A Dynamic Control Rod Decusping Method for the 2D/1D Scheme

by

Aaron M. Graham

Ph.D. Prospectus
October, 2016

Doctoral Committee:

Professor Thoms J. Downar, Co-Chair
Adjunct Assistant Professor Benjamin S. Collins, Co-Chair
Professor Edward W. Larsen
Professor William R. Martin
Assistant Professor Shravan Veerapaneni

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

For several decades, it has been typical for a two-step approach to be used in reactor analysis. This approach consisted of high-fidelity calculations on small regions which were used to inform the 3D nodal calculations for the full core problem. This allowed for efficient calculations on limited computing resources. However, the drawback to these methods is that the approximations required to make them fast limited their accuracy. With computing power increasing rapidly in recent years, interest has shifted toward more direct methods for reactor analysis. Performing direct 3D transport calculations is still too burdensome for practical calculations, so other methods have been developed to bridge the gap between the older nodal calculations and direct 3D calculations. One of the more popular methods to do this is the 2D/1D method, which takes advantage of reactor geometry to perform very detailed calculations in the radial direction, then couple those calculations with faster, approximate calculations in the axial direction. This calculation scheme provides much greater detail than the older nodal methods were with less computational burden than direct 3D calculations.

One chronic problem for nodal methods was that of rod cusping. This effect arises from the assumption that each node in the core was axially homogeneous. Reactor control rods may be placed in locations that violate that assumption, requiring the rodded and unrodded materials to be homogenized together. A simple volume homogenization is the simplest way to handle this, but introduces large errors by increasing the amount of absorption in the node. While the 2D/1D method may significantly improve on nodal methods, it still requires homogeneity in the axial direction for each 2D plane. Because of this, rods which are partially inserted into a plane can introduce rod cusping effects when using 2D/1D as with nodal methods. Many different methods have been proposed to address this problem in both nodal methods and 2D/1D. Some of the methods are fast and simple to implement, but only reduce the errors introduced by rod cusping. Others do an excellent job of eliminating the errors, but require additional detailed calculation, increasing the computational expense

of the solution. The work presented here seeks to resolve the rod cusping problem in the 2D/1D code MPACT through new methods which are both fast and accurate.

First, the neutron transport equation will be introduced, along with some of the most important approximations commonly used to solve it. Several important numerical methods will then be introduced, with a focus on those used to solve the rod cusping problem in this work. This will be followed by a more detailed history and description of 2D/1D scheme, which will lead into a discussion of the various control rod decusping methods which have been developed in the past. With this foundation laid, the recent methods development in MPACT will be described, and the results of these methods will be presented. Finally, a new transport-based decusping method will be proposed which builds on the work presented here to address the rod cusping problem in a manner that is efficient and physically consistent with the 2D-1D method.

<center>**CHAPTER 2**</center>

Neutron Transport Methodology

This chapter is dedicated to introducing the neutron transport equation and numerical methods commonly used to solve it. First, the transport equation will be introduced and all its terms defined. Some common approximations that are needed to make the transport equation more practically solvable will also be introduced. Second, numerical methods will be discussed and, when appropriate, derived from the transport equation. The theory is covered in detail in a variety of textbooks [1], and other texts discuss numerical methods to solve the transport equation [2]. This chapter will not attempt to reiterate all that can be found in these references, but rather to highlight the theory and methods which are relevant to this work.

2.1 Boltzmann Equation

The time-dependent Boltzmann equation for neutron transport is shown below:

$$
\frac{1}{v}\frac{\partial \psi}{\partial t} + \mathbf{\Omega} \cdot \mathbf{\nabla}\psi + \Sigma_t(\boldsymbol{x},E,t)\psi(\boldsymbol{x},E,\mathbf{\Omega},t)
$$

$$
= \frac{1}{4\pi}\int_0^\infty \int_{4\pi} \Sigma_s(\boldsymbol{x},E'\to E,\mathbf{\Omega}'\to\mathbf{\Omega})\psi(\boldsymbol{x},E',\mathbf{\Omega}')d\mathbf{\Omega}'dE'
$$

$$
+\frac{\chi_P(\boldsymbol{x},E)}{4\pi}\int_0^\infty \int_{4\pi}(1-\beta(\boldsymbol{x},E'))v\Sigma_f(\boldsymbol{x},E',t)\psi(\boldsymbol{x},E',\mathbf{\Omega}',t)d\mathbf{\Omega}'dE'
$$

$$
+\sum_{j=1}^{N_d}\frac{\chi_{d,j}(\boldsymbol{x},E)}{4\pi}\lambda_j C_j(\boldsymbol{x},t)+Q(\boldsymbol{x},E,\mathbf{\Omega},t)\ ,
$$

<div align="right">(2.1a)</div>

$$
\psi(\boldsymbol{x}_b,E,\mathbf{\Omega},t)=\psi^b(\boldsymbol{x}_b,E,\mathbf{\Omega},t)\ ,\quad \mathbf{\Omega}\cdot\boldsymbol{n}<0\ .
$$

<div align="right">(2.1b)</div>

Before addressing the methods used to solve Equation 2.1, we will briefly define each of the terms in the Equation 2.2. The first term is the time derivative term, shown in , which

<center>3</center>

accounts for the change in the angular flux over time in dV about x, dE about E, and dΩ about $\boldsymbol{\Omega}$.

$$\frac{1}{v}\frac{\partial \psi}{\partial t} \ . \tag{2.2}$$

The second term (Equation 2.3 is the streaming term. This describes neutrons with energy E traveling out of the volume dV in the direction $\boldsymbol{\Omega}$.

$$\boldsymbol{\Omega} \cdot \boldsymbol{\nabla} \psi \ . \tag{2.3}$$

The third term is the total reaction rate (Equation 2.4). This describes the total number of collisions experienced in dV by neutrons with energy E and direction $\boldsymbol{\Omega}$. Equations 2.2-2.4 together make up the total loss of neutrons.

$$\Sigma_t(x, E)\psi(x, E, \boldsymbol{\Omega}) \ . \tag{2.4}$$

Equation 2.5 shows the scattering source written in a simplified form. This is the total number of neutrons scattering into energy E and direction $\boldsymbol{\Omega}$ from all other energies and directions E' and $\boldsymbol{\Omega}'$ in dV. Because scattering is symmetric around the incident angle, the scattering cross-section depends only on the dot product $\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}$ rather than each of the two angles independently.

$$\int\limits_0^\infty \int\limits_{4\pi} \Sigma_s(x, E' \to E, \boldsymbol{\Omega}' \cdot \boldsymbol{\Omega})\psi(x, E', \boldsymbol{\Omega}')\,d\Omega'\,dE' \ . \tag{2.5}$$

Equation 2.6 shows the prompt fission source, neutrons entering dV with energy and direction E and $\boldsymbol{\Omega}$ directly from a fission event. Fission is an isotropic process, so the total fission source is calculated then distributed evenly across 4π. Furthermore, the energy distribution of fission is practically independent of incident neutron energy, so the fission neutron distribution $\chi_p(E)$ can be outside the integral over energy. A small fraction of fission neutrons are considered "delayed," meaning they are emitted by the radioactive decay of a fission product. The prompt fission source must be adjusted by the factor $(1 - \beta(x, E'))$ to account for this. Typically β is less than 1% and different for each fissionable isotope.

$$\frac{\chi_p(x, E)}{4\pi} \int\limits_0^\infty \int\limits_{4\pi} (1 - \beta(x, E'))\nu\Sigma_f(x, E', t)\psi(x, E', \boldsymbol{\Omega}', t)\,d\Omega'\,dE' \ . \tag{2.6}$$

Equation 2.7 shows the fission source due to delayed neutrons. The precursors, fission products which emit delayed neutrons, are divided into N_d groups based on the magni-

4

tude of their decay constant λ_j. Like the prompt neutrons, delayed neutrons are isotropic and distributed in energy with some distribution $\chi_{d,j}(x, E)$ based on which precursors are produced.

$$\sum_{j=1}^{N_d} \frac{\chi_{d,j}(x, E)}{4\pi} \lambda_j C_j(x, t) .$$

(2.7)

Equation 2.8 shows external source term. This term accounts for all neutrons entering dV with energy and direction E and $\boldsymbol{\Omega}$ from sources other than scatter and fission. Equation 2.8 shows the source term as a function of angle, but it is often considered to be isotropic like the fission source.

$$Q(x, E, \boldsymbol{\Omega}) .$$

(2.8)

Finally, Equation 2.1b shows the boundary condition for the transport equation. The dot product $\boldsymbol{\Omega} \cdot \boldsymbol{n}$ is between the direction of flight and the outward normal vector on the boundary of the problem. The angular flux boundary condition defines the angular flux entering the problem domain. for the entire surface x_b, all energies, and all times.

Many problems of interest are steady-state, allowing Equation 2.1 to be simplified significantly. The time derivative becomes 0, eliminating the term in 2.2, and the precursor concentrations are unchanging in time, allowing the fission source terms in 2.6 and 2.7 to be lumped into a single term. To ensure balance between the loss and source terms without a time derivative, the equation is reformulated as an eigenvalue equation. The fission source is multiplied by the eigenvalue $\lambda = \frac{1}{k_{eff}}$, allowing the equation to be balanced. The cross-sections can then be adjusted until $\lambda = 1$ is achieved, providing a physically meaningful steady-state solution to the equation. The steady-state form of the Boltzmann equation is shown below:

$$\boldsymbol{\Omega} \cdot \boldsymbol{\nabla} \psi + \Sigma_t(x, E) \psi(x, E, \boldsymbol{\Omega}) = \frac{1}{4\pi} \int_0^\infty \int_{4\pi} \Sigma_s(x, E' \to E, \boldsymbol{\Omega}' \to \boldsymbol{\Omega}) \psi(x, E', \boldsymbol{\Omega}') d\Omega' dE'$$

$$+ \frac{\chi(E)}{4\pi} \int_0^\infty \int_{4\pi} \nu \Sigma_f(x, E') \psi(x, E', \boldsymbol{\Omega}') d\Omega' dE' + \frac{Q(x, E, \boldsymbol{\Omega})}{4\pi} ,$$

(2.9a)

$$\psi(x_b, E, \boldsymbol{\Omega}) = \psi^b(x_b, E, \boldsymbol{\Omega}) , \quad \boldsymbol{\Omega} \cdot \boldsymbol{n} < 0 .$$

(2.9b)

2.1.1 Multi-group Approximation

One important approximation that is commonly made to the transport equation is the multi-group approximation. To make this approximation, an appropriate energy range for the problem of interest is selected. This energy range is divided up into N energy groups, with each group going from E_g up to E_{g-1}. For light-water reactor problems, it is common to select 0 eV for E_N and 20 MeV for E_0.

We first define the multi-group flux, cross-sections, and chi distribution in Equations 2.10. These definitions ensure that the total reaction rates are preserved in each energy interval g.

$$\psi_g(\boldsymbol{x}, \boldsymbol{\Omega}) = \int_{E_n}^{E_{n-1}} \psi(\boldsymbol{x}, E, \boldsymbol{\Omega}) \, dE , \tag{2.10a}$$

$$\Sigma_{x,g}\psi_g = \int_{E_n}^{E_{n-1}} \psi(E)\Sigma_x(E) \, dE \Rightarrow \Sigma_{x,g} = \frac{\int_{E_n}^{E_{n-1}} \psi(E)\Sigma_x(E) \, .dE}{\int_{E_n}^{E_{n-1}} \psi(E) \, dE} \tag{2.10b}$$

Using the definitions above, Equation 2.9 can be operated on by $\int_{E_n}^{E_{n-1}} (\cdot) \, dE$ to obtain the multi-group transport Equation 2.11.

$$\boldsymbol{\Omega} \cdot \boldsymbol{\nabla}\psi_g + \Sigma_{t,g}(\boldsymbol{x})\psi_g(\boldsymbol{x}, \boldsymbol{\Omega}) = \frac{1}{4\pi} \sum_{g'=1}^{G} \int_{4\pi} \Sigma_{s,g' \to g}(\boldsymbol{x}, \boldsymbol{\Omega}' \to \boldsymbol{\Omega})\psi_{g'}(\boldsymbol{x}, \boldsymbol{\Omega}') \, d\boldsymbol{\Omega}'$$

$$+ \frac{1}{k_{eff}} \frac{\chi_g}{4\pi} \sum_{g'=1}^{G} \int_{4\pi} \nu\Sigma_{f,g'}(\boldsymbol{x})\psi_{g'}(\boldsymbol{x}, \boldsymbol{\Omega}') \, d\boldsymbol{\Omega}' + \frac{Q_g(\boldsymbol{x})}{4\pi} , \tag{2.11a}$$

$$\psi_g(\boldsymbol{x}_b, \boldsymbol{\Omega}) = \int_{E_n}^{E_{n-1}} \psi^b(\boldsymbol{x}_b, E, \boldsymbol{\Omega}) \, dE , \quad \boldsymbol{\Omega} \cdot \boldsymbol{n} < 0 . \tag{2.11b}$$

2.1.2 Discrete Ordinates Approximation

In addition to energy discretization, it is also useful to discretize the transport equation in angle. The angular variable $\boldsymbol{\Omega}$ is made up of a polar angle (μ_s) and an azimuthal angle (α), both of which can be discretized:

$$\boldsymbol{\Omega} = \cos(\alpha)\sqrt{1 - \mu_s^2}\boldsymbol{i} + \sin(\alpha)\sqrt{1 - \mu_s^2}\boldsymbol{j} + \mu_s\boldsymbol{k} \tag{2.12a}$$

$$\Rightarrow \boldsymbol{\Omega}_n = \cos(\alpha_n)\sqrt{1 - \mu_{s,n}^2}\boldsymbol{i} + \sin(\alpha_n)\sqrt{1 - \mu_{s,n}^2}\boldsymbol{j} + \mu_{s,n}\boldsymbol{k} . \tag{2.12b}$$

For each $\mathbf{\Omega}_n$, there is an associated weight w_n. These weights and angles together make up an angular quadrature set which simplifies the integrals in Equations 2.13.

$$\int d\Omega = \sum_{n=1}^{N} w_n = 4\pi \, , \tag{2.13a}$$

$$\int \mathbf{\Omega} d\Omega = \sum_{n=1}^{N} \mathbf{\Omega}_n w_n = 0 \, , \tag{2.13b}$$

$$\int_{4\pi} f(\mathbf{\Omega}) d\Omega \approx \sum_{n=1}^{N} f_n w_n \, . \tag{2.13c}$$

Applying this discretization to the multi-group transport Equation 2.11, we obtain the following discrete ordinates (S_N) equations:

$$\mathbf{\Omega}_n \cdot \nabla \psi_{g,n} + \Sigma_{t,g}(\mathbf{x}) \psi_{g,n}(\mathbf{x}) = \frac{1}{4\pi} \sum_{g'=1}^{G} \sum_{n'=1}^{N} \Sigma_{g' \to g, n' \to n}(\mathbf{x}) \psi_{g',n'}(\mathbf{x}) w_{n'}$$

$$+ \frac{1}{k_{eff}} \frac{\chi_g}{4\pi} \sum_{g'=1}^{G} \sum_{n'=1}^{N} \nu \Sigma_{f,g'}(\mathbf{x}) \psi_{g',n'}(\mathbf{x}) w_{n'} + \frac{Q_{g,n}(\mathbf{x})}{4\pi} \, , \tag{2.14a}$$

$$\psi_{g,n}(\mathbf{x}_b) = \psi_g^b(\mathbf{x}_b, \mathbf{\Omega}_n) \, , \quad \mathbf{\Omega}_n \cdot \mathbf{n} < 0 \, . \tag{2.14b}$$

2.1.3   Scattering Approximations

One of the biggest challenges to solving the transport equation is angle dependence of the scattering cross-sections and angular flux. To simplify the scattering cross-sections, there are two different types of approximations that can be made in MPACT.

2.1.3.1 P_N Scattering

The first scattering approximation that can be made is P_N scattering. To make this approximation, the $\mathbf{\Omega}' \cdot \mathbf{\Omega}$ is rewritten as a single angular variable μ_s, the cosine between the incoming and outgoing scattering angles. The scattering cross-section can then be expanded in terms of Legendre polynomials, defined by Equations 2.15.

$$P_{n+1}(\mu_s) = \frac{(2n+1)\mu_s P_n(\mu_s) - n P_{n-1}(\mu_s)}{n+1} \, , \tag{2.15a}$$

$$P_0(\mu_s) = 1 \, , \tag{2.15b}$$

$$P_1(\mu_s) = \mu_s \,, \tag{2.15c}$$

$$\int_{-1}^{1} P_n(\mu_s) P_m(\mu_s) d\mu_s = \frac{2}{2n+1} \delta_{n,m} \,. \tag{2.15d}$$

Equations 2.16 show the expansion of the scattering cross-section using Legendre polynomials. Using more terms in the expansion improves the accuracy. For most reactor problems, $N \leq 3$ is sufficient. Problems such as shielding and others may require many more terms to be kept to obtain sufficient accuracy.

$$\Sigma_s(\boldsymbol{x}, \mu_s) = \sum_{n=0}^{N} \frac{2n+1}{4\pi} P_n(\mu_s) \Sigma_{sn}(\boldsymbol{x}) \,, \tag{2.16a}$$

$$\Sigma_{s,n}(\boldsymbol{x}) = 2\pi \int_{-1}^{1} \Sigma_s(\boldsymbol{x}, \mu_s) P_n(\mu_s) d\mu_s \,. \tag{2.16b}$$

2.1.3.2 Transport Correction

A second simplification of the scattering source that can be used is transport-corrected isotropic (TCP_0) scattering. When using this approximation, the equation is solved using only the zeroth order term in 2.16. The cross-section data used to develop the multi-group scattering cross-sections is modified beforehand to still preserve some of the higher order scattering physics.

Currently, the MPACT code uses [3] uses a combination of three transport-correction methods for its TCP_0 scattering treatments [4, 5]. The Neutron Leakage Conservation (NLC) method [6] is used for H-1; the nTRACER in-scatter method [7] is used for B-11, C-12, and O-16; and the out-scatter method [8] is used for all remaining isotopes. The out-scatter approximation is defined as

$$\Sigma_{s0,g \to g} = \Sigma_{s0,g \to g} - \sum_{g'=1}^{G} \Sigma_{s1,g \to g'} \,. \tag{2.17}$$

The in-scatter approximation is defined as

$$\Sigma_{s0,g \to g} = \Sigma_{s0,g \to g} - \frac{1}{\phi_{1,g}} \sum_{g'=1}^{G} \Sigma_{s1,g' \to g} \phi_{1,g'} \,, \tag{2.18}$$

where $\phi_{1,g}$ is the first moment of the angular flux. Finally, the NLC correction is defined as

follows:

$$\Sigma_{s0,g \to g} = \Sigma_{s0,g \to g} + \frac{1}{3D_g} - \Sigma_{t,g} \; , \tag{2.19}$$

where D_g is a diffusion coefficient determined from leakage calculations in a 1D fixed-source transport calculation. For each of these corrections, the same modification made to the self-scatter cross-sections in the above equations is also made to the total cross-section $\Sigma_{t,g}$ to obtain the transport cross-section $\Sigma_{tr,g}$.

2.1.4 Diffusion Approximation

Another approximation that can be made to the angular dependence of the transport equation is the diffusion approximation. To derive this approximation, we again begin with the multi-group transport equation from Equation 2.11. First, we obtain the P_1 form of the equation by operating on it by $\int (\cdot) d\Omega$ and $\int (\cdot) \Omega_i d\Omega$. To simplify these integrals, we make use of the following identities for integrating over $\boldsymbol{\Omega}$:

$$\int_{4\pi} d\Omega = 4\pi \; , \tag{2.20a}$$

$$\int_{4\pi} \Omega_i d\Omega = 0 \; , \tag{2.20b}$$

$$\int_{4\pi} \Omega_i \Omega_j d\Omega = \frac{4\pi}{3} \delta_{i,j} \; , \tag{2.20c}$$

$$\int_{4\pi} \Omega_i \Omega_j \Omega_k d\Omega = 0 \; . \tag{2.20d}$$

We must also assume that the angular flux is linearly anisotropic. This leads to the following expression of the angular flux as a function of the scalar flux ϕ and current J:

$$\psi_g(\boldsymbol{x}, \boldsymbol{\Omega}) \approx \frac{1}{4\pi} \left(\phi_g(\boldsymbol{x}) + 3\boldsymbol{\Omega} \cdot \boldsymbol{J}(\boldsymbol{x}) \right) \; , \tag{2.21a}$$

$$\phi_g(\boldsymbol{x}) = \int_{4\pi} \psi_g(\boldsymbol{x}, \boldsymbol{\Omega}) d\Omega \; , \tag{2.21b}$$

$$J_g(\boldsymbol{x}) = \int_{4\pi} \psi_g(\boldsymbol{x}, \boldsymbol{\Omega}) \boldsymbol{\Omega} d\Omega \; . \tag{2.21c}$$

9

Applying this assumption and integrating, we obtain a coupled set of four equations for the scalar flux and the three components of the current vector.

$$
\frac{dJ_{x,g}}{dx} + \frac{dJ_{y,g}}{dy} + \frac{dJ_{z,g}}{dz} + \Sigma_{t,g}(\boldsymbol{x})\phi_g(\boldsymbol{x}) = \sum_{g'=1}^{G} \Sigma_{s0,g'\to g}(\boldsymbol{x})\phi_{g'}(\boldsymbol{x})
$$

$$
+ \frac{1}{k_{eff}}\frac{\chi_g}{4\pi}\sum_{g'=1}^{G}\nu\Sigma_{f,g'}(\boldsymbol{x})\phi_{g'}(\boldsymbol{x}) + Q_g(\boldsymbol{x}) \ ,
$$

(2.22a)

$$
\frac{d\phi_g}{dx} + \Sigma_{t,g}(\boldsymbol{x})J_{x,g}(\boldsymbol{x}) = \sum_{g'=1}^{G}\Sigma_{s1,g'\to g}(\boldsymbol{x})J_{x,g'}(\boldsymbol{x}) \ ,
\tag{2.22b}
$$

$$
\frac{d\phi_g}{dy} + \Sigma_{t,g}(y)J_{y,g}(\boldsymbol{x}) = \sum_{g'=1}^{G}\Sigma_{s1,g'\to g}(\boldsymbol{x})J_{y,g'}(\boldsymbol{x}) \ ,
\tag{2.22c}
$$

$$
\frac{d\phi_g}{dz} + \Sigma_{t,g}(\boldsymbol{x})J_{z,g}(\boldsymbol{x}) = \sum_{g'=1}^{G}\Sigma_{s1,g'\to g}(\boldsymbol{x})J_{z,g'}(\boldsymbol{x}) \ .
\tag{2.22d}
$$

Solving Equations 2.22b-2.22d for the components of the current, we obtain a relationship between the current and scalar flux, shown in Equation 2.23. Equation 2.22 naturally leads to the out-scatter approximation for the transport cross-section, but either of the other methods from section 2.1.3.2 can be used instead.

$$
\boldsymbol{J}_g(\boldsymbol{x}) = -\boldsymbol{D}_g(\boldsymbol{x})\boldsymbol{\nabla}\phi_g(\boldsymbol{x}) \ ,
\tag{2.23a}
$$

$$
\boldsymbol{D}_g(\boldsymbol{x}) = \frac{1}{3}\left(\Sigma_{tr,g}\right)^{-1} \ .
\tag{2.23b}
$$

Substituting 2.23a into 2.22a gives us the diffusion form of the transport equation, which has only the scalar flux ϕ and eigenvalue $\frac{1}{k_{eff}}$ as unknowns:

$$
-\boldsymbol{\nabla} \cdot \boldsymbol{D}_g(\boldsymbol{x})\boldsymbol{\nabla}\phi_g(\boldsymbol{x}) + \Sigma_{t,g}(\boldsymbol{x})\phi_g(\boldsymbol{x}) = \sum_{g'=1}^{G}\Sigma_{s0,g'\to g}(\boldsymbol{x})\phi_{g'}(\boldsymbol{x})
$$

$$
+ \frac{1}{k_{eff}}\frac{\chi_g}{4\pi}\sum_{g'=1}^{G}\nu\Sigma_{f,g'}(\boldsymbol{x})\phi_{g'}(\boldsymbol{x}) + Q_g(\boldsymbol{x}) \ .
$$

(2.24)

To derive boundary conditions, Equation 2.11b is used, making the same linear anisotropy assumption as in the derivation of the diffusion equation itself:

$$\int\limits_{\boldsymbol{\Omega}\cdot\boldsymbol{n}<0} \left|\boldsymbol{\Omega}\cdot\boldsymbol{n}\right|\psi_g(\boldsymbol{x}_b,\boldsymbol{\Omega})\,d\Omega = \int\limits_{\boldsymbol{\Omega}\cdot\boldsymbol{n}<0} \left|\boldsymbol{\Omega}\cdot\boldsymbol{n}\right|\psi_g^b(\boldsymbol{x}_b,\boldsymbol{\Omega})\,d\Omega$$

$$\int\limits_{\boldsymbol{\Omega}\cdot\boldsymbol{n}<0} \left|\boldsymbol{\Omega}\cdot\boldsymbol{n}\right|\frac{1}{4\pi}\left[\phi_g(\boldsymbol{x}_b)+3\boldsymbol{\Omega}\cdot\boldsymbol{J}_g(\boldsymbol{x}_b)\right]d\Omega = J^-(\boldsymbol{x}_b)$$

$$-\frac{1}{4\pi}\int\limits_{\boldsymbol{\Omega}\cdot\boldsymbol{n}<0} \boldsymbol{\Omega}\cdot\boldsymbol{n}\left[\phi_g(\boldsymbol{x}_b)+3\boldsymbol{\Omega}\cdot\boldsymbol{J}_g(\boldsymbol{x}_b)\right]d\Omega = J^-(\boldsymbol{x}_b)$$

$$\frac{1}{4}\phi_g(\boldsymbol{x}_b)-\frac{1}{2}\boldsymbol{n}\cdot\boldsymbol{J}(\boldsymbol{x}_b) = J^-(\boldsymbol{x}_b)\;. \tag{2.25}$$

This is the Marshak boundary condition, which preserves the total incident flux at every point on the boundary of the system while assuming linear anisotropy. This boundary condition can be further simplified so that the only unknown is the scalar flux ϕ_g by substituting Equation 2.23a:

$$\frac{1}{4}\phi_g(\boldsymbol{x}_b)+\frac{\boldsymbol{D}_g(\boldsymbol{x})}{2}\cdot\boldsymbol{\nabla}\phi_g(\boldsymbol{x}_b) = J^-(\boldsymbol{x}_b)\;. \tag{2.26}$$

2.2 Numerical Methods

This section will present some of basic numerical methods used to solve the transport equation. There are many different numerical methods that can be used, but only those which are important to this work will be described here. For the most extensively used ones, detailed descriptions or derivations will be included. Others will simply be described at a high level with details left to the appendices and references

2.2.1 Method of Characteristics

One method that is commonly used to solve the transport equation is the Method of Characteristics (MOC) [9, 10]. This method allows the transport equation to be solved along a characteristic direction with limited approximations, making it useful for complicated geometries such as those found in nuclear reactors. Doing this along many lines for a variety of angles and integrating allows an accurate calculation of the scalar flux for large, geometrically complex problems, making MOC useful for many reactor calculations. This section focuses on a detailed derivation of MOC.

To derive MOC, we make use of both the multi-group and discrete ordinates approximations by beginning with Equation 2.14. First, the right-hand side is lumped into a single

source term for a given energy group g and direction n:

$$\mathbf{\Omega}_n \cdot \nabla \psi_{g,n} + \Sigma_{t,g}(\mathbf{x})\psi_{g,n}(\mathbf{x}) = q_{g,n}(\mathbf{x}) , \tag{2.27a}$$

$$q_{g,n}(\mathbf{x}) = \frac{1}{4\pi} \sum_{g'=1}^{G} \sum_{n'=1}^{N} \Sigma_{s,g' \to g,n' \to n}(\mathbf{x})\psi_{g',n'}(\mathbf{x})w_{n'}$$

$$+ \frac{1}{k_{eff}} \frac{\chi_g}{4\pi} \sum_{g'=1}^{G} \sum_{n'=1}^{N} \nu\Sigma_{f,g'}(\mathbf{x})\psi_{g',n'}(\mathbf{x})w_{n'} + \frac{Q_{g,n}(\mathbf{x})}{4\pi} . \tag{2.27b}$$

Now we introduce a characteristic direction $\mathbf{r}$ in the direction $\mathbf{\Omega}_n$

$$\mathbf{r} = \mathbf{r_0} + s\mathbf{\Omega}_n \Rightarrow \begin{cases} x(s) = x_0 + s\Omega_{n,x} \\ y(s) = y_0 + s\Omega_{n,y} \\ z(s) = z_0 + s\Omega_{n,z} \end{cases} . \tag{2.28}$$

Substituting this variable into Equation 2.27 gives the characteristic form of this equation:

$$\frac{\partial \psi_{g,n}}{\partial s} + \Sigma_{t,g}(\mathbf{r}_0 + s\mathbf{\Omega}_n)\psi_{g,n}(\mathbf{r}_0 + s\mathbf{\Omega}_n) = q_{g,n}(\mathbf{r}_0 + s\mathbf{\Omega}_n) , \tag{2.29a}$$

$$q_{g,n}(\mathbf{r}_0 + s\mathbf{\Omega}_n) = \frac{1}{4\pi} \sum_{g'=1}^{G} \sum_{n'=1}^{N} \Sigma_{s,g' \to g,n' \to n}(\mathbf{r}_0 + s\mathbf{\Omega}_n)\psi_{g',n'}(\mathbf{r}_0 + s\mathbf{\Omega}_n)w_{n'}$$

$$+ \frac{1}{k_{eff}} \frac{\chi_g}{4\pi} \sum_{g'=1}^{G} \sum_{n'=1}^{N} \nu\Sigma_{f,g'}(\mathbf{r}_0 + s\mathbf{\Omega}_n)\psi_{g',n'}(\mathbf{r}_0 + s\mathbf{\Omega}_n)w_{n'} + Q_{g,n}(\mathbf{r}_0 + s\mathbf{\Omega}_n) . \tag{2.29b}$$

This equation is easily solved with the integrating factor

$$\exp\left(-\int_0^s \Sigma_{t,g}(\mathbf{r}_0 + s'\mathbf{\Omega}_n)ds'\right) \tag{2.30}$$

giving the following solution:

$$\psi_{g,n}(\mathbf{r}_0 + s\mathbf{\Omega}_n) = \psi_{g,n}(\mathbf{r}_0)\exp\left(-\int_0^s \Sigma_{t,g}(\mathbf{r}_0 + s'\mathbf{\Omega}_n)ds'\right)$$

$$+ \int_0^s q_{g,n}(\mathbf{r}_0 + s'\mathbf{\Omega}_n)\exp\left(-\int_0^{s'} \Sigma_{t,g}(\mathbf{r}_0 + s''\mathbf{\Omega}_n)ds''\right)ds' . \tag{2.31}$$

Now if $\boldsymbol{r}_0$ is on the boundary of the region of interest, the incoming flux $\psi(\boldsymbol{r}_0, E, \boldsymbol{\Omega})$ is equal to the specified boundary condition ψ^{in}. This allows the angular flux to be calculated at any point s to be calculated along the characteristic direction $\boldsymbol{r}$.

Up to this point, no approximations have been made in deriving the method of characteristics. One approximation that is necessary to use MOC for practical problems is to assume some spatial shape for the source term q. The simplest of these is the flat source approximation. In this approximation, q is assumed to be flat along $\boldsymbol{r}$. Additionally, it is assumed that the cross-sections along each segment of the ray will be constant. This simplifies Equation 2.31 to the following:

$$\psi_{g,n}(\boldsymbol{r}_0 + s\boldsymbol{\Omega}_n) = \psi_{g,n}(\boldsymbol{r}_0)e^{-\int_0^s \Sigma_{t,g}ds'} + \int_0^s q_{g,n}e^{-\int_0^{s'} \Sigma_{t,g}ds''}ds'$$

$$= \psi^{in}e^{-\Sigma_{t,g}s} + \int_0^s q_{g,n}e^{-\Sigma_{t,g}s'}ds'$$

$$= \psi^{in}e^{-\Sigma_{t,g}s} + \frac{q_{g,n}}{\Sigma_{t,g}}\left(1 - e^{-\Sigma_{t,g}s}\right) . \tag{2.32}$$

For a segment of length A beginning on the boundary of region i, traveling in direction n along track j, Equation 2.32 allows us to solve for the outgoing angular flux at the end of track j (2.33a), as well as the average angular flux along the characteristic direction for a specific track j (2.33b).

$$\psi^{out}_{g,i,n,j} = \psi_g\left(\boldsymbol{r}_0 + A_j\boldsymbol{\Omega}_n\right) = \psi^{in}_{g,i,n,j}e^{-\Sigma_{t,g,i}A_j} + \frac{q_{g,i,n}}{\Sigma_{t,g,i}}\left(1 - e^{-\Sigma_{t,g,i}A_j}\right) , \tag{2.33a}$$

$$\overline{\psi}_{g,i,n,j} = \frac{\int_0^{A_j} \psi_g(\boldsymbol{r}_0 + s\boldsymbol{\Omega}_n)ds}{\int_0^{A_j} ds}$$

$$= \frac{1}{A_j}\int_0^{A_j} \psi^{in}_{g,i,n,j}e^{-\Sigma_{t,g,i}s} + \frac{q_{g,n,i}}{\Sigma_{t,g,i}}\left(1 - e^{-\Sigma_{t,g,i}s}\right)ds$$

$$= \frac{1}{A_j}\int_0^{A_j} \frac{q_{g,n,i}}{\Sigma_{t,g,i}} + \left(\psi^{in}_{g,i,n,j} - \frac{q_{g,n,i}}{\Sigma_{t,g,i}}\right)e^{-\Sigma_{t,g,i}s}ds$$

$$= \frac{q_{g,n,i}}{\Sigma_{t,g,i}} + \frac{1}{A_j\Sigma_{t,g,i}}\left(\psi^{in}_{g,i,n,j} - \frac{q_{g,n,i}}{\Sigma_{t,g,i}}\right)\left(1 - e^{-\Sigma_{t,g,i}A_j}\right) . \tag{2.33b}$$

If the transport cross-section is constant along s for the region of interest, then the only

13

approximation that has been made in this derivation is that the source is constant. It is possible to assume higher order shapes for the source (linear, quadratic, etc.) to improve the accuracy of MOC [11], but these will not be discussed here.

Typically multiple tracks are laid down across a region in each direction and integrated, with some spacing δx between them. To calculate the region average angular flux for region i, the average angular flux from Equation 2.33b for each track must be area-averaged, as shown below:

$$\overline{\psi}_{g,i,n} = \frac{\sum_j \overline{\psi}_{g,i,n,j} \delta x A_j}{\sum_j \delta x A_j} \ . \tag{2.34}$$

Using this, a quadrature can be used to integrate the average region angular flux for each angle to obtain a region-averaged angular flux.

Using the outgoing flux in Equation 2.33a along the edge of a region boundary as the incoming angle flux along a neighboring region allows MOC to be used to solve across entire domains regardless of geometry. Doing this in many directions and applying Equations 2.33b and 2.34.

2.2.2 Coarse-Mesh Finite Difference

Because transport calculations can take many iterations to solve, it is important to accelerate the convergence of the iteration scheme when possible. The most common way of doing this is the Coarse-Mesh Finite Difference (CMFD) method [12]. This method involves solving a diffusion problem on a coarse grid to get the average magnitude of the flux in each coarse cell. This is then used to scale the flux solution on the fine grid. In order to ensure consistency between the transport solution and the lower order diffusion solution, coupling coefficients are calculated on the boundaries of the coarse mesh cells to correct the leakage between the cells.

To describe the CMFD method, the scalar flux, current, and cross-sections on the coarse grid must be related to the same quantities on the fine grid. We assume the fine mesh has already been spatially discretized and solved so that quantities such as volume, cell-averaged flux, cell-averaged cross-sections, and cell-averaged source terms are known for each fine cell. With this assumption, we can define the needed coarse mesh values for each coarse cell i in terms of the fine cells it owns. N_f is the total number of fine cells owned by a coarse cell, (x_i, y_i, z_i) are the coordinates for the center of coarse cell i, and $x_{i+\frac{1}{2}}$, $x_{i-\frac{1}{2}}$,

$y_{i+\frac{1}{2}}$, $y_{i-\frac{1}{2}}$, $z_{i+\frac{1}{2}}$, and $z_{i-\frac{1}{2}}$ define the edges of the coarse cell in each direction.

$$A_{x,i} = \left(y_{i+\frac{1}{2}} - y_{i-\frac{1}{2}}\right)\left(z_{i+\frac{1}{2}} - z_{i-\frac{1}{2}}\right), \tag{2.35a}$$

$$A_{y,i} = \left(x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}\right)\left(z_{i+\frac{1}{2}} - z_{i-\frac{1}{2}}\right), \tag{2.35b}$$

$$A_{z,i} = \left(x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}\right)\left(y_{i+\frac{1}{2}} - y_{i-\frac{1}{2}}\right), \tag{2.35c}$$

$$V_i = \sum_{j=1}^{N_f} V_j, \tag{2.35d}$$

$$\phi_{g,i} = \frac{1}{V_i} \sum_{j=1}^{N_f} \phi_{g,j} V_j, \tag{2.35e}$$

$$\Sigma_{t,g,i} = \frac{1}{\phi_{g,i} V_i} \sum_{j=1}^{N_f} \Sigma_{tr,g,j} \phi_{g,j} V_j, \tag{2.35f}$$

$$\Sigma_{s0,g' \to g,i} = \frac{1}{\phi_{g',i} V_i} \sum_{j=1}^{N_f} \Sigma_{s0,g' \to g,j} \phi_{g',j} V_j, \tag{2.35g}$$

$$\nu\Sigma_{f,g,i} = \frac{1}{\phi_{g,i} V_i} \sum_{j=1}^{N_f} \nu\Sigma_{f,g,j} \phi_{g,j} V_j, \tag{2.35h}$$

$$\chi_{g,i} = \frac{\sum_{j=1}^{N_f} \chi_{g,j} \sum_{g'=1}^{G} \nu\Sigma_{f,g',j} \phi_{g',j} V_j}{\sum_{j=1}^{N_f} \sum_{g'=1}^{G} \nu\Sigma_{f,g',j} \phi_{g',j} V_j}, \tag{2.35i}$$

$$Q_{g,i} = \frac{1}{V_i} \sum_{j=1}^{N_f} Q_{g,j} V_j. \tag{2.35j}$$

Furthermore, currents must be tallied on the coarse mesh cell boundaries by the fine mesh transport calculations. Using these definitions, we now operate on the multi-group diffusion equation (2.22a) by Equation 2.36a to obtain Equation 2.36b.

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{i-\frac{1}{2}}}^{y_{i+\frac{1}{2}}} \int_{z_{i-\frac{1}{2}}}^{z_{i+\frac{1}{2}}} (\cdot)\, dz\, dy\, dz, \tag{2.36a}$$

$$A_{x,i}\left(J_{x,g}\left(x_{i+\frac{1}{2}}\right) - J_{x,g}\left(x_{i-\frac{1}{2}}\right)\right) + A_{y,i}\left(J_{y,g}\left(y_{i+\frac{1}{2}}\right) - J_{y,g}\left(y_{i-\frac{1}{2}}\right)\right) + A_{z,i}\left(J_{z,g}\left(z_{i+\frac{1}{2}}\right) - J_{z,g}\right.$$

$$\left.\left(z_{i-\frac{1}{2}}\right)\right) + \Sigma_{tr,g,i} \phi_{g,i} V_i = \sum_{g'=1}^{G} \Sigma_{s0,g' \to g,i} \phi_{g',i} V_i + \frac{1}{k_{eff}} \frac{\chi_{g,i}}{4\pi} \sum_{g'=1}^{G} \nu\Sigma_{f,g',i} \phi_{g',i} V_i + Q_{g,i} V_i. \tag{2.36b}$$

15

Using Equations 2.23, we can define the interface currents in terms of the diffusion constants on the positive (p) and negative (n) sides of the interface (Equation 2.37a. The interface diffusion coefficients are defined in 2.37b-2.37d.

$$J_g = -D_g \left(\phi_{g,p} - \phi_{g,n} \right) , \tag{2.37a}$$

$$D_g = \frac{1}{3} \left(\left(\Delta x_p \Sigma_{tr,g,i+1} + \Delta x_n \Sigma_{tr,g,i} \right) \right)^{-1} , \tag{2.37b}$$

$$D_{g,\frac{1}{2}} = \alpha \left(1 + 3 \Delta x_p \alpha \Sigma_{tr,g,1} \right)^{-1} , \tag{2.37c}$$

$$D_{g,N+\frac{1}{2}} = \alpha \left(3 \Delta x_n \alpha \Sigma_{tr,g,N} + 1 \right)^{-1} , \tag{2.37d}$$

$$\alpha = \begin{cases} 0 & , \text{ reflecting }, \\ 0.5 & , \text{ vacuum }. \end{cases} \tag{2.37e}$$

Defining the interface currents as in Equation 2.37a will cause inconsistency between the high-order transport solver and the diffusion-based CMFD solver. To account for this, a correction factor can be defined in terms of the transport solution and added to the CMFD currents to enforce consistency between the two solutions.

$$J_{CMFD,g} = -D_g \left(\phi_{g,p} - \phi_{g,n} \right) + \hat{D}_g \left(\phi_{g,p} \phi g, n \right) , \tag{2.38a}$$

$$\hat{D}_g = \frac{J_{trans,g} + D_g \left(\phi_{g,p} - \phi g, n \right)}{\left(\phi_{g,p} \phi g, n \right)} . \tag{2.38b}$$

Using this definition of the current, all terms in Equation 2.36b are defined. This gives a system of $N \times G$ equations, where N is the number of coarse nodes and G is the number of energy groups in the problem, which can be written in matrix form and solved as an eigenvalue problem:

$$\underline{\underline{M}} \phi = \frac{1}{k} \underline{\underline{F}} \phi . \tag{2.39}$$

Using the homogenized transport solution as an initial guess on the right-hand side of Equation 2.39, this equation can be easily solved using a standard eigenvalue solver. MPACT generally uses either power iteration or generalized Davidson methods to solve this problem, but these methods will not be discussed in detail here.

After obtaining a solution Equation 2.39, the solution must be used to update the fine mesh solution for the next transport calculation. To do this, a simple scaling is applied to the previous transport solution as shown in Equation 2.40. This scaling is applied to all fine mesh cells j in each coarse mesh cell i and ensures preservation of reaction rates between

the CMFD and transport solutions prior to the next transport sweep.

$$\phi_{trans,g,j}^{k} = \frac{\phi_{CMFD,g,i}^{k}}{\phi_{CMFD,g,i}^{k-1}} \phi_{trans,g,j}^{k-1} \, . \tag{2.40}$$

Performing the next transport calculation and continuing to iterate between the fine mesh transport calculations and CMFD will provide a converged solution far more quickly than performing iterations consisting only of fine mesh transport calculations.

2.2.3 Nodal Methods

2.2.3.1 Simplified Pn

One method that can be used to solve the transport equation is the Spherical Harmonics (P_n) method [13]. This method is able to resolve the angle dependence of the angular flux by expanding the flux using spherical harmonic functions. These functions are defined in Equation 2.41a, with the Associated Legendre Functions and Legendre Polynomials defined in Equations 2.41b and 2.15, respectively.

$$Y_n^m (\Omega) = \left[\frac{2n+1}{4\pi} \frac{(n-|m|)!}{(n+|m|)!} \right]^{\frac{1}{2}} P_n^{|m|} (\mu) e^{im\gamma}, \quad 0 \le |m| \le n < \infty \, , \tag{2.41a}$$

$$P_n^m (\mu) = \left(1-\mu^2\right)^{\frac{m}{2}} \frac{d}{d\mu}^m P_n (\mu), \quad 0 \le m \le n < \infty \, . \tag{2.41b}$$

The angular flux can then be expanded in terms of the spherical harmonic functions, as shown in Equation 2.42. When using the P_n approximation, some integer N is specified. The first summation in Equation 2.42 is then truncated at N. This gives $(N+1)^2 G$ expansion coefficients that must be determined. To solve for these coefficients, the transport equation must be multiplied by $Y_n^{-m} (\Omega)$ and integrated over Ω for each combination of m, n, and g.

$$\psi_g (x, \Omega) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \psi_{n,m,g} (x) Y_n^m (\Omega) \, . \tag{2.42}$$

While the P_n method can be used to solve the transport equation, this is rarely done in practice due to the increasing complexity of the equations and the quadratic growth in the number of unknowns as N increases. Instead, the more common method used in practice is the Simplified Spherical Harmonics (SP$_n$) method. In 1D planar geometry, the angular variable γ vanishes, leaving only μ. In this case, only the spherical harmonic functions for $m = 0$ are needed, since these are the functions independent of γ. Thus, the flux expansion

reduces to a Legendre polynomial expansion:

$$\psi_g(x,\mu) = \sum_{m=0}^{\infty} \frac{2m+1}{2} \psi_{m,g}(x) P_m(\mu) \ , \tag{2.43a}$$

$$\psi_{m,g}(x) = \int_{-1}^{1} P_m(\mu') \psi_g(x,\mu') d\mu' \ . \tag{2.43b}$$

To solve, we apply the expansion in Equation 2.43 to the transport equation, multiply by Legendre polynomials $P_n(\mu)$ for $0 \leq n \leq N$, and integrate over μ. Following this procedure results in the 1D planar geometry P_1 equations, shown in Equation 2.44:

$$\frac{d\psi_{1,g}}{dx} + \Sigma_{t,g}(x) \psi_{0,g}(x) = \sum_{g'=1}^{G} \sigma_{s0,g' \to g}(x) \psi_{0,g'}(x) + \frac{1}{k_{eff}} \frac{\chi_g}{4\pi} \sum_{g'=1}^{G} \nu \Sigma_{f,g'}(x) \psi_{0,g'}(x) + Q_g(x) \ , \tag{2.44a}$$

$$\frac{d\psi_{0,g}}{dx} + \Sigma_{t,g}(x) \psi_{1,g}(x) = \sum_{g'=1}^{G} \Sigma_{s1,g' \to g}(x) \psi_{1,g'}(x) \ . \tag{2.44b}$$

It is clear that if $\frac{d}{dx}$ in Equation 2.44 is replaced by $\boldsymbol{\nabla}$, then the 3D P_1 equations are obtained exactly as shown in Equation 2.22. For any odd $N > 1$, this following this same process and making the same observation about the derivative terms will result in the 3D SP_n equations. For $N = 1$, these equations are exactly equal to the P_n equations. However, for $N > 1$, this results in a simplified system of equations which still preserve much of the transport physics that are not preserved by the P_1 or diffusion approximations. Appendix A contains a detailed derivation of the SP_3 equations, but the final 3D SP_3 equations are shown in Equation 2.45.

$$-\boldsymbol{\nabla} \cdot D_{0,g}(\boldsymbol{x}) \boldsymbol{\nabla} \Phi_{0,g}(\boldsymbol{x}) + \left[\Sigma_{tr,g}(\boldsymbol{x}) - \Sigma_{s0,g}(\boldsymbol{x}) \right] \Phi_{0,g}(\boldsymbol{x})$$
$$= Q_g(\boldsymbol{x}) + 2 \left[\Sigma_{tr,g}(\boldsymbol{x}) - \Sigma_{s0,g}(\boldsymbol{x}) \right] \Phi_{2,g}(\boldsymbol{x}) \ , \tag{2.45a}$$

$$-\boldsymbol{\nabla} \cdot D_{2,g}(\boldsymbol{x}) \boldsymbol{\nabla} \Phi_{2,g}(\boldsymbol{x}) + \left[\Sigma_{tr,g}(\boldsymbol{x}) - \Sigma_{s2,g}(\boldsymbol{x}) \right] \Phi_{2,g}(\boldsymbol{x})$$
$$= \frac{2}{5} \left\{ \left[\Sigma_{tr,g}(\boldsymbol{x}) - \Sigma_{s0,g}(\boldsymbol{x}) \right] \left[\Phi_{0,g}(\boldsymbol{x}) - 2\Phi_{2,g}(\boldsymbol{x}) \right] - Q_g(\boldsymbol{x}) \right\} \ . \tag{2.45b}$$

Each of these equations is similar in form to the P_1 equation and can be solved by iterating between the the 0th moment equation (2.45a) and the 2nd moment equation (2.45b). This results in an accurate 3D flux distribution without the expense and complexity of performing a 3D P_N calculation.

2.2.3.2 Nodal Expansion Method

While SP_N can be used to capture angular moments of the flux, the Nodal Expansion Method (NEM) [14] is used to capture intra-nodal flux shapes to calculate accurate currents at the interface between two nodes. This is done by expanding the source and flux using quadratic and quartic Polynomials, respectively:

$$Q(\xi) = \sum_{i=0}^{2} q_i P_i(\xi) , \qquad (2.46a)$$

$$\phi(\xi) = \sum_{i=0}^{4} \phi_i P_i(\xi) , \qquad (2.46b)$$

where the variable ξ is simply the spatial variable normalized so the problem is being solved on the interval $[-1, 1]$. To solve for these moments, five equations are required. The first three are the moment balance equations, obtained by multiplying the diffusion equation by $P_n(\xi)$ for moments 0-2 and integrating (Equation 2.47a). The other two equations are found by enforcing flux and current continuity at the interfaces between nodes (Equations 2.47b-2.47c).

$$\int_{-1}^{1} P_n(\xi) \left(-\frac{D}{h^2} \frac{d^2}{d\xi^2} \phi(\xi) + \Sigma_r \phi(\xi) - Q(\xi) \right) d\xi = 0, \; n = 0, 1, 2 , \qquad (2.47a)$$

$$\phi_L(1) = \phi_R(-1) , \qquad (2.47b)$$

$$J_L(1) = J_R(-1) . \qquad (2.47c)$$

The source moments q_i are constructed using the flux moments from the previous iteration. Using this method, an intra-nodal flux shape can be calculated within each node, and interface currents can be calculated at the boundaries of each node.

2.2.4 Collision Probabilities Method

One method that can be used to calculate flux spectra inside a pin cell is the method of Collision Probabilities (CP) [15]. This method is used in MPACT as part of the control rod decusping methods described in Chapter 4. The details of the derivation will be left to Appendix A, but a brief overview of the method will be presented here that focuses on a 1D radial calculation for a single pin cell.

To begin, the multi-group transport Equation 2.11 can be rewritten so that the entire

right-hand side is a single source term:

$$\mathbf{\Omega} \cdot \nabla \psi_g + \Sigma_{tr,g}(\mathbf{x}) \psi_g(\mathbf{x}, \mathbf{\Omega}) = q_g(\mathbf{x}, \mathbf{\Omega}) \; , \tag{2.48a}$$

$$q_g(\mathbf{x}, \mathbf{\Omega}) = \frac{1}{4\pi} \sum_{g'=1}^{G} \int_{4\pi} \Sigma_{s,g' \to g}(\mathbf{x}, \mathbf{\Omega}' \to \mathbf{\Omega}) \psi_{g'}(\mathbf{x}, \mathbf{\Omega}') d\mathbf{\Omega}'$$

$$+ \frac{1}{k_{eff}} \frac{\chi_g}{4\pi} \sum_{g'=1}^{G} \int_{4\pi} \nu \Sigma_{f,g'}(\mathbf{x}) \psi_{g'}(\mathbf{x}, \mathbf{\Omega}') d\mathbf{\Omega}' \; . \tag{2.48b}$$

This form of the equation assumes no source of neutrons except scatter and fission. It also assumes isotropic (or transport-corrected) scattering, implying that the CP method cannot resolve highly anisotropic scattering. This gives way to a simplified isotropic source, shown in Equation 2.49:

$$q_g(\mathbf{x}, \mathbf{\Omega}) \approx q_g(\mathbf{x}) = \frac{1}{4\pi} \sum_{g'=1}^{G} \Sigma_{s,g' \to g}(\mathbf{x}) \phi_{g'}(\mathbf{x}) + \frac{1}{k_{eff}} \frac{\chi_g}{4\pi} \sum_{g'=1}^{G} \nu \Sigma_{f,g'}(\mathbf{x}) \phi_{g'}(\mathbf{x}) \; , \tag{2.49}$$

where ϕ_g is the scalar flux in group g. At this point, we assume the problem is discretized into R regions and that the cross-sections and fluxes are flat in each region, which leads to a flat source as well for each region r:

$$q_{g,r} = \frac{1}{4\pi} \sum_{g'=1}^{G} \Sigma_{s,g' \to g,r} \phi_{g',r} + \frac{1}{k_{eff}} \frac{\chi_{g,r}}{4\pi} \sum_{g'=1}^{G} \nu \Sigma_{f,g',r} \phi_{g'} \; . \tag{2.50}$$

Now the total source of neutrons in each region r can be calculated by multiplying by the region volume V_r. If the probability $T_{g,r' \to r}$ of a neutron born in region r' reaching region r is known for all regions r', then the scalar flux can be calculated as follows:

$$\phi_{g,r} = \sum_{r'=1}^{R} T_{g,r' \to r} q_{g,r'} V_{r'} \; . \tag{2.51}$$

Equation 2.51 gives the general solution to any problem using the CP method. The only remaining work that must be done is to calculate the transport matrix $\underline{\underline{T_g}}$ for each group, which is done in detail for a pin cell in the appendix.

Several important details about this method should be noted. First, it is common to use a buffer region in the calculation. When applying the CP method to a fuel pin, the buffer region is not necessary because the fuel pin has its own fission source to drive the problem.

However, when using this method on a different pin cell such as a control rod, it is useful to place a homogenized mixture of fuel and moderator around the outside of the pin cell. This provides a source to drive the problem. Second, because the calculation is a 1D radial calculation, the moderator region in the rectangular pin cell must be cylindricized. This is done by transforming the moderator region into a ring with the same inner radius and volume as the rectangular moderator region. Finally, the boundary conditions assumed on the edge of the problem are "white" boundary conditions. A reflecting boundary condition assumes that all neutron exiting the problem return at the same energy and traveling in a reflected direction. However, because the boundary of this problem is a circle, it is possible for a neutron to be born at such an angle that it continuously reflects off the boundary without ever traveling toward the fuel. The white boundary condition takes all exiting neutrons and returns them isotropically to prevent this error.

CHAPTER 3

2D/1D Framework

3.1   Background

The Boltzmann transport equation can be solved directly in 3D to obtain 3D flux and power distributions. One method to do this is the 3D Method of characteristics, which is implemented in MPACT [16]. However, performing these 3D transport calculations becomes too computationally burdensome to be of practical use, even with today's improved computing resources. Because LWRs have most of their material heterogeneity in the radial direction with very little change in the axial direction, it was recognized that approximations could be made in the axial direction to increase the efficiency of the calculations while still performing high-fidelity transport calculations in the radial direction. Two different groups of researchers pursued this concept and developed two different methods of solving the transport equation for LWR problems.

The first of these methods was the "2D/1D Fusion" technique, developed by researchers at Korea Advanced Institute of Science and Technology (KAIST) and implemented in codes such as CRX [17–19]. In this method, the 3D problem is decomposed into a stack of 2D planes. These planes are solved using 2D MOC, with incoming angular fluxes on the top and bottom boundaries of the plane as source terms. To couple the planes, the problem domain is integrated in the x- and y-directions for each pin cell. The angular fluxes at the radial edges are obtained from the 2D MOC calculations and used as source terms. The angular fluxes are then solved in the axial direction using the Diamond Difference method. These results, in turn, are fed back into the radial calculations. Iterating between the radial and axial calculations then produces a full 3D solutions.

The second group of researchers was at Korea Atomic Energy Research Institute (KAERI). They developed what is known more simply as the "2D/1D" scheme, first implemented in the DeCART code [20–22]. This employs very similar technique to the 2D/1D Fusion method described above. However, rather than using angular fluxes from each solver as a
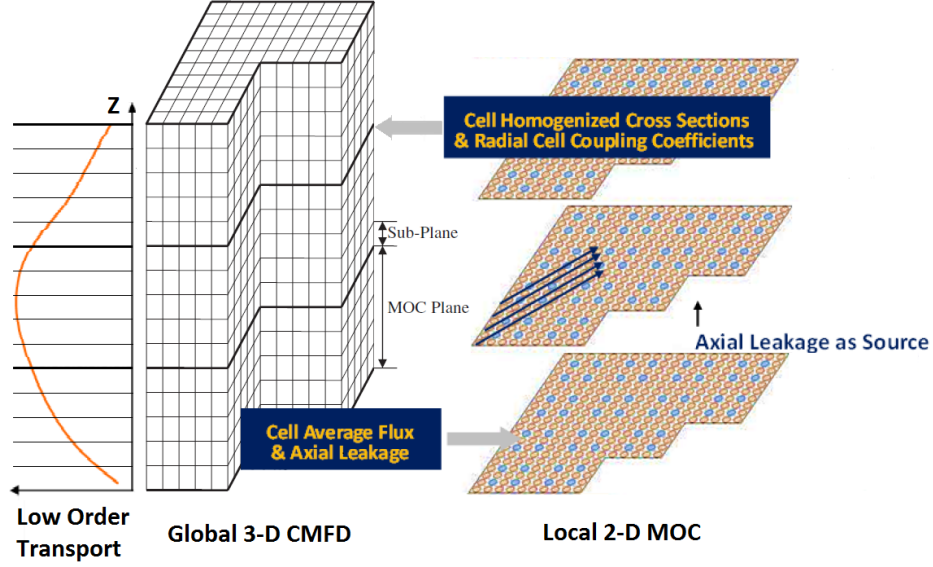
Figure 3.1: The 2D/1D Method illustrated with the sub-plane scheme for the axial and CMFD calculations

source term, currents are tallied on each of the six faces of each pin cell. The currents can then be used to compute axial and radial "transverse leakage" sources for the radial and axial solvers, respectively. This change allows for the storage of group-wise currents at each interface instead of storing the group-wise angular fluxes for each angle, significantly reducing the memory burden of the calculation.

After some development, the DeCART code was forked into several different versions for different institutions, one of them being the University of Michigan (UM). After some development, it was determined that there would be no further development of the De-CART code at UM and a new 2D/1D implementation would be put in MPACT [23, 24]. In MPACT's implementation of 2D/1D (shown in Figure 3.1), 2D MOC is used for each of the radial planes, as with earlier 2D/1D codes. The axial calculation done on a pin-homogenized mesh usually with SP_3, though a variety of other solvers are available such as NEM, SENM, SP_1, SP_5, or S_N. Finally, MPACT also uses 3D CMFD on the same pin-homogenized mesh to provide convergence acceleration to the calculations. The remainder of this chapter will look at the derivation of the 2D/1D equations, the details of how they are implemented in MPACT, and some of the approximations and sources of errors related to this method.

3.2 Derivation

3.2.1 Radial Equations

To derive the radial equations, we begin with the multigroup approximation in Equation 2.11 and integrate in the z-direction over some range $\Delta z_i = z_{k+\frac{1}{2}} - z_{k-\frac{1}{2}}$. To do this, we assume the cross-sections are all constant in the interval $z \in \left[z_{k-\frac{1}{2}}, z_{k+\frac{1}{2}} \right]$. With this assumption, we obtain the following equation:

$$
\begin{aligned}
\Omega_x \frac{\partial \psi_g^Z}{\partial x} + \Omega_y \frac{\partial \psi_g^Z}{\partial y} &+ \frac{\Omega_z}{\Delta z_k} \left(\psi_{g, z_{k+\frac{1}{2}}} - \psi_{g, z_{k-\frac{1}{2}}} \right) + \Sigma_{t,g}(x,y) \psi_g^Z(x,y,\boldsymbol{\Omega}) \\
&= \frac{1}{4\pi} \sum_{g'=1}^{G} \int_{4\pi} \Sigma_{s,g' \to g}^Z(x,y,\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) \psi_{g'}^Z(x,y,\boldsymbol{\Omega}') d\Omega' \\
&\quad + \frac{1}{k_{eff}} \frac{\chi_g^Z}{4\pi} \sum_{g'=1}^{G} \int_{4\pi} \nu \Sigma_{f,g'}^Z(x,y) \psi_{g'}^Z(x,y,\boldsymbol{\Omega}') d\Omega' + \frac{Q_g^Z(x,y)}{4\pi} ,
\end{aligned}
\tag{3.1a}
$$

$$
\psi_g^Z(x,y,\boldsymbol{\Omega}) = \frac{1}{\Delta z_k} \int_{z_{k-\frac{1}{2}}}^{z_{k+\frac{1}{2}}} \psi_g^Z(x,y,z,\boldsymbol{\Omega}) dz .
\tag{3.1b}
$$

where a superscript Z indicates the average of a quantity over a given plane. The z-component of the streaming can now be moved to the right-hand side of the equation and treated as a source term, giving a 2D transport problem which could be solved with a variety of methods:

$$
\Omega_x \frac{\partial \psi_g^Z}{\partial x} + \Omega_y \frac{\partial \psi_g^Z}{\partial y} + \Sigma_{t,g}(x,y) \psi_g^Z(x,y,\boldsymbol{\Omega}) = q_g^Z(x,y) + L_g^Z(x,y,\Omega_z) ,
\tag{3.2a}
$$

$$
\begin{aligned}
q_g^Z(x,y) &= \frac{1}{4\pi} \sum_{g'=1}^{G} \int_{4\pi} \Sigma_{s,g' \to g}^Z(x,y,\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) \psi_{g'}^Z(x,y,\boldsymbol{\Omega}') d\Omega' \\
&\quad + \frac{1}{k_{eff}} \frac{\chi_g^Z}{4\pi} \sum_{g'=1}^{G} \int_{4\pi} \nu \Sigma_{f,g'}^Z(x,y) \psi_{g'}^Z(x,y,\boldsymbol{\Omega}') d\Omega' + \frac{Q_g^Z(x,y)}{4\pi} ,
\end{aligned}
\tag{3.2b}
$$

$$
L_g^Z(x,y,\Omega_z) = \frac{\Omega_z}{\Delta z_k} \left(\psi_{g, z_{k-\frac{1}{2}}} - \psi_{g, z_{k+\frac{1}{2}}} \right) ,
\tag{3.2c}
$$

where $L_g^Z(x,y,\Omega_z)$ is the axial transverse leakage source term for plane z. To simplify the source term, the axial transverse leakage term is often handled isotropically. This is done

by averaging over angle:

$$L_g^Z(x,y) = \frac{1}{4\pi} \int L_g^Z(x,y,\Omega_z)\, d\Omega \approx \frac{J_{g,z_{k-\frac{1}{2}}} - J_{g,z_{k+\frac{1}{2}}}}{4\pi\Delta z_k} \;, \tag{3.3}$$

where $J_{z_{i\pm\frac{1}{2}}}$ is the current at the top (+) or bottom (−) of the plane. This eliminates the need for storing all the angluar fluxes on the top and bottom of every plane. Other methods exist that allow the axial transverse leakage source to maintain its angular dependence without storing the angular fluxes [25], but these methods are not discussed here since they were not used by this work.

3.2.2   Axial Equations

The axial equations can be derived in a manner similar to the radial equations. Again, we begin with the multi-group approximation shown in Equation 2.11. This time, we integrate in both the x and y directions over intervals $x \in \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$ and $y \in \left[y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}\right]$, giving the following equations in the axial direction, which are analogous to the radial equations in the previous section:

$$\Omega_z \frac{\partial \psi_g^{XY}}{\partial z} + \Sigma_{t,g}^{XY}(z)\psi_g^{XY}(z,\mathbf{\Omega}) = q_g^{XY}\left(z,\Omega_x,\Omega_y\right) + L_g^{XY}\left(z,\Omega_x,\Omega_y\right) \;, \tag{3.4a}$$

$$
\begin{aligned}
q_g^{XY}\left(z,\Omega_x,\Omega_y\right) &= \frac{1}{4\pi}\sum_{g'=1}^{G} \int_{4\pi} \Sigma_{s,g'\to g}^{XY}(z,\mathbf{\Omega}'\cdot\mathbf{\Omega})\psi_{g'}^{XY}(z,\mathbf{\Omega}')\, d\Omega' \\
&+ \frac{1}{k_{eff}}\frac{\chi_g^{XY}}{4\pi}\sum_{g'=1}^{G}\int_{4\pi} \nu\Sigma_{f,g'}^{XY}(z)\psi_{g'}^{XY}(z,\mathbf{\Omega}')\, d\Omega' + \frac{Q_g^{XY}(z)}{4\pi} \;,
\end{aligned}
\tag{3.4b}
$$

$$L_g^{XY}\left(z,\Omega_x,\Omega_y\right) = \frac{\Omega_x}{\Delta y_i}\int_{y_{i-\frac{1}{2}}}^{y_{i+\frac{1}{2}}}\left(\psi_{g,x_{i-\frac{1}{2}}}(y) - \psi_{g,x_{i+\frac{1}{2}}}(y)\, dy\right) + \frac{\Omega_y}{\Delta x_i}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\left(\psi_{g,y_{i-\frac{1}{2}}}(x) - \psi_{g,y_{i+\frac{1}{2}}}(x)\, dx\right),$$

$$\tag{3.4c}$$

$$\psi_g^{XY} = \frac{1}{\Delta_i\Delta_j}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \psi_g(x,y,z,\mathbf{\Omega})\, dx\, dy \;, \tag{3.4d}$$

where a superscript XY now corresponds to a particular x- and y-region which extends the full height of the problem in the z-direction. Again, it is assumed that the cross-sections

are constant in the x- and y-directions inside the region of integration. How this is accomplished will be discussed in more detail when discussing MPACT's implementation of SP$_3$ and CMFD.

As with the radial equations, we can treat the transverse leakage source isotropically by averaging over angle:

$$L_g^{XY}(z) = \frac{1}{4\pi} \int\limits_{4\pi} L_g^{XY}\left(z, \Omega_x, \Omega_y\right)$$

$$\approx \frac{1}{4\pi \Delta x_i \Delta y_j} \int\limits_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \left(J_g\left(x_{i-\frac{1}{2}}, y, z\right) - J_g\left(x_{i+\frac{1}{2}}, y, z\right) \right) dy \tag{3.5}$$

$$+ \frac{1}{4\pi \Delta x_i \Delta y_j} \int\limits_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left(J_g\left(x, y_{j-\frac{1}{2}}, z\right) - J_g\left(x, y_{j+\frac{1}{2}}, z\right) \right) dx \; .$$

As an additional approximation we assume that the currents are tallied across an entire transverse surface, meaning that the current terms in the integrals above are constant over the direction of integration. With this simplification, the final form of the radial TL is

$$L_g^{XY}(z) = \frac{J_{g, x_{i-\frac{1}{2}}, y_j}(z) - J_{g, x_{i+\frac{1}{2}}, y_j}(z)}{4\pi \Delta x_i} + \frac{J_{g, x_i, y_{j-\frac{1}{2}}}(z) - J_{g, x_i, y_{j+\frac{1}{2}}}(z)}{4\pi \Delta y_j} \tag{3.6}$$

Again, methods have been developed to angle the angle-dependence of the radial transverse leakage source [26], but this work used only isotropic radial leakage.

3.3 Implementation

Now that the general 2D/1D scheme has been described, some attention should be given to the details of its implementation in MPACT. Figure 3.2 shows the calculation flow used by MPACT. The first step is to perform a global 3D CMFD calculation to obtain pin-averaged flux and interface currents between each cell. Next, the axial solver uses the radial currents calculated by CMFD as a radial transverse leakage source to obtain an axial transverse leakage source for the radial solver. Finally, 2D MOC is used as the radial solver to obtain a solution with sub-pin resolution in each plane.
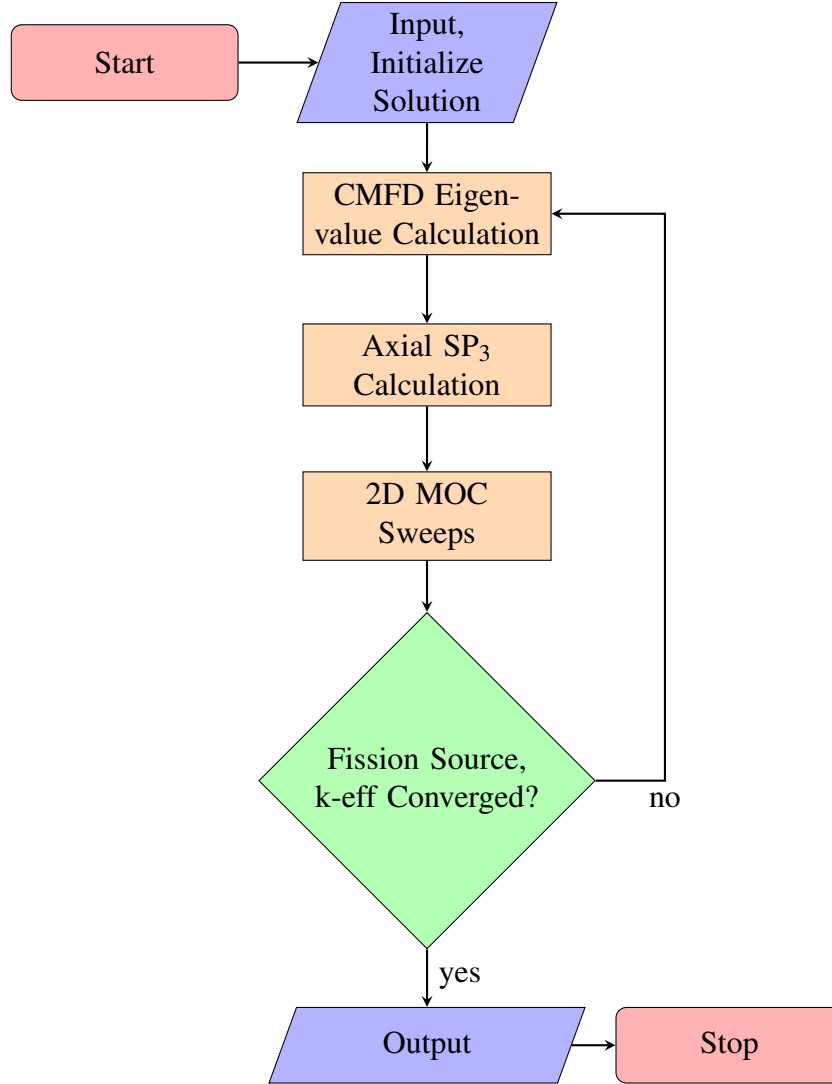
Figure 3.2: Calculation flow for 2D/1D scheme

3.3.1 3D Sub-plane CMFD

The CMFD method was originally implemented in MPACT just as described in section 2.2.2. To do this, each pin cell is homogenized using the quantities defined in Equation 2.35 in every plane in the model. The radial coupling coefficients defined in Equation 2.38 are obtained by calculating the current at the interface between each pair of pin cells using the 2D MOC sweeper, while the axial coupling coefficients are obtained from the axial currents calculated by the axial solve during the previous iteration. The matrix for the 3D multi-group system can then be set up and solved, typically using the generalized Davidson eigenvalue solver.

As part of this work, the traditional CMFD capability was extended to use the sub-

plane scheme, first developed by Cho et al. for the DeCART code [27] and later used in the nTRACER code [28]. Thin MOC planes are capable of causing instability in the 2D/1D scheme, but are sometimes required to maintain accuracy. The sub-plane scheme allows users to increase the thickness of the 2D planes while still maintaining the accuracy of a fine axial mesh. While DeCART used the sub-plane scheme primarily to allow for thicker MOC planes, nTRACER also uses the sub-plane scheme as part of its rod decusping methods [29]. This section will focus only on the sub-plane scheme as a means of using fewer MOC planes, while chapter 4 will discuss how the sub-plane scheme has been modified to be used in decusping methods.
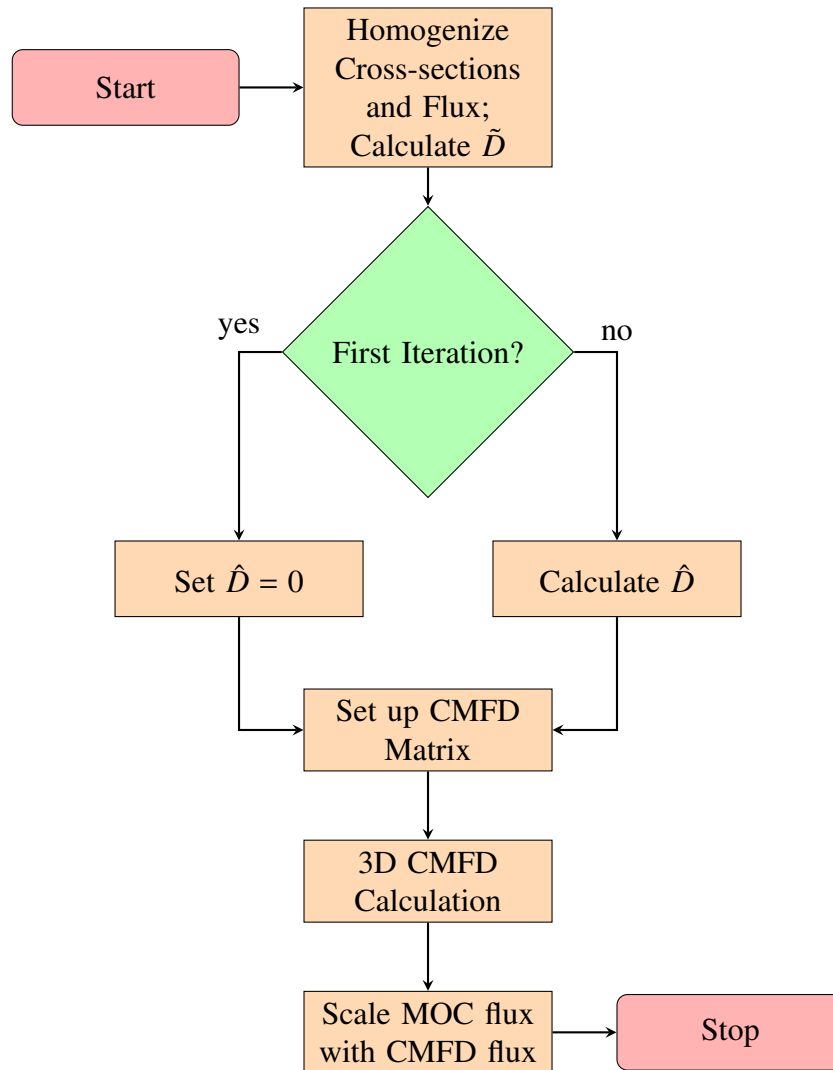

Figure 3.3: Calculation flow for 3D sub-plane CMFD

28

3.3.1.1 Homogenization

For the traditional CMFD calculations, each pin cell in an MOC plane is homogenized into a single CMFD cell. When using the sub-plane scheme, the homogenized pin cell is divided axially into a stack of cells. This causes the CMFD system to have multiple planes per MOC plane, allowing CMFD to capture sub-plane axial flux shapes that would otherwise be ignored. To do this, a sub-plane scaling factor is introduced which will be used to provide an axial shape within a 2D plane:

$$
\begin{aligned}
c_{g,i}^{k} &= \frac{\phi_{g,i}^{k-1}}{\overline{\phi}_{g,i}^{k-1}} \\
&= \frac{\phi_{g,i}^{k-1} \sum_{i'=1}^{N_{sp}} V_{i'}}{\sum_{i'=1}^{N_{sp}} \phi_{g,i'}^{k-1} V_{i'}} \ ,
\end{aligned}
\tag{3.7}
$$

where superscripts indicate which iteration the values are taken from and N_{sp} is the number of sub-planes for the pin cell of interest. Now when the homogenized values are calculated from the 2D transport solution using Equation 2.35, the fine mesh flux is multiplied by this sub-plane scaling factor everywhere it appears. Because the 2D/1D scheme assumes a constant material axially in each plane, this sub-plane factor has no impact on the homogenized cross-sections. However, the homogenized flux $\phi_{g,i}$ and fission source distribution $\chi_{g,i}$ will be changed, providing an axial shape for the source term in the eigenvalue calculation.

3.3.1.2 Coupling Coefficients

In addition to the homogenized cell terms, the coupling coefficients described by Equations 2.37 and 2.38 must be calculated for each sub-plane. To maintain consistency, the area-averaged current calculated by the radial sweeper must be preserved across the sub-surfaces used by the sub-plane scheme. Thus, the current calculated by the radial sweeper at an interface is used at the corresponding interfaces for all sub-planes in that plane. Additionally, to maintain consistency, this requires that the cell-homogenized flux used in the calculation of the diffusion coefficients be defined for the entire MOC plane as in Equation 2.35e rather than using the sub-plane scaling factor for each sub-plane.

The axial coupling coefficient can be treated in a more straightforward manner. Because the 1D axial solvers use the same pin-homogenized mesh as the CMFD solver, axial currents are naturally calculated at the top and bottom of each of the sub-planes. Thus, these currents can be used together with the sub-planes fluxes to calculate sub-plane–dependent axial coupling coefficients.

3.3.1.3 Projection

The projection of the CMFD flux back to the 2D planes must also account for the presence of the sub-planes. To do this, the solution is volume-averaged over all sub-planes for each pin cell, resulting in an equation similar to 2.40:

$$\phi^k_{trans,g,j} = \frac{\sum_{i'=1}^{N_{sp}} \phi^k_{CMFD,g,i'} V_i}{\sum_{i'=1}^{N_{sp}} \phi^k_{CMFD,g,i'} V_i} \phi^{k-1}_{trans,g,j} . \tag{3.8}$$

The calculation flow for 3D CMFD is shown in Figure 3.3.

3.3.2 1D NEM-SP$_3$
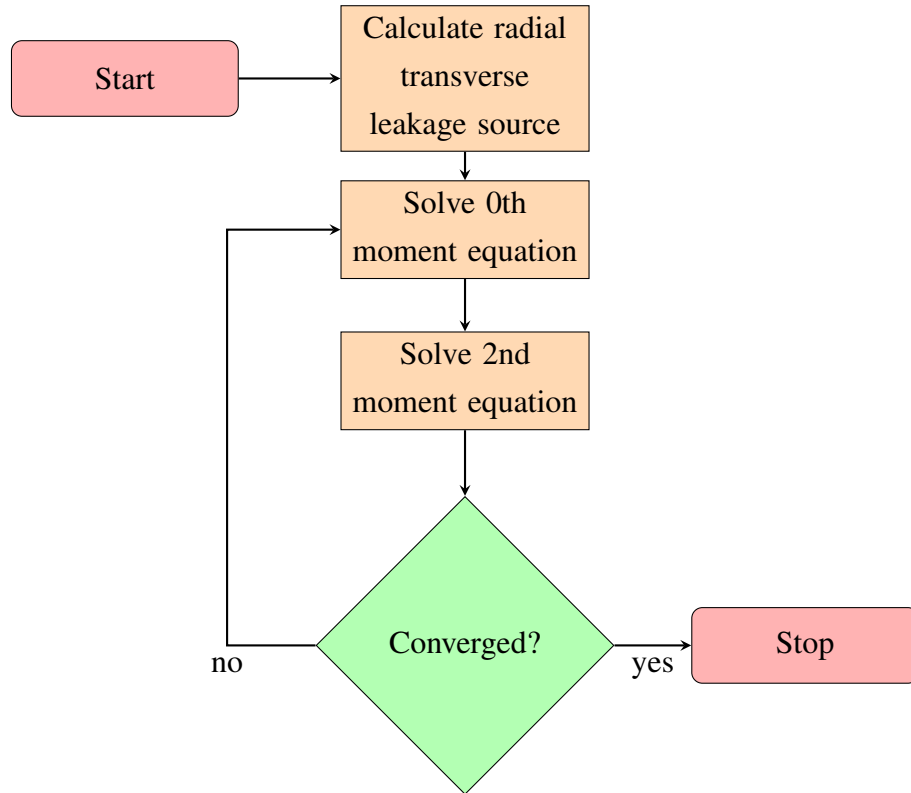
Figure 3.4: Calculation flow for 1D axial calculations in MPACT

In MPACT, the 1D axial solvers operate on the same mesh as the 3D CMFD calculations, meaning that cell-homogenized quantities and radial currents have already been obtained from the CMFD calculation. All the 1D axial solver must do is construct a source term from the radial currents for each cell, then perform a calculation to obtain currents on the axial interfaces at the top and bottom of each node.

MPACT has a variety of 1D nodal methods that are capable of performing these calculations, including diffusion-based such as NEM and SENM and higher-order solvers such as SP_N and S_N. For most calculations, MPACT uses SP_3 wrapped in NEM kernels. The SP_3 portion handles the angular dependence of the solution by calculating angular moments. The NEM kernels then handle the spatial dependence through the fourth-order polynomial expansion. Because the SP_3 equations can be written as two sets of P_1 equations, they naturally lend themselves to being solved this way.

The SP_3 equations consist of equations for angular flux moments 0 through 3. These equations can be combined into two equations for just the 0th and 2nd moments, as shown in Equations 2.45. Formulating SP_3 this way makes it straightforward to use the NEM kernels for each of the two moment equations. The iteration scheme for this procedure is shown in Figure 3.4.

3.3.3 2D MOC

For the radial calculations, 2D MOC is used. This allows MPACT to easily calculate scalar fluxes and currents in each plane regardless of the geometric complexity. This section is devoted to discussing some of the details of the MOC implementation and sweeping algorithm in MPACT.

3.3.3.1 Ray Tracing

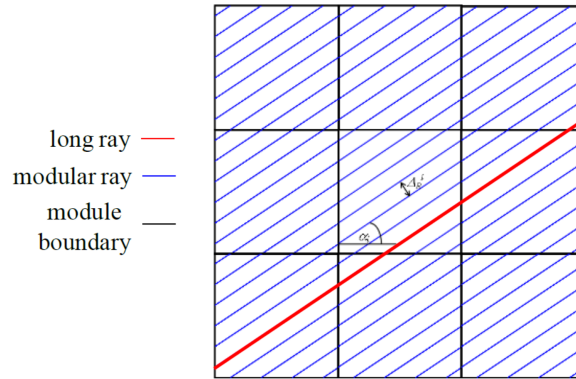**Figure 3.5: Modular ray tracing depiction [3]**

One of the key features of MPACT's MOC implementation is that of modular ray tracing. Ray tracing is performed once at the beginning of a calculation and stored for the remainder of the calculation. Doing this greatly reduces the runtime of the MOC sweeps since the length of each ray segment and the region it is crossing are already known ahead

of time. Furthermore, MPACT takes advantage of the repeatable nature of a reactor's geometry. Because reactor geometry is repetitive, small portions of the geometry which repeat frequently can be ray-traced separately instead of tracing the entire core. These smaller units of geometry are known as ray tracing modules, and in MPACT can be a full fuel assembly, a quarter fuel assembly, or a single fuel pin. After the unique modules are identified, each of them is ray-traced in such a way that the endpoints of a ray in each module will line up with the beginning of a ray in the neighboring module. This significantly reduces the storage requirements for the ray data since a small number of ray tracing modules can represent a full core problem.

As a result of the modular ray tracing, several corrections are required. First, to successfully perform the ray tracing the angles of the rays had to be adjusted slightly to line up. However, because a quadrature is used to integrate the angular flux, this angle modification requires a correction to the quadrature weights as well to maintain accuracy. Second, the spacing between the rays must also be adjusted slightly to ensure that all rays align. Along with these corrections, there are other MOC concepts such as volume corrections, cyclic rays, and others which are important to be aware of but will be deferred to the MPACT theory manual for details [3].

3.3.3.2   Sweeping Algorithm

To perform the MOC sweeps, MPACT uses a multi-group sweeping method. To do this, multi-group sources and cross-sections are prepared for each of the fine mesh regions in each plane. There are then four total loops for the sweeping algorithm. From outermost to innermost, these loops are over azimuthal angle, ray (across the entire domain), polar angle, and energy group. Each ray is divided into many segments that the MOC steps its way along as described in section 2.2.1. As these sweeps are performed, scalar flux is tallied in each fine mesh region, and, assuming CMFD is being used, currents are tallied at the interfaces between each pin cell. Normally only one inner MOC iteration is required per 2D/1D iteration. This algorithm is shown in Figure 3.6.

MPACT also has the capability of performing the MOC sweeps with the energy loop as the outermost. This has two advantages. First, the previous group is used to construct the scattering source for the current group. This means that the iteration scheme is a Gauss-Seidel iteration instead of a Jacobi iteration, which can speed up the convergence of the problem. Secondly, the cross-sections and sources only need to be stored for one group at a time, minimizing the storage requirements for the calculations. However, when using transport-corrected cross-sections, some instabilities have been observed in this iteration scheme when using only a single inner iteration. Further more, having the energy loop on
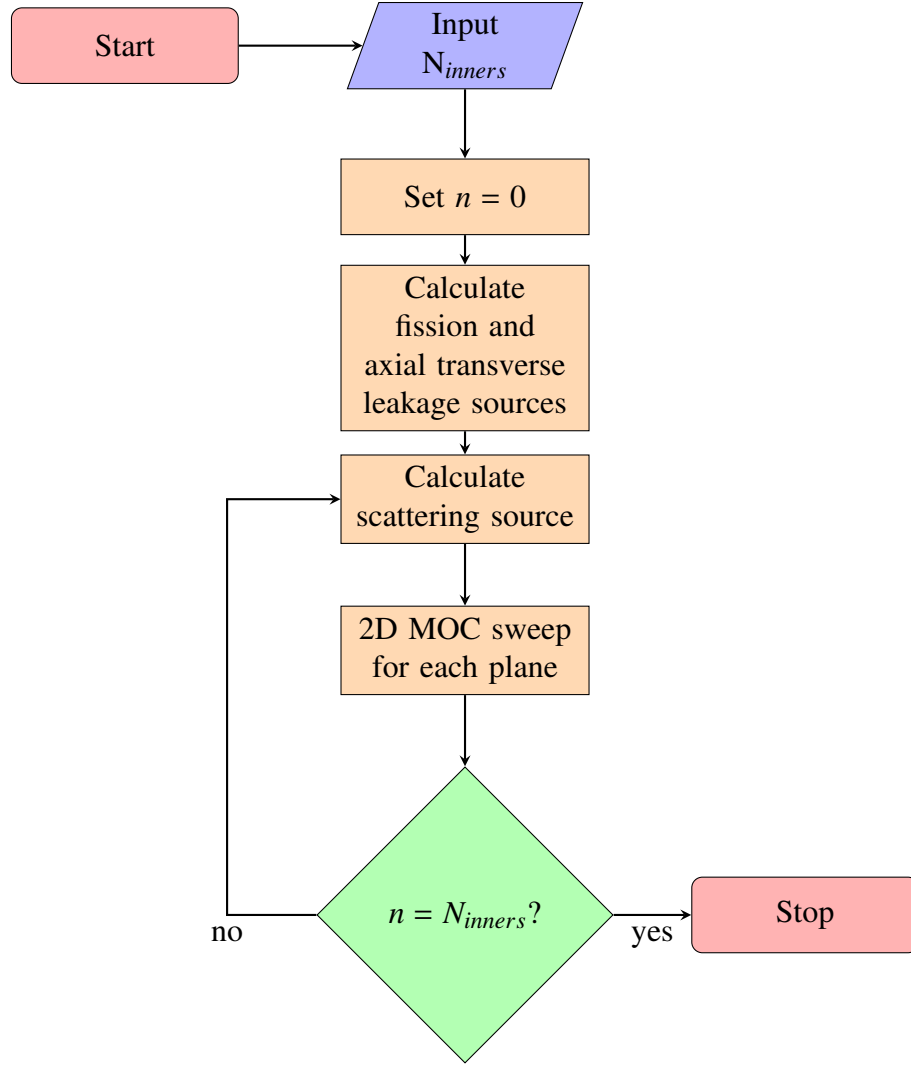
Figure 3.6: Calculation flow for 2D MOC calculation in MPACT

the inside results in some improved cache efficiency when it comes to traversing the rays, reducing the runtime for a single MOC sweep [30]. For these two reasons, MPACT defaults to the Jacobi-style iteration scheme described first.

3.4   Parallel Decomposition

While the 2D/1D scheme greatly reduces runtime from a direct 3D transport calculation, it is still computationally expensive when compared to nodal methods traditionally used by industry. To minimize the walltime required for 2D/1D calculations in MPACT, several different methods of decomposing the problem for parallel execution have been implemented. These methods allow MPACT to easily scale to hundreds or thousands of CPUs. Each of

these methods will be briefly described in this section.

1. **Spatial Decomposition**

   When using this decomposition, each parallel process only has a portion of the model. Each portion is solved locally by one process, then boundary data is communicated to all processes which own neighboring portions of the model. The updated boundary data is then used in the following iteration. When using spatial decomposition, planar decomposition is performed first. This means that if the total number of parallel processes being used is less than or equal to the number of 2D MOC planes, then one or more entire planes is simulated by each process. If more processes are used than there are planes in the model, then radial decomposition is performed. This decomposes every plane radially into smaller pieces. Every plane must be radially decomposed in the same way, and the smallest unit allowed in radial decomposition is a single ray-tracing module. Because spatial decomposition does not duplicate much memory and does not decrease the computational efficiency significantly, it is usually the preferred choice of decomposition methods.

2. **Angle Decomposition**

   For angle decomposition, each process has the entire spatial domain. When the MOC sweeps are performed, each process only sweeps a subset of the angles in the selected quadrature. After the sweep, a reduction is performed to get the actual scalar flux and currents on all processes. For the CMFD calculation, the angle processes are re-purposed as spatial processes. Each angle process owns the full domain, but only solves a portion of it as if it were spatially decomposed.

   It is possible to use both spatial and angle decomposition together. When this is done, spatial decomposition is performed first, then angle decomposition is done within each spatial domain. In general, the efficiency of angle decomposition calculations is less than that of spatial decompositions. Furthermore, it also requires that each angle process models all of the spatial domain, increasing the total memory required for the calculation compared with finer spatial decomposition. However, angle decomposition is still useful for reducing the runtime of cases where further spatial decomposition is not possible.

3. **Ray Decomposition**

   A third type of decomposition that can be done is to decompose the rays in the MOC calculation. Unlike the previous methods, the ray decomposition makes use of shared-memory threading instead of distributed memory message passing. While performing

the MOC sweeps, several threads are used to solve all the rays in each angle. For the CMFD calculation, MPACT has internal RBSOR solvers which are capable of using threading. However, when third-party libraries are used for the CMFD calculations, the threading will be used only during the CMFD calculation. Threading can also be combined with both spatial and angle decomposition to further increase the parallelism of MPACT.

4. **Energy Decomposition**

   At this time, energy decomposition is not available in MPACT. However, when it is added, it will be similar to the angle decomposition. For the MOC calculation, each process will solve a subset of the energy groups on the spatial domain, and for the CMFD calculation, the energy processes will be re-purposed as space processes.

3.5   Sources of Error

The 2D/1D approximation has several sources of error. Some of these are addressed by mesh, ray spacing, or quadrature refinements, but others are due to approximations made in the method itself. The sources of error which are due to fundamental approximations in the 2D/1D method will be discussed first, followed by a brief (not comprehensive) list of some other common sources of error.

3.5.1   Axial Homogenization

When deriving the radial equations in section 3.2.1, it was assumed that the cross-sections were constant in the axial direction for each of the planes. While this is often the case if an appropriate axial mesh is selected, sometimes it is impractical to mesh the problem finely enough to ensure this. When an axial material heterogeneity is present in a plane, 2D MOC requires that these materials be homogenized. In some cases, a simple volume-averaging is sufficient, but if a material with a large cross-section is being homogenized with a material that has a significantly different cross-section, significant errors can result. To prevent this without refining the axial mesh, some modification to the 2D/1D scheme is required to improve the homogenization. This will be addressed in chapter 4.

3.5.2   Axial Transverse Leakage Source

Another approximation relates to the assumptions made while deriving the axial equations in section 3.2.2. The SP_3 calculations are performed on a pin-homogenized mesh. Because

of this, the axial currents used in the axial TL source are assumed to be flat across the entire pin cell. However, the currents will obviously be quite different in the fuel and moderator regions. Furthermore, the axial TL is treated isotropically. While this simplifies the axial calculations and MOC storage requirements, it does not perfectly reflect reality. Both these spatial and angular assumptions introduce some error into the axial TL source used by MOC.

3.5.3 Radial Currents

The radial TL source used by the axial SP$_3$ solver has the same two approximations as the axial TL source did. Radial currents are used to generate the source, which assumes isotropy. Additionally, the spatial shape is flat across each pin cell. This is corrected to some extent since the axial solver produces a quadratic source shape using the neighboring nodes, but this is not a perfect solution.

Additionally, when using the sub-plane method, the $\hat{D}$ correction terms used by CMFD are assumed to be axially flat within each MOC plane. While this assumption improves the stability of the calculations, it forces CMFD to capture any axial shape the current has within an MOC plane. For most problems, this error is probably negligible, but for cases such as a partially inserted rod or other strong absorber, it would be beneficial to be able to have sub-plane–dependent $\hat{D}$ terms to more accurately calculate the radial currents. Doing this would improve the radial TL source in the SP$_3$ solver, and the the overall 2D/1D results.

3.5.4 Other Sources of Error

Several other sources of error in the 2D/1D method will be briefly mentioned here, but not discussed in detail.

1. **Ray Spacing**

   The spacing between the rays in the MOC calculation is important to the accuracy of the calculation. At minimum, one ray needs to pass through each of the fine mesh regions, but multiple rays will improve the accuracy. A typical ray spacing is 0.05 cm, but sometimes finer spacing may be required.

2. **Radial Meshing**

   The radial and azimuthal meshing of each of the pin cells must be fine enough to give a good solution. In MPACT, fuel pins usually have 3 radial rings, with an extra ring in the moderator region to resolve the change in the flux near the edge of the fuel pin. Each ring is typically divided into 8 azimuthal regions.

3. **Axial Meshing**

   The axial mesh must be refined enough to capture the axial shape of the solution. Usually MOC planes of about 8 cm thick are used for a typical PWR calculation, which some thinner planes to resolve spacer grids, burnable poison inserts, or other components. Using thicker planes could decrease accuracy and worsen the convergence of the CMFD calculations. This effect can be minimized for thick MOC planes by using the sub-plane method.

4. **Scattering Treatment**

   Scattering in a reactor, especially off the hydrogen atoms in the moderator, is anisotropic. To resolve this, a sufficiently high-order scattering treatment must be used in the MOC calculations. For PWRs, P_1 to P_3 is a typical range. MPACT is capable of going up to P_5 scattering treatment for libraries which have the required data.

   An alternative is to use transport-correction P_0 scattering. This can capture the anisotropy without increasing the runtime of the calculations. However, there are several methods of calculating the transport cross-sections, and none of them are perfect. Thus, using the TCP_0 option in MPACT also has some non-trivial error associated with it.

5. **Cross-Section Library**

   To perform any calculations using the 2D/1D method, a multi-group cross-section library must be available. While this is not technically a source of error in the 2D/1D method itself, the cross-section library can be difficult to generate correctly. Any error in any isotope in the library will cause error in the 2D/1D calculations if the isotope is used in the model. Thus, the 2D/1D method is useless if the a bad cross-section library is being used.

6. **Self-Shielding**

   Another potential source of error is related to spatial and energy self-shielding. To correctly deal with resonance absorption in the fuel while also accounting for the spatial self-shielding in the fuel, MPACT uses the subgroup method [31, 32]. Without using this method, the k_{eff} calculated by MPACT is off by several percent, along with an inaccurate flux distribution.

7. **Quadrature**

   One final source of error that can arise is in the selection of a quadrature. It is important to select both an appropriate number of azimuthal and polar angles as well as an appropriate type of quadrature. Typically around 16 azimuthal angles and 3 polar angles

is sufficient. There are several different types of quadratures implemented in MPACT, but generally a Tabuchi-Yamamoto quadrature is used for the polar angles [33] with a Chebyshev quadrature for the azimuthal angles [34].

CHAPTER 4

Rod Cusping

This chapter will focus specifically on control rod cusping effects, which are the focus of this work. First, a more thorough definition of the problem and motivation for solving it will be presented. The next section will then present some of the solutions that have been used to minimize the cusping effects in the past, including a simplified decusping model implemented in MPACT itself. The third section will then discuss some newer methods based on the sublpane CMFD scheme that have recently been implemented in MPACT. Finally, a new "sub-ray" MOC method will be proposed to deal with the cause of the cusping effects on a more fundamental level.
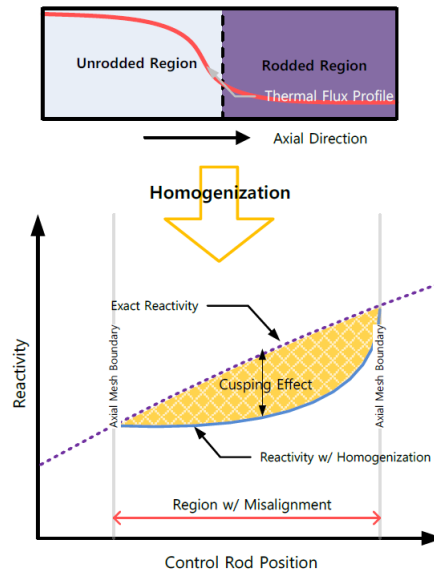
4.1   Background

Figure 4.1: Illustration of Rod "Cusping" [29]

In Section 3.5, some potential sources of errors for the 2D/1D scheme were introduced. One of these was the error introduced by axial homogenization within a 2D MOC plane. In some cases, this can be done without introducing significant errors. For example, MPACT often homogenizes components outside the active fuel region, such as the end plugs and gaps at the end of the fuel rods. However, when strong neutron absorbers, such as control rods, are homogenized axially in active fuel region, this has the effect of introducing absorption in regions where there should be none. This effect is known as "cusping," [14] and is illustrated in Figure 4.1.
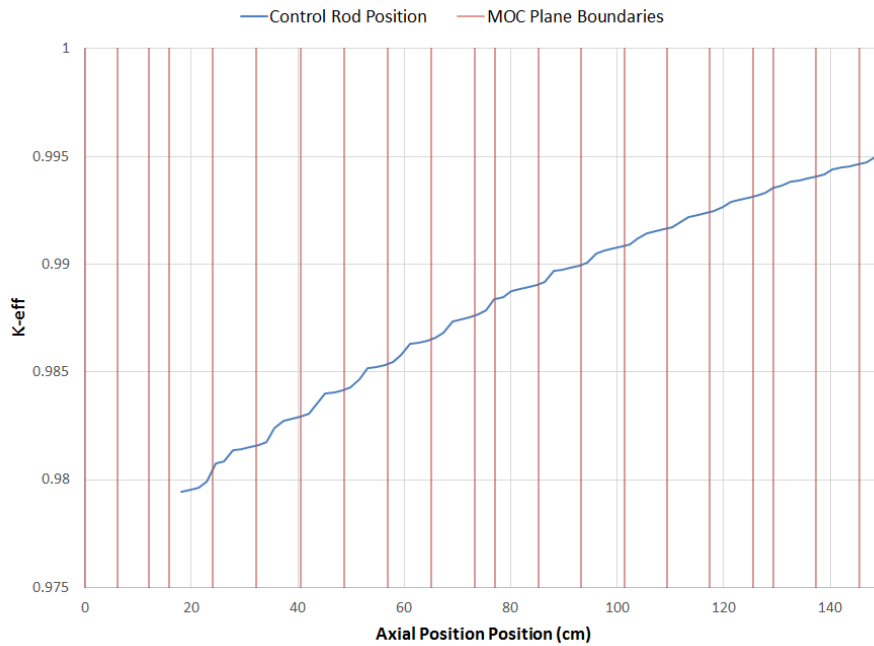

Figure 4.2: Control Rod Cusping Effects in MPACT for 3x3 Assembly

In some cases, this is easily handled by setting up an appropriate axial mesh which prevents the need for the homogenization, but this is not always a practical solution. Throughout the course of an entire cycle of operation (usually about 18 months), several different control banks in the reactor may move to a variety of positions to maintain criticality in the core. Control rods in a PWR typically have step sizes of approximately 1.5 cm, but a typical MOC plane in MPACT is about 8 cm thick in the active fuel region. In order to prevent cusping effects for an entire cycle, the user may have to create a very detailed axial mesh to ensure that all the control rod positions used throughout the cycle align with the edge of an MOC plane. Not only is this tedious for the user, but it also greatly increases the computational burden due to the increased number of MOC planes. Figure 4.2 shows the calculated k-eff as a function of control rod position. The cusping effects in this figure

40

are further complicated by a heterogeneous rod with AIC and B$_4$C poison regions and a stainless steal tip. Thus, cusping effects occur not just at the control rod tip, but also at material interfaces throughout the rod.

4.2 Decusping Methods History

Before discussing the recent and proposed methods to address the rod cusping problem, it is useful to provide an overview of past methods. These will then be used as a comparison for recent work and justification for the proposed work. First, we will look at some of the ways rod cusping was address in older nodal codes. Then we will look at the effects in 2D/1D codes and recent work to address them.

4.2.1 Nodal Codes

Control rod decusping methods have been developed for nodal codes for over thirty years now. Many different methods have been developed over that time frame, making a comprehensive discussion impractical. However, several different methods will be discussed here to provide some context for how the rod cusping problem was handled in nodal codes.

4.2.1.1 Tabulation Methods

One of the most basic methods of handling the cusping problem was through pre-tabulation of cross-section for the partially rodded node (PRN) [35]. For the nodal calculations, node-averaged cross-sections were generated using some higher fidelity method on small portions of the domain. For the PRN, this calculation could simply be repeated many times for each of the possible rod positions. However, because of the many positions the control rod could have in a reactor, this required many different cross-sections to be calculated. Furthermore, these calculations needed to be done on more than a single assembly to obtain acceptable accuracy due to inter-assembly effects [36]. Others tried using multi-assembly calculations [37] and color sets [38] to improve the accuracy of this method. However, the number of rod possible positions and the size of the problems required to generate accurate homogenized cross-sections made this method impractical at the time, motivating further research into rod decusping methods.

One variation of this method which improved on the runtime used response matrices and node surface currents [39,40]. For this method, response matrices were tabulated ahead of time based on the surface current boundary conditions. An iterative process could then be carried out between the global nodal calculation and a local fixed sources calculation.

During each iteration, the surface currents from the global calculation were used to update the homogenized cross-sections in the PRN using the response matrices. At convergence, the cross-sections in the PRN were calculated from the actual solution boundary conditions, greatly increasing the accuracy of the global solution compared with the simpler cross-section tabulation method. However, this method still required up-front calculation of the response matrices, which could be expensive for some problems. Consistency was also not perfectly maintained between the local and global problems, which sometimes resulted in convergence difficulties.

4.2.1.2 Polynomial Flux Expansions

Another early method which sought to improve on the brute-force tabulation method used polynomial expansions of the intra-nodal flux shape in the axial direction [41,42]. Methods such as NEM which are commonly used in nodal codes typically assume some shape to the source and flux inside each node. For this method, a quadratic shape was assumed and used to flux-volume weight the rodded and unrodded cross-sections to generate homogenized cross-sections for the PRN. This resulted in a reduction of the cusping errors by about 50% [43], which was not sufficient for many applications. Higher order polynomials were also attempted [44], but the axial shapes these generated were non-physical and sometimes led to oscillations and other numerical problems.

4.2.1.3 Collector-Predictor Method

Joo [35] focused on developing a method which would address rod decusping using only material compositions and node dimensions, rather than relying on boundary conditions as well. To do this, he developed an asymptotic method in which the PRN was modeled as two semi-infinite slabs, with one slab being composed of rodded material and the other slab being unrodded material. A 1D multi-group calculation was then carried out to determine the flux shape around the interface between the two materials. This flux shape was then used in the homogenization process for the PRN.

This method was found to be inaccurate because the infinite system is not actually a good representation of the flux shape in a reactor. To improve on this method, Joo developed what is known as the Collector-Predictor Method. There are two variations to this method. The first, simpler variation assumes that the first calculation has the tip of the control rod aligned with the boundary between two nodes. The collector step occurs at the end of the global calculation, collecting the radial transverse leakages and axial boundary currents for the rodded and unrodded nodes which neighbor each other. Then, rather than

performing calculations using semi-infinite slabs as in the previous paragraph, a 1D calculation is instead performed using the information from the collector step. This predictor step generates a more accurate flux profile around the interface between the rodded and unrodded materials. Then, when the rod is moved throughout the calculation, it is assumed that this axial flux profile in the vicinity of the rod tip does not change significantly, so the profile can just be shifted with the rod and used in all subsequent cross-section homogenizations. This method then improved to allow the first calculation to also have a PRN. Overall, this method significantly improved on previous ones by reducing decusping methods dynamically that converged consistently and did not rely on tabulated values. Furthermore, because the extra calculations were 1D on a small sub-domain of the problem, the additional computational expense was small.

4.2.1.4 Bi-Linear Weighting Method

Many decusping methods which followed the Collector-Predictor Method focused on improvements to the 1D calculation or boundary condition information to improve accuracy [45–48]. However, one significant variation that was developed by Kim and Cho [49] is the Bi-Linear Weighting Method. This method uses the same concept as the Collector-Predictor method, but solves for both the forward and adjoint axial flux shapes. Both of these shapes are then used in the cross-section homogenization, as shown in Equation 4.1. The use of the adjoint flux in the homogenization process resulted in significant improvements in the accuracy of the nodal calculations.

$$\overline{\Sigma} = \frac{\int \phi(z)\phi^*(z)\Sigma(z)\,dz}{\int \phi(z)\phi^*(z)\,dz} \ . \tag{4.1}$$

4.2.1.5 Nodal Expansion Method Modification

Another method which calculates axial flux profiles to improve homogenization is a modified form of NEM [50, 51]. To improve the accuracy of nodal calculations, NEM assumes a quartic polynomial for the intra-nodal flux shape to calculate interface currents. For the PRN, this quartic shape can can also be used to calculate improved cross-sections. This is convenient because this axial profile must already be calculated when using NEM, so the only additional calculation is to integrate the shape over the rodded and unrodded region and use the result to homogenize the cross-sections. Because Legendre polynomials are used for the flux expansion, these integrals can be calculated analytically, resulting in negligible increase in computational cost.

4.2.1.6 Equivalent-Node Method

The Equivalent-Node Method [52] attempts to calculate the PRN cross-sections as if the PRN were modeled as two separate nodes: one fully rodded and one fully unrodded. This method sets up two different diffusion problems for the PRN: on with a single region and one with two regions. This is done in combination with NEM for each of the spatial moments being calculated. The cross-sections in the case of the single region (homogenized cross-sections) are formulated in terms weighting factors multiplied by the heterogeneous cross-sections. It is then enforced that the integral of the reaction rates for the one- and two-region problems are equal. This preserves the reaction rates of the two-node problem in the homogenized problem and provides correction factors for the higher-order spatial moments. Solving these equations produces accurate, smoothly varying homogenized cross-sections.

4.2.1.7 Inverse Spectral Index Method

With increased computing power, interest grew in performing full-core calculations with finer spatial discretizations by perform nodal calculations using homogenized pins instead of homogenized assemblies or quarter assemblies. However, with the smaller spatial regions, the radial leakages become more important than before, causing many of the older rod decusping methods to become inaccurate. In 2004, Yamamoto developed the Inverse Spectral Index (ISI) method to address rod cusping for a pin-by-pin nodal calculation [53].

To develop this method, it must be noted that if the axial leakage is small compared to the radial leakage, the flux spectrum in a cell will be similar for an assembly calculation or a full-core calculation. When performing assembly calculations to generate homogenized cross-sections, one quantity which can be tabulated is the spectral index, defined as the ratio of the fast flux to the thermal flux in a pin. Because the fast flux is smoothly varying across the core compared to the thermal flux, it can be accurately approximated using the PRN's neighbor nodes. The spectral index from the assembly calculations, which explicitly modeled the rodded and unrodded regions, can then be used with the fast flux to calculate the thermal flux. A flux-volume weighting is then used to average the rodded and unrodded cross-sections for the following iteration. Because the fast flux is affected by the PRN cross-sections, iterations are performed to converge the flux and cross-sections together

4.2.1.8 CIAMA Nodal Method

Recently, a new nodal method known as Channel-wise Intrinsic Axial Mesh Adaptation (CIAMA) has been developed [54, 55]. This method eliminates all rod cusping effects im-

plicitly by eliminating the traditional constraint of other nodal methods: all nodes must be homogeneous. With this requirement removed, partially inserted control rods can be handled implicitly by the method, without need for any auxiliary calculations or corrections.

To do this, a sub-plane–like scheme is employed in the NEM formulation. As with a traditional NEM formulation, the nodes are coupled through transverse leakage terms. However, within each node, a refined heterogeneous axial mesh is used. The transverse leakage terms are still calculated on the coarse mesh with a quadratic polynomial fit. For each node, this polynomial can simply be integrated over the height of the sub-nodes to determine the sub-node leakage source. Because the inter-node coupling is done on the coarse mesh, the axial sub-mesh can be unique for each node. This allows the method to implicitly handle axial heterogeneities with minimal increases in runtime.

4.2.2 2D/1D Codes

Unfortunately, moving away from nodal methods to higher fidelity transport codes does not eliminate the rod cusping problem, as shown in Figure 4.2. There have not been as many 2D/1D codes as there have been nodal codes, but each of them still had to contend with this problem. This section will discuss some of the different 2D/1D codes that have been developed and how the rod cusping problem was dealt with in each of them.

4.2.2.1 Neighbor Spectral Index Method

The code CRX-2K [19] uses the 2D/1D fusion method to perform LWR simulations. To address the rod cusping problem in this code, a modified version of the ISI method is applied, called the Neighbor Spectral Index (NSI) method. The NSI method uses the same methodology as the ISI with one modification. Because the 2D/1D fusion method does not require standalone calculations to generate homogenized cross-sections, the spectral index must be calculated on-the-fly during the 2D/1D iteration. To do this, the neighboring node above the PRN is used to obtain a rodded spectral index, and the neighboring node below the PRN is used to obtain an unrodded spectral index. These two indexes are then used with the rodded and unrodded cross-sections and volume fractions to obtain homogenized cross-sections for the PRN. This method causes significant improvements over other decusping methods and requires negligible additional computation since the fluxes are stored for every region already.

4.2.2.2 Sub-Plane Decusping

Another 2D/1D code is nTRACER, which is under active development by Seoul National University [28]. To address rod cusping effects in nTRACER, Jung and Joo developed a more general method than the polynomial correction method used by MPACT [29]. This method pre-generates correction factors at the start of a simulation, rather than relying on hard-coded corrections. To do this, the assembly that will have a partially inserted control rod is identified, and a single-plane 3x3 assembly problem is set up using the partially-rodded assembly and its neighbors. The radial and axial cusping effects are then determined separately. First, the radial cusping effects are determined by performing 2D MOC calculations on the 3x3 sub-domain with the rod fully inserted and fully withdrawn. This provides radial flux profiles in the rodded assembly for both rodded and unrodded regions, as well as current coupling coefficients for CMFD for the rodded and unrodded CMFD nodes. Once this is done, the rod is simulated at positions of 25%, 50%, and 75% withdrawn from the plane. To reduce runtime, these calculations are done using only 3D sub-plane CMFD, using the heterogeneous rodded and unrodded cross-section for the appropriate sub-planes. This generates axial flux profiles for the full MOC plane for each of the possible rod positions. During the full-core 2D/1D calculation, these axial flux profiles are then used to generate improved homogenized cross-sections for the MOC calculation using equation.

$$\overline{\Sigma_i} = \frac{\phi_{rad,i}^R \phi_{ax,i}^R \Sigma_i^R h^R + \phi_{rad,i}^U \phi_{ax,i}^U \Sigma_i^U h^U}{\phi_{rad,i}^R \phi_{ax,i}^R h^R + \phi_{rad,i}^U \phi_{ax,i}^U h^U} \ . \tag{4.2}$$

4.2.2.3 Polynomial Corrections

MPACT currently uses a simple polynomial correction to the volume fractions used to homogenize the control rod [56]. To develop this method, a 3x3 assembly was set up with a control rod in the center assembly. This problem was simulated with the rod tip at nine different positions in the plane. These simulations were then repeated, but with the axial mesh refined so the rod tip aligned with a plane boundary. The k_{eff} differences between the two sets of simulations were fitted with a sixth-order polynomial which is used in MPACT to reduce the volume fraction of the control rod by an appropriate amount to offset the cusping effects. This process was repeated for different control rod materials such as AIC, B_4C, and Tungsten, since each material has unique cross-sections. This method has the advantages of being simple to implement and requiring no increase in computational requirements. However, the results obtained from this decusping method are tied to the control rod material and reactor model used to develop the corrections, limiting its usefulness to a small subset of reactors.

4.3   Recent Decusping Methods Development

This section discusses two new decusping treatments added to MPACT as part of this work. These methods were developed to try to improve on the accuracy and flexibility of MPACT's polynomial corrections without significant runtime increases. Both of these methods rely on the sub-plane scheme described in section 3.3.1. The first method modifies the sub-plane scheme to use heterogeneous cross-sections to capture the axial effects of the partially inserted rod. The second method builds on this by also capturing some of the radial effects of the rod.

4.3.1   Sub-plane Decusping

In section 3.3.1, the sub-plane scheme was introduced as a means of coarsening the 2D/1D axial mesh to improve runtime while maintaining accuracy. However, one negative consequence of using the sub-plane scheme is that it increases the likelihood of partially inserted rods being present since there are fewer, thicker MOC planes. To address this, two modifications were made to the sub-plane scheme to minimize cusping effects. The resulting method shares some similarities with the one employed by nTRACER discussed in section 4.2.2.2. However, this method is built directly into the 2D/1D iteration scheme, requiring no additional MOC calculations to inform the decusping calculations.

The first modification that was made was to modify the cross-sections used for the sub-plane CMFD and axial SP$_3$ calculations. In the most basic form of the sub-plane scheme, the same cross-sections and radial coupling coefficients are used for all sub-planes in each MOC plane. To use sub-plane as a decusping method, the requirement for constant cross-sections was removed. Thus, when Equation 2.35 is applied, explicit rodded or unrodded cross-sections are used for the cross-section homogenization in each sub-plane, rather than a single homogenized cross-section being using in all sub-planes. This allows the CMFD and axial solvers to capture the sharp flux gradients that occur around the edge of the control rods.

In order to improve the MOC calculations as well, the CMFD flux projection described in Equation 2.40 is extended to also include a cross-section projection for the partially rodded pin cells. This is done using Equation 4.2 with $\phi^R_{rad,i} = \phi^U_{rad,i}$. This allows the 2D MOC calculations to also capture some of the axial effects of the partially inserted rod.

4.3.2 Auxiliary 1D Collision Probabilities

While the sub-plane decusping method is better than none, it captures only the axial effects of the partially inserted rod. To also capture the radial effects, an additional radial calculation is needed to obtain radial flux profiles for the rodded and un-rodded regions. In MPACT, this is done using a 1D Collision Probabilities (CP) solver.
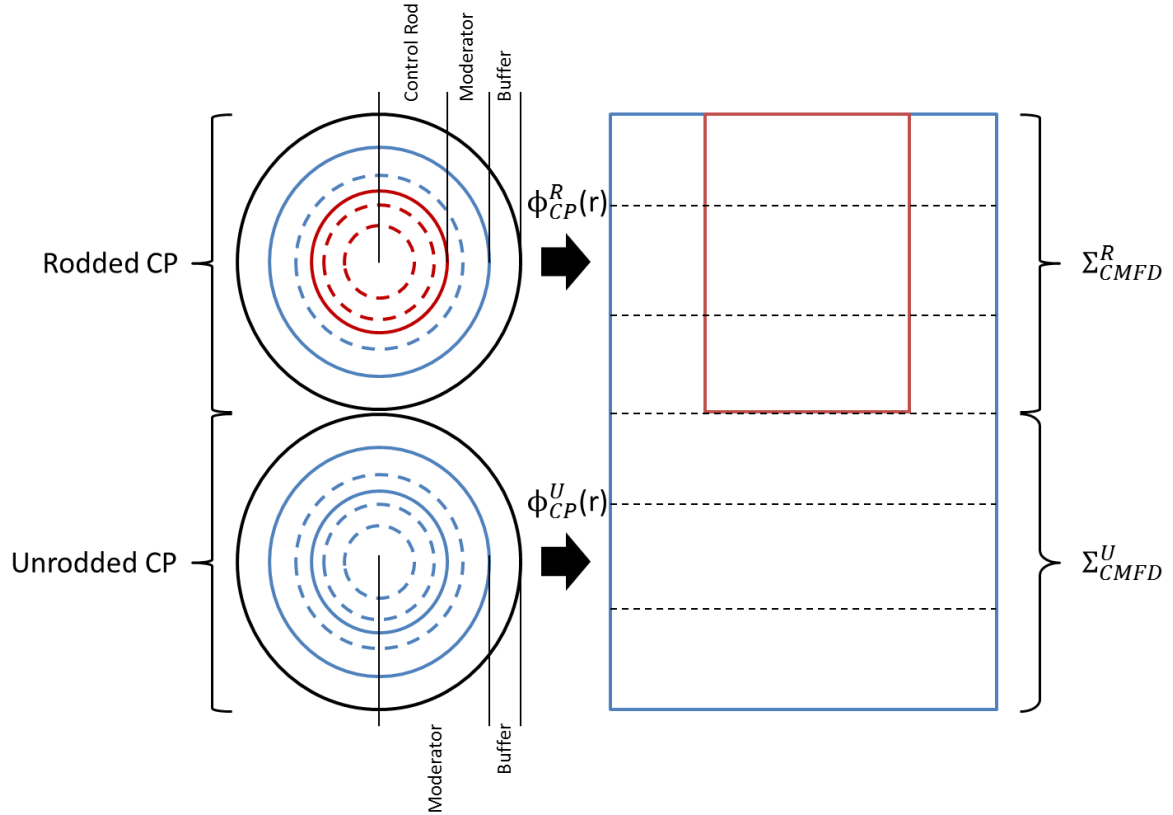

Figure 4.3: Illustration of 1D Collision Probabilities rod decusping method

After the CMFD homogenization, but prior to the calculation itself, a 1D CP kernel is solved for the rodded and un-rodded regions to obtain a radial flux profile. This calculation uses the MOC cross-sections in each radial zone, along with a buffer region that consists of the eight neighboring CMFD cells homogenized together. These profiles are normalized so that the volume-average of each profile is exactly unity. These profiles are then used with the sub-plane CMFD flux and material cross-sections to calculate improved homogenized cross-sections, as shown in Equation 4.3. Homogenizing the cross-sections in this way allows CMFD to capture both axial and radial effects of the partially inserted rod with

negligible increase in computational expense. This process is illustrated in Figure 4.3.

$$\Sigma_{g,i} = \frac{1}{\phi_{g,i}V_i} \sum_{r=1}^{N_{rings}} \Sigma_{g,r}\phi_{CMFD,g,i}\phi_{CP,g,r}V_r \; . \tag{4.3}$$

After the CMFD calculation, the MOC cross-sections must be modified as with the sub-plane decusping. This is done using by using Equation 4.2 directly, where the radial flux terms are the projected MOC fluxes and the axial flux terms are the results of the sub-plane CMFD calculation that used the CP-homogenized cross-sections. The projection of the CMFD flux to the MOC mesh is unchanged from the standard sub-plane scheme. The CMFD calculation flow when using 1D CP is shown in Figure 4.4
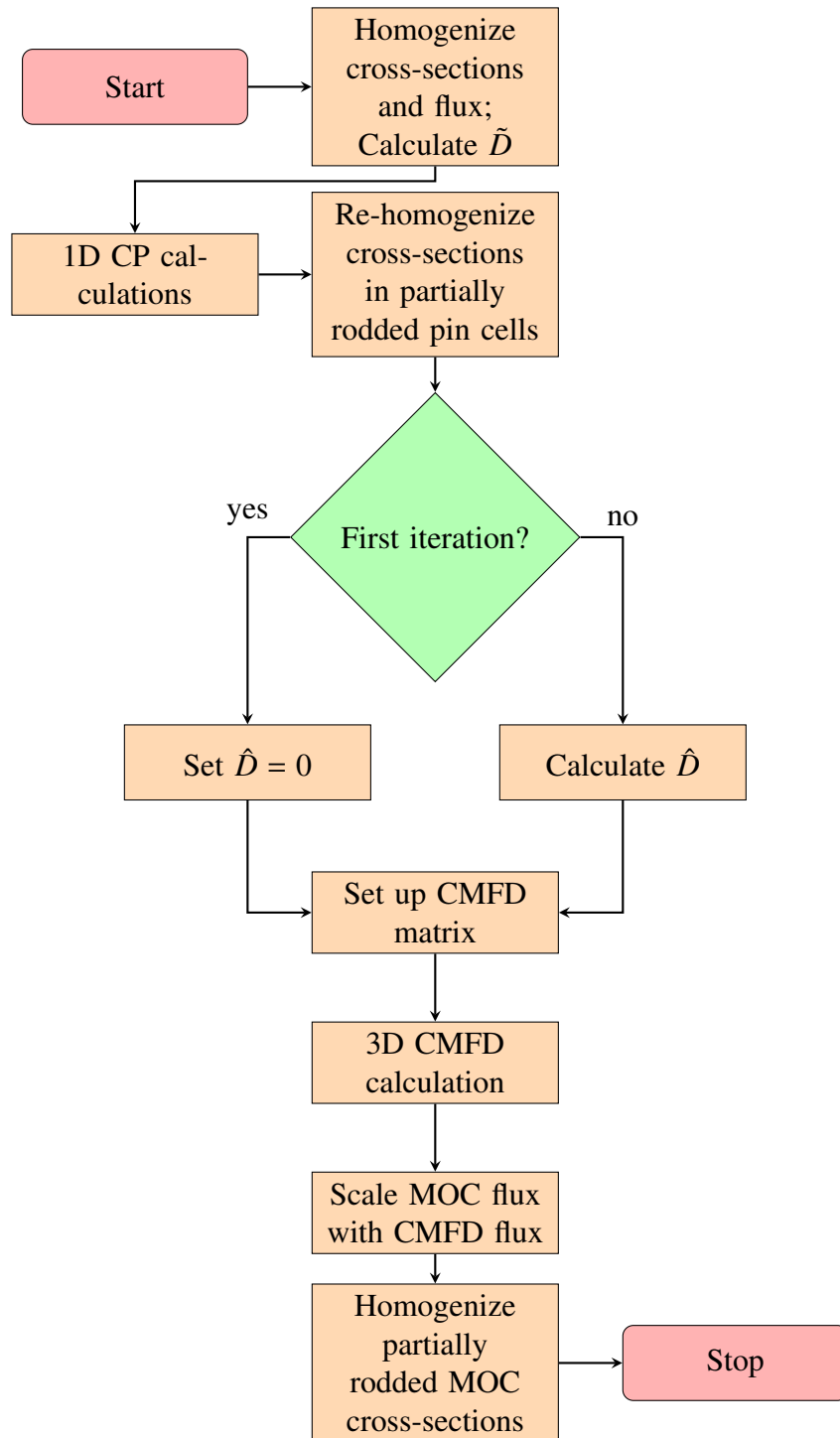
Figure 4.4: Calculation flow for CMFD calculation in MPACT with 1D CP decusping treatment

CHAPTER 5

Results

To test MPACT's newly developed decusping methods, VERA Progression Problems 4 and 5 [57] were used. These problems are based on Watts Bar Unit 1, and provide realistic test cases for the 2D/1D method. The control rods and meshing were modified slightly from the specifications to introduce rod cusping effects which may not normally be there. However, these changes will be useful in identifying the effects of cusping and improvements from each decusping method for each of the two different problems.

5.1   VERA Problem 4

Problem 4 is composed of a 3x3 set of assemblies, with a control bank in the center assembly. The radial layout of the problem is shown in Figure 5.1(a), and the axial layout of each assembly is shown in Figure 5.1(b). The control rods were placed at an axial elevation of 257.9 cm above the core plate, about one third inserted into the core. The rod in the original problem specification is made of AIC with a B_4C follower and a stainless steel tip. However, to simplify analysis of the results, this was changed so the rod was a single AIC region.

For the reference solution, 58 MOC planes were used. It was also ensured that the end of the control rods were exactly aligned with one of the MOC plane boundaries. The cases using decusping methods used the same mesh, but with the 2 MOC planes around the tip of the control rod merged into a single plane to introduce cusping effects. The accuracy and convergence data for these cases are shown in Table 5.1

The "No Treatment" case shows the magnitude of the cusping effects for this problem. The k_{eff} difference is 30 pcm, which is not too alarming. However, the RMS and maximum power differences are almost 4% and over 20%, respectively, which is an unacceptable level of error. The polynomial decusping significantly reduces these errors to about 1% and 6.5% respectively. This is much better, but still quite high. The sub-plane decusping
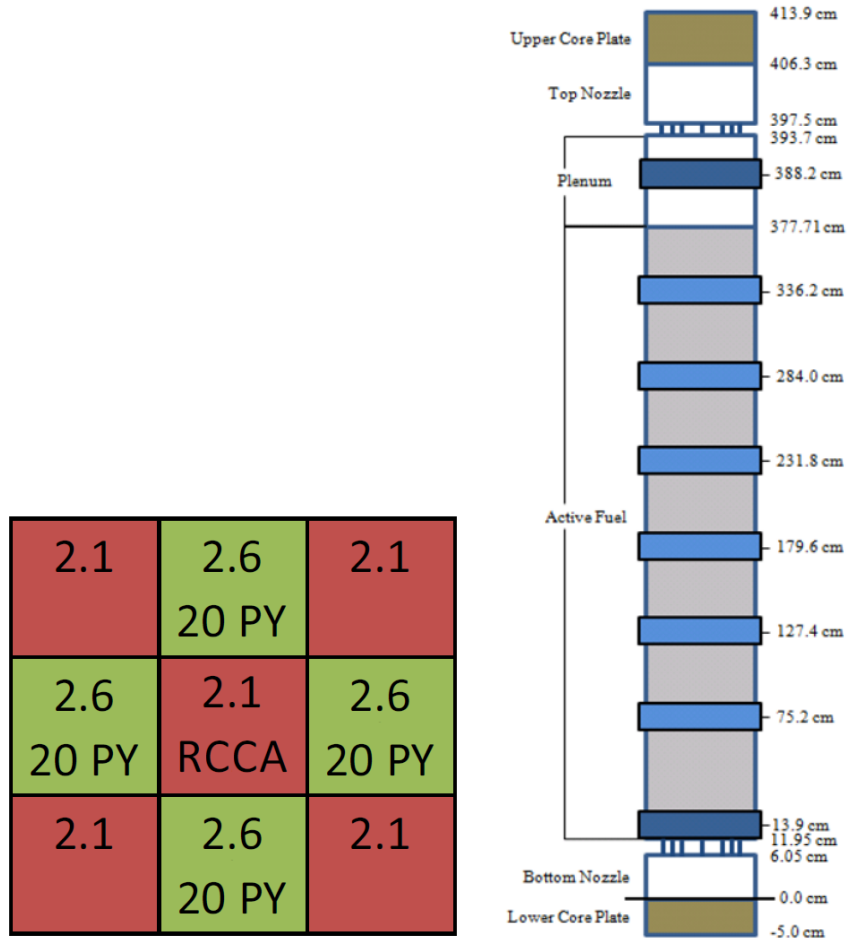
Figure 5.1: VERA Problem 4 radial (left) and axial (right) layouts

with no radial treatment performs similarly to the polynomial decusping, but slightly worse. Because the polynomials were generated using this problem, it is expected that they would perform well. Thus, the fact that the sub-plane decusping is comparable indicates that it is capturing the axial shape well. Finally, the CP-based decusping gives the best results, with an RMS of about 0.5% and a maximum error just under 5%. The maximum error is still larger than what is typically desired from the 2D/1D method, but is better than the old decusping treatment and far better than no treatment at all.

As far as runtime is convergence and runtime is concerned, all cases took the same number of 2D/1D iterations. The sub-plane–based decusping methods incurred a few more CMFD iterations, but this did not have a significant impact on runtime. The difference in runtime between the reference and no treatment cases is due to the decomposition of the problem. Both cases use the same sub-plane CMFD mesh (with homogeneous cross-sections in each plane). However, the reference case was run with two MOC planes instead

Table 5.1: VERA Problem 4 Decusping Results

Case	k-eff Difference (pcm)	Pin Power Differences		Iterations		Runtime (Core-Hours)
		RMS	Max	2D/1D	CMFD	
Reference	–	–	–	12	364	8.59
No Treatment	-30	3.84%	21.81%	12	352	9.23
Polynomial	-8	1.03%	6.58%	12	360	9.50
Sub-plane	-7	1.13%	7.11%	12	409	9.26
Sub-plane + 1D-CP	-2	0.54%	4.94%	12	364	9.45

of one. This also means that an extra core was used in the calculation. Because of this, the time required for the transport calculations was the same, but the CMFD solve was slower for the no treatment case since a single core was solving a portion of the CMFD system handled by two cores in the reference calculation. Thus, the runtime increase is due only to the parallel partitioning, not the methods themselves. This is important, because the runtimes of the decusping cases were all about the same as the no treatment case. This implies that the runtime penalty due to the decusping solvers is negligible. Some work simply needs to be done to improve the parallel balance when using the sub-plane scheme.

5.2   VERA Problem 5

To demonstrate the behavior of the decusping methods on a full-core problem, VERA Problem 5 was also run. Problem 5 is the a beginning-of-cycle simulation of the Watts Bar Unit 1 PWR. The model of this reactor uses the same axial layout shown in Figure 5.1(b) with the radial layout shown in Figure 5.2. For these calculations, Bank D was set to a position of 257.9 cm above the core plate while all other banks were fully withdrawn to 383.3125 cm, about 6 cm above the top of the active fuel.

Like problem 4, the reference case was run with 58 planes while the decusping cases were run with 57 planes. Radial decomposition was used with 16 cores per MOC plane. This resulted in a slightly different number of cores for the reference case compared with the others, as seen in the Problem 4 calculations. The accuracy and convergence results for the partially rodded plane are shown in Table 5.2.

As with Problem 4, we see that the CP results are the best, with less than 3% maximum error in the roded plane. The maximum errors occur in the partially rodded assembly, as expected. The sub-plane decusping without the CP treatment is worse than in Problem 4, showing the importance of correctly treating the radial effects. The runtime increase is also between 15% and 20% for the sub-plane and CP decusping methods. Some of this is due to parallel imbalance in the CMFD systems as discussed in Problem 4, but some is also due
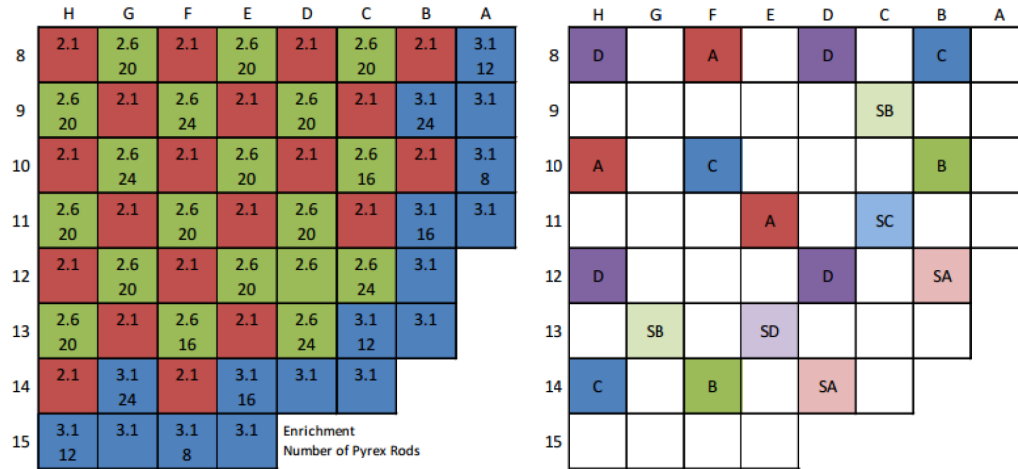
Figure 5.2: VERA Problem 5 radial layout

Table 5.2: VERA Problem 5 decusping results for the partially rodded plane

Case	k-eff Difference (pcm)	Pin Power Differences		Iterations		Runtime
		RMS	Max	2D/1D	CMFD	(Core-Hours)
Reference	–	–	–	13	481	361.7
No Treatment	-22	6.90%	30.55%	13	523	410.7
Polynomial	-5	1.15%	4.85%	13	463	373.7
Sub-plane	-5	2.09%	10.20%	13	499	399.0
Sub-plane + 1D-CP	-1	0.50%	2.74%	13	529	425.6

to the increase in CMFD iterations when using these methods. Overall, the new methods perform well. However, nearly 3% error is still higher than desired when using the 2D/1D method, so there is still room for improvement.

CHAPTER 6

Proposed Work

Moving forward, there are several improvements that can be made to the 2D/1D scheme with regard to the cusping problem. Four of these improvements will be discussed in this section. The first two are modifications to methods already in MPACT, while the second two would involve implementation of new solvers.

6.1   Improvements to Current Decusping Techniques

The sub-plane–based decusping solvers presented in chapters 4-5 are capable of reducing the cusping errors significantly over having no treatment. They also provide a more general solution to the problem than the polynomial corrections that existed in MPACT previously. However, there are at least two improvements that can be made going forward. The first deals with approximations made in the calculation of the radial currents, and the second deals with the auxiliary solver used to generate the radial flux profiles.

6.1.1   Radial Transverse Leakage Source

The current implementation of the sub-plane scheme assumes the same $\hat{D}$ current correction term for all sub-plane surfaces between two pin cells. This assumption is made to improve the stability of the sub-plane scheme, but certainly introduces some error in the CMFD and SP$_3$ calculations. In most cases, this error is likely quite small, since the axial shape of the solution usually changes gradually. However, when a control rod is partially inserted into an MOC plane, the currents in the rodded sub-planes are much different from those in the unrodded sub-plane. The CMFD calculations predict that this is the case, but using the same $\hat{D}$ for both of these regions likely introduces a more significant error than for other regions of the problem. Moving forward, work will be done to develop sub-plane–dependent $\hat{D}$ terms to improve the accuracy of the radial TL source for the SP$_3$ solver.

6.1.2 Auxiliary Solver

Another potential source of error for the decusping treatments discussed here is in the choice of auxiliary solver. The 1D CP kernel was useful for capturing some of the radial cusping effects, but it does have some deficiencies. First, because it is 1D, it ignores the corner effects of the pin cell by treating the moderator as a ring. This minimizes the directional dependence of the neutrons, changing the radial flux profile generated by the solver. Second, the 1D CP solver relies on a buffer region obtained by homogenizing the surrounding pin cells. As with the moderator region, this buffer region is assumed to be annular and homogeneous. This affects the source term which drives the problem and the behavior of the neutrons which escape the pin cell of interest. Third, the CP method is inherently unable to treat anisotropic scattering, which will be especially important around a control rod. While transport-corrected scattering can capture some of these effects, the accuracy of transport-correction methods is limited. To eliminate these problems, a different method could be used in place of the 1D CP solver.

One alternative method that would resolve both of the deficiencies mentioned here would be a highly optimized 2D MOC solver. Using 2D MOC, an octant-symmetric pin cell could be simulated quickly, capturing the effects of the corner on the flux in the pin cell. Furthermore, the boundary conditions could be handled several different ways. The first would be to simply have an isotropic incoming angular flux calculated from the radial currents generated by the planar MOC calculations. Another option would be to explicitly store the incoming angular fluxes along the surfaces of the partially rodded pin cell during the planar MOC calculation. A third way of dealing with the boundary conditions would be simulate the neighboring cells as well. Doing this would allow use of isotropic angular flux boundary conditions on the neighbor's boundary while still having an angular shape upon reaching the partially rodded cell. This third option would allow the 2D MOC solver to improve on the 1D CP solver without any modifications to the 2D planar MOC calculations.

6.2 Sub-Ray Method of Characteristics

While the current methods can be improved in the near term, they still are not able to fully resolving the rod cusping issue in the long run. This section will describe some of the circumstances in which the decusping techniques described in this work will either provide inaccurate results or fail altogether. After this, some results will be presented from a 1D MOC code to show the effects of cross-section homogenization on the angular flux. Finally,

a new way of performing MOC calculations will be presented. This method will provide a fully general way of dealing with axial material heterogeneities in 2D/1D by directly addressing the angular flux, which is a more fundamental quantity than the scalar flux. Additionally, a timeline for the development, implementation and testing of this method will be included.

6.2.1 Motivation

There are two reasons to pursue a more general, advanced decusping technique. The first is that there are some components which will be more difficult to model using the sub-plane–based CP decusping method. The PWR control rods used in the Watts Bar plants and other similar reactors fit regularly into the lattice of fuel pins, resulting in a rodded pin cell of similar shape and size. However, some plants use larger absorber rods which take up multiple positions in the lattice. Furthermore, BWRs use control blades, which are located between fuel assemblies and do not have a convenient cylindrical shape like the PWR control rods. Modeling components such as these using the current decusping techniques would be difficult or impossible.

A second reason to develop a more advanced method is to address the cusping effects on a more fundamental level. Current techniques in MPACT involve correcting homogenized cross-sections or scalar fluxes after a calculation. Techniques employed by other codes, such as nTRACER, discussed in section 4.2.2, directly perform MOC calculations to treat the cusping effects. However, these calculations must be done up front, in addition to the regular 2D/1D calculation. A more desirable method would be able to deal directly with the effects of the axial heterogeneity on the angular flux during the 2D/1D iteration scheme without performing any additional calculations, producing accurate answers for general geometries with minimal increase in computational expense.

6.2.2 1D MOC Results

To analyze the behavior of the angular flux in relationship to homogenized cross-sections, a 1D MOC code was written that could solve eigenvalue and fixed source problems. This code can solve small test problems quickly while making it much easier to obtain detailed solution data, as opposed to trying to extract angular flux data from MPACT. This section will briefly explain the code and how it performs its calculations and the test problem that was used, then present results from two different types of calculations.

6.2.2.1 Code and Problem Description

The code is set up to take in a description of pins and materials to be used for the calculations. For the geometry, a pin pitch is specified which is used for all pins. Each pin consists of a list of radii. Assuming a square pin cell, these pins are then transformed from cylindrical geometry to slab geometry while preserving the volume fraction of each material. Thus, the thickness of the pin slab is equal to the pin pitch, but the width of the fuel material will not be equal to input radius, since the volume fractions are preserved. One material is then specified for each region, though each material region is divided into many sub-regions for the MOC calculations. These materials are defined by a separate cross-section library file. This file uses the "user library" format supported by MPACT, which allows the user to put in macroscopic cross-sections for absorption, nu-fission, kappa-fission, chi, and scattering moments. For all these calculations, the C5G7 benchmark cross-sections [58, 59] were used. These cross sections are included in Appendix B.

For the MOC sweeps, a Gaussian quadrature [34] is used with 2, 4, 8, 16, or 32 polar angles, with half of the angles being used in each direction. The MOC sweeps are done similarly to how they are done in MPACT, with the loop over energy groups being the innermost loop. The code can be run as either a fixed source solver or an eigenvalue solver. For the eigenvalue mode, power iteration is used after each MOC calculation to determine and updated k_{eff}. The fixed source mode can be used to run either a specified number of iterations or to run until the scattering source is converged below some tolerance. This allows some flexibility on exactly what kinds of results can be obtained.

The problem used for these calculations was a 1D variation of VERA Problem 4. The center row of pins across all three assemblies was pulled out and used for the 1D model, resulting in a row of 3×17 pins with a pin pitch of 1.26 cm (the inter-assembly gap was neglected for this model). The center assembly had 4 guide tubes in it which contained a mixture of moderator and control rod to represent a partially inserted control rod. These partially rodded locations were the only part of the problem that had any material changes. This allowed the effects of the cross-section homogenization to be isolated for each calculations.

6.2.2.2 Specified Total Source

The first set of calculations performed were done using a specified total source. To do this, the guide tubes were filled with 50% control rod and 50% moderator by volume fraction, and a full eigenvalue calculation was performed. The source distribution from this calculation (both fission and scattering source) were then passed to the the fixed source solver.

A single iteration was run using this source on three different variations of the problem: the 50-50 mixture, fully rodded, and fully unrodded. Because the multi-group source is set up before performing any MOC sweeps, this resulted in all three of those calculations having an identical source for the MOC sweep. The only difference between them was the cross-sections used in the guide tubes.

Figure 6.1 shows the scalar flux resulting from these three calculations. The most important thing to note in this data is that the effects of the rod are very local in the MOC calculation. Moving through the rodded pin cell, the rodded, unrodded, and mixed cases have converged back to the same shape by the time the edge of the pin cell is reached. This indicates that whatever treatment is used for the partially rodded cell likely does not need to worry about interference between neighboring rods because the effects are so localized.
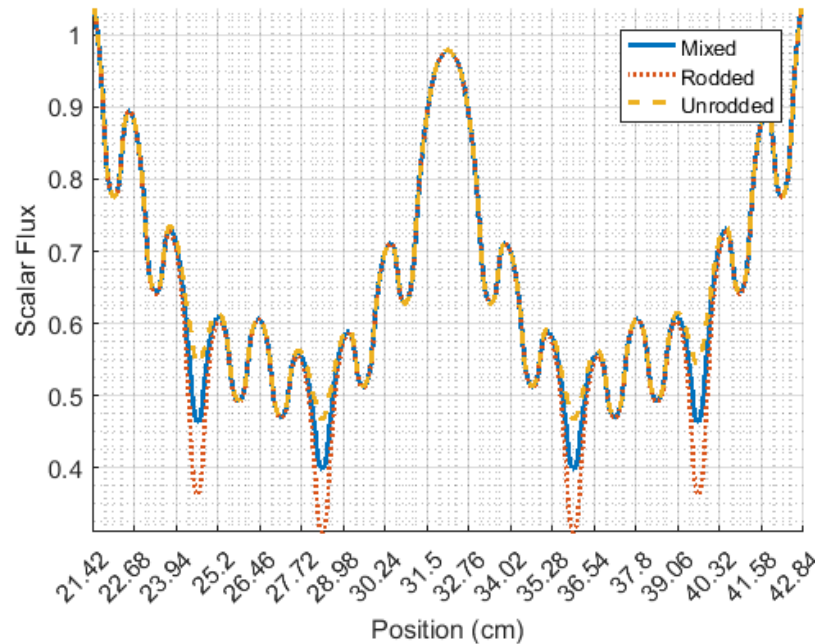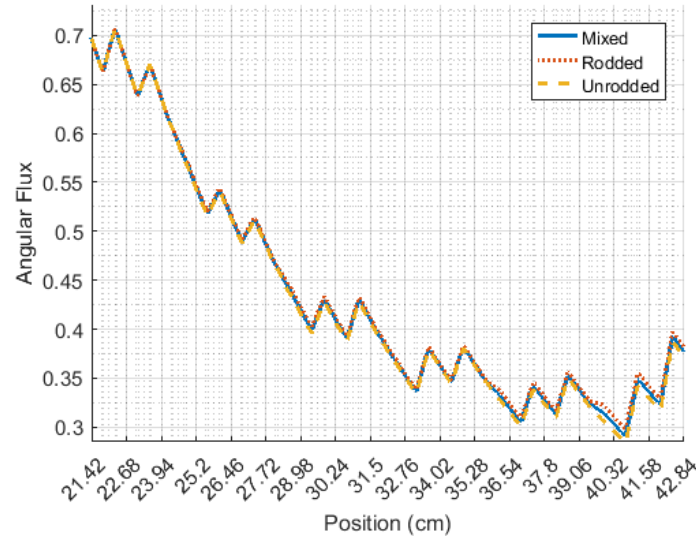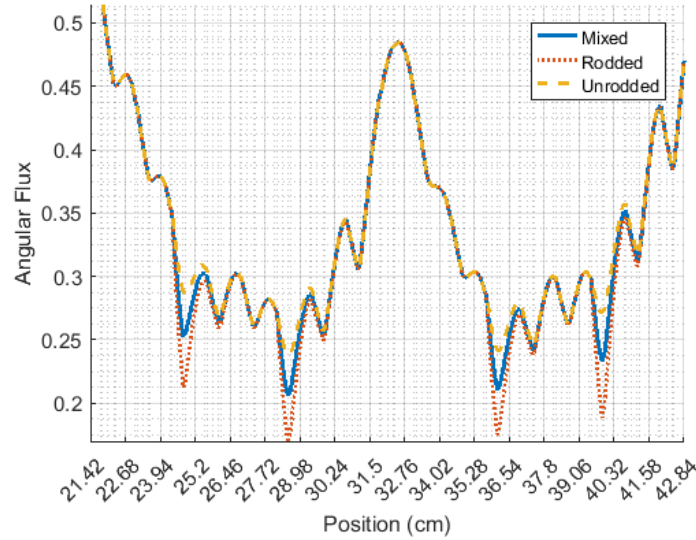

Figure 6.1: Group 7 scalar flux comparisons for a fixed fission and scattering source calculation

Figure 6.2 shows the right-going angular flux in groups 1 (fast) and group 7 (thermal). The group 7 angular flux behaves similarly to the group 7 scalar flux in that the effects are localized around each rod. The three different angular flux shapes are still somewhat different at the edge of the neighboring pin cell, but have pretty much converged upon reaching the clad and fuel. The reason for this is that the mean free path of thermal neutrons is small. The total group 7 cross-section in the moderator is about 2.65 cm^{-1}, which corresponds to a mean free path of about 0.38 cm, which is less than one third of the pin pitch for a typical PWR. Because of this, the differences between the rodded and unrodded cases are washed

out quickly if the source distribution is kept the same between the two calculations.

(a) Group 1

(b) Group 7

**Figure 6.2: Angular flux comparisons for a fixed fission and scattering source calcu-
lation**

The same cannot be said for the fast flux. The mean free path of the fast flux is about
6.3 cm, which is the width of five pin cells. Thus, it can be seen that each of the control
rods after the first builds on the effects of the previous control rod. While the fast flux does
not have a significant impact on the fission source distribution, it does impact the scattering
source distribution, which is not shown in these results.

6.2.2.3 Fixed Fission Source
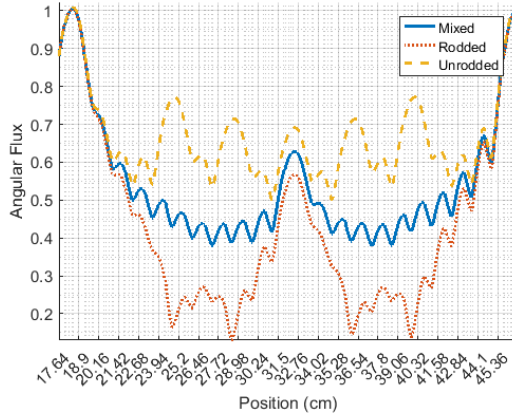
(a) 25% Mixture
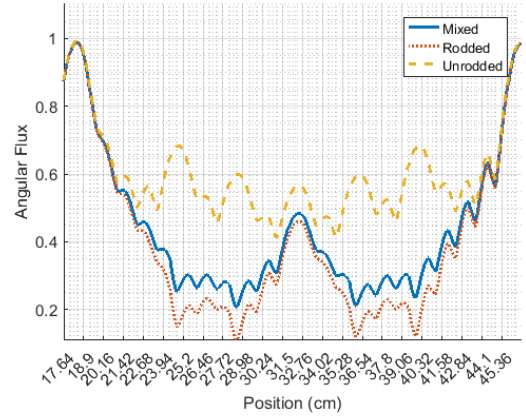
(b) 50% Mixture

(c) 75% Mixture

Figure 6.3: Group 7 angular flux comparisons for 25% and 75% mixtures

The second set of calculations that was performed used a fixed fission source, but allowed the scattering source to fully converge for each calculation. As in the previous section, an eigenvalue calculation was completed using partially rodded cross-sections. This was done for 25%, 50%, and 75% rodded cases. For each case, a fixed fission source calculation was done with the fully rodded and fully unrodded cross-sections. This time, multiple iterations were allowed for each material to converge the scattering source. This allows us to see the effects of the rod on the scattering source distribution without worrying about changes in the eigenvalue and fission source distribution.
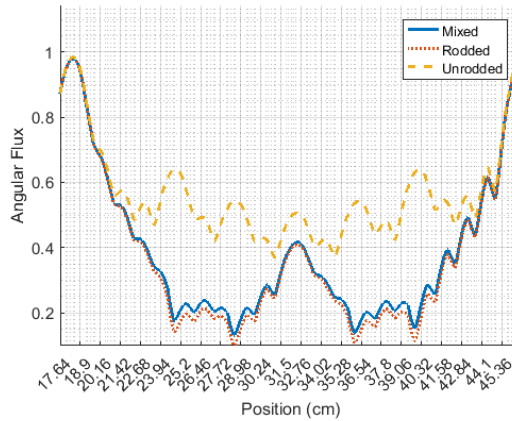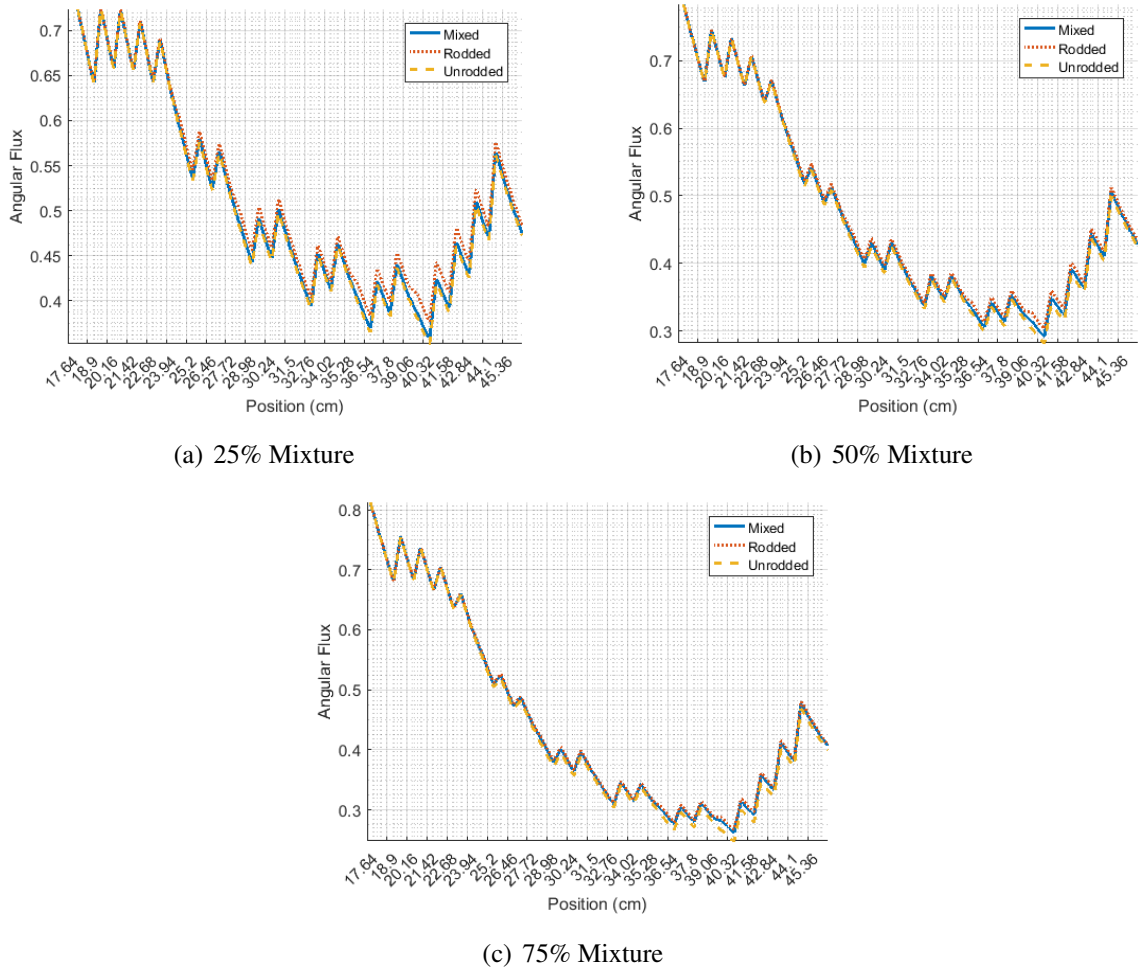
(a) 25% Mixture

(b) 50% Mixture

(c) 75% Mixture

Figure 6.4: Group 1 angular flux comparisons for 25% and 75% mixtures

Figure 6.3 shows the group 7 angular flux comparisons for each of the three mixtures. We immediately see that for each of them, the angular flux for the mixture is closer to the rodded result than the unrodded result than what might be expected based on the volume fraction of the rod. For example, comparing Figures 6.3(b) and 6.2(b) shows that the angular flux is much closer to the rodded solution than when the scattering source was fixed, despite the small mean free path of thermal neutrons. Figure 6.4 shows the same comparisons for the group 1 angular flux. While the difference in magnitude between each of the three cases is small for any mixture, the long mean free path means that these differences get spread out over a larger area. This changes the shape of the scattering source for the thermal groups, which then has a more widespread effect on the the scalar flux distribution. It should be noted as well that all angular flux plots shown here are for the flattest polar angle. Since the steeper angles travel through more material in each region, the differences between them go away more quickly. Thus, the angle shown in these plots is the one having

the largest impact on the solution.

6.2.3   Sub-Ray MOC Method Description

The 1D MOC results tells us that the effects of changing the rodded cross-section has pretty local effects on the flux itself, but the source distribution that results from these small changes in the flux then leads to significant differences in the results. To address this, a new "sub-ray" MOC method will be developed. Currently, the angular flux resulting from MPACT's MOC calculations for a partially rodded plane are simply the results of a regular MOC calculation on volume-homogenized cross-sections. The new method will calculate two separate angular fluxes using the heterogeneous cross-sections, then combine the angular fluxes instead of the cross-sections. This should provide a more accurate result for the scalar flux calculation, which will in turn improve the source calculations in the upcoming iterations as well.

This method will still make use of the sub-plane scheme for the CMFD and SP$_3$ calculations. As before, sub-planes will allows the homogenized meshes to capture the axial shape of the flux. However, they will also be beneficial to the MOC solver as well. The rodded portion of the MOC plane will generally have a faster spectrum than the unrodded region due to absorption and a weaker ability to thermalize neutrons compared to the moderator. Thus, two difference sources must be used in the sub-ray MOC calculations in addition to heterogeneous cross-sections. The sub-plane scheme will provide a way to calculate those sources in the partially rodded regions.

It is clear that this method should improve the results of the calculation. However, one of the most important outcomes of this must be that the increased computational expense must be low, otherwise there are other decusping methods that could be used, such as simply refining the axial mesh. A single 2D assembly in MPACT (VERA Problem 2a) has 715,675 ray segments. With a ray spacing of 0.05 cm and 16 azimuthal angles, these segments should be close to uniformly distributed amongst the 289 pin cells in the problem. This gives an average number of segments of about 2,475 segments per pin. For a partially rodded pin cell, we can assume that every segment in that pin cell will have two sub-rays, effectively doubling the number of segments for that pin cell. A typical PWR such as Watts Bar has control rods in 24 positions in the lattice, so this gives a total number of segments of about 775,000 segments for a 2D assembly, or an increase of about 8% over the original total. Furthermore, while this is likely an accurate estimate of the increase in calculations, the actual runtime increase should be less than this. Because the sub-rays are not actually distinct ray segments and share length and angle data, a good implementation

of this method should be quite efficient in performing the extra calculations, as mentioned in section 3.3.3.2 concerning the Jacobi scattering source MOC sweeping algorithm.

An execution plan was developed to map out the activities required to complete the sub-ray MOC development and implementation in MPACT. This plan is shown in Table 6.1

Table 6.1: Execution plan for sub-ray MOC development and implementation

Task	Description	Target Date
1	Analysis of cross-section and source effects on angular flux	10/2016
2	Development of sub-ray MOC method	12/2016
3	Prototype of method in 1D MOC code	03/2017
4	Implementation of method in MPACT	06/2017
5	Testing on VERA Problem 4, 5, 9, and transient test problem	08/2017

CHAPTER 7

Conclusions

The theory and history of rod cusping and decusping methods were presented and used to motivate development of new methods which improved on some aspects of previous methods. These methods were tested using VERA Problem 4, with the results showing significant improvement over the previous decusping method in MPACT. Furthermore, it was shown that the decusping methods introduced negligible runtime increases.

Despite the improved results using the new decusping methods, there is still much room for improvement. Some of this improvement would be found by addressing some of the fundamental approximations of the 2D/1D method. One example of this would be through addressing the spatial and angular shapes of the axial and radial transverse leakages. This could prove especially important around a control rod tip where there are strong gradients in the shape of the flux. Other improvements relate more to the decusping methods themselves. Increasing the number of dimensions of the 1D CP solver or replacing it with a different method such as the MOC would also likely improve the results of the decusping calculations. Making these improvements should result in a significant reduction in the remaining error caused by partially inserted rods.

Lastly, a new method was introduced to address the rod cusping problem. Previous methods focus on correcting the results of a calculation and improving homogenized cross-sections, but the sub-ray MOC method would improve on this by directly accounting for the partially inserted rod when calculating the angular flux, the most fundamental quantity calculated in the 2D/1D method. Doing this should greatly improve the results of the calculations, while also generalizing the decusping method's applicability to any geometry without significant increases in runtime.

To complete this work, the plan laid out in Table 6.1 will be followed. This consists of developing the new sub-ray MOC method and prototyping it in a 1D MOC code to work out the details of the MOC implementation. From there, the method will be implemented in MPACT, using the sub-plane scheme to provide the necessary source information. The

work will be considered complete after a successful implementation of the method and thorough testing to show the effectiveness of the method. The cases chosen for this testing are VERA Progression Problems 4 (3x3 assembly), Problem 5 (steady-state full core), and Problem 9 (cycle depletion with feedback). Additionally, a rod ejection case will also be tested to show the method's effectiveness for transient problems in addition to steady-state and slowly changing depletion problems.

Appendix A

Supplemental Derivations

A.1   Simplified Spherical Harmonics

This section will provide a detailed derivation of the SP_3 equations. The SP_1 equations are easily obtained by neglecting higher order moments of the flux, and SP_N approximations of order greater than 3 are easily obtained by following the same procedure. These higher order approximations are not of interest because they were not used in this work and only marginally improve on the accuracy of SP_3.

To begin, we start with the mono-energetic planar geometry transport equation with anisotropic scattering:

$$\mu \frac{\partial \psi}{\partial x} + \Sigma_t(x)\psi(x,\mu) = \int_{-1}^{1} \Sigma_s(x,\mu,\mu')\psi(x,\mu')d\mu' + \frac{Q(x)}{2} \ . \qquad (A.1)$$

Now we assume that the angular flux and scattering cross-sections are both expanded in terms of Legendre polynomials, with these expansions truncated at the Nth term as follows:

$$\psi(x,\mu) \approx \sum_{n=0}^{N} \frac{2n+1}{2}\psi_n(x)P_n(\mu) \ , \qquad (A.2a)$$

$$\Sigma_s(x,\mu,\mu') \approx \sum_{n=0}^{N} \frac{2n+1}{2}P_n(\mu)P_n(\mu')\Sigma_{s,n}(x) \ . \qquad (A.2b)$$

Substituting these expansions into the transport equation, multiplying by $P_n(\mu)$, and integrating, we obtain the following:

$$\frac{d}{dx}\left[\frac{n+1}{2n+1}\psi_{n+1}(x) + \frac{n}{2n+1}\psi_{n-1}(x)\right] + \Sigma_t(x)\psi_n(x) = \Sigma_{s,n}(x)\psi_n(x) + Q(x)\delta_{0,n} \ . \qquad (A.3)$$

To obtain the SP$_3$ equations, we solve Equation A.3 for moments 0-3:

n=0: $\quad \dfrac{d\psi_1}{dx} + \Sigma_t(x)\psi_0(x) = \Sigma_{s,0}(x)\psi_0(x) + Q(x)$

$$\frac{d\psi_1}{dx} = Q - (\Sigma_t(x) - \Sigma_{s0}(x))\psi_0(x) , \tag{A.4a}$$

n=1: $\quad \dfrac{d}{dx}\left[\dfrac{2}{3}\psi_2(x) + \dfrac{1}{3}\psi_0(x)\right] + \Sigma_t(x)\psi_1(x) = \Sigma_{s,1}(x)\psi_1(x)$

$$\psi_1(x) = -D_0\frac{d}{dx}\left[2\psi_2(x) + \psi_0(x)\right] , \tag{A.4b}$$

n=2: $\quad \dfrac{d}{dx}\left[\dfrac{3}{5}\psi_3(x) + \dfrac{2}{5}\psi_1(x)\right] + \Sigma_t(x)\psi_2(x) = \Sigma_{s,2}(x)\psi_2(x)$

$$\frac{d\psi_3}{dx} = -\frac{5}{3}(\Sigma_t(x) - \Sigma_{s2}(x))\psi_2(x) - \frac{2}{3}\left[Q(x) - (\Sigma_t(x) - \Sigma_{s0}(x))\psi_0(x)\right] , \tag{A.4c}$$

n=3: $\quad \dfrac{3}{7}\dfrac{d\psi_2}{dx} + \Sigma_t(x)\psi_3(x) = \Sigma_{s,3}(x)\psi_3(x)$

$$\psi_3(x) = -\frac{5}{3}D_2(x)\frac{d\psi_2}{dx} , \tag{A.4d}$$

$$D_0(x) = \frac{1}{3(\Sigma_t(x) - \Sigma_{s1}(x))} , \tag{A.4e}$$

$$D_2(x) = \frac{9}{35(\Sigma_t(x) - \Sigma_{s3}(x))} , \tag{A.4f}$$

where Equation A.4c was obtained by substituting Equation A.4a into it.

Now we make the following definitions:

$$\Phi_0(x) = \psi_0(x) - 2\psi_2(x) , \tag{A.5a}$$

$$\Phi_2(x) = \psi_2(x) . \tag{A.5b}$$

Solving these equations for the 0th and 2nd moments gives the following:

$$\psi_0(x) = \Phi_0(x) - 2\Phi_2(x) , \tag{A.6a}$$

$$\psi_2(x) = \Phi_2(x) . \tag{A.6b}$$

Substituting these expressions into each of the moment equations gives the final form of the SP$_3$ equations. These forms of the 0th and 2nd moment equations were obtained by

substituting the 1st and 3rd moment equations into the 0th and 2nd moment equations.

$$\text{n=0:} \quad -D_0(x)\frac{d^2\Phi_2}{dx^2} + (\Sigma_t(x) - \Sigma_{s0}(x))\Phi_0(x) = Q(x) + 2(\Sigma_t(x) - \Sigma_{s0}(x))\Phi_2(x) , \quad \text{(A.7a)}$$

$$\text{n=1:} \quad \psi_1(x) = -D_0(x)\frac{d\Phi_0}{dx} , \quad \text{(A.7b)}$$

$$\text{n=2:} \quad -D_2(x)\frac{d^2\Phi_2}{dx^2} + \left(\frac{9}{5}\Sigma_t(x) - \Sigma_{s2}(x) - \frac{4}{5}\Sigma_{s0}(x)\right)\Phi_2(x) =$$
$$-\frac{2}{5}\left[Q(x) - (\Sigma_t(x) - \Sigma_{s0}(x))\Phi_0(x)\right] , \quad \text{(A.7c)}$$

$$\text{n=3:} \quad \psi_3(x) = -\frac{5}{3}D_2(x)\frac{d\Phi_2}{dx} . \quad \text{(A.7d)}$$

Because the odd moments have been completely eliminated from the even moment equations, Equations A.7a and A.7c can be solved by iterating between the two equations without solving explicitly for the odd moments.

A.2 Method of Collision Probabilities

A more thorough derivation of the Collision Probabilities (CP) method is conducted in this section. The derivation begins by considering the probability of a neutron reaching a vertical line in space form its point of emission or scatter. From here, the derivation follows closely with [15, 60] to eventually arrive at a linear system for the flux.

A.2.1 Derivation

To derive the transport matrix for a cylindrical pin cell, we begin by considering the fraction of neutrons from a point source which will reach a line whose closest distance to the point source is τ mean free paths. We will consider the point source to be an isotropic unit point source. The polar angle is θ and the azimuthal angle is α. The fraction of neutrons emitted into a specific direction $d\Omega$ about $\mathbf{\Omega}$ from the point source to the line in question is given by

$$\frac{d\Omega}{4\pi} = \frac{sin(\theta)\,d\theta d\alpha}{4\pi} . \quad \text{(A.8)}$$

Integrating this expression over the polar angle gives the probability that neutrons will be emitted from the source in $d\alpha$ about α and reach the line a distance of τ away:

$$\int_0^\pi e^{-\frac{\tau}{\sin\theta}} \frac{\sin\theta d\theta d\alpha}{4\pi} . \quad \text{(A.9)}$$

We are only concerned with the fraction of neutrons in $d\alpha$ which are also in $d\Omega$. The fraction of neutrons emitted into $d\alpha$ is given by $\frac{d\alpha}{2\pi}$, so if we divide the previous expression by this fraction, we obtain the probability that a neutron emitted form the source in direction $d\alpha$ about α will reach the line:

$$
\begin{aligned}
p(\tau) &= \frac{1}{2} \int_0^\pi e^{-\frac{\tau}{\sin\theta}} \sin\theta d\theta \\
&= \int_0^{\frac{\pi}{2}} e^{-\frac{\tau}{\sin\theta}} \sin\theta d\theta \\
&= Ki_2(\tau) ,
\end{aligned}
\tag{A.10}
$$

where $Ki_2(x)$ is the second-order Bickley-Naylor function. The Bickley-Naylor function can be defined as follows:

$$
Ki_n(x) = \int_0^{\frac{\pi}{2}} \cos^{n-1}\theta e^{-\frac{x}{\cos\theta}} d\theta ,
\tag{A.11a}
$$

$$
\frac{dKi_n(x)}{dx} = -Ki_{n-1}(x) ,
\tag{A.11b}
$$

$$
\int_a^b Ki_n(y) dy = Ki_{n+1}(a) - Ki_{n+1}(b) .
\tag{A.11c}
$$

Using Equations A.10 and A.11, we can now determine the probability of a neutron which escapes from region i having its next collision in region j. This probability is given by the probability of the escaped neutron reaching the first edge of region j minus the probability of reaching the second edge of j:

$$
p_{ij}(\tau,\alpha,y) = Ki_2\left(\tau_{ij} + \tau_j + \tau\right) - Ki_2\left(\tau_{ij} + \tau\right) ,
\tag{A.12}
$$

where τ_{ij} is the number of mean free paths between i and j, τ_j is the number of mean free paths across j, and τ is the number of mean free paths from the neutron's point of emission to the edge of region i. The variable y is defined along an axis in the plane of the problem perpendicular to the direction of streaming. The combination of τ and y specify a specific point in region i for each angle α.

Next, we define a strip in i of length $t_i = \frac{\tau}{\Sigma_i}$ along the streaming direction with width dy. To obtain the fraction of neutrons born in this strip that collide in j, we integrate the strip

and divide by its length:

$$p_{ij}(\alpha,y) = \frac{1}{t_i} \int_0^{t_i} p_{ij}(\tau,\alpha,y)\,dt$$

$$= \frac{1}{t_i} \int_0^{t_i} Ki_2\left(\tau_{ij}+\tau\right) - Ki_2\left(\tau_{ij}+\tau_j+\tau\right) dt$$

$$= \frac{1}{t_i} \int_0^{t_i} Ki_2\left(\tau_{ij}+\tau_i-\Sigma_i t\right) - Ki_2\left(\tau_{ij}+\tau_j+\tau_i-\Sigma_i t\right) dt \ . \tag{A.13}$$

Now we apply a change of variables $x = \tau_{ij}+\tau_i-\Sigma_i t$. Doing this, we obtain

$$p_{ij}(\alpha,y) = -\frac{1}{\Sigma_i t_i} \int_{\tau_{ij}+\tau_i}^{\tau_{ij}} Ki_2(x) - Ki_2\left(x+\tau_j\right) dx$$

$$= \frac{1}{\Sigma_i t_i}\left[\left(Ki_3\left(\tau_{ij}\right) - Ki_3\left(\tau_{ij}+\tau_i\right)\right) - \left(Ki_3\left(\tau_{ij}+\tau_i\right) - Ki_3\left(\tau_{ij}+\tau_i+\tau_j\right)\right)\right] \ . \tag{A.14}$$

This expression can now be multiplied by the fraction of neutrons in each strip and integrated over y to obtain the total fraction of neutrons born anywhere in i that stream in direction α and collide in j:

$$p_{ij}(\alpha) = \int_{y_{min}(\alpha)}^{y_{max}(\alpha)} p_{ij}(\alpha,y)\frac{t_i}{V_i}dy$$

$$= \frac{1}{\Sigma_i V_i} \int_{y_{min}(\alpha)}^{y_{max}(\alpha)} \left[\left(Ki_3\left(\tau_{ij}\right) + Ki_3\left(\tau_{ij}+\tau_i+\tau_j\right)\right) - \left(Ki_3\left(\tau_{ij}+\tau_i\right) + \left(Ki_3\left(\tau_{ij}+\tau_j\right)\right)\right)\right] dy \ .$$

$$\tag{A.15}$$

Finally, we obtain element ij of the transport matrix by multiplying by the volume and cross-section. When multiplied by ϕ_i, this gives the total contribution to ϕ_j from region i:

$$P_{ij}(\alpha) = \Sigma_i V_i p_{ij}(\alpha)$$

$$= \int_{y_{min}(\alpha)}^{y_{max}(\alpha)} \left[\left(Ki_3\left(\tau_{ij}\right) + Ki_3\left(\tau_{ij}+\tau_i+\tau_j\right)\right) - \left(Ki_3\left(\tau_{ij}+\tau_i\right) + \left(Ki_3\left(\tau_{ij}+\tau_j\right)\right)\right)\right] dy \ .$$

$$\tag{A.16}$$

Now that this probability has been derived, it can be integrated over all α to obtain the transport matrix elements for a specific geometry.

We must also handle the self-transport case, where $i = j$. Again we follow the same procedure and define the probability $p_{ii}(t,\alpha,y)$ as the probability the neutron reaches region i minus the probability it escapes region i. Since the neutron was born in region i, the first probability is 1. This gives the following expressions:

$$p_{ii}(t,\alpha,y) = 1 - Ki_2(\tau) \, , \tag{A.17a}$$

$$p_{ii}(\alpha,y) = \frac{1}{t_i} \int_0^{t_i} p_{ii}(t,\alpha,y)\,dt$$

$$= 1 - \frac{1}{\Sigma_i t_i}\left[Ki_3(0) - Ki_3(\tau_i)\right] \, , \tag{A.17b}$$

$$p_{ii}(\alpha) = \int_{y_{min}(\alpha)}^{y_{max}(\alpha)} p_{ii}(\alpha,y)\,\frac{t_i\,dy}{V_i}$$

$$= 1 - \frac{1}{\Sigma_i V_i} \int_{y_{min}(\alpha)}^{y_{max}(\alpha)} \left[Ki_3(0) - Ki_3(\tau_i)\right]dy \, , \tag{A.17c}$$

$$P_{ii}(\alpha) = \Sigma_i V_i p_{ii}(\alpha)$$

$$= \Sigma_i V_i - \int_{y_{min}(\alpha)}^{y_{max}(\alpha)} \left[Ki_3(0) - Ki_3(\tau_i)\right]dy \, . \tag{A.17d}$$

Now this self-transport kernel can be used with the kernel in A.16 to set up the full transport matrix for a problem. This matrix is dependent on the geometry of the problem, so it must be done for each unique problem being solved. The following section discusses the details of this process for a cylindrical pin cell.

A.2.2   CP in Cylindrical Coordinates

To obtain the transport matrix for a cylindrical pin cell, the pin cell must first be cylindricized. To do this, the moderator region around the outside of the fuel pin is changed to an annular ring which preserves the total volume of the cylinder. This allows the calculation to be 1D spatially. Secondly, to ensure that there is a sufficient source driving the problem, a fuel and moderator mixture can be placed in a ring beyond the moderator ring. This is especially important for using the CP method for decusping since the control rod pin cell has no fission source of its own to drive the problem.
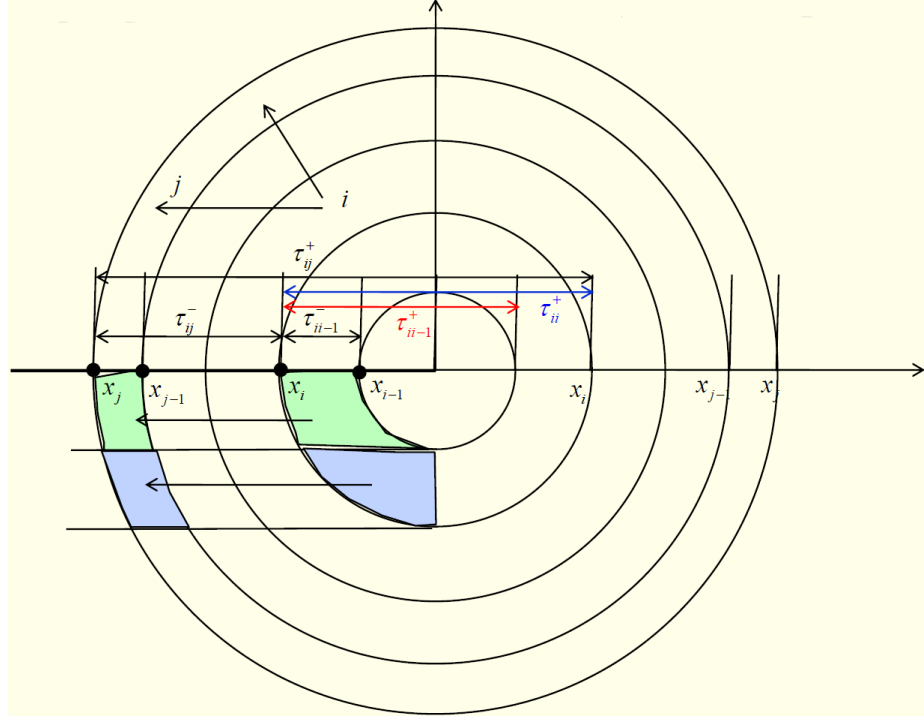
Figure A.1: Cylindrical Geometry Collision Probabilities [60]

Now that the geometry is set up, we can take advantage of the symmetry of the problem and only model on quarter of the volume to simplify the calculation. The kernels from Equations A.16 and A.17 can be used, but some modification is required. The reason for these modifications is that from a ring i, a ring j which is outside ring i can be intersected from two different directions. Two "sub-kernels" are defined for teh positive and negative directions. Each covers only one of two directions, so they should be multiplied by $\frac{1}{2}$. However, each only accounts for $\frac{1}{4}$ the total volume, so the final expression $P_{ij}(y)$ should be multiplied by 2 to account for each of these. The sub-kernels are each multiplied by $\frac{1}{\Sigma_i V_i}$ to account for the unit source density in the volume V_i as well as a $\frac{1}{\Sigma_i}$ term that comes from the change of variables during the integration over t_i. The positive and negative τ terms are shown in the geometry in A.1.

$$P_{ij}^-(y) = \frac{1}{\Sigma_i V_i}\left(Ki_3\left(\tau_{ij-1}^-\right) + Ki_3\left(\tau_{i-1j}^-\right) - Ki_3\left(\tau_{ij}^-\right) - Ki_3\left(\tau_{i-1j-1}^-\right)\right) \tag{A.18a}$$

$$P_{ij}^+(y) = \frac{1}{\Sigma_i V_i}\left(Ki_3\left(\tau_{i-1j-1}^+\right) + Ki_3\left(\tau_{ij}^+\right) - Ki_3\left(\tau_{i-1j}^+\right) - Ki_3\left(\tau_{ij-1}^+\right)\right) \tag{A.18b}$$

$$P_{ij}(y) = 2\left(P_{ij}^-(y) + P_{ij}^+(y)\right). \tag{A.18c}$$

Now we introduce notation to simplify this expression:

$$S_{ij} = \int_0^{R_i} \left(Ki_3\left(\tau_{ij}^+\right) - Ki_3\left(\tau_{ij}^-\right) \right) dy \tag{A.19a}$$

$$\Rightarrow P_{ij} = \Sigma_i V_i \int_0^{R_i} P_{ij}(y)\, dy$$

$$= 2\left(S_{ij} + S_{i-1\,j-1} - S_{ij-1} - S_{i-1j} \right). \tag{A.19b}$$
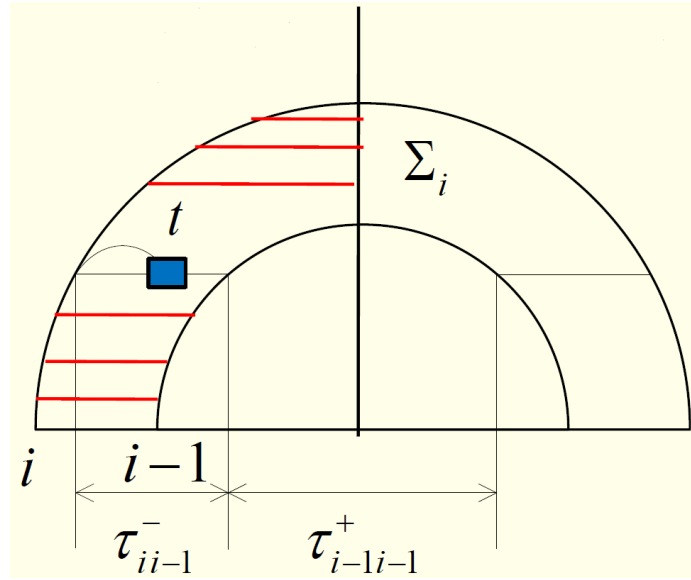

Figure A.2: Cylindrical Geometry Collision Probabilities Self-Transport [60]

It is useful to note at this point that the principle of reciprocity exists for these kernels. What is meant by this is that $P_{ij} = P_{ji}$. This is because the S_{ij} terms are functions only of τ_{ij}^+ and τ_{ij}^-. These variables are defined as the distance between radii i and j, to $\tau_{ij}^+ = \tau_{ji}^+$ and $\tau_{ij}^- = \tau_{ji}^-$. Because of this, only half of the possible combinations of i and j must be calculated. Likewise, when the final transport matrix is set up, it will be symmetric, so only one of the upper and lower triangles must be explicitly calculated. This saves some time in the computation of the matrix elements.

A similar process is followed for the self-transport kernel. Again, positive and negative directions are set up. Additionally, the possibility of a neutron being born in i, escaping, then re-entering the other side of i must be accounted for as well. This is illustrated in

Figure A.2. This leads to the kernel in Equation A.20:

$$P_{ii}^-(y) = \frac{t_i(y)}{V_i} + \frac{1}{\Sigma_i V_i}\left(Ki_3\left(\tau_{ii-1}^-\right) + Ki_3\left(\tau_{i-1i}^-\right) - Ki_3\left(\tau_{ii}^-\right) - Ki_3\left(\tau_{i-1i-1}^-\right)\right) \tag{A.20a}$$

$$P_{ii}^+(y) = \frac{t_i(y)}{V_i} + \frac{1}{\Sigma_i V_i}\left(Ki_3\left(\tau_{ii}^+\right) + Ki_3\left(\tau_{i-1i-1}^+\right) - Ki_3\left(\tau_{ii-1}^+\right) - Ki_3\left(\tau_{i-1i}^+\right)\right) \tag{A.20b}$$

$$P_{ii}(y) = 2\left(P_{ii}^-(y) + P_{ii}^+(y)\right) \tag{A.20c}$$

$$P_{ii} = \Sigma_i V_i \int_0^{R_i} P_{ii}(y)\,dy$$

$$= \Sigma_i V_i + 2\left(S\,ii + S_{i-1i-1} - S_{ii-1} - S_{i-1i}\right). \tag{A.20d}$$

P_{ij} is the probability that a neutron born in cell i has its first collision in cell j. Likewise, P_{ii} is the probability that a neutron born in cell i has its first collision in the same cell. To obtain the transport matrix elements T_{ij} and T_{ii}, we must consider the actual linear system we wish to solve. The goal is to find the reaction rates in each cell, from which we can easily find the flux in the cell ϕ_i. There are two main contributions to the reaction rates in cell i. The first contribution is from neutrons born in another cell j which have their first collision in cell i. This source is given by $\frac{Q_j}{\Sigma_{t,j}}$, which gives the contribution to the reaction rates in i when multiplied by P_{ji}. The second part of the source comes from neutrons which streamed into j and collided, scattered, then streamed into i before having their next collision. This source is given by $P_{ji}\phi_j c_j$, where c_j is the scattering ration in cell j, defined as $\frac{\Sigma_{s,j}}{\Sigma_{t,j}}$. The linear sysmem which needs to be solved is then shown in Equation A.21:

$$\Sigma_{t,i}\phi_i V_i = sum_{j=1}^{N_R} P_{ji}\left(c_j\phi_j + \frac{Q_j}{\Sigma_{t,j}}\right). \tag{A.21}$$

The linear system, with the matrix and source elements, can now be explicitly defined in Equation A.22:

$$\underline{\underline{T}}\underline{\phi} = \underline{B}, \tag{A.22a}$$

$$T_{ij} = -P_{ji}c_j = -2c_j\left(S\,ii + S_{i-1j-1} - S_{ij-1} - S_{i-1j}\right) \tag{A.22b}$$

$$T_{ii} = \Sigma_{t,i}V_i - P_{ii}c_i = \Sigma_{t,i}V_i - \Sigma_{s,i}V_i - 2c_j\left(S\,ii + S_{i-1j-1} - S_{ij-1} - S_{i-1j}\right), \tag{A.22c}$$

$$B_i = \sum_{j=1}^{N_R} P_{ji}\frac{Q_j}{\Sigma_{t,j}}. \tag{A.22d}$$

The only remaining unknowns required to construct this linear system are the S_{ij} terms that

involve integrals over the third-order Bickley-Naylor functions. This will be discussed in the next section.

A.2.3 Bickley-Naylor Function Integration

In the previous sections, we defined the following functions which contain integrals over third-order Bickley-Naylor functions:

$$S_{ij} = \int\limits_{0}^{R} \left(Ki_3\left(\tau_{ij}^+(y)\right) - Ki_3\left(\tau_{ij}^-(y)\right) \right) dy \,. \tag{A.23}$$

To numerically integrate these functions, we first break them up into regions:

$$S_{ij} = \sum_{k=1}^{i} S_{ij}^k \tag{A.24a}$$

$$S_{ij}^k = \int\limits_{R_{k-1}}^{R_k} \left(Ki_3\left(\tau_{ij}^+(y)\right) - Ki_3\left(\tau_{ij}^-(y)\right) \right) dy \,. \tag{A.24b}$$

Next, we perform a coordinate transformation to change the bounds of integration to be on the interval $[-1, 1]$. This allows us to apply a Gaussian quadrature to evaluate the integral numerically. The results of the transformation are shown in A.25, where $f(p)$ is the transformed S_{ij}^k and p_i and ω_i are some points and weights associated with the quadrature.

$$p = 2\frac{y - R_{k-1}}{R_k - R_{k-1}} - 1 = 2\frac{y - R_{k-1}}{\Delta_k} - 1 \,, \tag{A.25a}$$

$$dp = \frac{2}{\Delta_k} dy \,, \tag{A.25b}$$

$$\Rightarrow S_{ij}^k = \frac{\Delta_k}{2} \int\limits_{-1}^{1} f(p)\,dp = \frac{\Delta_k}{2} \sum_i \omega_i f(p_i) \,. \tag{A.25c}$$

This can now be integrated using a standard Gaussian quadrature [34]. Because this quadrature is being applied to sections of each integral rather than the whole integral, MPACT uses a 4-point Gaussian quadrature (shown in Table A.1), which is sufficient to give accurate solutions over the entire bounds of the integral.

Table A.1: Four-Point Gaussian Quadrature

Point	Weight
$\pm\sqrt{\frac{3}{7} - \frac{2}{7}\sqrt{\frac{6}{5}}}$	$\frac{18+\sqrt{30}}{36}$
$\pm\sqrt{\frac{3}{7} + \frac{2}{7}\sqrt{\frac{6}{5}}}$	$\frac{18-\sqrt{30}}{36}$

Appendix B

C5G7 Cross-Sections

```
1   C5G7 macroscopic cross section data
2    7 5
3    2.0E+07 1.0E+06  5.0E+05 1.0E+03 1.0E+02 10. 0.0635
4   !
5   !Comments can appear after the first 3 lines and between macro/micro blocks
6   !
7   !In the second line, the first number is number of groups and the other is
8   !number of cross section sets.
9   !
10  !In the third line the energy group bounds are made up.
11  !
12  !Data here is derived from NEA/NSC/DOC(2003)16 or ISBN 92-64-02139-6
13  !Table 1 of Appendix A.
14  !
15  !The control rod cross sections come from NEA/NSC/DOC(2005)16 or
16  !ISBN 92-64-01069-6 Table 1 of Appendix A.
17  !
18  !For each cross-section set:
19  ! XSMACRO <name> <scattering_order>
20  !  <absorption_xs> <nu-fission_xs> <kappa-fission_xs> <chi>  ! Repeats for each group
21  !  <scattering_matrix> !<ngroup>x<ngroup> matrix, repeated <scattering_order>+1 times
22  !
23
24  !Moderator
25  XSMACRO Moderator 0
26    6.0105E-04 0.000000E+00 0.00000E+00 0.0000E+00
27    1.5793E-05 0.000000E+00 0.00000E+00 0.0000E+00
28    3.3716E-04 0.000000E+00 0.00000E+00 0.0000E+00
29    1.9406E-03 0.000000E+00 0.00000E+00 0.0000E+00
30    5.7416E-03 0.000000E+00 0.00000E+00 0.0000E+00
31    1.5001E-02 0.000000E+00 0.00000E+00 0.0000E+00
32    3.7239E-02 0.000000E+00 0.00000E+00 0.0000E+00
33    4.44777E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
34    1.13400E-01 2.82334E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
35    7.23470E-04 1.29940E-01 3.45256E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
36    3.74990E-06 6.23400E-04 2.24570E-01 9.10284E-02 7.14370E-05 0.00000E+00 0.00000E+00
37    5.31840E-08 4.80020E-05 1.69990E-02 4.15510E-01 1.39138E-01 2.21570E-03 0.00000E+00
38    0.00000E+00 7.44860E-06 2.64430E-03 6.37320E-02 5.11820E-01 6.99913E-01 1.32440E-01
39    0.00000E+00 1.04550E-06 5.03440E-04 1.21390E-02 6.12290E-02 5.37320E-01 2.48070E+00
40
```

```
41
42  !Guide tube
43  XSMACRO GuideTube 0
44    5.1132E-04 0.000000E+00 0.00000E+00 0.0000E+00
45    7.5801E-05 0.000000E+00 0.00000E+00 0.0000E+00
46    3.1572E-04 0.000000E+00 0.00000E+00 0.0000E+00
47    1.1582E-03 0.000000E+00 0.00000E+00 0.0000E+00
48    3.3975E-03 0.000000E+00 0.00000E+00 0.0000E+00
49    9.1878E-03 0.000000E+00 0.00000E+00 0.0000E+00
50    2.3242E-02 0.000000E+00 0.00000E+00 0.0000E+00
51    6.61659E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
52    5.90700E-02 2.40377E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
53    2.83340E-04 5.24350E-02 1.83297E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
54    1.46220E-06 2.49900E-04 9.23970E-02 7.88511E-02 3.73330E-05 0.00000E+00 0.00000E+00
55    2.06420E-08 1.92390E-05 6.94460E-03 1.70140E-01 9.97372E-02 9.17260E-04 0.00000E+00
56    0.00000E+00 2.98750E-06 1.08030E-03 2.58810E-02 2.06790E-01 3.16765E-01 4.97920E-02
57    0.00000E+00 4.21400E-07 2.05670E-04 4.92970E-03 2.44780E-02 2.38770E-01 1.09912E+00
58
59
60  !Gap -- Actually just guide tube
61  XSMACRO Gap 0
62    5.1132E-04 0.000000E+00 0.00000E+00 0.0000E+00
63    7.5801E-05 0.000000E+00 0.00000E+00 0.0000E+00
64    3.1572E-04 0.000000E+00 0.00000E+00 0.0000E+00
65    1.1582E-03 0.000000E+00 0.00000E+00 0.0000E+00
66    3.3975E-03 0.000000E+00 0.00000E+00 0.0000E+00
67    9.1878E-03 0.000000E+00 0.00000E+00 0.0000E+00
68    2.3242E-02 0.000000E+00 0.00000E+00 0.0000E+00
69    6.61659E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
70    5.90700E-02 2.40377E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
71    2.83340E-04 5.24350E-02 1.83297E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
72    1.46220E-06 2.49900E-04 9.23970E-02 7.88511E-02 3.73330E-05 0.00000E+00 0.00000E+00
73    2.06420E-08 1.92390E-05 6.94460E-03 1.70140E-01 9.97372E-02 9.17260E-04 0.00000E+00
74    0.00000E+00 2.98750E-06 1.08030E-03 2.58810E-02 2.06790E-01 3.16765E-01 4.97920E-02
75    0.00000E+00 4.21400E-07 2.05670E-04 4.92970E-03 2.44780E-02 2.38770E-01 1.09912E+00
76
77  !UO2 fuel-clad
78  XSMACRO UO2-3.3 0
79    8.0248E-03 2.005998E-02 7.21206E-03 5.8791E-01
80    3.7174E-03 2.027303E-03 8.19301E-04 4.1176E-01
81    2.6769E-02 1.570599E-02 6.45320E-03 3.2990E-04
82    9.6236E-02 4.518301E-02 1.85648E-02 1.0000E-07
83    3.0020E-02 4.334208E-02 1.78084E-02 0.0000E+00
84    1.1126E-01 2.020901E-01 8.30348E-02 0.0000E+00
85    2.8278E-01 5.257105E-01 2.16004E-01 0.0000E+00
86    1.27537E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
87    4.23780E-02 3.24456E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
88    9.43740E-06 1.63140E-03 4.50940E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
89    5.51630E-09 3.14270E-09 2.67920E-03 4.52565E-01 1.25250E-04 0.00000E+00 0.00000E+00
90    0.00000E+00 0.00000E+00 0.00000E+00 5.56640E-03 2.71401E-01 1.29680E-03 0.00000E+00
91    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 1.02550E-02 2.65802E-01 8.54580E-03
92    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 1.00210E-08 1.68090E-02 2.73080E-01
93
94  !Control Rod
95  XSMACRO CRod 0
```

```
 96    1.70490E-03 0.000000E+00 0.00000E+00 0.0000E+00
 97    8.36224E-03 0.000000E+00 0.00000E+00 0.0000E+00
 98    8.37901E-02 0.000000E+00 0.00000E+00 0.0000E+00
 99    3.97797E-01 0.000000E+00 0.00000E+00 0.0000E+00
100    6.98763E-01 0.000000E+00 0.00000E+00 0.0000E+00
101    9.29508E-01 0.000000E+00 0.00000E+00 0.0000E+00
102    1.17836E+00 0.000000E+00 0.00000E+00 0.0000E+00
103    1.70563E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
104    4.44012E-02 4.71050E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
105    9.83670E-05 6.85480E-04 8.01859E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
106    1.27786E-07 3.91395E-10 7.20132E-04 5.70752E-01 6.55562E-05 0.00000E+00 0.00000E+00
107    0.00000E+00 0.00000E+00 0.00000E+00 1.46015E-03 2.07838E-01 1.02427E-03 0.00000E+00
108    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 3.81486E-03 2.02465E-01 3.53043E-03
109    0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 3.69760E-09 4.75290E-03 6.58597E-01
```

References

[1] A. K. Prinja and E. W. Larsen. *Handbook of Nuclear Engineering*, chapter Principles of Neutron Transport. D. G. Cacuci (Ed.) (2010).

[2] A. Hebert. *Applied Reactor Physics*. Presses inter Polytechnique (2009).

[3] MPACT Team. *MPACT Theory Manual. Technical report*, Oak Ridge National Laboratory and the University of Michigan (2015).

[4] S. Stimpson *et al. Improved diffusion coefficents for SPN axial solvers in the MPACT 2D/1D method applied to the AP1000 PWR start-up core models*, volume 3, (pp. 1702–1716). American Nuclear Society (2015).

[5] B. C. Yee, E. W. Larsen, and B. Kochunas. "An ananalytic derivation of transport-corrected p_0 cross sections and diffusion coefficients." In: *Proceesings of the Physics of Reactors (PHYSOR) conference*. Sun Valley, ID (2016).

[6] B. R. Herman *et al. Improved diffusion coefficients generated from Monte Carlo codes. Technical report*, American Nuclear Society, 555 North Kensington Avenue, La Grange Park, IL 60526 (United States) (2013).

[7] M. Ryu *et al.* "Incorporation of anisotropic scattering in ntracer." In: *Transactions of the Korean Nuclear Society Autumn Meeting*. Pyeongchang, Korea (2014).

[8] A. Yamamoto, Y. Kitamura, and Y. Yamane. "Simplified treatments of anisotropic scattering in lwr core calculations." *Journal of Nuclear Science and Technology*, **45(3)**: pp. 217–229. URL `http://dx.doi.org/10.1080/18811248.2008.9711430` (2008).

[9] J. Askew. *A characteristics formulation of the neutron transport equation in complicated geometries. Technical report*, United Kingdom Atomic Energy Authority (1972).

[10] M. Halsall. *CACTUS, a characteristics solution to the neutron transport equations in complicated geometries. Technical report*, UKAEA Atomic Energy Establishment (1980).

[11] E. Masiello, R. Clemente, and S. Santandrea. "High-order method of characteristics for 2-d unstructured meshes." In: *Proc. Int. Conf. on Mathematics, Computational Methods & Reactor Physics (M&C 2009)*. Saratoga Springs, NY, USA (2009).

[12] K. Smith. "Nodal method storage reduction by nonlinear iteration." *Trans. Am. Nucl. Soc.*, **44**: p. 265 (1983).

[13] R. G. McClarren. "Theoretical aspects of the simplified pn equations." *Transport Theory and Statistical Physics*, **39(2-4)**: pp. 73–109 (2011).

[14] H. Finnemann, F. Bennewitz, and M. Wagner. "Interface current techniques for multidimensional reactor calculations." *Atomkernenergie*, **30(2)**: pp. 123–128 (1977).

[15] D. Knott and A. Yamamoto. *Handbook of Nuclear Engineering*, chapter Lattice Physics Computations. D. G. Cacuci (Ed.) (2010).

[16] B. M. Kochunas. *A Hybrid Parallel Algorithm for the 3-D Method of Characteristics Solution of the Boltzmann Transport Equation on High Performance Compute Clusters*. Ph.D. thesis, University of Michigan (2013).

[17] 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." In: *Proc. Korean Nucl. Soc.* Kwangju, Korea (2002).

[18] N. Z. Cho, G. S. Lee, and C. J. Park. "Fusion method of characteristics for 3d whole core transport calculation." In: *Trans. Am. Nucl. Soc.*, volume 86, (pp. 322–324) (2002).

[19] B. Cho and N. Z. Cho. "A nonoverlapping local/global iterative method with 2-d/1-d fusion transport kernel and p-cmfd wrapper for transient reactor analysis." *Annals of Nuclear Energy*, **85**: pp. 937–957 (2015).

[20] J. Y. Cho *et al.* "Three-dimensional heterogeneous whole core transport calculations employing planar moc solutions." In: *Trans. Am. Nucl. Soc.*, volume 87, (pp. 234–236) (2002).

[21] M. Hursin, B. Kochunas, and T. Downar. *DeCART Theory Manual. Technical report*, University of Michigan (2008).

[22] H. G. Joo *et al.* "Methods and performance of a three-dimensional whole-core transport code decart." In: *PHYSOR 2004 – The Physics of Fuel Cycles and Advanced Nuclear Systems: Global Developments*. Chicago, Illinois (2004).

[23] B. W. Kelley and E. W. Larsen. "2d/1d approximations to the 3d neutron transport equation. i:theory." In: *Proc. M&C 2013*. Sun Valley, ID (2013).

[24] B. S. Collins *et al.* "Stability and accuracy of 3d neutron transport simulations using the 2d/1d method in mpact." *Journal of Computational Physics*, **326**: pp. 612–628 (2016).

[25] B. W. Kelley. *An Investigation of 2D/1D Approximations to the 3D Boltzmann Transport Equation*. Ph.D. thesis, University of Michigan. URL https://deepblue.lib.umich.edu/bitstream/handle/2027.42/113576/kelleybl_1.pdf?sequence=1&isAllowed=y (2015).

[26] S. G. Stimpson. *An Azimuthal, Fourier Moment-Based Axial S_N Solver for the 2D/1D Scheme*. phdthesis, University of Michigan. URL `https://deepblue.lib.umich.edu/bitstream/handle/2027.42/111446/sgstim_1.pdf?sequence=1&isAllowed=y` (2015).

[27] J. Y. Cho *et al.* "Axial sp$_n$ and radial MOC coupled whole core transport calculation." *Journal of Nucl. Sci. Tech.*, **44(9)**: pp. 1156–1171 (2007).

[28] M. Ryu *et al.* "Solution of the beavrs benchmark using the ntracer direct whole core calculation code." *Journal of Nuclear Science and Technology*, **52(7-8)**: pp. 961–969. URL `http://dx.doi.org/10.1080/00223131.2015.1038664` (2015).

[29] Y. S. Jung and H. G. Joo. "Control rod decusping treatment based on local 3-d cmfd calculation for direct whole core transport solvers." In: *Proceedings of the International Congress on Advances in Nuclear Power Plants (ICAPP)* (2014).

[30] S. Stimpson, B. S. Collins, and B. Kochunas. *MOC Efficiency Improvements Using a Jacobi Inscatter Approximation. techreport CASL-U-2016-1056-002*, Oak Ridge National Laboratory. URL `http://info.ornl.gov/sites/publications/files/Pub69335.pdf` (2016).

[31] D. E. Cullen. "Application of the probability table method to multigroup calculations of neutron transport." *Nucl. Sci. Eng.*, **55**: p. 387 (1974).

[32] Y. Liu *et al.* "Resonance self-shielding methodology in mpact." In: *Proceedings of the Conference on Mathematics and Computation (M&C 2013)*. Sun Valley, ID, USA. URL `http://www.casl.gov/docs/CASL-U-2015-0008-000.pdf` (2013).

[33] A. Yamamoto *et al.* "Derivation of optimum polar angle quadrature set for the method of characteristics based on approximation for the bickley function." *Journal of Nucl. Sci. Tech.*, **44(2)** (2007).

[34] P. J. Davis and I. Polonsky. *A Handbook of Mathematical Functions*, chapter Numerical Interpolation, Differentiation and Integration. Dover (1972).

[35] H.-S. Joo. *Resolution of the control rod cusping problem for nodal methods*. Ph.D. thesis, Massachusetts Institute of Technology (1984).

[36] K. S. Smith. *Spatial homogenization methods for light water reactor analysis*. Ph.D. thesis, Massachusetts Institute of Technology (1980).

[37] C. L. Hoxie. *Application of nodal equivalence theory to the neutronic analysis of PWRs*. Ph.D. thesis, Massachusetts Institute of Technology (1982).

[38] H. S. Khalil. *The application of nodal methods to PWR analysis*. Ph.D. thesis, Massachusetts Institute of Technology (1983).

[39] A. Y. Cheng. *Homogenization of BWR assemblies by response matrix methods*. Ph.D. thesis, Massachusetts Institute of Technology (1981).

[40] P. J. Finck. *Homogenization and dehomogenization schemes for BWR assemblies*. Ph.D. thesis, Massachusetts Institute of Technology (1982).

[41] F. Bennewitz, H. Finnemann, and M. Wagner. "Higher order corrections in nodal reactor calculations." *Transactions of the American Nuclear Society*, **22** (1975).

[42] H. Finnemann and W. Gundlach. "Space-time kinetics code iqsbox for pwr and bwr." *Atomkernenergie Kerntechnik*, **37** (1981).

[43] K. S. Smith. *An Analytic Nodal Method for Solving the Two-Group, Multidimensional, Static and Transient Neutron Diffusion Equation*. mathesis, Massachusetts Institute of Technology (1979).

[44] S. Langenbuch, W. Maurer, and W. Werner. "Coarse-mesh flux-expansion method for the analysis of space-time effects in large light water reactor cores." *Nuclear Science and Engineering*, **63(4)**: pp. 437–456 (1977).

[45] J. C. Gehin. *A quasi-static polynomial nodal method for nuclear reactor analysis*. *Technical report*, Oak Ridge Inst. for Science and Education, TN (United States); Massachusetts Inst. of Tech., Cambridge, MA (United States) (1992).

[46] K. Smith *et al.* "Enhancements of the studvick core management system." In: *Proceedings of the Topical Meeting on Advances in Reactor Physics. Charleston SC USA* (1992).

[47] K. B. Lee *et al.* "Correction of the control rod cusping effect using one-dimensional fine mesh flux profiles." In: *Proceedings of the KNS Spring Meeting*, volume 2 (1998).

[48] T. Downar *et al.* "Theory manual for the parcs neutronics core simulator." *School of Nuclear Engineering, Purdue University, W. Lafayette, Indiana*, **47907** (2004).

[49] Y. H. Kim and N. Z. Cho. "A flux-and adjoint flux-weighting method for the control rod cusping problem in nodal methods." In: *Proceedings of the KNS Spring Meething* (1990).

[50] Z. R. de Lima *et al.* "Correcting the cusping problem in three-dimensional transients through nem modification." *Nuclear Science and Engineering*, **170(1)**: pp. 66–74 (2012).

[51] A. Martinez, V. Pereira, and E. DA SILVA. "A system for the prediction and determination of the subcritical multiplication condition." *Kerntechnik*, **64(4)**: pp. 230–234 (1999).

[52] A. DallOsso. "Reducing rod cusping effect in nodal expansion method calculations." In: *Proceedings of the International Conference on the New Frontiers of Nuclear Technology: Reactor Physics, Safety and High-Performance Computing, PHYSOR, Seoul, Korea* (2002).

[53] A. Yamamoto. "A simple and efficient control rod cusping model for three-dimensional pin-by-pin core calculations." *Nuclear technology*, **145(1)**: pp. 11–17 (2004).

[54] L. Yu, D. Lu, and S. Zhang. "Further development of the ciama nodal method for kinetic applications." In: *Proceedings of the Reactor Physics Asia 2015 (RPHA15) Conference*. Jeju, South Korea (2016).

[55] D. Lu, L. Yu, and S. Zhang. "A three-dimensional nodal method with channel-wise intrinsic axial mesh adaptation." *Annals of Nuclear Energy*, **79**: pp. 152–157 (2015).

[56] B. M. Kochunas *et al.* "VERA core simulator methodology for pwr cycle depletion." In: *Proceedings of the ANS Joint International Conference on Mathematics and Computation (M&C 2015), Supercomputing in Nuclear Applications (SNA), and the Monte Carlo (MC) Method*. Nashville, TN, USA. URL `http://www.casl.gov/docs/CASL-U-2015-0155-000.pdf` (2015).

[57] A. T. Godfrey. *VERA Core Physics Benchmark Progression Problem Specifications. Technical report*, Oak Ridge National Laboratory. URL `http://www.casl.gov/docs/CASL-U-2012-0131-004.pdf` (2014).

[58] E. E. Lewis *et al. Benchmark on Deterministic Transport Calculations without Spatial Homogenization. Technical report*, Nuclear Energy Agency Organisation for Economic Cooperation and Development (NEA-OECD) (2003).

[59] E. E. Lewis *et al. Benchmark on Deterministic Transport Calculations without Spatial Homogenization: MOX Fuel Assembly 3-D Extension Case. Technical report*, Nuclear Energy Agency Organisation for Economic Cooperation and Development (NEA-OECD) (2005).

[60] H. G. Joo. "Lecture Notes in NERS 561: Reactor Analysis I." (2011).