Linear Diffusion Acceleration for Neutron Transport Problems

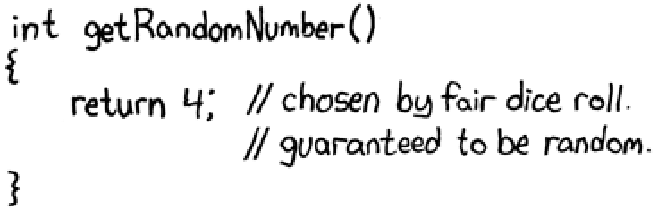
by

Zackary Glen Dodson

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Nuclear Engineering and Radiological Sciences and Scientific Computing)
in the University of Michigan
2021

Doctoral Committee:

Professor Brendan M. Kochunas, Co-Chair
Professor Edward W. Larsen, Co-Chair
Professor Thomas J. Downar
Professor Shravan K. Veerapaneni



```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

— xkcd Comics, Randall Munroe, 2007

“Render unto man the things which are man’s and unto the computer the things which are the computer’s.”

— Norbert Wiener, “God and Golem, Inc,” 1964

Zackary Glen Dodson
zdodson@umich.edu
ORCID ID: 0000-0002-7170-4465

Dedication

For my mother, Jaclynn, whose encouragement and support
allowed me to succeed.

Acknowledgments

First, I would like to thank my thesis committee. Special thanks goes to the co-chairs of my committee, Professor Emeritus Edward Larsen and Assistant Professor Brendan Kochunas, for their guidance and patience. We spent quite a while iterating on this thesis under circumstances that were not ideal, but we were able to pull through. During my time working with them, I gained invaluable skills that will assist me in my career.

Thanks also to the funding sources for this work. This research was supported by the Consortium for Advanced Simulation of Light Water Reactors (`www.casl.gov`), an Energy Innovation Hub (`http://www.energy.gov/hubs`) for Modeling and Simulation of Nuclear Reactors under U.S. Department of Energy Contract No. DE-AC05-00OR22725.

I would also like to thank everyone who helped me with my work on MPACT. Special thanks goes to Daniel Jabaay, who tolerated my countless visits to his office for assistance. He was always helpful and kind, and I very much appreciated his patience with me. With his help, I was able to better navigate the source code of MPACT and implement my thesis work.

Next, I would like to thank the friends I have made during graduate school. Especially Mike, Nick A., and Julio. Thanks for all the fun times we had, and for all of the heart-to-hearts too. There is a cliché line about “the friends you made along the way”, because it’s important to remember and appreciate the people who make you smile. I will always remember the camaraderie that we shared during graduate school. I had great times with my D&D group too, which always gave me something to look forward to every week. Nick H., thanks for all the time you put into DMing. Special thanks also goes to my therapist, Catie, who was my lighthouse on stormy seas.

Finally, and most importantly, I give thanks to my family. Without their tremendous support throughout my time in graduate school, I would not have made it this far. Though times were often tough, they were always there to lend an ear and give advice. Very special thanks go to my mother and sister. Mom, thanks so much for picking up the phone whenever I needed it and always encouraging me to keep going. Sydnee, thanks for supporting me during the hard times and always having the best advice. I love you all, and know that I will always remember the love and support that you have shown me.

TABLE OF CONTENTS

Chapter

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

LIST OF ACRONYMS

BiCGSTAB	*Biconjugate Gradient Stabilized*
BOL	*Beginning of Life*
CMFD	*Coarse Mesh Finite Difference*
CMR	*Coarse Mesh Rebalance*
DNC	*Did Not Converge*
DSA	*Diffusion Synthetic Acceleration*
FAT	*Fredholm Alternative Theorem*
GAML	*Gelbard, Adams, McCoy, and Larsen*
GMRES	*Generalized Minimal Residual*
HZP	*Hot Zero Power*
ILU	*Incomplete LU*
LDA	*Linear Diffusion Acceleration*
LHS	*Left-Hand-Side*
LU	*Lower-Upper*
LWR	*Light Water Reactor*
MPACT	*Michigan Parallel Characteristics-based Transport*
MOC	*Method of Characteristics*
MSED	*Multilevel-in-Space-and-Energy Diffusion*
NBE	*Neutron Balance Equation*
NDA	*Nonlinear Diffusion Acceleration*
NTE	*Neutron Transport Equation*
odCMFD	*optimally-diffusive CMFD*
OI	*Outer Iteration*
PARCS	*Purdue Advanced Reactor Core Simulator*
PI	*Power Iteration*
QD	*Quasi-Diffusion*
RBBJ	*Red-Black Block Jacobi*
RHS	*Right-Hand-Side*
SDA	*Semilinear Diffusion Acceleration*
SI	*Source Iteration*
SOR	*Successive Over-Relaxation*
TCP$_0$	*Transport-Corrected* P_0
TL	*Transverse Leakage*
TS	*Transport Sweep*
VERA	*Virtual Environment for Reactor Applications*

ABSTRACT

Nuclear engineers are interested in solutions of the *Neutron Transport Equation* (NTE), with the goal of improving the safety and efficiency of reactors and critical nuclear systems. Complex simulations are used to obtain detailed solutions of the NTE, and can require immense computational resources to execute. A variety of methods have been developed to ease the computational burden of simulating full-scale, whole-core reactor problems. Among these is transport acceleration, which improves the convergence rate of iterative transport calculations.

In addition to the use of acceleration methods, certain approximations are often made when solving the NTE. The 2D/1D approximation is used to generate a 3D solution of the NTE by iteratively solving coupled 2D radial and 1D axial equations. This method is one of the foundational techniques used in the neutronics code MPACT. Also, the *Transport-Corrected* P_0 (TCP$_0$) approximation for neutron scattering is often used in reactor analysis codes to simplify higher-order scattering physics. Unfortunately, both of these approximations allow for non-positive flux solutions of the NTE. More importantly, some spatial discretizations of the NTE also permit negative solutions. Under certain conditions, this can cause instability for nonlinear acceleration methods such as *Coarse Mesh Finite Difference* (CMFD). In this thesis, we propose a novel acceleration scheme called *Linear Diffusion Acceleration* (LDA) that does not possess the nonlinearities present in CMFD.

This thesis work presents LDA as an alternative acceleration scheme to CMFD. As the name suggests, the LDA method is linear with respect to the scalar flux. Therefore, LDA is not susceptible to the same nonlinear modes of numerical failure as CMFD. In addition, LDA is shown to possess similar convergence properties as CMFD for practical problems that have no negative scalar fluxes. Transport acceleration with LDA allows for the use of some of the aforementioned approximations, in which the positivity of the scalar flux is not guaranteed. Fourier analysis of CMFD and LDA is performed to compare the theoretical convergence rates of the two methods for simple, spatially-heterogeneous problems. In addition, simple and practical case studies are presented in which CMFD fails due to nonlinearity. For these cases, LDA is shown to retain stability. Certain other advantages of LDA, which are a consequence of its mathematical structure, are also discussed.

CHAPTER 1

Introduction

This chapter provides motivation for the use of acceleration methods to improve the convergence rate of iterative methods for computational solutions of the *Neutron Transport Equation* (NTE). Motivation for the formulation of a novel acceleration technology called *Linear Diffusion Acceleration* (LDA), which is the focus of this thesis, is provided in Section 1.1. A brief history of transport acceleration is then given for context in Section 1.2. The novelty of LDA lies in the ability to accelerate *Source Iteration* (SI) using a set of equations that are linear with respect to the scalar flux for eigenvalue problems. Lastly, Section 1.3 outlines of the layout of this thesis document.

1.1 Motivation

For over 60 years, computers have been used in the design and analysis of nuclear systems due to the impracticality of obtaining sufficiently accurate and detailed analytical models [1]. The methods used in simulations have generally been constrained by the available computational resources. As computing power and availability have increased over the intervening decades, simulation methods have evolved to reflect these advancements. The accuracy and resolution of computational results have improved over time to keep up with the demand of reactor designers and analysts.

For nuclear reactor design and analysis, the primary quantities of interest are the fundamental eigenvalue and eigenvector (sometimes referred to as the *fundamental eigenpair*) of the *Neutron Transport Equation* (NTE). The NTE is a version of the Boltzmann transport equation that describes the relevant features of neutron transport physics across seven dimensions of phase space [2]. Though many solutions to this equation exist for a given geometric configuration, only one corresponds to a given core power level in a reactor. Generally, the reciprocal of the fundamental eigenvalue, called the multiplication factor (k_{eff}), is used as a measure of the neutron multiplicity. The fundamental eigenvector, after integrating over the angular variable, is denoted as the neutron scalar flux, which describes the spatial and energy distribution of neutrons in the system for a given state. Useful information, such as various types of reaction rates, fuel burnup, and power distribution can be estimated using the scalar flux.

Accurate knowledge of the neutronic and thermal hydraulic properties of the system allows for the crucial determination of reactor safety and efficiency. For example, the determination of fuel temperature in a nuclear core can indicate locations where the melting point may be exceeded. This illustrates the importance of computational modeling for reactor design and operation. In general, accurate simulations make it possible to reduce the safety margins employed with legacy models that rely on approximations and coarse discretizations. These older models require large safety margins, to account for inaccuracies [3]. Easing such excessive margins confidently with sufficiently-verified and validated simulation tools can lead to improved efficiency and profit for power plant operators.

Two disparate classes of methods for solving the NTE have been widely used for the aforementioned analyses. The first is *deterministic methods*, in which a solution of the NTE is obtained through discretization of the phase space and the use of numerical algorithms that are usually iterative in nature. Common discretization methods include *discrete ordinates* ($\mathrm{S_N}$), *spherical harmonics* ($\mathrm{P_N}$), the *Method of Characteristics* (MOC), and the *multigroup approximation*. Discrete ordinates, in which neutrons are assumed to travel along discrete directions of flight, utilizes a quadrature set to approximate integrals over the angular variable. The accuracy of this method depends largely on the choice of quadrature set and the number of discrete angles used. Spherical harmonics uses a different angular approximation, in which angularly-dependent quantities are expanded using moments of the spherical harmonics functions. This leads to a set of coupled equations that can be solved for the expansion coefficients that then determine the angular flux. The accuracy of $\mathrm{P_N}$ solutions depends largely on the number of moments used for the expansion. With MOC, the NTE is cast along a set of characteristic rays that pass through the problem geometry. Solutions of the resulting ordinary differential equations, resembling traditional representations of exponential attenuation, are used to construct the neutron flux distribution. MOC solution accuracy depends largely on the coverage of rays relative to the spatial discretization. Finally, multigroup methods assume that neutrons exist in discrete energy bands rather than a continuous spectrum. The accuracy of multigroup methods depends on the number of groups, the choice of group energy bounds, and weighting functions.

The second class of solution techniques are *stochastic* (also known colloquially as *Monte Carlo*) methods for solving the NTE [4]. These methods take an entirely different approach than deterministic methods and do not require discretization of the phase space.

Instead, the stochastic nature of neutron collision and nuclei fission is exploited by employing random number sequences, in conjunction with probability distribution functions that describe each relevant element of physics, to model individual, random neutron histories. Estimates of neutronic quantities, including the aforementioned eigenpair, can be obtained by sampling the probability distributions of these events. A major advantage of Monte Carlo methods is the avoidance of discretization of the phase space. Such discretization can be intractable for highly-resolved cases. However, the trade-off compared to deterministic methods is the need for an extremely large (and often impractical) number of histories to mitigate stochastic error for large and detailed models. Though both solution classes possess their own advantages and disadvantages, this thesis work focuses on deterministic applications.

Initially, two-group diffusion theory was the predominant deterministic method for obtaining the aforementioned eigenpair [5]. Diffusion theory is based on (among other approximations) the assumption that the neutron flux is a linear function of angle. The use of two-group diffusion theory transitioned to multigroup diffusion theory as *Light Water Reactor* (LWR) simulation became common and the need for more energy resolution increased (due to the nuances of neutron moderation and fuel materials) [6]. Advancements in computing power would allow for more complex and detailed simulations. This was driven by the desire for higher power density in commercial nuclear plants, to increase efficiency while ensuring safety. To fill the need for increased accuracy and resolution, transport methods would replace diffusion approximations. Though diffusion theory provides a reasonable solution, the increased spatial resolution would create sharp material discontinuities that cannot be accurately modeled by the neutron diffusion equation. Additionally, diffusion theory is not accurate near strong neutron absorbers, such as control rods. To solve the NTE without making assumptions such as the diffusion approximation, different (previously mentioned) computationally intensive solution methodologies began to be utilized. An algorithm called *Source Iteration* (SI) was used to iteratively solve the NTE by updating the fission and scattering source using updated estimates of the neutron flux. However, SI converges very slowly for problems containing materials with a high scattering interaction probability and low leakage probability [7]. Unfortunately, the moderator material for thermal reactors (water) is highly scattering to help slow the neutrons down to thermal temperature, where the probability of fission is high for the fuel isotope of choice (U-235). The result is slow convergence for many of the problems of interest. Various methods have been developed to mitigate this issue.

For this thesis, *acceleration methods* refer to methods that "accelerate" the convergence of SI by augmenting the base SI algorithm with additional steps. Each "outer" iteration consists of a “high-order” transport sweep, followed by an update of the scattering (and fission, if applicable) source. Acceleration methods include in each outer iteration an extra step, in which the transport sweep is followed by a “low-order” calculation. The inclusion of the low-order calculation makes a single accelerated outer iteration more expensive than the transport sweep alone. However, if this low-order calculation is able to significantly reduce the number of outer iterations required for convergence, then the additional cost of the low-order calculation will be worthwhile. For modern acceleration methods such as *Diffusion Synthetic Acceleration* (DSA) and *Coarse Mesh Finite Difference* (CMFD), the cost of the low-order calculation is (for many problems) at least one order of magnitude smaller than the transport sweep. Therefore, the cost of the accelerated outer iteration is not expected to be much greater than the cost of an “unaccelerated” outer iteration (consisting of just the transport sweep). However, the number of outer iterations is drastically reduced – from hundreds (or thousands, depending on the problem) to roughly 10. For such problems, the small extra cost of the low-order calculation is very much worthwhile.

Certain acceleration methods utilize the solution of a diffusion-like problem that allows for rapid convergence of SI for problems in which unaccelerated SI suffers from slow convergence. One acceleration scheme that has become prolific in the last couple of decades is CMFD [8, 9]. CMFD owes its popularity to ease of implementation, efficiency, and the ability to accelerate the convergence of fission source problems. One major downside to the method, however, is the presence of nonlinear terms in the CMFD formulation that allow for the possibility of numerical instability. Specifically, these terms are nonlinear with respect to the scalar flux. Nonlinear solution methodologies and approximations that do not guarantee the positivity of the solution are susceptible to instability for cases in which near-zero or negative discrete flux values appear. For example, some spatial discretization choices for the NTE (such as diamond-difference) allow for non-positive flux iterates or converged solutions [10]. In these situations, the nonlinear terms in CMFD can become unphysically large and degrade the performance of the method, even resulting in divergence. We refer to these aspects of CMFD as *nonlinear instabilities*. In some computer codes, algorithms have been implemented that prevent the possibility of negative flux estimates [11]. These algorithms (i) are approximate, and (ii) almost always degrade the accuracy of the final converged solution.

This thesis work is focused on the development of a new transport acceleration method called *Linear Diffusion Acceleration* (LDA). In 1982 (prior to the introduction of CMFD), a precursor of LDA was introduced by Gelbard, Adams, McCoy, and Larsen [12]. (We refer to this method as GAML – an acronym built using the first letters of the last names of the authors.) The GAML method was capable of accelerating SI for both fixed-source and eigenvalue problems, while avoiding local nonlinearities with respect to the scalar flux. LDA is an adaptation of GAML for modern applications. (Appendix A discusses the differences between GAML and LDA.) Though both LDA and CMFD are susceptible to degraded performance as the optical thickness of the low-order mesh increases, LDA is not susceptible to numerical failure due to the inclusion of nonlinear terms (LDA remains stable for cases in which non-positive flux iterates or converged solutions are possible). The former *linear* instability, pertaining to mesh cell optical thickness, is present for both methods and most likely relates to disparate discretization choices for the transport and acceleration problems. Resolving this linear instability is an open research problem outside of the scope of this thesis. However, the *nonlinear* instabilities, pertaining to the presence of nonlinear terms, is an issue for CMFD and other nonlinear methods, but not for LDA. Recent work has shown that the convergence rate of LDA is the same as that of CMFD for spatially-homogeneous problems where the nonlinear instabilities of CMFD do not present issues [13]. For many practical, spatially-heterogeneous problems, LDA possesses a different but similar convergence rate as CMFD, while avoiding problematic nonlinearities. For problems in which the nonlinear instabilities are present, LDA possesses an improved convergence rate compared to CMFD. The motivation for developing LDA is to provide a viable alternative to CMFD for cases in which CMFD exhibits the nonlinear instability. *To restate, LDA exhibits the linear instability related to mesh cell optical thickness, but avoids the nonlinear instability observed in CMFD.* Ideally, the advantages of LDA are such that it can replace CMFD entirely.

There are two primary goals for the work presented in this thesis. The first goal is to characterize how CMFD "breaks down" due to the aforementioned nonlinear instabilities. To support this goal, LDA should be shown to have similar convergence properties as CMFD for cases in which the nonlinear instability of CMFD is not relevant. This would demonstrate that LDA can be used in place of CMFD for general cases without adding additional computational expense. The second goal is to develop the LDA method to have the following properties: (i) linearity with respect to the scalar flux, (ii) improved power iteration properties, and (iii) potentially reduced linear solve computational effort. Because LDA does not possess the nonlinear instabilities present in CMFD, numerical instability due to non-positive quantities is avoided. The nonlinear instabilities of CMFD, which are caused by non-positive flux quantities, can result not only in divergence, but also, in a significantly slower rate of convergence. Negative fluxes cause neither slower convergence nor divergence in LDA. Also, the power iteration convergence rate of LDA can be more easily optimized as a result of the use of the fixed diffusion operator. Finally, the fixed diffusion operator of LDA allows for the possibility of reduced computational work when solving the low-order problem (compared to cases in which the operator is not fixed, which is the case for CMFD) using linear solver techniques that take advantage of this characteristic. To achieve these goals, it should be demonstrated that LDA possesses similar or better convergence properties and computational costs compared to CMFD for realistic problems.

1.2 History of Transport Acceleration

The use and development of techniques to improve the iterative convergence of SI has spanned several decades, beginning in the 1960s. Due to the slow convergence rate of SI for typical reactor problems (which is a result of the presence of highly scattering materials, such as water, or fissile materials), researchers sought methods to mitigate this issue. Such methods are referred to as *acceleration* methods, referring to their ability to improve the iterative convergence rate of SI (often drastically) using the solution of a lower-order (relative to the angular and spatial order of the transport equation) set of equations. This allows for accelerated SI schemes to converge to a given tolerance with fewer transport sweeps, which are generally computationally expensive. To be practical, an acceleration method should have a low-order component whose computational cost is a small fraction of the cost of the high-order transport sweep. Accelerated SI algorithms usually consist of the following components (each of which are performed every iteration): (i) solving the transport equation given a source (referred to as a transport sweep), (ii) solving a set of low-order acceleration equations (typically via the solution of a linear system), and (iii) transfer of information from the low-order solution to the high-order solution.

Early acceleration methods include *Chebyshev acceleration* and *Rebalance*. The former uses a linear combination of previous scalar flux iterates, which amounts to a weighted average informed by Chebyshev polynomials, to construct an improved estimate for the final converged solution [14]. A major drawback of this method is that the performance is dependent on the condition number of the transport operator. Therefore, the method can be ineffective for certain problems. A different approach is taken by the Rebalance methods, which use direction-dependent rebalance factors (these can be chosen to exist on the high-order transport mesh or a lower-order spatial mesh) to construct the *Neutron Balance Equation* (NBE). This equation is solved for the *rebalance factors*, which are then used to estimate the next scalar flux iterate. Some major drawbacks of rebalance methods are that they diverge for problems with (i) a high scattering ratio (which are generally the problems of interest for analysis of LWRs) and (ii) a highly-resolved spatial mesh (thus limiting the resolution of problems accelerated by this method) [15]. Rebalance acceleration methods were implemented in early transport codes such as ONETRAN, which uses the $\mathrm{S_N}$ method to solve the 1D multigroup transport equation [16].

Alongside the aforementioned methods, there were others that can be referred to as *preconditioning methods* because they can be shown to be mathematically equivalent to preconditioning matrix iteration schemes [17]. These methods have the effect of reducing the condition number of the transport operator, extending their applicability to problems that are considered difficult [18]. Though other similar methods exist, we will focus on DSA, which was the dominant preconditioning method in the west. Initially proposed as the *synthetic method*, DSA was applied to accelerate transport problems [19, 20, 21]. In this method, low-order equations are constructed from the difference between two subsequent iterations. The low-order equations resemble the neutron diffusion equation, and the solution to this system is an estimate of the additive correction to the transport scalar flux (this correction limits to zero upon convergence). DSA was shown to solve some issues that other acceleration methods suffered from. Namely, (i) DSA is stable for problems with a high scattering ratio and (ii) much more efficient than previous acceleration schemes. However, a major drawback is that the linear form of this method (which is invulnerable to the presence of near-zero or negative flux values) is strictly limited to fixed-source problems and is not naturally extended to eigenvalue problems. DSA can be applied to eigenvalue problems, but the method must be modified in a way that introduces nonlinearities. Alcouffe acknowledged that the inclusion of nonlinearities into the DSA method introduced the possibility of numerical complications [21]. Additionally, DSA loses efficiency for optically thick mesh cells and eventually becomes divergent. It should be noted, however, that if the discretization of the low-order problem is kept consistent with the discretization of the high-order problem, the latter drawback is eliminated (this is currently limited to situations in which the same spatial mesh is used for the high- and low-order problems, and was shown in Alcouffe's DSA paper) [21].

Another method for improving the convergence rate of SI is *Quasi-Diffusion* (QD), which involves the use of angular moments of the angular flux that are higher than the first moment. With QD, nonlinear Eddington factors, which contain the second moment of the angular flux, are used to construct the low-order equations [22]. Though QD is rapidly convergent, it is not considered a "true" acceleration method, because it does not preserve the transport solution as a result of truncation errors (this is not true for acceleration methods such as DSA) [18]. However, it has been shown that QD can be made a "true" acceleration scheme with the proper choice of spatial discretization [23].

Initially aimed at reducing the storage requirements and cost of solving nodal diffusion problems, CMFD was proposed in the early 1980s [8]. A couple of decades later, CMFD was shown to be useful for transport acceleration in the production code CASMO [9]. Through the incorporation of terms that are nonlinear with respect to the scalar flux, CMFD is naturally applicable to both fixed-source and eigenvalue problems without awkward modifications (unlike DSA). The appeal of CMFD over other acceleration schemes is (i) it is easily applicable to both fixed-source and eigenvalue problems, (ii) for fixed-source problems, it has the same rapid convergence rate as DSA and other similar methods, and (iii) CMFD was conceived for problems in which the low-order spatial grid can be a "coarse," Cartesian grid relative to the "fine," possibly unstructured grid for the high-order transport equation. Likewise, the energy grid for the low-order diffusion operator can be coarse. Using a coarser spatial (or energy) mesh for the low-order problem is appealing because it can reduce the computational cost of the acceleration step (however, this can reduce the effectiveness of the acceleration step, so care must be taken when choosing the low-order discretization). These features popularized the CMFD method, which has become the method of choice for deterministic transport acceleration. The method has even been successfully applied to stochastic transport problems [24]. However, the major disadvantages of CMFD are (i) its performance degrades to the point of divergence with increasing low-order mesh cell optical thickness, and (ii) it contains several nonlinear terms that render it susceptible to numerical instability in certain scenarios.

In the last couple of decades, substantial work has been done to improve the CMFD method. These improvements have focused exclusively on the aforementioned linear instability of the method (relating to the optical thickness of the low-order mesh cells) through a variety of means including, but not limited to, (i) artificially increasing the diffusivity of the low-order operator [25], relaxing the flux prolongation update [26], and modifying the form of the prolongation update [27]. The linear instability of CMFD is suppressed by each of these methods. Specifically, they improve the performance and stability for optically thick low-order meshes (often achieving near-unconditional stability as a function of cell optical thickness for certain problems [26]) for cases in which the nonlinear instability is not an issue. With these methods, the linear instability of CMFD is mitigated but not eliminated. Additionally, a form of CMFD for parallel spatial domain decomposed cases (in which different computational processors handle calculations relating to their assigned portion of space) was developed and implemented in the neutron transport code *Michigan Parallel Characteristics-based Transport* (MPACT) [11, 28, 29].

Measures to mitigate the effect of non-positive flux values have been taken in MPACT and other codes in the form of *source splitting* [11, 30]. Due to the possibility of a negative source as a consequence of negative cross sections or large leakage from a spatial region, the source term can become negative, which may result in a negative flux solution for a region of the phase space. Source splitting makes a modification that eliminates the negative source through an adjustment to the cross sections of affected regions, at the expense of solution accuracy. Also, negative cross sections may result from the *Transport-Corrected P_0* (TCP$_0$) scattering approximation, which adjusts the scattering cross section in an attempt to maintain P$_1$ accuracy when only explicitly treating P$_0$ scattering physics [31]. The susceptibility of CMFD to non-positive solutions has made it necessary to take preventative measures, such as source splitting, that can impact the overall accuracy of the solution [32, 33], or a negative flux fix-up for intermediate iterates.

A key insight over the history of transport acceleration is that acceleration performance degradation as a function of cell optical thickness is due to disparate discretization choices for the transport and acceleration equations. Fourier analysis has been used to show that the convergence rate of the iteration scheme is rendered independent of cell optical thickness if the acceleration equation is derived directly from the discretized transport equation [21, 34]. However, such a "consistent" method that retains the ability to use a coarser spatial mesh for the low-order problem has yet to be derived (such a method would be a breakthrough in the field of transport acceleration).

Though much work has focused on the performance of CMFD for optically thick problems, the method remains susceptible to nonlinear instability. Negative flux fix-up methods, such as the aforementioned source splitting technique, circumvent the issue by avoiding opportunities for numerical breakdown. However, doing so may impact the overall convergence rate and solution accuracy. Changes to the low-order step that eliminate the possibility of nonlinear instability have been ignored in favor of these detrimental fix-up techniques. The reason for this is likely that the nonlinear instability is difficult to predict and diagnose in a transport code. Issues relating to nonlinearity only occur in certain cases, and when a small detail of the problem or solution method is changed, the issue may vanish. Notably, the GAML acceleration method that eliminates this issue while retaining applicability to eigenvalue problems was proposed a year before the first paper on CMFD [12]. We expand the functionality of GAML to modern practical applications through the development of *Linear Diffusion Acceleration* (LDA), which is the principal contribution of this thesis work.

LDA is designed to eliminate the nonlinear instabilities of CMFD through the explicit avoidance of terms that are nonlinear with respect to the scalar flux. Therefore, LDA is not susceptible to numerical failure in problems that are not guaranteed to have a strictly positive solution. If CMFD or another nonlinear method is used for such problems, the potential for numerical failure exists. LDA avoids this possibility, while retaining similar linear stability properties as CMFD for both fixed-source and eigenvalue problems. However, LDA is susceptible to the same linear instability of CMFD, in which performance degrades with increasing optical thickness of the low order mesh cells. LDA possesses the following additional advantageous qualities as a result of the fixed diffusion operator: (i) enhanced Wielandt shift properties for eigenvalue problems, and (ii) the potential for reduced cost of solving the low-order linear system through the use of advanced linear solver techniques.

1.3 Outline

Here we outline the remainder of this thesis. Chapter 2 provides a brief overview of neutron transport theory, with a focus on the relevant topics of the present thesis work. This includes descriptions of (i) the different terms that compose the NTE, (ii) some of the most common deterministic solution methods for solving the NTE, (iii) the discretization methods used in deterministic simulations, and (iv) acceleration methods. These topics serve as necessary background for the main work of this dissertation.

Chapter 3 focuses on CMFD, providing an overview of the method. We provide the detailed derivation of CMFD for different problem types to explain the inclusion of nonlinear terms. A description of the CMFD algorithm for transport applications is also given for reference and later comparison to LDA. Additionally, some modifications that have been made to the original method to improve the performance of *Power Iteration* (PI) are described. Finally, we enumerate the nonlinear terms that are present in CMFD.

Chapter 4 introduces LDA, providing the derivation of the method and overviews of the acceleration algorithm for different problem types. Utilization of the *Fredholm Alternative Theorem* (FAT) in the derivation of LDA for eigenvalue problems is shown to allow for the preservation of linearity with respect to the scalar flux. The advantages of LDA over CMFD are described here, as a consequence of the derivation details. We also provide the details of a Wielandt-shifted version of LDA that allows for improved PI convergence (which is similar to shifted CMFD, but with an advantage). The differences between LDA and the aforementioned GAML method are discussed in Appendix A.

Chapter 5 presents the process and results of discrete Fourier analysis of CMFD and LDA for fixed-source and eigenvalue problems. Fourier analysis allows for the analytic estimation of the asymptotic convergence rate of iterative, fixed-point methods by tracking the propagation of errors in the iterates across subsequent iterations. In this way, the performance can be analytically predicted for model problems without the need of numerical experiments. We compare the convergence rates of LDA and CMFD for spatially-homogeneous and heterogeneous problems, and compare the analytic prediction to estimates from numerical experiments.

Chapter 6 presents the numerical results of LDA in a 1D $\mathrm{S_N}$ transport test code, with comparisons to CMFD. The analytic predictions of the Fourier analysis are tested for problems to which the Fourier analysis cannot be applied. We also provide results that show that LDA is not susceptible to the nonlinearities of CMFD for some contrived cases. The properties of Wielandt-shifted LDA are explored for simple problems, and the potential advantages of LDA within the context of linear solver methods are discussed as well.

In Chapter 7, we provide some implementation details of LDA in the transport code MPACT. The behavior of Wielandt-shifted LDA is examined for a large, realistic case. Then, the performance of LDA is compared to CMFD for realistic problems in which the nonlinearities of CMFD do not present an issue. We also provide a case study for a problem in which CMFD is shown to diverge due to nonlinear instability, and LDA remains stable.

Finally, Chapter 8 presents the conclusion of this work. This includes the motivation for the development of the LDA technology, and a summary of the results of this thesis work. Proposed future work for the LDA method is also discussed.

CHAPTER 2

Neutron Transport Theory

This chapter provides an overview of some relevant topics to the present thesis work. The *Neutron Transport Equation* (NTE), including each of its terms, is described. Additionally, the procedure for solving *k*-eigenvalue problems is given. Then, a brief overview of computational methods for solving the NTE is provided, with an emphasis on deterministic methods. This includes selected methods for discretizing the NTE. Finally, the process of iteratively solving the NTE, with optional transport acceleration, is described.

2.1 Neutron Transport Equation

To mathematically describe the distribution of neutrons within the phase space of a nuclear system for a given state, the steady-state NTE shown below is used:

$$\begin{gathered}\boldsymbol{\Omega}\cdot\nabla\psi(\boldsymbol{x},\boldsymbol{\Omega},E)+\Sigma_t(\boldsymbol{x},E)\psi(\boldsymbol{x},\boldsymbol{\Omega},E)\\ +\int_0^\infty\int_{4\pi}\Sigma_s(\boldsymbol{x},\boldsymbol{\Omega}'\cdot\boldsymbol{\Omega},E'\to E)\psi(\boldsymbol{x},\boldsymbol{\Omega}',E')\,\mathrm{d}\Omega'\,\mathrm{d}E'\\ =\frac{1}{4\pi}\left(Q(\boldsymbol{x},E)+\chi(\boldsymbol{x},E)\int_0^\infty\nu\Sigma_f(\boldsymbol{x},E')\int_{4\pi}\psi(\boldsymbol{x},\boldsymbol{\Omega}',E')\,\mathrm{d}\Omega'\,\mathrm{d}E'\right)\ ,\\ \forall\boldsymbol{x}\in V\ ,\quad\forall\boldsymbol{\Omega}\in4\pi\ ,\quad\forall E\in[0,\infty)\ ,\end{gathered}\tag{2.1a}$$

$$\begin{gathered}\psi(\boldsymbol{x},\boldsymbol{\Omega},E)=\psi^b(\boldsymbol{x},\boldsymbol{\Omega},E)\ ,\\ \forall\boldsymbol{x}\in\delta V\ ,\quad\forall\left(\boldsymbol{\Omega}\cdot\boldsymbol{n}\right)<0\ ,\quad\forall E\in[0,\infty)\ ,\end{gathered}\tag{2.1b}$$

$$\boldsymbol{x}\equiv\begin{bmatrix}x\\y\\z\end{bmatrix}\ ,\tag{2.1c}$$

$$\boldsymbol{\Omega} \equiv \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} \equiv \begin{bmatrix} \sqrt{1-\mu^2}\cos\gamma \\ \sqrt{1-\mu^2}\sin\gamma \\ \mu \end{bmatrix} . \tag{2.1d}$$

The symbols in the NTE are the following: $\boldsymbol{x}$ is the spatial position vector, $\boldsymbol{\Omega}$ is the direction vector, E is the neutron energy, Σ is the macroscopic cross section, ψ is the neutron angular flux, ν is the average number of neutrons produced per fission, χ is the fission spectrum, Q is the inhomogeneous source, V is the convex physical domain, δV is the boundary of the convex physical domain, and $\boldsymbol{n}$ is the outward surface normal vector. Additionally, μ is the cosine of the polar angle, which is the angle that $\boldsymbol{\Omega}$ makes with the z-axis, and γ is the azimuthal angle.

Equation (2.1a) is a linear integro-differential equation spanning six dimensions of phase space (three in space, two in angle, and one in energy). Time-dependence can also be included for transient problems. In this thesis, we are primarily concerned with steady-state problems. The solution of Eq. (2.1a) is the neutron angular flux ψ, which is the product of the neutron number density and neutron velocity at a given point in phase space. For problems with non-negative internal and boundary sources, the solution of Eq. (2.1a) is guaranteed to be non-negative. The NTE describes the mean neutron flux averaged over an infinite number of neutron histories. Therefore, it is only valid for cases with sufficiently large neutron populations.

The NTE describes the "balance" of neutrons in a nuclear system and can be considered a conservation equation. For a given infinitesimally small piece of phase space, the *Left-Hand-Side* (LHS) describes the mechanisms of neutron loss and the *Right-Hand-Side* (RHS) describes the mechanisms of neutron gain. If neutron gain is greater than neutron loss, or vice-versa, the neutron population will change with time. In this thesis work, only steady-state problems are considered. Thus, overall neutron losses and gains are equal (the rate of change in the neutron population with time is zero). The mechanisms of neutron loss for steady-state problems are:

- streaming of neutrons $(\boldsymbol{\Omega} \cdot \nabla\psi(\boldsymbol{x}, \boldsymbol{\Omega}, E))$, and
- collisional loss $(\Sigma_t(\boldsymbol{x}, E)\psi(\boldsymbol{x}, \boldsymbol{\Omega}, E))$.

The mechanisms of neutron gain are:

- scattering source $\left(\int_0^\infty \int_{4\pi} \Sigma_s(\boldsymbol{x}, \boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}, E' \to E)\psi(\boldsymbol{x}, \boldsymbol{\Omega}', E')\, \mathrm{d}\Omega'\, \mathrm{d}E'\right)$,
- fission source $\left(\frac{\chi(\boldsymbol{x},E)}{4\pi} \int_0^\infty \nu\Sigma_f(\boldsymbol{x}, E') \int_{4\pi} \psi(\boldsymbol{x}, \boldsymbol{\Omega}', E')\, \mathrm{d}\Omega'\, \mathrm{d}E'\right)$, and
- neutrons born from an inhomogeneous source $\left(\frac{1}{4\pi} Q(\boldsymbol{x}, E)\right)$.

When solving the NTE computationally, the scalar flux is generally stored rather than the full angular flux. This is because most quantities of interest are a function of the scalar flux, ϕ_0. The scalar flux is the zeroth angular moment of the angular flux:

$$\phi_0(\boldsymbol{x}, E) \equiv \int_{4\pi} \psi(\boldsymbol{x}, \boldsymbol{\Omega}, E)\,\mathrm{d}\Omega \ , \tag{2.2}$$

and can be interpreted as the neutron track length generated per unit volume and unit time. When multiplied by the macroscopic cross section for a given reaction type, the scalar flux can be used to obtain reaction rates. Another useful quantity that is necessary for many acceleration methods is the *neutron current*, which is the first angular moment of the angular flux, shown below:

$$\boldsymbol{\phi}_1(\boldsymbol{x}, E) \equiv \int_{4\pi} \boldsymbol{\Omega}\, \psi(\boldsymbol{x}, \boldsymbol{\Omega}, E)\,\mathrm{d}\Omega \ . \tag{2.3}$$

This and other vector quantities are represented in boldface notation.

Often, nuclear engineers are interested in a specific form of Eq. (2.1a) that is applicable to physical nuclear cores. This form is mathematically described as an eigenvalue problem, and possesses an infinite number of solutions that differ by an arbitrary multiplicative constant. The next section discusses the eigenvalue form of the NTE.

2.2 *k*-Eigenvalue Problems

Generally, if fissionable isotopes are present, Eqs. (2.1) are cast as an eigenvalue problem. To obtain the equation that describes *k*-eigenvalue problems, the following changes are made to Eqs. (2.1): (i) the boundary and inhomogeneous sources are set to zero, and (ii) the fission source is multiplied by the eigenvalue $\lambda = \frac{1}{k}$. Thus, the fission source is scaled to balance other forms of neutron gain and loss. By making these modifications, we obtain:

$$\begin{gathered}
\boldsymbol{\Omega} \cdot \nabla \psi(\boldsymbol{x}, \boldsymbol{\Omega}, E) + \Sigma_t(\boldsymbol{x}, E)\psi(\boldsymbol{x}, \boldsymbol{\Omega}, E) \\
= \int_0^\infty \int_{4\pi} \Sigma_s(\boldsymbol{x}, \boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}, E' \to E)\psi(\boldsymbol{x}, \boldsymbol{\Omega}', E')\,\mathrm{d}\Omega'\,\mathrm{d}E' \\
+\lambda \frac{\chi(\boldsymbol{x}, E)}{4\pi} \int_0^\infty \nu\Sigma_f(\boldsymbol{x}, E') \int_{4\pi} \psi(\boldsymbol{x}, \boldsymbol{\Omega}', E')\,\mathrm{d}\Omega'\,\mathrm{d}E' \ , \\
\forall \boldsymbol{x} \in V \ , \quad \forall \boldsymbol{\Omega} \in 4\pi \ , \quad \forall E \in [0, \infty) \ ,
\end{gathered} \tag{2.4a}$$

$$\begin{gathered}
\psi(\boldsymbol{x}, \boldsymbol{\Omega}, E) = 0 \ , \\
\forall \boldsymbol{x} \in \delta V \ , \quad \forall\, (\boldsymbol{\Omega} \cdot \boldsymbol{n}) < 0 \ , \quad \forall E \in [0, \infty) \ ,
\end{gathered} \tag{2.4b}$$

with the following arbitrarily-chosen normalization condition to define a unique solution:

$$P = \int_V \int_0^\infty \int_{4\pi} \nu\Sigma_f(\boldsymbol{x}, E)\psi(\boldsymbol{x}, \boldsymbol{\Omega}, E)\, \mathrm{d}\Omega\, \mathrm{d}E\, \mathrm{d}V \ , \tag{2.4c}$$

where P is the total neutron production rate of the system. As with all eigenvalue problems, Eqs. (2.4a) and (2.4b) have the trivial solution $\psi = 0$. However, the goal is to obtain λ for which a non-zero solution ψ exists. Specifically, we seek the *fundamental eigenpair*, which we define as the pair of corresponding λ and ψ for which λ has the smallest real part.

Rather than report the λ-eigenvalue, nuclear engineers usually work with k_{eff} (referred to as the *criticality eigenvalue* or *multiplication factor*) which describes the criticality of the system. k_{eff} is the reciprocal of the fundamental eigenvalue of the NTE, defined below:

$$k_{\text{eff}} \equiv \frac{1}{\lambda_0} \ , \tag{2.5}$$

where λ_0 is the fundamental eigenvalue. If $k_{\text{eff}} = 1$, the system is considered *critical* and a steady-state, non-negative solution of the NTE exists that corresponds to this criticality eigenvalue. If $k_{\text{eff}} > 1$, the system is *supercritical*, and for a physical problem, the magnitude of the flux would increase in time. If $k_{\text{eff}} < 1$, the system is *subcritical*, and for a physical problem, the magnitude of the flux would decrease in time. By analyzing eigenvalue problems, it is possible to study systems that, in their natural physical state, would not have steady-state solutions.

Equation (2.4a) is fundamental to reactor analysis, and obtaining a solution for nuclear cores is usually computationally expensive. Iterative methods exist that are suitable for computing a solution to Eq. (2.4a), including *Power Iteration* (PI). Section 2.4.1.2 describes PI in more detail. Prior to the discussion of iterative methods, the next section provides a discussion of two classes of methods for solving the NTE computationally.

2.3 Computational Methods

There are two overarching categories of transport solution methods: *stochastic* and *deterministic*. Stochastic methods utilize random number generation and probability distributions to track individual particle interactions that collectively contribute to estimating quantities of interest. Deterministic methods combine discretization methods that approximate the NTE by a (hopefully, accurate) algebraic system of equations, and iterative methods that solve the algebraic system (hopefully, efficiently). The two categories are described in the following subsections (with emphasis on deterministic methods, which are the focus of this thesis).

2.3.1 Stochastic Methods

Often referred to colloquially as "Monte Carlo" methods, stochastic approaches to solving the NTE track the "history" of a large number of randomly generated, individual neutrons (and sometimes other particles, such as photons or electrons). The results are then averaged to obtain an estimate of the collective neutron behavior. A "history" refers to the events from inception to termination of a particle's lifetime. Therefore, solution accuracy and precision depends largely on (i) the number of particles simulated and (ii) properly simulating the physics involved in the lifetime of each particle. Using a random number generator, probability distributions that describe different phenomena can be sampled to determine different quantities. For neutrons, these quantities include:

- the distance to collision with a nucleus,
- the type of collision interaction with a nucleus,
- the outgoing energy and angle from a scattering event,
- the number of neutrons born from fission, and
- the energy and angle of a neutron born from fission.

By sampling probability distributions that describe these phenomena, each event in a neutron's history can be simulated. A neutron history ends when it is absorbed or leaks from the system.

Generally, the discretization of phase space is not needed for stochastic methods. Therefore, detailed solutions obtained through these methods are considered highly accurate and are often used as a reference for deterministic solutions [35]. Unfortunately, while accuracy is a strength of stochastic methods, precision is a weakness. Each quantity of interest must be selected for computation before the simulation is begun and is subject to statistical error due to the probabilistic nature of the method. The statistical error reduces with the number of particle histories, but the scaling of the statistical error is proportional to $\frac{1}{\sqrt{N}}$ where N is the number of histories simulated. Thus, the statistical error decreases increasingly slowly as N increases. For example, to decrease the statistical error by a factor of 10, it is necessary to increase the number of histories (and the total cost of the calculation) by a factor of 100.

The number of particle histories required to obtain reasonably precise results is extremely large for full-scale, whole-core calculations. Though techniques of variance reduction exist to reduce the number of particles required for a given precision, and the simulation of particle histories is highly parallelizable, the number of particles required for full-core simulations can be excessive and enormously computationally expensive.

For eigenvalue problems, stochastic simulations emulate PI by continuously updating the fission source. "Batches" of neutrons are simulated in "cycles", and the resulting fission neutrons are stored in a fission bank. By carefully tracking the number of fission neutrons between subsequent batches, k_{eff} can be estimated. However, the fission source must be sufficiently converged before accurate estimates can be made. Therefore, several "inactive" batches are typically simulated before results are recorded in the "active" cycles. [36]

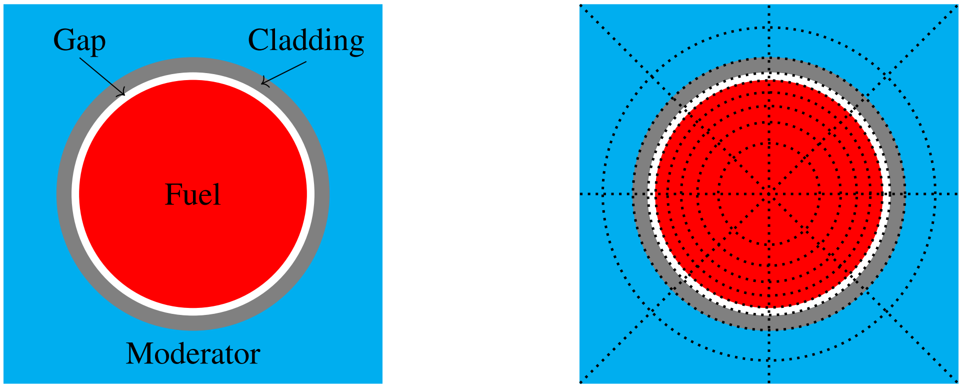
2.3.2 Deterministic Methods

Deterministic methods provide solutions to the NTE by (i) *discretizing* the NTE, thereby approximating it by a (typically large) algebraic system of equations, and (ii) typically, iteratively solving the (large) system of equations. This section discusses different approximations that are made to discretize the NTE in space, angle, and energy. Additionally, a relevant method for solving the 3D NTE is described, as well as a common method used for simplifying the scattering physics.

2.3.2.1 Spatial Discretization

Spatial discretization refers to the partitioning of the continuous space into a system of discrete cells. A system is composed of many different contiguous material regions that can be subdivided into smaller cells. The material properties and flux solution within a cell are typically assumed to be constant. The source can also be assumed to be constant within a cell, but higher-order sources are possible. Depending on the type of deterministic method, the true geometry may be homogenized within a computational cell so that the material properties can be assumed constant. A potential computational mesh for a pin cell, such as those used in MPACT, is shown in Fig. 2.1.

To discretize neutronics equations in space, there are three different classes of spatial discretization methods that are commonly used. The first is *finite difference* methods in which spatial derivatives are approximated by numerical differentiation. This method is commonly used for the NBE, which is obtained by integrating the NTE over angle, and also for $\mathrm{S_N}$ transport (to be described shortly). In finite difference methods, a grid is first imposed on the spatial domain. Then, spatial derivatives are approximated with numerical difference relations. These methods are easily applied to problems with Cartesian meshes, but are more difficult for problems with more complex spatial meshes. Finite difference is also commonly used to discretize the low-order equations in acceleration methods, such as



(a) Material regions (b) Computational mesh

Figure 2.1: Material regions and computational mesh for a pin cell

CMFD or LDA. [37]

The second common spatial discretization method is *finite volume*, which involves (i) an integral of the governing equation over a spatial cell (to obtain a “conservation” equation), combined with (ii) auxiliary finite-difference-like equations. Often, the problem is divided into a set of control volumes in which the solution is assumed to be flat. Neutron loss and gain are carefully accounted for by keeping track of the incoming and outgoing particles, as well as any internal losses or gains. One such method is the *Method of Characteristics* (MOC), in which the NTE is cast along a set of characteristic lines that pass through the spatial domain. In this way, the NTE is converted from a set of partial differential equations to ordinary differential equations that can be more easily solved [38]. In 1D, MOC is considered a finite volume method. In multidimensional problems, MOC is related to finite volume methods.

Finally, there are *finite element* methods, in which the spatial domain is discretized into a mesh of contiguous, non-overlapping finite elements. A set of basis functions are chosen to represent the spatial shape of the solution, with unknown coefficients. Upon obtaining these unknown coefficients, the solution is determined. Different choices for the mesh shape and basis functions are available for finite element methods, which are becoming more widely used in reactor physics codes. [37]

2.3.2.2 Energy Discretization (Multigroup Approximation)

For deterministic methods, the multigroup approximation is almost always made for the energy discretization. The continuous energy spectrum is discretized into intervals in which the microscopic cross sections are considered constant. For the energy bounds of each group g, the value E_g is the group lower bound and E_{g-1} is the group upper bound. Thus, a smaller group index refers to a higher energy. A total of G energy groups are defined in this way. To reduce the computational resources to solve the energy-dependent NTE, a few hundred energy groups are typically used, rather than the hundreds of thousands that would be required to accurately "resolve" the continuous spectrum. Because the cross section has a strong energy dependence, care must be taken when choosing the energy intervals for the energy discretization. This is especially true in the resolved *resonance region*, where the cross section can vary by orders of magnitude in very small energy intervals.

The multigroup approximation is made by integrating the NTE over a given energy interval, as shown below:

$$\begin{aligned}\left[\boldsymbol{\Omega}\cdot\nabla + \Sigma_{t,g}(\boldsymbol{x},\boldsymbol{\Omega})\right]\psi_g(\boldsymbol{x},\boldsymbol{\Omega}) &= \frac{1}{4\pi}\left[\sum_{g'=1}^{G}\int_{4\pi}\Sigma_{s,g'\to g}(\boldsymbol{x},\boldsymbol{\Omega}\cdot\boldsymbol{\Omega}')\psi_{g'}(\boldsymbol{x},\boldsymbol{\Omega}')\,\mathrm{d}\Omega'\right.\\ &\left.\quad+\frac{\chi_g(\boldsymbol{x})}{k_{\text{eff}}}\sum_{g'=1}^{G}\left(\nu\Sigma_f\right)_{g'}(\boldsymbol{x})\phi_{0,g'}(\boldsymbol{x})\right] , \\ &\forall\boldsymbol{x}\in V\ ,\quad \forall\boldsymbol{\Omega}\in 4\pi,\quad \{g\,|\,g\in\mathbb{N}, 1\le g\le G\}\ ,\end{aligned} \tag{2.6a}$$

with the following multigroup form of the previously-chosen normalization condition:

$$P = \sum_{g=1}^{G}\int_V \left(\nu\Sigma_f\right)_g(\boldsymbol{x})\phi_{0,g}(\boldsymbol{x})\,\mathrm{d}V\ , \tag{2.6b}$$

where the group-wise scalar flux is defined as

$$\phi_{0,g}(\boldsymbol{x}) \equiv \int_{4\pi}\psi_g(\boldsymbol{x},\boldsymbol{\Omega})\,\mathrm{d}\Omega\ . \tag{2.6c}$$

The multigroup quantities are defined below:

$$\psi_g(\boldsymbol{x},\boldsymbol{\Omega}) \equiv \int_{E_g}^{E_{g-1}}\psi(\boldsymbol{x},\boldsymbol{\Omega},E')\,\mathrm{d}E'\ , \tag{2.7a}$$

$$\chi_g(\boldsymbol{x}) \equiv \int_{E_g}^{E_{g-1}}\chi(\boldsymbol{x},E')\,\mathrm{d}E'\ , \tag{2.7b}$$

$$\Sigma_{t,g}(\boldsymbol{x},\boldsymbol{\Omega}) \equiv \frac{\int_{E_g}^{E_{g-1}}\Sigma_t(\boldsymbol{x},E')\psi(\boldsymbol{x},\boldsymbol{\Omega},E')\,\mathrm{d}E'}{\psi_g(\boldsymbol{x},\boldsymbol{\Omega})}\ , \tag{2.7c}$$

$$\left(\nu\Sigma_f\right)_g(\boldsymbol{x}) \equiv \frac{\int_{E_g}^{E_{g-1}}\nu\Sigma_f(\boldsymbol{x},E')\phi_0(\boldsymbol{x},E')\,\mathrm{d}E'}{\phi_{0,g}(\boldsymbol{x})}\ , \tag{2.7d}$$

$$\Sigma_{s,g'\to g}(\boldsymbol{x},\boldsymbol{\Omega},\boldsymbol{\Omega}') \equiv \frac{\int_{E_g}^{E_{g-1}}\int_{E_g'}^{E_{g'-1}}\Sigma_s(\boldsymbol{x},\boldsymbol{\Omega}'\cdot\boldsymbol{\Omega},E'\to E)\psi(\boldsymbol{x},\boldsymbol{\Omega}',E')\,\mathrm{d}E'\,\mathrm{d}E}{\psi_{g'}(\boldsymbol{x},\boldsymbol{\Omega}')}\ . \tag{2.7e}$$

No approximations have been made in Eq. (2.6a) and Eqs. (2.7). However, Eqs. (2.7) require knowledge of the angular flux, which is the solution to be obtained by solving the NTE. Therefore, it is necessary to assume a solution before the cross sections are "collapsed" in Eqs. (2.7). A problem that is simplified in space can be solved to generate a weighting function (or *neutron spectrum*) that approximates the true solution to evaluate Eqs. (2.7) before performing a more rigorous spatial calculation with the collapsed cross sections. The use of a weighting function assumes that the angular flux is separable in angle and energy in a given energy group:

$$\begin{gathered}\psi(\boldsymbol{x},\boldsymbol{\Omega},E) \approx \varphi(\boldsymbol{x},E)\Psi_g(\boldsymbol{x},\boldsymbol{\Omega}) \ , \\ E_{g-1} < E < E_g \ , \end{gathered} \tag{2.8}$$

where φ is the energy-dependent weighting function and Ψ_g is the energy-independent angular flux. One may preserve some spatial dependence by choosing a weighting function that is a local solution for a spatial subregion of the problem. Upon making this assumption, the multigroup cross sections can be defined for use in practical problems. As an example, the total cross section then becomes:

$$\Sigma_{t,g}(\boldsymbol{x}) \equiv \frac{\int_{E_g}^{E_{g-1}} \Sigma_t(\boldsymbol{x},E')\varphi(\boldsymbol{x},E')\,\mathrm{d}E'}{\int_{E_g}^{E_{g-1}} \varphi(\boldsymbol{x},E')\,\mathrm{d}E'} \ . \tag{2.9}$$

Often, the angular dependence of the weighting function is neglected, and the scalar flux is used. Once the cross sections have been collapsed, they may resemble those shown in Fig. 2.2, which depicts a hypothetical set of collapsed multigroup cross section overlaid on the continuous cross section spectrum [39].

2.3.2.3 Angular Discretization

There are two common treatments of the discrete directional variable: the spherical harmonics ($\mathrm{P_N}$) expansion and discrete ordinates ($\mathrm{S_N}$). With $\mathrm{S_N}$, the angular variable is approximated with a quadrature set of M discrete directions and weights (except for in 1D, generally $M > N$):

$$\{\boldsymbol{\Omega}_n, w_n \mid n \in \mathbb{N}, 1 \le n \le M\} \ . \tag{2.10a}$$

Given this set, an integration of a function over angle can be approximated as

$$\int_{4\pi} f(\boldsymbol{\Omega})\,\mathrm{d}\Omega \approx \sum_n w_n f(\boldsymbol{\Omega}_n) \ , \tag{2.10b}$$

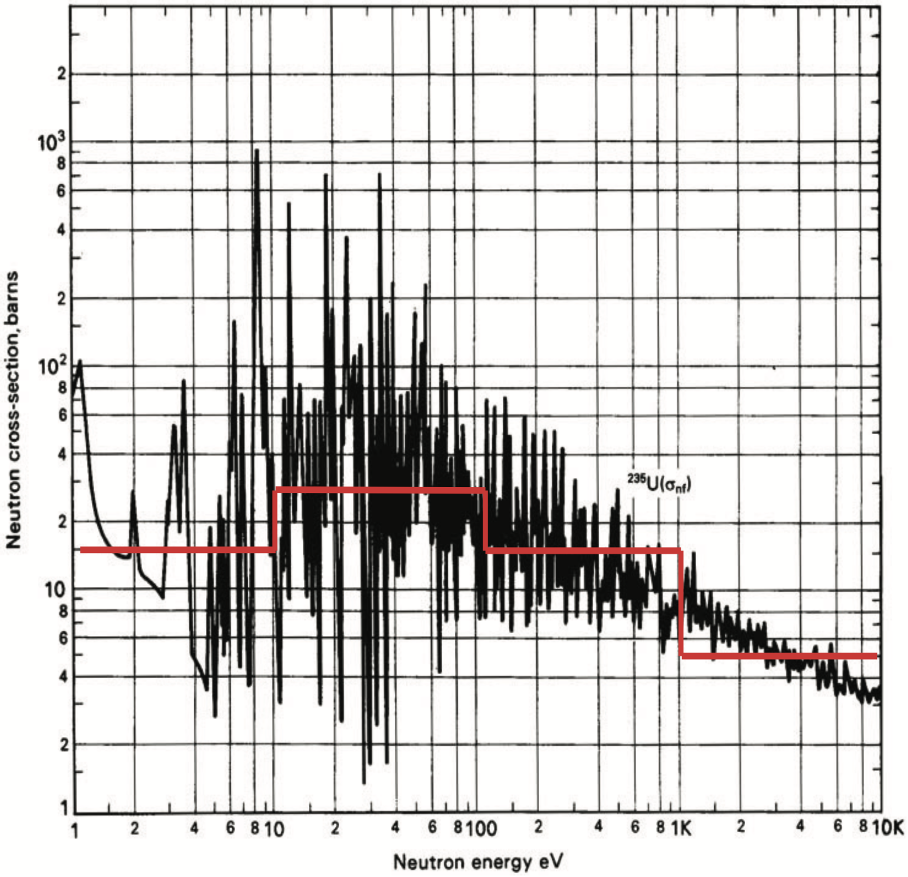


Figure 2.2: Theoretical representation of multigroup cross sections for U-235. Multigroup cross section estimates are overlaid on the continuous spectrum. [39]

where

$$\sum_n w_n = 4\pi \ . \tag{2.10c}$$

The choice of quadrature set can change the accuracy of the solution for a given number of discrete directions. Therefore, an informed choice can be made depending on the expected angular dependence of the problem. Combining the multigroup approximation with the discrete ordinates approximation yields the following form of the isotropically-scattering NTE for eigenvalue problems:

$$\begin{aligned} \left[\boldsymbol{\Omega}_n \cdot \nabla + \Sigma_{t,g}(\boldsymbol{x})\right] \psi_{g,n}(\boldsymbol{x}) &= \sum_{g'=1}^{G} \sum_{n'=1}^{N} w_n \Sigma_{s,g' \to g, n' \to n}(\boldsymbol{x}) \psi_{g',n'}(\boldsymbol{x}) \\ &+ \lambda \frac{\chi_g(\boldsymbol{x})}{4\pi} \sum_{g'=1}^{G} \left(\nu \Sigma_f\right)_{g'}(\boldsymbol{x}) \phi_{0,g'}(\boldsymbol{x}) \end{aligned} \tag{2.11a}$$

$$\forall \boldsymbol{x} \in V \ , \quad \{g \mid g \in \mathbb{N}, 1 \le g \le G\} \ , \quad \{\boldsymbol{\Omega}_n, w_n \mid n \in \mathbb{N}, 1 \le n \le M\} \ ,$$

$$\begin{gathered}\psi_{g,n}(\boldsymbol{x}) = 0 \ , \\ \forall \boldsymbol{x} \in \delta V \ , \quad \forall \left(\boldsymbol{\Omega}_n \cdot \boldsymbol{n}\right) < 0 \ , \quad \forall g \ , \end{gathered} \tag{2.11b}$$

with the following previously-chosen normalization condition:

$$P = \sum_{g=1}^{G} \int_V \left(\nu\Sigma_f\right)_g (\boldsymbol{x}) \phi_{0,g}(\boldsymbol{x}) \, \mathrm{d}V \ . \tag{2.11c}$$

With $\mathrm{P_N}$, the angular variable is removed entirely by expanding angularly-dependent quantities in terms of the spherical harmonics functions. The multigroup angular flux is expanded as shown in Eq. (2.12), where $\psi_{n,m,g}(\boldsymbol{x})$ are the expansion coefficients and $Y_n^m(\boldsymbol{\Omega})$ are the spherical harmonics functions:

$$\psi_g(\boldsymbol{x}, \boldsymbol{\Omega}) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \psi_{n,m,g}(\boldsymbol{x}) Y_n^m(\boldsymbol{\Omega}) \ . \tag{2.12}$$

In practice, the order of the expansion is truncated at N:

$$\psi_g(\boldsymbol{x}, \boldsymbol{\Omega}) \approx \sum_{n=0}^{N} \sum_{m=-n}^{n} \psi_{n,m,g}(\boldsymbol{x}) Y_n^m(\boldsymbol{\Omega}) \ . \tag{2.13}$$

By utilizing this expansion method, the angular variable can be eliminated entirely from the NTE. A set of equations can be obtained for the expansion coefficients that are solved to obtain an approximation for the angular flux.

2.3.2.4 2D/1D Method

The 2D/1D method is based on the separation of the 3D NTE into a set of coupled radial and axial equations, with *Transverse Leakage* (TL) terms that couple the two equations [40, 41]. This method is based on the observation that nuclear reactors are highly heterogeneous in the radial (x, y) direction and mostly homogeneous in the axial (z) direction. Therefore, different discretization choices and solution methodologies may be used for each equation to more efficiently solve the problem while maintaining reasonable accuracy. Axially, the core is split into a set of planes within which the radial equation is solved. Radially, the core is usually discretized based on the grid of pin cells. In this section, k is used as the plane index and ij are the radial cell indices.

The monoenergetic 2D/1D equations are shown in Eqs. (2.14), where Eq. (2.14a) is the *radial equation* and Eq. (2.14b) is the *axial equation*. These equations are derived by separately integrating the NTE over the radial and axial directions:

$$\left[(\mathbf{\Omega}\cdot\nabla)_{xy} + \Sigma_{t,k}(x,y)\right]\psi_k(x,y,\mathbf{\Omega}) = \frac{Q_k(x,y)}{4\pi} - TL_k(x,y,\mathbf{\Omega})\ , \tag{2.14a}$$

$$\left[\mu\frac{\partial}{\partial z} + \Sigma_{t,k,ij}(z,\mathbf{\Omega})\right]\psi_{ij}(z,\mathbf{\Omega}) = \frac{Q_{ij}(z)}{4\pi} - TL_{ij}(z,\mathbf{\Omega})\ . \tag{2.14b}$$

In these equations, TL represents the coupling leakage terms and Q represents the fission and scattering sources. The assumed angular dependence of the TL terms that couple the 2D and 1D equations can vary, which affects the solution accuracy, but the angular dependence is generally assumed to be isotropic. Further details about other treatments are not discussed here, but can be found in [32, 42].

The positivity of the solution of the 2D/1D equations is unknown. However, iterates of the solution to these equations can be, and have been observed to be, negative. One way in which the solution can become negative is if the TL term in a spatial region is large enough to drive the RHS negative. In this case, the solution to the 2D/1D equations may contain negative values for certain portions of the phase space. This can present issues for certain nonlinear iterative methodologies, in which non-positive solutions can drive the iterative process unstable.

The *splitting method* was developed to avoid the presence of negative source terms [32, 43]. In this method, the RHS is combined with the total cross section in the LHS as shown in Eqs. (2.15). This ensures that the source term is zero, and thus, non-negative. Using this technique, we obtain:

$$\left[(\mathbf{\Omega}\cdot\nabla)_{xy} + \tilde{\Sigma}_{t,k}(x,y)\right]\psi_k(x,y,\mathbf{\Omega}) = 0\ , \tag{2.15a}$$

$$\tilde{\Sigma}_{t,k}(x,y) \equiv \Sigma_{t,k}(x,y) + \frac{4\pi\tilde{L}_z}{\phi_{0,k}(x,y)}\ , \tag{2.15b}$$

$$\tilde{L}_z \equiv -\left[\frac{Q_k(x,y)}{4\pi} - TL_k(x,y,\mathbf{\Omega})\right]\ , \tag{2.15c}$$

where in Eq. (2.15b) we have made the following (detrimental) assumption:

$$\psi_k(x,y,\mathbf{\Omega}) \approx \frac{\phi_{0,k}(x,y)}{4\pi}\ , \tag{2.15d}$$

which is made because the angular flux is unknown. Though splitting removes the issue of negative sources, this approximation lowers the accuracy of the overall solution due to incorrect weighting. The errors introduced by splitting can be significant enough to overshadow other improvements to the 2D/1D method [42]. We hope to remove the need for splitting with this thesis work.

2.3.2.5 Transport-Corrected Scattering

The *Transport-Corrected* P_0 (TCP$_0$) scattering methodology was developed with the aim of obtaining the accuracy of linearly-anisotropic scattering (P$_1$) with the computational expense of modeling isotropic scattering (P$_0$) [44]. This is accomplished in monoenergetic problems by modifying the self-scatter and total cross sections, as shown in Eqs. (2.16). A few TCP$_0$ approximations exist, and in the most common formulation, the linearly anisotropic moment of the scattering cross section is simply subtracted from the zeroth scattering moment and the total cross section is modified accordingly:

$$\hat{\Sigma}_{s0} \equiv \Sigma_{s0} - \Sigma_{s1} \ , \tag{2.16a}$$

$$\Sigma_{tr} \equiv \Sigma_t - \Sigma_{s1} \ , \tag{2.16b}$$

where Σ_{tr} is the transport cross section. Intuitively, by reducing the total cross section, neutrons will travel further in a given direction before colliding with the medium. This emulates linearly-anisotropic scattering, in which neutrons may be preferentially forward-scattered.

For multigroup problems, different methods have been developed to correct the self-scatter and total cross sections, with one of the most accurate being the *inscatter method*. In this method, an estimate of the neutron current spectrum $\phi_{1,g}$, often obtained by solving a simple infinite medium problem, is used to weight the cross section modification:

$$\hat{\Sigma}_{s0,g\to g} \equiv \Sigma_{s0,g\to g} - \frac{1}{\phi_{1,g}} \sum_{g'=1}^{G} \Sigma_{s1,g'\to g}\phi_{1,g'} \ , \tag{2.17a}$$

$$\Sigma_{tr,g} \equiv \Sigma_{t,g} - \frac{1}{\phi_{1,g}} \sum_{g'=1}^{G} \Sigma_{s1,g'\to g}\phi_{1,g'} \ . \tag{2.17b}$$

Though the TCP$_0$ method can provide improved accuracy over the assumption of isotropic scattering, it opens the possibility of negative cross sections. In particular, the transport-corrected self scatter cross section of water is often negative in the epithermal energy region due to the strong scattering anisotropy of hydrogen nuclei. Figure 2.3 shows a plot of the transport-corrected self-scatter cross sections for water at 293 K in the 47-group MPACT library [31], with many negative values in the epithermal range. This is especially true for problems with a large number of energy groups G, due to the summation over the anisotropic inscatter cross sections. For cases in which the cross sections can become

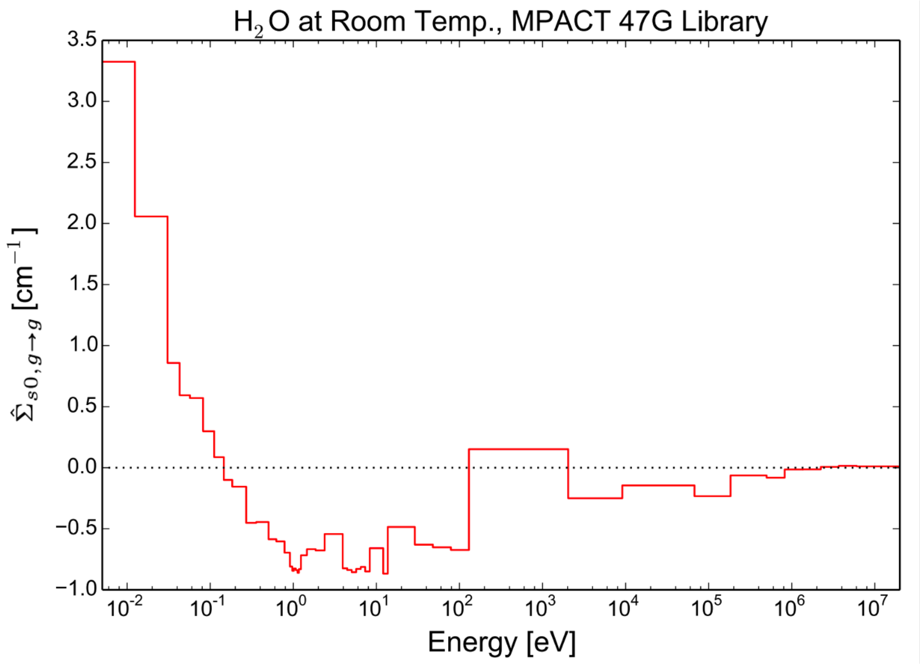


Figure 2.3: TCP_0 self-scatter XS for water at 293 K in the 47-group MPACT library [31]

negative, the solution can also become negative as a result of the TCP_0 approximation. If nonlinear iterative methods are employed to solve such approximate equations, the iterative process may become unstable as a result of these non-positive solutions.

2.4 Transport Iterative Methods

In the previous sections, we have summarized methods for discretizing the NTE so that it can be solved computationally. Now, we consider a sample discretized problem solved using MPACT. For this problem, we assume the following reasonable number of unknowns for each phase space variable: (i) $O(10^6)$ spatial cells in a plane, (ii) $O(10^2)$ axial planes, (iii) $O(10^2)$ energy groups, and (iv) $O(10^2)$ discrete angles. For such a problem, there are $O(10^{12})$ unknowns for each source iteration. If the problem converges slowly, requiring $O(10^3)$ iterations for sufficient convergence, then these $O(10^{12})$ unknowns must be computed $O(10^3)$ times. Given these estimates, the total number of unknowns that must be found (including intermediate values before convergence has been achieved) is $O(10^{15})$. Obtaining a solution within a reasonable amount of time given this amount of computational work would be unrealistic with current computational resources, even with high-performance machines.

Transport acceleration utilizes the solution of a simpler, low-order problem (relative to the transport problem) to reduce the computational effort required to iteratively obtain a solution to the NTE. Modern acceleration methods generally reduce the number of iterations to converge to a reasonable tolerance from $O(10^3)$ to $O(10^1)$. The subject of this thesis is LDA, which has certain advantages over current acceleration methods. Sections 2.4.1 and 2.4.2 provide an overview of iterative transport and acceleration methods, with more details on CMFD and LDA provided in Chapters 3 and 4, respectively. Section 2.4.3 provides a discussion of the linear solver methods used to obtain the low-order solution.

2.4.1 Iterative Algorithms for Solving the NTE

In this section, a few different iterative methods for solving the NTE are briefly described. This includes *Source Iteration* (SI) for converging the scattering source and *Power Iteration* (PI) for obtaining the fundamental eigenpair for eigenvalue problems. The terminology used in this subsection refers to the case in which no acceleration method is used.

2.4.1.1 Source Iteration

Source Iteration (SI) refers to the process of converging the scattering source for a given problem. In this type of iteration scheme, sources other than the scattering source are considered to be constant. Thus, in the case of an eigenvalue problem, the fission source is fixed and is not updated throughout the SI process. An iterative approach is usually taken to converge the group-wise scattering source by first estimating the scattering source in the highest energy group and then looping over the remaining lower energy groups. The cumulative inscatter source from all other energy groups is used as the scattering source for a given energy group. Within a given energy group, the source is updated for all spatial regions.

The SI process is described in Algorithm 1. First, a guess is made for the initial scattering source. Then, a *transport sweep* is performed in which the angular flux is updated for all spatial regions for a single group, starting with the highest energy group (usually corresponding to $g = 1$) and working down through the lower groups. Multiple transport sweeps can be performed, but the default in MPACT is one transport sweep per source iteration per group. During the sweep process, the scalar flux is usually also computed and stored. Other information, such as angular flux moments that are necessary for reconstructing the scattering source, can also be stored. (We note that the default behavior in MPACT is to setup the scattering source before the group and transport sweep loops.)

Algorithm 1 Source Iteration Algorithm

```
Assume an initial guess for the scattering source in all energy groups
repeat
    for each energy group $g$ do
        for each transport sweep do
            Compute the scattering source from other groups for all regions and angles
            Compute the angular flux estimate for all regions and angles
            Compute and store the scalar flux and other relevant quantities
        end for
    end for
until convergence of the scattering source
```

SI is highly stable, but it converges slowly for scattering-dominated problems. The physical interpretation of SI provides insight for the convergence rate of the iterative process. For fixed-source cases, each source iteration l describes neutrons that have been scattered l times. That is, the solution for the zeroth iteration is the uncollided flux (for a zero initial guess), the solution to the first iteration is the once-collided flux, and so on. Therefore, $\psi^{(l)}$ is the angular flux of neutrons that have experienced exactly l collisions since their birth. Once this is understood, it is clear that problems with a high scattering ratio (meaning that for at least one material in at least one region of space, neutrons will preferentially scatter rather than be absorbed) are slow to converge (requiring a large number of iterations to arrive at a converged solution).

It can be shown mathematically through Fourier analysis that the convergence rate of SI for fixed-source problems depends directly on two quantities: (i) the maximum non-absorption probability and (ii) the maximum non-leakage probability. The product of these two quantities is an upper bound of the spectral radius ρ, which mathematically represents the error reduction in the scalar flux for each iteration. In other words, an upper bound of the spectral radius is equal to the maximum probability that a neutron's next event will not be the termination of its history. For convergent iterative methods (in which the algorithm is expected to produce a converged solution within a finite number of iterations), the spectral radius exists in the interval $0 < \rho < 1$. As ρ decreases, the number of iterations required to converge to a specified tolerance decreases. The spectral radius describes the reduction in error of the scalar flux (or more generally, the solution) and can be estimated in practice as:

$$\rho^{(l)} \equiv \frac{\left\| \phi_0^{(l)} - \phi_0^{(l-1)} \right\|}{\left\| \phi_0^{(l-1)} - \phi_0^{(l-2)} \right\|} . \tag{2.18}$$

As $l \to \infty$, $\rho^{(l)}$ becomes a better estimate of the true spectral radius (which is why we can also refer to the spectral radius as the *asymptotic convergence rate*). For eigenvalue problems, *Power Iteration* (PI) (described in Section 2.4.1.2) effectively behaves like SI for *very highly-scattering* fixed-source problems.

2.4.1.2 Power Iteration

PI refers to the process of iteratively solving an eigenvalue problem by (i) estimating the neutron source, (ii) estimating the flux solution using the source estimate, and (iii) checking for convergence of the quantities of interest. This process is repeated if sufficient convergence has not been achieved. If the NTE is organized such that loss terms are on the LHS and source terms are on the RHS (as in Eq. (2.1a)), then the operator form is shown below, where the LHS operator L is the neutron loss operator, F is the fission source, and S is the scattering source:

$$L\psi(\boldsymbol{x}, \boldsymbol{\Omega}, E) = F + S \ , \tag{2.19a}$$

$$L \equiv \boldsymbol{\Omega} \cdot \nabla + \Sigma_t \ , \tag{2.19b}$$

$$F \equiv \frac{1}{k_{\text{eff}}} \frac{\chi(\boldsymbol{x}, E)}{4\pi} \int_0^\infty \nu\Sigma_f(\boldsymbol{x}, E') \int_{4\pi} \psi(\boldsymbol{x}, \boldsymbol{\Omega}', E') \, \mathrm{d}\Omega' \, \mathrm{d}E' \ , \tag{2.19c}$$

$$S \equiv \int_0^\infty \int_{4\pi} \Sigma_s(\boldsymbol{x}, \boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}, E' \to E) \psi(\boldsymbol{x}, \boldsymbol{\Omega}', E') \, \mathrm{d}\Omega' \, \mathrm{d}E' \ . \tag{2.19d}$$

In this approach, an initial guess is made for the fundamental eigenpair and the fission source is calculated. The scattering source $S^{(l)}$ is computed for a given power iteration l using SI as described in Algorithm 1. Then, an approximate solution for the next iteration is calculated as shown below, where l is the power iteration index:

$$\psi^{(l+1)} = \left(L - S^{(l)}\right)^{-1} F^{(l)} \ . \tag{2.20a}$$

Here, the iteration-dependent fission source, computed using the previous scalar flux iterate $\phi_0^{(l)}$, is defined as

$$F^{(l)} \equiv \frac{1}{k_{\text{eff}}^{(l)}} \frac{\chi(\boldsymbol{x}, E)}{4\pi} \int_0^\infty \nu\Sigma_f(\boldsymbol{x}, E') \phi_0^{(l)}(\boldsymbol{x}, E') \, \mathrm{d}E' \ , \tag{2.20b}$$

and the scalar flux iterate is defined as

$$\phi_0^{(l)} \equiv \int_{4\pi} \psi^{(l)}(\boldsymbol{x}, \boldsymbol{\Omega}, E) \, \mathrm{d}\Omega \ . \tag{2.20c}$$

Additionally, k_{eff} for iteration l, which represents the multiplication of neutrons over successive iterations, can be computed as shown below [2]:

$$k_{\text{eff}}^{(l+1)} = \frac{\int_V \int_0^\infty \nu\Sigma_f(\boldsymbol{x}, E') \int_{4\pi} \psi^{(l+1)}(\boldsymbol{x}, \boldsymbol{\Omega}, E)\, \mathrm{d}\Omega\, \mathrm{d}E\, \mathrm{d}V}{\frac{1}{k_{\text{eff}}^{(l)}} \int_V \int_0^\infty \nu\Sigma_f(\boldsymbol{x}, E') \int_{4\pi} \psi^{(l)}(\boldsymbol{x}, \boldsymbol{\Omega}, E)\, \mathrm{d}\Omega\, \mathrm{d}E\, \mathrm{d}V} \,. \tag{2.21}$$

If the quantities of interest are not converged (typically ϕ_0 and k_{eff}), the next iteration is started with the new estimates for ϕ_0 and k_{eff}. Algorithm 2 describes the iterative process of PI for eigenvalue problems.

Algorithm 2 Power Iteration Algorithm for Eigenvalue Problems

```
Assume an initial guess for $\phi_0^{(0)}$ and $k_{\text{eff}}^{(0)}$
repeat
    Compute the scattering source $S^{(l)}$ using Algorithm 1
    Compute the fission source $F^{(l)}$ using Eq. (2.20b)
    Compute the next flux estimate $\psi^{(l+1)}$ using Eq. (2.20a)
    Compute the scalar flux $\phi_0^{(l+1)}$ using Eq. (2.20c)
    Compute $k_{\text{eff}}^{(l+1)}$ using Eq. (2.21)
    Normalize the flux according to the chosen normalization condition
until convergence        {generally, convergence of $\phi_0$ and $k_{\text{eff}}$ are checked}
```

PI can converge very slowly, and acceleration methods have been developed to improve the convergence rate of this process. For PI, the convergence rate behaves as if the maximum non-absorption rate is unity. Thus, it can require hundreds or even thousands of iterations to converge to a reasonable tolerance for common reactor physics cases. This motivates the development and usage of acceleration methods to improve the convergence rate, which are described in Section 2.4.2.

2.4.2 Diffusion Acceleration Algorithms

For iterative methods of solving the NTE, diffusion-based acceleration techniques generally involve exact, "low-order" representations of the governing equations. This low-order problem is obtained by first integrating the NTE over angle, which eliminates all but the isotropic differential scattering moment (resulting in the NBE). To obtain the NBE for eigenvalue problems, Eq. (2.4a) is integrated over angle by operating by $\int_{4\pi} (\cdot)\, \mathrm{d}\Omega$, resulting in the following:

$$\begin{aligned}\nabla \cdot \boldsymbol{\phi}_1(\boldsymbol{x}, E) + \Sigma_t(\boldsymbol{x}, E)\phi_0(\boldsymbol{x}, E) = \int_0^\infty \Sigma_{s0}(\boldsymbol{x}, E' \to E)\phi_0(\boldsymbol{x}, E')\,\mathrm{d}E' \\ + \lambda\chi(\boldsymbol{x}, E)\int_0^\infty \nu\Sigma_f(\boldsymbol{x}, E')\phi_0(\boldsymbol{x}, E')\,\mathrm{d}E' \ .\end{aligned} \tag{2.22}$$

Here, $\Sigma_{s0}(\boldsymbol{x}, E' \to E)$ is the zeroth Legendre moment of the differential scattering cross section. Equation (2.22) requires a relationship between the scalar flux and neutron current to obtain the scalar flux solution. Certain approximations, such as *Fick's Law*, can provide this relationship. After introducing Fick's Law, the equation can be made exact through a transport consistency term (which is, in practice, lagged in the iteration process). This allows us to write the NBE in terms of only one unknown (the scalar flux ϕ_0, as opposed to both the scalar flux and neutron current).

If the angular flux is nearly *linearly-anisotropic*, meaning that it is a linear function of angle, then Fick's Law describes the approximate relationship between the neutron current and scalar flux:

$$\boldsymbol{\phi}_1(\boldsymbol{x}, E) = -D(\boldsymbol{x}, E)\nabla\phi_0(\boldsymbol{x}, E) \ , \tag{2.23a}$$

$$D(\boldsymbol{x}, E) \equiv \frac{1}{3\Sigma_{tr}(\boldsymbol{x}, E)} \ , \tag{2.23b}$$

$$\Sigma_{tr}(\boldsymbol{x}, E) \equiv \Sigma_t(\boldsymbol{x}, E) - \Sigma_{s1}(\boldsymbol{x}, E) \ . \tag{2.23c}$$

In Eqs. (2.23), D is called the *diffusion coefficient*, Σ_{s1} is the first angular moment of the differential scattering cross section, and Σ_{tr} is the transport cross section. The assumption that the angular flux is linearly-anisotropic is generally inaccurate near strong absorbers or vacuum boundaries, where the gradient of the neutron scalar flux is not small. Figure 2.4 depicts the inaccuracy of the neutron scalar flux obtained through diffusion theory compared to the exact solution near a vacuum or air boundary [5].

Different acceleration methods utilize the NBE as the foundation of the low-order problem through various modifications. For CMFD (the focus of Chapter 3), the divergence of the neutron current in Eq. (2.22) is replaced by a combination of the Fick's Law approximation (described in Eq. (2.23a)) and a nonlinear transport correction term, as shown below:

$$\begin{aligned}-\nabla \cdot D(\boldsymbol{x}, E)\nabla\phi_0(\boldsymbol{x}, E) + \left[\Sigma_t(\boldsymbol{x}, E) + \hat{D}(\boldsymbol{x}, E)\right]\phi_0(\boldsymbol{x}, E) \\ = \int_0^\infty \Sigma_{s0}(\boldsymbol{x}, E' \to E)\phi_0(\boldsymbol{x}, E)\,\mathrm{d}E \\ + \lambda\chi(\boldsymbol{x}, E)\int_0^\infty \nu\Sigma_f(\boldsymbol{x}, E')\phi_0(\boldsymbol{x}, E')\,\mathrm{d}E' \ ,\end{aligned} \tag{2.24a}$$

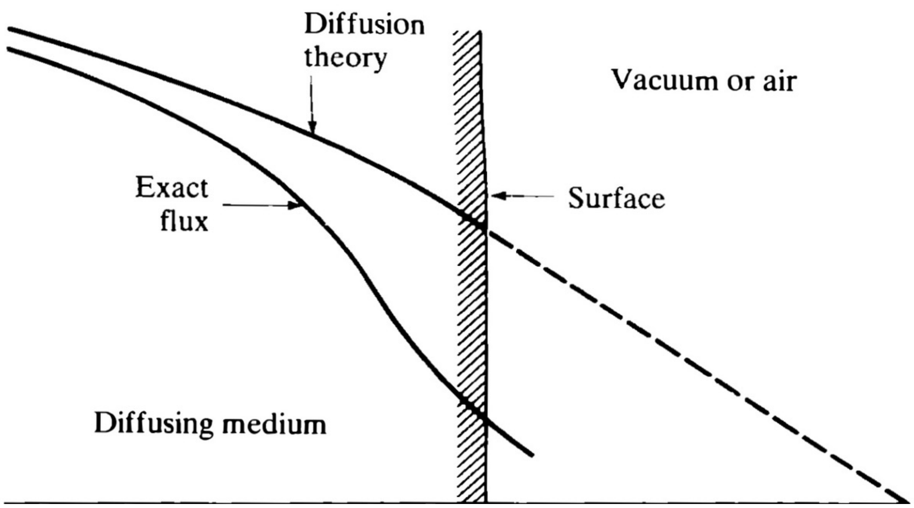


Figure 2.4: Comparison of diffusion and exact neutron scalar flux [5]

where we define the *transport correction term* $\hat{D}$ as

$$\hat{D}(\boldsymbol{x}, E) \equiv \frac{\nabla \cdot [\boldsymbol{\phi}_1(\boldsymbol{x}, E) + D(\boldsymbol{x}, E)\nabla\phi_0(\boldsymbol{x}, E)]}{\phi_0(\boldsymbol{x}, E)} , \tag{2.24b}$$

and the diffusion coefficient D is the same as in Eq. (2.23b). Eqs. (2.24) is an algebraically equivalent form of Eq. (2.22), and if the term $\hat{D}$ is known, then Eq. (2.24a) is a diffusion equation with the same scalar flux solution as the transport equation. $\hat{D}$ is usually small since it is a correction to Fick's Law, so it is generally "lagged" in the iteration scheme – meaning that it is estimated from the most recent transport sweep. We note that Eq. (2.24a) reduces to the traditional neutron diffusion equation if $\hat{D}$ is set to zero (the solution of which is generally different, and less accurate, than the transport equation). The presence of $\hat{D}$ and other nonlinear terms render CMFD-accelerated iteration schemes susceptible to instability as a result of approximations such as 2D/1D or TCP$_0$ (which are discussed in more detail in Sections 2.3.2.4 and 2.3.2.5, respectively) that allow for non-positive solutions. Section 3.5 provides a more detailed discussion of the nonlinearity of CMFD.

The low-order CMFD problem is obtained after discretizing Eqs. (2.24) in a specific way. By carefully performing these steps, the high-order (transport) and low-order (acceleration) equations are kept consistent. Then, a linear system of equations that describes the low-order problem is constructed, which can be solved directly or iteratively using various methods as discussed in Section 2.4.3. The solution to the low-order problem can be used to update the transport solution in a way that improves the rate of convergence without modifying the solution.

By including a low-order "solve" in the iterative algorithm for solving the NTE and applying information from its solution to the high-order solution, the general "accelerated" transport algorithm for eigenvalue problems is defined. When an acceleration method is used, the terminology used to describe the iterative process changes. Rather than referring to the outermost iterations as power iterations, they are now referred to as *outer iterations*. Within a single outer iteration, SI is still used to converge the scattering source (as discussed in Section 2.4.1.1). Then, PI is used to solve the low-order problem and obtain a low-order form of the fundamental eigenpair. The details of the low-order PI process for obtaining the fundamental eigenpair, as well as how the solution is used, are covered in Chapters 3 and 4.

Algorithm 3 Diffusion-Accelerated Algorithm for Eigenvalue Problems

```
Assume an initial guess for $\phi_0^{(0)}$ and $k_{\text{eff}}^{(0)}$
repeat
    Compute the scattering source and intermediate flux solution using SI as described in Algorithm 1
    Calculate $\hat{D}$ [Eq. (2.24b)]
    Compute the next scalar flux estimate $\phi_0^{(l+1)}$ and $k_{\text{eff}}^{(l+1)}$ by solving the low-order problem [Eq. (2.24a)] using PI
until convergence          {generally, convergence of $\phi_0$ and $k_{\text{eff}}$ are checked}
```

Acceleration methods such as DSA and CMFD have been shown to significantly reduce the number of iterations required for convergence of the scalar flux iterates [7, 9]. This corresponds with a reduction in the spectral radius. Specifically, for monoenergetic infinite-medium 1D fixed-source problems with no discretization, these methods bound the spectral radius to ~ $0.2247c$ where c is the scattering ratio of the medium. For eigenvalue problems, the spectral radius of CMFD behaves as it does for fixed-source problems with $c = 1$, i.e. with a spectral radius of ~ 0.2247. Another way to understand this is that each accelerated outer iteration reduces the error in the scalar flux by a factor of approximately $\frac{1}{5}$.

2.4.3 Linear Solvers for the Low-Order Acceleration Equations

Generally, the low-order acceleration equations are solved by constructing and obtaining the solution to a linear system of the form shown in Eq. (2.25), where A is the $n \times n$ discrete diffusion operator, $\boldsymbol{q}$ is the $n \times 1$ source vector, and $\boldsymbol{f}$ is the $n \times 1$ solution vector (which may be the next scalar flux iterate or a related quantity):

$$A\boldsymbol{f} = \boldsymbol{q} \,. \tag{2.25}$$

(For the high-order problem previously considered, A would theoretically be $10^{12} \times 10^{12}$ in size.) Because of the structure and size of A, which is generally a block matrix, the operator is typically stored as a *sparse* matrix, in which only the non-zero elements and their locations are stored. This system can be solved directly or iteratively, as discussed in Section 2.4.3.1 and Section 2.4.3.2, respectively.

The low-order acceleration problem is generally much smaller than the transport problem, and is, thus, computationally cheaper to solve. To demonstrate this, let us consider the same MPACT problem mentioned at the beginning of Section 2.4. For the low-order problem, homogenization is usually used to reduce the number of spatial unknowns. Additionally, integration over angle eliminates angular unknowns entirely. Therefore, the following estimates for the the number of unknowns can be made: $O(10^4)$ coarse (pin) cells per plane, $O(10^2)$ axial cells, and $O(10^2)$ energy groups. This brings the total number of unknowns to $O(10^8)$ which is *four orders of magnitude* less than in the previously-considered, high-order transport problem. By performing a group-collapse to $O(10)$ groups, this can be further reduced to $O(10^7)$ unknowns. Therefore, the expense of solving Eq. (2.25) is generally much less than performing a transport sweep.

2.4.3.1 Direct Methods

The most naïve method of solving for $\boldsymbol{f}$ in Eq. (2.25) is by computing the inverse of A and multiplying by it:

$$\boldsymbol{f} = A^{-1}\boldsymbol{q} \, . \tag{2.26}$$

However, computing A^{-1} in practice can be computationally expensive. Combining this with the fact that A may change for each iteration of the accelerated algorithm, this method is generally avoided. One method of computing the matrix inverse is a modified form of Gaussian elimination called the Gauss-Jordan method [45]. This operation requires $O(n^3)$ operations and is, therefore, computationally impractical for large n (and nearly intractable for the high-order problem). If A is not fixed, then this inversion would need to be done *every outer iteration* (for either the low- or high-order problem).

If the low-order operator A is fixed for all source iterations (as it is for LDA), then an alternative direct method may be used that takes advantage of this characteristic. By factorizing A into a set of operators that allow for quick and efficient solutions, the overall computational complexity to solve for $\boldsymbol{f}$ is reduced compared to computing the inverse. One method is *Lower-Upper* (LU) decomposition, in which A is factorized into a set of lower- (L) and upper-triangular (U) matrices (as well as an optional permutation matrix P) such that $PA = LU$ and $LU\boldsymbol{f} = P\boldsymbol{q}$. This can be expensive for a large A, but does not need to be repeated if A is constant (the factorization step generally requires $O(n^3)$ operations and is, thus, more feasible for the low-order problem compared to the high-order problem). Additionally, if the factorization of the low-order operator must be done every outer iteration (in the case where A is not fixed, which is true for CMFD), then LU decomposition becomes less desirable. After the factorization, the solution to Eq. (2.25) is computed in two relatively efficient steps that are shown in Eqs. (2.27) which can be solved by forward and backward substitution (requiring $O(n^2)$ operations each):

$$L\boldsymbol{y} = P\boldsymbol{q} \,, \tag{2.27a}$$

$$U\boldsymbol{f} = \boldsymbol{y} \,. \tag{2.27b}$$

Thus, directly solving the low-order problem in this way is theoretically less expensive for LDA compared to CMFD.

A drawback of the LU decomposition method is that the sparsity of the factors is not the same as the sparsity of A. Thus, the storage requirement for this method can be impractical for large problems. However, the literature suggests that some degree of sparsity for the factors of A can be preserved [46, 47] in the LU decomposition. Additionally, advanced procedures exist for computing the LU decomposition that are more computationally efficient [48]. We note that the advantages of the LU decomposition approach are speculative, as it has not been compared with other solver methods for large, practical problems.

2.4.3.2 Iterative Methods

Rather than computing $\boldsymbol{f}$ directly, iterative methods can also be used in which the solution is obtained through an iterative algorithm. Often, iterative methods are used for very large and sparse linear systems where direct solutions are less computationally efficient. One such method is the *Generalized Minimal Residual* (GMRES) method, which approximates $\boldsymbol{f}$ by minimizing the residual in a Krylov subspace [49]. Other iterative methods exist, such as *Successive Over-Relaxation* (SOR) and *Red-Black Block Jacobi* (RBBJ), but we will focus on GMRES in this work because it is the default solver method in MPACT for the low-order problem. Iterative solver methods such as these can reduce the amount of computational work required to solve for $\boldsymbol{f}$, but require multiple iterations (which we refer to as *inner* iterations) to converge the solution within a specified tolerance. For a dense matrix, GMRES requires approximately $O(kn^2)$ operations, where k is the number of inner iterations. In the worst case, GMRES can require n inner iterations to arrive at a converged solution, resulting in overall $O(n^3)$ operations. The number of iterations required depends on the properties of A, and can be reduced through informed *preconditioning*. Because the amount of computational resources varies, the solver method of choice depends on the computational resources available and the problem complexity.

One property that can affect the convergence rate of GMRES is the *condition number* of the operator A. This metric describes how sensitive the solution vector $\boldsymbol{f}$ is to a perturbation in the source vector $\boldsymbol{q}$. If the condition number becomes too large, the solution can become inaccurate due to numerical roundoff. The condition number of a matrix is defined as

$$\kappa_p(A) \equiv \left\|A\right\|_p \left\|A^{-1}\right\|_p \,, \tag{2.28}$$

where $\left\|(\cdot)\right\|_p$ is the operator p-norm. The magnitude of the condition number depends on the choice of p. In many cases, the operator 2-norm is used. Generally, the condition number of the low-order diffusion operator increases with the number of mesh cells in the problem [50].

As the operator A becomes more singular and $\kappa_p(A) \to \infty$, the convergence rate of most iterative solvers suffer. In these cases, the matrix is considered to be *ill-conditioned*. Though a high condition number is more of a concern for iterative methods, the accuracy of direct methods can also degrade when the condition number is extremely large. Therefore, one must take care when handling ill-conditioned problems. We discuss how the fixed nature of the LDA operator may offer advantages for iterative linear solvers in Section 6.5.

2.4.3.3 Practical Application to LDA and CMFD

The methods discussed in this section are available in high performance computing packages that are accessible to code developers. Due to the size of transport problems, they are not typically used for the high-order transport calculation. However, the low-order problem generally contains several orders of magnitude fewer unknowns, depending on the choice of the low-order grid. As a consequence, the computational cost of the low-order step is much lower than the high-order transport solve. Thus, these linear solver methods are applied to the low-order problem in practice.

Though the linear systems of both CMFD and LDA can be solved directly or iteratively, one difference provides an advantage for LDA. The diffusion operator of LDA is fixed between outer iterations, while the CMFD operator changes between outer iterations. This is because the terms that maintain consistency between the high- and low-order solutions are located in the diffusion operator (referred to in this section as A) for CMFD, but reside in the source vector (referred to in this section as $\boldsymbol{q}$) for LDA. One may be able to take advantage of this characteristic to reduce the computational cost of solving the LDA system (as discussed here and in Section 6.5). However, these advantages are still speculative in nature and may not manifest in practice.

CHAPTER 3

Coarse Mesh Finite Difference

This chapter provides a detailed description of the *Coarse Mesh Finite Difference* (CMFD) acceleration method. This includes: (i) background information about CMFD in Section 3.1, (ii) a derivation of CMFD for fixed-source and eigenvalue problems in Section 3.2, (iii) algorithm overviews for CMFD-accelerated fixed-source and eigenvalue problems in Section 3.3, (iv) information on variations of *Power Iteration* (PI) that can be used to solve the CMFD equations for eigenvalue problems in Section 3.4, and (v) details of the nonlinearities in CMFD in Section 3.5.

3.1 Background

CMFD is a nonlinear acceleration method that can be used to improve the convergence rate of iteration schemes for linear neutron transport problems. The method was first developed to reduce the storage of nodal diffusion problems in the early 1980s [8]. Once iterative transport schemes became a popular method to simulate nuclear systems, CMFD was adapted for this purpose [9]. The popularity of CMFD can be attributed to the following properties: (i) superior performance compared to some previous acceleration methods (like *Coarse Mesh Rebalance* (CMR)), (ii) ease of application to eigenvalue problems (which other methods with similar performance, such as DSA, lacked [21]), and (iii) focus on a "coarse" mesh for the low-order problem that reduces the computational cost of the acceleration step. We refer to the mesh employed in the acceleration step as the "coarse" mesh or "low-order" mesh interchangeably.

CMFD is closely related to *Nonlinear Diffusion Acceleration* (NDA), which is a similar nonlinear acceleration method that utilizes the same transport consistency term as that used for CMFD. This consistency term, which will be derived later in this chapter, is usually denoted as $\hat{D}$. It ensures that the solution of the high- and low-order problems are consistent. In effect, $\hat{D}$ corrects the approximation made by introducing Fick's Law into the low-order equations as discussed in Section 3.2. CMFD and NDA are identical if the low-order spatial mesh is the same as that used for the high-order transport problem. However, the low-order spatial mesh is usually chosen to be coarser than the high-order spatial mesh, in an effort to reduce the computational cost of solving the low-order problem. Therefore, CMFD is more common in reactor physics codes than NDA. In MPACT, the coarse mesh exists on the pin cell grid. The spatial mesh used for the CMFD problem is usually regular, but has been extended to unstructured spatial meshes [51].

In contrast with other acceleration methods such as DSA, CMFD is well-suited for eigenvalue problems. Nonlinear terms are introduced to set up a traditional eigenvalue problem for the CMFD linear system. Specifically, the low-order differential equation takes the form of a traditional eigenvalue problem as shown below:

$$\mathbb{M}\boldsymbol{f} = \lambda\mathbb{F}\boldsymbol{f} \ , \tag{3.1}$$

where $\mathbb{M}$ is the low-order *migration* operator, $\mathbb{F}$ is the *fission* operator, $\boldsymbol{f}$ is a given eigenvector, and λ is the corresponding eigenvalue. Then, a method such as *Power Iteration* (PI) is used to iteratively solve Eq. (3.1) to a suitable tolerance. DSA can be modified for application to eigenvalue problems, but is made nonlinear in the process [21].

Like other similar acceleration methods, CMFD possesses a *linear* instability in which the performance of the method degrades with increasing coarse cell optical thickness. This degradation continues to the point of divergence (when the coarse grid optical thickness is equal to approximately two mean free paths) and the iterative process must be restarted with a different choice of low-order grid. Over the last few decades, the initial formulation of CMFD has been modified to suppress the linear instability [25, 26, 27].

In addition to the aforementioned linear instability, CMFD also possesses a less-common and less-studied *nonlinear* instability. This nonlinear instability is disparate from the linear instability, and is a consequence of the presence of terms that are nonlinear with respect to the scalar flux in the low-order system of equations. Nonlinear instability shows up in cases where the strict positivity of the flux iterates is not guaranteed. If the flux can become near-zero or negative in a portion of phase space for any iteration, then nonlinear instability may affect the iterative process. Nonlinear instability is much less systematic and predictable than the linear instability, and can manifest in a reactor physics code as degradation in the performance of the acceleration method up to the point of divergence. The signs of nonlinear instability are convergence issues (reduced performance or divergence) in a case that would not otherwise exhibit the linear instability. In cases affected by nonlinear instability, normally small nonlinear terms (such as $\hat{D}$) may become unreasonably large and numerically unstable. Certain techniques have been employed to mitigate this issue, such as negative flux fix-ups, that retain iterative stability at the cost of solution accuracy [32, 33]. Additionally, a limitation can be placed on the magnitude of $\hat{D}$ that may affect the iterative properties. The goal of this thesis is to present a viable alternative acceleration method to CMFD called *Linear Diffusion Acceleration* (LDA) (discussed in Chapter 4) that avoids these issues without the need to resort to fix-up techniques.

3.2 Derivation of CMFD

In this section, the low-order CMFD equation is derived from the steady-state, monoenergetic, 1D NTE for both fixed-source and eigenvalue problems. Each problem type is treated separately. As an overview, the following steps are performed: (i) integration of the NTE over angle to obtain the *Neutron Balance Equation* (NBE), (ii) substitution of the neutron current with a combination of Fick's Law and a transport correction term, and (iii) integration over a given coarse cell. The resulting set of equations is solved differently depending on whether the problem type is fixed-source or eigenvalue. For a fixed-source problem, the system is solved once every outer iteration using one of the methods discussed in Section 2.4.3. If CMFD is applied to an eigenvalue problem, PI is used to iteratively solve for the fundamental eigenpair. In this situation, the system of equations must be solved multiple times to converge the low-order problem for a given outer iteration.

3.2.1 Fixed-Source Problems

We begin with the steady-state, monoenergetic, 1D NTE for a fixed-source problem shown below, with prescribed incident flux boundary conditions:

$$\mu\frac{\mathrm{d}}{\mathrm{d}x}\psi(x,\mu) + \Sigma_t(x)\psi(x,\mu) = \frac{1}{2}\left(\int_{-1}^{1}\Sigma_s(x,\mu,\mu')\psi(x,\mu')\,\mathrm{d}\mu' + Q(x)\right) , \quad 0 \le x \le X , \tag{3.2a}$$

$$\psi(0,\mu) = \psi^b(\mu) , \quad \mu > 0 , \tag{3.2b}$$

$$\psi(X,\mu) = \psi^b(\mu) , \quad \mu < 0 , \tag{3.2c}$$

where X is the width of the spatial domain and $\psi^b(\mu)$ is the boundary source function. Next, we average this equation over a given spatial cell j by operating by $\frac{1}{h_j}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}}(\cdot)$, where h_j is the width of cell j and $x_{j\pm\frac{1}{2}}$ are the edges of cell j. Additionally, we apply the discrete ordinates approximation in which angular integrals are approximated by the following sum:

$$\int_{-1}^{1} f(\mu)\,\mathrm{d}\mu \approx \sum_{n=1}^{N} w_n f(\mu_n) \ , \tag{3.3a}$$

where

$$f(\mu_n) \equiv f_n \ , \tag{3.3b}$$

and

$$\sum_{n=1}^{N} w_n = 2 \ . \tag{3.3c}$$

The integer N is the total number of discrete angles, μ_n is the cosine of a given discrete polar angle n, and w_n is the weight (or angular bin size) for a given angle index n. If we assume that scattering is isotropic, these steps yield the following discretized form of Eqs. (3.2):

$$\frac{\mu_n}{h_j}\left(\psi_{n,j+\frac{1}{2}} - \psi_{n,j-\frac{1}{2}}\right) + \Sigma_{t,j}\psi_{n,j} = \frac{1}{2}\left(\Sigma_{s,j}\phi_j + q_j\right) \ , \tag{3.4a}$$

$$\phi_j \equiv \sum_{n=1}^{N} w_n \psi_{n,j} = \text{cell-average scalar flux} \ , \tag{3.4b}$$

$$\psi_{n,j} \equiv \frac{1}{h_j}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \psi_n(x)\,\mathrm{d}x = \text{cell-average angular flux} \ , \tag{3.4c}$$

$$\psi_{n,j\pm\frac{1}{2}} \equiv \psi_n(x_{j\pm\frac{1}{2}}) = \text{cell-edge angular flux} \ , \tag{3.4d}$$

$$\{j \mid j \in \mathbb{N}, 1 \le j \le J\} \ ,$$

$$\psi_{n,\frac{1}{2}} = \psi_n^b \ , \quad \mu_n > 0 \ , \tag{3.4e}$$

$$\psi_{n,J+\frac{1}{2}} = \psi_n^b \ , \quad \mu_n < 0 \ , \tag{3.4f}$$

$$\{n \mid n \in \mathbb{N}, 1 \le n \le N\} \ .$$

In these equations, cross sections are assumed to be constant in a given spatial bin. Additionally, the set $\mathbb{N}$ refers to the set of natural integers. To form a complete system of equations for $\psi_{n,j\pm\frac{1}{2}} \ \forall \ j \ \forall \ n$, a spatial closure relationship must be assumed that relates the cell-edge $\left(\psi_{n,j\pm\frac{1}{2}}\right)$ and cell-center $(\psi_{n,j})$ angular flux quantities. We assume the diamond-difference relationship shown below (we note, though, that other relationships are possible):

$$\psi_{n,j} = \frac{1}{2}\left(\psi_{n,j+\frac{1}{2}} + \psi_{n,j-\frac{1}{2}}\right) \ . \tag{3.4g}$$

With this, we have a total of $N(2J+1)$ equations to compute the cell-edge angular flux. Next, we integrate over angle by operating by $\sum_{n=1}^{N} w_n\,(\cdot)$ to obtain the discrete NBE shown below,

$$\frac{1}{h_j}\left(\phi_{1,j+\frac{1}{2}} - \phi_{1,j-\frac{1}{2}}\right) + \Sigma_{a,j}\phi_{0,j} + q_j \ , \tag{3.5a}$$

where the neutron current $\phi_{1,j\pm\frac{1}{2}}$ at the edge of a given fine cell j is defined as

$$\phi_{1,j\pm\frac{1}{2}} \equiv \sum_{n=1}^{N} \mu_n w_n \psi_{n,j\pm\frac{1}{2}} \ . \tag{3.5b}$$

We next define the "coarse" mesh, in which each coarse cell k contains a contiguous set of fine cells referred to as $j \in k$. There are a total of K coarse cells, and the total number of coarse cells is less than or equal to the total number of fine cells J $(K \le J)$. If $K = J$, then the coarse mesh is identical to the fine mesh. We define the following relationship between the fine and coarse mesh indices:

$$X_{k+\frac{1}{2}} \equiv x_{p_k+\frac{1}{2}} \ , \tag{3.6}$$

where p_k is the index of the rightmost fine cell in coarse cell k. An example of a fine and coarse grid is shown in Fig. 3.1, in which each coarse cell contains 3 fine cells. In reality, each given coarse cell can contain an arbitrary number of fine cells.

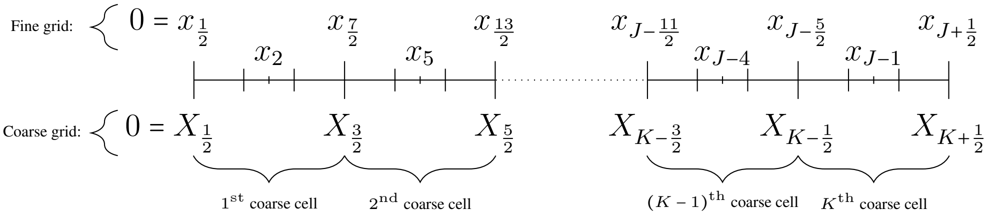


Figure 3.1: Example fine and coarse spatial grids. There are 3 fine cells per coarse cell, J fine cells, and K coarse cells in the problem. The center of the jth fine cell is located at x_j, with the left and right edges located at $x_{j\pm\frac{1}{2}}$. Similarly, the center of the coarse cell k is located at X_k, with left and right edges located at $X_{k\pm\frac{1}{2}}$.

After operating by $\sum_{j\in k}(\cdot)\,h_j$ on Eq. (3.5a), we obtain the following discretized NBE on the coarse mesh:

$$\left(\Phi_{1,k+\frac{1}{2}} - \Phi_{1,k-\frac{1}{2}}\right) + \bar{\Sigma}_{a,k}\Phi_{0,k}\Delta_k = \bar{Q}_k\Delta_k \tag{3.7a}$$

where we define the flux- and volume-weighted average absorption cross section in a coarse cell as

$$\bar{\Sigma}_{a,k} \equiv \frac{\sum\limits_{j\in k} \Sigma_{a,j}\phi_{0,j}h_j}{\sum\limits_{j\in k} \phi_{0,j}h_j} \ , \tag{3.7b}$$

the volume-average coarse-cell scalar flux $\Phi_{0,k}$ as

$$\Phi_{0,k} \equiv \frac{\sum_{j\in k} \phi_{0,j} h_j}{\Delta_k} , \tag{3.7c}$$

the coarse-cell width Δ as

$$\Delta_k \equiv \sum_{j\in k} h_j , \tag{3.7d}$$

the volume-average source $\bar{Q}_k$ as

$$\bar{Q}_k \equiv \frac{\sum_{j\in k} q_j h_j}{\Delta_k} , \tag{3.7e}$$

and the neutron current at a coarse-cell edge $\Phi_{1,k+\frac{1}{2}}$ as

$$\Phi_{1,k+\frac{1}{2}} \equiv \sum_{n=1}^{N} \mu_n w_n \psi_{n,p_k+\frac{1}{2}} . \tag{3.7f}$$

To obtain the CMFD equation, we create the following definition for the coarse cell-edge neutron current:

$$\Phi_{1,k+\frac{1}{2}} \equiv \underbrace{-\tilde{D}_{k+\frac{1}{2}} \left(\Phi_{0,k+1} - \Phi_{0,k}\right)}_{\text{Fick's Law}} + \underbrace{\hat{D}_{k+\frac{1}{2}} \left(\Phi_{0,k+1} + \Phi_{0,k}\right)}_{\text{transport correction term}} , \tag{3.7g}$$

where $\tilde{D}$ is the numerical diffusion coefficient, defined as

$$\tilde{D}_{k+\frac{1}{2}} \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k+1}\Delta_{k+1} + \bar{\Sigma}_{t,k}\Delta_k\right)} , \tag{3.7h}$$

and $\hat{D}$ is the *transport correction* term, defined as

$$\hat{D}_{k+\frac{1}{2}} \equiv \frac{\Phi_{1,k+\frac{1}{2}} + \tilde{D}_{k+\frac{1}{2}} \left(\Phi_{0,k+1} - \Phi_{0,k}\right)}{\Phi_{0,k+1} + \Phi_{0,k}} . \tag{3.7i}$$

In Eq. (3.7g), the first term is the discretized form of Fick's Law and the second term is a "transport correction" to Fick's Law. This term accounts for the deficiency in Fick's Law to treat neutron streaming and should be small if Fick's Law is reasonably accurate. To obtain Eq. (3.7i), we simply rearrange Eq. (3.7g) for $\hat{D}$. We note that different choices can be made for Eq. (3.7g), resulting in different definitions of $\hat{D}$ [52].

Next, we introduce iteration superscripts, where the index l denotes the outer iteration number and l exists in the set

$$\{l \mid l \in \mathbb{N}, 0 \le l < L\} \ , \tag{3.8a}$$

where L is the required number of outer iterations performed to achieve a desired convergence criterion. Fractional indices are used to denote quantities that are computed within a given outer iteration, and larger indices indicate quantities that are computed later in the iteration process. Because we assume $\hat{D}$ to be small, we choose to lag this term in the iteration process. This amounts to computing $\hat{D}$ from known quantities. If the transport sweep is done before the acceleration step, then $\hat{D}$ is computed from quantities computed in the transport sweep. In this case, we define the transport equation with iteration superscripts as

$$\frac{\mu_n}{h_j}\left(\psi_{n,j+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} - \psi_{n,j-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\right) + \Sigma_{t,j}\psi_{n,j}^{\left(l+\frac{1}{2}\right)} = \frac{1}{2}\left(\Sigma_{s,j}\phi_{0,j}^{(l)} + q_j\right) \ , \tag{3.8b}$$

$$\psi_{n,j}^{\left(l+\frac{1}{2}\right)} = \frac{1}{2}\left(\psi_{n,j+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} + \psi_{n,j-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\right) \ , \tag{3.8c}$$

where the superscript $\left(l+\frac{1}{2}\right)$ denotes the solution to the transport equation in Eq. (3.8b). Next, we place iteration superscripts on the CMFD equations. As previously discussed, we lag the transport consistency term when solving the low-order problem:

$$\begin{gathered}\left(\Phi_{1,k+\frac{1}{2}}^{(l+1)} - \Phi_{1,k-\frac{1}{2}}^{(l+1)}\right) + \bar{\Sigma}_{a,k}^{\left(l+\frac{1}{2}\right)}\Phi_{0,k}^{(l+1)}\Delta_k = \bar{Q}_k\Delta_k \ , \\ \{k \mid k \in \mathbb{N}, 1 \le k \le K\} \ ,\end{gathered} \tag{3.8d}$$

where in Eq. (3.8d) we have

$$\begin{gathered}\Phi_{1,k+\frac{1}{2}}^{(l+1)} \equiv -\tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\left(\Phi_{0,k+1}^{(l+1)} - \Phi_{0,k}^{(l+1)}\right) + \hat{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\left(\Phi_{0,k+1}^{(l+1)} + \Phi_{0,k}^{(l+1)}\right) \ , \\ \{k \mid k \in \mathbb{N}, 1 \le k < K\} \ ,\end{gathered} \tag{3.8e}$$

at the interior coarse-cell edges, with

$$\tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k+1}^{\left(l+\frac{1}{2}\right)}\Delta_{k+1} + \bar{\Sigma}_{t,k}^{\left(l+\frac{1}{2}\right)}\Delta_k\right)} \ , \tag{3.8f}$$

$$\hat{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \frac{\Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} + \tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\left(\Phi_{0,k+1}^{\left(l+\frac{1}{2}\right)} - \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}\right)}{\Phi_{0,k+1}^{\left(l+\frac{1}{2}\right)} + \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}} \ , \tag{3.8g}$$

and

$$\bar{\Sigma}_{u,k}^{(l+\frac{1}{2})} \equiv \frac{\sum\limits_{j\in k} \Sigma_{u,j}\phi_{0,j}^{(l+\frac{1}{2})} h_j}{\sum\limits_{j\in k} \phi_{0,j}^{(l+\frac{1}{2})} h_j} \ , \forall \ u \in \{a, s, t\} \ . \tag{3.8h}$$

Here the superscript $(l+1)$ denotes the solution to the low-order problem. We must also derive the boundary conditions for the low-order problem from the prescribed incident flux boundary conditions stated in Eqs. (3.4). This is done for the left boundary as shown below:

$$2\Phi_{1,\frac{1}{2}}^{+} = 2\sum_{\mu_n>0} \mu_n w_n \psi_n^b \tag{3.8i}$$

$$= \sum_{n=1}^{N} \left(\mu_n + |\mu_n|\right) w_n \psi_{n,\frac{1}{2}} \tag{3.8j}$$

$$= \Phi_{1,\frac{1}{2}} + \left(\frac{\sum\limits_{n=1}^{N} |\mu_n| w_n \psi_{n,\frac{1}{2}}}{\Phi_{0,1}} \right) \Phi_{0,1} \ . \tag{3.8k}$$

Now, we choose to lag the nonlinear term in the following way:

$$2\Phi_{1,\frac{1}{2}}^{+} = \Phi_{1,\frac{1}{2}}^{(l+1)} + B_{\frac{1}{2}}^{(l+\frac{1}{2})} \Phi_{0,1}^{(l+1)} \ , \tag{3.8l}$$

where the term $B_{\frac{1}{2}}^{(l+\frac{1}{2})}$ is defined as

$$B_{\frac{1}{2}}^{(l+\frac{1}{2})} \equiv \frac{\sum\limits_{n=1}^{N} |\mu_n| w_n \psi_{n,\frac{1}{2}}^{(l+\frac{1}{2})}}{\Phi_{0,1}^{(l+\frac{1}{2})}} \ . \tag{3.8m}$$

Following a similar procedure for the right side of the boundary yields:

$$2\Phi_{1,K+\frac{1}{2}}^{-} = -\Phi_{1,K+\frac{1}{2}}^{(l+1)} + B_{K+\frac{1}{2}}^{(l+\frac{1}{2})} \Phi_{0,K}^{(l+1)} \ , \tag{3.8n}$$

where we define the incoming current $\Phi_{1,K+\frac{1}{2}}^{-}$ as

$$\Phi_{1,K+\frac{1}{2}}^{-} \equiv \sum_{\mu_n<0} \mu_n w_n \psi_n^b \ , \tag{3.8o}$$

and the lagged term $B^{(l+\frac{1}{2})}_{K+\frac{1}{2}}$ as

$$B^{(l+\frac{1}{2})}_{K+\frac{1}{2}} \equiv \frac{\sum\limits_{n=1}^{N} |\mu_n| w_n \psi^{(l+\frac{1}{2})}_{n,J+\frac{1}{2}}}{\Phi^{(l+\frac{1}{2})}_{0,K}} \ . \tag{3.8p}$$

Lastly, we seek to update the fine-mesh transport solution with information from the solution of the CMFD equations. This is done by scaling each fine-cell scalar flux value by the ratio of the low-order solution to the homogenized transport scalar flux, as shown below:

$$\phi^{(l+1)}_{0,j} = \phi^{(l+\frac{1}{2})}_{0,j} \left(\frac{\Phi^{(l+1)}_{0,k}}{\Phi^{(l+\frac{1}{2})}_{0,k}} \right) , \forall \ j \in k \ . \tag{3.8q}$$

With this, we have obtained all necessary equations for CMFD applied to fixed-source problems. The method described here is traditional CMFD, but various modifications can be made to Eqs. (3.8) [27]. In section Section 3.3.1, we carefully describe how the equations derived here are used when CMFD is applied to accelerate fixed-source transport problems.

3.2.2 Eigenvalue Problems

We begin with the steady-state, monoenergetic, 1D NTE for eigenvalue problems with vacuum boundary conditions shown below:

$$\begin{gathered} \mu \frac{\mathrm{d}}{\mathrm{d}x} \psi(x,\mu) + \Sigma_t(x)\psi(x,\mu) \\ = \frac{1}{2} \left(\int_{-1}^{1} \Sigma_s(x,\mu,\mu')\psi(x,\mu') \, \mathrm{d}\mu' + \lambda \nu \Sigma_f(x) \int_{-1}^{1} \psi(x,\mu') \, \mathrm{d}\mu' \right) \ , \\ 0 \le x \le X \ , \end{gathered} \tag{3.9a}$$

$$\psi(0,\mu) = 0 \ , \quad \mu > 0 \ , \tag{3.9b}$$

$$\psi(X,\mu) = 0 \ , \quad \mu < 0 \ , \tag{3.9c}$$

where the term $\nu\Sigma_f(x) \int_{-1}^{1} \psi(x,\mu') \, \mathrm{d}\mu'$ is the fission source and λ is the eigenvalue corresponding to the eigenvector $\psi(x,\mu)$. A normalization condition must be imposed to select a unique solution. As an example, we impose the following condition:

$$P = \int_0^X \int_{-1}^{1} \nu \Sigma_f(x) \psi(x,\mu) \, \mathrm{d}\mu \, \mathrm{d}x \ , \tag{3.9d}$$

in which P is the desired neutron production rate. To discretize Eqs. (3.9) in space and angle, we follow the same steps as those taken in Section 3.2.1. Namely, we average over a spatial cell and apply the discrete ordinates approximation. For the case of isotropic scattering the resulting discrete $\mathrm{S_N}$ transport equation is shown below, where we have applied the diamond-difference closure relation:

$$\frac{\mu_n}{h_j}\left(\psi_{n,j+\frac{1}{2}} - \psi_{n,j-\frac{1}{2}}\right) + \Sigma_t \psi_{n,j} = \frac{1}{2}\left(\Sigma_{s,j} + \lambda\left(\nu\Sigma_f\right)_j\right)\phi_{0,j}\ , \tag{3.10a}$$

$$\psi_{n,j} = \frac{1}{2}\left(\psi_{n,j+\frac{1}{2}} + \psi_{n,j-\frac{1}{2}}\right)\ , \tag{3.10b}$$

$$\phi_{0,j} \equiv \sum_{n=1}^{N} w_n \psi_{n,j}\ , \tag{3.10c}$$

$$\left\{j \mid j \in \mathbb{N}, 1 \le j \le J\right\}\ ,$$

$$\psi_{n,\frac{1}{2}} = 0\ , \quad \mu_n > 0\ , \tag{3.10d}$$

$$\psi_{n,J+\frac{1}{2}} = 0\ , \quad \mu_n < 0\ , \tag{3.10e}$$

$$\left\{n \mid n \in \mathbb{N}, 1 \le n \le N\right\}\ ,$$

$$P = \sum_{j=1}^{J}\left(\nu\Sigma_f\right)_j \phi_{0,j} h_j\ . \tag{3.10f}$$

As with the fixed-source case, we begin the derivation of the CMFD equations by first integrating over angle by operating on Eq. (3.10a) by $\sum_{n=1}^{N} w_n\left(\cdot\right)$, yielding

$$\frac{1}{h_j}\left(\phi_{1,j+\frac{1}{2}} - \phi_{1,j-\frac{1}{2}}\right) + \Sigma_{a,j}\phi_{0,j} = \lambda\left(\nu\Sigma_f\right)_j \phi_{0,j}\ . \tag{3.11a}$$

Next, we perform a spatial integration over a given coarse cell by operating by $\sum_{j\in k} h_j\left(\cdot\right)$, yielding

$$\left(\Phi_{1,k+\frac{1}{2}} - \Phi_{1,k-\frac{1}{2}}\right) + \bar{\Sigma}_{a,k}\Phi_{0,k}\Delta_k = \lambda\overline{\left(\nu\Sigma_f\right)}_k \Phi_{0,k}\Delta_k\ , \tag{3.11b}$$

where the flux- and volume-weighted *production cross section* is defined as

$$\overline{\left(\nu\Sigma_f\right)_k} \equiv \frac{\sum\limits_{j\in k}\left(\nu\Sigma_f\right)_j \phi_{0,j} h_j}{\sum\limits_{j\in k}\phi_{0,j} h_j}\ , \tag{3.11c}$$

and the width of a given coarse cell is defined as

$$\Delta_k \equiv \sum_{j \in k} h_j \ . \qquad \text{(Eq. (3.7d) revisited)}$$

The same substitution made for the coarse cell-edge neutron current that was done for the fixed-source case is done here:

$$\Phi_{1,k+\frac{1}{2}} \equiv \underbrace{-\tilde{D}_{k+\frac{1}{2}} \left(\Phi_{0,k+1} - \Phi_{0,k}\right)}_{\text{Fick's Law}} + \underbrace{\hat{D}_{k+\frac{1}{2}} \left(\Phi_{0,k+1} + \Phi_{0,k}\right)}_{\text{transport correction term}} \ , \qquad \text{(Eq. (3.7g) revisited)}$$

$$\tilde{D}_{k+\frac{1}{2}} \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k+1}\Delta_{k+1} + \bar{\Sigma}_{t,k}\Delta_k\right)} \ , \qquad \text{(Eq. (3.7h) revisited)}$$

$$\hat{D}_{k+\frac{1}{2}} \equiv \frac{\Phi_{1,k+\frac{1}{2}} + \tilde{D}_{k+\frac{1}{2}} \left(\Phi_{0,k+1} - \Phi_{0,k}\right)}{\Phi_{0,k+1} + \Phi_{0,k}} \ . \qquad \text{(Eq. (3.7i) revisited)}$$

Now, we introduce iteration superscripts where we “lag” the RHS source in the iteration scheme. In addition, we lag the transport correction term $\hat{D}$. The purpose of this is to be able to write the iteration scheme as a power iteration, which is similar to the source iteration in fixed-source problems. Upon placing these iteration superscripts, the following set of $\mathrm{S_N}$ equations is obtained:

$$\frac{\mu_n}{h_j}\left(\psi_{n,j+\frac{1}{2}}^{(l+\frac{1}{2})} - \psi_{n,j-\frac{1}{2}}^{(l+\frac{1}{2})}\right) + \Sigma_{t,j}\psi_{n,j}^{(l+\frac{1}{2})} = \frac{1}{2}\left(\Sigma_{s,j} + \lambda^{(l)}\left(\nu\Sigma_f\right)_j\right)\phi_{0,j}^{(l)} \ , \qquad (3.12a)$$

$$\psi_{n,j}^{(l+\frac{1}{2})} = \frac{1}{2}\left(\psi_{n,j+\frac{1}{2}}^{(l+\frac{1}{2})} + \psi_{n,j-\frac{1}{2}}^{(l+\frac{1}{2})}\right) \ , \qquad (3.12b)$$

and the following set of low-order CMFD equations:

$$\begin{gathered}\left(\Phi_{1,k+\frac{1}{2}}^{(l+1)} - \Phi_{1,k-\frac{1}{2}}^{(l+1)}\right) + \bar{\Sigma}_{a,k}^{(l+\frac{1}{2})}\Phi_{0,k}^{(l+1)}\Delta_k = \lambda^{(l)}\overline{\left(\nu\Sigma_f\right)}_k\Phi_{0,k}^{(l+\frac{1}{2})}\Delta_k \ , \\ \{k \mid k \in \mathbb{N}, 1 \leq k \leq K\} \ ,\end{gathered} \qquad (3.12c)$$

$$\begin{gathered}\Phi_{1,k+\frac{1}{2}}^{(l+1)} \equiv -\tilde{D}_{k+\frac{1}{2}}^{(l+\frac{1}{2})}\left(\Phi_{0,k+1}^{(l+1)} - \Phi_{0,k}^{(l+1)}\right) + \hat{D}_{k+\frac{1}{2}}^{(l+\frac{1}{2})}\left(\Phi_{0,k+1}^{(l+1)} + \Phi_{0,k}^{(l+1)}\right) \ , \\ \{k \mid k \in \mathbb{N}, 1 \leq k < K\} \ ,\end{gathered} \qquad \text{(Eq. (3.8e) revisited)}$$

$$\tilde{D}_{k+\frac{1}{2}}^{(l+\frac{1}{2})} \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k+1}^{(l+\frac{1}{2})}\Delta_{k+1} + \bar{\Sigma}_{t,k}^{(l+\frac{1}{2})}\Delta_k\right)} \ , \qquad \text{(Eq. (3.8f) revisited)}$$

$$\hat{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \frac{\Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} + \tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \left(\Phi_{0,k+1}^{\left(l+\frac{1}{2}\right)} - \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}\right)}{\Phi_{0,k+1}^{\left(l+\frac{1}{2}\right)} + \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}} \,. \qquad \text{(Eq. (3.8g) revisited)}$$

We note that Eq. (3.8e) holds only for the interior coarse-cell edges. To obtain the boundary conditions for the low-order problem, we follow the same steps as for the fixed-source case. However, there is a slight difference in this derivation as a result of the vacuum boundary conditions. For the left side of the problem, the derivation is as follows:

$$\begin{aligned} 0 &= \sum_{n=1}^{N} \left(\mu_n + |\mu_n|\right) w_n \psi_{n,\frac{1}{2}} && \text{(3.12d)} \\ &= \Phi_{1,\frac{1}{2}} + \left(\frac{\sum_{n=1}^{N} |\mu_n| w_n \psi_{n,\frac{1}{2}}}{\Phi_{0,1}}\right) \Phi_{0,1} \,. && \text{(3.12e)} \end{aligned}$$

Here, we lag the nonlinear term:

$$0 = \Phi_{1,\frac{1}{2}}^{(l+1)} + B_{\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \Phi_{0,1}^{(l+1)} \,, \tag{3.12f}$$

$$B_{\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \frac{\sum_{n=1}^{N} |\mu_n| w_n \psi_{n,\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}}{\Phi_{0,1}^{\left(l+\frac{1}{2}\right)}} \,. \tag{3.12g}$$

Similarly, for the right side we obtain:

$$0 = -\Phi_{1,K+\frac{1}{2}}^{(l+1)} + B_{K+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \Phi_{0,K}^{(l+1)} \,, \tag{3.12h}$$

$$B_{K+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \frac{\sum_{n=1}^{N} |\mu_n| w_n \psi_{n,J+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}}{\Phi_{0,K}^{\left(l+\frac{1}{2}\right)}} \,. \tag{3.12i}$$

We recognize that the low-order CMFD problem listed in Eq. (3.12c) is an eigenvalue problem, and as such, cannot be solved directly. Rather, we employ PI to obtain a converged CMFD solution, which requires multiple iterations. To do so, we introduce the power iteration index m. If the same convention of lagging the RHS terms is used, the following set of equations is obtained:

$$\begin{aligned} &\left(\Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} - \Phi_{1,k-\frac{1}{2}}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\right) + \bar{\Sigma}_{a,k}^{\left(l+\frac{1}{2}\right)} \Phi_{0,k}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} \Delta_k \\ &\quad = \lambda^{(l,m)} \overline{\left(\nu\Sigma_f\right)}_k^{\left(l+\frac{1}{2}\right)} \Phi_{0,k}^{\left(l+\frac{1}{2},m\right)} \Delta_k \,, \end{aligned} \tag{3.13a}$$

$$\Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} \equiv -\tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \left(\Phi_{0,k+1}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} - \Phi_{0,k}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\right) + \hat{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \left(\Phi_{0,k+1}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} + \Phi_{0,k}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\right) , \tag{3.13b}$$

$$0 = \Phi_{1,\frac{1}{2}}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} + B_{\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \Phi_{0,1}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} , \tag{3.13c}$$

$$0 = -\Phi_{1,K+\frac{1}{2}}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} + B_{K+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \Phi_{0,K}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} . \tag{3.13d}$$

Here we can make the following initial guesses for the low-order eigenpair:

$$\boldsymbol{\Phi}_0^{\left(l+\frac{1}{2},0\right)} \equiv \boldsymbol{\Phi}_0^{\left(l+\frac{1}{2}\right)} , \tag{3.13e}$$

$$\lambda^{(l,0)} \equiv \lambda^{(l)} , \tag{3.13f}$$

in which we define the vector $\boldsymbol{\Phi}_0$ as

$$\boldsymbol{\Phi}_0 \equiv \begin{pmatrix} \Phi_{0,1} \\ \Phi_{0,2} \\ \vdots \\ \Phi_{0,K} \end{pmatrix} , \quad \boldsymbol{\Phi}_0 \in \mathbb{R}^{K\times 1} , \tag{3.13g}$$

where $\mathbb{R}^{K\times 1}$ denotes real number space of dimension $K \times 1$. These equations can be rewritten in operator form as shown below:

$$\mathbb{M}^{\left(l+\frac{1}{2}\right)} \boldsymbol{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} = \lambda^{(l,m)} \mathbb{F}^{\left(l+\frac{1}{2}\right)} \boldsymbol{\Phi}_0^{\left(l+\frac{1}{2},m\right)} , \tag{3.13h}$$

where $\mathbb{M}^{\left(l+\frac{1}{2}\right)}$ is the *migration* operator, and $\mathbb{F}^{\left(l+\frac{1}{2}\right)}$ is the *fission* operator. Solving Eq. (3.13h) yields $\boldsymbol{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}$.

Next, we wish to obtain the next eigenvalue iterate $\lambda^{(l,m+1)}$. To do so, we first make the following observation using Eq. (3.13h):

$$\mathbb{M}^{\left(l+\frac{1}{2}\right)} \boldsymbol{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} \approxeq \lambda^{(l,m+1)} \mathbb{F}^{\left(l+\frac{1}{2}\right)} \boldsymbol{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} . \tag{3.13i}$$

Now, combining Eqs. (3.13h) and (3.13i) results in:

$$\lambda^{(l,m+1)} \mathbb{F}^{\left(l+\frac{1}{2}\right)} \boldsymbol{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} \approxeq \lambda^{(l,m)} \mathbb{F}^{\left(l+\frac{1}{2}\right)} \boldsymbol{\Phi}_0^{\left(l+\frac{1}{2},m\right)} . \tag{3.13j}$$

By weighting each side of Eq. (3.13j) by $\mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}$, we obtain:

$$\begin{aligned} &\lambda^{(l,m+1)}\left\langle \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}, \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}\right\rangle \\ &\approxeq \lambda^{(l,m)}\left\langle \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m)}, \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}\right\rangle , \end{aligned} \tag{3.13k}$$

where the $\langle\cdot,\cdot\rangle$ notation indicates a vector inner product such that

$$\langle \boldsymbol{f}, \boldsymbol{g}\rangle \equiv \sum_k f_k g_k \Delta_k \ . \tag{3.13l}$$

Rearranging Eq. (3.13k) for $\lambda^{(l,m+1)}$ allows us to make the following definition:

$$\lambda^{(l,m+1)} \equiv \lambda^{(l,m)} \frac{\left\langle \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m)}, \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}\right\rangle}{\left\langle \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}, \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}\right\rangle} \ . \tag{3.13m}$$

Because the coarse-mesh scalar flux is unique up to an arbitrary multiplicative constant, we must impose a normalization condition to obtain a unique solution. As an example, we will impose the following condition:

$$P = \left\langle \mathbf{1}, \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+1)}\right\rangle , \tag{3.13n}$$

where P is the desired total neutron production rate of the system, and $\mathbf{1}$ is a one-vector of length K. To enforce the condition stated in Eq. (3.13n) and obtain $\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+1)}$, we make the following definition:

$$\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+1)} \equiv \left(\frac{P}{\left\langle \mathbf{1}, \mathbb{F}^{(l+\frac{1}{2})}\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}\right\rangle} \right) \boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})} \ . \tag{3.13o}$$

With these definitions, a single power iteration consists of obtaining $\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+1)}$ and $\lambda^{(l,m+1)}$ using $\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m)}$ and $\lambda^{(l,m)}$. Once M power iterations have been completed and sufficient convergence of the quantities of interest has been achieved, we update the eigenpair iteration indices to be only in terms of the outer iteration index:

$$\boldsymbol{\Phi}_0^{(l+1)} \equiv \boldsymbol{\Phi}_0^{(l+\frac{1}{2},M)} , \tag{3.13p}$$

$$\lambda^{(l+1)} \equiv \lambda^{(l,M)} \ . \tag{3.13q}$$

Finally, the fine mesh scalar flux is updated with the solution of the CMFD equations as shown in Eq. (3.8q) to obtain $\boldsymbol{\phi}_0^{(l+1)}$ such that

$$\boldsymbol{\phi}_0 \equiv \begin{pmatrix} \phi_{0,1} \\ \phi_{0,2} \\ \vdots \\ \phi_{0,j} \end{pmatrix} , \quad \boldsymbol{\phi}_0 \in \mathbb{R}^{J\times 1} . \tag{3.13r}$$

Section 3.3.2 carefully describes how the equations derived here are used for CMFD-accelerated transport problems.

3.3 Algorithm Overview

In this section, the CMFD-accelerated transport algorithm is fully described for fixed-source and eigenvalue problems. We have placed the transport calculation at the beginning of the iteration, but the order of the high- and low-order calculations can be reversed (as it is in MPACT). For simplicity, we assume a discretized 1D, 1-group $\mathrm{S_N}$ transport problem. Outer iterations are indexed with l and power iterations are indexed with m (which are only relevant for eigenvalue problems). The fine-mesh cells of the transport problem are indexed with j and the coarse cells of the low-order problem are indexed with k; each coarse cell consists of a contiguous set of fine cells. For the closure relation of the fine-mesh cell-edge and cell-average angular fluxes, the diamond-difference scheme is assumed.

3.3.1 Iteration Scheme for Fixed-Source Problems

0. **Setup**: This first step is done only once before the iterative process begins. We make the following initial guess for the fine mesh scalar flux:

$$\boldsymbol{\phi}_0^{(0)} \equiv \boldsymbol{f} , \tag{3.14}$$

where $\boldsymbol{f}$ is an arbitrarily-chosen initial guess vector.

1. **Transport Sweep**: The transport problem is solved using the latest transport scalar flux estimate for the scattering source. The fixed-source $\mathrm{S_N}$ transport problem is shown below:

$$\frac{\mu_n}{h_j}\left(\psi_{n,j+\frac{1}{2}}^{(l+\frac{1}{2})} - \psi_{n,j-\frac{1}{2}}^{(l+\frac{1}{2})}\right) + \Sigma_{t,j}\psi_{n,j}^{(l+\frac{1}{2})} = \frac{1}{2}\left(\Sigma_{s,j}\phi_{0,j}^{(l)} + q_j\right) , \qquad \text{(Eq. (3.8b) revisited)}$$

$$\psi_{n,j}^{(l+\frac{1}{2})} = \frac{1}{2}\left(\psi_{n,j+\frac{1}{2}}^{(l+\frac{1}{2})} + \psi_{n,j-\frac{1}{2}}^{(l+\frac{1}{2})}\right) , \qquad \text{(Eq. (3.8c) revisited)}$$

$$\{j \mid j \in \mathbb{N}, 1 \le j \le J\} ,$$

$$\psi_{n,\frac{1}{2}}^{(l+\frac{1}{2})} = \psi_n^b , \quad \{n \mid \mu_n > 0\} , \tag{3.15a}$$

$$\psi_{n,J+\frac{1}{2}}^{(l+\frac{1}{2})} = \psi_n^b , \quad \{n \mid \mu_n < 0\} , \tag{3.15b}$$

$$\{n \mid n \in \mathbb{N}, 1 \le n \le N\} .$$

Rather than store the full angular flux ψ, which can require a large amount of memory, the following quantities (scalar flux and neutron current) are used as the iterates during the transport sweep:

$$\phi_{0,j}^{(l+\frac{1}{2})} \equiv \sum_{n=1}^{N} w_n \psi_j^{(l+\frac{1}{2})} , \tag{3.16}$$

$$\Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})} \equiv \sum_{n=1}^{N} w_n \mu_n \psi_{n,p_k+\frac{1}{2}}^{(l+\frac{1}{2})} . \tag{3.17}$$

We also wish to note that the neutron current need only be stored on the edges of the *coarse mesh*. The vector containing all coarse cell-edge currents is defined as:

$$\boldsymbol{\Phi}_1 \equiv \begin{pmatrix} \Phi_{1,\frac{1}{2}} \\ \Phi_{1,\frac{3}{2}} \\ \vdots \\ \Phi_{1,K+\frac{1}{2}} \end{pmatrix} , \quad \boldsymbol{\Phi}_1 \in \mathbb{R}^{(K+1)\times 1} . \tag{3.18}$$

2. **Restriction**: The next step is *restriction*, in which information on the fine mesh is collapsed onto the coarse mesh. This information takes the form of the quantities defined below, and represents the coarse-mesh scalar flux, coarse-cell physical thickness, homogenized cross sections, the coarse-mesh transport correction factor (defined in terms of the coarse-mesh scalar flux and neutron current obtained from the transport sweep, as well as mesh and material properties), and the numerical diffusion coefficient:

$$\Phi_{0,k}^{(l+\frac{1}{2})} \equiv \frac{\sum_{j\in k} h_j \phi_{0,j}^{(l+\frac{1}{2})}}{\Delta_k} , \tag{3.19}$$

$$\Delta_k \equiv \sum_{j\in k} h_j , \qquad \text{(Eq. (3.7d) revisited)}$$

$$\bar{\Sigma}_{u,k}^{\left(l+\frac{1}{2}\right)} \equiv \frac{\sum\limits_{j\in k} \Sigma_{u,j}\phi_{0,j}^{\left(l+\frac{1}{2}\right)} h_j}{\sum\limits_{j\in k} \phi_{0,j}^{\left(l+\frac{1}{2}\right)} h_j} \ , \forall \ u \in \{a, s, t\} \ , \tag{Eq. (3.8h) revisited}$$

$$\hat{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \frac{\Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} + \tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \left(\Phi_{0,k+1}^{\left(l+\frac{1}{2}\right)} - \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}\right)}{\Phi_{0,k+1}^{\left(l+\frac{1}{2}\right)} + \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}} \ , \tag{Eq. (3.8g) revisited}$$

$$\tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k+1}^{\left(l+\frac{1}{2}\right)}\Delta_{k+1} + \bar{\Sigma}_{t,k}^{\left(l+\frac{1}{2}\right)}\Delta_k\right)} \ . \tag{Eq. (3.8f) revisited}$$

   We note that the definition of $\tilde{D}$ does not affect the accuracy of the solution, but can affect the performance of the acceleration method. Additionally, of the quantities defined above, the following are nonlinear with respect to the scalar flux: the homogenized cross sections (Eq. (3.8h)), the transport correction term (Eq. (3.8g)), and the numerical diffusion coefficient Eq. (3.8f). These quantities will be further discussed in Section 3.5.

3. **Low-Order Solve**: Now, the low-order CMFD equation is solved as a linear system (using one of the methods discussed in Section 2.4.3):

$$\begin{gathered}\left(\Phi_{1,k+\frac{1}{2}}^{(l+1)} - \Phi_{1,k-\frac{1}{2}}^{(l+1)}\right) + \bar{\Sigma}_{a,k}^{\left(l+\frac{1}{2}\right)}\Phi_{0,k}^{(l+1)}\Delta_k = \bar{Q}_k\Delta_k \ , \\ \{k \mid k \in \mathbb{N}, 1 \le k \le K\} \ ,\end{gathered} \tag{Eq. (3.8d) revisited}$$

$$\begin{gathered}\Phi_{1,k+\frac{1}{2}}^{(l+1)} \equiv -\tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\left(\Phi_{0,k+1}^{(l+1)} - \Phi_{0,k}^{(l+1)}\right) + \hat{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\left(\Phi_{0,k+1}^{(l+1)} + \Phi_{0,k}^{(l+1)}\right) \ , \\ \{k \mid k \in \mathbb{N}, 1 \le k < K\} \ ,\end{gathered} \tag{Eq. (3.8e) revisited}$$

$$2\sum_{\mu_n>0}|\mu_n|w_n\psi_n^b = \left(\frac{\sum\limits_{n=1}^{N}|\mu_n|w_n\psi_{n,\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}}{\Phi_{0,1}^{\left(l+\frac{1}{2}\right)}}\right)\Phi_{0,1}^{(l+1)} + \Phi_{1,\frac{1}{2}}^{(l+1)} \ , \tag{3.20a}$$

$$2\sum_{\mu_n<0}|\mu_n|w_n\psi_n^b = \left(\frac{\sum\limits_{n=1}^{N}|\mu_n|w_n\psi_{n,J+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}}{\Phi_{0,K}^{\left(l+\frac{1}{2}\right)}}\right)\Phi_{0,K}^{(l+1)} - \Phi_{1,K+\frac{1}{2}}^{(l+1)} \ . \tag{3.20b}$$

4. **Prolongation**: Last is the *prolongation* step, in which information from the low-order solution is transferred to the high-order solution. The fine-mesh scalar flux in a given coarse cell is scaled by the ratio of the low-order scalar flux solutions from

before and after the evaluation of Eqs. (3.20). This correction is shown below:

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{\left(l+\frac{1}{2}\right)} \left(\frac{\Phi_{0,k}^{(l+1)}}{\Phi_{0,k}^{\left(l+\frac{1}{2}\right)}} \right) , \forall\, j \in k \, . \qquad \text{(Eq. (3.8q) revisited)}$$

This step is also nonlinear, and will be discussed in Section 3.5. If the quantities of interest (usually the fine mesh scalar flux) are not converged, we return to step 1 and resume the iterative process. To check convergence of the fine mesh scalar flux, the following normalized L_2 norm formulation can be used,

$$\left\| \frac{\boldsymbol{\phi}_0^{(l+1)} - \boldsymbol{\phi}_0^{(l)}}{\boldsymbol{\phi}_0^{(l+1)}} \right\|_2 < \epsilon_1 \, , \tag{3.21}$$

where ϵ_1 is the chosen flux tolerance. This convergence check is code-dependent.

The algorithm presented in this section is summarized in Algorithm 4.

Algorithm 4 CMFD-Accelerated SI Applied to Fixed-Source Problems

```
Assume an initial guess for the fine mesh scalar flux $\boldsymbol{\phi}_0^{(l)}$ as defined in Eq. (3.14)
repeat
    Obtain $\boldsymbol{\phi}_0^{(l+\frac{1}{2})}$ and $\boldsymbol{\Phi}_1^{(l+\frac{1}{2})}$ by solving Eq. (3.8b), Eq. (3.8c), and Eqs. (3.15)
    Obtain $\boldsymbol{\Phi}_0^{(l+\frac{1}{2})}$ using Eq. (3.19) and other coarse mesh quantities
    Obtain $\boldsymbol{\Phi}_0^{(l+1)}$ by solving Eqs. (3.8d) and (3.8e) and Eqs. (3.20)
    Obtain $\boldsymbol{\phi}_0^{(l+1)}$ using Eq. (3.8q)
until convergence                              {generally, convergence of $\boldsymbol{\phi}_0$ is checked}
```

3.3.2 Iteration Scheme for Eigenvalue Problems

0. **Setup**: Just as with the fixed-source algorithm, this step is done only once before the iterative process begins. An initial guess for the fundamental eigenpair is made, as below:

$$\boldsymbol{\phi}_0^{(0)} \equiv \boldsymbol{f} \, , \tag{3.22a}$$

$$\lambda^{(0)} \equiv \lambda_{\text{ini}} \, , \tag{3.22b}$$

where $\boldsymbol{f}$ is an arbitrary initial guess vector and λ_{ini} is a positive constant.

1. **Transport Sweep**: The transport problem is solved using the latest transport scalar flux and λ-eigenvalue estimates in the scattering and fission source terms. This prob-

lem is outlined below:

$$\frac{\mu_n}{h_j}\left(\psi_{n,j+\frac{1}{2}}^{(l+\frac{1}{2})} - \psi_{n,j-\frac{1}{2}}^{(l+\frac{1}{2})}\right) + \Sigma_{t,j}\psi_{n,j}^{(l+\frac{1}{2})} = \frac{1}{2}\left(\Sigma_{s,j} + \lambda^{(l)}\left(\nu\Sigma_f\right)_j\right)\phi_{0,j}^{(l)} \,, \tag{Eq. (3.12a) revisited}$$

$$\psi_{n,j}^{(l+\frac{1}{2})} = \frac{1}{2}\left(\psi_{n,j+\frac{1}{2}}^{(l+\frac{1}{2})} + \psi_{n,j-\frac{1}{2}}^{(l+\frac{1}{2})}\right) \,, \tag{Eq. (3.12b) revisited}$$

$$\{j \mid j \in \mathbb{N}, 1 \le j \le J\} \,,$$

$$\psi_{n,\frac{1}{2}}^{(l+\frac{1}{2})} = 0 \,, \quad \{n \mid \mu_n > 0\} \,, \tag{3.23a}$$

$$\psi_{n,J+\frac{1}{2}}^{(l+\frac{1}{2})} = 0 \,, \quad \{n \mid \mu_n < 0\} \,, \tag{3.23b}$$

$$\{n \mid n \in \mathbb{N}, 1 \le n \le N\} \,.$$

After the above fixed-source transport problem has been solved, the scalar flux and neutron current are stored as defined in Eqs. (3.16) and (3.17).

2. **Restriction**: Next is the restriction step, in which the low-order quantities, defined in Eqs. (3.7d), (3.8f) to (3.8h) and (3.19), are computed.

3. **Low-Order Solve & Eigenvalue Calculation**: The transport-corrected diffusion "power iteration" process is now begun. We define an initial guess for the low-order, transport-corrected scalar flux. In this case, we use our most recent estimate from the transport sweep:

$$\boldsymbol{\Phi}_0^{(l+\frac{1}{2},0)} \equiv \boldsymbol{\Phi}_0^{(l+\frac{1}{2})} \,, \tag{Eq. (3.13e) revisited}$$

$$\lambda^{(l,0)} \equiv \lambda^{(l)} \,. \tag{Eq. (3.13f) revisited}$$

For our power iterations we use the iteration index m. We proceed with a total number of M power iterations which take on values $m + 1 = 1, \ldots, M$. The total number of power iterations executed depends on the convergence criterion of the low-order problem, and we may choose to limit the total number of power iterations that can be executed.

First, in the power iteration, we solve the problem given below for $\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}$:

$$\begin{aligned}\left(\Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2},m+\frac{1}{2})} - \Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2},m+\frac{1}{2})}\right) + \bar{\Sigma}_{a,k}^{(l+\frac{1}{2})}\Phi_{0,k}^{(l+\frac{1}{2},m+\frac{1}{2})}\Delta_k \\ = \lambda^{(l,m)}\overline{\left(\nu\Sigma_f\right)}_k^{(l+\frac{1}{2})}\Phi_{0,k}^{(l+\frac{1}{2},m)}\Delta_k \,,\end{aligned} \tag{Eq. (3.13a) revisited}$$

$$\begin{aligned}\Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} &\equiv -\tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\left(\Phi_{0,k+1}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} - \Phi_{0,k}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\right) \\ &\quad +\hat{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\left(\Phi_{0,k+1}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} + \Phi_{0,k}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\right) ,\end{aligned} \qquad \text{(Eq. (3.13b) revisited)}$$

$$0 = \left(\frac{\sum_{n=1}^{N}|\mu_n|w_n\psi_{n,\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}}{\Phi_{0,1}^{\left(l+\frac{1}{2}\right)}}\right)\Phi_{0,1}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} + \Phi_{1,\frac{1}{2}}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} , \tag{3.24a}$$

$$0 = \left(\frac{\sum_{n=1}^{N}|\mu_n|w_n\psi_{n,J+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}}{\Phi_{0,K}^{\left(l+\frac{1}{2}\right)}}\right)\Phi_{0,K}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} - \Phi_{1,K+\frac{1}{2}}^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} . \tag{3.24b}$$

Next, the λ-eigenvalue is computed:

$$\lambda^{(l,m+1)} \equiv \lambda^{(l,m)}\frac{\left\langle \mathbb{F}^{\left(l+\frac{1}{2}\right)}\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m\right)}, \mathbb{F}^{\left(l+\frac{1}{2}\right)}\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\right\rangle}{\left\langle \mathbb{F}^{\left(l+\frac{1}{2}\right)}\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}, \mathbb{F}^{\left(l+\frac{1}{2}\right)}\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\right\rangle} . \qquad \text{(Eq. (3.13m) revisited)}$$

Additionally, we enforce the normalization condition:

$$\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+1\right)} \equiv \left(\frac{P}{\left\langle \mathbf{1}, \mathbb{F}^{\left(l+\frac{1}{2}\right)}\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\right\rangle}\right)\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} . \qquad \text{(Eq. (3.13o) revisited)}$$

After M power iterations have been completed and the quantities of interest have been sufficiently converged for the low-order problem, we update the outer iteration scalar flux and λ-eigenvalue estimates with those from the power iteration process:

$$\mathbf{\Phi}_0^{(l+1)} \equiv \mathbf{\Phi}_0^{\left(l+\frac{1}{2},M\right)} , \qquad \text{(Eq. (3.13p) revisited)}$$

$$\lambda^{(l+1)} \equiv \lambda^{(l,M)} . \qquad \text{(Eq. (3.13q) revisited)}$$

4. **Prolongation**: Lastly, the prolongation step is performed using Eq. (3.8q) to obtain $\phi_0^{(l+1)}$. If the fundamental eigenvalue and eigenvector have converged between outer iterations, then the overall iteration scheme is halted after L steps. For the scalar flux, the convergence check in Eq. (3.21) can be used, and the following can be used for the eigenvalue,

$$\left|\frac{\lambda^{(l+1)} - \lambda^{(l)}}{\lambda^{(l+1)}}\right| < \epsilon_2 , \tag{3.25}$$

where ϵ_2 is the chosen eigenvalue tolerance.

Algorithm 5 summarizes the algorithm presented in this section, where the nested iterative process starting on line 7 is PI. We have focused on the use of PI to solve the low-order eigenvalue problem, but other methods exist for obtaining the eigenpairs of a system. Some of these other methods are: Arnoldi iteration [53], Generalized Davidson [54], and Wielandt-shifted PI (also known as the shifted power method) [55]. Section 3.4 will discuss PI and Wielandt-shifted PI, which are common methods used in reactor physics calculations to obtain the fundamental eigenpair. These methods are also relevant for solving the low-order LDA problem described in Chapter 4.

Algorithm 5 CMFD-Accelerated SI Applied to Eigenvalue Problems

```
 1: Assume an initial guess for the eigenpair as defined in Eqs. (3.22)
 2: repeat
 3:     Obtain $\phi_0^{(l+\frac{1}{2})}$ and $\mathbf{\Phi}_1^{(l+\frac{1}{2})}$ by solving Eq. (3.12a), Eq. (3.12b), and Eqs. (3.23)
 4:     Obtain $\mathbf{\Phi}_0^{(l+\frac{1}{2})}$ using Eq. (3.19) and other coarse mesh quantities
 5:     $\mathbf{\Phi}_0^{(l+\frac{1}{2},0)} \leftarrow \mathbf{\Phi}_0^{(l+\frac{1}{2})}$
 6:     $\lambda^{(l,0)} \leftarrow \lambda^{(l)}$
 7:     repeat
 8:         Obtain the scalar flux $\mathbf{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}$ using Eq. (3.13a), Eq. (3.13b), and Eqs. (3.24)
 9:         Compute the eigenvalue $\lambda^{(l,m+1)}$ using Eq. (3.13m)
10:         Obtain the normalized scalar flux $\mathbf{\Phi}_0^{(l+\frac{1}{2},m+1)}$ using Eq. (3.13o)
11:     until convergence              {generally, convergence of $\mathbf{\Phi}_0$ and $\lambda$ are checked}
12:     $\mathbf{\Phi}_0^{(l+1)} \leftarrow \mathbf{\Phi}_0^{(l+\frac{1}{2},M)}$
13:     $\lambda^{(l+1)} \leftarrow \lambda^{(l,M)}$
14:     Obtain $\phi_0^{(l+1)}$ using Eq. (3.8q)
15: until convergence                  {generally, convergence of $\phi_0$ and $\lambda$ are checked}
```

3.4 Power Iteration Methods

For eigenvalue problems, *Power Iteration* (PI) is performed to compute the fundamental eigenpair for the low-order CMFD problem. It can be shown that the PI process yields the smallest eigenvalue, which is the eigenvalue of interest for reactor applications. (Though we colloquially refer to this process as *power iteration* in the field of reactor physics, it is known mathematically as the *inverse power method* or simply *inverse iteration*.) For the discussions in this section, the CMFD equation (defined in Eq. (3.13a)) is depicted in operator notation as defined below:

$$\mathbb{M}^{(l+\frac{1}{2})}\mathbf{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})} = \lambda^{(l,m)}\mathbb{F}^{(l+\frac{1}{2})}\mathbf{\Phi}_0^{(l+\frac{1}{2},m)} , \qquad \text{(Eq. (3.13h) revisited)}$$

where $\mathbb{M}$ is the neutron loss operator and $\mathbb{F}$ is the fission source operator. By modifying this eigenvalue problem, the convergence rate of PI can be improved.

3.4.1 Standard (Unshifted) Power Iteration

With the operator form of the CMFD equation shown in Eq. (3.13h), the PI algorithm for CMFD can be summarized as shown in Algorithm 6. A single power iteration consists of the execution of steps 3 through 5.

Algorithm 6 Power Iteration Algorithm for CMFD-Accelerated Eigenvalue Problems

```
1: Assume an initial guess for $\mathbf{\Phi}_0^{(l+\frac{1}{2},0)}$ and $\lambda^{(l,0)}$
2: repeat
3:     Compute the scalar flux $\mathbf{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}$ by solving Eq. (3.13h)
4:     Compute the eigenvalue $\lambda^{(l,m+1)}$ using Eq. (3.13m)
5:     Normalize the scalar flux using Eq. (3.13o) to obtain $\mathbf{\Phi}_0^{(l+\frac{1}{2},m+1)}$
6: until convergence        {generally, convergence of $\mathbf{\Phi}_0$ and $\lambda$ are checked}
```

The convergence rate of PI is dependent on the eigenvalues of the problem. Specifically, the rate of convergence, ρ_{PI}, is equal to the ratio of the second largest eigenvalue to the largest eigenvalue:

$$\rho_{\mathrm{PI}} = \frac{|\lambda_{\text{2nd largest}}|}{|\lambda_{\text{largest}}|} , \tag{3.26}$$

which is also referred to as the *dominance ratio* [55]. (As a reminder, we are interested in computing the *smallest* eigenvalue using the PI process, which we simply refer to as λ throughout this chapter.) Therefore, PI can converge slowly if there is an eigenvalue that is close in magnitude to the dominant eigenvalue. Unfortunately, this is often the case for reactor physics problems. Thus, a modified iterative process, discussed in the following section, can be used to improve the convergence rate of PI.

3.4.2 Power Iteration with Wielandt Shift

Here, we discuss the Wielandt-shifted eigenvalue problem [56, 57] (also known as *inverse-shifted power method* or simply *inverse power method* in the field of mathematics [55]). A simple modification is made to Eq. (3.13h), in which the term $\lambda_{\mathrm{sh}}\mathbb{F}^{(l+\frac{1}{2})}\mathbf{\Phi}_0^{(l+\frac{1}{2})}$

is subtracted from the equation, as shown below:

$$\begin{gathered}\left(\mathbb{M}^{(l+\frac{1}{2})} - \lambda_{\text{sh}}\mathbb{F}^{(l+\frac{1}{2})}\right)\mathbf{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})} = \left(\lambda^{(l,m)} - \lambda_{\text{sh}}\right)\mathbb{F}^{(l+\frac{1}{2})}\mathbf{\Phi}_0^{(l+\frac{1}{2},m)} \ , \\ 0 \le \lambda_{\text{sh}} < \lambda^{(l,m)} \ . \end{gathered} \tag{3.27}$$

The shift amount may be constant, iteration-dependent, or even space-dependent [11]. Each of these approaches affect the performance of PI differently, but the general goal is to select a shift magnitude λ_{sh} such that $\lambda_{\text{sh}} < \lambda$ and $|\lambda - \lambda_{\text{sh}}|$ is small. This improves the convergence rate of PI, thereby reducing the total number of power iterations required for convergence to a given tolerance (and a corresponding reduction in computation expense). By default, the shift amount is constant and equal to $\frac{2}{3}$ in MPACT.

MPACT also contains an option for an "adaptive" shift in which the magnitude of the shift is changed each power iteration. This method is based on the *Purdue Advanced Reactor Core Simulator* (PARCS) code [58], and an optimal shift is estimated using the following empirical formula:

$$\lambda_{\text{sh}}^{(l,m)} \equiv \max\left\{\lambda^{(l,m)} - c_1\left|\lambda^{(l,m)} - \lambda^{(l,m-1)}\right| - c_0 \ , \lambda_{\text{min}}\right\} \ , \tag{3.28}$$

where c_0 and c_1 are user-specified constants. By computing the shift in this way, the shift magnitude should always be close to λ but never larger (due to the small, positive constant c_0, which adds a degree of conservatism to the shift estimate). Additionally, the shift magnitude will never be smaller than the chosen value of λ_{min}. In this way, the risk of λ_{sh} exceeding λ is minimized while maintaining that λ_{sh} is close in magnitude to λ.

By shifting the eigenvalue problem as shown in Eq. (3.27), the dominance ratio is effectively reduced. As a result, the convergence rate of PI with this modification is improved compared to the unshifted case. The convergence rate of the shifted case is shown below [59]:

$$\rho_{\text{sh,PI}} = \frac{1 - \frac{\lambda_{\text{sh}}}{\lambda_{\text{2nd largest}}}}{1 - \frac{\lambda_{\text{sh}}}{\lambda_{\text{largest}}}} \cdot \rho_{\text{PI}} \ , \tag{3.29}$$

where the convergence rate of the unshifted case, ρ_{PI}, is defined in Eq. (3.26). Because $\lambda_{\text{largest}} > \lambda_{\text{2nd largest}} > \lambda^{(l,m)} > \lambda_{\text{sh}}$, $\rho_{\text{sh,PI}}$ is always less than ρ_{PI}.

There are two notable limitations to the Wielandt shift method. The first is that choosing the optimal shift relies on knowledge of $\lambda^{(l,m)}$ beforehand. However, this quantity is unknown until the problem reaches sufficient convergence. Thus, the shift quantity must be estimated. The second limitation is that the shift must be less than the fundamental (smallest) eigenvalue of the system, as shown in Eq. (3.27). If $\lambda_{\text{sh}} = \lambda^{(l,m)}$, then the operator $\mathbb{M}^{(l+\frac{1}{2})} - \lambda_{\mathrm{sh}}\mathbb{F}^{(l+\frac{1}{2})}$ is singular and, therefore, non-invertable. In this case, the linear system is unsolvable by traditional means. Additionally, if $\lambda_{\mathrm{sh}} > \lambda^{(l,m)}$ then the problem may converge to the incorrect eigenpair. Care must be taken when choosing a suitable shift parameter to avoid these issues. (As previously stated, the adaptive shift method attempts to avoid these issues but still requires some degree of conservatism.)

One additional consideration for Wielandt-shifted cases is that the condition number of the shifted LHS operator increases as λ_{sh} approaches $\lambda^{(l,m)}$. Linear solver methods for which performance is affected by the condition number of the operator may suffer if λ_{sh} is very close to $\lambda^{(l,m)}$. In this case, one may choose λ_{sh} such that the iterative performance of PI is sufficiently improved while avoiding a dramatic reduction in performance of the iterative linear solver method.

3.5 CMFD Nonlinearities

The CMFD formulation contains terms that are nonlinear with respect to the scalar flux. In other words, there are terms in the formulation that involve division by (i) the scalar flux or (ii) quantities that depend on the scalar flux. For iterative transport schemes that do not explicitly guarantee the following strict positivity condition for the scalar flux iterates:

$$\begin{gathered} \phi^{(l)}(\boldsymbol{x}, E, t) > 0 \,, \\ \forall \boldsymbol{x} \in V \,, \quad \forall E \in [0, \infty) \,, \quad \forall t \,, \quad \forall l \,, \end{gathered} \tag{3.30}$$

(where l is the iteration index and t is the time variable) such as the 2D/1D method, the possibility of numerical failure exists due to these nonlinear terms. This is fundamentally due to the possibility that terms in the denominator become very small, resulting in large changes in a nonlinear term for a small change in the solution. The nonlinear terms in CMFD that are susceptible to numerical instability are listed in this section. A 1D formulation is assumed for simplicity, but these nonlinear terms are present regardless of the spatial dimensionality.

3.5.1 Homogenized Cross Sections

The definition of iteration-dependent homogenized cross sections in a coarse cell is reproduced here:

$$\bar{\Sigma}_{u,k}^{\left(l+\frac{1}{2}\right)} \equiv \frac{\sum\limits_{j\in k} \Sigma_{u,j} \phi_{0,j}^{\left(l+\frac{1}{2}\right)} h_j}{\sum\limits_{j\in k} \phi_{0,j}^{\left(l+\frac{1}{2}\right)} h_j} \;, \forall\; u \in \{a, s, t\} \;. \qquad \text{(Eq. (3.8h) revisited)}$$

As a result of the flux- and volume-weighting of the cross sections on the fine mesh, the denominator contains a volume sum of the scalar flux iterate in a coarse cell. If the scalar flux iterates are not strictly positive in the coarse cell, or if the magnitude of the flux in the cell is very small, then the homogenized cross section in that cell is susceptible to numerical instability if $\sum\limits_{j\in k} h_j \phi_{0,j}$ evaluates to near zero.

3.5.2 Numerical Diffusion Coefficient

The traditional expression for the diffusion coefficient used in Fick's Law is reproduced here:

$$\tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k+1}^{\left(l+\frac{1}{2}\right)} \Delta_{k+1} + \bar{\Sigma}_{t,k}^{\left(l+\frac{1}{2}\right)} \Delta_k\right)} \;. \qquad \text{(Eq. (3.8f) revisited)}$$

The numerical expression for $\tilde{D}$ possesses the potential for numerical instability in the denominator term. Specifically, the terms $\bar{\Sigma}_{t,k}^{\left(l+\frac{1}{2}\right)} \Delta_k$ and $\bar{\Sigma}_{t,k+1}^{\left(l+\frac{1}{2}\right)} \Delta_{k+1}$ have the potential to be equal in magnitude and opposite in sign for any iterate. Since $\bar{\Sigma}_t$ is a flux-weighted average over a coarse cell, there is a possibility that the presence of negative scalar flux quantities can result in a negative cross section. If the two denominator terms are nearly equal in magnitude and opposite in sign, $\tilde{D}$ can become numerically unstable.

3.5.3 Transport Correction Term

The transport correction term to the diffusion equation, $\hat{D}$, is reproduced here:

$$\hat{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \frac{\Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} + \tilde{D}_{k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \left(\Phi_{0,k+1}^{\left(l+\frac{1}{2}\right)} - \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}\right)}{\Phi_{0,k+1}^{\left(l+\frac{1}{2}\right)} + \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}} \;. \qquad \text{(Eq. (3.8g) revisited)}$$

$\hat{D}$ possesses the potential for numerical instability in two ways:

1. The first is due to the presence of $\tilde{D}$ in the numerator. If $\tilde{D}$ becomes unstable, $\hat{D}$ will as well.
2. If the coarse cell flux in two adjacent cells are equal in magnitude and opposite in sign, $\hat{D}$ can become unstable.

These issues are possible for any iteration l.

3.5.4 Multiplicative Prolongation Factor

The prolongation equation for updating the fine-mesh scalar flux using the most recent coarse-mesh scalar flux solution is reproduced here:

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{\left(l+\frac{1}{2}\right)} \left(\frac{\Phi_{0,k}^{(l+1)}}{\Phi_{0,k}^{\left(l+\frac{1}{2}\right)}} \right) , \forall\ j \in k\ . \qquad \text{(Eq. (3.8q) revisited)}$$

If the scalar flux $\Phi_{0,k}^{\left(l+\frac{1}{2}\right)}$ in a single coarse cell approaches zero for any iterate, the update ratio that scales the fine mesh scalar flux can become sensitive to small fluctuations. In this case, the fine mesh scalar flux can become numerically unstable.

3.5.5 Summary of CMFD Nonlinearities

The CMFD formulation contains several nonlinear terms that render the method susceptible to nonlinear instability. These terms are nonlinear with respect to either (i) the scalar flux iterates or (ii) terms that depend on the scalar flux iterates. If any of these terms become numerically unstable, then the performance of CMFD can suffer and the method can behave unexpectedly. If the converged solution contains non-positive flux values, the nonlinear instability can appear. Additionally, this issue can also appear for *any scalar flux iterate* that is non-positive. Evidence of this phenomenon is provided in Chapters 5 to 7.

CHAPTER 4

Linear Diffusion Acceleration

This chapter describes the novel *Linear Diffusion Acceleration* (LDA) method, which is the focus of this thesis. All chapters before this contain known and published knowledge, and the novelty of this thesis project begins here. Some background and supporting information for LDA is provided first in Sections 4.1 and 4.2, before the method is derived for both fixed-source and eigenvalue problems in Section 4.3. Next, algorithm overviews are given for both problem types in Section 4.4. A "shifted" variation of the method is presented in Section 4.5 that allows for improved *Power Iteration* (PI) convergence when LDA is applied to eigenvalue problems. Lastly, we provide a discussion on the implications of the LDA formulation in Section 4.6.

4.1 Background

LDA is a transport acceleration method that is designed to avoid nonlinearities with respect to the scalar flux. The LDA method possesses some similarities to the GAML method [12], which was previously discussed in Chapter 1. In the GAML paper, the authors point out that nonlinear terms render a method susceptible to numerical instability. This is the same nonlinear instability that is discussed in Section 1.1, in which the algorithm can become numerically unstable if certain quantities (generally the scalar flux) are not strictly positive. The authors of the GAML method cite the use of modified DSA to accelerate eigenvalue problems as the motivation for the development of their method. DSA, in its "original" form, is a linear acceleration method that can only be used for fixed-source problems. However, a few different modifications to the method were proposed that allow it to be used for eigenvalue problems [21]. Each modification introduces nonlinearities into the DSA method. For example, one proposed modification to DSA suggested the following alternate definition of the diffusion coefficient:

$$\begin{gathered}\{D_g(\boldsymbol{x})\}_{ii} = -\frac{\phi_{1,g,i}(\boldsymbol{x})}{\nabla_i \phi_{0,g}(\boldsymbol{x})}\ , \\ \{i \mid i \in N\ , 1 \le i \le 3\}\ , \\ \{g \mid g \in N\ , 1 \le g \le G\}\ , \end{gathered} \tag{4.1}$$

where g is the energy group index, i is one of the orthogonal coordinate directions, ϕ_0 is the scalar flux, and ϕ_1 is the neutron current [21]. In this way, the diffusion coefficient becomes a diagonal tensor and is lagged in the iteration process. However, it is nonlinear with respect to the gradient of the scalar flux, rendering the method susceptible to nonlinear instabilities. The GAML method was a response to the nonlinearity of DSA when used for eigenvalue problems in the transport code ONEDANT [60].

In the GAML method, the low-order problem is formulated as a non-invertable operator acting on the scalar flux. Rather than embedding a nonlinear transport consistency term in the low-order LHS operator, linear consistency terms are placed in the source vector. To solve this singular problem, the *Fredholm Alternative Theorem* (FAT) is employed so that the fundamental transport eigenvalue is computed in such a way that allows for a solution to be obtained. The GAML method serves as the inspiration for the LDA method. LDA extends the capabilities of the GAML method to cases in which the low-order and high-order problems do not necessarily have the same spatial grids. Further, LDA employs the use of a Wielandt shift that improves the iterative performance of PI for solving the low-order eigenvalue problem (as shown later in Chapters 6 and 7). Appendix A reproduces the GAML paper, and provides a short discussion of the differences between GAML and LDA.

An intermediate version of LDA called *Semilinear Diffusion Acceleration* (SDA) was developed during this thesis project [13]. SDA more closely resembles the GAML method than LDA, with the focus of SDA being problems in which the spatial discretization of the transport and low-order problems are the same. Like LDA, SDA was developed to avoid nonlinearities in the low-order problem. However, SDA was not well-developed for problems in which the low-order problem possesses a different spatial mesh than the transport problem. As a result, when the SDA method was adapted for these types of problems, a substantial amount of additional computational work would be required to satisfy the FAT. LDA was adapted from SDA with the coarse low-order grid in mind, resulting in a more computationally efficient method.

In Chapter 3, the nonlinearities of CMFD were discussed. The presence of nonlinear terms render CMFD susceptible to nonlinear instabilities, in much the same way as DSA becomes susceptible with nonlinear modifications for application to eigenvalue problems. For cases in which the positivity of the scalar flux is not guaranteed, such as when using the 2D/1D method or TCP$_0$ scattering approximation, these nonlinear quantities have the potential to cause numerical instabilities. The motivation for using LDA comes from the desire to avoid numerical failure due to the presence of nonlinear terms.

Therefore, in cases where CMFD does not fail due to these nonlinear terms, LDA should have similar convergence behavior. For cases where CMFD exhibits degradation in performance as a result of its nonlinear instability, LDA should demonstrate superior performance. Therefore, LDA would not need to be only used instead of CMFD for cases in which CMFD is susceptible to numerical instability. LDA could then serve as a replacement to CMFD because it has similar linear stability properties. Because (i) the occurrence of nonlinear instabilities is generally unpredictable, and (ii) the cost of running CMFD and LDA would be comparable, it is hoped that LDA will be viewed as a more robust alternative to CMFD for all practical problems. We investigate these properties of LDA in Chapters 5 to 7.

For the notation in this chapter, boldface quantities represent spatial vectors. As an example, the term $\boldsymbol{\phi}_0$ represents the spatial vector of fine-grid scalar flux values. Additionally, lower-case terms generally represent fine-grid quantities and upper-case terms represent coarse-grid quantities. For example, the term $\boldsymbol{\Phi}_0$ represents a spatial vector of coarse-grid scalar flux values.

4.2 Fredholm Alternative Theorem

The *Fredholm Alternative Theorem* (FAT) is a useful mathematical tool that we employ in the formulation of LDA for eigenvalue problems. Specifically, the FAT states the conditions under which a solution of a system with a singular LHS operator exists by ensuring that a certain solvability condition is satisfied for the RHS vector. This section provides an overview of the FAT, as it is central to the LDA method.

The FAT applies when seeking a solution $\boldsymbol{f}$ of the following equation:

$$L\boldsymbol{f} = \boldsymbol{q} \,, \tag{4.2}$$

where L is a linear, singular operator. This implies that L possesses an eigenvalue $\lambda = 0$. In this work, we assume that the eigenvalue $\lambda = 0$ is *simple* – it has a single eigenfunction. This assumption is valid for all practical neutron transport problems. Now, let us consider the following:

$$L\boldsymbol{g} = 0 \;, \tag{4.3}$$

where $\boldsymbol{g}$ is the eigenfunction of L corresponding to $\lambda = 0$. Additionally, we consider:

$$L^*\boldsymbol{g}^* = 0 \;, \tag{4.4}$$

where L^* is the adjoint of L and $\boldsymbol{g}^*$ is the eigenfunction of L^* corresponding to $\lambda = 0$. The FAT states that Eq. (4.2) has a solution if and only if the following equations hold:

$$\boldsymbol{q} \in R(L) \;, \tag{4.5a}$$

$$\boldsymbol{q} \in N(L^*)^{\perp} \;, \tag{4.5b}$$

$$\langle \boldsymbol{g}^*, \boldsymbol{q} \rangle = 0 \;, \tag{4.5c}$$

where $R(L)$ refers to the range of L, $N(L^*)^{\perp}$ is the orthogonal complement of the nullspace of L, and the $\langle \cdot, \cdot \rangle$ notation indicates the inner product of vectors. Given that these conditions are satisfied, Eq. (4.2) has a unique particular solution $\tilde{\boldsymbol{f}}$ satisfying the same solvability condition as $\boldsymbol{q}$: $\left\langle \boldsymbol{g}^*, \tilde{\boldsymbol{f}} \right\rangle = 0$. The general solution of Eq. (4.2) is then shown below:

$$\boldsymbol{f} = \tilde{\boldsymbol{f}} + a\boldsymbol{g}, \quad -\infty < a < \infty \;, \tag{4.6}$$

where a is an arbitrary multiplicative constant.

To summarize, the FAT states that a solution to Eq. (4.2) exists (where the operator L is singular) if the RHS of Eq. (4.2) is orthogonal to the adjoint eigenfunction (the solution of Eq. (4.4)). This ensures that the RHS vector of Eq. (4.2) exists in the range of the operator L. In the derivation of LDA for eigenvalue problems, the low-order problem *initially* resembles Eq. (4.2). We note that the FAT simply states the conditions under which a solution to Eq. (4.2) exists. However, it does not provide the means to solve this linear system for $\boldsymbol{f}$, which would require inverting the singular operator L. Thus, we make some modifications to the low-order problem to avoid the need to invert a singular operator. In doing so, the actual LDA system to be solved in a practical application will not explicitly resemble Eq. (4.2). As shown in Section 4.3.2, we ultimately make use of the solvability condition in Eq. (4.5c) to derive an expression for the transport eigenvalue.

4.3 Derivation of LDA

This section describes the derivation of the LDA equations for both fixed-source and eigenvalue problems. For clarity, we derive the LDA equations for simple 1D, monoenergetic problems. However, LDA can easily be applied to more complex problem types in a straightforward way.

4.3.1 Fixed-Source Problems

Our goal in this section is to derive the low-order LDA equations on the coarse grid from the transport equation for fixed-source problems. The LDA equations will consist of (i) a coarse-grid balance equation with transport correction terms, (ii) equations to relate fine-grid quantities to coarse-grid quantities, and (iii) an equation to update the fine-grid solution from the coarse-grid solution.

We begin with the steady-state, monoenergetic, 1D NTE with prescribed boundary conditions shown below:

$$\mu\frac{\partial}{\partial x}\psi(x,\mu) + \Sigma_t(x)\psi(x,\mu) = \frac{1}{2}\left(\int_{-1}^{1}\Sigma_s(x,\mu,\mu')\psi(x,\mu')\,\mathrm{d}\mu' + Q(x)\right)\ , \qquad 0 \le x \le X\ , \quad -1 \le \mu \le 1\ , \tag{4.7a}$$

$$\psi(0,\mu) = \psi^b(\mu)\ , \quad \mu > 0\ , \tag{4.7b}$$

$$\psi(X,\mu) = \psi^b(\mu)\ , \quad \mu < 0\ , \tag{4.7c}$$

where $Q(x)$ is the inhomogeneous source. We first obtain the NBE by integrating Eq. (4.7a) over angle (by operating by $\int_{-1}^{1}(\cdot)\,\mathrm{d}\mu$) and rearranging:

$$\frac{\mathrm{d}}{\mathrm{d}x}\phi_1(x) + \Sigma_a(x)\phi_0(x) = Q(x)\ . \tag{4.8a}$$

Boundary conditions are obtained by operating on Eqs. (4.7b) and (4.7c) by $\int_{-1}^{1}\mu(\cdot)\,\mathrm{d}\mu$, as shown below:

$$\phi_1(0) = \int_{-1}^{0}\mu\psi(0,\mu)\,\mathrm{d}\mu + \int_{0}^{1}\mu\psi^b(\mu)\,\mathrm{d}\mu\ , \tag{4.8b}$$

$$\phi_1(X) = \int_{-1}^{0}\mu\psi^b(\mu)\,\mathrm{d}\mu + \int_{0}^{1}\mu\psi(X,\mu)\,\mathrm{d}\mu\ . \tag{4.8c}$$

To discretize Eq. (4.8a) in space, we average over a given "fine cell" that spans from $x_{j-\frac{1}{2}}$ to $x_{j+\frac{1}{2}}$. This amounts to operating by $\frac{1}{h_j}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}}(\cdot)\,\mathrm{d}x$, where h_j is the width of fine cell j, which yields the following discretized NBE (the cross sections are assumed to be constant in a given fine cell):

$$\begin{gathered}\left(\phi_{1,j+\frac{1}{2}}-\phi_{1,j-\frac{1}{2}}\right)+\Sigma_{a,j}\phi_{0,j}h_j=q_jh_j\ ,\\ \{j\mid j\in\mathbb{N},1\le j\le J\}\ ,\end{gathered}\tag{4.9a}$$

$$\phi_{1,\frac{1}{2}}=\int_{-1}^{0}\mu\psi_{\frac{1}{2}}(\mu)\,\mathrm{d}\mu+\int_{0}^{1}\mu\psi^{b}(\mu)\,\mathrm{d}\mu\ ,\tag{4.9b}$$

$$\phi_{1,J+\frac{1}{2}}=\int_{-1}^{0}\mu\psi^{b}(\mu)\,\mathrm{d}\mu+\int_{0}^{1}\mu\psi_{J+\frac{1}{2}}(\mu)\,\mathrm{d}\mu\ .\tag{4.9c}$$

In Eqs. (4.9), $\phi_{1,j\pm\frac{1}{2}}$ is the neutron current at a fine-cell edge, q_j is the magnitude of the neutron source in fine cell j, $\mathbb{N}$ is the set of natural integers, and J is the number of fine spatial cells. Next, Eq. (4.9a) is spatially integrated over a given "coarse cell" by operating by $\sum_{j\in k}(\cdot)$ where k is the coarse cell index. A coarse cell consists of a contiguous collection of fine cells, which are referred to as $j\in k$. The following coarse-grid balance equation is obtained:

$$\left(\Phi_{1,k+\frac{1}{2}}-\Phi_{1,k-\frac{1}{2}}\right)+\bar{\Sigma}_{a,k}\left\langle\!\left\langle\boldsymbol{\phi}_0\right\rangle\!\right\rangle\Phi_{0,k}\Delta_k=\bar{Q}_k\Delta_k\ ,\tag{4.10a}$$

with the various terms defined as:

$$\Phi_{1,k\pm\frac{1}{2}}\equiv\int_{-1}^{1}\mu\psi(X_{k\pm\frac{1}{2}},\mu)\,\mathrm{d}\mu\ ,\tag{4.10b}$$

$$\bar{\Sigma}_{a,k}\left\langle\!\left\langle\boldsymbol{f}\right\rangle\!\right\rangle\equiv\frac{\sum\limits_{j\in k}\Sigma_{a,j}f_jh_j}{\sum\limits_{j\in k}f_jh_j},\tag{4.10c}$$

$$\Phi_{0,k}\equiv\frac{\sum\limits_{j\in k}\phi_{0,j}h_j}{\Delta_k}\ ,\tag{4.10d}$$

$$\Delta_k\equiv\sum_{j\in k}h_j\ ,\tag{4.10e}$$

$$\begin{gathered}\bar{Q}_k\equiv\frac{\sum\limits_{j\in k}q_jh_j}{\Delta_k}\ ,\\ \{k\mid k\in\mathbb{N},1\le k\le K\}\ .\end{gathered}\tag{4.10f}$$

We note that the $g\left\langle\!\left\langle\boldsymbol{f}\right\rangle\!\right\rangle$ notation indicates that the coarse-mesh quantity g is constructed using the fine-mesh weighting vector $\boldsymbol{f}$.

Now, we define a weighted diffusion operator $\mathcal{L}_{d,fs}\left\langle\!\left\langle\boldsymbol{f}\right\rangle\!\right\rangle$ (with the subscripts d for "diffusion" and fs for "fixed-source") operating on the coarse-mesh scalar flux vector $\boldsymbol{\Phi}_0$ defined as

$$\mathbf{\Phi}_0 \equiv \begin{pmatrix} \Phi_{0,1} \\ \Phi_{0,2} \\ \vdots \\ \Phi_{0,K} \end{pmatrix} , \quad \mathbf{\Phi}_0 \in \mathbb{R}^{K\times 1} . \tag{4.11}$$

The operator $\mathcal{L}_{d,fs}$ requires a fine-mesh weighting function $\boldsymbol{f}$ that is used when generating the coarse-mesh cross sections and $\tilde{D}$. *This operator represents a weighted version of the migration operator of the neutron diffusion equation that corresponds to the transport problem that we are solving.* The operator is defined as

$$\begin{aligned}\left(\mathcal{L}_{d,fs} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \mathbf{\Phi}_0\right)_k \equiv {} & -\tilde{D}_{k+\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,k+1} - \Phi_{0,k}\right) + \tilde{D}_{k-\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,k} - \Phi_{0,k-1}\right) \\ & + \bar{\Sigma}_{a,k} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Phi_{0,k} \Delta_k \ ,\end{aligned} \tag{4.12a}$$

$$\begin{gathered}\tilde{D}_{k\pm\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Delta_k + \bar{\Sigma}_{t,k\pm1} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Delta_{k\pm1}\right)} \ , \\ \{k \mid k \in \mathbb{N}, 2 \le k \le K-1\} \ ,\end{gathered} \tag{4.12b}$$

for the interior coarse cells, where the $\tilde{D}$ term is the numerical diffusion coefficient from the traditional form of Fick's Law (shown in Eq. (2.23a)). However, Eq. (4.12a) does not hold at the boundaries of the system. (If this equation were applied to the boundaries, the unphysical terms $\Phi_{0,0}$ and $\Phi_{0,K+1}$ would appear.) At the boundaries, we make the following definitions of the diffusion operator:

$$\begin{aligned}\left(\mathcal{L}_{d,fs} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \mathbf{\Phi}_0\right)_1 \equiv {} & -\tilde{D}_{\frac{3}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,2} - \Phi_{0,1}\right) - \Phi_{1,\frac{1}{2},d} \\ & + \bar{\Sigma}_{a,1} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Phi_{0,1} \Delta_1 \ ,\end{aligned} \tag{4.12c}$$

$$\begin{aligned}\left(\mathcal{L}_{d,fs} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \mathbf{\Phi}_0\right)_K \equiv {} & \Phi_{1,K+\frac{1}{2},d} + \tilde{D}_{K-\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,K} - \Phi_{0,K-1}\right) \\ & + \bar{\Sigma}_{a,K} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Phi_{0,K} \Delta_K \ ,\end{aligned} \tag{4.12d}$$

where $\Phi_{1,\frac{1}{2},d}$ and $\Phi_{1,K+\frac{1}{2},d}$ are "diffusion neutron current" terms that are determined by the diffusion boundary conditions (which are discussed next).

Now, we derive boundary conditions for the diffusion problem stated in Eqs. (4.12):

$$2\Phi_1^+ = 2\sum_{\mu_n>0} \mu_n w_n \psi_n^b \tag{4.13}$$

$$= \sum_{n=1}^{N} \left(\mu_n + |\mu_n|\right) w_n \psi_{n,\frac{1}{2}} \tag{4.14}$$

$$= \Phi_{1,\frac{1}{2},d} + \sum_{n=1}^{N} |\mu_n| w_n \psi_{n,\frac{1}{2}} \ . \tag{4.15}$$

In the diffusion approximation, the angular flux is assumed to a linear function of angle:

$$\psi_{n,\frac{1}{2}} = \frac{1}{2}\Phi_{0,\frac{1}{2}} + \frac{3}{2}\mu_n \Phi_{1,\frac{1}{2}} \ . \tag{4.16}$$

Upon substituting Eq. (4.16) into the second term on the RHS of Eq. (4.15) and simplifying, we obtain:

$$\sum_{n=1}^{N} |\mu_n| w_n \psi_{n,\frac{1}{2}} = \frac{1}{2}\Phi_{0,\frac{1}{2}} \ . \tag{4.17}$$

Inserting Eq. (4.17) into Eq. (4.15) yields the following diffusion boundary condition for the left side of the system:

$$2\Phi_1^+ = \Phi_{1,\frac{1}{2},d} + \frac{1}{2}\Phi_{0,\frac{1}{2}} \ . \tag{4.18}$$

$\Phi_{0,\frac{1}{2}}$ can be found in terms of $\Phi_{1,\frac{1}{2},d}$ and $\Phi_{0,1}$ and eliminated from Eq. (4.18) by applying Fick's Law at the system boundary. As an example, we will perform these steps for the left side of the problem. Using the following definition of Fick's Law at the left side of the system,

$$\Phi_{1,\frac{1}{2},d} = -\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,1} - \Phi_{0,\frac{1}{2}}\right) \ , \tag{4.19}$$

where $\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle$ is defined as

$$\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \equiv \frac{2}{3\bar{\Sigma}_{t,1} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Delta_1} \ , \tag{4.20}$$

we find the following relationship between $\Phi_{0,\frac{1}{2}}$, $\Phi_{0,1}$, and $\Phi_{1,\frac{1}{2},d}$:

$$\Phi_{0,\frac{1}{2}} = \Phi_{0,1} + \frac{\Phi_{1,\frac{1}{2},d}}{\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle} \ . \tag{4.21}$$

Upon substituting Eq. (4.21) into Eq. (4.18) we obtain:

$$\Phi_{1,\frac{1}{2},d} = \frac{4\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle}{1 + 2\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle} \Phi_1^+ - \frac{\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle}{1 + 2\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle} \Phi_{0,1} \ . \tag{4.22}$$

Then, inserting Eq. (4.22) into Eq. (4.12c) and rearranging yields the following:

$$\begin{gathered}\left(\mathcal{L}_{d,fs} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \boldsymbol{\Phi}_0\right)_1 \equiv -\tilde{D}_{\frac{3}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Phi_{0,2} \\ + \left(\frac{\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle}{1 + 2\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle} + \tilde{D}_{\frac{3}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle + \bar{\Sigma}_{a,1} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Delta_1 \right) \Phi_{0,1} - \frac{4\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle}{1 + 2\tilde{D}_{\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle} \Phi_1^+ \ .\end{gathered} \tag{4.23}$$

With this, we have fully determined the form of Eq. (4.12c) with the diffusion boundary conditions incorporated into the operator.

A similar procedure to that started in Eq. (4.13) for the right side yields the following boundary condition for the diffusion system:

$$2\Phi_1^- = -\Phi_{1,K+\frac{1}{2},d} + \frac{1}{2}\Phi_{0,K+\frac{1}{2}} \ . \tag{4.24}$$

One can eliminate $\Phi_{0,K+\frac{1}{2}}$ from the system in a similar manner to that shown for the left side. The terms Φ_1^+ and Φ_1^- are the prescribed incoming neutron current at the left and right edges of the system, respectively, and are defined as

$$\Phi_1^+ \equiv \sum_{\mu_n > 0} \mu_n w_n \psi_n^b \ , \tag{4.25}$$

$$\Phi_1^- \equiv \sum_{\mu_n < 0} \mu_n w_n \psi_n^b \ . \tag{4.26}$$

For problems with vacuum boundary conditions, these incoming current terms are zero. Further, for problems with reflective boundary conditions, the diffusion current $(\Phi_{1,d})$ at the reflecting boundary is simply zero. With this, we have derived the operator $\left(\mathcal{L}_{d,fs} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \boldsymbol{\Phi}_0\right)_k$ for all coarse cells, including at the boundaries (which incorporate the diffusion boundary conditions).

We choose the weighting function $\boldsymbol{f}$ to be a one-vector, defined as

$$\boldsymbol{1} \equiv \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} , \quad \boldsymbol{1} \in \mathbb{R}^{J \times 1} \ . \tag{4.27}$$

This choice causes the coarse-grid cross sections to be volume-weighted (not flux- and volume-weighted, as with CMFD), and is made to ensure positivity of the mesh quantities contained in the diffusion operator. After adding the term $\left(\mathcal{L}_{d,fs}\langle\!\langle \mathbf{1} \rangle\!\rangle \boldsymbol{\Phi}_0\right)_k$ to both sides of Eq. (4.10a) and rearranging, we obtain *the operator form of the coarse-mesh, fixed-source LDA equation* shown below:

$$\begin{aligned}\left(\mathcal{L}_{d,fs}\langle\!\langle \mathbf{1} \rangle\!\rangle \boldsymbol{\Phi}_0\right)_k &= \bar{Q}_k\Delta_k + \left(\Phi_{1,k-\frac{1}{2}} - \Phi_{1,k+\frac{1}{2}}\right) \\ &+ \left(\mathcal{L}_{d,fs}\langle\!\langle \mathbf{1} \rangle\!\rangle \boldsymbol{\Phi}_0\right)_k - \bar{\Sigma}_{a,k}\langle\!\langle \boldsymbol{\phi}_0 \rangle\!\rangle \Phi_{0,k}\Delta_k \ , \\ &\{k \mid k \in \mathbb{N}, 1 \leq k \leq K\} \ ,\end{aligned} \tag{4.28}$$

where the operator $\left(\mathcal{L}_{d,fs}\langle\!\langle \mathbf{1} \rangle\!\rangle \boldsymbol{\Phi}_0\right)_k$ is defined for all coarse cells in Eqs. (4.12). The diffusion boundary conditions, shown in Eqs. (4.18) and (4.24), are incorporated into the definitions of this operator at the boundaries of the system, yielding a system of K equations. We recognize that one of the RHS terms of Eq. (4.28) can be simplified into a linear form, as shown below:

$$\bar{\Sigma}_{a,k}\langle\!\langle \boldsymbol{\phi}_0 \rangle\!\rangle \Phi_{0,k}\Delta_k = \frac{\sum\limits_{j\in k}\Sigma_{a,j}\phi_{0,j}h_j}{\cancel{\sum\limits_{j\in k}\phi_{0,j}h_j}} \cdot \frac{\cancel{\sum\limits_{j\in k}\phi_{0,j}h_j}}{\cancel{\Delta_k}} \cdot \cancel{\Delta_k} \tag{4.29}$$

$$= \sum_{j\in k}\Sigma_{a,j}\phi_{0,j}h_j \tag{4.30}$$

$$\equiv \dot{\Sigma}_{a,k}\langle\!\langle \boldsymbol{\phi}_0 \rangle\!\rangle \ , \tag{4.31}$$

where

$$\boldsymbol{\phi}_0 \equiv \begin{pmatrix} \phi_{0,1} \\ \phi_{0,2} \\ \vdots \\ \phi_{0,J} \end{pmatrix} , \quad \boldsymbol{\phi}_0 \in \mathbb{R}^{J\times 1} \ . \tag{4.32}$$

The term $\dot{\Sigma}_{a,k}\langle\!\langle \boldsymbol{\phi}_0 \rangle\!\rangle$ represents the total absorption rate in a coarse cell, and is linear with respect to $\boldsymbol{\phi}_0$.

To finalize the derivation of the low-order, fixed-source LDA equation, iteration superscripts are added. The LHS quantities are denoted as the solution of the LDA equations, and the RHS quantities are constructed from the most recent estimate of the scalar flux. In this formulation, the most recent estimate of the scalar flux $\boldsymbol{\Phi}_0^{(l+\frac{1}{2})}$ is computed as the homogenized solution of the transport equation, before the LDA equation is solved for $\boldsymbol{\Phi}_0^{(l+1)}$:

$$\mathcal{L}_{d,fs}\langle\!\langle \mathbf{1} \rangle\!\rangle \boldsymbol{\Phi}_0^{(l+1)} = \tilde{\boldsymbol{Q}}^{(l+\frac{1}{2})} \ , \tag{4.33a}$$

$$\tilde{Q}_k^{(l+\frac{1}{2})} \equiv \bar{Q}_k\Delta_k + \left(\Phi^{(l+\frac{1}{2})}_{1,k-\frac{1}{2}} - \Phi^{(l+\frac{1}{2})}_{1,k+\frac{1}{2}}\right) + \left(\mathcal{L}_{d,fs}\left\langle\!\left\langle \mathbf{1}\right\rangle\!\right\rangle \mathbf{\Phi}_0^{(l+\frac{1}{2})}\right)_k - \dot{\Sigma}_{a,k}\left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})}\right\rangle\!\!\right\rangle \ , \tag{4.33b}$$

$$\Phi^{(l+\frac{1}{2})}_{0,k} \equiv \frac{\sum\limits_{j\in k} \phi^{(l+\frac{1}{2})}_{0,j} h_j}{\Delta_k} \ , \tag{4.33c}$$

$$\{k \mid k \in \mathbb{N}, 1 \le k \le K\} \ .$$

Equation (4.33a) represents a system of K equations with K unknowns $\left(\mathbf{\Phi}_0^{(l+1)}\right)$. Practically, the operator $\mathcal{L}_{d,fs}\left\langle\!\left\langle \mathbf{1}\right\rangle\!\right\rangle$ takes the form of a $K \times K$ matrix that acts on the $K \times 1$ vector $\mathbf{\Phi}_0^{(l+1)}$. The RHS of Eq. (4.33a) is a $K \times 1$ vector as well, with elements corresponding to a given coarse cell k defined in Eq. (4.33b).

To check that the solution of Eq. (4.33a) results in the coarse-mesh transport solution, we take the limit as $l \to \infty$. First, we consider the following form of Eq. (4.33a) with $\tilde{Q}^{(l+\frac{1}{2})}$ expanded (as defined in Eq. (4.33b)):

$$\begin{aligned}\left(\mathcal{L}_{d,fs}\left\langle\!\left\langle \mathbf{1}\right\rangle\!\right\rangle \mathbf{\Phi}_0^{(l+1)}\right)_k &= \bar{Q}_k\Delta_k + \left(\Phi^{(l+\frac{1}{2})}_{1,k-\frac{1}{2}} - \Phi^{(l+\frac{1}{2})}_{1,k+\frac{1}{2}}\right) \\ &+ \left(\mathcal{L}_{d,fs}\left\langle\!\left\langle \mathbf{1}\right\rangle\!\right\rangle \mathbf{\Phi}_0^{(l+\frac{1}{2})}\right)_k - \dot{\Sigma}_{a,k}\left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})}\right\rangle\!\!\right\rangle \ .\end{aligned} \tag{4.34}$$

Now, we take the limit as $l \to \infty$ and cancel the identical diffusion operators on each side of the equation:

$$\begin{aligned}\cancel{\left(\mathcal{L}_{d,fs}\left\langle\!\left\langle \mathbf{1}\right\rangle\!\right\rangle \mathbf{\Phi}_0^{(\infty)}\right)_k} &= \bar{Q}_k\Delta_k + \left(\Phi^{(\infty)}_{1,k-\frac{1}{2}} - \Phi^{(\infty)}_{1,k+\frac{1}{2}}\right) \\ &+ \cancel{\left(\mathcal{L}_{d,fs}\left\langle\!\left\langle \mathbf{1}\right\rangle\!\right\rangle \mathbf{\Phi}_0^{(\infty)}\right)_k} - \dot{\Sigma}_{a,k}\left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(\infty)}\right\rangle\!\!\right\rangle \ .\end{aligned} \tag{4.35}$$

Upon rearranging, we obtain the following converged neutron balance equation:

$$\left(\Phi^{(\infty)}_{1,k+\frac{1}{2}} - \Phi^{(\infty)}_{1,k-\frac{1}{2}}\right) + \dot{\Sigma}_{a,k}\left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(\infty)}\right\rangle\!\!\right\rangle = \bar{Q}_k\Delta_k \ , \tag{4.36}$$

which is the same as Eq. (4.10a) (derived directly from the NTE). We note the following equivalence:

$$\bar{\Sigma}_{a,k}\left\langle\!\left\langle \boldsymbol{\phi}_0\right\rangle\!\right\rangle \Phi_{0,k}\Delta_k = \dot{\Sigma}_{a,k}\left\langle\!\left\langle \boldsymbol{\phi}_0\right\rangle\!\right\rangle \ , \tag{4.37}$$

as shown in Eq. (4.29). Thus, we have shown that the solution of the LDA equation limits to the coarse-mesh transport equation upon convergence.

If we use Eq. (4.12a) to expand the $\tilde{Q}_k^{(l+\frac{1}{2})}$ operator and rearrange, we obtain:

$$\begin{gathered}
\tilde{Q}_k^{(l+\frac{1}{2})} = \bar{Q}_k \Delta_k \\
+\left(\Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2})} - \Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})}\right) - \tilde{D}_{k+\frac{1}{2}} \langle\!\langle \mathbf{1} \rangle\!\rangle \left(\Phi_{0,k+1}^{(l+\frac{1}{2})} - \Phi_{0,k}^{(l+\frac{1}{2})}\right) + \tilde{D}_{k-\frac{1}{2}} \langle\!\langle \mathbf{1} \rangle\!\rangle \left(\Phi_{0,k}^{(l+\frac{1}{2})} - \Phi_{0,k-1}^{(l+\frac{1}{2})}\right) \\
+\bar{\Sigma}_{a,k} \langle\!\langle \mathbf{1} \rangle\!\rangle \Phi_{0,k}^{(l+\frac{1}{2})} \Delta_k - \dot{\Sigma}_{a,k} \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle ,
\end{gathered} \tag{4.38}$$

where the second line of the equation represents the error in Fick's Law for approximating neutron streaming in the operator $\left(\mathcal{L}_{d,fs} \langle\!\langle \mathbf{1} \rangle\!\rangle \mathbf{\Phi}_0\right)_k$. This term is analogous to $\hat{D}$ in the CMFD method, which also accounts for the error introduced by Fick's Law. However, an additional error is present in $\left(\mathcal{L}_{d,fs} \langle\!\langle \mathbf{1} \rangle\!\rangle \mathbf{\Phi}_0\right)_k$ due to the use of volume-weighted cross sections (as opposed to flux- and volume-weighted cross sections). The third line in Eq. (4.38) accounts for this error. We expect the terms in the second and third lines to be small, and therefore, stable throughout the iteration process. This is reinforced by the linearity of Eq. (4.38), which prevents numerical instability as a result of nonlinear terms. None of the terms in Eqs. (4.33) are nonlinear with respect to the scalar flux, which is the primary goal and motivation for the construction of LDA.

Once the low-order solution $\mathbf{\Phi}_0^{(l+1)}$ is obtained by solving Eq. (4.33a), a prolongation step is performed to update the fine-mesh solution $\boldsymbol{\phi}_0^{(l+\frac{1}{2})}$ with information from the low-order solution. With CMFD, this is typically done through multiplicative prolongation as shown below:

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{(l+\frac{1}{2})} \left(\frac{\Phi_{0,k}^{(l+1)}}{\Phi_{0,k}^{(l+\frac{1}{2})}} \right) , \forall\ j \in k\ . \tag{Eq. (3.8q) revisited}$$

To linearize this step, a simple equation can be constructed in which we wish to obtain the additive update term $c_k^{(l+1)}$ that provides an update that is equivalent to the multiplicative term:

$$\phi_{0,j}^{(l+\frac{1}{2})} + c_k^{(l+1)} = \phi_{0,j}^{(l+\frac{1}{2})} \left(\frac{\Phi_{0,k}^{(l+1)}}{\Phi_{0,k}^{(l+\frac{1}{2})}} \right) , \tag{4.39}$$

where $c_k^{(l+1)}$ is constant in a coarse cell. After volume averaging over a coarse cell and solving for $c_k^{(l+1)}$, the following result is obtained:

$$c_k^{(l+1)} = \Phi_{0,k}^{(l+1)} - \Phi_{0,k}^{(l+\frac{1}{2})} . \tag{4.40}$$

With this term defined, an additive prolongation equation may be developed as shown below. This is a linear form of Eq. (3.8q):

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{(l+\frac{1}{2})} + \left(\Phi_{0,k}^{(l+1)} - \Phi_{0,k}^{(l+\frac{1}{2})}\right) \ , \forall \ j \in k \ . \tag{4.41}$$

At this point, the next iterate of the scalar flux $\phi_0^{(l+1)}$ has been determined, and the iteration process can proceed until sufficient convergence is achieved. It should be noted, once again, that *all terms in the LDA formulation are linear with respect to the solution vector*.

4.3.2 Eigenvalue Problems

In this section, we have the same aim as the previous section: to derive the LDA equations from the transport equation – except we now focus on the eigenvalue form instead of the fixed-source form. We begin again with a continuous, monoenergetic, steady-state, 1D NTE, now with vacuum boundary conditions and an eigenvalue to represent multiplying media:

$$\begin{gathered}\mu\frac{\partial}{\partial x}\psi(x,\mu) + \Sigma_t(x)\psi(x,\mu) \\ = \frac{1}{2}\left(\int_{-1}^{1}\Sigma_s(x,\mu,\mu')\psi(x,\mu')\,\mathrm{d}\mu' + \lambda_t\nu\Sigma_f(x)\int_{-1}^{1}\psi(x,\mu')\,\mathrm{d}\mu'\right) \ , \\ 0 \le x \le X \ , \quad -1 \le \mu \le 1 \ ,\end{gathered} \tag{4.42a}$$

$$\psi(0,\mu) = 0 \ , \quad \mu > 0 \ , \tag{4.42b}$$

$$\psi(X,\mu) = 0 \ , \quad \mu < 0 \ , \tag{4.42c}$$

where λ_t is the fundamental transport eigenvalue. We choose this notation to distinguish the fundamental transport eigenvalue from the fundamental diffusion eigenvalue λ_d, which will become relevant later in this derivation. To select a unique solution to Eqs. (4.42), a normalization condition must be imposed. As an example, we enforce the following condition:

$$P = \int_0^X \nu\Sigma_f(x)\phi(x)\,\mathrm{d}x \ , \tag{4.42d}$$

where the scalar flux is defined as

$$\phi(x) \equiv \int_{-1}^{1}\psi(x,\mu)\,\mathrm{d}\mu \ , \tag{4.42e}$$

and P is the neutron production rate for the system.

First, we operate on Eq. (4.42a) by $\int_{-1}^{1}(\cdot)\,\mathrm{d}\mu$ and on Eqs. (4.42b) and (4.42c) by

$\int_{-1}^{1} \mu(\cdot)\, \mathrm{d}\mu$ to obtain:

$$\frac{\mathrm{d}}{\mathrm{d}x}\phi_1(x) + \Sigma_a(x)\phi_0(x) = \lambda_t \nu\Sigma_f(x)\phi_0(x) \ , \tag{4.43a}$$

$$\phi_1(0) = \int_{-1}^{0} \mu\psi(0,\mu)\,\mathrm{d}\mu \ , \tag{4.43b}$$

$$\phi_1(X) = \int_{0}^{1} \mu\psi(X,\mu)\,\mathrm{d}\mu \ . \tag{4.43c}$$

Then we discretize Eq. (4.43a) in space by averaging over a given fine cell. This is done by operating on Eqs. (4.43) by $\frac{1}{h_j}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} (\cdot)\,\mathrm{d}x$, resulting in the following discrete form:

$$\begin{gathered}\left(\phi_{1,j+\frac{1}{2}} - \phi_{1,j-\frac{1}{2}}\right) + \Sigma_{a,j}\phi_{0,j}h_j = \lambda_t \left(\nu\Sigma_f\right)_j h_j \ , \\ \{j \mid j \in \mathbb{N}, 1 \le j \le J\} \ , \end{gathered} \tag{4.44a}$$

$$\phi_{1,\frac{1}{2}} = \int_{-1}^{0} \mu\psi_{\frac{1}{2}}(\mu)\,\mathrm{d}\mu \ , \tag{4.44b}$$

$$\phi_{1,J+\frac{1}{2}} = \int_{0}^{1} \mu\psi_{J+\frac{1}{2}}(\mu)\,\mathrm{d}\mu \ , \tag{4.44c}$$

where fine mesh quantities are assumed to be constant within a fine cell (just as in the fixed-source derivation).

Now, Eq. (4.44a) is integrated over all fine cells j in a given contiguous coarse cell k by operating by $\sum_{j\in k}(\cdot)$, resulting in the following:

$$\left(\Phi_{1,k+\frac{1}{2}} - \Phi_{1,k-\frac{1}{2}}\right) + \bar{\Sigma}_{a,k}\left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \Phi_{0,k}\Delta_k = \lambda_t \overline{\left(\nu\Sigma_f\right)}_k \left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \Phi_{0,k}\Delta_k \ , \tag{4.45a}$$

$$\bar{\Sigma}_{a,k}\left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \equiv \frac{\sum\limits_{j\in k} \Sigma_{a,j}\phi_{0,j}h_j}{\sum\limits_{j\in k} \phi_{0,j}h_j} \ , \tag{4.45b}$$

$$\overline{\left(\nu\Sigma_f\right)}_k \left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \equiv \frac{\sum\limits_{j\in k} \left(\nu\Sigma_f\right)_j \phi_{0,j}h_j}{\sum\limits_{j\in k} \phi_{0,j}h_j} \ . \tag{4.45c}$$

Before continuing, we first consider the uniformly-weighted *critical* diffusion operator $\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ (with the subscripts d for diffusion and eig for eigenvalue) acting on the scalar flux $\mathbf{\Phi}_0$:

$$\begin{gathered}\left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0\right)_k \equiv -\tilde{D}_{k+\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \left(\Phi_{0,k+1} - \Phi_{0,k}\right) + \tilde{D}_{k-\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \left(\Phi_{0,k} - \Phi_{0,k-1}\right) \\ + \left(\bar{\Sigma}_{a,k} - \lambda_d \overline{\left(\nu\Sigma_f\right)}_k\right)\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \Phi_{0,k}\Delta_k \ , \end{gathered} \tag{4.46a}$$

$$\tilde{D}_{k\pm\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\Delta_k + \bar{\Sigma}_{t,k\pm1}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\Delta_{k\pm1}\right)} \ , \qquad \{k \mid k \in \mathbb{N}, 2 \le k \le K-1\} \ , \tag{4.46b}$$

which holds at the interior coarse cells. For the boundary cells, we make the following definitions of the critical diffusion operator:

$$\begin{aligned}\left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_0\right)_1 \equiv & -\tilde{D}_{\frac{3}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\left(\Phi_{0,2} - \Phi_{0,1}\right) - \Phi_{1,\frac{1}{2},d} \\ & + \left(\bar{\Sigma}_{a,1} - \lambda_d\overline{(\nu\Sigma_f)}_1\right)\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\Phi_{0,1}\Delta_1 \ ,\end{aligned} \tag{4.46c}$$

$$\begin{aligned}\left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_0\right)_K \equiv & \ \Phi_{1,K+\frac{1}{2},d} + \tilde{D}_{K-\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\left(\Phi_{0,K} - \Phi_{0,K-1}\right) \\ & + \left(\bar{\Sigma}_{a,K} - \lambda_d\overline{(\nu\Sigma_f)}_K\right)\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\Phi_{0,K}\Delta_K \ ,\end{aligned} \tag{4.46d}$$

where $\Phi_{1,\frac{1}{2},d}$ and $\Phi_{1,K+\frac{1}{2},d}$ are "diffusion current" terms that are determined by the diffusion boundary conditions (as in Section 4.3.1). Here λ_d is the fundamental eigenvalue of the following neutron diffusion equation:

$$\begin{aligned}& -\tilde{D}_{k+\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\left(\Phi_{0,d,k+1} - \Phi_{0,d,k}\right) + \tilde{D}_{k-\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\left(\Phi_{0,d,k} - \Phi_{0,d,k-1}\right) \\ & + \bar{\Sigma}_{a,k}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\Phi_{0,d,k}\Delta_k = \lambda_d\overline{(\nu\Sigma_f)}_k\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle\Phi_{0,d,k}\Delta_k \ ,\end{aligned} \tag{4.47}$$

with the following boundary conditions:

$$0 = \Phi_{1,\frac{1}{2},d} + \frac{1}{2}\Phi_{0,\frac{1}{2}} \ , \tag{4.48}$$

$$0 = -\Phi_{1,K+\frac{1}{2},d} + \frac{1}{2}\Phi_{0,K+\frac{1}{2}} \ . \tag{4.49}$$

(The boundary conditions were derived in Section 4.3.1, and the prescribed incoming neutron current is zero for eigenvalue problems.) The operator $\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ defined in Eq. (4.46a) is obtained by rearranging Eq. (4.47). When the weighting vector $\mathbf{1}$ is used, the coarse mesh quantities are only volume-weighted over each coarse spatial cell.

We note that the fundamental eigenpair of Eq. (4.47) is different than the fundamental eigenpair of the NTE. Therefore, the subscript d is used for the solutions to this equation in order to distinguish the two solutions. The $\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ operator is referred to as *critical* because the following equation holds:

$$\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d} = 0 \ . \tag{4.50}$$

Therefore, the $\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ operator represents a critical diffusion system. The eigenvalue λ_d

in this operator is the fundamental eigenvalue corresponding to the fundamental eigenvector $\mathbf{\Phi}_{0,d}$. Additionally, we note that $\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ is singular and non-invertable.

Equation (4.50) is an eigenvalue problem, and a normalization condition must be imposed to select a unique solution. We impose the following normalization condition on the neutron production rate that corresponds to the condition stated in Eq. (4.42d):

$$P = \sum_{k=1}^{K} \left(\nu\Sigma_f\right)_k \Phi_{0,d,k}\Delta_k \ , \tag{4.51}$$

where P is the desired neutron production rate.

Next, we add $\left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0\right)_k$ to both sides of Eq. (4.45a) and rearrange. The following LDA equation is obtained:

$$\begin{aligned}\left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0\right)_k &= \left(\Phi_{1,k-\frac{1}{2}} - \Phi_{1,k+\frac{1}{2}}\right) \\ &+ \left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0\right)_k + \left(\lambda_t (\nu\dot{\Sigma}_f)_k - \dot{\Sigma}_{a,k}\right)\left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \ ,\end{aligned} \tag{4.52a}$$

$$\dot{\Sigma}_{a,k}\left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \equiv \sum_{j\in k} \Sigma_{a,j}\phi_{0,j}h_j \ , \tag{4.52b}$$

$$(\nu\dot{\Sigma}_f)_k\left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \equiv \sum_{j\in k} \left(\nu\Sigma_f\right)_j \phi_{0,j}h_j \ , \tag{4.52c}$$

where the diffusion boundary conditions are incorporated into the definition of $\left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0\right)_k$ (in Eqs. (4.46)) at the edges of the problem. We note that the mesh quantities have been linearized in the same way as in Section 4.3.1. Finally, iteration superscripts are added in a similar fashion as the fixed-source problem:

$$\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{(l+1)} = \tilde{\boldsymbol{Q}}^{(l+1)} \ , \tag{4.53a}$$

$$\begin{aligned}\tilde{Q}_k^{(l+1)} &\equiv \left(\Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2})} - \Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})}\right) + \left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{(l+\frac{1}{2})}\right)_k \\ &+ \left(\lambda_t^{(l+1)} (\nu\dot{\Sigma}_f)_k - \dot{\Sigma}_{a,k}\right)\left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle \ .\end{aligned} \tag{4.53b}$$

Just as in Section 4.3.1, we take the limit as $l \to \infty$ to check that the LDA equation shown in Eq. (4.53a) produces the coarse-mesh transport solution. First, we expand Eq. (4.53a):

$$\begin{aligned}\left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{(l+1)}\right)_k &= \left(\Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2})} - \Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})}\right) + \left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{(l+\frac{1}{2})}\right)_k \\ &+ \left(\lambda_t^{(l+1)} (\nu\dot{\Sigma}_f)_k - \dot{\Sigma}_{a,k}\right)\left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle \ .\end{aligned} \tag{4.54}$$

Taking the limit as $l \to \infty$ yields the following, with the diffusion operators cancelling on each side of the equation:

$$\begin{aligned} \cancel{\left(\mathcal{L}_{d,eig} \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{(\infty)}\right)_k} &= \left(\Phi^{(\infty)}_{1,k-\frac{1}{2}} - \Phi^{(\infty)}_{1,k+\frac{1}{2}}\right) + \cancel{\left(\mathcal{L}_{d,eig} \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{(\infty)}\right)_k} \\ &+ \left(\lambda_t^{(\infty)} (\nu\dot{\Sigma}_f)_k - \dot{\Sigma}_{a,k}\right) \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(\infty)} \right\rangle\!\!\right\rangle \ . \end{aligned} \tag{4.55}$$

Rearranging yields the following equivalent form of the NBE shown in Eq. (4.45a):

$$\left(\Phi^{(\infty)}_{1,k+\frac{1}{2}} - \Phi^{(\infty)}_{1,k-\frac{1}{2}}\right) + \dot{\Sigma}_{a,k} \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(\infty)} \right\rangle\!\!\right\rangle = \lambda_t^{(\infty)} (\nu\dot{\Sigma}_f)_k \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(\infty)} \right\rangle\!\!\right\rangle \ . \tag{4.56}$$

Thus, we conclude that the solution of the LDA equation is the same as the solution of the NBE upon convergence.

The term $\tilde{Q}$ contains correction terms that are analogous to $\hat{D}$ in CMFD. Specifically, the first line of Eq. (4.53b) acts as the linear transport correction to Fick's Law. The operator $\mathcal{L}_{d,eig} \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ contains numerical representations of Fick's Law, and the lagged transport current difference accounts for this error. Similarly, the last line of Eq. (4.53b) accounts for the use of volume-weighted mesh quantities in the $\mathcal{L}_{d,eig} \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ operator. With these lagged transport corrections, which we assume to be small in much the same way that $\hat{D}$ is assumed to be small in CMFD, the LDA solution is made consistent with the solution of the NTE.

As previously discussed, the LHS operator of Eq. (4.53a) is singular and non-invertable. However, according to the FAT, this equation has a solution if and only if the solvability condition shown below is satisfied [61]:

$$\left\langle \mathbf{\Phi}^*_{0,d}, \tilde{\boldsymbol{Q}}^{(l+1)} \right\rangle \equiv \sum_{k=1}^{K} \Phi^*_{0,d,k} \tilde{Q}_k^{(l+1)} = 0 \ , \tag{4.57}$$

where the term $\mathbf{\Phi}^*_{0,d}$ is the fundamental eigenvector of the adjoint diffusion equation defined below:

$$\mathcal{L}^*_{d,eig} \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}^*_{0,d} = 0 \ . \tag{4.58}$$

Thus, $\Phi^*_{0,d}$ must be obtained before a solution to Eq. (4.53a) can be found. In these equations, the $*$ superscript on a matrix refers to the adjoint (or conjugate transpose) of that matrix. For square matrices containing real numbers (which is the case here), the adjoint is the same as the transpose.

By expanding and rearranging Eq. (4.57), the solvability condition provides the definition for $\lambda_t^{(l+1)}$, shown below:

$$\lambda_t^{(l+1)} = \frac{\sum\limits_{k=1}^{K} \Phi_{0,d,k}^{*} \left[\left(\Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})} - \Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2})} \right) + \dot{\Sigma}_{a,k} \left\langle\!\!\left\langle \phi_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle - \left(\mathcal{L}_{d,eig} \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \Phi_0^{(l+\frac{1}{2})} \right)_k \right]}{\sum\limits_{k=1}^{K} \Phi_{0,d,k}^{*} (\nu\dot{\Sigma}_f)_k \left\langle\!\!\left\langle \phi_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle} \,. \tag{4.59}$$

Once $\lambda_t^{(l+1)}$ is computed, $\tilde{\boldsymbol{Q}}^{(l+1)}$ can be computed and the solution to Eq. (4.53a) can be obtained using PI. In practice, a modified form of Eq. (4.53a) is solved to obtain the low-order solution, as described in Section 4.4.2. The final step to determine $\phi_0^{(l+1)}$ is additive prolongation, which was derived in Section 4.3.1.

The FAT states that if a solution to Eq. (4.53a) exists, $\lambda_t^{(l+1)}$ must be defined as in Eq. (4.59) (which satisfies the solvability condition). However, the method for obtaining this solution is not provided by the FAT. We will make some changes to allow Eq. (4.53a) to be solved practically. Further, the FAT states that an infinite number of solutions to Eq. (4.53a) exist of the following form:

$$\boldsymbol{\Phi}_0 = \tilde{\boldsymbol{\Phi}}_0 + \alpha \boldsymbol{\Phi}_{0,d} \,, \tag{4.60}$$

where the particular solution, $\tilde{\boldsymbol{\Phi}}_0$, satisfies the following solvability condition:

$$\left\langle \boldsymbol{\Phi}_{0,d}^{*}, \tilde{\boldsymbol{\Phi}}_0 \right\rangle = 0 \,, \tag{4.61}$$

and α is an arbitrary multiplicative constant. To obtain a unique solution, the value of α is computed such that the resulting unique solution satisfies the imposed normalization condition. The process of solving Eq. (4.53a) in practice and determining α are described in Section 4.4.2. However, in practice, this specific normalization procedure can be modified because the actual linear system to be solved differs from Eq. (4.53a).

4.4 Algorithm Overview

This section describes how LDA is used to accelerate both fixed-source and eigenvalue transport problems. Specifically, we describe the sequence of calculations by which the equations derived in the previous section are evaluated. We note that the method used to calculate the high-order solution is arbitrary as long as the required quantities are obtained and the solution is kept consistent with the low-order solution. In these equations, the outer iteration index is l and the power iteration index for eigenvalue problems is m.

4.4.1 Fixed-Source Problems

The following description of the LDA method applies to the steady-state, fixed-source, monoenergetic, planar problem described by Eqs. (4.7). This algorithm can be extended in a straightforward manner to multidimensional, multigroup problems.

0. **Setup**: Before the iterative process begins, we first construct the coarse-grid operator $\mathcal{L}_{d,fs} \langle\!\langle \mathbf{1} \rangle\!\rangle$ as defined in Eqs. (4.12):

$$\begin{aligned}\left(\mathcal{L}_{d,fs} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \boldsymbol{\Phi}_0\right)_k \equiv -\tilde{D}_{k+\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,k+1} - \Phi_{0,k}\right) + \tilde{D}_{k-\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,k} - \Phi_{0,k-1}\right) \\ +\bar{\Sigma}_{a,k} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Phi_{0,k} \Delta_k \ ,\end{aligned}$$

(Eq. (4.12a) revisited)

which is reproduced here for the problem interior. This operator is fixed throughout the outer iteration process and serves as the LHS operator for the low-order LDA system. Thus, this operator can optionally be factorized (using LU decomposition or other method) at this point to reduce the computational cost of performing the low-order solve in step 3. Additionally, an initial guess for the fine-mesh scalar flux is made. For example, a one-vector can be chosen:

$$\boldsymbol{\phi}_0^{(0)} = \mathbf{1} \ . \tag{4.62}$$

1. **Transport Sweep**: A discrete solution $\boldsymbol{\phi}_0^{(l+\frac{1}{2})}$ to the discrete form of Eq. (4.7a) is obtained on the fine mesh by performing a transport sweep. (The discrete form of Eq. (4.7a) is shown in Eq. (3.8b) with iteration superscripts.) During this process, the current at the coarse mesh edges $\left(\Phi_{1,k\pm\frac{1}{2}}^{(l+\frac{1}{2})} \ \forall \ k\right)$ is also stored.
2. **Restriction**: The next step is restriction, in which the low-order coarse mesh quantities are computed from the fine-mesh quantities. We compute $\boldsymbol{\Phi}_0^{(l+\frac{1}{2})}$ as defined in Eq. (4.33c):

$$\Phi_{0,k}^{(l+\frac{1}{2})} \equiv \frac{\sum\limits_{j\in k} \phi_{0,j}^{(l+\frac{1}{2})} h_j}{\Delta_k} \ , \qquad \text{(Eq. (4.33c) revisited)}$$

and $\tilde{\boldsymbol{Q}}^{(l+\frac{1}{2})}$ as defined in Eq. (4.33b):

$$\begin{aligned}\tilde{Q}_k^{(l+\frac{1}{2})} \equiv \bar{Q}_k \Delta_k + \left(\Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2})} - \Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})}\right) \\ + \left(\mathcal{L}_{d,fs} \langle\!\langle \mathbf{1} \rangle\!\rangle \boldsymbol{\Phi}_0^{(l+\frac{1}{2})}\right)_k - \dot{\Sigma}_{a,k} \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle \ .\end{aligned} \qquad \text{(Eq. (4.33b) revisited)}$$

3. **Low-Order Solve**: Now, the solution, $\boldsymbol{\Phi}_0^{(l+1)}$, to Eq. (4.33a) is obtained by solving

the LDA system:

$$\mathcal{L}_{d,fs}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_0^{(l+1)} = \tilde{\boldsymbol{Q}}^{(l+\frac{1}{2})} \, . \qquad \text{(Eq. (4.33a) revisited)}$$

4. **Prolongation**: Finally, the prolongation step is performed in which the fine mesh solution is updated with information from the low-order solve. This is done using Eq. (4.41) to obtain $\boldsymbol{\phi}_0^{(l+1)}$:

   $$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{(l+\frac{1}{2})} + \left(\Phi_{0,k}^{(l+1)} - \Phi_{0,k}^{(l+\frac{1}{2})}\right) \, , \forall \; j \in k \, . \qquad \text{(Eq. (4.41) revisited)}$$

   If sufficient convergence of the quantities of interest is not achieved, then the next outer iteration is begun (at step 1).

The outer iteration process for LDA is summarized in Algorithm 7.

Algorithm 7 LDA Applied to Fixed-Source Problems

```
Construct $\mathcal{L}_{d,fs}\langle\langle \mathbf{1} \rangle\rangle$ as defined in Eq. (4.12a) and assume an initial guess for $\boldsymbol{\phi}_0$
repeat
    Calculate $\boldsymbol{\phi}_0^{(l+\frac{1}{2})}$ and $\Phi_{1,k\pm\frac{1}{2}}^{(l+\frac{1}{2})}$ $\forall\ k$ by solving the discrete form of Eq. (4.7a)
    Calculate $\boldsymbol{\Phi}_0^{(l+\frac{1}{2})}$ using Eq. (4.33c) and construct $\tilde{\boldsymbol{Q}}_f^{(l+\frac{1}{2})}$ using Eq. (4.33b)
    Calculate $\boldsymbol{\Phi}_0^{(l+1)}$ by solving Eq. (4.33a)
    Calculate $\boldsymbol{\phi}_0^{(l+1)}$ using Eq. (4.41)
until convergence        {generally, convergence of $\boldsymbol{\phi}_0$ is checked}
```

4.4.2 Eigenvalue Problems

The following description of the LDA method applies to the monoenergetic, planar, eigenvalue problem described by Eqs. (4.42).

0. **Setup**: This step is performed only once, before the outer iteration process begins. To satisfy the solvability condition imposed by the FAT when solving the low-order LDA equations for eigenvalue problems, we must first obtain the forward $\boldsymbol{\Phi}_{0,d}$ and adjoint $\boldsymbol{\Phi}_{0,d}^*$ solutions of the neutron diffusion equation defined in Eq. (4.46a). Because these are eigenvalue problems, a method such as PI must be used to obtain a converged solution. The fundamental eigenvalue λ_d is the same for both of these solutions, and must also be recorded during this process.

Here we outline the solution of the diffusion equation using PI. We note that the spatial mesh of this equation is the same as the mesh for the LDA equations. The following set of equations (reproduced here for the problem interior, with the boundary operators defined in Eqs. (4.46)) is solved iteratively until the forward eigenfunction is converged:

$$\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}^{(m+1)} = \lambda_d^{(m)} \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}^{(m)} \ , \tag{4.63a}$$

$$\begin{aligned} \left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}^{(m+1)}\right)_k &\equiv -\tilde{D}_{k+\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \left(\Phi_{0,d,k+1} - \Phi_{0,d,k}\right) \\ &+\tilde{D}_{k-\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \left(\Phi_{0,d,k} - \Phi_{0,d,k-1}\right) + \bar{\Sigma}_{a,k}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \Phi_{0,d,k}\Delta_k \ , \end{aligned} \tag{4.63b}$$

$$\left(\mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}\right)_k \equiv \overline{\left(\nu\Sigma_f\right)_k} \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \Phi_{0,d,k}\Delta_k \ , \tag{4.63c}$$

$$\lambda_d^{(m)} = \frac{\left\langle \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}^{(m)}, \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}^{(m)} \right\rangle}{\left\langle \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}^{(m+1)}, \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}^{(m)} \right\rangle} \ , \tag{4.63d}$$

$$P = \left\langle \mathbf{1}, \mathbb{F}_d \boldsymbol{\Phi}_{0,d}^{(m+1)} \right\rangle \ , \tag{4.63e}$$

where P is the chosen total neutron production rate. The normalization condition in Eq. (4.63e) can be arbitrarily chosen. The same PI procedure is repeated using $\mathcal{L}_d^* \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ and $\mathbb{F}_d^* \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ to obtain $\boldsymbol{\Phi}_{0,d}^*$ by solving the following system:

$$\mathcal{L}_{d,eig}^* \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}^{*,(m+1)} = \lambda_d \mathbb{F}_d^* \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \boldsymbol{\Phi}_{0,d}^{*,(m)} \ , \tag{4.64}$$

where the converged λ_d from the forward diffusion problem is used. At the end of this step, the following quantities have been obtained and stored: $\boldsymbol{\Phi}_{0,d}$, $\boldsymbol{\Phi}_{0,d}^*$, and λ_d.

Additionally, an initial guess for the fine-mesh scalar flux and transport eigenvalue are made. Optionally, these can be set to the solutions of the forward diffusion equation:

$$\phi_{0,j}^{(0)} = \Phi_{0,d,k} \ \forall \ j \in k \ \forall \ k \ , \tag{4.65a}$$

$$\lambda_t^{(0)} = \lambda_d \ , \tag{4.65b}$$

as the solution of the diffusion equation is most likely closer to the converged transport solution than a flat guess.

1. **Transport Sweep**: The fine mesh solution $\phi_0^{(l+\frac{1}{2})}$ to the discrete form of Eqs. (4.42) is obtained by performing a transport sweep. During this step, the coarse mesh neutron current $\left(\Phi_{1,k\pm\frac{1}{2}}^{(l+\frac{1}{2})} \ \forall \ k\right)$ is also stored.

2. **Restriction**: In this restriction step, the following low-order mesh quantities are computed: $\mathbf{\Phi}_0^{(l+\frac{1}{2})}$ using Eq. (4.33c):

$$\Phi_{0,k}^{(l+\frac{1}{2})} \equiv \frac{\sum\limits_{j\in k} \phi_{0,j}^{(l+\frac{1}{2})} h_j}{\Delta_k} \ , \qquad \text{(Eq. (4.33c) revisited)}$$

$\dot{\Sigma}_{a,k} \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle$ using Eq. (4.52b):

$$\dot{\Sigma}_{a,k} \left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \equiv \sum_{j\in k} \Sigma_{a,j} \phi_{0,j} h_j \ , \qquad \text{(Eq. (4.52b) revisited)}$$

and $\left(\nu\dot{\Sigma}_f\right)_k \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle$ using Eq. (4.52c):

$$\left(\nu\dot{\Sigma}_f\right)_k \left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \equiv \sum_{j\in k} \left(\nu\Sigma_f\right)_j \phi_{0,j} h_j \ . \qquad \text{(Eq. (4.52c) revisited)}$$

3. **Eigenvalue Calculation**: The next iterate of the transport eigenvalue $\lambda_t^{(l+1)}$ is computed using Eq. (4.59):

$$\lambda_t^{(l+1)} = \frac{\sum\limits_{k=1}^{K} \Phi_{0,d,k}^{*} \left[\left(\Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})} - \Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2})} \right) + \dot{\Sigma}_{a,k} \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle - \left(\mathcal{L}_{d,eig} \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \Phi_0^{(l+\frac{1}{2})} \right)_k \right]}{\sum\limits_{k=1}^{K} \Phi_{0,d,k}^{*} \left(\nu\dot{\Sigma}_f\right)_k \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle} \ .$$

(Eq. (4.59) revisited)

4. **Low-Order Solve**: Now, the solution $\mathbf{\Phi}_0^{(l+1)}$ to the low-order LDA equations is computed. We have used the solvability condition of the FAT to obtain $\lambda_t^{(l+1)}$ such that a solution to Eq. (4.53a) exists.

   We choose the PI process to outline the path to a converged solution, although other approaches are possible. A modification is made to the LDA equations derived in Section 4.3.2. Specifically, the diffusion fission source is added to both sides of the system and lagged on the RHS in the PI process. This is done to avoid inverting a singular operator when finding the solution of the linear system. After this modification is made, the LDA equations are defined as:

$$\left(\mathcal{L}_{d,eig} + \lambda_d \mathbb{F}_d\right) \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})} = \tilde{\boldsymbol{Q}}^{(l+1)} + \lambda_d \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{(l+\frac{1}{2},m)} \ , \qquad (4.66a)$$

$$\begin{aligned}\left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0\right)_k \equiv &-\tilde{D}_{k+\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \left(\Phi_{0,k+1} - \Phi_{0,k}\right) + \tilde{D}_{k-\frac{1}{2}}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \left(\Phi_{0,k} - \Phi_{0,k-1}\right) \\ &+ \left(\bar{\bar{\Sigma}}_{a,k} - \lambda_d \overline{\left(\nu\Sigma_f\right)}_k\right)\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \Phi_{0,k}\Delta_k \ ,\end{aligned} \qquad \text{(Eq. (4.46a) revisited)}$$

$$\begin{aligned}\tilde{Q}_k^{(l+1)} \equiv &\left(\Phi_{1,k-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} - \Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\right) + \left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{\left(l+\frac{1}{2}\right)}\right)_k \\ &+ \left(\lambda_t^{(l+1)}\left(\nu\dot{\Sigma}_f\right)_k - \dot{\Sigma}_{a,k}\right)\left\langle\!\!\left\langle \boldsymbol{\phi}_0^{\left(l+\frac{1}{2}\right)} \right\rangle\!\!\right\rangle \ ,\end{aligned} \qquad \text{(Eq. (4.53b) revisited)}$$

$$\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+1\right)} = \mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} + \alpha^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\mathbf{\Phi}_{0,d} \ , \tag{4.66b}$$

$$\alpha^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} \equiv \frac{P - \left\langle \mathbf{1}, \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} \right\rangle}{\left\langle \mathbf{1}, \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_{0,d} \right\rangle} \ , \tag{4.66c}$$

where α is a multiplicative constant determined by the normalization condition stated in Eq. (4.63e). (The LHS operator shown here is for the problem interior, with the full definition shown in Eqs. (4.46).) Because an infinite number of solutions to Eq. (4.66a) exist, a particular solution that satisfies the chosen normalization condition can be selected through the choice of α. To derive Eq. (4.66c), we simply insert Eq. (4.66b) into Eq. (4.63e).

Upon sufficient convergence of the coarse mesh scalar flux after performing M power iterations, the LDA power iteration process concludes and we make the following update:

$$\mathbf{\Phi}_0^{(l+1)} = \mathbf{\Phi}_0^{\left(l+\frac{1}{2},M\right)} \ . \tag{4.67}$$

5. **Prolongation**: The last step is prolongation, in which the fine mesh scalar flux is updated with information from the LDA solution. We obtain $\boldsymbol{\phi}_0^{(l+1)}$ using Eq. (4.41):

   $$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{\left(l+\frac{1}{2}\right)} + \left(\Phi_{0,k}^{(l+1)} - \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}\right) \ , \forall \ j \in k \ . \qquad \text{(Eq. (4.41) revisited)}$$

   The fine mesh scalar flux is also normalized with the chosen normalization condition, such as the one defined below:

   $$P = \sum_j \left(\nu\Sigma_f\right)_j \phi_0^{(l+1)} h_j \ . \tag{4.68}$$

   (In practice, this additional normalization step may not be necessary if $\mathbf{\Phi}_0^{(l+1)}$ is properly normalized during PI.)

   If the quantities of interest are not converged, we return to step 1 and resume the outer iteration process.

The iterative algorithm for LDA applied to an eigenvalue problem is summarized in Algorithm 8.

Algorithm 8 LDA Applied to Eigenvalue Problems

```
Construct $(\mathcal{L}_{d,eig} + \lambda_d \mathbb{F}_d) \langle\!\langle \mathbf{1} \rangle\!\rangle$ as defined in Eqs. (4.46a) and (4.63c)
Calculate $\mathbf{\Phi}_{0,d}$ , $\mathbf{\Phi}^*_{0,d}$ , and $\lambda_d$ iteratively using Eqs. (4.63) and (4.64)
$\lambda_t^{(0)} \leftarrow \lambda_d$                                   {optional, but recommended}
$\phi_j^{(0)} \leftarrow \Phi_{0,d,k} \ \forall\ j \in k \ \forall\ k$                    {optional, but recommended}
repeat                     {outer iteration loop, indexed with $l \in 0, \ldots, L-1$}
    Calculate $\boldsymbol{\phi}_0^{(l+\frac{1}{2})}$ and $\Phi_{1,k\pm\frac{1}{2}}^{(l+\frac{1}{2})}$ $\forall\ k$ by solving the discrete form of Eq. (4.42a)
    Calculate $\mathbf{\Phi}_0^{(l+\frac{1}{2})}$, $\dot{\Sigma}_{a,k} \left\langle\!\!\left\langle \phi_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle$ $\forall\ k$, and $(\nu\dot{\Sigma}_f)_k \left\langle\!\!\left\langle \phi_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle$ $\forall\ k$ using Eqs. (4.33c), (4.52b) and (4.52c), respectively
    Calculate $\lambda_t^{(l+1)}$ using Eq. (4.59)
    Construct $\tilde{\boldsymbol{Q}}^{(l+1)}$ using Eq. (4.53b)
    $\mathbf{\Phi}_0^{(l+\frac{1}{2},0)} \leftarrow \mathbf{\Phi}_0^{(l+\frac{1}{2})}$
    repeat                 {power iteration loop, indexed with $m \in 0, \ldots, M-1$}
        Calculate $\mathbf{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})}$ by solving Eq. (4.66a)
        Calculate $\alpha^{(l+\frac{1}{2},m+\frac{1}{2})}$ using Eq. (4.66c)
        Calculate $\mathbf{\Phi}_0^{(l+\frac{1}{2},m+1)}$ using Eq. (4.66b)
    until convergence                 {generally, convergence of $\mathbf{\Phi}_0$ is checked}
    $\mathbf{\Phi}_0^{(l+1)} \leftarrow \mathbf{\Phi}_0^{(l+\frac{1}{2},M)}$
    Calculate $\boldsymbol{\phi}_0^{(l+1)}$ using Eq. (4.41)
    Normalize the scalar flux using Eq. (4.68)
until convergence            {generally, convergence of $\boldsymbol{\phi}_0$ and $\lambda_t$ are checked}
```

4.5 Wielandt-Shifted LDA

In this section, we discuss a technique to improve the convergence rate of PI for LDA applied to eigenvalue problems. This technique is similar to the Wielandt shift technique for CMFD discussed in Section 3.4. Notably, LDA offers a unique advantage over CMFD in this regard. With CMFD, one must take care not to "over-shift" the problem because the upper bound of the shift is unknown. Therefore, one must be conservative when choosing the magnitude of the shift for CMFD. When applying this technique to LDA, the upper bound of the shift *is known beforehand.* This is a consequence of constructing the LDA equations such that the LHS operator is the critical diffusion operator.

As discussed in Section 4.4.2, an initial diffusion calculation is performed before the outer iteration process is started. One of the outcomes of solving this initial diffusion problem is that the diffusion eigenvalue becomes known and is fixed for a given steady-state problem. Because the LHS operator for the LDA equations is the same critical diffusion operator, we can take advantage of the known diffusion eigenvalue in a shifted problem. This offers the following advantages over shifted CMFD problems:

1. there is no danger of over-shifting because the upper bound is known (unlike with CMFD, in which over-shifting can result in divergence or an inaccurate solution), and
2. the shift parameter does not need to change throughout the outer iteration process, which lends the LHS operator to factorization (or another technique that takes advantage of the fixed nature) to improve the efficiency of solving the linear system.

To apply a Wielandt shift to the LDA equations, we redefine the LHS operator for eigenvalue problems (initially defined in Eq. (4.46a)) as shown below:

$$\begin{aligned}\left(\mathcal{L}_{d,eig,sh} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \boldsymbol{\Phi}_0\right)_k \equiv {} & -\tilde{D}_{k+\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,k+1} - \Phi_{0,k}\right) + \tilde{D}_{k-\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,k} - \Phi_{0,k-1}\right) \\ & + \left(\bar{\Sigma}_{a,k} - f_{sh}\lambda_d \overline{\left(\nu\Sigma_f\right)}_k\right) \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Phi_{0,k}\Delta_k \;, \\ & 0 \le f_{sh} < 1 \;,\end{aligned} \tag{4.69}$$

where f_{sh} represents a fraction of the fundamental diffusion eigenvalue. With this new LHS operator, we make the following modification to Eq. (4.66a):

$$\mathcal{L}_{d,eig,sh} \langle\!\langle \boldsymbol{1} \rangle\!\rangle \boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})} = \tilde{\boldsymbol{Q}}^{(l+1)} + (1 - f_{sh})\lambda_d \mathbb{F}_d \langle\!\langle \boldsymbol{1} \rangle\!\rangle \boldsymbol{\Phi}_0^{(l+\frac{1}{2},m)} \;. \tag{4.70}$$

With these new definitions, the operator $\mathcal{L}_{d,eig,sh} \langle\!\langle \boldsymbol{1} \rangle\!\rangle$ becomes more singular as f_{sh} approaches unity. Therefore, one has the choice of how “shifted” the problem will be. To avoid a singular operator, f_{sh} should not be set to unity. Additionally, as f_{sh} approaches unity, the condition number of the LHS operator increases. If the performance of the linear solver is dependent on the condition number, then caution should be taken when choosing f_{sh}. The effect of f_{sh} on the convergence rate of PI for LDA, and the effect of a “high” (near-unity) shift, are explored in Chapters 6 and 7.

4.6 Discussion

In this chapter, we present the novel LDA method for accelerating the convergence of transport problems. This method is intended to serve as a viable replacement for CMFD, which is vulnerable to nonlinear instability as a result of nonlinear terms. Because LDA is linear with respect to the scalar flux (and quantities derived from the scalar flux), it should not be subject to the same type of instability. Evidence for this is provided in Chapters 5 to 7.

During the course of the derivation of LDA, we assumed a *uniform* weighting function $(\mathbb{1})$ for each operator, resulting in the overall volume-weighting of coarse-grid quantities. This choice was made because it is constant throughout the iteration process. If the weighting function is not constant, then the diffusion forward and adjoint eigenpair would need to be recalculated any time the weight function is updated. (The choice of the weighting function is arbitrary, as long as it is strictly positive to avoid the possibility of nonlinear instability.) However, the result is that there are a few extra terms included in the RHS vector of the LDA system to account for the improper weighting of these quantities (effectively maintaining consistency with the transport solution). These terms are in addition to terms that account for the inaccuracy of the Fick's Law approximation. In lagging these terms throughout the iterative process, we assume that they are "small". This is also done for CMFD, but because flux- and volume-weighted quantities are used, the former corrective terms, relating to the weighting choice, are not present. Therefore, LDA lags more terms in the iterative process compared to CMFD. This leads to the possibility of slower convergence for LDA compared to CMFD, depending on how large the lagged terms may become. In a sense, the "price" paid for avoiding nonlinear instabilities is the possibility of worse iterative performance compared to CMFD. Chapters 5 to 7 present results regarding this topic.

A possible approach to mitigating reduced performance compared to CMFD is to use a "better" choice for the weighting function. For example, rather than a uniform function, an approximation of the converged fine-grid scalar flux could be used. One might use the solution from the first transport sweep to reconstruct the LDA operator, while ensuring that any negative flux values are omitted from the weighting function (to guarantee that nonlinear instability is avoided). Conceivably, this would reduce the magnitude of the lagged terms that account for incorrect weighting – ultimately leading to improved iterative properties compared to the case with uniform weighting. We suggest the testing of this approach as future work.

Further, we consider the method of normalizing the low-order LDA scalar flux for eigenvalue problems. Specifically, we consider Eq. (4.66b), which is reproduced below:

$$\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+1\right)} = \mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)} + \alpha^{\left(l+\frac{1}{2},m+\frac{1}{2}\right)}\mathbf{\Phi}_{0,d} \, , \qquad \text{(Eq. (4.66b) revisited)}$$

where $\mathbf{\Phi}_{0,d}$ is the forward diffusion solution. This method of normalization comes directly from the FAT, which states that the general solution to Eq. (4.53a), reproduced below:

$$\mathcal{L}_{d,eig} \langle\!\langle \mathbf{1} \rangle\!\rangle \, \boldsymbol{\Phi}_0^{(l+1)} = \tilde{\boldsymbol{Q}}^{(l+1)} \, , \qquad \text{(Eq. (4.53a) revisited)}$$

(where $\mathcal{L}_{d,eig} \langle\!\langle \mathbf{1} \rangle\!\rangle$ is a singular operator) is a linear combination of the solution to Eq. (4.53a) and the forward diffusion solution $\boldsymbol{\Phi}_{0,d}$. We maintain this normalization method throughout this chapter. However, in practice we manipulate Eq. (4.53a) such that the LHS operator is no longer singular by moving the fission source (or a fraction of the fission source) to the lagged RHS, as shown in Eq. (4.66a) which is reproduced below:

$$\left(\mathcal{L}_{d,eig} + \lambda_d \mathbb{F}_d\right) \langle\!\langle \mathbf{1} \rangle\!\rangle \, \boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})} = \tilde{\boldsymbol{Q}}^{(l+1)} + \lambda_d \mathbb{F}_d \langle\!\langle \mathbf{1} \rangle\!\rangle \, \boldsymbol{\Phi}_0^{(l+\frac{1}{2},m)} \, . \qquad \text{(Eq. (4.66a) revisited)}$$

Because the resulting system no longer resembles one to which the FAT applies, the normalization method shown in Eq. (4.66b) is not necessary. As such, obtaining the forward diffusion solution no longer becomes necessary (because it is only used in Eq. (4.66b), and λ_d can be obtained from the adjoint calculation) and the low-order solution can be normalized using conventional means such as that shown in Eq. (3.13o), reproduced below:

$$\boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+1)} \equiv \left(\frac{P}{\left\langle \mathbf{1}, \mathbb{F}^{(l+\frac{1}{2})} \boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})} \right\rangle} \right) \boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})} \, , \qquad \text{(Eq. (3.13o) revisited)}$$

where P is the global neutron production rate. In a practical implementation of LDA, omitting the unnecessary work of converging the forward diffusion solution can reduce the overall computational cost (as was done for the implementation of LDA in MPACT). We note that the *adjoint* diffusion flux calculation is still necessary, as it is used to compute the transport eigenvalue.

CHAPTER 5

Fourier Analysis

In this chapter, we perform Fourier analyses for heterogeneous, spatially-periodic, fixed-source and eigenvalue transport problems. First, some background is provided in Section 5.1. Then, the analysis is conducted for the following iteration schemes: (i) unaccelerated SI (Section 5.2), (ii) SI accelerated with CMFD (Section 5.3), and (iii) SI accelerated with LDA (Section 5.4). This analysis is used to compare the asymptotic convergence rates, in the form of the spectral radius (ρ), of each case for different spatial configurations. The results of the heterogeneous Fourier analysis are shown to also be applicable to spatially homogeneous problems. Additionally, the analytic prediction of the spectral radius for spatially homogeneous and heterogeneous problems is compared to estimates obtained with numerical experiments using a 1D, $\mathrm{S_N}$ research code in Section 5.5. We also present analytic and numerical results that show the prediction of the nonlinear instability of CMFD. LDA is shown to be insensitive to this nonlinear instability. Finally, we present the conclusions of this chapter in Section 5.6.

5.1 Background

Fourier analysis is a mathematical tool that can be used to theoretically predict the convergence rate of a fixed-point iterative algorithm, and to compare the performance of different algorithms. Due to the complex nature of the process of performing the Fourier analysis, problems that are analyzed are simplified compared to relatively complex practical problems. In this chapter, we employ Fourier analysis to compare the convergence rate of LDA and CMFD. With this tool, the iterative properties of the two acceleration methods are theoretically characterized for certain, “idealized” problems under a range of conditions.

Previous work has shown that CMFD and DSA are equivalent for fixed-source problems, provided the traditional DSA formulation is modified to allow for a coarser spatial mesh to be used for the low-order problem [62]. Other previous work has taken advantage of the predictive capabilities of Fourier analysis to improve the convergence rate of CMFD [25, 26, 63]. Specifically, a "stability parameter" has been used to tune the diffusivity of the low-order problem for a range of problems types based on Fourier analysis. Careful selection of this parameter, introduced by Yamamoto, has been shown to suppress the linear instability of CMFD. However, the introduction of this parameter does not affect the nonlinear instability of CMFD, since the inclusion of the stability parameter does not eliminate nonlinear terms from the formulation.

Though Fourier analysis is commonly performed for spatially-homogeneous problems, it has also been used for spatially-heterogeneous problems [64, 65]. We take the approach of performing the analysis for spatially-heterogeneous problems, rather than homogeneous, for the following reasons:

1. analysis of heterogeneous problems provides results for more realistic cases compared to homogeneous problems,
2. the homogeneous result is easily obtained from the heterogeneous analysis,
3. we are interested in the analytic prediction of the nonlinear instability of CMFD (and the stability of LDA) which is demonstrated more effectively in heterogeneous problems, and
4. we are interested in the performance differences between LDA and CMFD as a result of the use of different (i) weighting functions for the homogenization of the macroscopic cross sections and (ii) prolongation operators.

To clarify the final point, we consider the definition of homogenized cross sections in the CMFD formulation, which are weighted by the scalar flux and volume of the contiguous fine cells that make up a coarse cell $(j \in k)$:

$$\bar{\Sigma}_{u,k} \equiv \frac{\sum\limits_{j\in k} \Sigma_{u,j}\phi_{0,j}h_j}{\sum\limits_{j\in k} \phi_{0,j}h_j} \, , \forall \; u \in \{a,s,t\} \; . \tag{5.1}$$

In contrast, LDA utilizes volume-weighted cross sections (if the chosen weighting function is uniform, as in Chapter 4):

$$\bar{\Sigma}_{u,k} \equiv \frac{\sum\limits_{j\in k} \Sigma_{u,j}h_j}{\sum\limits_{j\in k} h_j} \, , \forall \; u \in \{a,s,t\} \; , \tag{5.2}$$

the use of which is "corrected" by the inclusion of linear reaction rate sums in the source:

$$\dot{\Sigma}_{u,k} \equiv \sum_{j \in k} \Sigma_{u,k} \phi_{0,j} h_j \ . \tag{5.3}$$

Additionally, the numerical diffusion coefficient $\tilde{D}$ is generated using uniformly-weighted cross sections rather than flux- and volume-weighted cross sections. We are interested in possible differences in the iterative properties of LDA and CMFD as a result of these different weighting functions for the cross sections.

Beyond the different weighting functions for cross sections, LDA and CMFD also use different prolongation operators. The prolongation operator in CMFD is nonlinear and multiplicative:

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{(l+\frac{1}{2})} \left(\frac{\Phi_{0,k}^{(l+1)}}{\Phi_{0,k}^{(l+\frac{1}{2})}} \right) , \forall \ j \in k \ , \tag{Eq. (3.8q) revisited}$$

while the LDA prolongation operator is linear and additive:

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{(l+\frac{1}{2})} + \left(\Phi_{0,k}^{(l+1)} - \Phi_{0,k}^{(l+\frac{1}{2})} \right) , \forall \ j \in k \ , \tag{Eq. (4.41) revisited}$$

where outer iterations are indexed with l. We are interested in any differences in the iterative properties of each method resulting from these disparate prolongation operators.

The ultimate goal of the Fourier analysis is to obtain the *spectral radius* ρ. This parameter represents the overall error reduction rate and asymptotic convergence rate of the scalar flux for the iteration scheme. In words, it is the quantity by which the error in the scalar flux is scaled each outer iteration. If $0 < \rho < 1$, the method is considered convergent since

$$\lim_{l \to \infty} \rho^l = 0 \ . \tag{5.4}$$

On the other hand, if $\rho > 1$, then the method is considered divergent because

$$\lim_{l \to \infty} \rho^l = \infty \ . \tag{5.5}$$

In the case where $\rho = 1$, the method is also considered divergent because the error is constant. The value of ρ depends on the approximations used in the problem, the problem type, the solution methodology, the spatial configuration of the problem, and the acceleration scheme used (or lack thereof). In a practical application, the numerical spectral radius, computed as

$$\rho^{(l)} \approx \frac{\|\phi_0^{(l)} - \phi_0^{(l-1)}\|_2}{\|\phi_0^{(l-1)} - \phi_0^{(l-2)}\|_2}\,, \tag{5.6}$$

can be compared to the analytic prediction.

The Fourier analysis is performed for an "idealized" problem that does not represent practical reactor physics problems. Therefore, the results of the Fourier analysis can be reasonably accurate but will not exactly represent the behavior of practical simulations. However, the convergence characteristics predicted by the Fourier analysis can establish general trends that practical problems may exhibit. For example, the results in this chapter show that the performance of certain accelerated problems degrades with increasing coarse cell optical thickness. This trend has also been seen in practical problems, and the Fourier analysis can provide an estimated range of coarse cell optical thickness values that allow for rapid convergence. Further, Fourier analysis can be used to derive modifications to the acceleration scheme that mitigate this performance degradation (as previously mentioned [25, 26]).

The process of theoretically obtaining ρ is nuanced and somewhat abstract. A general description of the steps are as follows:

1. first, the transport and acceleration equations are perturbed around the converged solution,
2. second, the equations are linearized with respect to the scalar flux, and
3. finally, the *Fourier ansatz* is introduced to the linearized equations.

The goal of step 1 is to obtain a set of equations that describes how the error in the scalar flux changes between subsequent iterations. Step 2 is only required for a nonlinear method such as CMFD, and allows for a linear system of equations to be formulated for the spectral radius. Lastly, step 3 allows for the formulation of *spectral relation* equations, which describe how the error reduction rate relates to the iteration index. These equations can then be used to construct an *overall error transition matrix*, $\tilde{S}$, which is part of an eigenvalue problem that can be solved for the spectral radius. $\tilde{S}$ can also be referred to as the *iteration matrix*. This matrix contains separate contributions from the transport sweep and the application of the acceleration step.

The Fourier ansatz contains a spatial modulation function that is dependent on the *Fourier frequency* ω, which may only take on a discrete set of allowable values that satisfy the boundary conditions. An eigenspectrum of error reduction rates θ can be obtained for a given error transition matrix:

$$\theta(\omega) = \operatorname{eig}\left(\tilde{S}\right) , \tag{5.7}$$

where $\tilde{S}$ is the error transition matrix corresponding to a particular discrete Fourier frequency ω, and $\operatorname{eig}\left(\tilde{S}\right)$ represents the eigenspectrum of $\tilde{S}$. The spectral radius is defined as the error reduction rate (or eigenvalue of $\tilde{S}$) with the largest magnitude over all allowable Fourier frequencies:

$$\rho \equiv \max_{\omega} |\theta(\omega)| . \tag{5.8}$$

Once the analytic spectral radius is derived, it can be compared to the numerical estimate that is computed with Eq. (5.6) to confirm the correctness of the analytic result.

In this chapter, we perform a Fourier analysis for spatially-periodic fixed-source and eigenvalue transport problems, optionally accelerated with either CMFD or LDA. Periodic boundary conditions are used for all cases, and the transport equation is solved using discrete ordinates with the Gauss-Legendre quadrature set. A given coarse cell, consisting of p fine cells, is repeated periodically throughout space, with a total of K coarse cells and pK fine cells. The k index is omitted from certain quantities that are identical for all coarse cells. Spatial heterogeneity within a coarse cell is described using vector quantities of the following form:

$$\boldsymbol{a} \equiv \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{pmatrix} , \tag{5.9}$$

where a_j indicates a property of the jth fine cell. In this way, the spatial heterogeneity of the problems is incorporated into the operators that compose the error transition matrices. Additionally, the following notation for mathematical space is used throughout this chapter:

1. $\mathbb{N}$: the set of natural numbers,
2. $\mathbb{R}$: real number space, and
3. $\mathbb{C}$: complex number space.

This notation will be used to denote the space in which each operator lies, with superscripts indicating the dimensionality of the operator. For example, the notation $\mathbb{C}^{n \times n}$ indicates that a given $n \times n$ operator lies in complex number space. Finally, we use the terms "operator" and "matrix" interchangeably.

5.2 Fourier Analysis of Source Iteration

In this section, we perform a Fourier analysis of the $\mathrm{S_N}$ transport equations for spatially-periodic, heterogeneous, fixed-source and eigenvalue problems. Because the steps in this analysis are necessary for the CMFD and LDA Fourier analyses, we present them first. In the subsequent CMFD and LDA sections, the equations and operators obtained here will be reused for brevity.

5.2.1 Fixed-Source Problems

5.2.1.1 Problem Statement

We begin with the steady-state, monoenergetic, discretized $\mathrm{S_N}$ equations for a periodic fixed-source problem with the diamond-difference spatial closure scheme and isotropic scattering:

$$\frac{\mu_n}{h_j}\left(\psi^{(l+\frac{1}{2})}_{n,j+\frac{1}{2}} - \psi^{(l+\frac{1}{2})}_{n,j-\frac{1}{2}}\right) + \Sigma_{t,j}\psi^{(l+\frac{1}{2})}_{n,j} = \frac{1}{2}\left(\Sigma_{s,j}\phi^{(l)}_{0,j} + q_j\right) , \tag{5.10a}$$

$$\psi^{(l+\frac{1}{2})}_{n,j} = \frac{1}{2}\left(\psi^{(l+\frac{1}{2})}_{n,j+\frac{1}{2}} + \psi^{(l+\frac{1}{2})}_{n,j-\frac{1}{2}}\right) , \tag{5.10b}$$

$$\phi^{(l+\frac{1}{2})}_{0,j} \equiv \sum_{n=1}^{N} w_n \psi^{(l+\frac{1}{2})}_{n,j} , \tag{5.10c}$$

$$\phi^{(l+\frac{1}{2})}_{1,j\pm\frac{1}{2}} \equiv \sum_{n=1}^{N} w_n \mu_n \psi^{(l+\frac{1}{2})}_{n,j\pm\frac{1}{2}} , \tag{5.10d}$$

$$\{j \mid j \in \mathbb{N}, 1 \leq j \leq pK\} ,$$

with the following boundary condition:

$$\psi^{(l+\frac{1}{2})}_{n,\frac{1}{2}} = \psi^{(l+\frac{1}{2})}_{n,pK+\frac{1}{2}} , \{n \mid n \in \mathbb{N}, 1 \leq n \leq N\} . \tag{5.10e}$$

These equations describe unaccelerated SI and were previously presented in Chapter 3 (with different boundary conditions). For this problem, we have imposed a periodic boundary condition as stated in Eq. (5.10e). We also include the following equation to complete the iteration scheme:

$$\phi^{(l+1)}_j = \phi^{(l+\frac{1}{2})}_j , \ \forall\ j . \tag{5.10f}$$

The equations are presented in this way so that the results from this section can be reused for accelerated cases (which assume the same iteration indexing as that in Eqs. (5.10a) to (5.10e)) to avoid unnecessary repetition. Equation (5.10f) does not apply to accelerated problems because $\phi^{(l+1)}_j\ \forall\ j$ is obtained differently (by solving the low-order equations).

In this problem, K coarse cells are present, with p fine cells per coarse cell, resulting in a total number of pK fine cells. Material properties of the problem repeat periodically in space, meaning that the following relationship holds for the macroscopic cross sections:

$$\Sigma_{u,j} = \Sigma_{u,j+p} \ , \tag{5.10g}$$

and the fixed source:

$$q_j = q_{j+p} \ , \tag{5.10h}$$

for all values of j. The solution *iterates* (referring to estimates of the solution obtained during the iteration process), however, are generally *not* spatially periodic as they converge. This is due to the possibility of a random, non-periodic initial guess for the solution as a function of space, resulting in ψ and ϕ_0 solution iterates that are not spatially periodic until convergence. As the outer iterations progress, the solution iterates approach the spatially-periodic, converged solution (if the iteration scheme is convergent). Figure 5.1 depicts the periodicity of the material properties and the lack thereof for the ϕ_0 solution iterates.

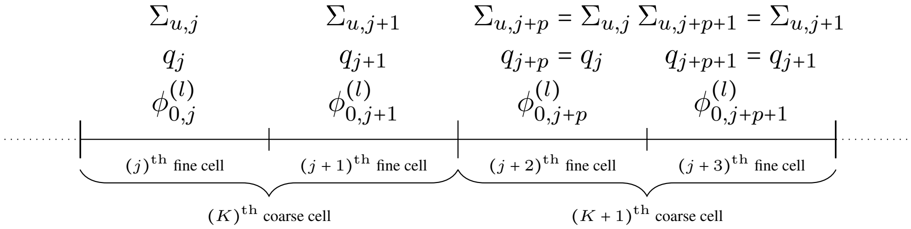


Figure 5.1: Depiction of heterogeneous problem. Spatially-periodic material properties $(\Sigma_{u,j},\ q_j)$ and non-periodic solution iterates $\left(\phi_{0,j}^{(l)}\right)$ are shown for $p = 2$.

5.2.1.2 Perturbation

First, we perturb each of the equations in Eqs. (5.10) around the converged, periodic solution. The following substitutions are made:

$$\psi_{n,j}^{(l+\frac{1}{2})} = \psi_{n,j}^{(\infty)} + \epsilon\tilde{\psi}_{n,j}^{(l+\frac{1}{2})} \ , \tag{5.11a}$$

$$\psi_{n,j\pm\frac{1}{2}}^{(l+\frac{1}{2})} = \psi_{n,j\pm\frac{1}{2}}^{(\infty)} + \epsilon\tilde{\psi}_{n,j\pm\frac{1}{2}}^{(l+\frac{1}{2})} \ , \tag{5.11b}$$

$$\phi_{0,j}^{(l)} = \phi_{0,j}^{(\infty)} + \epsilon\tilde{\phi}_{0,j}^{(l)} \ , \tag{5.11c}$$

$$\phi_{0,j}^{(l+\frac{1}{2})} = \phi_{0,j}^{(\infty)} + \epsilon\tilde{\phi}_{0,j}^{(l+\frac{1}{2})} \ , \tag{5.11d}$$

$$\{j \mid j \in \mathbb{N}, 1 \leq j \leq pK\} \ ,$$

$$|\epsilon| \ll 1 \ , \tag{5.11e}$$

where the superscript (∞) is used for converged quantities, the constant ϵ that scales the error quantities is small, and tilde quantities $(\tilde{\cdot})$ represent the error relative to the converged solution for the corresponding quantity.

After Eqs. (5.11) are substituted into Eqs. (5.10) and the resulting equations are simplified, we obtain the following $\mathrm{S_N}$ error equations:

$$\frac{\mu_n}{\tau_j}\left(\tilde{\psi}^{(l+\frac{1}{2})}_{n,j+\frac{1}{2}} - \tilde{\psi}^{(l+\frac{1}{2})}_{n,j-\frac{1}{2}}\right) + \tilde{\psi}^{(l+\frac{1}{2})}_{n,j} = \frac{1}{2}c_j\tilde{\phi}^{(l)}_{0,j} \ , \tag{5.12a}$$

$$\tilde{\psi}^{(l+\frac{1}{2})}_{n,j} = \frac{1}{2}\left(\tilde{\psi}^{(l+\frac{1}{2})}_{n,j+\frac{1}{2}} + \tilde{\psi}^{(l+\frac{1}{2})}_{n,j-\frac{1}{2}}\right) \ , \tag{5.12b}$$

$$\tau_j \equiv \Sigma_{t,j}h_j \ , \tag{5.12c}$$

$$c_j \equiv \frac{\Sigma_{s,j}}{\Sigma_{t,j}} \ , \tag{5.12d}$$

$$\tilde{\phi}^{(l+\frac{1}{2})}_{0,j} \equiv \sum_{n=1}^{N} w_n\tilde{\psi}^{(l+\frac{1}{2})}_{n,j} \ , \tag{5.12e}$$

$$\{j \mid j \in \mathbb{N}, 1 \le j \le pK\} \ ,$$

$$\tilde{\Phi}^{(l+\frac{1}{2})}_{1,k+\frac{1}{2}} \equiv \sum_{n=1}^{N} \mu_n w_n\tilde{\psi}^{(l+\frac{1}{2})}_{n,pk+\frac{1}{2}} \ , \quad \{k \mid k \in \mathbb{N}, 0 \le k \le K\} \ , \tag{5.12f}$$

$$\tilde{\psi}^{(l+\frac{1}{2})}_{n,\frac{1}{2}} = \tilde{\psi}^{(l+\frac{1}{2})}_{n,pK+\frac{1}{2}} \ , \quad \{n \mid n \in \mathbb{N}, 1 \le n \le N\} \ . \tag{5.12g}$$

We note that the $O(1)$ terms cancel out and the $O(\epsilon^2)$ and higher terms are neglected, leaving only the leading $O(\epsilon)$ terms. Additionally, these equations have been cast in terms of the following fundamental material properties: (i) the optical thickness, τ_j, defined in Eq. (5.12c) and (ii) the scattering ratio, c_j, defined in Eq. (5.12d).

5.2.1.3 Insertion of the Fourier Ansatz

The next step is to insert the Fourier *ansatz* into the error equations. The general form of the solution in the Fourier domain contains the following terms: (i) a spatial function modulated by the Fourier frequency $\left(e^{i\omega\bar{\Sigma}_t X_k}\right)$, (ii) an error reduction rate (θ) raised to the power of the iteration index l, and (iii) parameters with the same dimensionality as the error term. Our goal is to use this ansatz to construct an eigenvalue problem to determine the spectral radius, which is the largest error reduction rate over all Fourier frequencies. An iteration matrix will be constructed from the relationship between each scaling parameter, with the error reduction rates serving as the eigenvalues of this matrix.

For the scaling parameters, the following are defined on the basis of fine cells or fine cell edges: a, b, E, and d. The term g is defined at the left and right edges of a coarse cell. Keeping the periodic nature of the system in mind, we define these parameters *locally* for a coarse cell. Due to the periodic nature of the system, these values are the same for all coarse cells. Therefore, we only define a *single* set of scaling parameters that apply to all coarse cells in the problem. The local index r is used for parameters defined on the basis of the fine cells, with r ranging from 1 to p. For the error terms, the combination of the *local* spatial index r and *global* coarse-cell index k correspond to a given *global* fine-cell index j. For periodic properties of the fine cells, the local index r will be substituted for the global index j when relevant. Because there are only two values of g, these terms are simply subscripted with the side of the coarse cell to which it corresponds.

Each parameter is an unknown to be determined for the Fourier analysis, and the dimensionality of the scaling term matches the dimensionality of the term it represents. For example, the parameter b that scales $\tilde{\psi}$ has values for each angle n and fine spatial cell with local index r. Thus, there is a total of pN unique values of b. The ansatz substitutions are shown below:

$$\tilde{\psi}^{(l+\frac{1}{2})}_{n,j\pm\frac{1}{2}} = \theta^l a_{n,r\pm\frac{1}{2}} e^{i\omega\bar{\Sigma}_t X_k} \ , \tag{5.13a}$$

$$\tilde{\psi}^{(l+\frac{1}{2})}_{n,j} = \theta^l b_{n,r} e^{i\omega\bar{\Sigma}_t X_k} \ , \tag{5.13b}$$

$$\{n \mid n \in \mathbb{N}, 1 \le n \le N\} \ ,$$

$$\tilde{\phi}^{(l)}_{0,j} = \theta^l E_r e^{i\omega\bar{\Sigma}_t X_k} \ , \tag{5.13c}$$

$$\tilde{\phi}^{(l+\frac{1}{2})}_{0,j} = \theta^l d_r e^{i\omega\bar{\Sigma}_t X_k} \ , \tag{5.13d}$$

$$\{j \mid j \in \mathbb{N}, 1 \le j \le pK\} \ , \quad \{r \mid r \in \mathbb{N}, 1 \le r \le p\} \ ,$$

$$\tilde{\Phi}^{(l+\frac{1}{2})}_{1,k-\frac{1}{2}} = \theta^l g_{\text{left}} e^{i\omega\bar{\Sigma}_t X_k} \ , \tag{5.13e}$$

$$\tilde{\Phi}^{(l+\frac{1}{2})}_{1,k+\frac{1}{2}} = \theta^l g_{\text{right}} e^{i\omega\bar{\Sigma}_t X_k} \ , \tag{5.13f}$$

$$\{k \mid k \in \mathbb{N}, 1 \le k \le K\} \ , \quad \{l \mid l \in \mathbb{N}, 0 \le l < L\} \ ,$$

where X_k is the geometric center of a given coarse cell k, L is the total number of outer iterations performed, and ω is the Fourier frequency.

Now, the Fourier ansatz is substituted into the linearized error equations so that the error reduction rate can be computed. The resulting equations are referred to as the *spectral relation* equations, since they describe the relationships between the constants that scale the error reduction rate for a given Fourier frequency. We substitute Eqs. (5.13a) to (5.13c) into Eq. (5.12a) to yield the following form of the $\mathrm{S_N}$ transport equation:

$$\frac{\mu_n}{\tau_r}\left(a_{n,r+\frac{1}{2}} - a_{n,r-\frac{1}{2}}\right) + b_{n,r} = \frac{1}{2}c_r E_r \ . \tag{5.14a}$$

Next, Eqs. (5.13a) and (5.13b) are substituted into Eq. (5.12b) to yield the following form of the diamond-difference closure relationship:

$$b_{n,r} = \frac{1}{2}\left(a_{n,r+\frac{1}{2}} + a_{n,r-\frac{1}{2}}\right) \ . \tag{5.14b}$$

Now, we tackle the definitions of the scalar flux and neutron current defined in Eqs. (5.12e) and (5.12f) by inserting Eqs. (5.13a), (5.13b) and (5.13d) to (5.13f):

$$d_r = \sum_{n=1}^{N} b_{n,r} w_n \ , \tag{5.14c}$$

$$g_{\text{left}} = \sum_{n=1}^{N} \mu_n w_n a_{n,\frac{1}{2}} \ , \tag{5.14d}$$

$$g_{\text{right}} = \sum_{n=1}^{N} \mu_n w_n a_{n,p+\frac{1}{2}} \ . \tag{5.14e}$$

This implies that the relationship between g_{left} and g_{right} is the same as that between $a_{n,\frac{1}{2}}$ and $a_{n,p+\frac{1}{2}}$. To obtain an explicit relationship between g_{left} and g_{right}, we consider Eq. (5.13e) and insert the index $k+1$:

$$\tilde{\Phi}^{(l+\frac{1}{2})}_{(k+1)-\frac{1}{2}} = \tilde{\Phi}^{(l+\frac{1}{2})}_{k+\frac{1}{2}} = \theta^l g_{\text{left}} e^{i\omega\bar{\Sigma}_t X_{k+1}} \ , \tag{5.14f}$$

which can then be equated with Eq. (5.13f). Doing so yields:

$$\theta^l g_{\text{right}} e^{i\omega\bar{\Sigma}_t X_k} = \theta^l g_{\text{left}} e^{i\omega\bar{\Sigma}_t X_{k+1}} \ , \tag{5.14g}$$

which simplifies to the following relationship:

$$g_{\text{right}} = g_{\text{left}} e^{i\omega\Lambda} \ . \tag{5.14h}$$

Further, we note that Eq. (5.14c) can be written in vector form:

$$\boldsymbol{d} = \sum_{n=1}^{N} \boldsymbol{b}_n w_n \ , \tag{5.14i}$$

where

$$\boldsymbol{d} \equiv \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_p \end{pmatrix} , \quad \boldsymbol{d} \in \mathbb{C}^{p\times 1} , \tag{5.14j}$$

and

$$\boldsymbol{b}_n \equiv \begin{pmatrix} b_{n,1} \\ b_{n,2} \\ \vdots \\ b_{n,p} \end{pmatrix} , \quad \boldsymbol{b}_n \in \mathbb{C}^{p\times 1} . \tag{5.14k}$$

Finally, we obtain a spectral relation equation for the coarse-cell edges by equating Eq. (5.13a) for the right edge of coarse cell k (corresponding to the right edge of fine cell $j = pk$) and the left edge of coarse cell $k+1$ (corresponding to the left edge of fine cell $j = pk+1$):

$$\theta^l a_{n,\frac{1}{2}} e^{i\omega\bar{\Sigma}_t X_{k+1}} = \theta^l a_{n,p+\frac{1}{2}} e^{i\omega\bar{\Sigma}_t X_k} . \tag{5.14l}$$

This simplifies to

$$a_{n,\frac{1}{2}} e^{i\omega\Lambda} = a_{n,p+\frac{1}{2}} , \tag{5.14m}$$

where the coarse-cell total optical thickness Λ is defined as

$$\Lambda \equiv \bar{\Sigma}_t \Delta = \sum_{r=1}^{p} \Sigma_{t,r} h_r . \tag{5.14n}$$

The relationship in Eq. (5.14m) can now be inserted into Eq. (5.14e) to yield:

$$g_{\text{right}} = e^{i\omega\Lambda} \sum_{n=1}^{N} \mu_n w_n a_{n,\frac{1}{2}} . \tag{5.14o}$$

Then, we can insert Eq. (5.14d) to obtain the following relation:

$$g_{\text{right}} = g_{\text{left}} e^{i\omega\Lambda} , \tag{5.14p}$$

which agrees with Eq. (5.14h). With this, we have obtained the spectral relation equations corresponding to the $\mathrm{S_N}$ transport equations.

The next step is to construct operators that describe the relationships between the parameters present in Eqs. (5.14) for a given coarse cell. We begin by defining a closure operator that relates the parameters corresponding to the cell-edge and cell-center angular flux. Using Eqs. (5.14b) and (5.14m) we can define the following equation:

$$\bar{Y}\boldsymbol{a}_n = \boldsymbol{b}_n \ , \tag{5.15a}$$

where the diamond-difference operator $\bar{Y}$ is defined as

$$\bar{Y} \equiv \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & \cdots & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & \cdots \\ & & \vdots & \ddots & \\ \frac{1}{2}e^{i\omega\Lambda} & 0 & \cdots & 0 & \frac{1}{2} \end{pmatrix} , \quad \bar{Y} \in \mathbb{C}^{p\times p} \ , \tag{5.15b}$$

and

$$\boldsymbol{a}_n \equiv \begin{pmatrix} a_{n,1} \\ a_{n,2} \\ \vdots \\ a_{n,p} \end{pmatrix} , \quad \boldsymbol{a}_n \in \mathbb{C}^{p\times 1} \ . \tag{5.15c}$$

With this, we have obtained the relationships between each of the parameters that scale the error reduction rate for the cell-edge and cell-center angular flux.

Similarly, we can obtain a streaming operator Y_n that relates a for a given angle n and fine cell r to E_r using Eq. (5.14a):

$$Y_n\boldsymbol{a}_n = \frac{1}{2}C\boldsymbol{E} \ , \tag{5.16a}$$

where we define the streaming operator Y_n for a given angle n as

$$Y_n \equiv \bar{Y} + \begin{pmatrix} -\frac{\mu_n}{\tau_1} & \frac{\mu_n}{\tau_1} & 0 & \cdots & 0 \\ 0 & -\frac{\mu_n}{\tau_2} & \frac{\mu_n}{\tau_2} & 0 & \cdots \\ & & \vdots & \ddots & \\ \frac{\mu_n}{\tau_p}e^{i\omega\Lambda} & 0 & \cdots & 0 & -\frac{\mu_n}{\tau_p} \end{pmatrix} , \quad Y_n \in \mathbb{C}^{p\times p} \ , \tag{5.16b}$$

and the vector $\boldsymbol{E}$ as

$$\boldsymbol{E} \equiv \begin{pmatrix} E_1 \\ E_2 \\ \vdots \\ E_p \end{pmatrix} , \quad \boldsymbol{E} \in \mathbb{C}^{p\times 1} \ . \tag{5.16c}$$

Notably, $\boldsymbol{E}$, the elements of which scale $\tilde{\phi}_{0,j}$ for a given iteration l, will serve as the eigenvector of the eigenvalue problem that we will ultimately construct to compute the spectral radius. We also define the diagonal matrix C as

$$C \equiv \begin{pmatrix} c_1 & 0 & 0 & \cdots & 0 \\ 0 & c_2 & 0 & \cdots & 0 \\ 0 & 0 & \vdots & \ddots & \\ 0 & 0 & \cdots & 0 & c_p \end{pmatrix} , \quad C \in \mathbb{R}^{p\times p} , \tag{5.16d}$$

with elements corresponding to the scattering ratio in a given fine cell (c_r). With the operators $\bar{Y}$, Y_n, and C defined, the following relationship between $\boldsymbol{a}_n$ and $\boldsymbol{E}$ can be obtained using Eq. (5.16a):

$$\boldsymbol{a}_n = \frac{1}{2} Y_n^{-1} C \boldsymbol{E} , \tag{5.16e}$$

where Y_n^{-1} is the inverse of the streaming operator Y_n. Further, a relationship between $\boldsymbol{b}_n$ and $\boldsymbol{E}$ can be obtained by inserting Eq. (5.16e) into Eq. (5.15a), yielding

$$\boldsymbol{b}_n = \frac{1}{2} \bar{Y} Y_n^{-1} C \boldsymbol{E} . \tag{5.16f}$$

A relationship between $\boldsymbol{d}$ and $\boldsymbol{E}$ is found by inserting Eq. (5.16f) into Eq. (5.14i), yielding

$$\boldsymbol{d} = \frac{1}{2} \sum_{n=1}^{N} w_n \bar{Y} Y_n^{-1} C \boldsymbol{E} . \tag{5.16g}$$

This can be written as

$$\boldsymbol{d} = \tilde{H}_{\text{fs}} \boldsymbol{E} , \tag{5.16h}$$

where the matrix $\tilde{H}_{\text{fs}}$ is defined as

$$\tilde{H}_{\text{fs}} \equiv \frac{1}{2} \sum_{n=1}^{N} w_n \bar{Y} Y_n^{-1} C , \quad \tilde{H}_{\text{fs}} \in \mathbb{C}^{p\times p} . \tag{5.16i}$$

$\tilde{H}_{\text{fs}}$ relates the parameters that correspond to the error in the scalar flux from the previous outer iteration to the next outer iteration by solving the S$_\text{N}$ transport equation. Therefore, we refer to $\tilde{H}_{\text{fs}}$ as the *transport error transition matrix.* The $\tilde{H}_{\text{fs}}$ matrix describes how errors in $\boldsymbol{\phi}^{(l)}$ propagate into errors in $\boldsymbol{\phi}^{(l+1)}$. For unaccelerated, fixed-source, transport problems, $\tilde{H}_{\text{fs}}$ is the overall error transition matrix (or iteration matrix). The spectral radius of unaccelerated SI applied to fixed-source problems can then be computed by solving the following eigenvalue problem for the eigenvalue θ with the largest magnitude over all allowable Fourier frequencies:

$$\theta \boldsymbol{E} = \tilde{H}_{\text{fs}} \boldsymbol{E} . \tag{5.16j}$$

The equations and definitions obtained in this section are relevant for accelerated cases, and will be used in Sections 5.3 and 5.4. Rather than repeat these steps in subsequent sections, they are stated here. However, the relationship between $\phi^{(l+\frac{1}{2})}$ and $\phi^{(l+1)}$ stated in Eq. (5.10f) is much different in these later sections. Additionally, the overall error transition matrix (denoted as $\tilde{S}$ for accelerated cases) will take a different form that includes $\tilde{H}$ and a separate contribution from the low-order problem. Therefore, $\tilde{H}$ is replaced with $\tilde{S}$ in Eq. (5.16j) for accelerated problems. Next, we perform the Fourier analysis for unaccelerated SI applied to eigenvalue problems (in which $\tilde{H}$ takes a different form than that derived here for fixed-source problems).

5.2.2 Eigenvalue Problems

5.2.2.1 Problem Statement

For the transport equations, the only difference between the fixed-source and eigenvalue equations lies in the RHS of the NTE, which now contains the fission source scaled by the fundamental mode eigenvalue. Therefore, we state only this equation to avoid unnecessary repetition:

$$\frac{\mu_n}{h_j}\left(\psi_{n,j+\frac{1}{2}}^{(l+\frac{1}{2})} - \psi_{n,j-\frac{1}{2}}^{(l+\frac{1}{2})}\right) + \Sigma_{t,j}\psi_{n,j}^{(l+\frac{1}{2})} = \frac{1}{2}\left[\Sigma_{s,j} + \lambda^{(l)}\left(\nu\Sigma_f\right)_j\right]\phi_{0,j}^{(l)} , \tag{5.17}$$

$$\{j \mid j \in \mathbb{N}, 1 \le j \le pK\} \ ,$$

which replaces Eq. (5.10a). The remaining equations are the same as those in Eqs. (5.10).

5.2.2.2 Perturbation

For the perturbation step, the same perturbations listed in Eqs. (5.11) are used, with the addition of the following perturbation equation for the fundamental eigenvalue:

$$\lambda^{(l)} = \lambda = \frac{\bar{\Sigma}_a^{(\infty)}}{\overline{\nu\Sigma_f}^{(\infty)}} , \tag{5.18}$$

which takes advantage of the faster convergence of the eigenvalue compared to the flux.

After inserting Eqs. (5.11) and Eq. (5.18) into the $\mathrm{S_N}$ transport equations we obtain the same set of equations as those listed in Eqs. (5.14) with the exception of Eq. (5.14a). This equation now takes the following form:

$$\frac{\mu_n}{\tau_j}\left(\tilde{\psi}^{(l+\frac{1}{2})}_{n,j+\frac{1}{2}} - \tilde{\psi}^{(l+\frac{1}{2})}_{n,j-\frac{1}{2}}\right) + \tilde{\psi}^{(l+\frac{1}{2})}_{n,j} = \frac{1}{2}\gamma_j\tilde{\phi}^{(l)}_{0,j} \ , \tag{5.19a}$$

where we define γ_j as

$$\gamma_j \equiv c_j + \lambda f_j \ , \tag{5.19b}$$

and f_j as

$$f_j \equiv \frac{(\nu\Sigma_f)_j}{\Sigma_{t,j}} \ . \tag{5.19c}$$

Next, we insert the Fourier ansatz.

5.2.2.3 Insertion of the Fourier Ansatz

We now insert the Fourier ansatz listed in Eqs. (5.13) into the $\mathrm{S_N}$ error equations. This is the same ansatz used for fixed-source problems. After the substitution and performing the same steps as those in Section 5.2.1, we obtain the following $\mathrm{S_N}$ spectral relation equation for the NTE:

$$\frac{\mu_n}{\tau_r}\left(a_{n,r+\frac{1}{2}} - a_{n,r-\frac{1}{2}}\right) + b_{n,r} = \frac{1}{2}\gamma_r E_r \ , \tag{5.20a}$$

which replaces Eq. (5.14a). The remaining equations are the same as those in Eqs. (5.14).

Using the spectral relation equations, we define the error transition matrix $\tilde{H}_{\text{eig}}$ for the eigenvalue $\mathrm{S_N}$ transport equations as

$$\tilde{H}_{\text{eig}} \equiv \frac{1}{2}\sum_{n=1}^{N} w_n \bar{Y} Y_n^{-1}\Gamma \ , \quad \tilde{H}_{\text{eig}} \in \mathbb{C}^{p\times p} \ , \tag{5.20b}$$

where the Γ operator is defined as

$$\Gamma \equiv C + \lambda F \ , \quad \Gamma \in \mathbb{R}^{p\times p} \ , \tag{5.20c}$$

and the F operator is defined as

$$F \equiv \begin{pmatrix} f_1 & 0 & 0 & \cdots & 0 \\ 0 & f_2 & 0 & \cdots & 0 \\ 0 & 0 & \vdots & \ddots & \\ 0 & 0 & \cdots & 0 & f_p \end{pmatrix} \ , \quad F \in \mathbb{R}^{p\times p} \ . \tag{5.20d}$$

The only difference between Eq. (5.16i) and Eq. (5.20b) is that C is replaced with Γ, which accounts for both the scattering and fission sources. With this, we have obtained the set of spectral relation equations and associated operators resulting from the $\mathrm{S_N}$ equations for eigenvalue problems. The spectral radius of unaccelerated SI applied to eigenvalue problems is computed by solving the eigenvalue problem shown in Eq. (5.16j), with $\tilde{H}_{\text{fs}}$ replaced by $\tilde{H}_{\text{eig}}$. As previously stated, the results from this section will be reused for accelerated cases. Next, the Fourier analysis for CMFD-accelerated problems is performed.

5.3 CMFD Fourier Analysis

This section provides the process and details of a spatially-periodic, heterogeneous Fourier analysis for CMFD for both fixed-source and eigenvalue problems. We make use of many of the equations derived in Section 5.2, since the fine-mesh solution is still obtained by solving the transport equations. Because CMFD is nonlinear with respect to the scalar flux, the equations that describe the low-order step must be linearized before insertion of the Fourier ansatz.

5.3.1 Fixed-Source Problems

5.3.1.1 Problem Statement

We begin with the steady-state, monoenergetic, CMFD equations for a periodic, fixed-source problem:

$$\left(\Phi^{(l+1)}_{1,k+\frac{1}{2}} - \Phi^{(l+1)}_{1,k-\frac{1}{2}}\right) + \bar{\Sigma}^{(l+\frac{1}{2})}_{a,k}\Phi^{(l+1)}_{0,k}\Delta = \bar{Q}\Delta \ , \tag{5.21a}$$

$$\Phi^{(l+1)}_{1,k+\frac{1}{2}} \equiv -\tilde{D}^{(l+\frac{1}{2})}_{k+\frac{1}{2}}\left(\Phi^{(l+1)}_{0,k+1} - \Phi^{(l+1)}_{0,k}\right) + \hat{D}^{(l+\frac{1}{2})}_{k+\frac{1}{2}}\left(\Phi^{(l+1)}_{0,k+1} + \Phi^{(l+1)}_{0,k}\right) \ , \tag{5.21b}$$

$$\tilde{D}^{(l+\frac{1}{2})}_{k+\frac{1}{2}} \equiv \frac{2}{3\left(\bar{\Sigma}^{(l+\frac{1}{2})}_{t,k+1}\Delta_{k+1} + \bar{\Sigma}^{(l+\frac{1}{2})}_{t,k}\Delta_k\right)} \ , \tag{5.21c}$$

$$\hat{D}^{(l+\frac{1}{2})}_{k+\frac{1}{2}} \equiv \frac{\Phi^{(l+\frac{1}{2})}_{1,k+\frac{1}{2}} + \tilde{D}^{(l+\frac{1}{2})}_{k+\frac{1}{2}}\left(\Phi^{(l+\frac{1}{2})}_{0,k+1} - \Phi^{(l+\frac{1}{2})}_{0,k}\right)}{\Phi^{(l+\frac{1}{2})}_{0,k+1} + \Phi^{(l+\frac{1}{2})}_{0,k}} \ , \tag{5.21d}$$

$$\bar{\Sigma}^{(l+\frac{1}{2})}_{u,k} \equiv \frac{\sum\limits_{j\in k}\Sigma_{u,j}\phi^{(l+\frac{1}{2})}_{0,j}h_j}{\sum\limits_{j\in k}\phi^{(l+\frac{1}{2})}_{0,j}h_j} \ , \forall\ u \in \{a,s,t\} \ , \tag{5.21e}$$

$$\bar{Q} \equiv \frac{\sum\limits_{j\in k} q_j h_j}{\Delta} \ , \tag{5.21f}$$

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{(l+1)} \left(\frac{\Phi_{0,k}^{(l+1)}}{\Phi_{0,k}^{(l+\frac{1}{2})}} \right) , \forall \ j \in k \ , \tag{5.21g}$$

$$\Phi_{0,k}^{(l+\frac{1}{2})} \equiv \frac{\sum\limits_{j\in k} \phi_{0,j}^{(l+\frac{1}{2})} h_j}{\Delta} \ , \tag{5.21h}$$

$$\Delta \equiv \sum_{j\in k} h_j \ , \tag{5.21i}$$

$$\Phi_{1,\frac{1}{2}}^{(l+\frac{1}{2})} = \Phi_{1,K+\frac{1}{2}}^{(l+\frac{1}{2})} \ , \tag{5.21j}$$

$$\Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})} \equiv \sum_{n=1}^{N} \mu_n w_n \psi_{n,pk+\frac{1}{2}}^{(l+\frac{1}{2})} \ , \quad \{k \mid k \in \mathbb{N}, 0 \le k \le K\} \ . \tag{5.21k}$$

These equations were derived in Section 3.2.1. Next, we linearize these equations and perturb them around the converged solution.

5.3.1.2 Perturbation & Linearization

First, we perturb each of the CMFD equations around the converged solution. The following perturbation equations are substituted into Eq. (5.21a):

$$\phi_{0,j}^{(l+\frac{1}{2})} = \phi_{0,j}^{(\infty)} + \epsilon \tilde{\phi}_{0,j}^{(l+\frac{1}{2})} \ , \tag{5.22a}$$

$$\Phi_{0,k}^{(l+\frac{1}{2})} = \Phi_{0,k}^{(\infty)} + \epsilon \tilde{\Phi}_{0,k}^{(l+\frac{1}{2})} \ , \tag{5.22b}$$

$$\Phi_{0,k}^{(l+1)} = \Phi_{0,k}^{(\infty)} + \epsilon \tilde{\Phi}_{0,k}^{(l+1)} \ , \tag{5.22c}$$

$$\Phi_{1,k\pm\frac{1}{2}}^{(l+\frac{1}{2})} = \Phi_{1,k\pm\frac{1}{2}}^{(\infty)} + \epsilon \tilde{\Phi}_{1,k\pm\frac{1}{2}}^{(l+\frac{1}{2})} \ , \tag{5.22d}$$

$$|\epsilon| \ll 1 \ . \tag{5.22e}$$

The CMFD equations are nonlinear with respect to the scalar flux, and will not yield a linear system of equations that are conducive to computing the spectral radius. Therefore, we must *linearize* the CMFD equations after making the substitutions in Eqs. (5.22). This is accomplished with a Taylor expansion about $\epsilon\,(\tilde{\cdot}) = 0$, and by neglecting $O(\epsilon^2)$ terms. Each nonlinear term is treated separately. We begin with the absorption term in a given coarse cell:

$$\bar{\Sigma}_{a,k}^{(l+\frac{1}{2})}\Phi_{0,k}^{(l+1)} = \frac{\sum\limits_{j\in k}\Sigma_{a,j}\left(\phi_{0,j}^{(\infty)}+\epsilon\phi_{0,j}^{(l+\frac{1}{2})}\right)h_j}{\sum\limits_{j\in k}\left(\phi_{0,j}^{(\infty)}+\epsilon\phi_{0,j}^{(l+\frac{1}{2})}\right)}\left(\Phi_{0,k}^{(\infty)}+\epsilon\tilde{\Phi}_{0,k}^{(l+1)}\right)$$

$$= \frac{\Phi_{0,k}^{(\infty)}\sum\limits_{j\in k}\Sigma_{a,j}\phi_{0,j}^{(\infty)}h_j+\epsilon\left(\Phi_{0,k}^{(\infty)}\sum\limits_{j\in k}\Sigma_{a,j}\phi_{0,j}^{(l+\frac{1}{2})}h_j+\tilde{\Phi}_{0,k}^{(l+1)}\sum\limits_{j\in k}\Sigma_{a,j}\phi_{0,j}^{(\infty)}h_j\right)+\cancel{O(\epsilon^2)}}{\Phi_{0,k}^{(\infty)}\Delta+\epsilon\sum\limits_{j\in k}\tilde{\phi}_{0,j}^{(l+\frac{1}{2})}h_j}\,, \tag{5.23a}$$

with the following definition of the converged coarse cell scalar flux:

$$\Phi_{0,k}^{(\infty)} \equiv \frac{\sum\limits_{j\in k}\phi_{0,j}^{(\infty)}h_j}{\Delta}\,. \tag{5.23b}$$

We make use of the following first-order Taylor expansion for the denominator of Eq. (5.23a):

$$\frac{1}{\Phi_{0,k}^{(\infty)}\Delta+\epsilon\sum\limits_{j\in k}\tilde{\phi}_{0,j}^{(l+\frac{1}{2})}h_j} \approx \frac{1}{\Phi_{0,k}^{(\infty)}\Delta}\left(1-\frac{\epsilon}{\Phi_{0,k}^{(\infty)}\Delta}\sum_{j\in k}\tilde{\phi}_{0,j}^{(l+\frac{1}{2})}h_j\right)\,. \tag{5.23c}$$

With the above substitution, we obtain:

$$\begin{aligned}\bar{\Sigma}_{a,k}^{(l+\frac{1}{2})}\Phi_{0,k}^{(l+1)} &\approx \frac{1}{\Phi_{0,k}^{(\infty)}\Delta}\left(1-\frac{\epsilon}{\Phi_{0,k}^{(\infty)}\Delta}\sum_{j\in k}\tilde{\phi}_{0,j}^{(l+\frac{1}{2})}h_j\right)\Bigg[\Phi_{0,k}^{(\infty)}\sum_{j\in k}\Sigma_{a,j}\phi_{0,j}^{(\infty)}h_j+\\ &\qquad\epsilon\left(\Phi_{0,k}^{(\infty)}\sum_{j\in k}\Sigma_{a,j}\phi_{0,j}^{(l+\frac{1}{2})}h_j+\tilde{\Phi}_{0,k}^{(l+1)}\sum_{j\in k}\Sigma_{a,j}\phi_{0,j}^{(\infty)}h_j\right)\Bigg]\\ &\approx \Phi_{0,k}^{(\infty)}\bar{\Sigma}_{a,k}^{(\infty)}+\epsilon\left[\frac{1}{\Delta}\sum_{j\in k}\left(\Sigma_{a,j}-\bar{\Sigma}_{a}^{(\infty)}\right)\tilde{\phi}_{0,j}^{(l+\frac{1}{2})}h_j+\bar{\Sigma}_{a}^{(\infty)}\tilde{\Phi}_{0,k}^{(l+1)}\right]+\cancel{O(\epsilon^2)}\,,\end{aligned} \tag{5.23d}$$

using the following definition for the converged, flux- and volume- weighted macroscopic cross section:

$$\bar{\Sigma}_{a,k}^{(\infty)} \equiv \frac{\sum\limits_{j\in k}\Sigma_{a,j}\phi_{0,j}^{(\infty)}h_j}{\sum\limits_{j\in k}\phi_{0,j}^{(\infty)}h_j}\,. \tag{5.23e}$$

Next, we obtain the linearized form of the neutron current at the right side of the coarse cell by inserting Eq. (5.22b) into Eq. (5.21b):

$$\Phi^{(l+1)}_{1,k+\frac{1}{2}} = -\frac{1}{3\bar{\Sigma}^{(\infty)}_{t,k}\Delta}\left(\tilde{\Phi}^{(l+1)}_{0,k+1} - \tilde{\Phi}^{(l+1)}_{0,k}\right) + \hat{D}^{\left(l+\frac{1}{2}\right)}_{k+\frac{1}{2}}\left(\tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k+1} + \tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k}\right) , \tag{5.24a}$$

where the linearized form of $\hat{D}^{\left(l+\frac{1}{2}\right)}_{k+\frac{1}{2}}$ is defined as

$$\hat{D}^{\left(l+\frac{1}{2}\right)}_{k+\frac{1}{2}} = \tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{1,k+\frac{1}{2}} + \frac{1}{3\bar{\Sigma}^{(\infty)}_{t,k}\Delta}\left(\tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k+1} - \tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k}\right) . \tag{5.24b}$$

Eqs. (5.24) also hold for a given left coarse cell edge, where k is replaced with $(k-1)$. We note that converged quantities with superscript (∞), such as $\bar{\Sigma}^{(\infty)}_{t,k}$, are the same for all k.

With these terms defined, we can now insert the linearized expressions into Eq. (5.21a). After cancelling like terms and rearranging, the following linearized form of the fixed-source CMFD equation is obtained:

$$\begin{aligned}
&\frac{1}{3\bar{\Sigma}^{(\infty)}_{t,k}\Delta}\left(2\tilde{\Phi}^{(l+1)}_{0,k} - \tilde{\Phi}^{(l+1)}_{0,k+1} - \tilde{\Phi}^{(l+1)}_{0,k-1}\right) + \bar{\Sigma}^{(\infty)}_{a,k}\tilde{\Phi}^{(l+1)}_{0,k}\Delta \\
&= \left(\tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{1,k-\frac{1}{2}} - \tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{1,k+\frac{1}{2}}\right) + \frac{1}{3\bar{\Sigma}^{(\infty)}_{t,k}\Delta}\left(2\tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k} - \tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k+1} - \tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k-1}\right) \\
&+\bar{\Sigma}^{(\infty)}_{a,k}\tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k}\Delta - \sum_{j\in k}\Sigma_{a,j}\tilde{\phi}^{\left(l+\frac{1}{2}\right)}_{0,j}h_j .
\end{aligned} \tag{5.25a}$$

Next is the restriction equation defined in Eq. (5.21h). We insert Eqs. (5.22a) and (5.22b) into Eq. (5.21h) to yield:

$$\Phi^{(\infty)}_{0,k} + \epsilon\tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k} = \frac{\sum\limits_{j\in k}\left(\phi^{(\infty)}_{0,j} + \epsilon\tilde{\phi}^{\left(l+\frac{1}{2}\right)}_{0,j}\right)h_j}{\Delta} . \tag{5.25b}$$

After simplifying and cancelling like terms, the following error restriction equation is obtained:

$$\tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k} = \frac{\sum\limits_{j\in k}\tilde{\phi}^{\left(l+\frac{1}{2}\right)}_{0,j}h_j}{\Delta} . \tag{5.25c}$$

Lastly, we consider the prolongation equation defined in Eq. (5.21g). We first insert Eqs. (5.22b) and (5.22c) into Eq. (5.21g), yielding:

$$\phi^{(\infty)}_{0,j} + \epsilon\tilde{\phi}^{(l+1)}_{0,j} = \frac{\phi^{(\infty)}_{0,j}\Phi^{(\infty)}_{0,k} + \epsilon\left(\phi^{(\infty)}_{0,j}\tilde{\Phi}^{(l+1)}_{0,k} + \Phi^{(\infty)}_{0,k}\phi^{\left(l+\frac{1}{2}\right)}_{0,j}\right)}{\Phi^{(\infty)}_{0,k} + \epsilon\tilde{\Phi}^{\left(l+\frac{1}{2}\right)}_{0,k}} . \tag{5.25d}$$

A Taylor expansion around $\epsilon\tilde{\Phi}_{0,k}^{(l+\frac{1}{2})} = 0$ is employed to linearize this equation:

$$\phi_{0,j}^{(\infty)} + \epsilon\tilde{\phi}_{0,j}^{(l+1)} \approx \left[\phi_{0,j}^{(\infty)}\Phi_{0,k}^{(\infty)} + \epsilon\left(\phi_{0,j}^{(\infty)}\tilde{\Phi}_{0,k}^{(l+1)} + \Phi_{0,k}^{(\infty)}\phi_{0,j}^{(l+\frac{1}{2})}\right)\right]\left(\frac{1}{\Phi_{0,k}^{(\infty)}} - \epsilon\frac{\tilde{\Phi}_{0,k}^{(l+\frac{1}{2})}}{\left(\Phi_{0,k}^{(\infty)}\right)^2}\right) . \tag{5.25e}$$

After simplifying and cancelling like terms, the following linearized error prolongation equation is obtained:

$$\tilde{\phi}_{0,j}^{(l+1)} = \tilde{\phi}_{0,j}^{(l+\frac{1}{2})} + \beta_j\left(\tilde{\Phi}_{0,k}^{(l+1)} - \tilde{\Phi}_{0,k}^{(l+\frac{1}{2})}\right) , \forall\ j \in k , \tag{5.25f}$$

where we define β_j as

$$\beta_j \equiv \frac{\phi_{0,j}^{(\infty)}}{\Phi_{0,k}^{(\infty)}} . \tag{5.25g}$$

With this, we have obtained the linearized error forms of the CMFD equations that describe how the errors in the quantities of interest evolve across different outer iterations. Specifically, we have obtained the linearized error form of (i) the main CMFD equation shown in Eq. (5.25a) and (ii) the prolongation equation shown in Eq. (5.25f). These equations will be used for the remaining step of the Fourier analysis, which is the insertion of the Fourier ansatz.

5.3.1.3 Insertion of the Fourier Ansatz

The ansatz substitutions for the CMFD equations are shown below:

$$\tilde{\Phi}_{0,k}^{(l+\frac{1}{2})} = \theta^l A e^{i\omega\bar{\Sigma}_t X_k} , \tag{5.26a}$$

$$\tilde{\Phi}_{0,k}^{(l+1)} = \theta^l R e^{i\omega\bar{\Sigma}_t X_k} , \tag{5.26b}$$

$$\{k \mid k \in \mathbb{N}, 1 \leq k \leq K\} ,$$

$$\{l \mid l \in \mathbb{N}, 0 \leq l < L\} ,$$

which are used in combination with the ansatz equations stated in Eqs. (5.13). In Eqs. (5.26), the parameters A and R are scalar unknowns to be determined. First, using Eqs. (5.12f), (5.13a) and (5.13e), the following spectral relation is obtained for the left edge of a given coarse cell:

$$g_{\text{left}} = \sum_{n=1}^{N} \mu_n w_n a_{n,\frac{1}{2}} . \tag{5.27a}$$

This requires an explicit definition of $a_{n,\frac{1}{2}}$, which can be obtained using Eq. (5.16e) and a unit vector to select a at the left edge of a coarse cell:

$$g_{\text{left}} = \frac{1}{2}\boldsymbol{e}_1^T \sum_{n=1}^{N} \mu_n w_n Y_n^{-1} C \boldsymbol{E} \ , \tag{5.27b}$$

where the unit vector $\boldsymbol{e}_1$ is defined as

$$\boldsymbol{e}_1 \equiv \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} , \quad \boldsymbol{e}_1 \in \mathbb{N}^{p\times 1} \ . \tag{5.27c}$$

Now, we define the operator

$$\boldsymbol{G}_{\text{fs}} \equiv \frac{1}{2}\boldsymbol{e}_1^T \sum_{n=1}^{N} \mu_n w_n Y_n^{-1} C \ , \quad \boldsymbol{G}_{\text{fs}} \in \mathbb{C}^{1\times p} \ , \tag{5.27d}$$

which allows us to define g_{left} as

$$g_{\text{left}} = \boldsymbol{G}_{\text{fs}} \boldsymbol{E} \ . \tag{5.27e}$$

A similar procedure yields the following definition for g_{right}:

$$g_{\text{right}} = e^{i\omega\Lambda} \boldsymbol{G}_{\text{fs}} \boldsymbol{E} \ , \tag{5.27f}$$

which uses the relationship defined in Eq. (5.14m). Thus, we have determined that g_{left} and g_{right} are related by the constant $e^{i\omega\Lambda}$:

$$g_{\text{left}} e^{i\omega\Lambda} = g_{\text{right}} \ , \tag{5.27g}$$

which confirms the relationships established in Eqs. (5.14h) and (5.14p).

With the spectral relation equations for the cell-edge terms defined, we move on to the main CMFD error equation shown in Eq. (5.25a). First, we substitute Eqs. (5.13e), (5.13f), (5.26a), (5.26b), (5.27e) and (5.27f) into Eq. (5.25a). After simplifying and cancelling like terms, the following main CMFD spectral relation equation is obtained:

$$\begin{aligned} &\frac{R}{3\bar{\Sigma}_{t,k}^{(\infty)}\Delta}\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right) + \bar{\Sigma}_{a,k}^{(\infty)} R\Delta = \left(1 - e^{i\omega\Lambda}\right) \boldsymbol{G}_{\text{fs}} \boldsymbol{E} \\ &+ \frac{A}{3\bar{\Sigma}_{t,k}^{(\infty)}\Delta}\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right) + \bar{\Sigma}_{a,k}^{(\infty)} A\Delta - \sum_r \Sigma_{a,r} h_r d_r \ . \end{aligned} \tag{5.27h}$$

We can rewrite the above equation in terms of more fundamental quantities in vector form:

$$\begin{aligned}\frac{\boldsymbol{v}\boldsymbol{\phi}^{(\infty)}}{3\boldsymbol{\tau}\boldsymbol{\phi}^{(\infty)}}R\left(2-e^{i\omega\Lambda}-e^{-i\omega\Lambda}\right)+\frac{\left(\dot{C}\boldsymbol{\tau}\right)^{T}\boldsymbol{\phi}^{(\infty)}}{\boldsymbol{v}\boldsymbol{\phi}^{(\infty)}}R &= \left(1-e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{fs}}\boldsymbol{E} \\ +\frac{\boldsymbol{v}\boldsymbol{\phi}^{(\infty)}}{3\boldsymbol{\tau}\boldsymbol{\phi}^{(\infty)}}A\left(2-e^{i\omega\Lambda}-e^{-i\omega\Lambda}\right)+\frac{\left(\dot{C}\boldsymbol{\tau}\right)^{T}\boldsymbol{\phi}^{(\infty)}}{\boldsymbol{v}\boldsymbol{\phi}^{(\infty)}}A-\left(\dot{C}\boldsymbol{\tau}\right)^{T}\boldsymbol{d}\,,\end{aligned} \tag{5.27i}$$

where the converged coarse-grid cross sections have been recast in terms of the optical thickness, τ, absorption ratio, $\dot{c}$, and volume fraction, v. The following definitions are relevant to Eq. (5.27i):

$$\dot{C}\equiv I-C\ ,\quad \dot{C}\in\mathbb{R}^{p\times p}\ , \tag{5.27j}$$

$$\boldsymbol{\tau}\equiv\begin{pmatrix}\tau_1\\ \tau_2\\ \vdots\\ \tau_p\end{pmatrix}\ ,\quad \boldsymbol{\tau}\in\mathbb{R}^{p\times 1}\ , \tag{5.27k}$$

$$\boldsymbol{v}\equiv\frac{1}{\Delta}\begin{pmatrix}h_1\\ h_2\\ \vdots\\ h_p\end{pmatrix}\ ,\quad \boldsymbol{v}\in\mathbb{R}^{p\times 1}\ , \tag{5.27l}$$

$$\boldsymbol{\phi}_0^{(\infty)}\equiv\begin{pmatrix}\phi_{0,1}^{(\infty)}\\ \phi_{0,2}^{(\infty)}\\ \vdots\\ \phi_{0,p}^{(\infty)}\end{pmatrix}\ ,\quad \boldsymbol{\phi}_0^{(\infty)}\in\mathbb{R}^{p\times 1}\ , \tag{5.27m}$$

where I is the $p\times p$ identity matrix. In this way, the spatial heterogeneity of the periodic lattice is captured in vector and matrix quantities. Solving Eq. (5.27i) for the constant R yields:

$$R=\boldsymbol{F}_{\text{CMFD,fs}}\boldsymbol{E}+A\ , \tag{5.27n}$$

where we define the row vector $\boldsymbol{F}_{\text{CMFD,fs}}$ as

$$\boldsymbol{F}_{\text{CMFD,fs}}\equiv\frac{\left(1-e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{fs}}-\left(\dot{C}\boldsymbol{\tau}\right)^{T}\tilde{H}_{\text{fs}}}{\frac{2\boldsymbol{v}^{T}\boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^{T}\boldsymbol{\phi}_0^{(\infty)}}\left(1-\cos\omega\Lambda\right)+\frac{\left(\dot{C}\boldsymbol{\tau}\right)^{T}\boldsymbol{\phi}_0^{(\infty)}}{\boldsymbol{v}^{T}\boldsymbol{\phi}_0^{(\infty)}}}\ ,\quad \boldsymbol{F}_{\text{CMFD,fs}}\in\mathbb{C}^{1\times p}\ . \tag{5.27o}$$

We note that the following substitution has been made to obtain Eq. (5.27o):

$$\left(2-e^{i\omega\Lambda}-e^{-i\omega\Lambda}\right)=2\left(1-\cos\omega\Lambda\right)\ . \tag{5.27p}$$

Lastly, we obtain the CMFD prolongation spectral relation equation by inserting Eqs. (5.13c), (5.13d), (5.26a) and (5.26b) into Eq. (5.25f), resulting in the following:

$$\theta \boldsymbol{E} = \tilde{S}_{\text{CMFD,fs}} \boldsymbol{E} \ , \tag{5.27q}$$

where the overall error transition matrix $\tilde{S}_{\text{CMFD,fs}}$, containing information from the transport sweep and low-order CMFD solve, is defined as

$$\tilde{S}_{\text{CMFD,fs}} \equiv \tilde{H}_{\text{fs}} + \boldsymbol{\beta}_{\text{CMFD}} \boldsymbol{F}_{\text{CMFD,fs}} \ , \quad \tilde{S}_{\text{CMFD,fs}} \in \mathbb{C}^{p \times p} \ , \tag{5.27r}$$

and the vector $\boldsymbol{\beta}_{\text{CMFD}}$ is defined as the ratio of the converged fine-cell scalar flux to the converged coarse-cell average scalar flux:

$$\boldsymbol{\beta}_{\text{CMFD}} \equiv \frac{\phi_0^{(\infty)}}{\Phi_0^{(\infty)}} \ , \quad \boldsymbol{\beta}_{\text{CMFD}} \in \mathbb{R}^{p \times 1} \ . \tag{5.27s}$$

In Eq. (5.27r), the matrix $\tilde{H}_{\text{fs}}$ is the contribution to the overall error transition matrix from the transport sweep, and $\boldsymbol{\beta}_{\text{CMFD}} \boldsymbol{F}_{\text{CMFD,fs}}$ is the contribution from the CMFD solve. With this, we have obtained an eigenvalue problem (Eq. (5.27q)) that can be solved for ρ.

5.3.2 Eigenvalue Problems

5.3.2.1 Problem Statement

In this section, we reuse many of the same steps from Section 5.3.1 for brevity. Additionally, many of the definitions in Section 5.2.2 are relevant here. The following CMFD equations for eigenvalue problems are used for the Fourier analysis:

$$\left(\Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2},m+1)} - \Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2},m+1)} \right) + \bar{\Sigma}_a^{(l+\frac{1}{2})} \Phi_0^{(l+\frac{1}{2},m+1)} \Delta = \lambda^{(l)} \overline{(\nu\Sigma_f)}_k^{(l+\frac{1}{2})} \Phi_{0,k}^{(l+\frac{1}{2},m)} \Delta \ , \tag{5.28a}$$

$$\begin{aligned} \Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2},m+1)} \equiv & -\tilde{D}_{k+\frac{1}{2}}^{(l+\frac{1}{2})} \left(\Phi_{0,k+1}^{(l+\frac{1}{2},m+1)} - \Phi_{0,k}^{(l+\frac{1}{2},m+1)} \right) \\ & + \hat{D}_{k+\frac{1}{2}}^{(l+\frac{1}{2})} \left(\Phi_{0,k+1}^{(l+\frac{1}{2},m+1)} + \Phi_{0,k}^{(l+\frac{1}{2},m+1)} \right) \ , \end{aligned} \tag{5.28b}$$

$$\overline{(\nu\Sigma_f)}_k^{(l+\frac{1}{2})} \equiv \frac{\sum\limits_{j\in k} (\nu\Sigma_f)_j \, \phi_{0,j}^{(l+\frac{1}{2})} h_j}{\sum\limits_{j\in k} \phi_{0,j}^{(l+\frac{1}{2})} h_j} \ , \tag{5.28c}$$

with the remaining equations being the same as those in Eqs. (5.21). These equations were presented in Section 3.2.2. As a reminder, the outer iteration index is l and the power iteration index is m. The first step is to linearize and perturb these equations around the converged solution.

5.3.2.2 Perturbation & Linearization

For the perturbation step of the Fourier analysis, the same perturbation equations from Eq. (5.18) and Eqs. (5.22) are used. First, we linearize and perturb the CMFD equation shown in Eq. (5.28a). The same linearization techniques from Section 5.3.1.2 are employed in this process (mainly, the use of Taylor expansions), which yields the following error equation from the CMFD equation:

$$\begin{aligned}
&\frac{1}{3\bar{\Sigma}_{t,k}^{(\infty)}\Delta}\left(2\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2},m+1\right)} - \tilde{\Phi}_{0,k+1}^{\left(l+\frac{1}{2},m+1\right)} - \tilde{\Phi}_{0,k-1}^{\left(l+\frac{1}{2},m+1\right)}\right) + \bar{\Sigma}_{a,k}^{(\infty)}\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2},m+1\right)}\Delta \\
&\quad = \left(\tilde{\Phi}_{1,k-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\right) + \frac{1}{3\bar{\Sigma}_{t,k}^{(\infty)}\Delta}\left(2\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{0,k+1}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{0,k-1}^{\left(l+\frac{1}{2}\right)}\right) \\
&\quad +\lambda\sum_{j\in k}\left(\nu\Sigma_f\right)_j\tilde{\phi}_{0,j}^{\left(l+\frac{1}{2}\right)}h_j + \bar{\Sigma}_{a,k}^{(\infty)}\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2},m\right)}\Delta - \sum_{j\in k}\Sigma_{a,j}\tilde{\phi}_{0,j}^{\left(l+\frac{1}{2}\right)}h_j \ .
\end{aligned} \tag{5.29}$$

The remaining CMFD equations are the same as those from the fixed-source analysis listed in Eqs. (5.25), with Eq. (5.29) replacing Eq. (5.25a). These describe how the errors in the iterates evolve across different outer iterations and power iterations. Next, we insert the Fourier ansatz for each quantity of interest.

5.3.2.3 Insertion of the Fourier Ansatz

In this section, we insert the Fourier ansatz for acceleration equations. The ansatz relates the error magnitude for each iterative quantity to the iteration index. For eigenvalue problems, we are working with two different iteration indices: outer iterations indexed with l and power iterations indexed with m. The same ansatz equations as those listed in Eqs. (5.13) and Eq. (5.26a) are used. However, since the coarse-mesh scalar flux is now indexed by the outer and power iterations, Eq. (5.26b) is replaced with the following:

$$\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2},m\right)} = \theta^l R^{(m)} e^{i\omega\bar{\Sigma}_t X_k} \ , \tag{5.30}$$

$$\{k \mid k \in \mathbb{N}, 1 \le k \le K\} \ ,$$

$$\{l \mid l \in \mathbb{N}, 0 \le l < L\} \ ,$$

$$\{m \mid m \in \mathbb{N}, 0 \le m < M\} \ .$$

We note that the superscript m on the R term in Eq. (5.30) is not an exponent, but rather an index that indicates that R is also an iterative quantity.

The ansatz equations are now substituted into the CMFD error equation (Eq. (5.29)), yielding the following spectral relation equation:

$$\begin{aligned}
&R^{(m+1)}\left[\frac{1}{3\bar{\Sigma}_{t,k}^{(\infty)}\Delta}\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right) + \bar{\Sigma}_{a,k}^{(\infty)}\Delta\right] \\
&= \lambda \sum_r \left(\nu\Sigma_f\right)_r h_r d_r + \bar{\Sigma}_{a,k}^{(\infty)} R^{(m)}\Delta - \sum_r \Sigma_{a,r} h_r d_r \\
&+ \frac{A}{3\bar{\Sigma}_{t,k}^{(\infty)}\Delta}\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right) + \left(1 - e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{eig}}\boldsymbol{E} \ ,
\end{aligned} \tag{5.31a}$$

where the row vector $\boldsymbol{G}_{\text{eig}}$ is defined as

$$\boldsymbol{G}_{\text{eig}} \equiv \frac{1}{2}\boldsymbol{e}_1^T \sum_{n=1}^{N} \mu_n w_n Y_n^{-1}\Gamma \ , \quad \boldsymbol{G}_{\text{eig}} \in \mathbb{C}^{1\times p} \ . \tag{5.31b}$$

Just as in Section 5.3.1, this equation can be cast in terms of the previously-defined fundamental mesh properties:

$$\begin{aligned}
&R^{(m+1)}\left[\frac{\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T\boldsymbol{\phi}_0^{(\infty)}}\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right) + \frac{\left(\dot{C}\boldsymbol{\tau}\right)^T\boldsymbol{\phi}_0^{(\infty)}}{\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}}\right] \\
&= \left[\lambda\left(F\boldsymbol{\tau}\right)^T - \left(\dot{C}\boldsymbol{\tau}\right)^T\right]\tilde{H}_{\text{eig}}\boldsymbol{E} + R^{(m)}\frac{\left(\dot{C}\boldsymbol{\tau}\right)^T\boldsymbol{\phi}_0^{(\infty)}}{\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}} \\
&+ \frac{\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T\boldsymbol{\phi}_0^{(\infty)}} A\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right) + \left(1 - e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{eig}}\boldsymbol{E} \ .
\end{aligned} \tag{5.31c}$$

To solve for R, we first rearrange Eq. (5.31c) for $R^{(m+1)}$:

$$R^{(m+1)} = \frac{\alpha_1 R^{(m)} + \alpha_2}{\alpha_3} \ , \tag{5.31d}$$

where we have defined the following constants:

$$\alpha_1 \equiv \frac{\left(\dot{C}\boldsymbol{\tau}\right)^T\boldsymbol{\phi}_0^{(\infty)}}{\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}} \ , \tag{5.31e}$$

$$\alpha_2 \equiv \left(1 - e^{i\omega\Lambda}\right) \boldsymbol{G}_{\text{eig}} \boldsymbol{E} + 2A\left(1 - \cos\omega\Lambda\right) \frac{\boldsymbol{v}^T \boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T \boldsymbol{\phi}_0^{(\infty)}} + \left[\lambda \left(F\boldsymbol{\tau}\right)^T - \left(\dot{C}\boldsymbol{\tau}\right)^T\right] \tilde{H}_{\text{eig}} \boldsymbol{E} \ , \tag{5.31f}$$

$$\alpha_3 \equiv \frac{2\boldsymbol{v}^T \boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T \boldsymbol{\phi}_0^{(\infty)}} \left(1 - \cos\omega\Lambda\right) + \frac{\left(\dot{C}\boldsymbol{\tau}\right)^T \boldsymbol{\phi}_0^{(\infty)}}{\boldsymbol{v}^T \boldsymbol{\phi}_0^{(\infty)}} \ . \tag{5.31g}$$

Equation (5.31d) is a recurrence relation for $R^{(m)}$ with the following solution:

$$R^{(m)} = \frac{\alpha_2 \left(\frac{\alpha_1}{\alpha_3}\right)^m + \alpha_1 R^{(0)} \left(\frac{\alpha_1}{\alpha_3}\right)^m - \alpha_3 R^{(0)} \left(\frac{\alpha_1}{\alpha_3}\right)^m - \alpha_2}{\alpha_1 - \alpha_3} \ . \tag{5.31h}$$

To complete the calculation for R, we assume that the low-order problem is always "fully converged". This corresponds to taking the limit of $R^{(m)}$ as $m \to \infty$. Because the ratio $\frac{\alpha_1}{\alpha_3} < 1$ for all allowable Fourier frequencies, the following holds:

$$\lim_{m\to\infty} \left(\frac{\alpha_1}{\alpha_3}\right)^m = \lim_{m\to\infty} \left(\frac{\frac{\left(\dot{C}\boldsymbol{\tau}\right)^T \boldsymbol{\phi}_0^{(\infty)}}{\boldsymbol{v}^T \boldsymbol{\phi}_0^{(\infty)}}}{\frac{2\boldsymbol{v}^T \boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T \boldsymbol{\phi}_0^{(\infty)}} \left(1 - \cos\omega\Lambda\right) + \frac{\left(\dot{C}\boldsymbol{\tau}\right)^T \boldsymbol{\phi}_0^{(\infty)}}{\boldsymbol{v}^T \boldsymbol{\phi}_0^{(\infty)}}} \right)^m = 0 \ . \tag{5.31i}$$

Therefore, upon making this assumption, we obtain:

$$\begin{aligned}
\lim_{m\to\infty} R^{(m)} &\equiv R^{(\infty)} && (5.31\text{j}) \\
&= \frac{\alpha_2}{\alpha_3 - \alpha_1} && (5.31\text{k}) \\
&= \frac{\left(1 - e^{i\omega\Lambda}\right) \boldsymbol{G}_{\text{eig}} \boldsymbol{E} + \left[\lambda \left(F\boldsymbol{\tau}\right)^T - \left(\dot{C}\boldsymbol{\tau}\right)^T\right] \tilde{H}_{\text{eig}} \boldsymbol{E}}{\frac{2\boldsymbol{v}^T \boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T \boldsymbol{\phi}_0^{(\infty)}} \left(1 - \cos\omega\Lambda\right)} + A \ . && (5.31\text{l})
\end{aligned}$$

Now, we can insert $R^{(\infty)}$ into the spectral relation form of the prolongation equation, shown below:

$$\theta \boldsymbol{E} = \tilde{H}_{\text{eig}} \boldsymbol{E} + \boldsymbol{\beta}_{\text{CMFD}} \left(R^{(\infty)} - A\right) \ , \tag{5.31m}$$

which yields the following eigenvalue problem:

$$\theta \boldsymbol{E} = \tilde{S}_{\text{CMFD,eig}} \boldsymbol{E} \ . \tag{5.31n}$$

We note that the parameter A cancels upon substituting Eq. (5.31l) into Eq. (5.31m). Here the overall error transition matrix $\tilde{S}_{\text{CMFD,eig}}$ is defined as

$$\tilde{S}_{\text{CMFD,eig}} \equiv \tilde{H}_{\text{eig}} + \boldsymbol{\beta}_{\text{CMFD}} \boldsymbol{F}_{\text{CMFD,eig}} \ , \quad \tilde{S}_{\text{CMFD,eig}} \in \mathbb{C}^{p\times p} \ , \tag{5.31o}$$

and the row vector $\boldsymbol{F}_{\text{CMFD,eig}}$ is defined as

$$\boldsymbol{F}_{\text{CMFD,eig}} \equiv \frac{\left(1 - e^{i\omega\Lambda}\right) \boldsymbol{G}_{\text{eig}} + \left[\lambda \left(F\boldsymbol{\tau}\right)^T - \left(\dot{C}\boldsymbol{\tau}\right)^T\right] \tilde{H}_{\text{eig}}}{\frac{2\boldsymbol{v}^T \boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T \boldsymbol{\phi}_0^{(\infty)}} \left(1 - \cos\omega\Lambda\right)} \ , \quad \boldsymbol{F}_{\text{CMFD,eig}} \in \mathbb{C}^{1\times p} \ . \tag{5.31p}$$

With this, we have obtained the spectral relation form of all relevant equations. The spectral radius for CMFD applied to eigenvalue problems can be obtained by solving the eigenvalue problem in Eq. (5.31n).

5.4 LDA Fourier Analysis

Here, we present the Fourier analysis procedure for spatially-heterogeneous cases for both fixed-source and eigenvalue problem types accelerated with LDA. Many of the results from Section 5.2 are used here. Because LDA is linear with respect to the scalar flux, it does not need to be linearized before the introduction of the Fourier ansatz.

5.4.1 Fixed-Source Problems

5.4.1.1 Problem Statement

We begin with the monoenergetic LDA equations for a periodic, fixed-source problem:

$$\begin{aligned} &\tilde{D}\left(2\Phi_{0,k}^{(l+1)} - \Phi_{0,k-1}^{(l+1)} - \Phi_{0,k+1}^{(l+1)}\right) + \bar{\Sigma}_a \Phi_{0,k}^{(l+1)} \Delta \\ &= \bar{Q}\Delta + \left(\Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2})} - \Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})}\right) + \tilde{D}\left(2\Phi_{0,k}^{(l+\frac{1}{2})} - \Phi_{0,k-1}^{(l+\frac{1}{2})} - \Phi_{0,k+1}^{(l+\frac{1}{2})}\right) \\ &\quad + \bar{\Sigma}_a \Phi_{0,k}^{(l+\frac{1}{2})} \Delta - \sum_{j=1}^{p} \Sigma_{a,j} \phi_{0,j}^{(l+\frac{1}{2})} h_j \ , \end{aligned} \tag{5.32a}$$

$$\tilde{D} \equiv \frac{1}{3\sum\limits_{j\in k} \Sigma_{t,j} h_j} \ , \tag{5.32b}$$

$$\bar{\Sigma}_a \equiv \frac{\sum\limits_{j\in k} \Sigma_{a,j} h_j}{\sum\limits_{j\in k} h_j} \ , \tag{5.32c}$$

$$\bar{Q} \equiv \frac{\sum\limits_{j\in k} q_j h_j}{\Delta} \ , \tag{5.32d}$$

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{(l+1)} + \left(\Phi_{0,k}^{(l+1)} - \Phi_{0,k}^{(l+\frac{1}{2})}\right) \ , \forall\ j \in k \ , \tag{5.32e}$$

$$\Phi_{0,k}^{\left(l+\frac{1}{2}\right)} \equiv \frac{\sum\limits_{j\in k} \phi_{0,j}^{\left(l+\frac{1}{2}\right)} h_j}{\Delta} \ , \tag{5.32f}$$

$$\Phi_{1,\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} = \Phi_{1,K+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \ , \tag{5.32g}$$

$$\Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \equiv \sum_{n=1}^{N} \mu_n w_n \psi_{n,pk+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} \ , \quad \{k \mid k \in \mathbb{N}, 0 \le k \le K\} \ . \tag{5.32h}$$

These equations were presented in Section 4.3.1. Next, we perturb these equations around the converged solution.

5.4.1.2 Perturbation

Because LDA is already linear with respect to the scalar flux, we do not need to linearize the set of equations (as was necessary for CMFD in Section 5.3). The perturbation equations are the same as those used for CMFD, and are listed in Eqs. (5.22). Upon insertion of Eqs. (5.22) into Eqs. (5.32a) and (5.32e), we obtain the following LDA error equations:

$$\begin{aligned}
&\frac{1}{3\Lambda}\left(2\tilde{\Phi}_{0,k}^{(l+1)} - \tilde{\Phi}_{0,k-1}^{(l+1)} - \tilde{\Phi}_{0,k+1}^{(l+1)}\right) + \bar{\Sigma}_a \tilde{\Phi}_{0,k}^{(l+1)} \Delta = \left(\tilde{\Phi}_{1,k-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\right) \\
&+\frac{1}{3\Lambda}\left(2\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{0,k-1}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{0,k+1}^{\left(l+\frac{1}{2}\right)}\right) + \bar{\Sigma}_a \tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2}\right)} \Delta - \sum_{j=1}^{p} \Sigma_{a,j} \tilde{\phi}_{0,j}^{\left(l+\frac{1}{2}\right)} h_j \ ,
\end{aligned} \tag{5.33a}$$

$$\tilde{\phi}_{0,j}^{(l+1)} = \tilde{\Phi}_{0,j}^{\left(l+\frac{1}{2}\right)} + \left(\tilde{\Phi}_{0,k}^{(l+1)} - \tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2}\right)}\right) \ , \forall\ j \in k \ . \tag{5.33b}$$

These equations describe how errors in the quantities of interest evolve across iterations. Next, we introduce the Fourier ansatz.

5.4.1.3 Insertion of the Fourier Ansatz

The next step is to cast the error equations in terms of the error reduction rate, Fourier frequency, and a spatial modulation function. We use the same Fourier ansatz equations listed in Eqs. (5.13) and (5.26). After inserting these equations into Eqs. (5.33), we obtain the following LDA spectral relation equation:

$$\begin{aligned}
&\frac{R}{3\Lambda}\left(2e^{i\omega\bar{\Sigma}_t X_k} - e^{i\omega\bar{\Sigma}_t X_{k-1}} - e^{i\omega\bar{\Sigma}_t X_{k+1}}\right) + \dot{\boldsymbol{c}}^T \boldsymbol{\tau} R e^{i\omega\bar{\Sigma}_t X_k} \\
&= \left(g_{\text{left}} e^{i\omega\bar{\Sigma}_t X_k} - g_{\text{right}} e^{i\omega\bar{\Sigma}_t X_k}\right) + \frac{A}{3\Lambda}\left(2e^{i\omega\bar{\Sigma}_t X_k} - e^{i\omega\bar{\Sigma}_t X_{k-1}} - e^{i\omega\bar{\Sigma}_t X_{k+1}}\right) \\
&\quad +\dot{\boldsymbol{c}}^T \boldsymbol{\tau} A e^{i\omega\bar{\Sigma}_t X_k} - \left(\dot{C}\boldsymbol{\tau}\right)^T e^{i\omega\bar{\Sigma}_t X_k} \tilde{H}_{\text{fs}} \boldsymbol{E} \ .
\end{aligned} \tag{5.34a}$$

Dividing Eq. (5.34a) by $e^{i\omega\bar{\Sigma}_t X_k}$ and substituting in Eqs. (5.27e) and (5.27f) results in

$$\begin{aligned} &R\left[\frac{1}{3\Lambda}\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right) + \dot{\boldsymbol{c}}^T\boldsymbol{\tau}\right] \\ &= \left(1 - e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{fs}}\boldsymbol{E} + A\left[\frac{1}{3\Lambda}\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right) + \dot{\boldsymbol{c}}^T\boldsymbol{\tau}\right] - \left(\dot{C}\boldsymbol{\tau}\right)^T \tilde{H}_{\text{fs}}\boldsymbol{E} \ . \end{aligned} \tag{5.34b}$$

Then, rearranging for R yields:

$$R = \boldsymbol{F}_{\text{LDA,fs}}\boldsymbol{E} + A \ , \tag{5.34c}$$

where the row vector $\boldsymbol{F}_{\text{LDA,fs}}$ is defined as

$$\boldsymbol{F}_{\text{LDA,fs}} \equiv \frac{\left(1 - e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{fs}} - \left(\dot{C}\boldsymbol{\tau}\right)^T \tilde{H}_{\text{fs}}}{\frac{2}{3\Lambda}\left(1 - \cos\omega\Lambda\right) + \dot{\boldsymbol{c}}^T\boldsymbol{\tau}} \ , \quad \boldsymbol{F}_{\text{LDA,fs}} \in \mathbb{C}^{1\times p} \ . \tag{5.34d}$$

Lastly, we substitute the Fourier ansatz into Eq. (5.32e), resulting in

$$\theta^{l+1}\boldsymbol{E}e^{i\omega\bar{\Sigma}_t X_k} = \theta^l\boldsymbol{d}e^{i\omega\bar{\Sigma}_t X_k} + \boldsymbol{\beta}_{\text{LDA}}\left(\theta^l Re^{i\omega\bar{\Sigma}_t X_k} - \theta^l Ae^{i\omega\bar{\Sigma}_t X_k}\right) \tag{5.34e}$$

where we define $\boldsymbol{\beta}_{\text{LDA}}$ as

$$\boldsymbol{\beta}_{\text{LDA}} \equiv \boldsymbol{1} \ , \quad \boldsymbol{\beta}_{\text{LDA}} \in \mathbb{N}^{p\times 1} \ . \tag{5.34f}$$

After dividing by $\theta^l e^{i\omega\bar{\Sigma}_t X_k}$ and substituting in Eqs. (5.16h) and (5.34c), we obtain the following eigenvalue problem:

$$\theta\boldsymbol{E} = \tilde{S}_{\text{LDA,fs}}\boldsymbol{E} \ , \tag{5.34g}$$

where the overall error transition matrix $\tilde{S}_{\text{LDA,fs}}$ is defined as

$$\tilde{S}_{\text{LDA,fs}} = \tilde{H}_{\text{fs}} + \boldsymbol{\beta}_{\text{LDA}}\boldsymbol{F}_{\text{LDA,fs}} \ , \quad \tilde{S}_{\text{LDA,fs}} \in \mathbb{C}^{p\times p} \ . \tag{5.34h}$$

This eigenvalue problem can be used to solve for the spectral radius ρ.

5.4.2 Eigenvalue Problems

5.4.2.1 Problem Statement

In this section, we again utilize some of the work already presented in Section 5.3.2 to avoid unnecessary repetition. The LDA equations for an eigenvalue problem are the same as Eqs. (5.32), with the exception of Eq. (5.32a), which is replaced with the following LDA equation:

$$\begin{aligned}
&\tilde{D}\left(2\Phi_{0,k}^{\left(l+\frac{1}{2},m+1\right)} - \Phi_{0,k-1}^{\left(l+\frac{1}{2},m+1\right)} - \Phi_{0,k+1}^{\left(l+\frac{1}{2},m+1\right)}\right) + \bar{\Sigma}_a \Phi_0^{\left(l+\frac{1}{2},m+1\right)}\Delta \\
&= \lambda_d \overline{\nu\Sigma_f}\Phi_{0,k}^{\left(l+\frac{1}{2},m\right)}\Delta + \left(\Phi_{1,k-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} - \Phi_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\right) + \tilde{D}\left(2\Phi_{0,k}^{\left(l+\frac{1}{2}\right)} - \Phi_{0,k-1}^{\left(l+\frac{1}{2}\right)} - \Phi_{0,k+1}^{\left(l+\frac{1}{2}\right)}\right) \\
&+\bar{\Sigma}_a \Phi_{0,k}^{\left(l+\frac{1}{2}\right)}\Delta - \sum_{j=1}^{p}\Sigma_{a,j}\phi_{0,j}^{\left(l+\frac{1}{2}\right)}h_j + \lambda_t^{(l+1)}\sum_{j=1}^{p}\left(\nu\Sigma_f\right)_j\phi_{0,j}^{\left(l+\frac{1}{2}\right)}h_j - \lambda_d\overline{\nu\Sigma_f}\Phi_{0,k}^{\left(l+\frac{1}{2}\right)}\Delta \ .
\end{aligned} \tag{5.35}$$

This equation was presented in Section 4.3.2. Next, we perturb the LDA equations around the converged solution.

5.4.2.2 Perturbation

Here, we perturb the LDA equations about the converged solution by substituting Eq. (5.18) and Eqs. (5.22) into Eq. (5.35), yielding:

$$\begin{aligned}
&\frac{1}{3\Lambda}\left(2\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2},m+1\right)} - \tilde{\Phi}_{0,k-1}^{\left(l+\frac{1}{2},m+1\right)} - \tilde{\Phi}_{0,k+1}^{\left(l+\frac{1}{2},m+1\right)}\right) + \bar{\Sigma}_a\tilde{\Phi}_0^{\left(l+\frac{1}{2},m+1\right)}\Delta = \lambda_d \boldsymbol{f}^T\boldsymbol{\tau}\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2},m\right)} \\
&+\left(\tilde{\Phi}_{1,k-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\right) + \frac{1}{3\Lambda}\left(2\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{0,k-1}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{0,k+1}^{\left(l+\frac{1}{2}\right)}\right) \\
&+\lambda\sum_{j=1}^{p}\left(\nu\Sigma_f\right)_j\tilde{\phi}_{0,j}^{\left(l+\frac{1}{2}\right)}h_j - \sum_{j=1}^{p}\Sigma_{a,j}\tilde{\phi}_{0,j}^{\left(l+\frac{1}{2}\right)}h_j \ .
\end{aligned} \tag{5.36a}$$

The remaining LDA error equations are the same as those in Eqs. (5.33), with Eq. (5.36a) replacing Eq. (5.33a). By casting Eq. (5.36a) in terms of previously-defined fundamental mesh properties, we obtain:

$$\begin{aligned}
&\frac{1}{3\Lambda}\left(2\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2},m+1\right)} - \tilde{\Phi}_{0,k-1}^{\left(l+\frac{1}{2},m+1\right)} - \tilde{\Phi}_{0,k+1}^{\left(l+\frac{1}{2},m+1\right)}\right) + \dot{\boldsymbol{c}}^T\boldsymbol{\tau}\tilde{\Phi}_0^{\left(l+\frac{1}{2},m+1\right)} = \dot{\boldsymbol{c}}^T\boldsymbol{\tau}\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2},m\right)} \\
&+\left(\tilde{\Phi}_{1,k-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{1,k+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\right) + \frac{1}{3\Lambda}\left(2\tilde{\Phi}_{0,k}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{0,k-1}^{\left(l+\frac{1}{2}\right)} - \tilde{\Phi}_{0,k+1}^{\left(l+\frac{1}{2}\right)}\right) \\
&+\lambda\left(F\boldsymbol{\tau}\right)^T\tilde{\boldsymbol{\phi}}_0^{\left(l+\frac{1}{2}\right)} - \left(\dot{C}\boldsymbol{\tau}\right)^T\tilde{\boldsymbol{\phi}}_{0,j}^{\left(l+\frac{1}{2}\right)} \ .
\end{aligned} \tag{5.36b}$$

Lastly, we insert the Fourier ansatz.

5.4.2.3 Insertion of the Fourier Ansatz

The same Fourier ansatz equations that were used for the CMFD eigenvalue case are used here. These are given in Eqs. (5.13), Eq. (5.26a), and Eq. (5.30). Substituting the ansatz into Eq. (5.36b) yields:

$$\begin{aligned}\tilde{D}\left(2R^{(m+1)}e^{i\omega\bar{\Sigma}_t X_k} - R^{(m+1)}e^{i\omega\bar{\Sigma}_t X_{k-1}} - R^{(m+1)}e^{i\omega\bar{\Sigma}_t X_{k+1}}\right) + \dot{\boldsymbol{c}}^T\boldsymbol{\tau}R^{(m+1)}e^{i\omega\bar{\Sigma}_t X_k} \\ = \left(g_{\text{left}}e^{i\omega\bar{\Sigma}_t X_k} - g_{\text{right}}e^{i\omega\bar{\Sigma}_t X_k}\right) + \tilde{D}\left(2Ae^{i\omega\bar{\Sigma}_t X_k} - Ae^{i\omega\bar{\Sigma}_t X_{k-1}} - Ae^{i\omega\bar{\Sigma}_t X_{k+1}}\right) \\ + \dot{\boldsymbol{c}}^T\boldsymbol{\tau}R^{(m)}e^{i\omega\bar{\Sigma}_t X_k} + \left[\lambda\left(F\boldsymbol{\tau}\right)^T - \left(\dot{C}\boldsymbol{\tau}\right)^T\right]\tilde{H}_{\text{eig}}\boldsymbol{E}\ .\end{aligned} \tag{5.37a}$$

Dividing by $e^{i\omega\bar{\Sigma}_t X_k}$ and rearranging for $R^{(m+1)}$ results in the following recurrence relation:

$$R^{(m+1)} = \frac{\alpha_1 R^{(m)} + \alpha_2}{\alpha_3}\ , \tag{5.37b}$$

where

$$\alpha_1 = \dot{\boldsymbol{c}}^T\boldsymbol{\tau}\ , \tag{5.37c}$$

$$\alpha_2 \equiv \left(1 - e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{eig}}\boldsymbol{E} + A\left[\tilde{D}\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right)\right] + \left[\left(\lambda F - \dot{C}\right)\boldsymbol{\tau}\right]^T\tilde{H}_{\text{eig}}\boldsymbol{E}\ , \tag{5.37d}$$

$$\alpha_3 \equiv \tilde{D}\left(2 - e^{i\omega\Lambda} - e^{-i\omega\Lambda}\right) + \dot{\boldsymbol{c}}^T\boldsymbol{\tau}\ . \tag{5.37e}$$

Following the same procedure as that done in Section 5.3.2 to solve the recurrence relation and take the limit as $m \to \infty$ yields the following for $R^{(\infty)}$:

$$R^{(\infty)} = \boldsymbol{F}_{\text{LDA,eig}}\boldsymbol{E} + A\ , \tag{5.37f}$$

where the row vector $\boldsymbol{F}_{\text{LDA,eig}}$ is defined as

$$\boldsymbol{F}_{\text{LDA,eig}} \equiv \frac{\left(1 - e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{eig}} + \left[\left(\lambda F - \dot{C}\right)\boldsymbol{\tau}\right]^T\tilde{H}_{\text{eig}}}{2\tilde{D}\left(1 - \cos\omega\Lambda\right)}\ , \quad \boldsymbol{F}_{\text{LDA,eig}} \in \mathbb{C}^{1\times p}\ . \tag{5.37g}$$

With these definitions, the overall error transition matrix $\tilde{S}_{\text{LDA,eig}}$ can be constructed as

$$\tilde{S}_{\text{LDA,eig}} \equiv \tilde{H}_{\text{eig}} + \boldsymbol{\beta}_{\text{LDA}}\boldsymbol{F}_{\text{LDA,eig}}\ , \quad \tilde{S}_{\text{LDA,eig}} \in \mathbb{C}^{p\times p}\ . \tag{5.37h}$$

The following eigenvalue problem can be solved for the spectral radius ρ:

$$\theta\boldsymbol{E} = \tilde{S}_{\text{LDA,eig}}\boldsymbol{E}\ . \tag{5.37i}$$

5.5 Results & Discussion

Now, we discuss the results of the Fourier analysis in Sections 5.2 to 5.4. First, the equations are summarized in a concise form. Then, the spectral relation equations for LDA and CMFD are compared for cases that are spatially homogeneous. Finally, the spectral relation equations are used to compare the iterative properties of LDA and CMFD for homogeneous and heterogeneous problems.

5.5.1 Summary of Fourier Analysis

The final spectral relation equations of the Fourier analysis are presented here. These results are for spatially-periodic cases, with p fine cells per coarse cell. Discrete ordinates is used for the transport problem, with N discrete angles. For the spatial closure scheme, we use diamond-difference. Results from the four following combinations are presented:

1. CMFD for fixed-source problems,
2. CMFD for eigenvalue problems,
3. LDA for fixed-source problems, and
4. LDA for eigenvalue problems.

In each of the four cases, we seek to solve the following eigenvalue problem:

$$\theta \boldsymbol{E} = \tilde{S} \boldsymbol{E} \,, \tag{5.38a}$$

for the θ eigenvalues. The spectral radius is the largest magnitude eigenvalue of the overall error transition matrix $\tilde{S}$ over all discrete allowable Fourier frequencies:

$$\rho \equiv \max_{\omega \in \Omega} |\theta(\omega)| \,, \tag*{(Eq. (5.8) revisited)}$$

where the set Ω is defined as

$$\Omega = \left\{ \omega \mid \omega = \frac{2\pi k}{\Lambda K} \,, k \in \mathbb{N} \,, 1 \leq k < K \right\} \,. \tag{5.38b}$$

In all accelerated cases, $\tilde{S}$ takes the following form:

$$\tilde{S} \equiv \tilde{H} + \boldsymbol{\beta F} \,, \quad \tilde{S} \in \mathbb{C}^{p \times p} \,, \tag{5.38c}$$

and the different forms of $\tilde{S}$ for each of the four cases are shown in Table 5.1.

We now focus on the form of $\tilde{H}$, which is the component of $\tilde{S}$ that comes from the transport sweep. ($\tilde{H}$ is independent of the acceleration method.) For fixed-source problems,

Table 5.1: Forms of the Overall Error Transition Matrix $\tilde{S}$

Acceleration Method	Fixed-Source	Eigenvalue
CMFD	$\tilde{H}_{\text{fs}} + \boldsymbol{\beta}_{\text{CMFD}} \boldsymbol{F}_{\text{CMFD,fs}}$	$\tilde{H}_{\text{eig}} + \boldsymbol{\beta}_{\text{CMFD}} \boldsymbol{F}_{\text{CMFD,eig}}$
LDA	$\tilde{H}_{\text{fs}} + \boldsymbol{\beta}_{\text{LDA}} \boldsymbol{F}_{\text{LDA,fs}}$	$\tilde{H}_{\text{eig}} + \boldsymbol{\beta}_{\text{LDA}} \boldsymbol{F}_{\text{LDA,eig}}$

$\tilde{H}$ is defined as

$$\tilde{H}_{\text{fs}} \equiv \frac{1}{2} \sum_{n=1}^{N} w_n \bar{Y} Y_n^{-1} C \ , \quad \tilde{H}_{\text{fs}} \in \mathbb{C}^{p \times p} \ , \qquad \text{(Eq. (5.16i) revisited)}$$

with the following related definitions:

$$Y_n \equiv \bar{Y} + \begin{pmatrix} -\frac{\mu_n}{\tau_1} & \frac{\mu_n}{\tau_1} & 0 & \cdots & 0 \\ 0 & -\frac{\mu_n}{\tau_2} & \frac{\mu_n}{\tau_2} & 0 & \cdots \\ & & \vdots & \ddots & \\ \frac{\mu_n}{\tau_p} e^{i\omega\Lambda} & 0 & \cdots & 0 & -\frac{\mu_n}{\tau_p} \end{pmatrix} , \quad Y_n \in \mathbb{C}^{p \times p} \ , \qquad \text{(Eq. (5.16b) revisited)}$$

$$\bar{Y} \equiv \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & \cdots & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & \cdots \\ & & \vdots & \ddots & \\ \frac{1}{2} e^{i\omega\Lambda} & 0 & \cdots & 0 & \frac{1}{2} \end{pmatrix} , \quad \bar{Y} \in \mathbb{C}^{p \times p} \ , \qquad \text{(Eq. (5.15b) revisited)}$$

$$C \equiv \begin{pmatrix} c_1 & 0 & 0 & \cdots & 0 \\ 0 & c_2 & 0 & \cdots & 0 \\ 0 & 0 & \vdots & \ddots & \\ 0 & 0 & \cdots & 0 & c_p \end{pmatrix} , \quad C \in \mathbb{R}^{p \times p} \ , \qquad \text{(Eq. (5.16d) revisited)}$$

$$\Lambda \equiv \sum_{r=1}^{p} \Sigma_{t,r} h_r \ , \qquad \text{(Eq. (5.14n) revisited)}$$

$$c_r \equiv \frac{\Sigma_{s,r}}{\Sigma_{t,r}} \ , \qquad \text{(Eq. (5.12d) revisited)}$$

$$\tau_r \equiv \Sigma_{t,r} h_r \ . \qquad \text{(Eq. (5.12c) revisited)}$$

For eigenvalue problems, $\tilde{H}$ takes the following slightly different form:

$$\tilde{H}_{\text{eig}} \equiv \frac{1}{2} \sum_{n=1}^{N} w_n \bar{Y} Y_n^{-1} \Gamma \ , \quad \tilde{H}_{\text{eig}} \in \mathbb{C}^{p \times p} \ , \qquad \text{(Eq. (5.20b) revisited)}$$

with the following definition of Γ:

$$\Gamma \equiv C + \lambda F \ , \quad \Gamma \in \mathbb{R}^{p\times p} \ , \tag{Eq. (5.20c) revisited}$$

where λ is the transport eigenvalue and

$$F \equiv \begin{pmatrix} f_1 & 0 & 0 & \cdots & 0 \\ 0 & f_2 & 0 & \cdots & 0 \\ 0 & 0 & \vdots & \ddots & \\ 0 & 0 & \cdots & 0 & f_p \end{pmatrix} \ , \quad F \in \mathbb{R}^{p\times p} \ , \tag{Eq. (5.20d) revisited}$$

$$f_r \equiv \frac{(\nu\Sigma_f)_r}{\Sigma_{t,r}} \ . \tag{Eq. (5.19c) revisited}$$

Next, we define $\boldsymbol{\beta}$ for each acceleration method. This vector comes from the prolongation equation of the low-order problem. The form of $\boldsymbol{\beta}$ is independent of the problem type (fixed-source or eigenvalue), and differs only between LDA and CMFD. For LDA, $\boldsymbol{\beta}$ is defined as

$$\boldsymbol{\beta}_{\text{LDA}} \equiv \mathbf{1} \ , \tag{Eq. (5.34f) revisited}$$

where

$$\mathbf{1} \equiv \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \ , \quad \mathbf{1} \in \mathbb{N}^{p\times 1} \ . \tag{5.41a}$$

For CMFD, $\boldsymbol{\beta}$ takes the form:

$$\boldsymbol{\beta}_{\text{CMFD}} \equiv \frac{\boldsymbol{\phi}_0^{(\infty)}}{\Phi_0^{(\infty)}} \ , \tag{Eq. (5.27s) revisited}$$

and represents the relative scalar flux distribution in the coarse cell. In CMFD variants with higher-order spatial prolongation, $\boldsymbol{\beta}$ will differ. $\boldsymbol{\phi}_0^{(\infty)}$ is the converged fine-mesh scalar flux:

$$\boldsymbol{\phi}_0^{(\infty)} \equiv \begin{pmatrix} \phi_{0,1}^{(\infty)} \\ \phi_{0,2}^{(\infty)} \\ \vdots \\ \phi_{0,p}^{(\infty)} \end{pmatrix} \ , \quad \boldsymbol{\phi}_0^{(\infty)} \in \mathbb{R}^{p\times 1} \ , \tag{Eq. (5.27m) revisited}$$

and $\Phi_0^{(\infty)}$ is the converged coarse-mesh scalar flux:

$$\Phi_0^{(\infty)} \equiv \frac{\sum\limits_{r=1}^{p} \phi_{0,r}^{(\infty)} h_r}{\Delta} \,. \tag{Eq. (5.23b) revisited}$$

Lastly, we define the forms of the row vector $\boldsymbol{F}$, which takes a different form for each of the four combinations of problem type and acceleration method. This term comes from the solution of the low-order acceleration equations. For fixed-source CMFD, $\boldsymbol{F}_{\text{CMFD,fs}}$ is defined as

$$\boldsymbol{F}_{\text{CMFD,fs}} \equiv \frac{\left(1 - e^{i\omega\Lambda}\right) \boldsymbol{G}_{\text{fs}} - \left(\dot{C}\boldsymbol{\tau}\right)^T \tilde{H}_{\text{fs}}}{\frac{2\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T\boldsymbol{\phi}_0^{(\infty)}}\left(1 - \cos\omega\Lambda\right) + \frac{\left(\dot{C}\boldsymbol{\tau}\right)^T\boldsymbol{\phi}_0^{(\infty)}}{\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}}} \,, \quad \boldsymbol{F}_{\text{CMFD,fs}} \in \mathbb{C}^{1\times p} \,, \tag{Eq. (5.27o) revisited}$$

where the row vector $\boldsymbol{G}_{\text{fs}}$ for fixed-source problems is defined as

$$\boldsymbol{G}_{\text{fs}} \equiv \frac{1}{2}\boldsymbol{e}_1^T \sum_{n=1}^{N} \mu_n w_n Y_n^{-1} C \,, \quad \boldsymbol{G}_{\text{fs}} \in \mathbb{C}^{1\times p} \,, \tag{Eq. (5.27d) revisited}$$

and the following supporting definitions are made:

$$\dot{C} \equiv I - C \,, \quad \dot{C} \in \mathbb{R}^{p\times p} \,, \tag{Eq. (5.27j) revisited}$$

$$\boldsymbol{\tau} \equiv \begin{pmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_p \end{pmatrix} \,, \quad \boldsymbol{\tau} \in \mathbb{R}^{p\times 1} \,, \tag{Eq. (5.27k) revisited}$$

$$\boldsymbol{v} \equiv \frac{1}{\Delta} \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_p \end{pmatrix} \,, \quad \boldsymbol{v} \in \mathbb{R}^{p\times 1} \,, \tag{Eq. (5.27l) revisited}$$

$$\boldsymbol{e}_1 \equiv \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \,, \quad \boldsymbol{e}_1 \in \mathbb{N}^{p\times 1} \,. \tag{Eq. (5.27c) revisited}$$

For CMFD applied to eigenvalue problems, $\boldsymbol{F}_{\text{CMFD,eig}}$ takes the form

$$\boldsymbol{F}_{\text{CMFD,eig}} \equiv \frac{\left(1-e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{eig}} + \left[\left(\lambda F - \dot{C}\right)\boldsymbol{\tau}\right]^T \tilde{H}_{\text{eig}}}{\frac{2\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T\boldsymbol{\phi}_0^{(\infty)}}\left(1-\cos\omega\Lambda\right)} \,, \quad \boldsymbol{F}_{\text{CMFD,eig}} \in \mathbb{C}^{1\times p} \,,$$

(Eq. (5.31p) revisited)

where the row vector $\boldsymbol{G}_{\text{eig}}$ for eigenvalue problems is defined as

$$\boldsymbol{G}_{\text{eig}} \equiv \frac{1}{2}\boldsymbol{e}_1^T \sum_{n=1}^{N} \mu_n w_n Y_n^{-1}\Gamma \,, \quad \boldsymbol{G}_{\text{eig}} \in \mathbb{C}^{1\times p} \,. \qquad \text{(Eq. (5.31b) revisited)}$$

Table 5.2 summarizes the forms of $\tilde{H}$ and $\boldsymbol{G}$ for each problem type.

Table 5.2: Forms of $\tilde{H}$ and $\boldsymbol{G}$

Term	Fixed-Source	Eigenvalue
$\tilde{H}$	$\frac{1}{2}\sum_{n=1}^{N} w_n \bar{Y} Y_n^{-1} C$	$\frac{1}{2}\sum_{n=1}^{N} w_n \bar{Y} Y_n^{-1} \Gamma$
$\boldsymbol{G}$	$\frac{1}{2}\boldsymbol{e}_1^T \sum_{n=1}^{N} \mu_n w_n Y_n^{-1} C$	$\frac{1}{2}\boldsymbol{e}_1^T \sum_{n=1}^{N} \mu_n w_n Y_n^{-1} \Gamma$

Next, we define $\boldsymbol{F}_{\text{LDA,fs}}$ for fixed-source LDA cases:

$$\boldsymbol{F}_{\text{LDA,fs}} \equiv \frac{\left(1-e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{fs}} - \left(\dot{C}\boldsymbol{\tau}\right)^T \tilde{H}_{\text{fs}}}{\frac{2}{3\Lambda}\left(1-\cos\omega\Lambda\right) + \dot{\boldsymbol{c}}^T\boldsymbol{\tau}} \,, \quad \boldsymbol{F}_{\text{LDA,fs}} \in \mathbb{C}^{1\times p} \,, \qquad \text{(Eq. (5.34d) revisited)}$$

and $\boldsymbol{F}_{\text{LDA,eig}}$ for eigenvalue cases:

$$\boldsymbol{F}_{\text{LDA,eig}} \equiv \frac{\left(1-e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{eig}} + \left[\left(\lambda F - \dot{C}\right)\boldsymbol{\tau}\right]^T \tilde{H}_{\text{eig}}}{\frac{2}{3\Lambda}\left(1-\cos\omega\Lambda\right)} \,, \quad \boldsymbol{F}_{\text{LDA,eig}} \in \mathbb{C}^{1\times p} \,.$$

(Eq. (5.37g) revisited)

The forms of $\boldsymbol{F}$ for each case are summarized in Table 5.3. With these results, all of the spectral relation equations for each configuration have been provided.

Table 5.3: Forms of $\boldsymbol{F}$

Acceleration Method	Fixed-Source	Eigenvalue
CMFD	$\frac{\left(1-e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{fs}}-\left(\dot{C}\boldsymbol{\tau}\right)^T\tilde{H}_{\text{fs}}}{\frac{2\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T\boldsymbol{\phi}_0^{(\infty)}}\left(1-\cos\omega\Lambda\right)+\frac{\left(\dot{C}\boldsymbol{\tau}\right)^T\boldsymbol{\phi}_0^{(\infty)}}{\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}}}$	$\frac{\left(1-e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{eig}}+\left[\left(\lambda F-\dot{C}\right)\boldsymbol{\tau}\right]^T\tilde{H}_{\text{eig}}}{\frac{2\boldsymbol{v}^T\boldsymbol{\phi}_0^{(\infty)}}{3\boldsymbol{\tau}^T\boldsymbol{\phi}_0^{(\infty)}}\left(1-\cos\omega\Lambda\right)}$
LDA	$\frac{\left(1-e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{fs}}-\left(\dot{C}\boldsymbol{\tau}\right)^T\tilde{H}_{\text{fs}}}{\frac{2}{3\Lambda}\left(1-\cos\omega\Lambda\right)+\dot{\boldsymbol{c}}^T\boldsymbol{\tau}}$	$\frac{\left(1-e^{i\omega\Lambda}\right)\boldsymbol{G}_{\text{eig}}+\left[\left(\lambda F-\dot{C}\right)\boldsymbol{\tau}\right]^T\tilde{H}_{\text{eig}}}{\frac{2}{3\Lambda}\left(1-\cos\omega\Lambda\right)}$

We note that the form of $\boldsymbol{\beta}$ is interchangeable between the two acceleration methods. Linear prolongation can also be used for CMFD, and multiplicative prolongation can be used for LDA, without compromising the consistency of the low-order solution with the transport solution. This would change the form of $\tilde{S}$, and as a result, change the iterative properties of the method. These modifications are straightforward; thus, the original prolongation equations for each method are used for the results in this chapter.

5.5.2 Spatially Homogeneous Problems

For spatially homogeneous problems, the spectral radii of LDA and CMFD are identical. Rather than perform an additional Fourier analysis for spatially homogeneous problems to show this, we can simply use the results of the spatially heterogeneous analysis and assume a spatially-homogeneous geometry. To do this, we begin with the definition of $\boldsymbol{\beta}$ for CMFD. In an infinite, homogeneous problem, the converged scalar flux solution is uniform:

$$\boldsymbol{\phi}_0^{(\infty)} \equiv \Phi_0^{(\infty)} \mathbf{1} \,, \tag{5.45a}$$

where $\Phi_0^{(\infty)}$ is the magnitude of the scalar flux. Inserting this into the definition of $\boldsymbol{\beta}$ for CMFD yields

$$\boldsymbol{\beta} = \mathbf{1} \,, \tag{5.45b}$$

which is the same as for LDA.

Next, we turn our attention to the row vector $\boldsymbol{F}$, which depends on the operators $\tilde{H}$ and $\boldsymbol{G}$. In the homogeneous case for fixed-source problems, $\tilde{H}_{\text{fs}}$ reduces to the following form:

$$\tilde{H}_{\text{fs}} \equiv \frac{1}{2} \sum_{n=1}^{N} w_n \bar{Y} Y_n^{-1} c \,, \quad \tilde{H}_{\text{fs}} \in \mathbb{C}^{p \times p} \,, \tag{5.46a}$$

where c is the scattering ratio of the problem. Additionally, the definition of Y_n becomes

$$Y_n \equiv \bar{Y} + \begin{pmatrix} -\frac{\mu_n}{\tau} & \frac{\mu_n}{\tau} & 0 & \cdots & 0 \\ 0 & -\frac{\mu_n}{\tau} & \frac{\mu_n}{\tau} & 0 & \cdots \\ & & \vdots & \ddots & \\ \frac{\mu_n}{\tau} e^{i\omega\Lambda} & 0 & \cdots & 0 & -\frac{\mu_n}{\tau} \end{pmatrix} , \quad Y_n \in \mathbb{C}^{p \times p} \,, \tag{5.46b}$$

where τ is the optical thickness of each fine cell. Similarly, $\boldsymbol{G}_{\text{fs}}$ becomes

$$\boldsymbol{G}_{\text{fs}} \equiv \frac{1}{2} \boldsymbol{e}_1^T \sum_{n=1}^{N} \mu_n w_n Y_n^{-1} c \,, \quad \boldsymbol{G}_{\text{fs}} \in \mathbb{C}^{1 \times p} \,. \tag{5.46c}$$

In the case of eigenvalue problems, these operators contain the matrix Γ. For a spatially homogeneous case, the transport eigenvalue simply becomes

$$\lambda = \frac{\Sigma_a}{\nu\Sigma_f} = \frac{\dot{c}}{f} \; . \tag{5.46d}$$

Therefore, Γ becomes

$$\Gamma = cI + \frac{\dot{c}}{f} f I = (c + \dot{c}) I = I \; . \tag{5.46e}$$

Then, for eigenvalue problems $\tilde{H}_{\text{eig}}$ and $\boldsymbol{G}_{\text{eig}}$ become

$$\boldsymbol{G}_{\text{eig}} \equiv \frac{1}{2} \boldsymbol{e}_1^T \sum_{n=1}^{N} \mu_n w_n Y_n^{-1} \; , \quad \boldsymbol{G}_{\text{eig}} \in \mathbb{C}^{1 \times p} \; , \tag{5.46f}$$

and

$$\tilde{H}_{\text{eig}} \equiv \frac{1}{2} \sum_{n=1}^{N} w_n \bar{Y} Y_n^{-1} \; , \quad \tilde{H}_{\text{eig}} \in \mathbb{C}^{p \times p} \; , \tag{5.46g}$$

which are the same as the fixed-source forms where $c = 1$.

Lastly, we obtain the homogeneous forms of $\boldsymbol{F}$. For CMFD applied to fixed-source problems, the quantity $\left(\dot{C}\boldsymbol{\tau}\right)^T$ becomes $\dot{c}\tau\mathbf{1}^T$. Also, the vector of fine-cell volume fractions $\boldsymbol{v}$ becomes $\frac{1}{p}\mathbf{1}$. Then, $\boldsymbol{F}_{\text{CMFD,fs}}$ reduces to

$$\boldsymbol{F}_{\text{CMFD,fs}} \equiv \frac{\left(1 - e^{i\omega\Lambda}\right) \boldsymbol{G}_{\text{fs}} - \dot{c}\tau\mathbf{1}^T \tilde{H}_{\text{fs}}}{\frac{2}{3\Lambda}\left(1 - \cos\omega\Lambda\right) + \dot{c}\Lambda} \; , \quad \boldsymbol{F}_{\text{CMFD,fs}} \in \mathbb{C}^{1 \times p} \; . \tag{5.47}$$

If we make the same substitutions for $\boldsymbol{F}_{\text{LDA,fs}}$, we obtain

$$\boldsymbol{F}_{\text{LDA,fs}} \equiv \frac{\left(1 - e^{i\omega\Lambda}\right) \boldsymbol{G}_{\text{fs}} - \dot{c}\tau\mathbf{1}^T \tilde{H}_{\text{fs}}}{\frac{2}{3\Lambda}\left(1 - \cos\omega\Lambda\right) + \dot{c}\Lambda} \; , \quad \boldsymbol{F}_{\text{LDA,fs}} \in \mathbb{C}^{1 \times p} \; , \tag{5.48}$$

which is identical to Eq. (5.47). In the case of CMFD-accelerated eigenvalue problems, we first consider the following term, present in the numerator of $\boldsymbol{F}_{\text{CMFD,eig}}$, for spatially homogeneous cases:

$$\lambda \left(F\boldsymbol{\tau}\right)^T - \left(\dot{C}\boldsymbol{\tau}\right)^T = \frac{\dot{c}}{f} f\tau\mathbf{1}^T - \dot{c}\tau\mathbf{1}^T \; , \tag{5.49}$$

$$= \left(\dot{c}\tau - \dot{c}\tau\right) \mathbf{1}^T \; , \tag{5.50}$$

$$= \mathbf{0}^T \; . \tag{5.51}$$

Therefore, $\boldsymbol{F}_{\text{CMFD,eig}}$ becomes

$$\boldsymbol{F}_{\text{CMFD,eig}} \equiv \frac{\left(1 - e^{i\omega\Lambda}\right) \boldsymbol{G}_{\text{eig}}}{\frac{2}{3\Lambda}\left(1 - \cos\omega\Lambda\right)} \,, \quad \boldsymbol{F}_{\text{CMFD,eig}} \in \mathbb{C}^{1\times p} \,. \tag{5.52}$$

Similarly, $\boldsymbol{F}_{\text{LDA,eig}}$ simplifies to

$$\boldsymbol{F}_{\text{LDA,eig}} \equiv \frac{\left(1 - e^{i\omega\Lambda}\right) \boldsymbol{G}_{\text{eig}}}{\frac{2}{3\Lambda}\left(1 - \cos\omega\Lambda\right)} \,, \quad \boldsymbol{F}_{\text{LDA,eig}} \in \mathbb{C}^{1\times p} \,, \tag{5.53}$$

which is again identical to the CMFD case (and also to the fixed-source case with $c = 1$). With this, we have shown that the spectral relation equations for LDA and CMFD are the same for a given homogeneous problem type. Therefore, ρ is the same for each acceleration method for these homogeneous problems.

The spectral relation equations for LDA and CMFD differ for spatially-heterogeneous cases. This is due to different weighting choices for cross sections and different prolongation operators. With CMFD, the cross sections are weighted by the scalar flux, which is apparent by the presence of $\boldsymbol{\phi}_0^{(\infty)}$ in Eqs. (5.27o) and (5.31p), and prolongation is done multiplicatively. With LDA, the cross sections are volume-weighted, with linear correction terms that cancel out in the Fourier analysis (this is apparent by the lack of the converged scalar flux vector in Eqs. (5.34d) and (5.37g)), and prolongation is done additively. Other than these differences, the spectral relation equations for the two methods are similar for spatially-heterogeneous problems. Further, the equations in this section show that the spectral relation equations are the same for the two methods if the solution is spatially uniform (this is also true for problems with one fine cell per coarse cell). Thus, we expect that the two methods will have similar iterative properties for cases with low heterogeneity and possibly different iterative properties for cases with high heterogeneity.

We note that the presence of the scalar flux distribution in the heterogeneous Fourier analysis results for CMFD potentially allows for the analytic prediction of performance degradation due to these nonlinearities. The same nonlinearities are not present in the result for LDA. Theoretically, one could induce instability in CMFD for a contrived problem in which the nonlinear terms become numerically unstable. We explore this in Section 5.5.3 and Chapter 6.

There is an additional implication from the conclusion that LDA and CMFD possess the same spectral relation equations for homogeneous problems. Previous work has focused on scaling the diffusion coefficient of CMFD to optimize the spectral radius for optically thick problems [25, 26] with the resulting modified method called *optimally-diffusive CMFD* (odCMFD). Specifically, a stabilization parameter, θ_{od} (not to be confused with the error reduction rate, θ), is multiplied by the coarse cell geometric thickness, Δ, and added to the diffusion coefficient D for the low-order problem. The values of θ_{od} are chosen to minimize ρ as a function of the coarse cell optical thickness Λ by using the Fourier analysis result in an optimization problem. This study was performed for spatially homogeneous problems. Therefore, since LDA possesses the same spectral relation equations as CMFD for homogeneous problems, we expect that the same stability parameters could be used to improve the convergence properties of LDA.

5.5.3 Theoretical and Numerical Comparisons of LDA and CMFD

From the final forms of the spectral relation equations derived in this chapter, the dependence of the spectral radius on the spatial configuration of the lattice is apparent from the presence of space-dependent terms ($C, \dot{C}, \boldsymbol{\tau}$, etc.) in the overall error transition matrix, $\tilde{S}$. LDA and CMFD possess different dependencies on these terms, and as a result, the two methods behave differently for spatially-heterogeneous problems. As previously discussed, these differences are due to different weighting choices for the coarse-mesh cross section homogenization and the different prolongation operators. This section examines the different convergence properties of the two methods that arise from these different choices. We compare estimates of ρ obtained through "numerical experiments" using a 1D S_{16} research code to the analytic values of ρ from the Fourier analysis. The Gauss-Legendre S_{16} quadrature set and diamond-difference spatial closure scheme are used in all numerical calculations.

A number of studies are presented in this section, with each focusing on different iterative characteristics of LDA and CMFD. The goal of performing these studies is to explore the differences between the two methods for model problems. For reference, we provide Table 5.4, which briefly describes the focus of each study.

Table 5.4: Fourier Analysis Study Descriptions

Study #	Focus
1	Nonlinear instability of CMFD due to multiplicative prolongation
2	Linear instability of LDA & CMFD (homogeneous cases)
3	Effect of p on linear instability of LDA and CMFD (homogeneous cases)
4	Linear instability of LDA & CMFD (heterogeneous cases)
5	Effect of p on linear instability of LDA and CMFD (heterogeneous cases)

5.5.3.1 Nonlinear Instability of CMFD

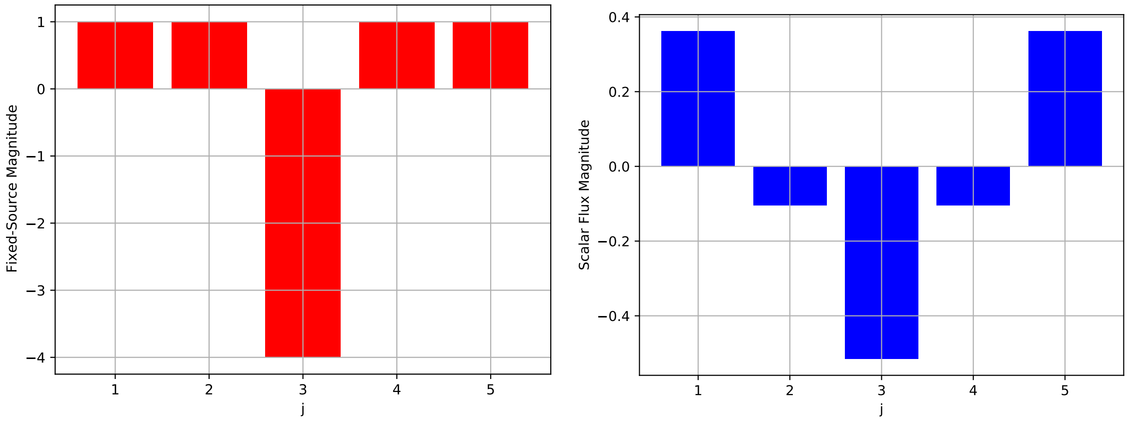
We can observe the nonlinearity of CMFD through the presence of terms in the error transition matrix that are nonlinear with respect to the converged scalar flux. This is present in (i) the prolongation operator $\boldsymbol{\beta}$ and (ii) terms in the denominator of $\boldsymbol{F}$. Regarding the former term, instability can occur if the converged coarse-cell scalar flux approaches zero. Regarding the latter term, instability can occur in multiple ways. The prolongation instability is present for both the fixed-source and eigenvalue form of the CMFD algorithm. Though other nonlinearities exist for CMFD, we focus on the prolongation nonlinearity for simplicity. (We note that alternate linear prolongation methods for CMFD have been investigated [27, 66]. However, the conventional nonlinear prolongation method is still widely used in many reactor physics codes.)

The presence of nonlinear terms in the spectral relation equations for CMFD suggests that the susceptibility of CMFD to degraded performance as $\Phi_0^{(\infty)}$ approaches zero can be analytically predicted through the use of the Fourier analysis. This can be demonstrated through the introduction of a negative source into the periodic lattice problem. For LDA, the prolongation operator is not dependent on the scalar flux and is, thus, not susceptible to this nonlinear instability. In this section, we present analytic predictions of the spectral radius made using the previously-derived spectral relation equations and compare them to results of numerical experiments. *The focus of this study is on the nonlinear instability of CMFD and the lack thereof in LDA.* This study is referred to as study #1.

A simple periodic problem in which $p = 5$ and the middle fine cell $(j = 3)$ possesses a negative source that is nearly equal in magnitude to the sum of the source in the other cells of the lattice, that possess equal and positive sources, was used to demonstrate the aforementioned vulnerability of CMFD to a near-zero solution. We modify the magnitude of the negative source in the middle cell while keeping the other material properties constant. For clarity, we have provided example plots of the source distribution and scalar flux for the 5 fine cells in a repeated coarse cell in Fig. 5.2. In Fig. 5.2a, we show the case in which the magnitude of the source in the center cell is equal to the sum of the sources in the other fine cells. The scalar flux solution corresponding to the source distribution in Fig. 5.2a and $c = 0.9$ is shown in Fig. 5.2b. As a result of the negative source, we see that the scalar flux solution is negative in the middle three fine cells. Because of these negativities, the corresponding coarse-cell scalar flux,

$$\Phi_{0,k} = \frac{1}{\Delta_k} \sum_{j \in k} \phi_{0,j} h_j \ , \tag{5.54}$$

is nearly zero. This is the condition needed to activate the nonlinear instability in CMFD.



(a) Fixed-source magnitude

(b) Scalar flux magnitude

Figure 5.2: (Study #1) Example fixed-source and converged scalar flux distributions

In Fig. 5.3, the x-axis indicates the value of the total source in the coarse cell (Q_{tot}), defined as

$$Q_{\text{tot}} = \sum_{j=1}^{p} q_j h_j \ . \tag{5.55}$$

In other words, as the x-axis values approach zero, so does the volume-sum of the sources over all fine cells in a given coarse cell. The characteristics of each cell in the lattice are $\Sigma_t = 1.0 \text{ cm}^{-1}$ and $h = 0.2 \text{ cm}$. Numerically-estimated spectral radii (markers) are shown alongside predicted spectral radii from the Fourier analysis (lines). This study was repeated for the following values of the scattering ratio c: 0.1, 0.3, 0.5, 0.7, and 0.9. As the x-axis values approach zero, the quantity $\Phi_0^{(\infty)}$ also approaches zero, and the performance of CMFD degrades. Once this term is sufficiently small, CMFD diverges while LDA remains stable. As the outer iterations progress, the CMFD prolongation equation resembles the following:

$$\phi_{0,j}^{(\infty)} = \phi_{0,j}^{(\infty)} \left(\frac{\Phi_{0,k}^{(\infty)}}{\Phi_{0,k}^{(\infty)}} \right) , \forall \ j \in k \ , \tag{5.56}$$

where $\Phi_{0,k}^{(\infty)}$ approaches zero as Q_{tot} approaches zero. Thus, the multiplicative prolongation factor (which is the term in parentheses in Eq. (5.56)) becomes numerically unstable due to division by zero (or near zero). These results demonstrate the vulnerability of CMFD to nonlinear instability, and the insensitivity of LDA to this effect. As c increases, the sensitivity of the nonlinear instability of CMFD increases. That is, the range of instability for CMFD is larger for higher values of c.

The results of study #1 are significant because they show that *LDA fundamentally lacks the nonlinear instability of CMFD* due to prolongation. That is, the performance of LDA

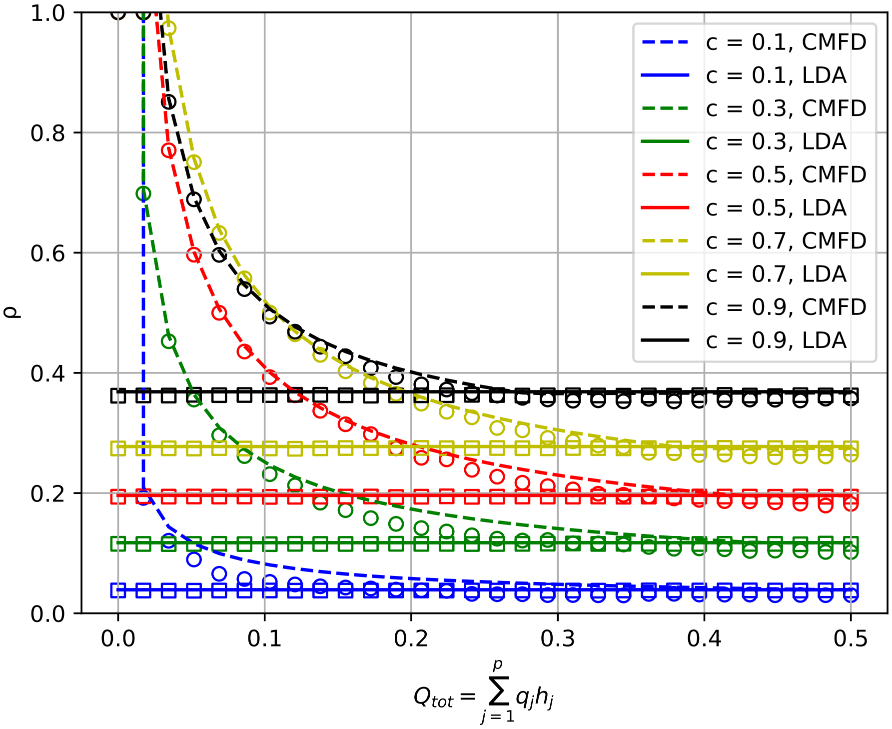


Figure 5.3: (Study #1) Numerical and Fourier analysis results. Analytic (solid lines for LDA and dashed lines for CMFD) and numerical (square markers for LDA and circle markers for CMFD) spectral radii of LDA and CMFD are shown for sample fixed-source problems with different scattering ratios in which the total coarse-cell source (Q_{tot}) approaches zero.

does not degrade in cases where the nonlinearity of CMFD causes numerical instability. In fact, LDA is shown to have superior performance in the region surrounding the range of divergence for CMFD. This supports our main goal for developing the LDA method, as discussed in Chapter 1. Additional studies that support this conclusion are provided in Chapters 6 and 7. In the next section, we compare the performance of the two acceleration methods for spatially-homogeneous problems.

5.5.3.2 Performance Comparisons for Spatially-Homogeneous Problems

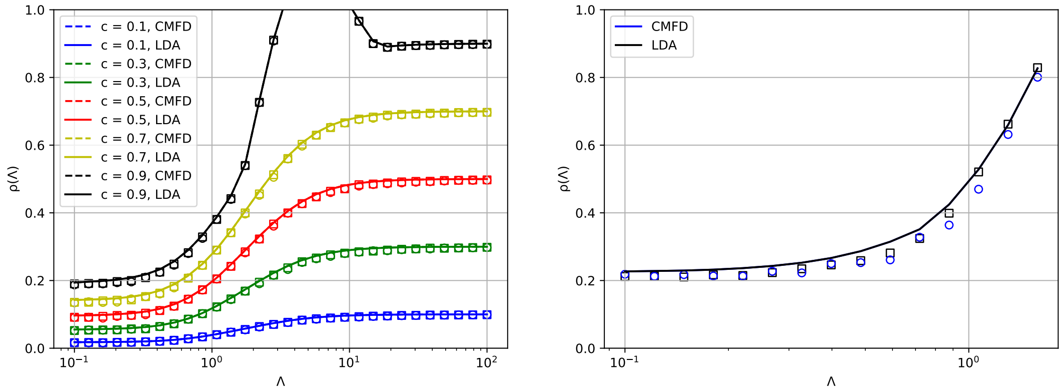
A known property of CMFD and other acceleration methods is the effect of coarse-cell optical thickness on the spectral radius [18, 26, 43]. Generally, the spectral radius of coarse-mesh, diffusion-based acceleration methods (such as LDA and CMFD) increases with the optical thickness of the coarse cell. We refer to this effect as linear instability. In this section, we investigate the linear instability of LDA and CMFD for spatially homogeneous problems with the following goals: (i) to confirm that the two methods have the same convergence properties for spatially-homogeneous problems (as stated in Section 5.5.2) and (ii) to confirm the correctness of the heterogeneous Fourier analysis results. Regarding the latter goal, the results in this section are obtained with the heterogeneous Fourier analysis equations (rather than the homogeneous equations shown in Section 5.5.2). However, the problems examined are homogeneous in space. First, we examine the effect of the coarse cell optical thickness on the spectral radius for fixed-source (study #2a) and eigenvalue problems (study #2b). Then, the effect of the number of fine cells per coarse cell p on the spectral radius is investigated for fixed-source (study #3a) and eigenvalue problems (study #3b) for a range of coarse-cell optical thicknesses. *Therefore, the studies in this section are focused on the linear instability of LDA and CMFD for spatially-homogeneous problems.*

For the fixed-source problems in study #2a, the convergence properties are dependent on the scattering ratio c. Therefore, we examine the following set of fine cell scattering ratios: 0.1, 0.3, 0.5, 0.7, and 0.9. For each value of c, a range of coarse cell optical thicknesses is tested and the analytic spectral radius is compared to the results of numerical experiments. Five fine cells are present in a coarse cell $(p = 5)$, and the width of each fine cell h is $1.0\ \mathrm{cm}$. The source magnitude is set to unity in each fine cell. To vary the optical thickness of the coarse cell (Λ), the total cross section in all fine cells is modified to achieve a given Λ.

The results of study #2a are shown in Fig. 5.4a. From these results, we observe that both the analytic and experimental spectral radii of LDA and CMFD are identical (within numerical noise) for all cases. The solid lines, indicating the LDA results, overlap the dashed lines, indicating the CMFD results. With the exception of $c = 0.9$, the spectral radius increases with Λ up to the value of c. In the case of $c = 0.9$, there is a range of Λ in which the methods become divergent. Once Λ becomes sufficiently large, the spectral radius decreases and then plateaus at the value of c. This behavior only occurs for values of c near unity in fixed-source problems. These results confirm the two aforementioned goals for fixed-source cases.

For study #2b, we examine the effect of Λ on the spectral radius for eigenvalue problems. We expect the spectral radius to increase with Λ, with no dependence on the scattering ratio (based on the discussion in Section 5.5.2). In these cases, the scattering ratio is fixed at 0.75 and f is set to unity. Otherwise, the cell properties are the same as those in study #2a. A smaller range of Λ is examined in this study because all cases become divergent above a critical value of Λ (around $\Lambda = 2$).

From the results in Fig. 5.4b, we observe that the analytic predictions of ρ for each acceleration method is the same for all values of Λ. Additionally, the experimental estimate of ρ is nearly the same for each method within numerical noise. Thus, we can confirm the applicability of the heterogeneous Fourier analysis results to homogeneous problems. Further, we observe that each of the two acceleration methods possess the same spectral radii for homogeneous problems.



(a) (Study #2a) Fixed-source results for different values of c

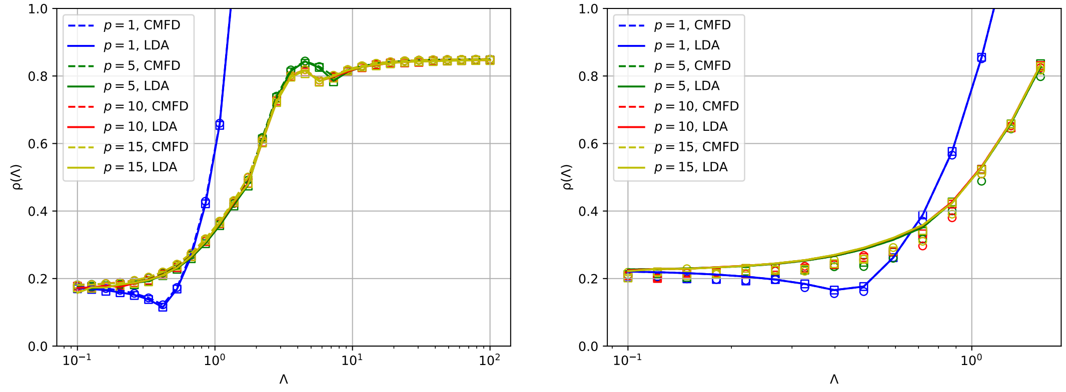
(b) (Study #2b) Eigenvalue results with c = 0.75

Figure 5.4: (Study #2) Numerical and Fourier analysis results. Analytic (solid lines for LDA and dashed lines for CMFD) and numerical (square markers for LDA and circle markers for CMFD) spectral radii are shown for sample problems as a function of coarse-cell optical thickness.

Next, we examine the results of study #3a, in which the number of fine cells per coarse cell (p) is changed for fixed-source problems. In this study, a range of coarse-cell optical thicknesses (Λ) is tested for each p value. As in the previous studies, a given Λ is obtained by scaling Σ_t in all of the fine cells. The scattering ratio is fixed at 0.85 for all cells (this value was chosen to be large so that differences can be observed for each p value), the width of each fine cell is 1.0 cm, and the fixed-source magnitude is unity. The following values of p were tested: 1, 5, 10, and 15.

Figure 5.5a shows the results of study #3a, in which p is varied for fixed-source problems. There are very slight differences in the spectral radius for different p values, except for $p = 1$ (which is the case where the fine and coarse mesh are identical). For values of p greater than 1, tangible differences are only observed in the region where Λ is $O(1)$. Within this region, larger values of p result in slightly smaller spectral radii. There are no observed differences between LDA and CMFD in any case.

Lastly, we examine the results of study #3b in which p is varied for eigenvalue problems. The same fine-cell properties are used here as those in study #3a, except that the fixed-source is absent and the value of f is equal to unity. From the results in Fig. 5.5b, we observe no tangible differences in the spectral radius when changing p for a given value of Λ except for $p = 1$. Further, LDA and CMFD have the same spectral radius for all cases within numerical noise.



(a) (Study #3a) Fixed-source results

(b) (Study #3b) Eigenvalue results

Figure 5.5: (Study #3) Numerical and Fourier analysis results. Analytic (solid lines for LDA and dashed lines for CMFD) and numerical (square markers for LDA and circle markers for CMFD) spectral radii are shown for sample problems as a function of Λ for different numbers of fine cells per coarse cell (p) and $c = 0.85$.

The results in this section show that the convergence properties of LDA and CMFD are *identical* for the homogeneous problems studied here. These results support one of the main goals for developing LDA discussed in Chapter 1, which is to develop an acceleration method that is more robust than CMFD but possesses similar convergence properties. By showing that the two methods behave identically for spatially-homogeneous problems, we are supporting this goal. The behavior of the two methods for spatially-heterogeneous problems is examined next.

5.5.3.3 Performance Comparisons for Spatially-Heterogeneous Problems

The results of the Fourier analysis of LDA and CMFD for heterogeneous problems show that the two methods possess different spectral radii. We are interested in how different degrees of heterogeneity affect the convergence properties of each method. *Therefore, this section is focused on the linear instability of each method for spatially-heterogeneous problems.* The following studies are performed in this section:

1. study #4a examines the effect of varying the optical thickness of one fine cell in the lattice for fixed-source problems,
2. study #4b examines the effect of varying the optical thickness and scattering ratio of one fine cell in the lattice for fixed-source problems,
3. study #4c is similar to study #4a, but performed for eigenvalue problems,

4. study #5a examines the effect of changing p for fixed-source cases, and
5. study #5b is the same as study #5a but for eigenvalue problems.

Study #4a contains fixed-source cases in which the sensitivity of each acceleration method to spatial heterogeneity was investigated for different values of coarse-cell optical thickness Λ. For each of these cases, the material properties of the center fine cell $(j = 3)$ were changed relative to the other fine cells $(j \neq 3)$ in a given coarse cell. Additionally, we assumed $p = 5$ and $h = 1.0$ cm for all fine cells. The fixed-source is uniform and equal to unity. For study #4, the value α, which is defined as the ratio of the optical thickness of the center cell relative to the other fine cells in the coarse cell, was varied. In equation form, α is defined as

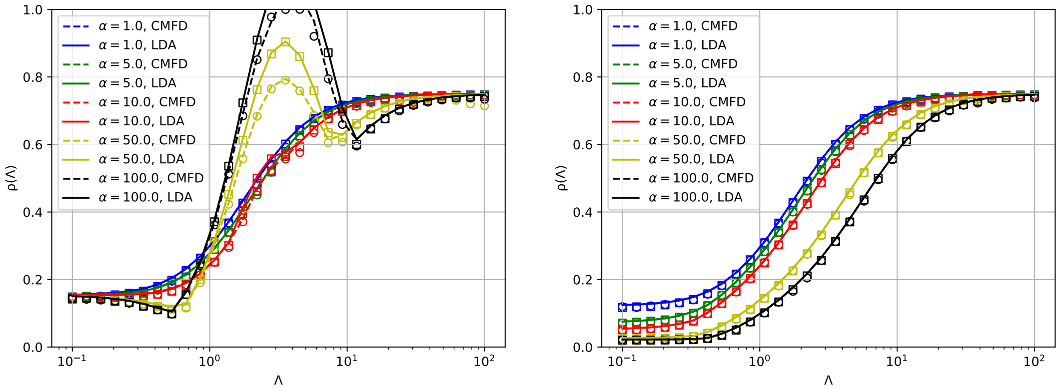
$$\alpha \equiv \frac{\Sigma_{t,3}}{\Sigma_{t,j\neq 3}} \ . \tag{5.57}$$

Thus, the total cross section of the center fine cell is changed relative to the other fine cells before normalizing the total cross sections of the fine cells such that a given Λ is achieved.

For the results of study #4a in Fig. 5.6a, the scattering ratio is uniform and equal to 0.75. This study demonstrates that there are differences in the two acceleration methods for heterogeneous problems, and that both methods exhibit the same qualitative behavior for sufficiently large fine-cell optical thicknesses. For Λ that is $O(1)$, LDA is observed to possess slightly worse convergence behavior compared to CMFD. However, both methods are seen to be sensitive to heterogeneity for Λ in this range. Previous work, which examined the impact of different heterogeneous parameters on convergence rate, has shown that CMFD performance is highly dependent on the fine-cell scalar flux distribution [67]. These results support this conclusion, since the heterogeneity of the converged scalar flux distribution corresponds to the heterogeneity of the material properties within a coarse cell.

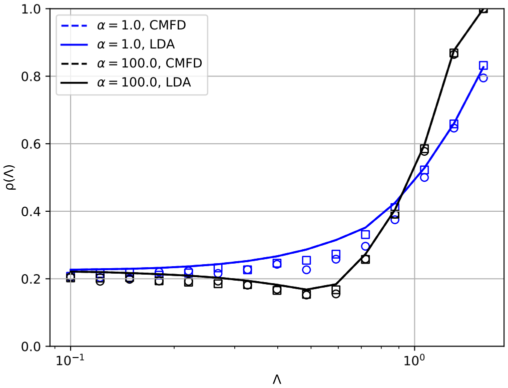
In Fig. 5.6b for study #4b, the same variation in α is studied, but the scattering ratio in the center cell is set to 0.1, while is it is set to 0.75 in the adjacent cells. This study is intended to investigate a pin-cell-like lattice, where the optically-thick and highly-absorbing center "fuel" cell is surrounded by optically-thin (less than 1 mfp) and highly-scattering "moderator" cells. In this study, we see almost no difference in performance between the two acceleration methods, and no predicted instability, indicating that for the practical reactor problem either method is not expected to become unstable and should perform similarly insofar as the local nonlinearities do not become problematic.

A similar study to #4a was performed for eigenvalue problems, in which the value of α is modified in study #4c. For these cases, the same mesh properties are used as in study #4a $(\Sigma_t = 1.0 \text{ cm}^{-1},\ h = 1.0 \text{ cm},\ c = 0.75,\ p = 5)$ and the value of f is uniform and equal to unity. We see very little difference between the homogeneous case and the highly



(a) (Study #4a) Fixed source results with constant scattering ratio

(b) (Study #4b) Fixed source results with non-uniform scattering ratio



(c) (Study #4c) Eigenvalue results

Figure 5.6: (Study #4) Numerical and Fourier analysis results. Analytic spectral radius of LDA (solid lines) and CMFD (dashed lines) compared to numerical estimates (square markers for LDA and circle markers for CMFD) as a function of coarse cell optical thickness when varying the relative optical thickness of the central fine cell (α) for sample problems.

heterogeneous case, in which $\alpha = 100$. Therefore, Fig. 5.6c only includes the homogeneous $(\alpha = 1)$ and highly heterogeneous case $(\alpha = 100)$ for readability. In these cases, the difference in convergence rate for LDA and CMFD is very small. In fact, the analytic prediction of the spectral radius is essentially identical for the two methods for a given value of α. Differences in the experimental results can be attributed to that fact that the experimental spectral radius is an estimate made using a finite problem. These results indicate that the differences in iterative performance for the two methods is less pronounced for eigenvalue problems than it is for fixed-source cases.

Next, we investigate the effect of the number of fine cells per coarse cell (p) on ρ for spatially-heterogeneous problems over a range of Λ in study #5a. To accomplish this, we consider a unit consisting of two fine cells. The right cell possesses an optical thickness

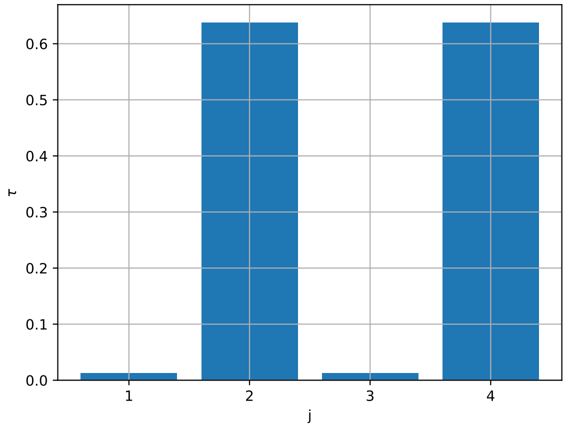


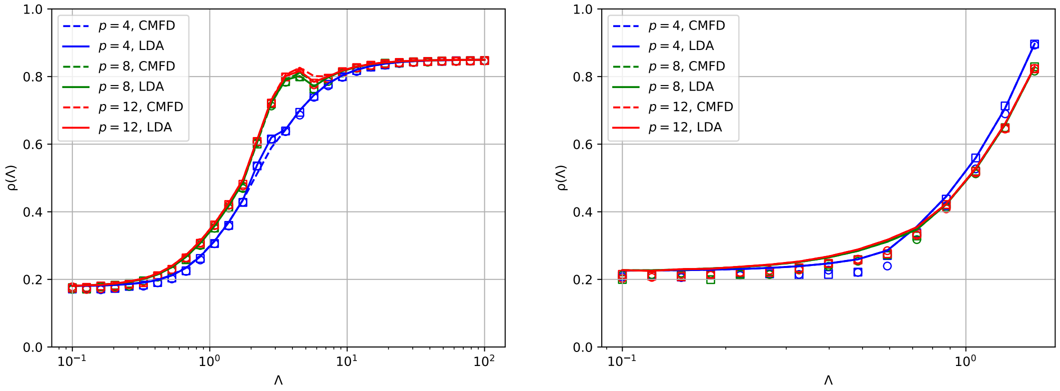
Figure 5.7: (Study #5) Spatial distribution of fine-cell optical thicknesses for $\Lambda = 1.0$

that is fifty times that of the left cell. By scaling the total cross section of the right cell relative to the left cell, we obtain these relative optical thickness values. This two-cell unit is repeated the required number of times to fill a single coarse cell with a given value of p. For example, if $p = 4$ then the two-cell unit is repeated twice to generate the coarse cell. Figure 5.7 shows the spatial distribution of fine-cell optical thicknesses for $p = 4$ and $\Lambda = 1.0$. Once the coarse cell is generated, the total cross section for all fine cells is normalized to achieve a given value of Λ. This process preserves the relative optical thickness of each fine cell. By conducting the study in this way, the spatial structure of the problem is the same for all cases except for the value of p. The following values of p are tested: 4, 8, and 12. Otherwise, the each fine cell possesses the following properties: $h = 1.0$ cm and $c = 0.85$. In study #5a, the fixed-source magnitude is equal to unity. For study #5b, the value of f is set to unity.

Figure 5.8a shows the results of study #5a, in which p is varied for fixed-source cases. We observe that the spectral radius is larger for larger values of p, which is the opposite trend compared to the results of study #3a for spatially-homogeneous problems. For these problems, LDA and CMFD possess similar spectral radii for all cases. Therefore, we conclude that changing the value of p does not result in tangible differences between the two acceleration methods for this study.

Finally, we focus on the results of study #5b, which is the same as study #5a except it is an eigenvalue problem. We observe very similar results as those in study #5a, with no tangible differences between LDA and CMFD for all cases. Additionally, we observe smaller differences between cases for different values of p compared to the fixed-source cases. Thus, eigenvalue problems appear to be less sensitive to this parameter.

The results in this section demonstrate that the differences in performance between



(a) (Study #5a) Fixed-source results

(b) (Study #5b) Eigenvalue results

Figure 5.8: (Study #5) Numerical and Fourier analysis results. Analytic (solid lines for LDA and dashed lines for CMFD) and numerical (square markers for LDA and circle markers for CMFD) spectral radii are shown for sample problems as a function of Λ for different numbers of fine cells per coarse cell (p) and $c = 0.85$.

LDA and CMFD are tangible only in cases with very high degrees of heterogeneity (as seen in study #4a). This observation supports one of the goals of this work discussed in Chapter 1, which is to develop an acceleration method with similar performance as CMFD for cases in which the nonlinear instability of CMFD is not an issue. We have also shown that the differences between the two methods is less significant for eigenvalue problems compared to fixed-source problems. Since the primary application of LDA is for eigenvalue problems, this is a favorable result.

5.6 Conclusions

In this chapter, we perform Fourier analyses for LDA and CMFD with the goal of comparing the convergence properties of each method. These analyses are carried out for simplified problems that are heterogeneous and periodic in space. The results of the Fourier analysis show that there are slight differences in the convergence properties of the two acceleration methods for spatially heterogeneous problems, which are attributed to (i) different choices of weighting functions for the coarse-mesh quantities and (ii) different prolongation operators. These differences are not expected to be significant for the practical problem. If the problem is uniform in space, which corresponds to volume-weighting functions for coarse-mesh quantities, the performance of the methods is identical.

By comparing the analytic prediction of the convergence rate of LDA and CMFD to estimates of this metric using a 1D research code, we have tested the accuracy of the Fourier analysis. Overall, the analytic prediction of the convergence rate is highly accurate for both homogeneous and heterogeneous problems. Further, we observe tangible differences between the performance of the two acceleration methods only for cases with very high degrees of heterogeneity. These differences are much less significant for eigenvalue problems than for fixed-source cases.

Through these studies, we have shown that LDA lacks the nonlinear instability of CMFD. Additionally, the linear instability properties of LDA are shown to be identical to CMFD for spatially-homogeneous problems and similar to CMFD for spatially-heterogeneous problems. These significant results support some of the goals for LDA established in Chapter 1. Chapters 6 and 7 present further numerical results for more complex problems to provide evidence in support of remaining goals relating to the development of LDA.

CHAPTER 6

1D Numerical Results

This chapter focuses on the use of LDA in a 1D research code. Section 6.1 provides a description of the code that was used to gather the results for this chapter. Then, we demonstrate the performance of LDA compared to CMFD when used to accelerate transport problems in Section 6.2. In Section 6.3, we show the nonlinear instability of CMFD, and the stability of LDA for some demonstrative cases. Section 6.4 contains a study of the f_{sh} parameter (discussed in Section 4.5) used to shift the LDA eigenvalue problem, resulting in faster convergence of the power iteration process. The potential advantages offered by LDA pertaining to linear solvers are discussed in Section 6.5. Finally, Section 6.6 provides a summary of the results in this chapter.

6.1 Code Description

A research code was developed to test the performance and iterative properties of LDA in relation to CMFD, and was used to obtain the results in this chapter. The code computes the numerical solution to the discrete, monoenergetic NTE in planar geometry using the discrete ordinates angular discretization. Angular integrals are approximated using a quadrature set, which was chosen to be the 1D Gauss-Legendre quadrature. The m^{th} moment of the angular flux is represented numerically as:

$$\int_{-1}^{1} P_m(\mu)\psi(x,\mu)\,\mathrm{d}\mu \approx \sum_{n=1}^{N} P_m(\mu_n) w_n \psi(x,\mu_n) \tag{6.1a}$$

$$= \sum_{n=1}^{N} P_m(\mu_n) w_n \psi_n(x)\ , \tag{6.1b}$$

where μ_n is the cosine of the polar angle corresponding to angular index n, w_n is the width of the "bin" corresponding to angular index n such that

$$\sum_{n=1}^{N} w_n = 1 - (-1) = 2 \, , \tag{6.1c}$$

P_m is the mth Legendre polynomial, and $\psi_n(x)$ represents the angular flux in angular bin n. Unless otherwise stated, all cases use $N = 16$. The scalar flux is the zeroth moment of the angular flux, corresponding to $m = 0$, and the neutron current corresponds to $m = 1$.

The fine spatial mesh is used for the transport problem, and is defined as shown in Fig. 3.1, where $x_{j\pm\frac{1}{2}}$ are the spatial locations of the edges of fine cell j. The following exact NBE (shown here for a fixed-source problem), presented in Chapter 3, is used to obtain the scalar flux solution on the fine mesh:

$$\frac{\mu_n}{h_j}\left(\psi_{n,j+\frac{1}{2}} - \psi_{n,j-\frac{1}{2}}\right) + \Sigma_{t,j}\psi_{n,j} = \frac{1}{2}\left(\Sigma_{s,j}\phi_j + q_j\right) \, . \tag*{(Eq. (3.4a) revisited)}$$

Here $\psi_{n,j\pm\frac{1}{2}}$ represents the angular flux at the edges of fine cell j, and h_j is the width of fine cell j. To relate the cell-edge and cell-average angular flux quantities, the diamond-difference relation is used:

$$\psi_{n,j} = \frac{1}{2}\left(\psi_{n,j+\frac{1}{2}} + \psi_{n,j-\frac{1}{2}}\right) \, , \tag*{(Eq. (3.4g) revisited)}$$

where $\psi_{n,j}$ represents the cell-average angular flux in angular bin n and fine cell j. Each coarse cell k is a contiguous union of fine cells, and is also depicted in Fig. 3.1. We use the notation $j \in k$ to refer to the fine cells that compose a given coarse cell k. Further details of the spatial and angular discretization of the NTE are provided in Chapter 3.

To accelerate the process of obtaining the numerical solution of the NTE, either LDA or CMFD can be used, and each acceleration method can be applied to both fixed-source and eigenvalue problems. The algorithms for these two acceleration methods are shown in Sections 3.3 and 4.4, respectively. For the convergence criterion, an L_∞ norm is used to test the relative difference between the fine-mesh scalar flux vector $\boldsymbol{\phi}_0$ across two iterations as shown below:

$$\max_j\left(\frac{\left|\boldsymbol{\phi}_0^{(l)} - \boldsymbol{\phi}_0^{(l-1)}\right|}{\boldsymbol{\phi}_0^{(l-1)}}\right) < \epsilon_1 \, , \tag{6.2}$$

where l is the outer iteration index and ϵ_1 is a user-specified tolerance. Once the condition in Eq. (6.2) is satisfied, code execution stops. A similar tolerance on the transport eigenvalue

λ is imposed for eigenvalue problems:

$$\left(\frac{\left| \lambda^{(l)} - \lambda^{(l-1)} \right|}{\lambda^{(l-1)}} \right) < \epsilon_2 \ , \tag{6.3}$$

where ϵ_2 is a user-specified tolerance for the eigenvalue. Regarding power iterations, a similar convergence criterion to Eq. (6.2) for the coarse-mesh scalar flux $\mathbf{\Phi}_0$ is imposed:

$$\max_j \left(\frac{\left| \mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+1\right)} - \mathbf{\Phi}_0^{\left(l+\frac{1}{2},m\right)} \right|}{\mathbf{\Phi}_0^{\left(l+\frac{1}{2},m\right)}} \right) < \epsilon_3 \ , \tag{6.4}$$

where ϵ_3 is the convergence criterion for the coarse-mesh scalar flux. Power iterations cease for outer iteration l once the condition in Eq. (6.4) is satisfied. To clarify, Eq. (6.2) is a condition on the fine-mesh scalar flux, and Eq. (6.4) is a condition on the coarse-mesh scalar flux. If LDA is used to accelerate an eigenvalue problem, the same condition in Eq. (6.4) is used for the initial forward and adjoint solutions of the diffusion problem. Unless otherwise stated, we use a tolerance of $\epsilon_1 = \epsilon_2 = \epsilon_3 = 10^{-8}$ for the results in this chapter.

The research code was written in the Julia programming language [68], and uses double-precision floating-point numbers. To solve the linear system of equations for the low-order problem, the LU method is used. This is a direct solver method, and the following procedure is followed for obtaining the low-order solution: (i) the $\mathcal{L}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle$ operator is decomposed before outer iterations begin, and (ii) to solve the low-order problem for each outer iteration, forward and backward substitution are used. Thus, the decomposition step only occurs once for a given problem setup. This method utilizes built-in capabilities of the Julia language, or readily available packages (the `LinearAlgebra` package was used).

6.2 LDA Performance

Here, we provide the results of sample fixed-source and eigenvalue cases, with the goal of comparing the performance of LDA and CMFD. We are interested in problems that were unable to be examined with the Fourier analysis in Chapter 5. In Chapter 5, each problem was conceptually (but not practically) infinite, with periodically-repeating material properties and converged solutions. This section provides results for test problems that are not infinite. Specifically, all problems possess vacuum boundary conditions on the left boundary and a reflective boundary condition on the right.

To measure the performance of each method, we provide either the total number of iterations required for convergence or the numerically-estimated spectral radius (ρ). As discussed in Chapter 5, the spectral radius describes the convergence rate of the method and can be estimated using the following equation:

$$\rho^{(l)} \approx \frac{\| \phi_0^{(l)} - \phi_0^{(l-1)} \|_2}{\| \phi_0^{(l-1)} - \phi_0^{(l-2)} \|_2} , \qquad \text{(Eq. (5.6) revisited)}$$

where ϕ_0 is the spatial scalar flux vector, l is the outer iteration index, and $\|(\cdot)\|_2$ is the L_2 norm. For unaccelerated PI, the number of iterations refers to the number of power iterations. For accelerated cases, this refers to the total number of outer iterations (OI), with one outer iteration consisting of a transport sweep and low-order solve.

We provide the results of two different studies for each problem type (fixed-source and eigenvalue). First, we examine cases with a fixed (i) number of total fine cells $(J = pK = 500)$, (ii) fine cells per coarse cell $(p = 5)$, and (iii) fine cell width $(h = 1.0$ cm$)$. For these cases, a range of coarse-cell optical thicknesses (Λ) is studied, with the total cross section adjusted to achieve a given value of Λ. This will be referred to as study #1. For study #2, we vary the total number of fine cells (pK) in the problem while keeping the number of fine cells per coarse cell and fine-cell width fixed ($p = 5$ and $h = 0.2$ cm, respectively). Additionally, the scattering ratio is fixed $(c = 0.99)$. In effect, changing pK changes the overall optical thickness of the system. For each value of pK, different values of Λ are tested. To achieve a given value of Λ, the total cross section is adjusted. Thus, study #1 examines the effect of increasing the optical thickness of an *individual* coarse cell, while study #2 examines the effect of increasing the *total* optical thickness of the system. A limit of 10^4 iterations was imposed for each problem, with "DNC" representing "Did Not Converge" within the imposed iteration limit.

6.2.1 Fixed-Source Problems

This section presents the results of study #1 and #2 for fixed-source cases. For all problems in this section, the internal fixed source is taken to be constant within the system, with a magnitude of 1.0 neutrons/(cm·s).

6.2.1.1 Study #1: Variable Optical Thickness by Increasing Total Cross Section

For this study, we examine the following values of the scattering ratio c: 0.4, 0.6, 0.8, 0.9, and 0.99. Figure 6.1 depicts the OI for LDA and CMFD, with the tabular data for $c = 0.4$ and $c = 0.99$ provided in Table 6.1 and Table 6.2, respectively. From these results, we observe that the performance of each method worsens (the spectral radius increases) with increasing Λ and c, which is expected from the Fourier analysis. For problems with small Λ, LDA and CMFD possess essentially the same performance (regardless of the scattering ratio). Unfortunately, the performance of LDA is worse than CMFD for problems with optically thicker coarse cells. This performance gap increases with increasing c, which can be clearly observed in Fig. 6.1. Specifically, for Λ = 10.0 and c = 0.4, LDA requires 30% more iterations to converge. For larger values of c, both acceleration methods become divergent for large Λ (as predicted by the Fourier analysis).

Table 6.1: Performance Comparison of LDA and CMFD for Study #1 with c = 0.4

Λ	#OI - PI	#OI - LDA	#OI - CMFD
0.1	23	8	8
0.25	24	9	8
0.5	25	10	9
0.75	26	12	10
1.0	26	13	11
2.5	27	19	15
5.0	28	23	18
7.5	28	25	19
10.0	29	26	20

Table 6.2: Performance Comparison of LDA and CMFD for Study #1 with c = 0.99

Λ	#OI - SI	#OI - LDA	#OI - CMFD
0.1	824	12	12
0.25	1200	13	12
0.5	1218	15	14
0.75	1187	18	17
1.0	1312	22	21
2.5	DNC	DNC	DNC
5.0	DNC	DNC	DNC
7.5	DNC	DNC	DNC
10.0	DNC	DNC	DNC

These performance differences were observed in the heterogeneous Fourier analysis in Chapter 5, and are not unexpected. In Chapter 5, the spectral radius of LDA was shown to be larger than CMFD for heterogeneous problems with $O(1)$ Λ. However, in these cases this difference persists for larger values of Λ, which was not observed in the Fourier analysis. This behavior may be a result of the (i) different weighting functions for LDA and

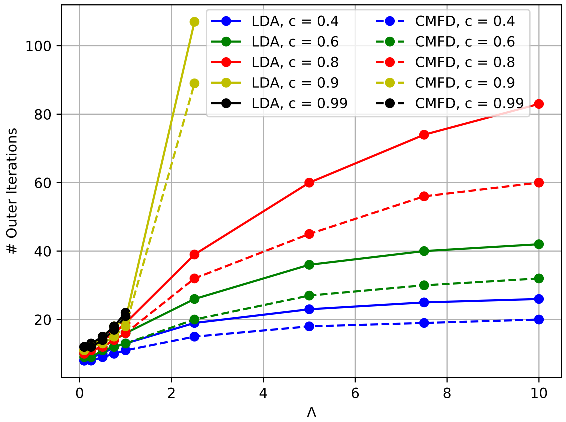


Figure 6.1: # outer iterations for LDA and CMFD for different values of c

CMFD (resulting in more lagged RHS terms for LDA), (ii) different prolongation operators for the two methods, or (iii) different boundary conditions for the problems studied here compared to the periodic boundary conditions assumed during the Fourier analysis. These differences may contribute to the performance differences seen here. Though the performance difference is significant for large Λ, the difference is much smaller for Λ around 1.0 mfp.

6.2.1.2 Study #2: Variable System Optical Thickness by Increasing Cell Number

In this study, we examine the effect of increasing the total optical thickness of the problem by increasing the number of fine cells pK. This is repeated for different values of Λ, which are achieved by adjusting Σ_t in a given fine cell. In all cases, $c = 0.99$. The numerically-estimated spectral radii of LDA and CMFD are reported for comparison to the prediction of the Fourier analysis. Because the leakage of the problem decreases with increasing pK, we expect the spectral radii to increase to the value predicted by the Fourier analysis (in which we assume no leakage).

Table 6.3 and Table 6.4 provide the tabular data for LDA and CMFD, respectively. From these results, we observe that LDA does not quite display the expected behavior. As pK increases, the spectral radius decreases slightly for small values of Λ and stays relatively constant for larger Λ. With CMFD we observe more expected behavior, with the numerically-estimated spectral radius increasing toward the Fourier analysis prediction as the optical thickness of the problem increases. In many cases with large pK, the numerically-estimated spectral radius is smaller for LDA compared to CMFD.

Table 6.3: LDA Numerical Spectral Radii for Fixed-Source Problems w/ $c = 0.99$

Λ pK	0.3	0.6	0.9	1.2	1.5
10	0.20	0.30	0.39	0.46	0.59
25	0.20	0.29	0.37	0.45	0.56
50	0.17	0.25	0.34	0.43	0.61
100	0.15	0.22	0.33	0.43	0.60
250	0.14	0.23	0.34	0.44	0.61
500	0.15	0.23	0.34	0.44	0.61
∞ (Fourier Analysis)	0.24	0.31	0.40	0.50	0.66

Table 6.4: CMFD Numerical Spectral Radii for Fixed-Source Problems w/ $c = 0.99$

Λ pK	0.3	0.6	0.9	1.2	1.5
10	0.18	0.25	0.36	0.37	0.55
25	0.21	0.30	0.33	0.43	0.59
50	0.22	0.29	0.35	0.47	0.60
100	0.22	0.29	0.37	0.49	0.64
250	0.22	0.28	0.37	0.49	0.64
500	0.23	0.28	0.37	0.48	0.65
∞ (Fourier Analysis)	0.24	0.31	0.40	0.50	0.66

6.2.2 Eigenvalue Problems

This section presents the results of study #1 and #2 for eigenvalue cases. For all problems in this section, $\nu\Sigma_f$ is set equal to Σ_t. The material properties are uniform throughout the system.

6.2.2.1 Study #1: Variable Optical Thickness by Increasing Total Cross Section

For this study, the scattering ratio is fixed at $c = 0.95$. We examine the effect of increasing the optical thickness of each coarse cell on OI. From the Fourier analysis, we expect the performance of LDA, CMFD, and unaccelerated PI to degrade with increasing Λ. We omit values of $\Lambda > 1.6$ because both acceleration methods become divergent in this range. Additionally, one value of c is tested because the impact of c on the convergence rate is negligible.

The results of this study are shown in Table 6.5 and plotted in Fig. 6.2. For low values of Λ, LDA and CMFD are observed to possesses similar performance. However, the linear instability of LDA is seen to be more sensitive to Λ compared to CMFD for optically-thicker problems. For Λ = 1.6, #OI for LDA is about 62% larger than for CMFD. This behavior was not observed in the Fourier analysis in Chapter 5, which assumed periodic boundary conditions.

Though the linear instability of LDA is seen to worsen with Λ more quickly than CMFD, this issue is significant only when Λ approaches the point of divergence for the two methods. For $\Lambda = 1.0$, LDA is seen to require only 13% more OI compared to CMFD. These performance differences are further explored for practical problems in Chapter 7.

Table 6.5: Performance of LDA & CMFD for Eigenvalue Problems w/ $c = 0.95$

Λ	#OI - PI	#OI - LDA	#OI - CMFD
0.2	942	13	12
0.4	3240	17	14
0.6	6712	23	16
0.8	DNC	23	19
1.0	DNC	26	23
1.2	DNC	36	31
1.4	DNC	55	44
1.6	DNC	123	76

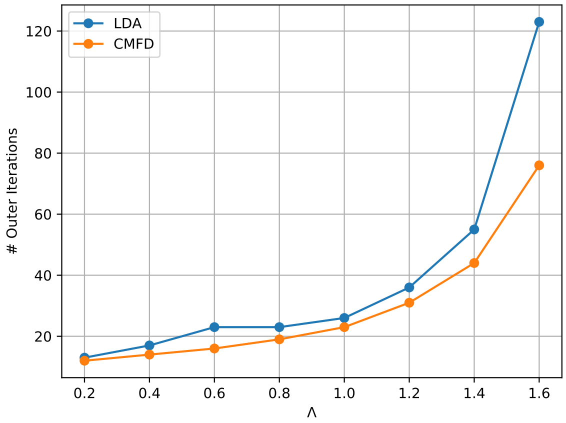


Figure 6.2: Number of outer iterations required for convergence for LDA and CMFD

6.2.2.2 Study #2: Variable System Optical Thickness by Increasing Cell Number

In this section, the results of study #2 are presented for eigenvalue problems, in which the total optical thickness of the problem is varied by changing pK. This is repeated for different values of Λ, and the scattering ratio c is fixed at 0.99. The tabular data is provided in Table 6.6 and Table 6.7 for LDA and CMFD, respectively. We again observe some anomalous behavior for LDA compared to the Fourier analysis. Though the spectral radius is generally seen to increase with pK for LDA, it slightly exceeds the Fourier analysis prediction for the most optically thick problems with optically-thick coarse cells. This behavior is not seen with CMFD, in which the spectral radius increases with K but does not exceed the Fourier analysis prediction. The use of different boundary conditions for these problems (compared to those used in the Fourier analysis) may contribute to these results.

Table 6.6: LDA Numerical Spectral Radii for Eigenvalue Problems w/ c = 0.99

Λ pK	0.3	0.6	0.9	1.2	1.5
10	0.25	0.27	0.40	0.46	0.64
25	0.23	0.29	0.38	0.55	0.73
50	0.23	0.30	0.41	0.59	0.76
100	0.23	0.31	0.40	0.59	0.77
250	0.23	0.29	0.44	0.60	0.77
500	0.20	0.30	0.47	0.64	0.83
∞ (Fourier Analysis)	0.25	0.32	0.44	0.60	0.78

Table 6.7: CMFD Numerical Spectral Radii for Eigenvalue Problems w/ c = 0.99

Λ pK	0.3	0.6	0.9	1.2	1.5
10	0.24	0.28	0.38	0.46	0.67
25	0.23	0.29	0.38	0.54	0.73
50	0.22	0.29	0.42	0.58	0.76
100	0.23	0.28	0.42	0.59	0.77
250	0.23	0.27	0.43	0.60	0.77
500	0.22	0.29	0.46	0.57	0.75
∞ (Fourier Analysis)	0.25	0.32	0.44	0.60	0.78

6.2.3 Discussion of LDA Performance

In this section, we observe that LDA exhibits worse convergence properties than CMFD for problems in which the mesh is optically-thick. In Chapters 4 and 5, the possible contributors to the performance differences between LDA and CMFD are discussed. Namely, these are:

1. the different weighting function used for the coarse-mesh quantities (flux- and volume-weighting for CMFD and volume-weighting for LDA), and
2. the different prolongation operators for each method (multiplicative for CMFD and additive for LDA).

In Chapter 5, we observe that the spectral radius of LDA is larger than CMFD for highly heterogeneous problems with relatively optically-thick meshes. The same behavior is seen here, with a highly heterogeneous solution being a result of an optically-thick mesh and the presence of a vacuum boundary condition. For problems with optically-thick mesh cells, an extreme material discontinuity is present at the vacuum edge of the system. In this case, the scalar flux changes sharply at this vacuum edge. For optically-thinner meshes, neutrons travel longer distances without a collision on average, resulting in a smoother changes in the scalar flux as a function of space compared to optically-thicker problems.

For optically-thick problems, the volume-weighting for coarse mesh quantities is likely the largest contributor to the observed poorer performance of LDA compared to CMFD. Considering that the choice of numerical diffusion coefficient can affect the convergence properties of CMFD, the use of a diffusion coefficient that is not informed by the transport solution iterate may significantly impact the performance of LDA. As discussed in Chapter 4, the use of a "better" weighting function may improve the convergence properties of LDA. We suggest the investigation of improved weighting functions for LDA as future work. Additionally, the use of different boundary conditions (rather than the diffusion boundary conditions used here) that incorporate information from the transport sweep may improve performance.

6.3 Nonlinear Numerical Stability of LDA and CMFD

This section presents three case studies that are paired with the proposed nonlinear instability failure modes of CMFD that were described in Section 3.5. Each of the three cases is a contrived problem designed to demonstrate the susceptibility of a numerical implementation of CMFD to failure as a result of the nonlinear instability inherent to the method. For each case, a range of configurations is investigated. In this range, CMFD transitions from stable to unstable. This manifests as an increase in the number of outer iterations required to converge the CMFD-accelerated problem to the point of divergence, where the number of outer iterations exceeds the imposed limit. The number of outer iterations required for convergence with CMFD and LDA is plotted, as well as the quantity that results in instability for CMFD.

For these problems, we consider the degradation in CMFD performance to be due to nonlinear terms becoming unphysically large during the iterative process. This is ultimately due to division by small quantities that depend on the scalar flux. As the denominator of each of the three nonlinear quantities decreases, magnitude of the term increases. Once the term becomes sufficiently large, the iterative process destabilizes. This phenomenon is observed in each of the three studies here. Further, there is a range in which the performance of CMFD suffers but the method still converges. Considerable noise in the number of iterations required for convergence is observed in this range, which we attribute to floating-point error. We note that the range over which instability occurs is small in these studies, and is much more "zoomed-in" than the plot shown in Fig. 5.3, so the noise associated with floating-point error is visible in the plots for these studies.

In each of the three studies, a fixed-source problem with 10 fine spatial mesh regions of width 0.1 cm is considered. Therefore, the width of the slab is 1.0 cm. The boundary conditions are vacuum on each side of the system. Unless otherwise stated, the source magnitude and total cross section are equal to unity. The coarse cells contain two fine cells each ($p = 2$), resulting in a total of 5 coarse cells of width 0.2 cm. A cap of 1000 outer iterations was imposed, with iterations halted once this limit is reached.

To probe the nonlinearities of CMFD, we set the fixed source in a single fine cell to a negative value. This emulates a negative source (possibly as a result of high transverse leakage from a spatial region in the 2D/1D method), which may cause the scalar flux solution to become non-positive in a portion of space. Further, if the negative source causes a coarse-mesh cross section to become negative, then the scenario emulates a case where the use of TCP_0 results in negative cross sections and, ultimately, negative flux values. Once this happens, the possibility of nonlinear instability exists for the CMFD method. In specific scenarios, explored in this section, each of the nonlinear terms in the CMFD formulation can become unphysically large and numerically unstable, leading to reduced performance and, eventually, divergence (as predicted by the Fourier analysis in Chapter 5). These issues are not observed for LDA.

6.3.1 Instability due to the Diffusion Coefficient

In this case, the value of the source in the 7th cell from the left side of the geometry is varied over a range of negative values. Around a critical value of the negative source in the corresponding coarse cell (4th from the left side), the volume- and flux-weighted total cross section for the cell becomes opposite in sign and equal in magnitude to the total cross section of the coarse cell to the right (5th from the left side). In this case, the value of

$\tilde{D}$ at the interface of these cells becomes unstable, leading to divergence of CMFD. The equation for $\tilde{D}$ is reproduced below for reference:

$$\tilde{D}_{k+\frac{1}{2}}^{(l+\frac{1}{2})} \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k+1}^{(l+\frac{1}{2})}\Delta_{k+1} + \bar{\Sigma}_{t,k}^{(l+\frac{1}{2})}\Delta_k\right)} \,, \tag{Eq. (3.8f) revisited}$$

where the homogenized cross section for a given reaction type u is defined as

$$\bar{\Sigma}_{u,k}^{(l+\frac{1}{2})} \equiv \frac{\sum\limits_{j\in k} \Sigma_{u,j}\phi_{0,j}^{(l+\frac{1}{2})} h_j}{\sum\limits_{j\in k} \phi_{0,j}^{(l+\frac{1}{2})} h_j} \,, \forall\ u \in \{a, s, t\}\ . \tag{Eq. (3.8h) revisited}$$

In contrast, the corresponding term in LDA is defined as

$$\tilde{D}_{k\pm\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \equiv \frac{2}{3\left(\bar{\Sigma}_{t,k}\langle\!\langle \boldsymbol{f} \rangle\!\rangle \Delta_k + \bar{\Sigma}_{t,k\pm1}\langle\!\langle \boldsymbol{f} \rangle\!\rangle \Delta_{k\pm1}\right)} \,, \tag{Eq. (4.12b) revisited}$$

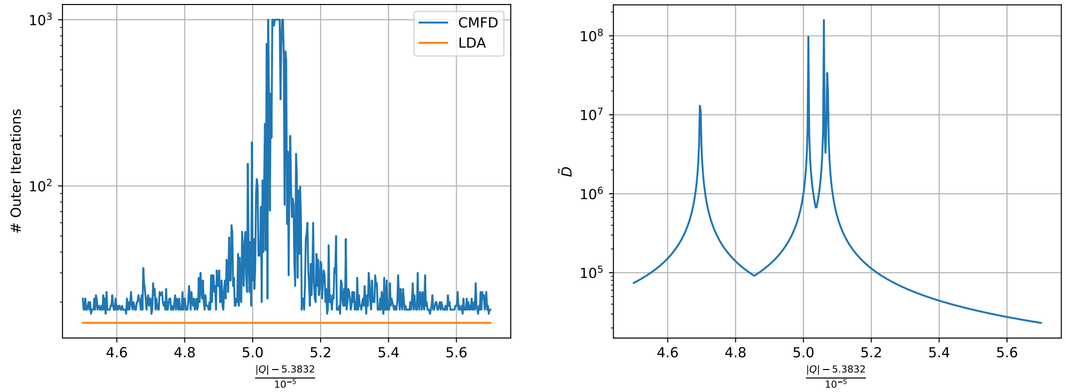
where the homogenized cross section is defined as

$$\bar{\Sigma}_{t,k}\langle\!\langle \boldsymbol{f} \rangle\!\rangle \equiv \frac{\sum\limits_{j\in k} \Sigma_{t,j} f_j h_j}{\sum\limits_{j\in k} f_j h_j}, \tag{Eq. (4.10c) revisited}$$

and the weighting function $\boldsymbol{f}$ is chosen to be $\mathbf{1}$. Because this weighting function is not the fine-mesh scalar flux, LDA is not susceptible to this failure mode.

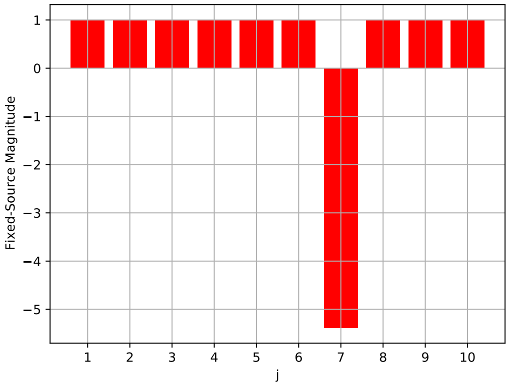
Figure 6.3a shows the number of outer iterations required for convergence as a function of the negative source magnitude in the 7th cell from the left for both CMFD and LDA. Because LDA uses volume-weighted cross sections, with linear correction terms in the source to account for error due to the lack of applying the flux weight, it remains stable in the region of instability for CMFD. Figure 6.3b shows the largest value of $\tilde{D}$ over all iterations at the cell interface when using CMFD, which becomes unphysically large ($O(10^8)$) in the region where CMFD diverges. For clarity, the approximate magnitude of the fixed source in each fine cell is provided in Fig. 6.3c for the cases in which CMFD diverges.

We observe that the peak magnitude of $\tilde{D}$ does not necessarily correspond directly with an increase in the number of outer iterations required for convergence. Rather, there is some degree of unpredictability pertaining to this failure mode as a result of floating-point error and the use of the L_∞ norm for the convergence criterion. What is clear is that the performance of the CMFD method is unstable and degrades significantly in the range



(a) Number of outer iterations to achieve convergence vs. negative source magnitude in the 7th fine mesh cell

(b) Peak value of $\tilde{D}$ at the interface of the 4th and 5th coarse cells over all iterations vs. negative source magnitude



(c) Example fixed-source distribution in fine cells for a case where CMFD diverges

Figure 6.3: Figures for the $\tilde{D}$ study

where $\tilde{D}$ becomes large. There is also a "gap" between the number of iterations required for convergence for LDA and CMFD in Fig. 6.3a for the full range of source values tested. It is possible for $\tilde{D}$ to become unphysically large and CMFD to still converge. However, once the method is strained beyond a certain point, the iterative process cannot recover and divergence occurs. Limiting the magnitude of $\tilde{D}$ is a potential remedy to the issues seen here. However, this could negatively impact the convergence rate of CMFD (since changing the magnitude of $\tilde{D}$ has been done to improve the convergence rate [25, 26]).

6.3.2 Instability due to the Transport Correction Term

For this study, the negative source magnitude was adjusted to result in CMFD failure due to instability in the $\hat{D}$ quantity, which is reproduced below:

$$\hat{D}_{k+\frac{1}{2}}^{(l+\frac{1}{2})} \equiv \frac{\Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})} + \tilde{D}_{k+\frac{1}{2}}^{(l+\frac{1}{2})}\left(\Phi_{0,k+1}^{(l+\frac{1}{2})} - \Phi_{0,k}^{(l+\frac{1}{2})}\right)}{\Phi_{0,k+1}^{(l+\frac{1}{2})} + \Phi_{0,k}^{(l+\frac{1}{2})}} \,. \qquad \text{(Eq. (3.8g) revisited)}$$

This term can become numerically unstable if the two denominator terms become equal in magnitude and opposite in sign. In LDA, transport consistency is achieved by lagging the numerator of $\hat{D}$ in the iteration process.

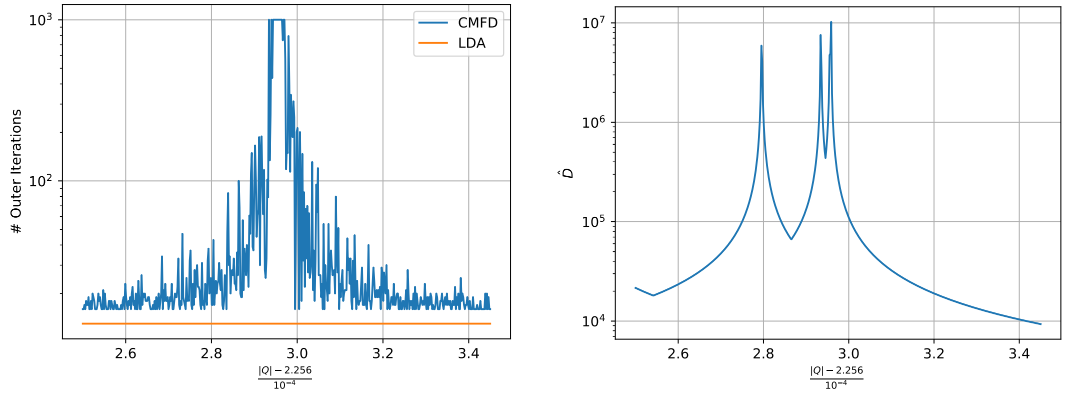
A critical range of negative source magnitudes in the 7th fine cell from the left side allows for the coarse cell scalar flux for the 4th and 5th coarse cells to become opposite in sign and nearly equal in magnitude. CMFD is seen to diverge once the two quantities are sufficiently close, as shown in Fig. 6.4a. LDA does not contain $\hat{D}$ in its formulation, and therefore remains stable. Figure 6.4b shows the largest value of $\hat{D}$ over all iterations at the interface of the 4th and 5th coarse cells, which becomes large $(O(10^7))$ in the region where CMFD diverges. For clarity, Fig. 6.4c shows the approximate source magnitudes for the cases where CMFD diverges.

In these results, we again observe certain cases where the unstable quantity becomes large but CMFD still converges. We attribute this to the unpredictability of the nonlinear instability. It is possible that in these cases, the magnitude of $\hat{D}$ was large enough to interfere with the convergence process but not large enough to prevent convergence. A large value of $\hat{D}$ does not necessarily directly result in divergence of the iterative method. Yet, we see that once the method is sufficiently strained, divergence occurs. If a limit was placed on the magnitude $\hat{D}$, the consistency of the low-order and transport solutions could be compromised since the purpose of $\hat{D}$ is to ensure this consistency. Therefore, such an approach could result in the code producing an inaccurate solution.

6.3.3 Instability due to Prolongation of the Scalar Flux

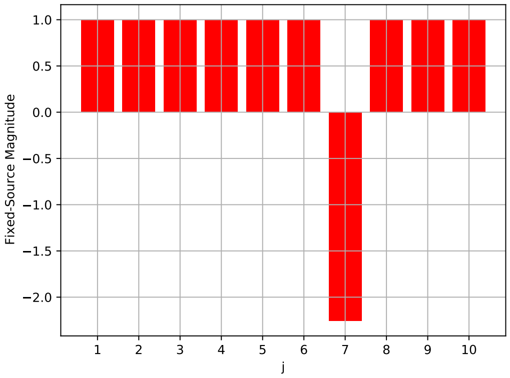
This section focuses on the multiplicative prolongation scalar that is used to update the fine-mesh scalar flux once the low-order problem has been solved. The multiplicative prolongation equations are shown below:

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{(l+\frac{1}{2})} c_k^{(l+1)} \,, \forall\, j \in k \,, \tag{6.5a}$$



(a) Number of outer iterations to achieve convergence vs. negative source magnitude in the 7th fine mesh cell

(b) Peak value of $\hat{D}$ at the interface of the 4th and 5th coarse cells over all iterations vs. negative source magnitude



(c) Example fixed-source distribution in fine cells for a case where CMFD diverges

Figure 6.4: Figures for the $\hat{D}$ study

where the scalar $c_k^{(l+1)}$ is defined as the ratio of the newly-computed coarse mesh scalar flux to that from the transport sweep:

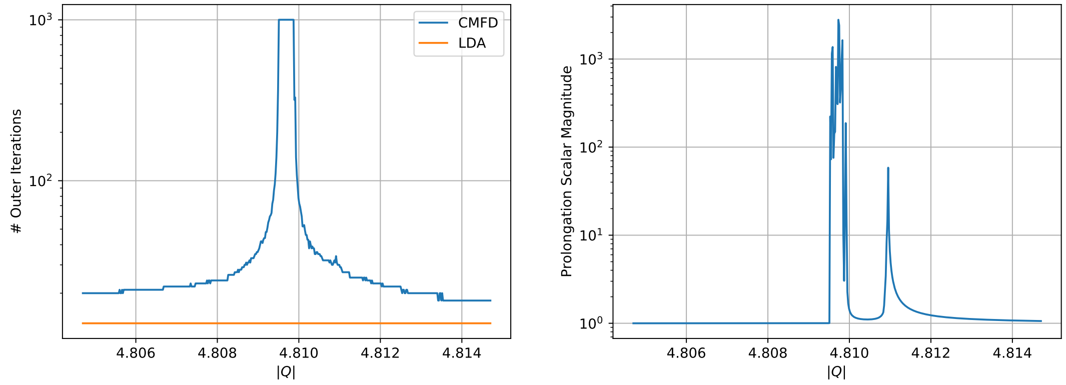
$$c_k^{(l+1)} \equiv \left(\frac{\Phi_{0,k}^{(l+1)}}{\Phi_{0,k}^{(l+\frac{1}{2})}} \right) . \tag{6.5b}$$

If the scalar flux in a coarse-mesh cell becomes small, this quantity can become unstable as shown in Fig. 6.5a. In LDA, the prolongation update is linear:

$$\phi_{0,j}^{(l+1)} = \phi_{0,j}^{(l+\frac{1}{2})} + \left(\Phi_{0,k}^{(l+1)} - \Phi_{0,k}^{(l+\frac{1}{2})} \right) , \forall\ j \in k . \qquad \text{(Eq. (4.41) revisited)}$$

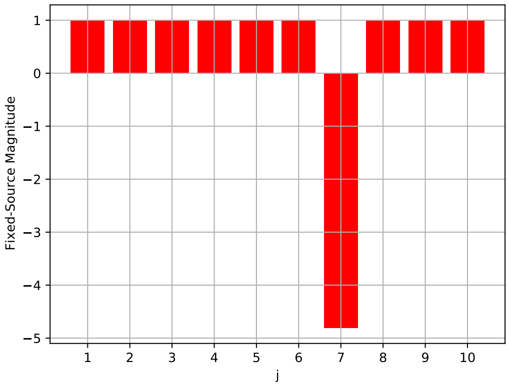
We note that this same linear prolongation update could be used in the CMFD method, but the multiplicative update is traditionally used in many reactor physics codes. Additionally, the use of Eq. (4.41) in CMFD may increase the probability of negative scalar flux iterates that could perturb the other nonlinearities of the method.

The number of outer iterations required for convergence for each method is shown in Fig. 6.5a. Figure 6.5b shows the largest value of $c_k^{(l+1)}$ over all outer iterations, which grows in magnitude as a critical value of the negative source in the 7th fine cell is approached. Figure 6.5c shows the approximate source magnitude for cases in which CMFD diverges. We observe that the performance of CMFD is significantly worse than LDA in the range near the critical point of instability. Though the magnitude of the prolongation scalar does not directly correspond to the number of outer iterations required for convergence, CMFD is shown to break down once sufficient strain is imposed on the method. Because prolongation in LDA is done in an additive manner, this instability is irrelevant.



(a) Number of outer iterations to achieve convergence vs. negative source magnitude in the 7th fine mesh cell

(b) Peak value of the multiplicative prolongation scalar over all iterations vs. negative source magnitude



(c) Example fixed-source distribution in fine cells for a case where CMFD diverges

Figure 6.5: Figures for the multiplicative prolongation study

We note that Fig. 6.5a possesses much less noise in the number of outer iterations compared to Fig. 6.3a and Fig. 6.4a. In part, this is due to the larger range of divergence for the prolongation study. Additionally, divergence in the $\tilde{D}$ and $\hat{D}$ studies depends on two denominator terms approaching the same magnitude. In contrast, one denominator term is the focus for the prolongation study. The smaller number of degrees of freedom lends the prolongation study to less noise compared to the other studies.

We consider the possibility of placing a limit on the magnitude of the prolongation factor. Because this factor updates the transport solution with information from the low-order solve, such a limit could affect the convergence rate of CMFD. Further, an incorrect solution could be produced if the low-order problem converges to an incorrect solution as a result of the limit.

6.3.4 Summary of Nonlinear Instability Studies

In this section, we have presented performance comparisons between LDA and CMFD for problems in which CMFD becomes unstable as a result of nonlinear instability. Specifically, we have examined scenarios where the following nonlinear terms in CMFD become numerically unstable: (i) the flux-weighted numerical diffusion coefficient $\tilde{D}$, (ii) the transport consistency term $\hat{D}$, and (iii) the multiplicative prolongation factor. By introducing a negative source, which emulates realistic scenarios where negative flux solutions are possible, these terms are forced to become unphysically large and unstable over a range of source values.

The results of each study show that CMFD diverges in a critical nonlinear instability range corresponding to a range of negative source values. This divergence is due to nonlinear terms in the CMFD formulation, which are normally assumed to be "small", becoming unphysically large. Further, the performance of CMFD degrades as this critical nonlinear instability range is approached. For these same cases, LDA is shown to be insensitive to these nonlinear instabilities, and retains the expected convergence behavior. *These results support the main goal for development of the LDA method as established in Chapter 1, which is to create a method that is not susceptible to the nonlinear instabilities inherent to CMFD.*

6.4 Wielandt-Shifted LDA

Here, we examine the effect of changing the "shift parameter" f_{sh} on the performance of PI for LDA applied to eigenvalue problems. We expect the condition number of the LDA operator to increase with f_{sh}, since the operator becomes more singular as f_{sh} approaches unity. Additionally, the condition number of the Laplace operator (contained in the LDA matrix) is expected to increase more quickly for more spatially-refined problems [50] (or problems with more spatial cells). Thus, we examine the effect of changing f_{sh} for problems with different coarse cell widths (Δ). As the problem becomes more refined, corresponding to smaller Δ, it should become more sensitive to numerical roundoff error. The goal of the studies in this section is to quantitatively estimate the range of reasonable values for f_{sh} and to observe the effect of spatial refinement on the upper bound of f_{sh}.

The f_{sh} parameter specifies the fraction of the diffusion fission source that is subtracted from the LHS operator for the LDA method applied to an eigenvalue problem. For reference, the shifted LDA equations discussed in Section 4.5 are reproduced here:

$$\mathcal{L}_{d,eig,sh} \langle\!\langle \mathbf{1} \rangle\!\rangle \boldsymbol{\Phi}_0^{(l+\frac{1}{2},m+\frac{1}{2})} = \tilde{\boldsymbol{Q}}^{(l+1)} + (1 - f_{sh})\lambda_d \mathbb{F}_d \langle\!\langle \mathbf{1} \rangle\!\rangle \boldsymbol{\Phi}_0^{(l+\frac{1}{2},m)} \ , \qquad \text{(Eq. (4.70) revisited)}$$

where the shifted LDA operator is defined as

$$\begin{aligned}\left(\mathcal{L}_{d,eig,sh} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \boldsymbol{\Phi}_0\right)_k \equiv & -\tilde{D}_{k+\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,k+1} - \Phi_{0,k}\right) + \tilde{D}_{k-\frac{1}{2}} \langle\!\langle \boldsymbol{f} \rangle\!\rangle \left(\Phi_{0,k} - \Phi_{0,k-1}\right) \\ & + \left(\bar{\Sigma}_{a,k} - f_{sh}\lambda_d \overline{(\nu\Sigma_f)}_k\right) \langle\!\langle \boldsymbol{f} \rangle\!\rangle \Phi_{0,k}\Delta_k \ ,\end{aligned}$$

(Eq. (4.69) revisited)

and f_{sh} exists in the range

$$0 \le f_{sh} < 1 \ .$$

As f_{sh} approaches unity, the shifted LDA operator becomes more critical, and thus, more singular. Compared to CMFD, there is no risk of “over-shifting”, in which the shift magnitude is large enough to result in an inaccurate solution (due to an extremely large condition number). With CMFD, one must be conservative with the shift magnitude because the upper bound is not known beforehand (as it is with LDA). This is a significant advantage of LDA compared to CMFD. However, numerical roundoff issues as a result of a large shift is a concern for both acceleration methods.

A set of test problems with 1000 fine cells and 5 fine cells per coarse cell (resulting in 200 coarse cells) was used to examine the effects of changing f_{sh}. The chosen metric for measuring the performance of PI is the total number of power iterations required for convergence (referred to as #PI for brevity). This value is the sum of the number of power iterations over all outer iterations. A range of f_{sh} was tested for the following values of fine cell width (h): 0.05, 0.04, 0.03, 0.02, and 0.01 cm. These choices of h correspond to coarse cell widths of 0.25, 0.20, 0.15, 0.10, and 0.05 cm, respectively. The left boundary condition is vacuum and the right is reflective. Each fine cell possesses the following properties: $\Sigma_t = 0.2 \text{ cm}^{-1}$, $c = 0.95$, and $\nu\Sigma_f = 1.0 \text{ neutrons/cm}$. The LU linear solver method is used for all problems. To check the accuracy of the solution, the shifted LDA scalar flux $\phi_{0,\text{LDA,sh}}$ (for $f_{sh} > 0$) was compared to the unshifted solution $\phi_{0,\text{LDA}}$ (for $f_{sh} = 0$) using the following equation:

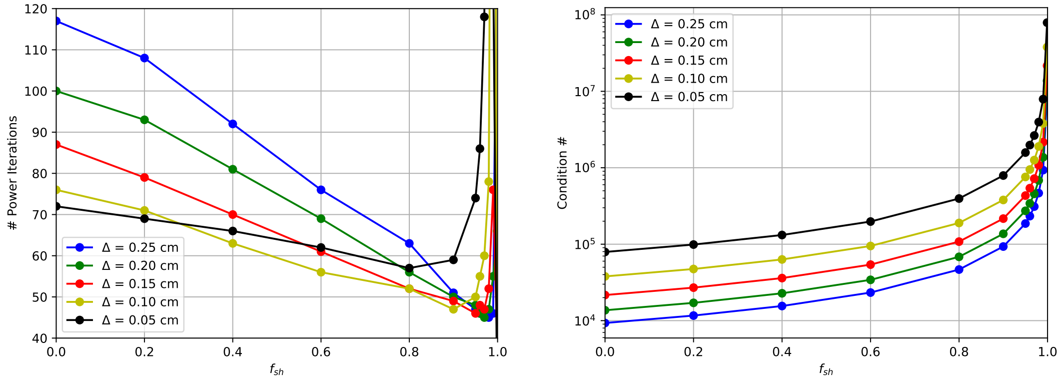
$$\text{Flux Error} \equiv \max_{j} \left(\frac{|\phi_{0,\text{LDA,sh}} - \phi_{0,\text{LDA}}|}{\phi_{0,\text{LDA}}} \right) . \tag{6.7}$$

Normally, the solution accuracy would not be relevant because LDA preserves the transport solution. However, in this context, we expect the accuracy of the solution to degrade as the condition number approaches ∞ (this is effectively true for any shifted eigenvalue problem). Therefore, the solution accuracy is reported in addition to performance metrics.

The results of these studies are shown in Fig. 6.6, with Fig. 6.6a showing the #PI and Fig. 6.6b showing the condition number. As Δ becomes smaller, the optimal value of f_{sh} corresponding to the minimum #PI decreases. This is expected behavior based on the data in Fig. 6.6b, which shows that the condition number for a given value of f_{sh} increases with decreasing Δ. At a certain Δ, the condition number becomes large enough to affect the convergence behavior of PI. We note that this is an extreme situation that is not likely to be encountered in practice. For $\Delta = 0.25$ cm, the optimal shift value is $f_{sh} = 0.98$. When $\Delta = 0.05$ cm, the optimal shift decreases to $f_{sh} = 0.8$. Therefore, the choice of f_{sh} is influenced by the spatial refinement of the problem.

For values of f_{sh} that are small enough not to cause numerical roundoff issues, the decrease in #PI decreases approximately linearly with f_{sh}. However, once the condition number approaches $O(10^6)$ for this problem, the convergence rate of PI starts to become affected by numerical roundoff error. Thus, we observe an increase in #PI for values of f_{sh} near unity. Further, when the condition number becomes very large, the solution accuracy becomes unacceptable. Tables 6.8 and 6.9 show the tabular data for $\Delta = 0.25$ cm and $\Delta = 0.05$ cm, respectively. From these tables, we see that the solution error becomes vastly greater than the convergence tolerance (10^{-8}) when the condition number approaches $O(10^7)$ for this problem.

From this study, we have determined that one can estimate a conservative value for f_{sh} that is applicable to a range of practical problems with similar spatial refinement. One can reasonably choose a conservative value for f_{sh} in the approximate range $0.8 - 0.95$. Choosing a shift value greater than 0.95 would offer little benefit, due to the linear decrease in #PI with f_{sh}. In Chapter 7, we determine the optimal shift value for a large, practical case.



(a) Number of power iterations required for convergence as a function of f_{sh} for different values of Δ

(b) LDA operator condition number as a function of f_{sh} for different values of Δ

Figure 6.6: Results of the f_{sh} studies

Table 6.8: Results of the Spectral Shift Study for $\Delta = 0.25$ cm

f_{sh}	#PI	Flux Error	Condition #
0.0	117	4.879E-09	9.319E+03
0.2	108	8.721E-10	1.165E+04
0.4	92	5.545E-10	1.553E+04
0.6	76	1.813E-09	2.330E+04
0.8	63	1.170E-09	4.659E+04
0.9	51	8.852E-10	9.318E+04
0.95	47	9.353E-10	1.864E+05
0.96	46	4.930E-09	2.330E+05
0.97	46	4.905E-10	3.106E+05
0.98	45	4.669E-09	4.659E+05
0.99	46	1.174E-09	9.318E+05
0.999	156	3.467E-02	9.318E+06

6.5 Discussion of LDA Linear Solver Methods

In this section, we consider the implications of the LDA method possessing a fixed LHS operator on the process of solving the low-order linear system. As discussed in Chapter 4, the LHS LDA operator is fixed for a given state. Additionally, the operator is generally sparse (containing relatively few non-zero elements compared to the matrix size). These properties can be taken advantage of in various ways, some of which will be briefly discussed here. Doing so can reduce the computational cost of solving the low-order problem for different source vectors (which change every power iteration) compared to other systems in which the LHS operator is not fixed (such as CMFD).

Table 6.9: Results of the Spectral Shift Study for Δ = 0.05 cm

f_{sh}	#PI	Flux Error	Condition #
0.0	72	4.767E-10	7.913E+04
0.2	69	1.429E-10	9.891E+04
0.4	66	1.503E-09	1.319E+05
0.6	62	1.563E-09	1.978E+05
0.8	57	1.639E-09	3.956E+05
0.9	59	1.998E-09	7.913E+05
0.95	74	2.560E-09	1.583E+06
0.96	86	5.813E-09	1.978E+06
0.97	118	1.126E-08	2.638E+06
0.98	341	6.754E-08	3.956E+06
0.99	130	7.993E-02	7.913E+06
0.999	24	1.890E-01	7.913E+07

6.5.1 Direct Solver Considerations

The cost of solving a linear system directly is usually considered prohibitive for large problems, as the cost scales with n^3 (for an $n \times n$ matrix). However, this cost can be reduced by decomposing the operator in a way that allows for reduced computational cost when solving the linear system. For the LU method (discussed in Section 2.4.3.1), the LDA operator can be decomposed once before the iterative process begins, which classically requires $O(n^3)$ operations to factor the operator. Then, the computational complexity reduces to $O(n^2)$ for each direct linear solve (using forward and backward substitution). In this way, the upfront work of factoring the operator is more than made up for by the reduced cost of solving the system (which is done every power iteration). This method would not be beneficial for CMFD, since the operator is not fixed.

We note that though the LU method can reduce the computational expense of the low-order step, it will increase the memory requirement. The low-order operator (for both LDA and CMFD) generally takes the form of a sparse matrix, but the LU decomposition will not possess the same sparsity. For a given $n \times n$ matrix, the fill-in (number of nonzero elements) for the matrix factors will generally be higher than the original matrix. This could significantly increase the memory requirements of the algorithm, especially for large, multigroup problems. However, the literature suggests that some degree of sparsity can be preserved for the matrix factors through specific procedures [46, 47]. This includes methods of dissecting the operator in a way that maximally leverages its sparsity structure, which reduces the fill-in of the LU factors [48]. However, the memory cost is still shown to become prohibitive for large 3D problems. Specialized software also exists for preserving the sparsity of the matrix factors, such as the SuperLU software package [69].

In a practical application, the size of the low-order operator can be $O(10^8)$ or larger (as discussed in Chapter 2), but is generally sparse. If the decomposition method does not preserve a high degree of sparsity for the matrix factors compared to the LDA operator, then the method discussed here becomes impractical due to computer memory limitations. Further, most neutronics applications that are used for large-scale problems (such as MPACT) require a high degree of parallelism to simulate realistic problems. Thus, the software employed to decompose the LDA operator and perform the direct linear solve must also employ parallelism and possess favorable parallel scaling. If this is not the case, then the LU method will likely be less favorable compared to a highly-efficient iterative solver method. A suggestion for future work is to implement the LU method in MPACT in a way that avoids the aforementioned issues, and compare the performance to iterative solvers such as GMRES.

6.5.2 Iterative Solver Considerations

Regarding iterative Krylov solvers such as GMRES or the *Biconjugate Gradient Stabilized* (BiCGSTAB) method [70], a preconditioner is generally used to improve the convergence of solving the low-order linear system. The preconditioner can consist of two matrices whose product approximates the original operator, and possesses a smaller condition number compared to the original operator. Because the convergence rate of Krylov methods depends on the condition number of the matrix, this can dramatically improve the convergence rate. If such an iterative method is used to solve the LDA linear system, then some computational work may be saved compared to CMFD because the LDA operator is fixed. For example, we consider the *Incomplete LU* (ILU) preconditioner [71]. In this case, the operator is decomposed into a lower and upper triangular matrix, and small values are "dropped", or omitted, to reduce the memory cost of storing the factors. As a result, the product of these matrices approximates the original matrix rather than producing the exact original. If this is done once before the outer iterations begin, then the preconditioner will always resemble the LDA matrix, because it is constant for a given state. We suggest the investigation of parallel ILU preconditioners in MPACT as future work. (A parallel method for computing the ILU preconditioner is considered difficult, but work on this topic exists in the literature [72, 73].)

Turning our attention to CMFD, we note that for this method, the low-order operator is not fixed. In this case, the preconditioner may need to be recomputed during the iteration process to retain the improved convergence rate if the CMFD operator changes significantly as iterations progress. These changes are likely to be larger during the first few outer iterations, as the homogenized cross sections and transport consistency terms converge. Thus, the preconditioner may be less effective for later iterations.

As further future work, we suggest the implementation and testing of solver methods that are focused on linear systems with multiple right hand sides in a practical application such as MPACT. "Linear systems with multiple right hand sides" refers to systems in which the LHS operator is fixed, and we wish to solve multiple systems with this same LHS and different RHS vectors. Many methods exist that are focused on situations in which all RHS vectors are known before solving. However, this is not the case with LDA because the RHS vector changes every power iteration. Therefore, the method must be conducive to these types of situations. Some methods make use of *related* RHS, which could potentially be applicable to LDA since the RHS vector contains lagged terms that we expect to be "small". In practice, this should mean that the RHS vector of the LDA low-order system does not change significantly between outer iterations (especially when the problem is close to convergence).

A concept for Krylov-based methods (such as GMRES) in which the Krylov subspace is generated and reused is called *Krylov subspace recycling*. This is a way of reducing computational work by saving the Krylov subspace that is generated in the process of obtaining an iterative solution. One such method that fits these requirements described above is a variation of GMRES called MHGMRES [74] (the acronym is not defined in the source). This method generates a Krylov subspace for the residual of one system and projects the remaining residuals onto that subspace, and employs certain methods to share information among the systems. The MHGMRES method is shown to perform best when the right-hand sides share spectral information. Another method called Generalized Conjugate Residual Method with Inner Orthogonalization and Deflated Recycling (GCRO-DR) makes no assumptions about the relationship between each right hand side vector, and focuses on the selection of the subspace to be recycled [75]. Iterative methods such as these that take advantage of the fixed LHS operator and similar RHS vectors could reduce the computational cost of solving the linear system compared to more traditional iterative solvers.

We further consider more recent direct linear solver technology related to computing the LU decomposition of an operator [48]. This work is recent and is the focus of current research in the field of numerical analysis, and the efficient, parallel implementation of the method discussed here is stated to be a challenging task. In this reference, there is a discussion of a fast direct sparse solver (referred to as FDSS). Because the low-order operator is a sparse system, this method is appropriate to apply to LDA. Though the name of the solver implies its use to solve linear systems directly, it is more appropriate to generate a preconditioner for use with an iterative solver. This is because the computational cost of the method scales with the required accuracy of the decomposition. For 3D problems, this cost can become prohibitive if high accuracy is desired (as would be the case for LDA). However, the method becomes significantly cheaper if the desired accuracy is low. Thus, this method could be used to generate an ILU decomposition of the LDA operator for use with an iterative method. The basis of the method is called *nested dissection partitioning*, and the goal is to cleverly decompose and restructure the matrix into sub-matrices, which can then be compressed to allow for reduced computational work when obtaining the LU decomposition. This is shown, in the most favorable case, to reduce the computational cost of the LU decomposition to $O(n)$ compared to $O(n^3)$ for the classical implementation. Though this the the best case scenario, this method could reduce the cost of generating the ILU preconditioner for an iterative method such as GMRES or BiCGSTAB.

6.6 Summary

In this chapter, we investigate the use of LDA in a 1D research code. From the performance comparisons of LDA and CMFD in Section 6.2, LDA is shown to be more susceptible to linear instability compared to CMFD. These differences were observed in the Fourier analysis for fixed-source problems, but not eigenvalue problems. Different choices for coarse-mesh weighting functions, the prolongation operator, or boundary conditions may be the cause of these discrepancies. However, the performance difference is significant only for cases in which the optical thickness is large and both methods are near divergence. For values of Λ near 1.0, the performance difference is tangible but not significant. The use of weighting functions that more resemble the converged scalar flux may improve the convergence properties of LDA.

Though the performance difference between LDA and CMFD is unfortunate, we have also demonstrated the lack of nonlinear instability in LDA, *which is the primary motivation of developing the method*. In Section 6.3, we showed that LDA is insensitive to the nonlinearities of CMFD through contrived problems in which the nonlinear terms of CMFD are intentionally forced to become unstable. This is done through the introduction of a negative source into one fine cell, which emulates situations in which a negative flux solution becomes possible. For a range of source values, CMFD becomes divergent. Further, the performance of CMFD is shown to degrade when the method nears divergence as a result of nonlinear instability. For these same problems, LDA remains stable.

We also present results of studies that show improved power iteration convergence properties as a function of the LDA shift parameter (f_{sh}). As f_{sh} increases, the number of power iterations required for convergence decreases up to a certain point. Once the condition number of the LDA operator becomes sufficiently large, the method begins to suffer from numerical roundoff. As the problem becomes more spatially refined, condition number increases more quickly with f_{sh}. Thus, the optimal value of f_{sh} depends on the spatial discretization of the problem. However, one can reasonably estimate a value of f_{sh} for a wide variety of practical problems with similar spatial refinement. Further, the risk of over-shifting is not present with LDA as it is with CMFD. With CMFD, the shift magnitude must be estimated and chosen conservatively to avoid over-shifting.

Finally, we discuss the advantages offered by LDA pertaining to solving the low-order linear system. Because the LDA operator is fixed and sparse, specialized solver methods can be used to reduce the computational cost of solving the LDA system compared to those methods typically used for CMFD. One direct method is LU decomposition, which can reduce the solve cost but increase the memory requirement. Other iterative methods take advantage of the fixed operator using Krylov subspace recycling, which improves the convergence rate of the solver method by reusing previously-computed subspaces. Other methods are focused on reducing the memory requirements and computational cost of obtaining a complete or incomplete LU decomposition. We suggest the implementation and testing of these methods in MPACT as future work.

CHAPTER 7

MPACT Numerical Results

The focus of this chapter is the use of LDA in the neutron transport code MPACT. First, we provide some relevant implementation details for LDA in MPACT in Section 7.1. Then, we perform a study to determine a suitable shift fraction using the *Virtual Environment for Reactor Applications* (VERA) problem 5a [76] in Section 7.2. In Section 7.3, the performance of LDA is compared to CMFD for several practical cases. The relative numerical stability of LDA compared to CMFD is demonstrated in Section 7.4. Finally, a summary of the results in this chapter is provided in Section 7.5.

7.1 Implementation Details

MPACT [11, 77] is a deterministic neutron transport code that uses the 2D/1D method [38] to solve the NTE. The *Method of Characteristics* (MOC) is used to obtain the radial solution in the x-y plane. For the axial direction, a nodal solver is used to solve the 1D P_3 equations on the coarse mesh. To accelerate the convergence of the transport problem, and to provide a framework for coupling the 2D and 1D solutions, various forms of CMFD are used. In contrast to the 1D code used in Chapter 6, the acceleration step is performed before the transport sweep. This choice should not significantly impact the performance of the accelerated algorithm.

The LDA method was implemented in MPACT using a similar algorithm as that described in Section 4.4, except for the following differences:

1. the acceleration step takes place before the transport sweep (as with the CMFD implementation in MPACT),
2. the low-order problem is 3D and multigroup, and
3. the problem is not normalized using the forward diffusion scalar flux solution,
4. the convergence criteria are different.

Regarding the first difference, this change should not have a substantial effect on the convergence properties of the accelerated method. In the first outer iteration, the low-order problem is solved using the initial guess for the scalar flux (rather than the solution from the transport sweep, which has not yet been performed). Regarding the second difference, the low-order problem becomes a sparse block matrix rather than a tridiagonal matrix (as it is in 1D, 1-group cases).

Regarding the third difference, there is a discussion in Chapter 4 about the normalization of the low-order LDA scalar flux during PI. In Chapter 4, the scalar flux is normalized during PI using the forward diffusion solution. However, normalizing this way is unnecessary because the low-order LDA problem is not of a form to which the FAT applies during the PI process. Therefore, the scalar flux can be normalized using traditional means. For the MPACT implementation of LDA, the scalar flux is not normalized using the forward diffusion solution. Thus, some computational expense is spared because the converged forward diffusion solution does not need to be obtained.

Regarding the fourth difference, the default convergence criterion for each of the iteration levels are as follows. For the outer iteration convergence criteria, the following conditions are used:

$$\sqrt{\frac{1}{N_{fj}}\sum_{j}\left(f_j^{(l)} - f_j^{(l-1)}\right)^2} < 5 \cdot 10^{-5} \,, \tag{7.1}$$

where l is the outer iteration index, N_{fj} is the number of spatial cells on the transport grid containing a fission source, and f_j is the normalized fission source in the fine cell j. Additionally, the following convergence criterion is imposed on the eigenvalue:

$$\left|\lambda^{(l)} - \lambda^{(l-1)}\right| < 10^{-6} \,. \tag{7.2}$$

Outer iterations cease once these two conditions are met.

For the convergence criterion of the low-order CMFD problem, power iterations cease once the normalized residual of the linear system is reduced by a factor of 100. If the low-order problem is written in operator notation, as below:

$$\mathbb{M}^{(l)}\phi_0^{(l,m+1)} = \lambda^{(l,m)}\mathbb{F}^{(l)}\phi_0^{(l,m)} \,, \tag{7.3}$$

where m is the power iteration index, $\mathbb{M}$ is the migration operator, and $\mathbb{F}$ is the fission operator, then the convergence condition is

$$\frac{\left(\frac{\left\|\mathbb{M}^{(l)}\phi_0^{(l,m)}-\lambda^{(l,m)}\mathbb{F}^{(l)}\phi_0^{(l,m)}\right\|}{\left\|\phi_0^{(l,m)}\right\|}\right)}{\left(\frac{\left\|\mathbb{M}^{(l)}\phi_0^{(l,0)}-\lambda^{(l,0)}\mathbb{F}^{(l)}\phi_0^{(l,0)}\right\|}{\left\|\phi_0^{(l,0)}\right\|}\right)} < 0.01\ . \tag{7.4}$$

For the power-iteration dependent eigenvalue, the following condition is imposed:

$$\left|\lambda^{(l,m)} - \lambda^{(l,m-1)}\right| < 10^{-6}\ . \tag{7.5}$$

Because the LDA linear system is not set up in the same way as the CMFD system, and contains a source term that is fixed during the power iteration process, a different convergence criterion is used. For reference, the structure of the low-order LDA problem is reproduced below:

$$\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+1\right)} = \tilde{\boldsymbol{Q}}^{(l+1)} + \lambda_d \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{\left(l+\frac{1}{2},m\right)}\ . \qquad \text{(Eq. (4.70) revisited)}$$

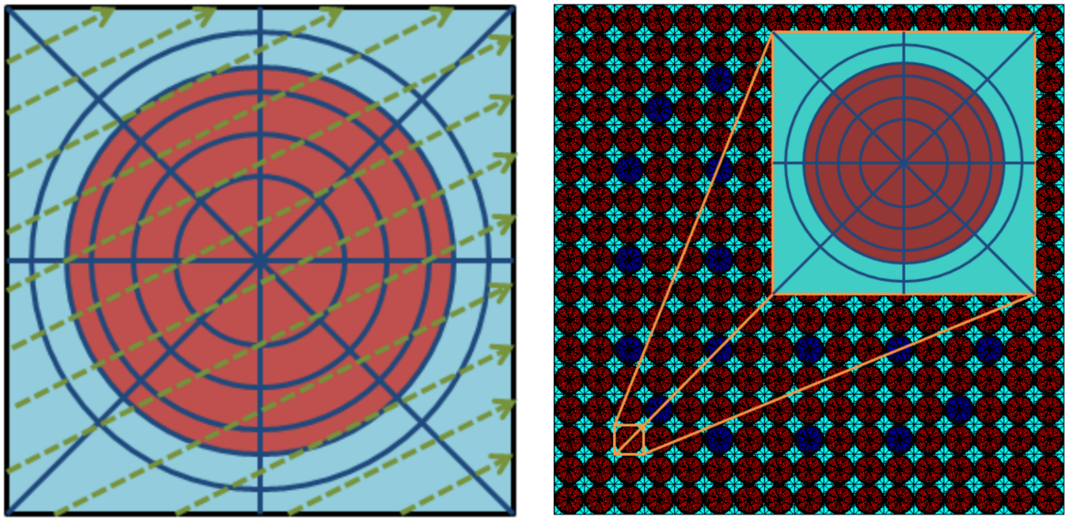
If the same convergence criterion is used for LDA, then the relative magnitude of $\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{\left(l+\frac{1}{2},m+1\right)} - \lambda_d \mathbb{F}_d \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0^{\left(l+\frac{1}{2},m\right)}$ and $\tilde{\boldsymbol{Q}}^{(l+1)}$ become a factor for convergence. To avoid issues and confusion related to this, a simpler condition is imposed that avoids the need to include $\tilde{\boldsymbol{Q}}^{(l+1)}$. Power iterations cease once the following criterion is met:

$$\frac{\max\limits_k\left(\frac{\left|\boldsymbol{f}^{(l,m)}-\boldsymbol{f}^{(l,m-1)}\right|}{\boldsymbol{f}^{(l,m-1)}}\right)}{\max\limits_k\left(\frac{\left|\boldsymbol{f}^{(l,0)}-\boldsymbol{f}^{(l-1,M_{l-1})}\right|}{\boldsymbol{f}^{(l-1,M_{l-1})}}\right)} < 0.01\ , \tag{7.6}$$

where k is the low-order grid index, M_l is the number of power iterations executed in outer iteration l, and $\boldsymbol{f}$ is the normalized fission source on the low-order grid. Thus, we have replaced Eq. (7.4), which checks 2-norms of the residual of the low-order problem, with an ∞-norm for successive changes in $\boldsymbol{f}$. (The use of the ∞-norm for the convergence criterion was chosen conservatively.) As discussed in Section 4.4.2, the transport eigenvalue is updated once every outer iteration using Eq. (4.59). Because the eigenvalue is not updated during every power iteration (as it is with CMFD), it does not play a role in the convergence of LDA power iterations. Rather, the convergence of the transport eigenvalue is checked using Eq. (7.5) once every outer iteration.

As previously mentioned, the low-order calculation is performed before the transport sweep in MPACT. Therefore, in the first outer iteration, there is no transport correction information. Thus, we set $\tilde{Q}$ to zero in the first outer iteration. Once this is done, the LDA equations reduce to the neutron diffusion equations. In this way, we obtain the adjoint diffusion solution before the outer iterations begin. The tolerance for the adjoint diffusion calculation is conservatively set to be a fixed fraction of 0.005 times the imposed outer iteration convergence criterion on the fission source to ensure sufficient convergence of the diffusion solutions. Because the adjoint diffusion solution is used to compute the transport eigenvalue, a converged solution must be obtained.

Unless otherwise stated, the default MPACT settings for code version 4.3 are used with the 51-group cross section library. Selected default settings are listed in Table 7.1. In MPACT, the fuel pins are divided into a specified number of radial and azimuthal regions that make up the fine mesh. Characteristic rays with a specified spacing are cast through these regions for the MOC calculation, as shown in Fig. 7.1a. A coarse cell for the low-order problem is generally composed of an axial slice of a single pin cell. Pin cells are arranged in an assembly as shown in Fig. 7.1b, which are arranged in a specified pattern to comprise the reactor core.



(a) Example transport mesh (blue lines) and characteristic rays (dashed green arrows) in a fuel pin

(b) Example assembly-level discretization

Figure 7.1: MPACT MOC discretization [42]

7.2 Wielandt Shift Study

In Section 6.4, the optimal shift parameter f_{sh} for LDA was found to be dependent on the spatial discretization of the problem. We wish to determine the best shift for a realistic

Table 7.1: Selected MPACT Default Settings

Setting	Value
Quadrature set	Chebyshev-Yamamoto
Azimuthal angles per octant	16
Polar angles per octant	2
MOC Ray spacing	0.05 cm
Scattering treatment	TCP_0
# Radial fuel divisions	3
# Azimuthal fuel divisions	8
# Radial moderator divisions	1
# Azimuthal moderator divisions	8

case, with a much larger linear system. To do so, a study was conducted in which the shift parameter f_{sh} was varied in MPACT for the 2D version of VERA problem 5a with default settings. This problem was chosen due to its size and complexity. Parallelization was used in the form of spatial decomposition, with 8 MPI processes.

The 2D version of VERA Problem 5a is a radial slice of a quarter-core at *Beginning of Life* (BOL) (no fuel burnup) and *Hot Zero Power* (HZP) (operating temperature but very low power) conditions. Reflecting boundary conditions are used on the top and left sides and vacuum boundary conditions on the bottom and right. This model contains a reflector, barrel, and neutron pad in the regions surrounding the core. Westinghouse 17x17-type fuel assemblies are used, with various fuel enrichments used in different fuel regions. The core configuration is shown in Fig. 7.2. Complete details of VERA Problem 5a are available in [76].

The results of the study are shown in Table 7.2. Each table entry is described below:

1. f_{sh}: the value of the shift parameter as defined in Eq. (4.69),
2. Power Iterations: total number of power iterations executed across all outer iterations,
3. Runtime (s): runtime of the full calculation in seconds,

From the results, we observe that the computation effort (in terms of runtime and the total number of power iterations) does not monotonically decline as f_{sh} increases for small values of f_{sh}. We note that by default, there is a limit of 20 power iterations per outer iteration in MPACT. The total runtime is observed to increase with f_{sh} for $f_{sh} < 0.8$. This behavior can be explained with the following rationale: the condition number of the low-order operator is increasing with f_{sh}, which outweighs the effect of reducing the number of required power iterations. Because the condition number increases with f_{sh} (as seen in Chapter 6),

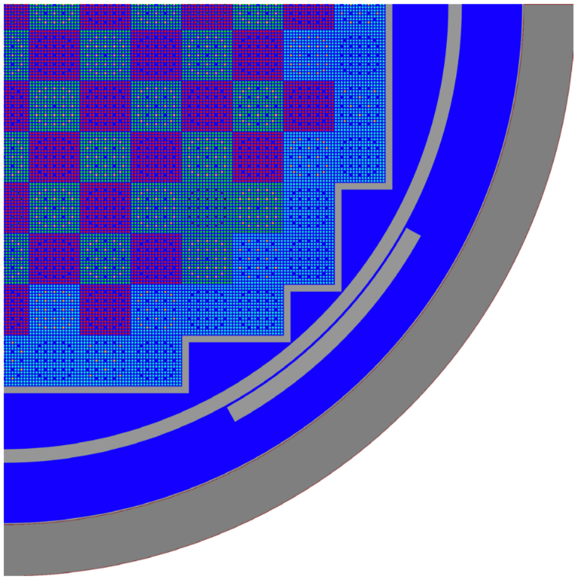


Figure 7.2: VERA Problem 5a core geometry [76]

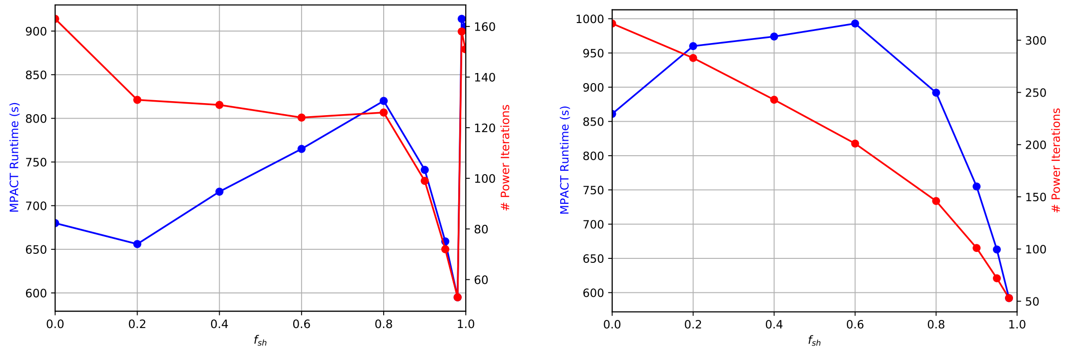
the difficulty of solving the linear system using GMRES increases (since the convergence rate of GMRES iterations is dependent on the matrix condition number). This suggests that f_{sh} should either be set to zero or near unity.

Table 7.2: MPACT Shift Study Results for Problem 5a 2D w/ PI Limit = 20

f_{sh}	Power Iterations	Runtime (s)
0.0	163	680
0.2	131	656
0.4	129	716
0.6	124	765
0.8	126	820
0.9	99	741
0.95	72	659
0.98	53	595
0.99	158	914
0.999	151	905

The results of this study suggest that the optimal value of f_{sh} is approximately 0.98, as can be seen from the plot of the runtime and total number of power iterations vs. f_{sh} in Fig. 7.3. For f_{sh} = 0.98, the total runtime and number of power iterations was minimized. Thus, this is the recommended value for this problem. However, the optimal value will likely be different for other cases due to the wide range of problems that are simulated with MPACT. If f_{sh} is greater than 0.98, the computation effort is seen to increase. This is likely a result of the large condition number of the near-singular operator. In the range $0.98 < f_{sh} < 1.0$, the improved convergence of PI is overshadowed by the difficulty of convergence encountered by GMRES. An increase in required power iterations in this range indicates that the linear solver has trouble producing a converged solution.

For comparison, we also provide the results of the same study with the maximum number of power iterations per outer iteration set to 1000. Tabular and graphical data are shown in Table 7.3 and Fig. 7.3b, respectively. We observe the same trend in these cases, with the runtime increasing with f_{sh} for small values of f_{sh}. The only values of f_{sh} that result in reduced runtime compared to the unshifted case $(f_{sh} = 0.0)$ are 0.9, 0.95, and 0.98, with $f_{sh} = 0.98$ again providing the smallest runtime. For greater values of f_{sh}, the number of power iterations reached the imposed limit of 1000, so these results are not reported. This behavior is again likely a result of the large condition number of the LDA operator, leading to convergence issues when iteratively solving the linear system with GMRES.



(a) Power Iteration Limit = 20 (default)

(b) Power Iteration Limit = 1000

Figure 7.3: MPACT runtime and # power iterations vs. f_{sh} for VERA Problem 5a 2D

Table 7.3: MPACT Shift Study Results for Problem 5a 2D w/o PI Limit = 1000

f_{sh}	Power Iterations	Runtime (s)
0.0	316	861
0.2	283	960
0.4	243	974
0.6	201	993
0.8	146	892
0.9	101	755
0.95	72	663
0.98	53	592
0.99	*	*
0.999	*	*

We observe a small range of f_{sh} in which the computational effort is reduced for the default settings: $0.95 \le f_{sh} \le 0.98$. A value of f_{sh} outside this range may not be beneficial for a large problem such as the one tested here. If the previously-discussed LU method is used, which has not been implemented in MPACT, then we may observe more of a monotonic decline in computational effort with f_{sh}, as observed in Section 6.4 (since the increasing condition number with f_{sh} will not affect the performance of a direct linear solver).

7.3 LDA Performance in MPACT

In this section, we explore the performance of LDA in MPACT in comparison to the CMFD method for several practical cases. *The studies here investigate how the linear stability of LDA and CMFD compare for cases in which the nonlinear instability of CMFD is not present.* We compare the performance of the two acceleration methods for the following practical cases:

1. VERA Problem 3a: A 3D Westinghouse 17x17 assembly at BOL and HZP conditions with 3.1% enriched UO_2 fuel in quarter symmetry. This is one of the assembly types present in VERA problem 5a. The radial boundary conditions are reflective and axial boundary conditions are vacuum. Figure 7.4 shows the quarter-assembly radial geometry plot for this problem.

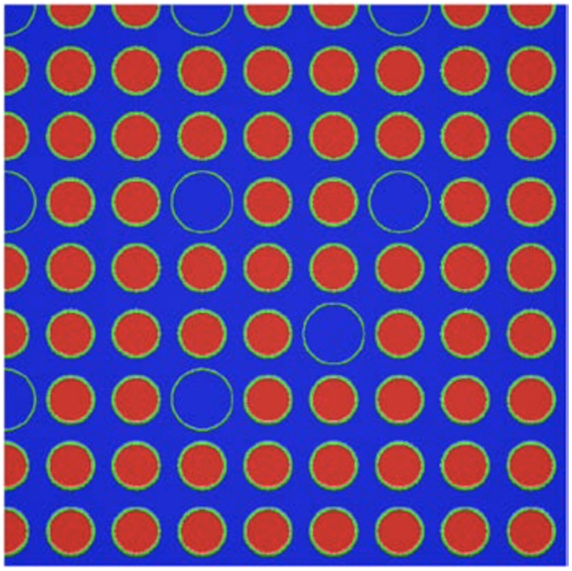


Figure 7.4: VERA Problem 3a geometry. The colors indicate fuel (red), cladding (green), and moderator (blue). The guide tubes are visible as empty tubes. [76]

2. VERA Problem 4a: Nine 3D Westinghouse 17x17 fuel assemblies arranged in a 3x3 checkerboard pattern (taken directly from the center of VERA problem 5a) at BOL and HZP conditions in quarter symmetry. The radial boundary conditions are reflective and the axial boundary conditions are vacuum. Figure 7.5 shows the radial quarter-symmetry geometry for this problem.

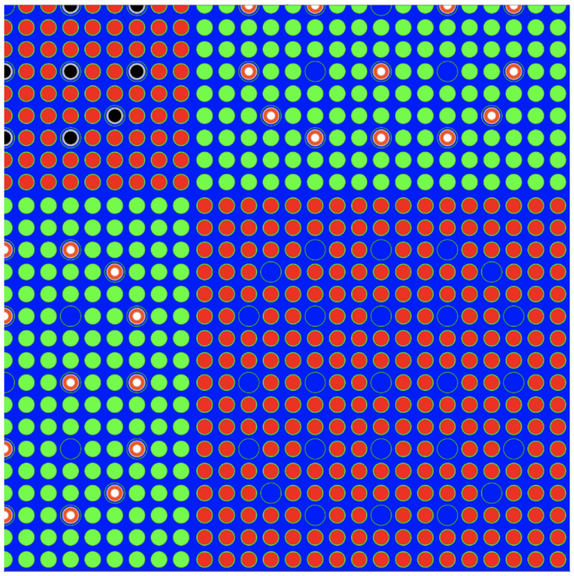


Figure 7.5: VERA Problem 4a geometry. The colors indicate fuel (red), cladding (green), and moderator (blue). Control rods are shown as black in the center assembly and Pyrex rods are shown as yellow. [76]

3. VERA Problem 5a: Described in Section 7.2.
4. C5G7: A 3D benchmark problem with two UO_2 fuel assemblies and two MOX assemblies with reflective boundary conditions on the top and left sides and vacuum boundary conditions on the bottom and right [35]. Moderator surrounds the assemblies on the non-reflective sides. The unrodded configuration was used, with no control rods inserted. Figure 7.6 shows the geometry for this problem.

Further details for each of these problems can be found in the following references: [35, 76].

For CMFD, the "adaptive" shift method in MPACT is used for the low-order eigenvalue problem (as described in Section 3.4.2). Additionally, the optimally-diffusive version of CMFD (odCMFD) [25] is used. This method adjusts the diffusion coefficient of each coarse cell based on its optical thickness to improve the convergence rate. The LDA results do not include these adjustments, so we expect slightly worse convergence behavior for LDA compared to odCMFD. For LDA, the value of f_{sh} = 0.98 determined in Section 7.2

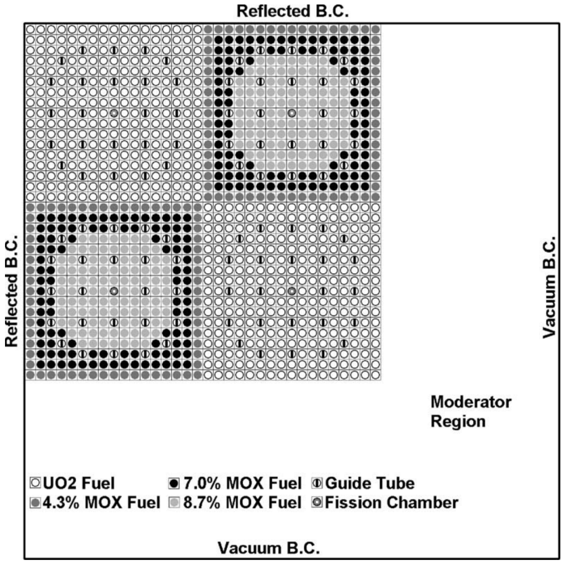


Figure 7.6: C5G7 Benchmark radial geometry [35]

is used. Otherwise, the default MPACT settings are used. We compare the total runtime using each method, as well as the number of outer iterations required for convergence. For each outer iteration, the number of power iterations is reported.

Table 7.4 presents the following performance metrics for each case:

1. CMFD OI: the number of outer iterations required for convergence when using CMFD as the low-order acceleration method,
2. LDA OI: the number of outer iterations required for convergence when using LDA as the low-order acceleration method,
3. CMFD Runtime (s): the total runtime in seconds of the calculation with CMFD as the acceleration method,
4. LDA Runtime (s): same as above for LDA, and
5. # Cores: the number of MPI processes used for the spatially decomposed problem.

Table 7.4: Comparison of LDA and CMFD Performance Metrics

Case	CMFD OI	LDA OI	CMFD Runtime (s)	LDA Runtime (s)	# Cores
VERA 3a	9	10	55	78	29
VERA 4a	12	12	553	873	29
VERA 5a	8	11	455	595	8
3D C5G7	13	13	328	387	60

From these results, we observe that the LDA runtime is higher than the CMFD runtime for all cases, and significantly higher for some. This is an unfortunate consequence of the following two contributors: (i) the initial adjoint diffusion calculation and (ii) the slightly different convergence rate of LDA compared to CMFD. For the former effect, the initial adjoint diffusion calculation is an additional expense that is used to update the transport eigenvalue in each outer iteration. This calculation requires a non-trivial amount of computational resources in MPACT to converge to the required tolerance, which is contrary to the discussion of the low-order problem in Chapter 2. Such an issue may be inherent to large, realistic, 3D calculations. This effect is the most significant contributor to the additional runtime for LDA compared to CMFD (which does not require this initial adjoint diffusion solution).

To demonstrate the effect of the additional adjoint calculation on the runtime, we provide the runtime of just the initial adjoint diffusion calculation in Table 7.5. In this table, the following metrics are provided:

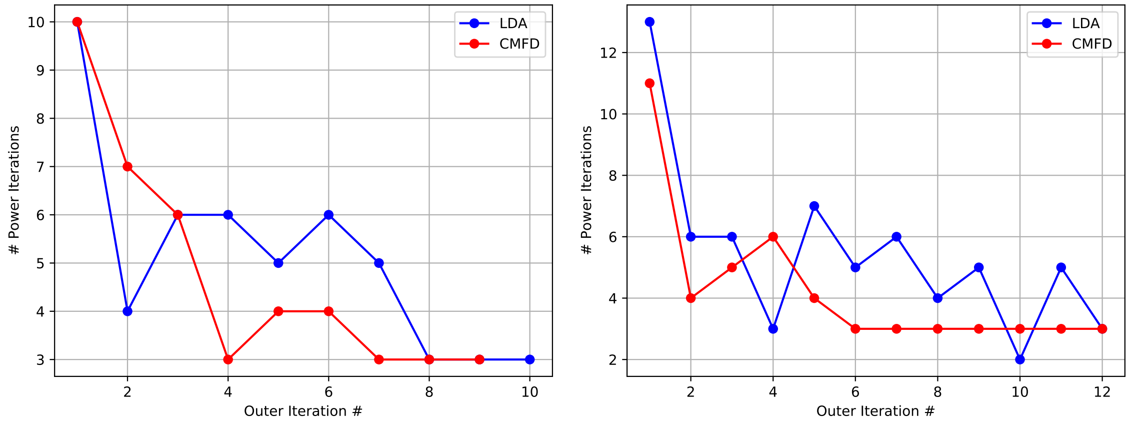
1. Adjoint Flux Calculation (s): the time in seconds required to converge the initial adjoint flux calculation only,
2. CMFD Runtime (s): same as above, and
3. LDA Runtime w/o AC: the total LDA runtime *minus* the time to complete the initial adjoint diffusion calculation.

Table 7.5: LDA Adjoint Flux Computational Cost

Case	Adjoint Flux Calculation (s)	CMFD Runtime (s)	LDA Runtime w/o AC
VERA 3a	23	55	55
VERA 4a	305	553	568
VERA 5a	110	455	485
3D C5G7	50	328	337

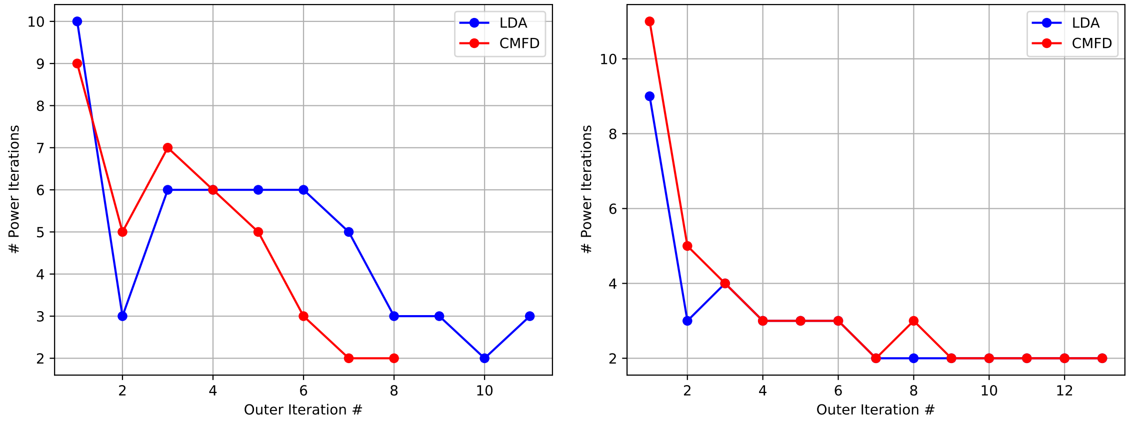
From the results in this table, we can see that the LDA runtime minus the time to complete the initial diffusion calculation is much closer to the CMFD runtime. The time requirement and efficiency for the low-order solve in MPACT is a known issue and has served as partial motivation for previous work [78].

As seen from the results in Table 7.4, LDA-accelerated cases may require a few additional outer iterations for convergence compared to CMFD. This is somewhat expected from the 1D results in Chapter 6, which show that the spectral radius of LDA is slightly worse than CMFD for spatially heterogeneous, finite problems. Additionally, odCMFD is used for these results, which slightly improves the convergence rate of the method compared to standard CMFD (which was tested in Chapter 6). We are also interested in the number of power iterations required for convergence within each outer iteration using the chosen f_{sh} value, which indicates the effectiveness of the shift method. In Fig. 7.7, we observe that the number of required power iterations for LDA is similar to CMFD using the chosen shift methods. The number of additional power iterations for LDA is small for all cases and does not exceed 3. The use of a more conservative norm for the PI convergence criterion (∞-norm for LDA compared to L_2-norm for CMFD) is likely contributing to the increase in the number of power iterations compared to CMFD.



(a) VERA Problem 3a

(b) VERA Problem 4a



(c) VERA Problem 5a

(d) C5G7 3D

Figure 7.7: Number of power iterations executed in each outer iteration

The results in this section suggest that the additional computational expense for LDA compared to CMFD is largely due to the initial adjoint diffusion calculation. Some additional work is due to the extra power iterations per outer iteration and additional outer iterations, but this is small relative to the former contribution. The use of a Wielandt shift for the adjoint diffusion calculation could potentially reduce this additional computational expense. Reducing the LDA linear solve time (potentially using one of the methods discussed in Section 6.5) may also help to offset the additional expense.

7.4 Nonlinear Instability Case Study

In this section, we investigate an example problem in which the nonlinearity of the CMFD method causes numerical instability in MPACT. When the problem is accelerated with LDA, the numerical instability is not observed. The problem in question is a 2D slice of a 2x2 GE-12 assembly case with the control blade inserted [79]. Each 10x10 assembly contains fuel pins of various enrichments. Large water rods are present in this assembly type as shown in Fig. 7.8. For this problem, the TCP_0 scattering treatment is used. The ray spacing is set to 0.01 cm, and there are 8 and 2 azimuthal and radial angles per octant, respectively. There are 5 radial divisions in the fuel and 1 in the moderator. For the water rods, there are 7 radial divisions in the rod and 1 in the outer region. The 47-group MPACT cross section library is used. Finally, the flux and eigenvalue tolerances are both set to $5 \cdot 10^{-5}$. We vary the number of transport sweeps per outer iteration for this problem (which is 1 by default). Otherwise, the default settings are used.

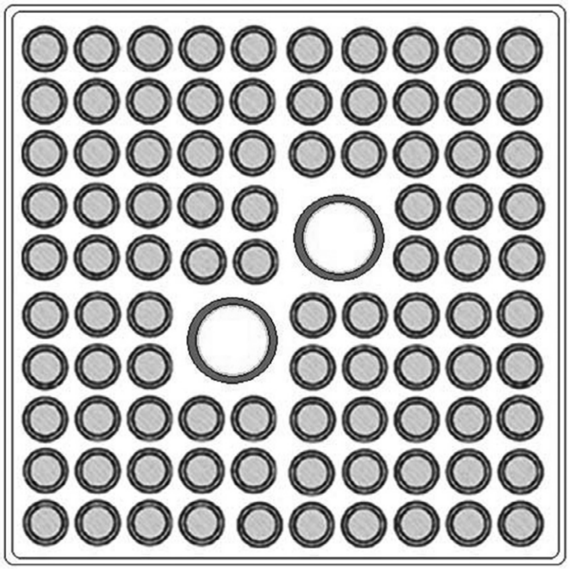


Figure 7.8: GE-12 assembly radial geometry [80]

When the number of transport sweeps per outer iteration is set to 1, the problem converges with CMFD. However, when the number of transport sweeps per outer iteration is set to 2, the problem diverges with CMFD. For shorthand, we refer to the number of transport sweeps per outer iteration as TS/OI. The underlying issue responsible for the divergence is most likely due to the presence of negative cross sections on the transport grid from TCP$_0$ scattering treatment, which invoke the nonlinear instability of CMFD. However, the presence of negative cross sections alone is not enough to cause this issue. The code must also encounter a state from which it cannot recover, which occurs when TS/OI is changed. When the scattering treatment is changed to P$_2$, the case converges with CMFD with no issues.

For the divergent case (TS/OI = 2), we inspect the value of the eigenpair residuals and $\hat{D}$ when using the CMFD method, as well as the presence of negative flux values on the transport and low-order grids. These metrics are tabulated in Table 7.6 as a function of outer iteration #, with the following datasets:

1. Eigenvalue Res.: the difference in the eigenvalue as defined in Eq. (7.2),
2. Fission Source Res.: the fission source difference as defined in Eq. (7.1),
3. # Neg. FF: the number of negative scalar flux values on the transport grid, and
4. # Neg. CF: the number of negative scalar flux values on the low-order grid.

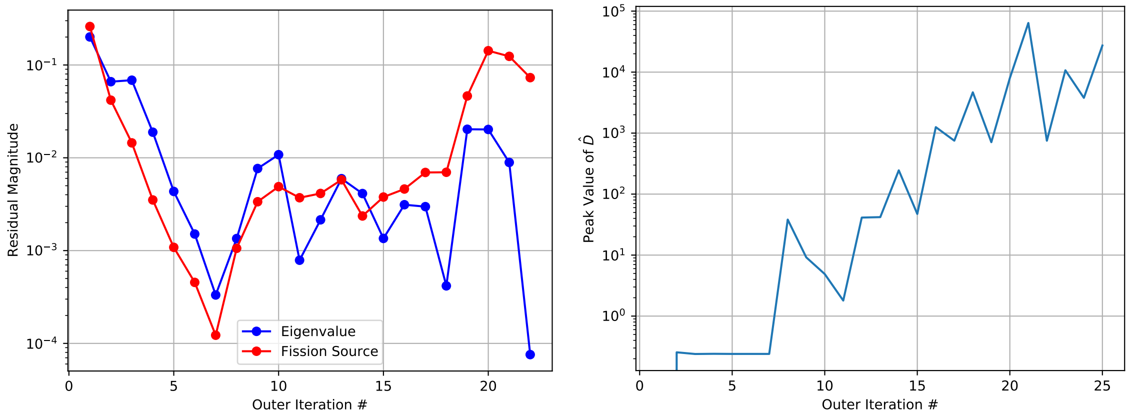
When using CMFD as the acceleration method, we observe divergence beginning in outer iteration 8. At this point, the fission source and eigenvalue residuals stop decreasing, as shown in Fig. 7.9a, where each residual at the end of a given outer iteration is plotted. Additionally, the peak value of $\hat{D}$ begins to increase significantly at this point in the iteration process, as shown in Fig. 7.9b. Negative flux values appear on the fine mesh very early in the iteration process, but do not result in negative coarse mesh values until iteration 8. We can infer that the presence of negative cross sections on the coarse grid destabilize the value of $\hat{D}$, ultimately leading to divergence.

Flux-weighted cross sections are not used in the low-order problem when LDA is used as the acceleration method, so negative coarse-grid cross sections are not expected. Additionally, LDA does not possess nonlinear terms such as $\hat{D}$. Therefore, the problem converges with no issues. Upon examining the same metrics as before with LDA in Table 7.7, we observe that no negative cross sections appear on the coarse grid. Without negative coarse mesh cross sections, the scalar flux solution on the coarse mesh does not become negative.

In the case of divergence as a result of nonlinearity, the cause is likely to be a combination of the following three contributors: (i) the use of an approximation that allows for a non-positive solution (in this case, the use of the TCP$_0$ scattering treatment), (ii) the use of a nonlinear method, such as CMFD, that is susceptible to nonlinear instability, and (iii)

Table 7.6: GE-12 Data w/ CMFD & TS/OI = 2

Outer Iteration #	Eigenvalue Res.	Fission Source Res.	# Neg. FF	# Neg. CF
1	2.005E-01	2.599E-01	0	0
2	6.610E-02	4.178E-02	0	0
3	6.858E-02	1.448E-02	320	0
4	1.887E-02	3.512E-03	800	0
5	4.340E-03	1.084E-03	800	0
6	1.508E-03	4.544E-04	780	0
7	3.324E-04	1.225E-04	792	0
8	1.348E-03	1.058E-03	896	8
9	7.664E-03	3.361E-03	1588	64
10	1.080E-02	4.888E-03	1136	56
11	7.876E-04	3.711E-03	2057	24
12	2.146E-03	4.128E-03	592	12
13	5.958E-03	5.793E-03	866	12
14	4.118E-03	2.360E-03	799	6
15	1.353E-03	3.776E-03	706	17
16	3.122E-03	4.608E-03	1049	38
17	2.976E-03	6.949E-03	876	41
18	4.169E-04	6.983E-03	700	28
19	2.030E-02	4.634E-02	1311	36
20	2.020E-02	1.425E-01	10453	230
21	8.911E-03	1.239E-01	47840	951
22	7.574E-05	7.308E-02	8523	194



(a) Eigenpair residuals vs. outer iteration number

(b) Peak value of $\hat{D}$ on the coarse grid vs. outer iteration number

Figure 7.9: CMFD data for the divergent GE-12 case

a specific setup that results in the problem entering a divergent state. Regarding the latter

Table 7.7: GE-12 Data w/ LDA & TS/OI = 2

Outer Iteration #	Eigenvalue Res.	Fission Source Res.	# Neg. FF	# Neg. CF
1	2.005E-01	2.600E-01	0	0
2	1.374E-02	2.043E-02	64	0
3	7.224E-02	1.668E-02	480	0
4	1.559E-02	5.427E-03	800	0
5	8.070E-03	3.325E-03	840	0
6	3.723E-03	2.040E-03	884	0
7	2.106E-03	1.300E-03	924	0
8	1.234E-03	8.590E-04	964	0
9	7.666E-04	5.618E-04	1000	0
10	4.845E-04	3.739E-04	1032	0
11	3.112E-04	2.476E-04	1064	0
12	1.998E-04	1.650E-04	1096	0
13	1.280E-04	1.099E-04	1100	0
14	8.052E-05	7.334E-05	1108	0
15	4.867E-05	4.900E-05	1120	0

point, we refer to the specific options used for this case. In particular, the specification of 2 transport sweeps per outer iteration as opposed to 1. When 2 transport sweeps are performed, the code passes through an unrecoverable state when accelerated with CMFD. This unrecoverable state is avoided when this option is changed, underscoring the unpredictability of nonlinear instability. The use of LDA allows the problem to pass through this otherwise unrecoverable state because it is tolerant of a non-positive solution.

To determine the difference between the simulations for each case, we investigate the difference in the convergence of the inscatter source when TS/OI = 1 or 2. Since source iteration is performed in each transport sweep, and the space- and energy-dependent scattering source is updated each transport sweep, the inscatter source is expected to converge faster when TS/OI is increased. The total magnitude of the inscatter source (with units of neutrons/cm^3·s) is graphed as a function of outer iteration number in Fig. 7.10 for 1 and 2 TS/OI. For these results, LDA is used as the acceleration method. The total inscatter source for a given outer iteration is defined as

$$\text{Total Inscatter Source} \equiv \sum_j \sum_g \sum_{g',\, g' \neq g} \Sigma_{s,j,g' \to g} \phi_{0,j,g'}^{(l)} \, , \tag{7.7}$$

where l is the outer iteration index, j is the transport grid index, and g is the group index. Though there is only a slight difference in the convergence rate of the inscatter source between the two options, we observe that the scattering source possesses slightly smaller values when TS/OI = 2. These smaller values likely lead to smaller scalar flux iterates across outer iterations, which then allow for negative coarse-grid cross sections. As a result of this difference, the nonlinearity of CMFD is encountered and the code becomes divergent. Specifically, at outer iteration 8, which is where the code enters an unrecoverable state when using CMFD and 2 TS/OI, the inscatter source magnitude is slightly smaller than the convergent case where TS/OI = 1. This is enough of a perturbation relative to the case where TS/OI = 1 to result in divergence with CMFD.

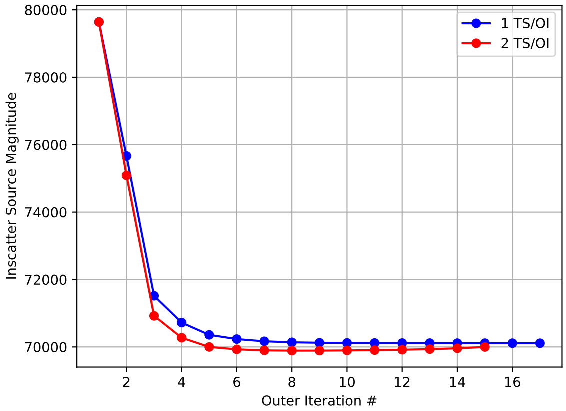


Figure 7.10: Inscatter source magnitude vs. outer iteration for 1 and 2 TS/OI with LDA

To ensure that the converged solutions with LDA for both TS/OI options are correct, we have provided comparisons to the CMFD solution with TS/OI = 1 in Table 7.8. Because the TS/OI option should not affect the converged solution, we expect differences that are on the order of the convergence criterion. The pin power difference is defined as

$$\text{Pin Power Difference} \equiv \sqrt{\frac{1}{N_k}\sum_k \left(p_{k,\text{LDA}} - p_{k,\text{CMFD}}\right)^2}\,, \tag{7.8}$$

where p_k is the pin power in a given pin k and N_k is the number of pins. This metric was chosen based on the convergence criterion of MPACT. The eigenvalue difference is given in percent mille (pcm) and is defined as

$$\text{pcm difference} \equiv \frac{|k_{\text{eff,LDA}} - k_{\text{eff,CMFD}}|}{k_{\text{eff,CMFD}}} \cdot 10^5\,. \tag{7.9}$$

We expect the LDA solution for both of these options to be the close to the CMFD solution with TS/OI = 1. From the data in Table 7.8, we observe that the LDA pin power and pcm differences are slightly larger than but close to the convergence criterion of $5 \cdot 10^{-5}$. Based on these small differences, we conclude that LDA converges to the same solution as that obtained by CMFD when CMFD is convergent.

Table 7.8: Comparison of LDA and CMFD Solutions for GE-12

Case	Pin Power Difference	pcm difference
LDA w/ TS/OI = 1	22E-05	9
LDA w/ TS/OI = 2	30E-05	18

7.5 Summary and Discussion

In this chapter, we investigate the use of LDA in MPACT. The optimal shift fraction is determined to be approximately 0.98 in Section 7.2 for a large practical case, but we note that this value may change for other problems. For larger shift fractions, the code performance suffers due to issues related to the convergence of the linear solver. In Section 7.3, the performance of LDA is shown to be slightly worse than CMFD. This is due to the large computational effort of solving the adjoint diffusion calculation and the lack of diffusion coefficient tuning for LDA compared to CMFD. Though the runtime for LDA is significantly larger than CMFD, the convergence properties are shown to be similar. Finally, we present a case study in which the nonlinear instability of CMFD (likely driven by the TCP_0 scattering approximation) results in divergence. When the same case is accelerated with LDA, the code is shown to accurately converge. *These results support the goals for the development of LDA as established in Chapter 1.*

Section 7.4 presents the results of a nonlinear instability study of CMFD in MPACT related to the TCP_0 scattering approximation. However, we also discussed the possibility of CMFD failure when used with the 2D/1D method in Section 2.3.2.4. Use of the 2D/1D method without splitting can drive the solution negative due to high transverse leakage in 3D problems. In such cases, the possibility of CMFD failure exists due to nonlinear instability.

In [42], a number of improvements were made to the axial solver in MPACT. Among these were the inclusion of anisotropic angular dependence for the transverse leakage terms and the homogenized cross sections used to couple the 1D and 2D equations. Normally, the angular dependence of these terms is assumed to isotropic, but this can reduce the accuracy of the solution for problems with large axial gradients. In [42], the splitting method was often necessary to allow MPACT to converge for problems in which these improvements to the axial solver were enabled. (When anisotropic transverse leakage is enabled, a negative solution is much more likely.) However, the splitting method contains assumptions that can also degrade the solution accuracy. When the improvements to the angular coupling of the 2D and 1D solutions were enabled, the use of splitting overshadowed these improvements (in some cases, to a certain degree). Though the exact cause of the convergence issues seen when splitting was disabled was unclear, we originally hoped that the use of LDA in place of CMFD in these cases would resolve the iteration issues. That is, if the nonlinear instability CMFD was the cause of these issues, then replacing it with LDA would potentially allow for the code to converge for cases with (i) axial solver improvements enabled and (ii) splitting disabled. In this thesis work, such cases were tested with LDA.

Unfortunately, when LDA was used in place of CMFD for the aforementioned cases, MPACT still exhibited convergence issues. The cause of these issues *may* be the nodal solver in MPACT. This reasoning comes from the following observations:

1. LDA has been shown to lack the nonlinear instability of CMFD (in Chapter 5, Chapter 6, and this chapter), so it is almost certainly not the cause of the convergence issues, and
2. upon investigation of the convergence issues for the aforementioned 3D cases, the solution produced by the nodal solver was seen to become unphysical immediately upon the appearance of negative flux values on the coarse mesh.

Therefore, we are unable to provide evidence that LDA alleviates the convergence issues seen in [42] (and, subsequently, ascertain the solution accuracy improvements due to the work in [42] without splitting enabled). However, we have shown in this thesis that the use of LDA (without other methods that are potentially susceptible to nonlinear instability) in place of CMFD allows for convergence of cases that, otherwise, cannot be accelerated with CMFD. We suggest the investigation and improvement of the stability of the nodal solver in MPACT (or any other source of nonlinear instability in the code) as future work.

CHAPTER 8

Conclusions

This chapter provides a brief summary of the work presented in this thesis. We review the initial motivation for the development of LDA in Section 8.1. Then, the principal findings are reported in Section 8.2. Lastly, suggested directions for future work are given in Section 8.3.

8.1 Motivation

CMFD is a diffusion-based transport acceleration method that has been shown to dramatically improve the convergence rate of SI for both fixed-source and eigenvalue problems. By utilizing the solution to the low-order CMFD problem, the computational work required to obtain a converged solution to the NTE can be substantially reduced (provided the optical thickness of the coarse mesh stays relatively small). We refer to the degradation in the ability of CMFD to accelerate SI with increasing coarse cell optical thickness as *linear instability*. Our focus is not on the linear instability of CMFD, but rather, the *nonlinear instability*. This type of instability refers to the susceptibility of CMFD to exhibit reduced performance, and even divergence, when certain nonlinear terms become unphysically large under certain conditions. These nonlinear terms contain division by quantities that may become very small during the iterative process, resulting in deviation from the expected behavior of the method. In practice, nonlinear instabilities are encountered seemingly randomly. LDA, which is the focus of this thesis, is a novel diffusion-based transport acceleration method that contains no local nonlinearities with respect to the scalar flux, and was specifically developed as a potential replacement for CMFD. The primary goal for the development of LDA was to construct a method with no local nonlinearities that can rival the performance of the widely-used CMFD acceleration method.

The nonlinear instability that CMFD exhibits is primarily caused by the possibility of non-positive flux iterates that may arise during the process of computing a converged solution. Certain approximations that are commonly applied to the NTE allow for non-positive iterates or solutions, including the TCP$_0$ scattering treatment and the 2D/1D approximation. Both of these approximations are present in the neutronics code MPACT, with the 2D/1D method being the basis of computing solutions for 3D models. Further, some spatial discretization choices for the NTE allow for the possibility of non-positive flux solutions. The presence of non-positive scalar flux quantities on either the transport or low-order mesh (for any iterate) can cause terms that are normally small, such as the transport consistency term $\hat{D}$, to become unphysically large. Other nonlinearities of CMFD that can cause issues are the flux- and volume-weighted macroscopic cross sections and the multiplicative prolongation factor.

The LDA is a novel diffusion-based transport acceleration method that was developed with the nonlinear instabilities of CMFD in mind. By avoiding the nonlinearities that are present in the CMFD formulation, LDA is not susceptible to performance degradation or divergence under the same conditions where CMFD exhibits nonlinear instability. Therefore, LDA can be used as a replacement for CMFD in situations where non-positive flux iterates or solutions can occur. However, because LDA possesses similar linear instability properties as CMFD, it can also be used generally. Certain properties of the LDA linear system of equations can also offer additional advantages compared to CMFD, although some disadvantages of LDA exist at the current state of the work.

8.2 Summary

LDA is inspired by a previously-proposed acceleration method, which we refer to as the GAML method [12]. The foundation of both methods is the *Fredholm Alternative Theorem* (FAT), which allows one to compute the solution to a linear system with a singular operator. The LDA linear system can be thought of as a critical diffusion system with linear transport consistency terms – that are analogous to $\hat{D}$ – in the source vector. By applying the FAT, the transport eigenvalue can be computed in a way that allows for a unique scalar flux solution to the low-order LDA system to exist. This requires the computation of the adjoint diffusion solution (and, optionally, the forward diffusion solution), which amount to additional computational work exclusive to LDA.

Because the LHS operator of the LDA system is the critical diffusion operator, it does not change during the iteration process. This is different from CMFD, which contains transport consistency terms in the LHS operator. Therefore, the CMFD operator must be reconstructed every outer iteration in order for the low-order solution to maintain consistency with the transport solution. The fixed nature of the LDA operator offers the following advantages:

1. the constant nature of the LDA operator may be utilized to reduce the computational work of solving the low-order linear system, either directly or iteratively, and
2. if the low-order eigenvalue problem is Wielandt-shifted (to improve the convergence rate of the PI process), then the upper bound of the shift is known (as opposed to CMFD, where one must conservatively choose the magnitude of the shift).

In addition to the derivation of CMFD, including the rise of nonlinear terms, Chapter 3 provides the rationale for the need to choose a conservative shift magnitude for the CMFD system. The detailed derivation of LDA, including the Wielandt-shifted variation, are provided in Chapter 4.

Because we intend for LDA to serve as a replacement acceleration method for CMFD, we are interested in the relative convergence properties of the two methods. Chapter 5 provides the details and results of a Fourier analysis that was performed for both acceleration methods. This analysis produced a set of deterministic equations that could be used to predict the convergence rate, in the form of the spectral radius ρ, for either LDA or CMFD in conceptually-infinite, spatially-heterogeneous, planar systems. The results of the analysis show that LDA and CMFD possess the same ρ for infinite, homogeneous systems. This was confirmed through numerical experiments using a 1D research code. However, the spectral radii of the two methods differ in the heterogeneous case. Studies in which various properties of the planar system are varied show that the difference in the spectral radius for LDA compared to CMFD is significant only in cases with very high spatial heterogeneity. Additionally, these differences become trivial for eigenvalue problems, which is the primary intended application for LDA. We also analytically demonstrate the nonlinear instability of CMFD, and the lack thereof for LDA. Overall, the Fourier analysis predictions are shown to be highly accurate compared to the numerical results from the research code.

In Chapters 6 and 7, the performances of LDA and CMFD are compared for finite problems that cannot be analyzed using the Fourier analysis. Chapter 6 compares them in a 1D research code, and Chapter 7 compares them for practical problems in MPACT. LDA is shown to be more susceptible than CMFD to linear instability, with the spectral radius of LDA increasing faster than CMFD as coarse cell optical thickness increases. However, the performance difference is small for moderate optical thickness values. In addition, the nonlinear instability of CMFD is demonstrated for contrived cases in Chapter 6 and a practical case in Chapter 7. LDA is shown to remain stable when used to accelerate these same cases. In Chapter 7, we also show that the additional computational work of computing the initial diffusion solution when using LDA becomes non-trivial in large, practical problems simulated with MPACT. We note that the efficient determination of the coarse-grid diffusion eigenfunction is an implementation issue in MPACT that should be considered to be independent of the LDA method. Efficient computational methods for solving such discrete diffusion problems exist and, in principle, should be used.

We also explore the behavior of LDA for Wielandt-shifted problems in Chapters 6 and 7. For small, 1D cases, the convergence rate of PI is shown to decrease monotonically with an increasing shift fraction for values that are not near unity. However, this behavior is not observed in practical MPACT cases, where the performance of the iterative linear solver suffers from the increasing condition number of the low-order operator with increasing shift fraction. In these cases the shift fraction must be large enough to outweigh this drawback. However, one must take care not to choose a shift fraction that is too close to unity. We show that the linear solver (either direct or iterative) suffers from roundoff error in these cases, resulting in inaccurate low-order solutions and degradation in the performance of the acceleration method. One can estimate a reasonable shift fraction that avoids these issues for practical problems with similar spatial refinement. In Chapter 7, we determine the optimal shift fraction for a large, practical problem. The performance of LDA with this shift fraction is shown to be close to, but slightly worse than, that of CMFD with a more advanced adaptive shift method.

Overall, these results support the goals for the development of the LDA method as established in Chapter 1. Namely, we have shown that LDA lacks the nonlinear instabilities of CMFD for both contrived and practical problems. Further, we have shown that the performance of LDA is comparable to CMFD, albeit under certain conditions. However, the additional computational work of obtaining the initial adjoint diffusion solution in MPACT has proven to be problematic for the use of LDA to accelerate large cases. We note that this may be an implementation issue in MPACT and is not inherent to LDA. The convergence rate of LDA is shown to be poor under extreme circumstances, but is more manageable in the practical context.

8.3 Future Work

As future work, we suggest the implementation of linear solvers that take advantage of the properties of LDA in a practical code such as MPACT. These solvers should utilize the fixed nature of the LDA LHS operator. For example, if the operator is LU decomposed before the outer iteration process begins, then the linear system can be efficiently solved using forward and backward substitution. Such an implementation would need to scale well in parallel, and take measures to mitigate the additional memory requirements of this approach. In Section 6.5, we provide some examples of past and present work that may provide promising research avenues on this topic. The implementation of advanced solver methods in MPACT may also allow for improved performance when obtaining the initial diffusion solution. Because this calculation amounts to significant additional computational work in MPACT, this work would improve the practical use of LDA in MPACT.

In this work, the computational expense of solving the low-order diffusion adjoint problem has proven to be non-trivial. Previous work has focused on reducing this expense through *multilevel* methods, in which the solution of lower-order systems with coarser spatial and/or energy grids are employed [78]. The *Multilevel-in-Space-and-Energy Diffusion* (MSED) method makes use of a "grey" diffusion equation, which is self-adjoint. By leveraging the solutions to these coarsened problems, the overall runtime of obtaining the CMFD solution is reduced. We suggest the extension of this technology to LDA, so that problems accelerated with LDA can also benefit from reduced computational expense. Additionally, the use of multilevel methods for the initial adjoint diffusion problem (which must be fully converged, and as such, adds non-trivial computational expense to problems accelerated with LDA) should also be investigated.

We further suggest that the procedure taken to obtain the "optimally-diffusive" adjustment to the diffusion coefficient be applied to spatially-heterogeneous problems. This study was originally performed for spatially-homogeneous problems, resulting in an adjustment that depends only on the optical thickness of the coarse cell. However, we have shown that high spatial heterogeneity can also affect the convergence rate of an acceleration method. Additionally, heterogeneity affects LDA and CMFD differently. One may be able to determine a method-dependent adjustment that takes into account both the optical thickness and the degree of heterogeneity. The value of this adjustment would likely differ between LDA and CMFD. This may take the form of (i) an adjustment to the numerical diffusion coefficient of a given coarse cell or (ii) more fine-grained adjustments to the diffusion coefficients of the fine cells in a coarse cell (ultimately resulting in a change in the numerical diffusion coefficient of the coarse cell). This work may help to mitigate the linear instability of LDA and CMFD for spatially-heterogeneous problems.

Further, the use of higher-order prolongation has been shown to improve the convergence rate of CMFD [66, 81]. In this method, information from neighboring coarse mesh fluxes is incorporated into the prolongation operator for the low-order problem. This method has a similar effect on the convergence rate as the aforementioned adjustments to the diffusion coefficient. The use of this method for LDA may also be able to improve the convergence rate of accelerated problems.

We provide a discussion of the choice of weighting function for the LDA operators in

Section 4.6. This weighting function, which is used to compute coarse-grid quantities for the low-order problem, is chosen to be uniform in space. This choice was made such that the diffusion operator that is present in the LDA equation is fixed throughout the iteration process. If the weighting function is not fixed, then the diffusion eigenpair would need to be recomputed each time the weighting function is updated. However, the choice of the uniform weighting function likely has a detrimental affect on the convergence properties of LDA. This is because there are additional lagged terms in the LDA formulation that are not present in CMFD (which uses flux- and volume weighting) to account for the incorrect weighting of homogenized quantities. The use of a "better" weighting function may improve the convergence properties of LDA by reducing the magnitude of the lagged terms. For example, one might use the scalar flux solution of the first transport sweep as the weighting function. In this case, the diffusion solution would need to be recomputed once, but subsequent iterations may see a benefit. (To prevent the possibility of nonlinear instability, the weighting function should always be strictly positive.)

In addition to the choice of weighting function, we also consider the boundary conditions for the LDA system. The boundary conditions used for LDA are those of the corresponding diffusion problem, which are contained in the diffusion operators. Upon convergence, these operators cancel out – leaving the neutron balance equation (as shown in Chapter 4). This choice for the boundary conditions may not be optimal in terms of the spectral radius of the method. The use of different boundary conditions that incorporate more information from the transport problem may improve the convergence properties of LDA. We suggest the investigation of alternate boundary conditions for the low-order LDA system as future work.

As discussed in Section 7.5, we were unable to show that LDA alleviates the convergence issues shown in [42] for cases in which improvements to the MPACT axial solver were enabled. These convergence issues, which were thought to be potentially brought about by the nonlinearities in CMFD, are still present when problems are accelerated with LDA. A potential source of these issues is the nodal solver in MPACT, which is observed to produce an unphysical solution once negative coarse mesh fluxes are present. We suggest the investigation of these issues as future work. If these issues are resolved, then the improvements in [42] may be able to be better observed.

Lastly, we consider multi-physics neutronics problems with nonlinear feedback. These are problems in which "feedback" in the form of material state information (such as temperature) from a separate solver affects the values of the cross sections in the neutronics calculation. In turn, information from the neutronics solver (such as the power distribution) provides the heat source for the temperatures calculated by the feedback solver. For this “coupled” calculation, the normally well-behaved convergence properties of CMFD can degrade compared to problems without feedback. Previous work has shown that incorporating some aspects of nonlinear feedback in the low-order problem can improve the convergence rate of multi-physics problems [82, 83]. For example, if the value of the cross sections is assumed to be dependent on the scalar flux (which serves as a proxy for the more detailed physics) then the low-order solver can be modified to incorporate this information. The result is improved convergence of CMFD-accelerated multi-physics problems. We suggest the extension of this idea to LDA, so that it can be used to better accelerate problems with feedback (while avoiding nonlinear instability).

APPENDIX A

Comparison of GAML and LDA

As previously discussed, the LDA method has some similarities to the GAML method (so named based on the last names of the authors of the paper) [12]. Here, we reproduce the GAML paper for reference in Appendix A.1 (because the original paper is not easily available), and then we provide a short discussion on the differences between GAML and LDA in Appendix A.2.

A.1 Reproduction of the GAML Paper

ACCELERATION OF TRANSPORT EIGENVALUE PROBLEMS

E. M. Gelbard
C. H. Adams
Applied Physics Division
Argonne National Laboratory
Argonne, Illinois 60439

and

D. R. McCoy
E. W. Larsen
Theoretical Division
University of California
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

SUMMARY

A general linear procedure has recently been developed for the acceleration of iteration methods for fixed-source discrete-ordinates transport problems in slab geometry [34, 84]. This procedure leads to unconditionally stable diffusion-synthetic acceleration methods for many spatial discretization schemes applicable to the discrete-ordinates equations. However, the procedure is not directly applicable to the acceleration of iteration schemes for eigenvalue problems. Alcouffe's diffusion-correction method [21], encoded in ONEDANT [60], has been developed for the acceleration of diamond-differenced eigenvalue problems, but this method is nonlinear and must be used with a negative flux fixup – even if negative fluxes occur in groups that have only slight influence on the eigenvalue – and the method is not easily generalizable to other differencing schemes. Here we derive an acceleration method that is based on the general fixed-source procedure mentioned above, and that does not have these two drawbacks.

We wish to solve the eigenvalue problem

$$\mu \frac{\mathrm{d}\psi}{\mathrm{d}x}(x,\mu) + \sigma_T \psi(x,\mu) = \left(\sigma_S + \lambda_H \nu \sigma_f\right) \phi_0(x) \quad , \tag{A.1}$$

where Eq. (A.1) holds in some system with suitable boundary conditions, and we define

$$\phi_n(x) = \frac{1}{2} \int_{-1}^{1} P_n(\mu) \psi(x,\mu) \, \mathrm{d}\mu \quad , \quad n = 0, 1, 2,$$

with P_n the n-th Legendre polynomial. Taking the zero-th and first angular moments of Eq. (A.1) and eliminating ϕ_1 gives

$$H_L \phi_0 = \left(\lambda_H - \lambda_L\right) \nu \sigma_f \phi_0 + \frac{\mathrm{d}}{\mathrm{d}x} \frac{2}{3\sigma_T} \frac{\mathrm{d}}{\mathrm{d}x} \phi_2 \quad , \tag{A.2}$$

where H_L is the operator

$$H_L \equiv -\frac{\mathrm{d}}{\mathrm{d}x} \frac{1}{3\sigma_T} \frac{\mathrm{d}}{\mathrm{d}x} + \sigma_T - \sigma_S - \lambda_L \nu \sigma_f$$

and λ_L is the smallest positive number such that

$$H_L \phi_L = 0 \quad .$$

Multiplying Eq. (A.2) by the adjoint eigenfunction ϕ_L^* and operating by the inner product $\langle \cdot, \cdot \rangle$ gives the condition on λ_H

$$0 = \left(\lambda_H - \lambda_L\right) \left\langle \phi_L^*, \nu \sigma_f \phi_0 \right\rangle + \left\langle \phi_L^*, \frac{\mathrm{d}}{\mathrm{d}x} \frac{2}{3\sigma_T} \frac{\mathrm{d}}{\mathrm{d}x} \phi_2 \right\rangle \quad , \tag{A.3}$$

which must be satisfied for a solution of Eq. (A.2) to exist. Now, Eqs. (A.1) to (A.3) suggest the iteration scheme

$$\mu \frac{\mathrm{d}\psi^{(l+\frac{1}{2})}}{\mathrm{d}x} + \sigma_T \psi^{(l+\frac{1}{2})} = \left(\sigma_S + \lambda_H^{(l)} \nu\sigma_f\right) \phi_0^{(l)} \quad , \tag{A.4}$$

$$\lambda_H^{(l+1)} = \lambda_L + \frac{\left\langle \phi_L^*, \frac{\mathrm{d}}{\mathrm{d}x}\frac{2}{3\sigma_T}\frac{\mathrm{d}}{\mathrm{d}x}\phi_2^{(l)} \right\rangle}{\alpha \left\langle \phi_L^*, \nu\sigma_f \phi_0^{(l+\frac{1}{2})} \right\rangle + 1 - \alpha} \quad , \tag{A.5}$$

$$H_L \phi_0^{(l+1)} = \left(\lambda_H^{(l+1)} - \lambda_L\right) \nu\sigma_f \left[\alpha \phi_0^{(l+\frac{1}{2})} + (1-\alpha)\,\phi_0^{(l)}\right] + \frac{\mathrm{d}}{\mathrm{d}x}\frac{2}{3\sigma_T}\frac{\mathrm{d}}{\mathrm{d}x}\phi_2^{(l+\frac{1}{2})} \quad , \tag{A.6}$$

$$\left\langle \phi_L^*, \nu\sigma_f \phi_0^{(l+1)} \right\rangle = 1 \quad . \tag{A.7}$$

Equation (A.5) is a necessary and sufficient condition for Eq. (A.6) to have a solution, and Eq. (A.7) is a normalization which makes the solution of Eq. (A.6) unique. [This normalization is used in the denominator of Eq. (A.5).] The parameter α in Eqs. (A.5) and (A.6) is arbitrary and will be discussed below.

This method can be rewritten in a more useful form as follows. Taking the zero-th and first moments of Eq. (A.4), eliminating $\phi_1^{(l+\frac{1}{2})}$, and subtracting the resulting equation from Eq. (A.6) gives

$$\begin{aligned} H_L\left(\phi_0^{(l+1)} - \phi_0^{(l+\frac{1}{2})}\right) &= \sigma_S\left(\phi_0^{(l+\frac{1}{2})} - \phi_0^{(l)}\right) \\ &+\nu\sigma_f \left\{\left[\lambda_H^{(l+1)}\phi_0^{(l+\frac{1}{2})} - \lambda_H^{(l)}\phi_0^{(l)}\right]\right. \\ &\left. + (1-\alpha)\left[\lambda_L\left(\phi_0^{(l+\frac{1}{2})} - \phi_0^{(l)}\right) + \left(\lambda_H^{(l+1)} - \lambda_H^{(l)}\right)\phi_0^{(l)}\right]\right\} \quad . \end{aligned} \tag{A.8}$$

An alternative expression [but equivalent to Eq. (A.6)] for $\lambda_H^{(l+1)}$, found by operating on Eq. (A.8) by $\langle \phi_L^*, \cdot \rangle$, is

$$\lambda_H^{(l+1)} = \lambda_L - \frac{\left\langle \phi_L^*, \sigma_S\left(\phi_0^{(l+\frac{1}{2})} - \phi_0^{(l)}\right) + \nu\sigma_f\left(\lambda_L \phi_0^{(l+\frac{1}{2})} - \lambda_H^{(l)}\phi_0^{(l)}\right)\right\rangle}{\alpha\left\langle \phi_L^*, \nu\sigma_f\phi_0^{(l+\frac{1}{2})}\right\rangle + 1 - \alpha} \quad . \tag{A.9}$$

The alternative (but equivalent) scheme consists of replacing Eq. (A.5) by Eq. (A.9) and Eq. (A.6) by Eq. (A.8). Thus, the l-th iteration consists of determining $\psi^{(l+\frac{1}{2})}$ from Eq. (A.4), $\lambda_H^{(l+1)}$ from Eq. (A.9), and $\phi_0^{(l+1)}$ from Eq. (A.8) and Eq. (A.7). This process repeats until convergence.

Numerical schemes are derived by repeating the above analysis basically line-for-line, but starting with the discretized (rather than the continuous) form of the transport equation. This procedure directly follows that which has been developed for fixed-source problems [34, 84]. From Eq. (A.6), it is apparent that the choice $\alpha = 1$ introduces the most recent information into the equation for $\phi_0^{(l+1)}$, whereas the choice $\alpha = 0$ equally apparently simplifies the equation for $\lambda_H^{(l+1)}$. In practice, we have found that for the diamond-differenced transport equation, either choice of α leads to virtually the same results. The form of Eq. (A.9) with $\alpha = 0$ has been used to obtain eigenvalue estimates (i.e., Eq. (A.9) was evaluated with $l = 0$ which gives a first-order perturbation theory estimate). These eigenvalue estimates were then used to assess the error in the low order $\mathrm{S_N}$ eigenvalue computations [85].

In comparing our diamond-differenced acceleration method with that in ONEDANT for one-group slab-geometry eigenvalue problems, we find that the two methods require essentially an equal amount of computational effort for problems in which both methods converge. Our method does not require the transport solution to be everywhere positive and we do not employ a negative-flux fixup; probably because of this, we have never observed our method to diverge. The ONEDANT method however does require a negative flux fixup and for coarse meshes this fixup can either degrade the results or cause divergence. As a very simple example, let us consider a three-region system: the left-boundary is reflecting; the leftmost region is 2 cm thick with $\sigma_T = 1.0$, $\sigma_a = 0.1$, and $\nu\sigma_f = 0.0$; (all cross sections have units of cm^{-1}); the center region is 20 cm thick with $\sigma_T = 1.0$, $\sigma_a = 0.1$, and $\nu\sigma_f = 0.1$; the rightmost region is 20 cm thick with $\sigma_T = 2.0$, $\sigma_a = 0.02$, and $\nu\sigma_f = 0.0$; and the right boundary is vacuum. The $\mathrm{S_8}$ eigenvalues $k = \lambda^{-1}$ are tabulated in Table A.1 for the two methods and for four separate meshes. (The numbers of equal cells in each region are given in parentheses.) The finest mesh solutions are seen to agree. For the next two meshes the solutions do not agree due to ONEDANT's use of a negative flux fixup; although the differences are slight, our results are closer to that of the fine mesh solution. Finally, for the coarsest mesh, the ONEDANT algorithm diverges.

In summary, we have developed an acceleration method for transport eigenvalue problems that is stable with respect to mesh size. Also, our method is applicable in multidimensional geometry and to non-diamond differencing schemes, subjects that we are currently investigating.

Mesh	Linear	Nonlinear
(4,40,80)	0.952901	0.952901
(3,30,60)	0.952881	0.952880
(2,20,40)	0.952833	0.952822
(1,10,20)	0.952827	*

Table A.1: Eigenvalues for the Linear and ONEDANT (nonlinear) acceleration methods. (* denotes divergence.)

A.2 GAML vs. LDA

The GAML and LDA methods possess some similarities in their formulations, but also some distinct differences. These differences come about through the practical consideration of LDA, which is derived with the assumption that the method will be used with a coarser spatial mesh than the transport problem. Further, the GAML equations are set up such that the correction to the most recent scalar flux (rather than the next scalar flux iterate) is computed, which is also done in DSA [21]. Additionally, we have discussed practical considerations of the use of LDA for large, complex problems in this thesis.

GAML uses a critical diffusion operator (H_L) for the low-order equation. To account for the differences between the diffusion operator and the transport problem, correction terms are present in the RHS of Eq. (A.6). In LDA, the low-order problem is set up differently, with the diffusion operator $(\mathcal{L}_{d,eig})$ being present on both sides of the low-order equation, as reproduced below:

$$\begin{aligned}\left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0\right)_k &= \left(\Phi_{1,k-\frac{1}{2}} - \Phi_{1,k+\frac{1}{2}}\right) \\ &+ \left(\mathcal{L}_{d,eig}\left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \mathbf{\Phi}_0\right)_k + \left(\lambda_t (\nu\dot{\Sigma}_f)_k - \dot{\Sigma}_{a,k}\right)\left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \ .\end{aligned} \tag{Eq. (4.52a) revisited}$$

Upon convergence, these operators cancel out, leaving the NBE. In GAML, the equations are written in such a way that the diffusion operator is present only on the LHS. Also, the correction to the most recent scalar flux estimate $\left(\phi_0^{(l+1)} - \phi_0^{(l+\frac{1}{2})}\right)$ is computed in GAML (using Eq. (A.8)), whereas the next scalar flux iterate $\left(\phi_0^{(l+1)}\right)$ is the solution of the LDA equation.

The GAML equations are presented in continuous form, with no information on how to obtain the discrete form of the low-order equation. (Numerical results are presented for the case in which there is only one spatial grid – that is, the transport and low-order problems use the same spatial grid.) The LDA equation is derived from the discrete NBE, and considers the coarse-mesh case in which the RHS correction factors take a specific form to account for volume-weighting of coarse-mesh quantities in the LHS diffusion operator. These correction factors must be carefully handled to avoid the appearance of nonlinearities in the LDA method. Specifically, the correction factors for the cross sections take the following form:

$$\dot{\Sigma}_{u,k} \left\langle\!\left\langle \boldsymbol{\phi}_0 \right\rangle\!\right\rangle \equiv \sum_{j \in k} \Sigma_{u,j} \phi_{0,j} h_j \ , \ \forall \ u \in \{a, s, t\} \ , \tag{A.10}$$

which are a function of the transport flux on the fine grid (ϕ_0). LDA also considers a linear prolongation operator to avoid a nonlinear prolongation update. Additionally, the transport eigenvalue calculation for LDA takes a different form than for GAML due to the consideration of the coarse mesh, as reproduced below:

$$\lambda_t^{(l+1)} = \frac{\sum\limits_{k=1}^{K} \Phi_{0,d,k}^{*} \left[\left(\Phi_{1,k+\frac{1}{2}}^{(l+\frac{1}{2})} - \Phi_{1,k-\frac{1}{2}}^{(l+\frac{1}{2})} \right) + \dot{\Sigma}_{a,k} \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle - \left(\mathcal{L}_{d,eig} \left\langle\!\left\langle \mathbf{1} \right\rangle\!\right\rangle \Phi_0^{(l+\frac{1}{2})} \right)_k \right]}{\sum\limits_{k=1}^{K} \Phi_{0,d,k}^{*} (\nu \dot{\Sigma}_f)_k \left\langle\!\!\left\langle \boldsymbol{\phi}_0^{(l+\frac{1}{2})} \right\rangle\!\!\right\rangle} \ .$$

(Eq. (4.59) revisited)

Both methods utilize a critical (singular) LHS diffusion operator, and the FAT is used to obtain a formulation for the transport eigenvalue. In this work, we consider (i) how to modify the low-order LDA problem so that it can be solved practically (by moving some fraction of the fission source to the RHS) as discussed in Chapter 4, (ii) the implications of the fixed LHS LDA operator (as discussed in Section 6.5), and (iii) the linear instability properties of LDA. Limited numerical results are presented in the GAML paper, with no information regarding the linear instability characteristics of the method.

In summary, the GAML and LDA methods have a similar goal. Namely, the avoidance of nonlinearities so that negative flux fix-ups do not need to be used in the neutronics calculation (which can degrade the solution accuracy). To achieve this goal, both methods utilize a diffusion operator and the FAT to obtain an expression for the transport eigenvalue. However, the low-order equations of each method take on different forms, with LDA taking into account the implications of a coarse spatial mesh (among other considerations).

BIBLIOGRAPHY

[1] G. C. Bilodeau, W. R. Cadwell, J. P. Dorsey, J. G. Fairey, and R. S. Varga, "PDQ: An IBM-704 Code to Solve the Two-dimensional Few-group Neutron-Diffusion Equations," tech. rep., Bettis Plant, 1957.

[2] G. I. Bell and S. Glasstone, *Nuclear Reactor Theory*. Litton Educational Publishing, Inc., 1970.

[3] B. Boyack, R. Duffey, P. Griffith, G. Lellouche, U. Rohatgi, G. Wilson, W. Wulff, and N. Zuber, "Quantifying Reactor Safety Margins: Application of Code Scaling, Applicability, and Uncertainty Evaluation Methodology to a Large-Break, Loss-of-Coolant Accident," tech. rep., U.S. Nuclear Regulatory Commission, 1989.

[4] L. L. Carter and E. D. Cashwell, *Particle Transport Simulation with the Monte Carlo Method*. Technical Information Center, 1975.

[5] J. R. Lamarsh and A. J. Baratta, *Introduction to Nuclear Engineering*. Upper Saddle River, NJ: Prentice-Hall, Inc., 3rd ed., 2001.

[6] K. F. Hansen, "Multi-group Diffusion Methods," tech. rep., M.I.T., 1962.

[7] E. W. Larsen, "Diffusion-Synthetic Acceleration Methods for Discrete-Ordinates Problems," *Transport Theory and Statistical Physics*, vol. 13, no. 1-2, pp. 107–126, 1984.

[8] K. Smith, "Nodal Method Storage Reduction by Nonlinear Iteration," *Transactions of the American Nuclear Society*, vol. 44, pp. 265–266, 1983.

[9] K. Smith and J. Rhodes, "Full-Core, 2-D, LWR Core Calculations with CASMO-4E," in *PHYSOR*, (Seoul, Korea), 2002.

[10] K. D. Lathrop, "Spatial Differencing of the Transport Equation: Positivity vs. Accuracy," *Journal of Computational Physics*, vol. 4, pp. 475–498, 1969.

[11] MPACT Team, "MPACT Theory Manual," tech. rep., Consortium for Advanced Simulation of Light Water Reactors, 2016.

[12] E. M. Gelbard, C. H. Adams, D. R. McCoy, and E. W. Larsen, "Acceleration of Transport Eigenvalue Problems," in *Transactions of the American Nuclear Society*, vol. 41, (Los Angeles,CA), pp. 309–310, 1982.

[13] Z. G. Dodson, N. Adamowicz, B. M. Kochunas, and E. W. Larsen, "SDA: A Semilinear CMFD-Like Transport Acceleration Scheme Without D-hat," in *M&C 2019 - International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering*, (Portland, OR), American Nuclear Society, 2019.

[14] D. R. Ferguson and K. L. Derstine, "Optimized Iteration Strategies and Data Management Considerations for Fast Reactor Finite Difference Diffusion Theory Codes," *Nuclear Science and Engineering*, vol. 64, no. 2, pp. 593–604, 1977.

[15] G. R. Cefus and E. W. Larsen, "Stability Analysis of Coarse-Mesh Rebalance," *Nuclear Science and Engineering*, vol. 105, pp. 31–39, 1990.

[16] T. R. Hill, "ONETRAN: A Discrete Ordinates Finite Element Code for the Solution of the One-Dimensional Multigroup Transport Equation," tech. rep., Los Alamos Scientific Laboratory, 1975.

[17] L. A. Hageman and D. M. Young, *Applied Iterative Methods*. New York, NY: Academic Press, Inc., 1st ed., 1981.

[18] M. L. Adams and E. W. Larsen, "Fast Iterative Methods for Discrete-Ordinates Particle Transport Calculations," *Progress in Nuclear Energy*, vol. 40, no. 1, pp. 3–159, 2002.

[19] H. J. Kopp, "Synthetic Method Solution of the Transport Equation," *Nuclear Science and Engineering*, vol. 17, pp. 65–74, sep 1963.

[20] E. M. Gelbard and L. A. Hageman, "The Synthetic Method as Applied to the Sn Equations," *Nuclear Science and Engineering*, vol. 37, no. 2, pp. 288–298, 1969.

[21] R. E. Alcouffe, "Diffusion Synthetic Acceleration Methods for the Diamond-Differenced Discrete-Ordinates Equations," *Nuclear Science and Engineering*, vol. 64, no. 2, pp. 344–355, 1977.

[22] V. Y. Gol'din, "A Quasi-Diffusion Method of Solving the Kinetic Equation," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 6, pp. 136–149, 1964.

[23] V. E. Troshchiev, V. F. Yudintsev, and V. I. Fedyanin, "Acceleration of the Convergence of Iterations in Solving the Kinetic Equation," *USSR Computational Mathematics and Mathematical Physics*, vol. 8, no. 2, pp. 298–308, 1968.

[24] M. J. Lee, H. G. Joo, D. Lee, and K. Smith, "Coarse Mesh Finite Difference Formulation for Accelerated Monte Carlo Eigenvalue Calculation," *Annals of Nuclear Energy*, vol. 65, pp. 101–113, 2014.

[25] A. Zhu, M. G. Jarrett, Y. Xu, B. M. Kochunas, E. W. Larsen, and T. J. Downar, "An Optimally Diffusive Coarse Mesh Finite Difference Method to Accelerate Neutron Transport Calculations," *Annals of Nuclear Energy*, vol. 95, pp. 116–124, 2016.

[26] M. G. Jarrett, B. M. Kochunas, A. Zhu, and T. J. Downar, "Analysis of Stabilization Techniques for CMFD Acceleration of Neutron Transport Problems," *Nuclear Science and Engineering*, vol. 184, no. 2, pp. 208–227, 2016.

[27] D. Wang and S. Xiao, "A Linear Prolongation Approach to Stabilizing CMFD," *Nuclear Science and Engineering*, vol. 190, no. 1, pp. 45–55, 2018.

[28] 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*. PhD thesis, University of Michigan, 2013.

[29] S. G. Stimpson, B. S. Collins, B. M. Kochunas, and T. J. Downar, "Boundary Acceleration Techniques for CMFD-Accelerated 2D-MOC," in *PHYSOR 2014 – The Role of Reactor Physics toward a Sustainable Future*, (Kyoto, Japan), 2014.

[30] M. Hursin, *Full Core, Heterogeneous, Time Dependent Neutron Transport Calculations with the 3D Code DeCART*. PhD thesis, University of California at Berkeley, 2010.

[31] B. C. Yee, E. W. Larsen, and B. M. Kochunas, "An Analytical Derivation of Transport-Corrected P0 Cross Sections and Diffusion Coefficients," in *PHYSOR 2016 – Unifying Theory and Experiments in the 21st Century*, 2016.

[32] S. Stimpson, *An Azimuthal, Fourier Moment-Based Axial SN Solver for the 2D/1D Scheme*. PhD thesis, University of Michigan, 2015.

[33] C. Zhao, Z. Liu, L. Liang, J. Chen, L. Cao, and H. Wu, "Improved Leakage Splitting Method for the 2D/1D Transport Calculation," *Progress in Nuclear Energy*, vol. 105, no. December 2017, pp. 202–210, 2018.

[34] D. R. McCoy and E. W. Larsen, "Unconditionally Stable Diffusion-Synthetic Acceleration Methods for the Slab Geometry Discrete Ordinates Equations. Part I: Theory," *Nuclear Science and Engineering*, vol. 82, no. 1, pp. 47–63, 1982.

[35] M. A. Smith, E. E. Lewis, and B.-C. Na, "Benchmark on Deterministic 3-D MOX Fuel Assembly Transport Calculations without Spatial Homogenization," *Progress in Nuclear Energy*, vol. 48, no. 5, pp. 383–393, 2006.

[36] X-5 Monte Carlo Team, "MCNP - A General Monte Carlo N-Particle Transport Code, Version 5 (Volume I: Overview and Theory)," tech. rep., Los Alamos National Laboratory, 2003.

[37] E. E. Lewis and W. Miller, Jr., *Computational Methods of Neutron Transport*. New York, NY: John Wiley & Sons, Inc., 1984.

[38] J. R. Askew, "A Characteristics Formulation of the Neutron Transport Equation in Complicated Geometries," Tech. Rep. AEEW–M1108, United Kingdom Atomic Energy Authority, 1972.

[39] E. W. Larsen and B. Kiedrowski, “NERS 543 Lecture Notes,” 2016.

[40] N. Cho, G. Lee, and C. Park, “A Fusion Technique of 2-D/1-D Methods for Three-Dimensional Whole-Core Transport Calculations,” in *Transactions of the Korean Nuclear Society*, 2002.

[41] H. G. Joo, J. Y. Cho, K. S. Kim, C. C. Lee, and S. Q. Zee, “Methods and Performance of a Three-Dimensional Whole-Core Transport Code DeCART,” *Proceedings of the PHYSOR 2004: The Physics of Fuel Cycles and Advanced Nuclear Systems - Global Developments*, pp. 21–34, 2004.

[42] M. G. Jarrett, *A 2D/1D Neutron Transport Method with Improved Angular Coupling*. PhD thesis, University of Michigan, 2018.

[43] B. W. Kelley and E. W. Larsen, “A Consistent 2D/1D Approximation to the 3D Neutron Transport Equation,” *Nuclear Engineering and Design*, vol. 295, pp. 598–614, 2015.

[44] A. Yamamoto, Y. Kitamura, and Y. Yamane, “Simplified Treatments of Anisotropic Scattering in LWR Core Calculations,” *Journal of Nuclear Science and Technology*, vol. 45, no. 3, pp. 217–229, 2008.

[45] S. Andrilli and D. Hecker, *Elementary Linear Algebra*. Elsevier Inc., 2010.

[46] A. F. van der Stappen, R. H. Bisseling, and J. G. van de Vorst, “Parallel Sparse LU Decomposition on a Mesh Network of Transputers,” *SIAM Journal on Matrix Analysis and Applications*, vol. 14, no. 3, pp. 853–879, 1993.

[47] L. Polok and P. Smrz, “Pivoting Strategy for Fast LU Decomposition of Sparse Block Matrices,” *Simulation Series*, vol. 49, no. 3, pp. 153–164, 2017.

[48] P.-G. Martinsson, *Fast Direct Solvers for Elliptic PDEs*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2019.

[49] Y. Saad and M. H. Schultz, “GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems,” *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 3, pp. 856–869, 1986.

[50] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations*. Society for Industrial and Applied Mathematics, 2007.

[51] S. Zhou, H. Wu, L. Cao, and Y. Zheng, “Unstructured Coarse Mesh Finite Difference Method to Accelerate k-eigenvalue and Fixed Source Neutron Transport Calculations,” *Annals of Nuclear Energy*, vol. 120, pp. 367–377, 2018.

[52] B. M. Kochunas, “Theoretical Convergence Rate Analysis of a Unified CMFD Formulation with Various Diffusion Coefficients,” in *M&C 2019 - International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering*, 2019.

[53] W. E. Arnoldi, “The Principle of Minimized Iterations in the Solution of the Matrix Eigenvalue Problem,” *Quarterly of Applied Mathematics*, vol. 9, no. 1, pp. 17–29, 1951.

[54] E. R. Davidson, “Super-Matrix Methods,” *Computer Physics Communications*, vol. 53, no. 1-3, pp. 49–60, 1989.

[55] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*. Society for Industrial and Applied Mathematics, 2nd ed., 2011.

[56] H. Wielandt, “Bestimmung höheren Eigenwerte durch Gebrochene Iteration,” tech. rep., Bericht der Aerodynamischen Versuchsanstalt Göttingen, 1944.

[57] E. L. Wachspress, *Iterative Solution of Elliptic Systems and Applications to the Neutron Diffusion Equations of Reactor Physics*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1966.

[58] T. J. Downar, D. A. Barber, R. Matthew Miller, C. H. Lee, T. Kozlowski, D. Lee, Y. Xu, J. Gan, H. G. Joo, J. Y. Cho, K. Lee, and A. P. Ulses, “PARCS: Purdue Advanced Reactor Core Simulator,” in *Proceedings of the PHYSOR 2002 - International Conference on the New Frontiers of Nuclear Technology : Reactor Physics, Safety and High-Performance Computing - The ANS 2002 RPD Topical Meeting*, 2002.

[59] F. Brown, “Wielandt Acceleration for MCNP5 Monte Carlo Eigenvalue Calculations,” in *Joint International Topical Meeting on Mathematics and Computations and Supercomputing in Nuclear Applications, M and C + SNA 2007*, 2007.

[60] R. D. O’Dell, F. W. Brinkley, and D. R. Marr, “User’s Manual for ONEDANT: A Code Package for One-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport,” tech. rep., Los Alamos National Laboratory, 1982.

[61] A. G. Ramm, “A Simple Proof of the Fredholm Alternative and a Characterization of the Fredholm Operators,” *American Mathematical Monthly*, vol. 108, no. 9, pp. 855–860, 2001.

[62] E. W. Larsen and B. W. Kelley, “The Relationship Between the Coarse-Mesh Finite Difference and the Coarse-Mesh Diffusion Synthetic Acceleration Methods,” *Nuclear Science and Engineering*, vol. 178, no. 1, pp. 1–15, 2014.

[63] A. Yamamoto, “Convergence Property of Response Matrix Method for Various Finite-Difference Formulations Used in the Nonlinear Acceleration Method,” *Nuclear Science and Engineering*, vol. 149, no. 3, pp. 259–269, 2005.

[64] R. Sanchez, “On the Spectral Analysis of Iterative Solutions of the Discretized One-Group Transport Equation,” *Annals of Nuclear Energy*, vol. 31, no. 17, pp. 2059–2075, 2004.

[65] A. Constantinescu and D. Y. Anistratov, “Stability Analysis of the Quasidiffusion Method on Periodic Heterogeneous 1D Transport Problems,” *Transport Theory and Statistical Physics*, vol. 38, no. 6, pp. 295–316, 2009.

[66] J. Li, Y. Xu, D. Wang, Q. Shen, B. Kochunas, and T. Downar, “Demonstration of a Linear Prolongation CMFD Method on MOC,” in *PHYSOR 2020 - Transition to a Scalable Nuclear Future*, 2021.

[67] Y. Oshima, T. Endo, A. Yamamoto, Y. Kodama, Y. Ohoka, and H. Nagano, “Impact of Various Parameters on Convergence Performance of CMFD Acceleration for MOC in Multigroup Heterogeneous Geometry,” *Nuclear Science and Engineering*, vol. 194, no. 6, pp. 477–491, 2020.

[68] J. Bezanson, S. Karpinski, V. B. Shah, and A. Edelman, “Julia: A Fast Dynamic Language for Technical Computing,” 2012.

[69] X. S. Li, J. W. Demmel, J. R. Gilbert, L. Grigori, P. Sao, M. Shao, and I. Yamazaki, “SuperLU Users’ Guide,” tech. rep., Lawrence Berkeley National Laboratory, 1999.

[70] H. A. van der Vorst, “Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems,” *SIAM Journal on Scientific and Statistical Computing*, vol. 13, no. 2, pp. 631–644, 1992.

[71] Y. Saad, *Iterative Methods for Sparse Linear Systems*. 2nd ed., 2003.

[72] C. Vuik, R. R. Van Nooyen, and P. Wesseling, “Parallelism in ILU-preconditioned GMRES,” *Parallel Computing*, vol. 24, no. 14, pp. 1927–1946, 1998.

[73] Y. Chen, X. Tian, H. Liu, Z. Chen, B. Yang, W. Liao, P. Zhang, R. He, and M. Yang, “Parallel ILU Preconditioners in GPU Computation,” *Soft Computing*, vol. 22, no. 24, pp. 8187–8205, 2018.

[74] V. Simoncini and E. Gallopoulos, “An Iterative Method for Nonsymmetric Systems with Multiple Right-Hand Sides,” *SIAM Journal on Scientific Computing*, vol. 16, no. 4, pp. 917–933, 1995.

[75] M. L. Parks, E. De Sturler, G. Mackey, D. D. Johnson, and S. Maiti, “Recycling Krylov Subspaces for Sequences of Linear Systems,” *SIAM Journal on Scientific Computing*, vol. 28, no. 5, pp. 1651–1674, 2006.

[76] A. Godfrey, “VERA Core Physics Benchmark Progression Problem Specifications (Rev. 4),” tech. rep., Oak Ridge National Laboratory, 2014.

[77] T. J. Downar, B. M. Kochunas, and S. M. Bowman, “Validation and Verification of the MPACT Code,” in *PHYSOR 2016 – Unifying Theory and Experiments in the 21st Century*, (Sun Valley, ID, USA), 2016.

[78] B. C. Yee, *A Multilevel in Space and Energy Solver for Multigroup Diffusion and Coarse Mesh Finite Difference Eigenvalue Problems*. PhD thesis, University of Michigan, 2018.

[79] N. H. Larsen, “Core Design and Operating Data for Cycles 1 and 2 of Peach Bottom 2,” *Electr Power Res Inst Rep EPRI NP*, no. 562, 1979.

[80] G. Alonso, S. Bilbao, and E. Del Valle, “Impact of the moderation ratio over the performance of different BWR fuel assemblies,” *Annals of Nuclear Energy*, vol. 85, pp. 670–678, 2015.

[81] Q. Shen, Y. Xu, and T. Downar, “Stability Analysis of the CMFD Scheme with Linear Prolongation,” *Annals of Nuclear Energy*, vol. 129, pp. 298–307, 2019.

[82] Q. Shen, N. Adamowicz, B. Kochunas, and E. Larsen, “X-CMFD: A robust iteration scheme for CMFD-based acceleration of neutron transport problems with nonlinear feedback,” in *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M and C 2019*, pp. 2238–2247, 2019.

[83] Q. Shen, *Robust and Efficient Methods in Transient Whole-Core Neutron Transport Calculations*. PhD thesis, University of Michigan, 2021.

[84] D. R. McCoy and E. W. Larsen, “Unconditionally Stable Diffusion-Synthetic Acceleration Methods for the Slab Geometry Discrete Ordinates Equations. Part II: Numerical Results.,” *Nuclear Science and Engineering*, vol. 82, no. 1, pp. 64–70, 1982.

[85] E. M. Gelbard, C. H. Adams, and D. R. McCoy, “Use of Perturbation Theory to Test Ordinate Arrays in Streaming Calculations,” *Transactions of the American Nuclear Society*, vol. 35, pp. 239–241, 1980.