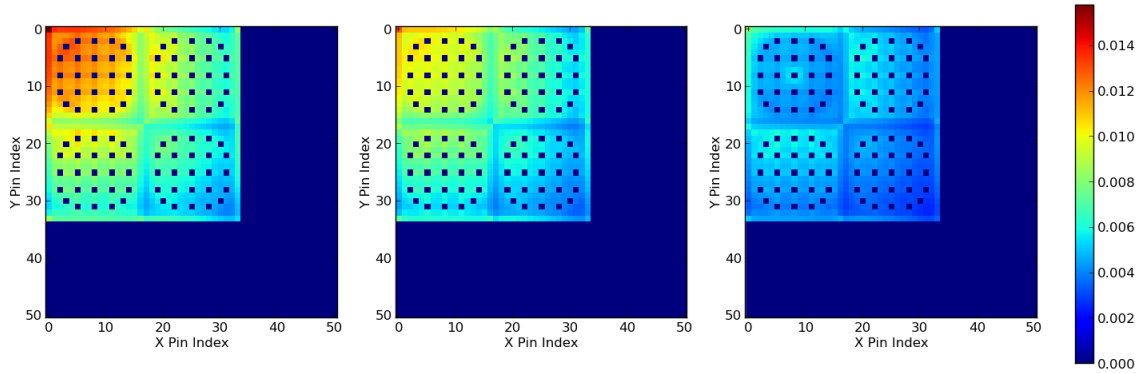

Figure B.17: SHIFT 3D-C5G7-rA (Hom. Pins) Pin Power Errors

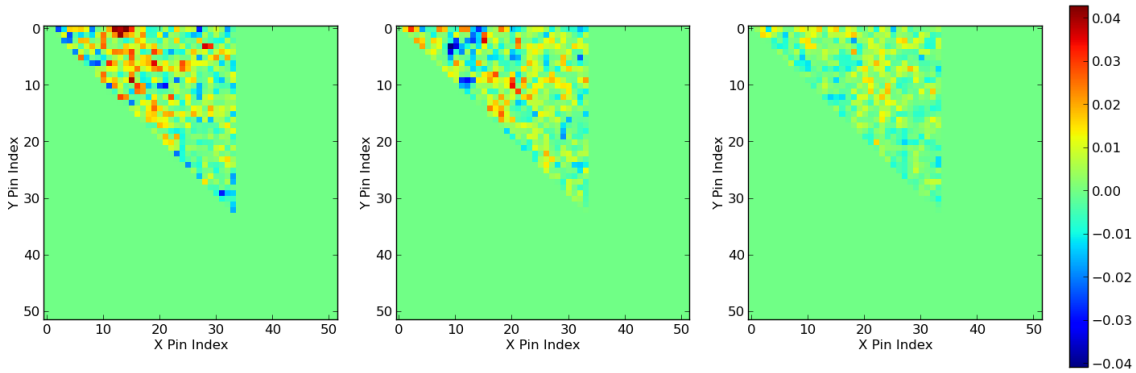

Figure B.18: SHIFT 3D-C5G7-rA (Hom. Pins) Pin Power Symmetry Difference

B.3.3 Rodded Configuration B

SHIFT yielded an eigenvalue of 1.074680 with a standard deviation of 0.582 pcm.

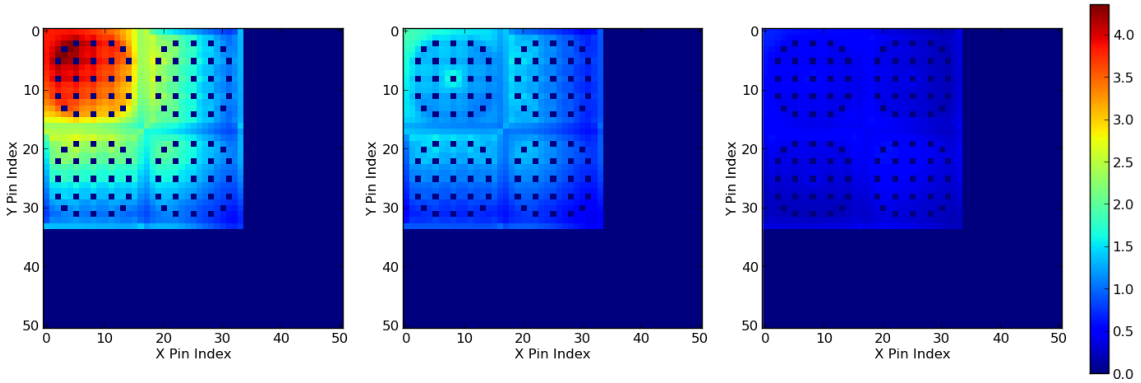

Figure B.19: SHIFT 3D-C5G7-rB (Hom. Pins) Pin Powers

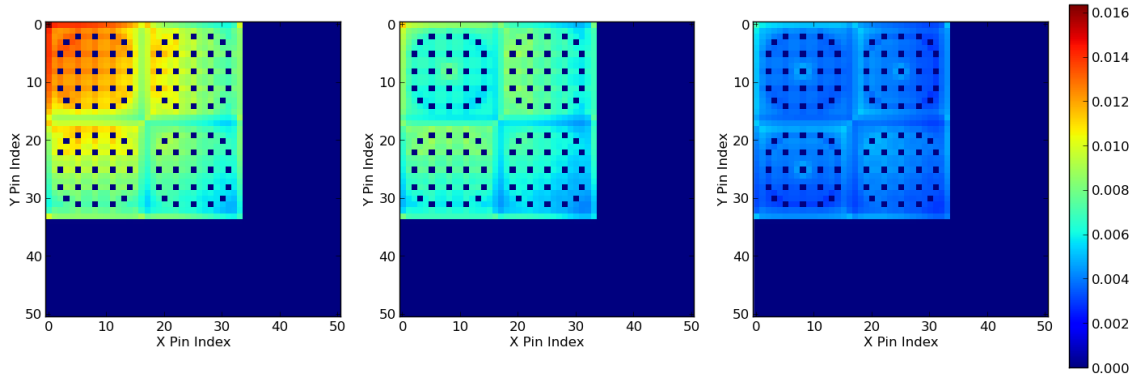

Figure B.20: SHIFT 3D-C5G7-rB (Hom. Pins) Pin Power Errors

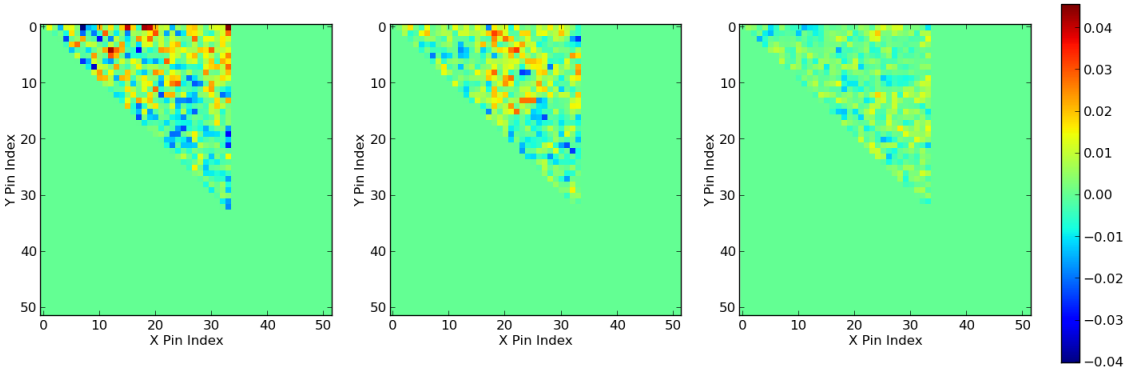

Figure B.21: SHIFT 3D-C5G7-rB (Hom. Pins) Pin Power Symmetry Difference

APPENDIX C

Homogenized-Pin 3D-C5G7 Cross Sections

Shown below are the volume homogenized cross sections used in MPACT for the homogenized pins 3D-C5G7 results. Included are the absorption (Σ_a), nu-fission ($\nu\Sigma_f$), fission (Σ_f) cross sections, and fission spectrum (χ). These four data are listed for each of the seven groups, then the scattering matrix ($\Sigma_{g'\rightarrow g}$) are shown below that.

```
!    Abs        nu-fiss        fiss         chi
!  scat mat
!UO2 fuel-clad
XSMACRO UO2-3.3 0
  4.88476E-03 1.15752E-02 4.16155E-03 5.87910E-01
  2.15172E-03 1.16981E-03 4.72759E-04 4.11760E-01
  1.55891E-02 9.06278E-03 3.72367E-03 3.39060E-04
  5.63516E-02 2.60718E-02 1.07124E-02 1.17610E-07
  1.97509E-02 2.50096E-02 1.02759E-02 0.00000E+00
  7.05451E-02 1.16611E-01 4.79133E-02 0.00000E+00
  1.78923E-01 3.03349E-01 1.24640E-01 0.00000E+00
  9.24052E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  7.24184E-02 3.06640E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  3.11454E-04 5.59024E-02 4.06239E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  1.58929E-06 2.63683E-04 9.65330E-02 2.99645E-01 1.02489E-04 0.00000E+00 0.00000E+00
  2.24954E-08 2.03035E-05 7.19011E-03 1.78961E-01 2.15457E-01 1.68547E-03 0.00000E+00
  0.00000E+00 3.15056E-06 1.11847E-03 2.69569E-02 2.22403E-01 4.49419E-01 6.09497E-02
  0.00000E+00 4.42218E-07 2.12941E-04 5.13447E-03 2.58982E-02 2.36971E-01 1.20684E+00

!4.3% MOX fuel-clad
XSMACRO MOX-4.3 0
  5.12082E-03 1.25521E-02 4.40101E-03 5.87910E-01
  2.17498E-03 1.46282E-03 5.05994E-04 4.11760E-01
  1.62821E-02 9.38707E-03 3.28810E-03 3.39060E-04
  6.09528E-02 3.77803E-02 1.32065E-02 1.17610E-07
  8.31777E-02 1.77286E-02 6.21083E-03 0.00000E+00
  2.42453E-01 3.84676E-01 1.34307E-01 0.00000E+00
  2.51957E-01 4.11992E-01 1.43661E-01 0.00000E+00
  9.31778E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  7.18615E-02 3.07214E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  3.10756E-04 5.59071E-02 4.07536E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  1.58901E-06 2.63682E-04 9.64955E-02 3.02304E-01 1.22806E-04 0.00000E+00 0.00000E+00
```

```
  2.24954E-08 2.03035E-05 7.19011E-03 1.78946E-01 2.18581E-01 2.09418E-03 0.00000E+00
  0.00000E+00 3.15056E-06 1.11847E-03 2.69569E-02 2.21860E-01 4.42010E-01 6.09202E-02
  0.00000E+00 4.42218E-07 2.12941E-04 5.13447E-03 2.58982E-02 2.35841E-01 1.20218E+00

!7.0% MOX fuel-clad
XSMACRO MOX-7.0 0
  5.48538E-03 1.37413E-02 4.76305E-03 5.87910E-01
  2.48599E-03 2.22657E-03 7.64936E-04 4.11760E-01
  1.91158E-02 1.39260E-02 4.85947E-03 3.39060E-04
  7.12355E-02 5.44519E-02 1.89686E-02 1.17610E-07
  1.08013E-01 2.64105E-02 9.21143E-03 0.00000E+00
  3.34362E-01 5.35586E-01 1.86838E-01 0.00000E+00
  3.53433E-01 6.01955E-01 2.09347E-01 0.00000E+00
  9.40901E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  7.20802E-02 3.08931E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  3.10919E-04 5.59095E-02 4.10526E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  1.58907E-06 2.63683E-04 9.64487E-02 3.06075E-01 1.31882E-04 0.00000E+00 0.00000E+00
  2.24954E-08 2.03035E-05 7.19011E-03 1.78910E-01 2.21754E-01 2.25049E-03 0.00000E+00
  0.00000E+00 3.15056E-06 1.11847E-03 2.69569E-02 2.21523E-01 4.40157E-01 6.11336E-02
  0.00000E+00 4.42218E-07 2.12941E-04 5.13447E-03 2.58982E-02 2.34839E-01 1.19902E+00

!8.7% MOX fuel-clad
XSMACRO MOX-8.7 0
  5.72802E-03 1.45330E-02 5.00403E-03 5.87910E-01
  2.69309E-03 2.73483E-03 9.37242E-04 4.11760E-01
  2.10541E-02 1.70096E-02 5.92699E-03 3.39060E-04
  7.74039E-02 6.47713E-02 2.25299E-02 1.17610E-07
  1.22681E-01 3.19113E-02 1.11122E-02 0.00000E+00
  3.86433E-01 6.20304E-01 2.16321E-01 0.00000E+00
  4.13998E-01 7.15109E-01 2.48467E-01 0.00000E+00
  9.46942E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  7.22268E-02 3.10071E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  3.11027E-04 5.59110E-02 4.12500E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  1.58910E-06 2.63683E-04 9.64151E-02 3.08563E-01 1.37526E-04 0.00000E+00 0.00000E+00
  2.24954E-08 2.03035E-05 7.19011E-03 1.78884E-01 2.23749E-01 2.31720E-03 0.00000E+00
  0.00000E+00 3.15056E-06 1.11847E-03 2.69569E-02 2.21331E-01 4.38924E-01 6.11934E-02
  0.00000E+00 4.42218E-07 2.12941E-04 5.13447E-03 2.58982E-02 2.34382E-01 1.19704E+00

!Fission chamber
XSMACRO FissCham 0
  5.49273E-04 7.63638E-09 2.76397E-09 5.87910E-01
  5.04262E-05 8.27746E-09 3.36155E-09 4.11760E-01
  3.25198E-04 6.51232E-07 2.67578E-07 3.39060E-04
  1.49450E-03 7.36459E-06 3.02597E-06 1.17610E-07
  4.38911E-03 2.04181E-07 8.38940E-08 0.00000E+00
  1.16471E-02 1.00408E-06 4.12558E-07 0.00000E+00
  2.91635E-02 2.92166E-06 1.20045E-06 0.00000E+00
  5.69924E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  8.20501E-02 2.58124E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  4.69503E-04 8.52175E-02 2.51875E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  2.42983E-06 4.07880E-04 1.48240E-01 8.41321E-02 5.17621E-05 0.00000E+00 0.00000E+00
  3.44064E-08 3.14050E-05 1.11927E-02 2.73838E-01 1.16414E-01 1.46656E-03 0.00000E+00
  0.00000E+00 4.87442E-06 1.74108E-03 4.18788E-02 3.35809E-01 4.78831E-01 8.47504E-02
  0.00000E+00 6.85377E-07 3.31480E-04 7.97667E-03 4.00227E-02 3.65043E-01 1.68348E+00

```

```
!Guide tube
XSMACRO GuideTube 0
  5.49273E-04 0.00000E+00 0.00000E+00 0.00000E+00
  5.04192E-05 0.00000E+00 0.00000E+00 0.00000E+00
  3.24789E-04 0.00000E+00 0.00000E+00 0.00000E+00
  1.48913E-03 0.00000E+00 0.00000E+00 0.00000E+00
  4.38899E-03 0.00000E+00 0.00000E+00 0.00000E+00
  1.16466E-02 0.00000E+00 0.00000E+00 0.00000E+00
  2.91623E-02 0.00000E+00 0.00000E+00 0.00000E+00
  5.69924E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  8.20501E-02 2.58124E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  4.69503E-04 8.52175E-02 2.51801E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  2.42983E-06 4.07880E-04 1.48303E-01 8.40018E-02 5.17581E-05 0.00000E+00 0.00000E+00
  3.44064E-08 3.14050E-05 1.11973E-02 2.73925E-01 1.16403E-01 1.46646E-03 0.00000E+00
  0.00000E+00 4.87442E-06 1.74183E-03 4.18909E-02 3.35809E-01 4.78826E-01 8.47499E-02
  0.00000E+00 6.85377E-07 3.31619E-04 7.97904E-03 4.00227E-02 3.65049E-01 1.68349E+00

!Moderator
XSMACRO Moderator 0
  6.0105E-04 0.000000E+00 0.00000E+00 0.0000E+00
  1.5793E-05 0.000000E+00 0.00000E+00 0.0000E+00
  3.3716E-04 0.000000E+00 0.00000E+00 0.0000E+00
  1.9406E-03 0.000000E+00 0.00000E+00 0.0000E+00
  5.7416E-03 0.000000E+00 0.00000E+00 0.0000E+00
  1.5001E-02 0.000000E+00 0.00000E+00 0.0000E+00
  3.7239E-02 0.000000E+00 0.00000E+00 0.0000E+00
  4.44777E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  1.13400E-01 2.82334E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  7.23470E-04 1.29940E-01 3.45256E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  3.74990E-06 6.23400E-04 2.24570E-01 9.10284E-02 7.14370E-05 0.00000E+00 0.00000E+00
  5.31840E-08 4.80020E-05 1.69990E-02 4.15510E-01 1.39138E-01 2.21570E-03 0.00000E+00
  0.00000E+00 7.44860E-06 2.64430E-03 6.37320E-02 5.11820E-01 6.99913E-01 1.32440E-01
  0.00000E+00 1.04550E-06 5.03440E-04 1.21390E-02 6.12290E-02 5.37320E-01 2.48070E+00

!Control Rod
XSMACRO CRod 0
  1.23800E-03 0.00000E+00 0.00000E+00 0.00000E+00
  4.83192E-03 0.00000E+00 0.00000E+00 0.00000E+00
  4.84918E-02 0.00000E+00 0.00000E+00 0.00000E+00
  2.30361E-01 0.00000E+00 0.00000E+00 0.00000E+00
  4.05634E-01 0.00000E+00 0.00000E+00 0.00000E+00
  5.42696E-01 0.00000E+00 0.00000E+00 0.00000E+00
  6.95697E-01 0.00000E+00 0.00000E+00 0.00000E+00
  1.17232E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  7.35858E-02 3.91228E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  3.62769E-04 5.53566E-02 6.08728E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
  1.65984E-06 2.63681E-04 9.54025E-02 3.67842E-01 6.80436E-05 0.00000E+00 0.00000E+00
  2.24954E-08 2.03035E-05 7.19011E-03 1.76592E-01 1.78780E-01 1.52821E-03 0.00000E+00
  0.00000E+00 3.15056E-06 1.11847E-03 2.69569E-02 2.18687E-01 4.12872E-01 5.80557E-02
  0.00000E+00 4.42218E-07 2.12941E-04 5.13447E-03 2.58982E-02 2.30014E-01 1.42930E+00
```

Bibliography

[1] S. BARLAY, et. al., "PETSc Users Manual," Argonne National Laboratory, ANL-95/11 - Revision 3.5, http://www.mcs.anl.gov/petsc (2014)

[2] T. L. Becker, "Hybrid Monte Carlo/Determistic Methods for Radiation Shielding Problems," Ph.D. Dissertation, University of Michigan (2009).

[3] "Consortium for Advanced Simulation of Light Water Reactors (CASL)," http://www.casl.gov/.

[4] J. Y. CHO, et. al., "Three-Dimensional Heterogeneous Whole Core Transport Calculation Employing Planar MOC Solutions," *Trans. Am. Nucl. Soc.*, Vol. 87, p. 234-236 (2002).

[5] J. Y. CHO and H. G. JOO, "Solution of the C5G7MOX benchmark three-dimensional extension problems by the DeCART direct whole core calculation code," *Progress in Nuclear Energy*, Vol. 48, p. 456-466 (2006).

[6] J. Y. CHO, et. al., "Axial SPN and radial MOC coupled transport calculation." *Journal of Nuclear Science and Technology*, Vol. 44, p. 1156-1171 (2007)

[7] N. Z. CHO, G. S. LEE, and C. J. PARK, "Fusion Method of Characteristics for 3D Whole Core Transport Calculation," *Trans. Am. Nucl. Soc.*, Vol. 86, p. 322-324 (2002).

[8] N. Z. CHO, G. S. LEE, and C. J. PARK, "A Fusion Technique of 2-D/1-D Methods for Three-Dimensional Whole-Core Transport Calculations," *Proc. Korean Nucl. Soc.*, Kwangju, Korea (2002).

[9] N. Z. CHO, et. al. "Refinement of the 2-D/1-D Fusion Method for 3-D Whole-Core Transport Calculation," *Trans. Am. Nucl. Soc.*, Vol. 87, p. 417-420 (2002).

[10] N. Z. CHO, "Fundamentals and Recent Developments of Reactor Physics Methods," *Nuclear Engineering and Technology*, Vol. 47, p. 25-78 (2005).

[11] T. EVANS, et. al., "Exhnihilo Documentation: Release 5.1 (Dev)," Oak Ridge National Laboratory (August 2014).

[12] R. FERRER, J. D. RHODES, and K. S. SMITH, "Linear Source Approximation in CASMO5," *Prof. PHYSOR 2012*, Knoxville, TN (April 15-20, 2012).

[13] F. FÉVOTTE and B. LATHUILIÈRE, "MICADO: Parallel Implementation of a 2D-1D Iterative Algorithm for 3D Neutron Transport Problem in Prismatic Geometries," *Proc. M&C 2013*, Sun Valley, ID (May 5-9, 2013).

[14] F. FRACNESCHINI, et. al., "Westinghouse VERA Test Stand: Zero Power Physics Test Simulations for the AP1000(R) PWR," Tech. Rep. CASL-U-2014-0012-000, http://www.casl.gov/publications.shtml (2014).

[15] F. FINNEMANN and M. WAGNER, "Interface nodal current technique for multi-dimensional reactor calculation," Atomkernenergie, Vol. 30, p. 123 (1977).

[16] A. T. GODFREY, "VERA Core Physics Benchmark Progression Problem Specifications (Rev. 2)," Tech. Rep. CASL-U-2012-0131-002, Oak Ridge National Laboratory, Oak Ridge, TN, http://www.casl.gov/publications.shtml (2013).

[17] A. HÉBERT, Applied Reactor Physics, Presses internationales Polytechnique (2009).

[18] A. HÉBERT, "Multigroup Neutron Transport and Diffusion Computations," Handbook of Nuclear Engineering, D.G. Cacuci (Ed.), p. 751-912 (2010).

[19] M. HURSIN, "Full Core, Heterogeneous, Time Dependent Neutron Transport Calculations with the 3D Code DeCART," Ph.D. Dissertation, University of California at Berkeley (2010).

[20] M. HURSIN, et. al., "The Development and Implementation of a One-Dimensional Sn Method in the 2D-1D Integral Transport Solution," *Nuclear Science and Engineering*, Vol. 176, p. 186-200 (2014).

[21] M. HURSIN, B. KOCHUNAS, and T. DOWNAR, "DeCART Theory Manual", University of Michigan (2008).

[22] A. JONSSON and R. A. LORETZ, "Historical and Recent Developments, Applications and Performance of the DIT^{TM} Assembly Lattice Code," *Proc. M&C 1991*, Pittsburgh, PA (1991).

[23] A. JONSSON, "The DIT Nuclear Fuel Assembly Physics Design Code," *Proc. M&C 1987*, Paris, France (1987).

[24] H. G. JOO, et. al, "Methods and Performance of a Three-Dimensional Whole-Core Transport Code DeCART", *Proc. PHYSOR 2004*, American Nuclear Society, Chicago, IL (April 25-29, 2004).

[25] B. W. KELLEY and E. W. LARSEN, "2D/1D approximations to the 3D neutron transport equation. I: Theory," *Proc. M&C 2013*, Sun Valley, ID (May 5-9, 2013).

[26] B. W. KELLEY, B. S. COLLINS, and E. W. LARSEN, "2D/1D approximations to the 3D neutron transport equation. II: Numerical Results," *Proc. M&C 2013*, Sun Valley, ID (May 5-9, 2013).

[27] B. W. KELLEY, "Relaxation Factors for NEM Axial Solvers," Personal Communication, 2013.

[28] "Table of Nuclides: Cross Section Plotter," Korea Atomic Energy Research Institute (KAERI), http://atom.kaeri.re.kr/ (2000).

[29] B. M. KOCHUNAS, "A Hybrid Parallel Algorithm for the 3-D Method of Charateristics Solution of the Boltzmann Transport Equation on High Performance Compute Clusters," Ph.D. Dissertation, University of Michigan (2013).

[30] B. M. KOCHUNAS, et. al., "Application of the SDD-CMFD Acceleration Method to Parallel 3-D MOC," *Proc. PHYSOR 2014*, Kyoto, Japan (September 28 - October 3, 2014).

[31] B. M. KOCHUNAS, et. al., "Overview of Development and Design of MPACT: Michigan Parallel Characteristics Transport Code," *Proc. M&C 2013*, Sun Valley, ID (May 5-9, 2013).

[32] A. LEONARD and C.T. MCDANIEL, "Optimal Polar Angles and Weights for the Characteristics Method." *Trans. Am. Nucl. Soc.*, Vol. 73, No. 172 (1995).

[33] G. S. LEE and N. Z. CHO, "2D/1D Fusion method solutions of the three-dimensional OECD benchmark problem C5G7 MOX," *Progress in Nuclear Energy*, Vol. 48, p. 410-423 (2006).

[34] E. E. LEWIS and W. F. MILLER, Jr., Computational Methods in Neutron Transport, Wiley, New York, New York (1984).

[35] Y. LIU, et. al., "Resonance Self-Shielding Methodology in MPACT," *Proc. M&C 2013*, Sun Valley, ID (May 5-9, 2013).

[36] Z. LIU, et. al., "A New Three-Dimensional Method of Characteristics for the Neutron Transport Calculation," *Annals of Nuclear Energy*, Vol. 38, p. 447-454 (2011).

[37] A. G. NELSON. "Improved Convergence Rate of Multi-Group Scattering Moment Tallies for Monte Carlo Neutron Transport Codes," Ph.D. Dissertation, University of Michigan (2014).

[38] A. MARIN-LAFLECHE, et. al. "Development Status of PREOTEUS-MOC", ANL/NE-12/46, (2012).

[39] R. G. MCCLARREN, "Theoretical Aspects of the Simplified Pn Equations." *Transport Theory and Statistical Physics*, Vol. 39 (2-4), p. 73-109 (2011).

[40] MPACT Team, "MPACT Theory Manual, Version 1.0," University of Michigan, Ann Arbor, MI (October 2013).

[41] Message Passing Interface Forum, "MPI: A Message-Passing Interface Standard Version 2.2, http://www.mpi-forum.org/docs/mpi-2.2/ (September 2009).

[42] M. RYU, et. al. "Solution of the BEAVRS Benchmark Using the nTRACER Direct Whole Core Transport Code," *Proc. PHYSOR 2014*, Kyoto, Japan (September 28 - October 3, 2014).

[43] Oak Ridge Leadership Computing Facility, "Introducting Titan - The World's #1 Open Science Supercomputer, http://www.olcf.ornl.gov/titan/ (2013).

[44] K. O. OTT and R. J. NEUHOLD, Introductory Nuclear Reactor Dynamics, American Nuclear Society (1985).

[45] OpenMP Architecture Review Board, "OpenMP Application Program Interface Version 3.1, http://openmp.org/wp/openmp-specifications/ (July 2011).

[46] W. D. POINTER, et. al., "Steady-State, Whole-Core VHTR Simulation with Consistent Coupling of Neutronics and Thermo-fluid Analysis," ANL-GenIV-121 (2009).

[47] A. K. PRINJA and E. W. LARSEN, "Principles of Neutron Transport." Handbook of Nuclear Engineering, D.G. Cacuci (Ed.), p. 427-542 (2010).

[48] P. K. ROMANO and B. FORGET, "The OpenMC Monte Carlo particle transport code," *Annals of Nuclear Energy*, Vol. 51, p. 274-281 (2013).

[49] Y. S. Jung, et. al., "Practical numerical reactor employing direction whole core neutron transport and subchannel thermal/hydraulic solvers", *Annals of Nuclear Energy*, Vol. 62, p. 357-374 (2013).

[50] Y. S. Jung, and H. G. JOO, "Direct Whole Core Calculation with Thermal Feedback Using Planar MOC Generated Cross Sections Functions". *M&C 2011*, Rio de Janeiro, Brazil (2011).

[51] M. A. SMITH, et. al., "Benchmark on Deterministic Transport Calculations Without Spatial Homogenization," Nuclear Energy Agency Organisation for Economic Co-operation and Development (NEA-OECD) (2003).

[52] M. A. SMITH, et. al., "Benchmark on Deterministic Transport Calculations Without Spatial Homogenization: MOX Fuel Assembly 3-D Extension Case," Nuclear Energy Agency Organisation for Economic Co-operation and Development (NEA-OECD) (2005).

[53] S. G. STIMPSON, et. al., "Boundary Acceleration Techniques for CMFD-Accelerated 2D-MOC," *Proc. PHYSOR 2014*, Kyoto, Japan (September 28 - October 3, 2014).

[54] S. G. STIMPSON, B. S. COLLINS, T. J. DOWNAR, "Axial Transport Solvers for the 2D/1D Scheme in MPACT." *Proc. PHYSOR 2014*, Kyoto, Japan (September 28 - October 3, 2014).

[55] S. G. STIMPSON, et. al., "Assessment and Improvement of the 2D/1D Method Stability in DeCART." *Proc. M&C 2013*, Sun Valley, ID (May 5-9, 2013).

[56] T. TAKEDA and H. IKEDA, "Final Report on the 3-D Neutron Transport Benchmarks." Tech. Rep. NEA/NEACRP/L(1990)330, Nuclear Energy Agency Organisation for Economic Co-operation and Development (NEA-OECD) (1991).

[57] C. TANG and S. ZHANG, "Development and Verification of an MOC Code Employing Assembly Modular Ray Tracing and Efficient Acceleration Techniques," *Annals of Nuclear Energy*, Vol. 36, p. 1013-1020 (2009).

[58] J. B. TAYLOR, "The Development of a Three-Dimensional Nuclear Reactor Kinetics Methodology Based on the Method of Characteristics," Ph.D. Dissertation, Pennsylvania State University (2007).

[59] X-5 Monte Carlo Team, "MCNP User Manual, Version 5, LA-UR-03-1987 (February 2008).

[60] A. YAMAMOTO, Y. KITAMURA, and Y. YAMANE, "Simplified Treatments of Anisotropic Scattering in LWR Core Calculations," *Journal of Nuclear Science and Technology*, Vol. 45, No. 3, p. 217-229 (2008).

[61] A. YAMAMOTO, et. al., "Derivation of Optimum Polar Angle Quadrature Set for the Method of Characteristics Based On Approximation Error for the Bickley Function," *Journal of Nuclear Science and Technology*, Vol. 4, No. 2, p. 129-136 (2007).

[62] J. I. YOON and H. G. JOO, "Two-Level Coarse Mesh Finite Difference Formulation with Multigroup Source Expansion Nodal Kernels," *Journal of Nuclear Science and Technology*, Vol. 45, No. 7, p. 668-682 (2008).

[63] M. T. H. YOUNG, B. S. COLLINS, and W. R. MARTIN, "Corrected Diamond Difference Method for Coupling from the Method of Characteristics to Discrete Ordinates". *Proc. PHYSOR 2014*, Kyoto, Japan (September 28 - October 3, 2014).