9-12-2014

An Sn Application of Homotopy Continuation in Neutral Particle Transport

Nicholas Myers

Nicholas T. Myers

Candidate

Chemical and Nuclear Engineering

Department

This dissertation is approved, and it is acceptable in quality and form for publication:

Approved by the Dissertation Committee:

Anil Prinja , Chairperson

Jim Warsa

Evangelos Coutsias

Cassiano De Oliveira

An S_N Application of Homotopy Continuation in Neutral Particle Transport

by

Nicholas T. Myers

H.B.S., Nuclear Engineering, Oregon State University, 2008
M.S., Nuclear Engineering, Oregon State University, 2011

DISSERTATION

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Doctorate of Philosophy
Engineering

The University of New Mexico
Albuquerque, New Mexico

July, 2014

Acknowledgments

I would like to thank my advisor, Professor Anil Prinja, for his guidance and support of this research, especially when the results seemed discouraging. I would also like to thank my mentor from my internships at Los Alamos National Laboratory, Dr. Jim Warsa, for starting me down the path of this research and helping me understand the numerical issues of the research as well as learning how to use and develop the Capsaicin code. I would also like to thank Dr. Evangelos Coutsias and Dr. Cassiano de Oliveira for being on my committee and supporting the research.

I would also like to thank my family for their support and encouragement to see this dissertation through to the end. I would like to thank Eric and Teresa Anderson for encouraging me during this research and allowing me to be part of their family during my time in New Mexico. I would like to thank all my friends in the department for their support and willingness to create a supportive environment for discussion of ideas and research.

Finally, I would like to thank Jesus Christ for being my lord and savior. I would never have been able to accomplish this work without the strength and peace that the Lord Jesus Christ has given me. I am very grateful for the mind that He has blessed me with and the educational opportunities that He has given me in my life.

An S_N Application of Homotopy Continuation in Neutral Particle Transport

by

Nicholas T. Myers

H.B.S., Nuclear Engineering, Oregon State University, 2008

M.S., Nuclear Engineering, Oregon State University, 2011

P.h.D., Engineering, University of New Mexico, 2014

ABSTRACT

The objective of this dissertation is to investigate the usefulness of homotopy continuation applied in the context of neutral particle transport where traditional methods of acceleration degrade. This occurs in higher dimensional heterogeneous problems [51]. We focus on utilizing homotopy continuation as a means of providing a better initial guess for difficult problems. We investigate various homotopy formulations for two primary difficult problems: a thick-diffusive fixed internal source, and a k-eigenvalue problem with high dominance ratio. We also investigate the usefulness of homotopy continuation for computationally intensive problems with 30-energy groups.

We find that homotopy continuation exhibits usefulness in specific problem formulations. In the thick-diffusive problem it shows benefit when there is a strong internal source in thin materials. In the k-eigenvalue problem, homotopy continuation provides an improvement in convergence speed for fixed point iteration methods in high dominance ratio problems. We also show that one of our imbeddings successfully stabilizes the nonlinear formulation of the k-eigenvalue problem with a high dominance ratio.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Computational Modeling in Nuclear Engineering Overview

Radiation transport modeling and simulation is a growing field of research in the nuclear, medical, and health physics industry. Computer modeling and simulation is becoming increasingly necessary for such applications as nuclear reactor design, weapons modeling, radiation treatment planning, and criticality safety analysis. Currently, in the United States, there are 104 operating nuclear reactor plants that provide about 20% of our country's energy needs. The Department of Energy has predicted that our energy needs will grow by 22% by 2035 [27]. Nuclear power provides a competitive option for meeting these energy demands, but the upfront cost of a power plant is estimated today to be $\$6 - \$8$ billion [17]. Computational modeling of radiation transport reduces the need to build expensive test facilities for design and safety purposes.

Radiation transport modeling is also necessary for nuclear weapons simulation.

By international law (1996 Comprehensive Nuclear-Test-Ban Treaty [40]), signatory nations are not allowed to perform nuclear explosions as experimental tests. Biological and infrastructure damage as a result of a nuclear explosion must instead be analyzed and understood with computational methods.

There are two general means of approaching radiation transport modeling, deterministic and monte carlo methods. Monte carlo methods employ pseudo-random sampling of statistical distributions to simulate the physical interactions of radiation particles in a system. The user must specify where tallying will occur and must account for the sampling statistical error. Monte carlo methods allow for modeling of complex geometry, but require many particle histories to reduce the statistical error. This generally leads to longer run times for transport problems than deterministic approaches.

Deterministic methods seek to solve the physical equations describing the radiation field distribution in a problem. Unlike monte carlo methods, solving a problem deterministically yields all information for the problem, not just where the user specified a tally. Instead of statistical error, the user must account for systematic error associated with the various numerical discretization methods. Deterministic methods are generally faster than monte carlo methods, but are difficult to employ in modeling complex problems. This research is focused on deterministic methods and will make no more mention of monte carlo methods.

Deterministic methods are generalized into two classifications; direct methods and iterative methods. A direct method uses a finite sequence of operations to solve a problem. The solution delivered will be the exact solution assuming there is no rounding error. An iterative method involves a series of operational sequences that repeat until convergence is achieved. As a solution is converged, the error introduced by lagging parameters in the operational sequence should diminish until machine precision is reached. Iterative methods are particularly valuable for solving

nonlinear problems as well as large linear problems where memory requirements and computing time render direct methods intractable.

An iterative method may not always be globally convergent, depending on its formulation and the nature of the problem. The iterative method may even converge to undesired alternative solutions. This is particularly an issue when solving nonlinear problems. Every iterative method requires an initial starting guess to begin the process. It is important that these starting guesses be within what is called the 'zone of attraction' for the iterative formulation such that a real solution can be converged. An homotopy approach is a means of dealing with this potential sensitivity to initial guesses.

1.2 Homotopy Introduction

An homotopy is a continuous transformation between two functions. The use of homotopy continuation methods as mathematical tools were first developed by Poincare' (1881-1886), Klein (1882-1883), and Bernstein (1910) [2]. Wasserstrom [54] reports that continuation methods have been fruitful in theoretical proofs of existence and uniqueness of problems; finding roots of a polynomial; finding solutions to boundary value problems of nonlinear ordinary differential equations; problems dealing with identification of parameters; and eigenvalue problems of linear ordinary differential operators.

Globalized Newton methods are a type of deformation, where the simple system is a globally convergent iterative scheme that is often linear in convergence rate, and the complex system is the quadratic convergent Newton method. In these methods, care must be taken in choosing when/how to switch from the slower, but more stable, iterative method and into the faster, but less stable, Newton method. However, these methods are not necessarily homotopic since the transition from the stable iterative

solver to the Newton method may not be continuous.

Generally, we are interested in a complex, or difficult to solve, system. A homotopy is formed when we postulate a nearby, easier in some sense to solve, system and then construct a continuous mapping between the two systems. A homotopy may be developed in a variety of ways. The first distinction to make is whether to deform the problem based on the available physics, or introduce artificial scaling. These two approaches are known respectively as natural parameter and artificial parameter continuation. Natural parameter continuation will frequently involve the scaling of physical constants in the desired system.

A numerical homotopy continuation method traces a solution path (or all solution paths) from the defined simple system to the desired difficult system. There are two general approaches to tracing the solution path. Both methods transform the problem into a system of related problems based upon the homotopy parameter chosen.

The first method is to simply discretize the path with respect to the chosen homotopy parameter and solve successively along these discrete points. Tangent vector information at each step of the homotopy may also be used to formulate a predictor-corrector scheme. Not using the tangent vector information results in a naive sequential iteration with an assumed predicted tangent vector of 0.0. Predictor steps are typically done via Forward Euler with a view to keeping the predicted step within the domain of attraction for a Newton like method. A Newton like method is then used to quadratically converge the solution at each step.

The other general path tracing procedure is to formulate the problem in terms of the arclength of the solution curve and trace the path length directly (as opposed to the homotopy parameter). This is called pseudo arc-length continuation and is typically the standard procedure for numerical homotopy continuation in the

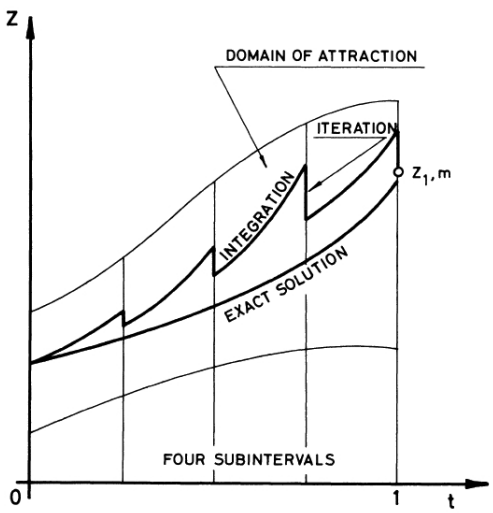

Figure 1.1: General Example of Numerical Homotopy Continuation (Wasserstrom, 1973) [54]

mathematical field. This method also makes use of tangent vector information to establish a predictor-corrector scheme. When using this arclength tracing method, we must also consider adaptive step length strategies as well as bifurcation point detection/handling. Figure 1.1 shows qualitatively what the procedure might look like as one incrementally traces the solution curve. At each point the solution from the previous step is used in projecting an initial guess. Then the problem is minimized to a tolerance and advanced to the next step. At $t = 1$ in Fig 1.1 the solution for the desired difficult problem is converged.

Mathematical considerations of a numerical homotopy formulation are explored by Allgower and Georg [2]. The existence of the solution path curve and its smoothness are guaranteed by satisfying the implicit function theorem. Allgower and Georg also mention that suitable boundary conditions at $\lambda = 0$ or $\lambda = \lambda^*$ will restrict the solution path from running off to infinity or looping back to the condition when $\lambda = 1$. These are generally problem dependent and are examined in relation to exis-

tence theorems for non-linear equations. A more in depth look at existence theorems is examined in [2] and will not be discussed further.

The use of continuation techniques in the field of nuclear engineering is limited. The invariant imbedding method is best thought of as a special case form of continuation method and was developed by Bellman, Kalaba, and Wing [8]. Time stepping algorithms have also been used to help converge a steady state solution for transport problems. This is also a type of special case continuation.

Very recently, Martin [38] showed that the Homotopy Perturbation Method (HPM) [25, 26] can be used to improve convergence of simple 1D slab neutron transport problems. The HPM is a known homotopy technique for solving nonlinear partial differential equations that involves a series expansion about the homotopy parameter (which varies from zero to one) in order to generate a convergent series solution of differential equations. It is a specialized form of the more general Homotopy Analysis Method (HAM) [26]. This method is different than what is used in this dissertation and won't be elaborated on further.

We investigate using numerical homotopy continuation techniques as a means of improving convergence of a solution for the difficult nuclear engineering problems that motivate our research. These problems are: the thick-diffusive neutral particle transport problem with a fixed internal source and the k-eigenvalue problem with a high dominance ratio. We are particularly interested with such problems in a multidimensional setting.

1.3 Motivation

In the nuclear engineering field we are concerned with the linearized Boltzmann equation for representing radiation fields. This equation is an integro-differential equation

that is difficult to solve analytically, except in simplified special case scenarios. A series of case studies for analytic solutions to the neutron transport equation can be found in [18].

Instead, numerical methods are employed in order to discretize the continuum phase space of time, space, angle, and energy. The discretized form of the transport equation is then solved with iterative methods on a computer. Accuracy, speed, and stability are typical concerns that we face when deciding which numerical method to employ for solving our desired radiation transport problem. These issues will be discussed more in the context of the fixed internal source and k-eigenvalue problems.

1.3.1 Thick-Diffusive Neutral Particle Transport with Fixed Internal Source

Particle transport problems are difficult to converge in the thick-diffusive limit [1]. This occurs when scattering is dominant and the mean free path of the particle is very small. This leads to a slow convergence rate when applying an iterative solver to the transport equation. For a robust transport code to be effective, it must be able to solve the transport problem in the thick-diffusive limit for complex geometry and heterogenous materials. This must also be accomplished in an efficient manner.

Many acceleration schemes have been researched in order to speed up convergence of this difficult problem. The most well known method is Diffusion Synthetic Acceleration (DSA) as first introduced by Kopp [33]. The Diffusion Synthetic Acceleration (DSA) method makes use of a lower order diffusion equation to provide a correction to the transport sweep iterate. Larsen [34, 35] showed through an asymptotic limit scheme that the diffusion equation represents the transport equation that is in the thick-diffusive limit and is indeed an appropriate lower order equation for the thick-diffusive problem. DSA is a very powerful acceleration method, but has

recently been shown to have some shortcomings. Azmy first showed that DSA is not unconditionally stable in heterogeneous higher dimensional problems [5–7]. Warsa, et, al. further examined the degraded effectiveness of DSA when Krylov methods are used [52].

Another similar acceleration method to DSA is Transport Synthetic Acceleration (TSA) [1]. This method chooses its lower order equation as the transport problem with a reduced S_N quadrature. Chang and Adams [14] have shown that for highly heterogeneous problems, the TSA method can become divergent. Other acceleration methods are the Quasi-Diffusion method as first introduced by Goldin [22], Coarse Mesh Rebalance [36], and Two-Cycle Acceleration [36] methods. These methods aren't implemented in this dissertation and will not be discussed further. We seek to use homotopy continuation as a means of generating an initial guess for these thick-diffusive difficult problems such that our numerical solvers have a global improvement in convergence time.

1.3.2 k-Eigenvalue Problems with High Dominance Ratio

k-eigenvalue problems arise often in reactor design and criticality safety for stored fissile material. It is an eigenvalue problem that describes the steady state system of multiplying fissile nuclear material. For time dependent problems, a different eigenvalue problem known as the "alpha" eigenvalue problem is employed. We will not examine such problems in this dissertation.

The standard method in the nuclear engineering industry [16] for solving k-eigenvalue problems is to use power iteration. Power iteration is a well known numerical iterative method used to converge the dominant eigenmode for an eigenvalue problem and converges to the largest eigenvalue at a rate determined by the dominance ratio. The dominance ratio is the ratio of the second largest eigenvalue with

respect to the largest eigenvalue for the problem.

The largest, or dominant, k-eigenvalue is also known as the multiplication factor and is defined as the ratio of the number of particles born in a new generation over the number of particles born in the previous generation. When the eigenvalue, k, is less than one, the system is said to be subcritical and is not self-sustaining. When k is greater than one, the system is said to be supercritical and the neutron population will rapidly grow. When k is exactly one, the system is said to be critical, or self-sustaining. For power reactor applications, we generally operate with systems where the k-eigenvalue is slightly greater than one due to the delayed neutron contribution. There are also other applications where we are concerned with pulsed prompt supercritical systems, such as pulsed reactor assembly experiments.

k-eigenvalue problems are difficult to solve when the dominance ratio is close to 1.0 [9]. A high dominance ratio is indicative of neutron multiplication with poor communication of neutron distribution changes between regions. Such cases occur often in reactor systems where there are multiple fissioning fuel assemblies with a moderating material in between, such as water, that causes a high amount of scattering interactions.

Recently Gill, et, al. [21] have formulated the k-eigenvalue problem in a nonlinear fashion such that Newton type methods can be utilized to improve convergence speed. Recasting the problem in this way allows for the eigenvalue to show up anywhere in the formulation, rather than as a standard or generalized eigenvalue problem. This has allowed for beneficial operator splitting and utilization of inner iterations to improve convergence for large multigroup problems. However, a nonlinear formulation of the k-eigenvalue problem is more sensitive to the initial guess.

A study by Chu [15] proved that homotopy paths do not cross for symmetric real eigenvalue problems with a diagonal matrix chosen as the initial imbedding state.

This guarantees that the dominant eigenmode can be determined via homotopy path tracing since no bifurcation points are encountered. However, in reality, our matrices are real non-symmetric where there is no such guarantee [41]. Li and Zeng [37] show that perturbations in path tracing are necessary to avoid bifurcation. They comment that round off error from numerically converging the initial imbedding state is usually sufficient to avoid a bifurcation point. However, undesired 'path jumping' may occur during the eigenvalue path tracing.

We seek to use homotopy continuation to robustly provide a stable initial guess for k-eigenvalue problems with high dominance ratios and whether homotopy continuation can also be used to improve convergence speed.

1.4 Scope of Document

The remainder of this dissertation is organized in the following way:

II Chapter 2 contains a deeper discussion of homotopy. We develop the theory behind homotopy continuation in greater detail. We present algorithms for implementing homotopy continuation in a numerical code and we discuss the pseudo-arclength formulation and the mathematical considerations apply to the implementation of the homotopy method. Finally, we give a 1D polynomial example that utilizes homotopy continuation to solve for the roots of a given polynomial problem.

III Chapter 3 discusses the neutral particle transport equation model that is used in this research. We outline the numerical discretization techniques as well as the various numerical iterative solvers that are applied in this research.

IV Chapter 4 presents the numerical formulation and implementation of the homotopy concept with regard to thick-diffusive problems with a fixed internal source.

We develop a natural parameter continuation homotopy called diffusion length preserving continuation. We also develop two artificial parameter continuations; a reduced S_2 quadrature imbedding and a diffusion imbedding. We test these homotopy formulations on problems for which diffusion synthetic acceleration is known to degrade. We compare the effectiveness of each method in terms of function evaluations and observe where homotopy continuation is useful.

V Chapter 5 presents the numerical formulation and implementation of the homotopy concept with regard to k-eigenvalue problems. We present results from a 1D homogeneous slab where we scale the dominance ratio and compare results for various solution methods. We also present heterogeneous results for low and high dominance ratio problems. We investigate the usefulness of pseudo-arclength tracing versus direct parameter tracing with the nonlinear formulation of the k-eigenvalue problem.

VI Chapter 6 contains the conclusions obtained from this work. We discuss the benefits of using numerical homotopy continuation for the purpose of improving convergence of solutions to specific thick-diffusive, fixed internal source problems. We also discuss the usefulness of numerical homotopy continuation with regard to k-eigenvalue problems that have high dominance ratios. We suggest a preferred approach to implementing homotopy continuation in neutral particle transport codes. Finally, we give suggestions for future work that can be performed in extending understanding and implementation of homotopy methods in numerical radiation transport.

Chapter 2

Homotopy Continuation

2.1 Overview

An homotopy is a continuous deformation between two functions. Generally, one is interested in a certain problem, $F(x)$, that is non-trivial to solve. The homotopy continuation method has one create a neighboring easy–or easier–problem, $G(x)$, and then trace the solution from $G(x)$ to $F(x)$. This transforms the original problem into a series of systems by adding an additional rank. This allows us to approach a problem as an initial value problem. Ideally, this will aid convergence to a solution for the difficult problem, $F(x)$.

There are multiple ways to formulate a homotopy. The choice of homotopy parameter(s) is variable, as are the path tracing methods. Path following can be performed very simply by tracing with respect to the chosen homotopy parameter. The more generally preferred method in the mathematical field is to implicitly trace the arc-length of the solution curve.

In this chapter we will outline some of the homotopy formulation possible as well

as different ways of tracing the solution path. We will further explore the numerical considerations for implementing homotopy continuation in a code. Finally, we will present a very simple polynomial root finding problem as an example of homotopy continuation.

2.1.1 Theory

Consider a system that we wish to converge an answer to

$$F(x) = 0, \tag{2.1}$$

where F represents a residual formulation of an equation and is a smooth mapping, $F : R^n \rightarrow R^n$. Generally speaking, $F(x)$ may be a multi-variate system. If the initial guess, x_0, is not well known *a priori* and the system is difficult to solve (i.e. is nonlinear), then a homotopy formulation becomes useful for converging a solution to $F(x)$ in a stable manner. We first construct a similar system, $G(x)$, where the solution is known, or trivial, to converge

$$G(x) = 0, \tag{2.2}$$

where G is also a smooth mapping, $G : R^n \rightarrow R^n$. A homotopy is then generally defined as

$$H : R^n \times R^m \rightarrow R^{n \times m}. \tag{2.3}$$

One or more scaling parameters are embedded to provide the mapping from G to F. We continue assuming a single parameter, λ. We embed the parameter such that $H(x, 0) = G(x)$ and $H(x, 1) = F(x)$. The existence of the solution path/curve and its smoothness are guaranteed by satisfying the implicit function theorem. We can state the implicit function theorem in terms of our defined quantities in the following manner:

Theorem 2.1.1 *let $H : R^n \times R \to R^n$ be a continuously differentiable function, and let $R^n \times R$ have coordinates (x, λ). Fix a point $(a, b) = (a_1, \ldots, a_n, b_1)$ with $H(a, b) = c$ where $c \in R$. If the Jacobian $[J_x(a, b)]$ is invertible, then there exists an open set U containing a, an open set V containing b, and a unique continuously differentiable function $Z : U \to V$ such that $\{(Z(\lambda), \lambda) | \lambda \in U\} = \{(x, \lambda) \in U \times V | H(x, \lambda) = c\}$.*

2.1.2 Homotopy Formulation

A homotopy may be developed in a variety of ways. The first distinction to make is whether to deform the given problem based on the available physics, or introduce an artificial scaling parameter. These two approaches are known respectively as natural parameter and artificial parameter continuation.

Natural parameter continuation will frequently involve the scaling of physical constants in the desired system. This scaling will alter the associated physical operators and forcing functions to the system. An example of such scaling is shown:

$$H(x, \lambda) = b(\lambda) - A(\lambda)\, x, \tag{2.4}$$

with $\lambda \in [\lambda^*, 1]$ as the natural homotopy parameter. At λ^* this problem is scaled to an easily converged system, $H(x, \lambda^*) \equiv G(x, \lambda^*)$. At $\lambda = 1$, the problem returns to the original difficult system, $H(x, 1) \equiv F(x, 1)$. The advantage of the natural parameter formulation is that there is more assurance that the embedded easy problem is in the neighborhood of the true physical problem. Ideally this means the solution path is straightforward to trace and has no turning points.

Artificial parameter continuation externally applies an arbitrary parameter to the chosen simple system, $G(x)$, and the original complex system, $F(x)$. The most common artificial parameter type is the convex homotopy:

$$H(x, \lambda) = (1 - \lambda)G(x) + \lambda F(x), \tag{2.5}$$

with $\lambda \in [0, 1]$ as the artificial homotopy parameter. It is not necessary to formulate $G(x)$ based on any physical parameters from the original system, $F(x)$. This allows for more freedom in choosing $G(x)$. However, there is a greater responsibility on the user's part to verify that the imbedding is a neighbor to $F(x)$. A poor choice of $G(x)$ may result in the path tracing algorithm degrading or even breaking down.

Another standard formulation is the global homotopy [2] where a manufactured solution is used to inform the initial system:

$$H(x, \lambda) = F(x) - (1 - \lambda)F(x_0). \tag{2.6}$$

This formulation requires $F(x)$ to be evaluated at every step and is unlikely to help convergence speed. However, possible instabilities from a poor $G(x)$ choice are avoided by restricting the system to F.

Once a homotopy formulation has been chosen, a path tracing method needs to be developed. The simplest way of tracing the solution path, is to directly trace along the homotopy parameter. A more sophisticated technique is to implicitly trace the arc-length of the solution path in what is known as the pseudo arc-length method. In either case, it is generally beneficial to formulate some type of predictor-corrector algorithm that makes use of the tangent vector information (Jacobian in multi-variate systems).

2.1.3 Homotopy Parameter Path Tracing

Homotopy continuation begins with the imbedded system that has a known solution or is trivial, or in a sense, easy to solve. Assuming that this solution, x_0, is easily acquired, we project the solution forward along the tangent vector with respect to the homotopy parameter and solve the homotopy system at the new state. The

tangent vector may be derived analytically as

$$\frac{\partial}{\partial x}H(x,\lambda)\frac{\partial x}{\partial s}+\frac{\partial}{\partial \lambda}H(x,\lambda)\frac{\partial \lambda}{\partial s}=0, \tag{2.7a}$$

$$\frac{\partial}{\partial x}H(x,\lambda)\frac{\partial x}{\partial s}=-\frac{\partial}{\partial \lambda}H(x,\lambda)\frac{\partial \lambda}{\partial s}, \tag{2.7b}$$

$$\frac{\partial x}{\partial s}=-\left(\frac{\partial}{\partial x}H(x,\lambda)\right)^{-1}\frac{\partial}{\partial \lambda}H(x,\lambda)\frac{\partial \lambda}{\partial s}, \tag{2.7c}$$

$$\vec{t}\equiv\left(\frac{\partial x}{\partial s}\right)\left(\frac{\partial \lambda}{\partial s}\right)^{-1}=-\left(\frac{\partial}{\partial x}H(x,\lambda)\right)^{-1}\frac{\partial}{\partial \lambda}H(x,\lambda). \tag{2.7d}$$

The predictor step is often performed as a simple forward Euler integration

$$\omega_{n+1}=x_n+\int_{\lambda_n}^{\lambda_{n+1}}d\lambda\vec{t}(x_n,\lambda_n), \tag{2.8}$$

where n is the discrete homotopy point index. The predicted solution is then corrected with some iterative minimization technique with respect to the new homotopy state, λ_{n+1}, using ω_{n+1} as the initial guess to the iterative solver.

$$x_{n+1}=\text{minimize}(\omega_{n+1},\lambda_{n+1}). \tag{2.9}$$

Since the initial guess is ideally near the true solution curve, a Newton type method is often preferred for a correcting solver. This sequence of steps is repeated until $\lambda_N=1.0$. With this approach the user must discretize the homotopy path *a priori* when it is not generally known what the optimal number of homotopy intervals ($\Delta\lambda$) should be.

Algorithm 1 Direct Continuation Method

Given x_0 and $H(x, \lambda)$

for $n = 0, \ldots, N$ **do**

$\lambda_n = \frac{n}{N}$

Calculate Tangent $\vec{t}(x_n, \lambda_n) = -H_x^{-1}(x_n, \lambda_n) H_\lambda(x_n, \lambda_n)$

Predict $\omega_{n+1} = x_n + \int_{\lambda_n}^{\lambda_{n+1}} d\lambda \vec{t}(x_n, \lambda_n)$

Correct for x_{n+1} by minimizing $H(x_{n+1}, \lambda_{n+1})$ with initial guess ω_{n+1}

end for

2.1.4 Pseudo-Arclength Continuation

A common homotopy tracing technique is to formulate the problem with respect to the arclength of the solution path. Tracing with respect to the arclength is advantageous because it is a natural parameter to the system and should not be as susceptible to turning point issues like an ill-suited artificial parameter would [2]. It is not necessary to explicitly trace the arclength, but rather to implicitly trace it using the pseudo-arclength continuation (PSARC) method. We present the basics of this concept in this section.

We can represent the arc-length as a vector with components equal to our homotopy parameter step and our solution parameter step

$$\vec{s} \equiv < \Delta x, \Delta\lambda > . \tag{2.10}$$

This allows us to represent our homotopy in terms of the arc-length parameter

$$H(c(\vec{s})) = 0, \tag{2.11}$$

where $c(s)$ is the solution curve of the homotopy. We can then define our discrete solution point on the homotopy curve as

$$c_n \equiv < x_n, \lambda_n > , \tag{2.12}$$

where n is the homotopy tracing index. We define the tangent of the curve as

$$\vec{t} \equiv \dot{c}(s) =< \frac{\partial x}{\partial s}, \frac{\partial \lambda}{\partial s} > . \tag{2.13}$$

From Allgower and George [2] we arrive at the three conditions necessary to uniquely determine a tangent vector. We restate the Lemma here

Lemma 2.1.2 *Let $c(s)$ be the positively oriented solution curve parametrized with respect to arclength $\vec{s}$ which satisfies $c(0) = u_0$ and $H(c(s)) = 0$ for $\vec{s}$ in some open interval J containing zero. Then for all $s \in J$, the tangent, $\vec{t}$ satisfies the following three conditions:*

$$H'(c(s))\vec{t} = 0, \tag{2.14a}$$

$$\| \vec{t} \| = 1, \tag{2.14b}$$

$$det \begin{pmatrix} H'(c(s)) \\ \vec{t}^* \end{pmatrix} > 0. \tag{2.14c}$$

We now form our problem as an initial value problem (IVP)

$$\frac{dc}{ds} = \vec{t}(H'(c(s))), \tag{2.15a}$$

$$c(0) = c_0, \tag{2.15b}$$

where $\vec{t}$ is computed by solving the higher rank system

$$\begin{pmatrix} H_x & H_\lambda \\ (\partial x/\partial s)^* & (\partial \lambda/\partial s)^* \end{pmatrix}_n \begin{pmatrix} \partial x/\partial s \\ \partial \lambda/\partial s \end{pmatrix}_n = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{2.16}$$

We can trace the solution curve by integrating over the arc-length

$$\omega_{n+1} = c_n + \int_{\xi_n}^{\xi_{n+1}} d\xi \vec{t}(H'(c(\xi)), \tag{2.17}$$

where ξ is in the existence of the solution curve c. Finally, we correct to the solution curve by solving the following higher rank system with a Newton type method

$$\begin{pmatrix} H_x(\omega_{n+1}) & H_\lambda(\omega_{n+1}) \\ (\partial x/\partial s)^*_n & (\partial \lambda/\partial s)^*_n \end{pmatrix} \begin{pmatrix} \partial x \\ \partial \lambda \end{pmatrix}_{n+1} = \begin{pmatrix} -H(\omega_{n+1}) \\ 0 \end{pmatrix}. \tag{2.18}$$

The additional rank that corresponds to the homotopy parameter is a constraint equation that seeks an orthogonal correction step with respect to the tangent vector. With this IVP, one can trace a solution curve from the imbedding state to the desired true problem state. We note that the term $H'(c(s))$ is the Jacobian of the homotopy formulation. It is not always efficient, or even possible, to directly calculate this for large scale scientific problems.

For large scale problems, calculating this tangent vector directly can be untenable. Georg recommends using the numerical secant approximation for determining a tangent vector to the solution curve [19]. The secant tangent vector is determined as follows

$$t_{n+1} := \frac{c_{n+1} - c_n}{\| c_{n+1} - c_n \|}. \tag{2.19}$$

A matrix-free type corrector is also recommended for large scale scientific problems. In this research, we use the Jacobian-Free Newton Krylov method. This method is outlined further in section 2.1.5.

A robust numerical implementation of the pseudo arc-length tracing concept generally involves an adaptive step length algorithm and special point handling. We implement a basic adaptive step length algorithm from Allgower and Georg's paper [2] that adapts the step length relative to the contraction rate of the Newton type method. We define our discrete step length of integration as h. The adaptive

algorithm to calculate a new step length, $\tilde{h}$, is given as follows

$$\tilde{h} = h * \tilde{p}, \tag{2.20a}$$

$$\tilde{p} = \max(\min(p, 2), 0.5), \tag{2.20b}$$

$$p = \sqrt{\frac{\kappa_0}{\kappa(c)}}, \tag{2.20c}$$

$$\kappa(c) = \frac{\|\epsilon_1\|}{\|\epsilon_0\|}, \tag{2.20d}$$

where κ represents a contraction rate, κ_0 is chosen baseline contraction rate, and ϵ is the residual error at the indicated Newton iterate.

There are two types of special points that are important to monitor--turning points and bifurcation points. A turning point can be detected during the path length tracing by monitoring when the Schur complement,Υ, is zero for the homotopy parameter constraint of the Jacobian in Eq. (2.16). For a our general homotopy formulation, $H(x, \lambda)$, this takes the following form

$$\Upsilon = \left(\frac{\partial \lambda}{\partial s}\right)^* - \left(\frac{\partial x}{\partial s}\right)^* H_x^{-1} H_\lambda. \tag{2.21}$$

If a turning point is detected, the path tracing algorithm perturbs the predicted values forward with respect to λ in an attempt to 'jump over' the special point. Bifurcation points can be detected when there is a sign change of the system determinant

$$\det \begin{pmatrix} H(\dot{c}(s)) \\ \vec{t}^* \end{pmatrix}. \tag{2.22}$$

If a bifurcation point is detected, the path tracing algorithm perturbs the entire system and traces this perturbed system for a few steps before returning to the un-perturbed system. Hopefully this bypasses the bifurcation point. Due to the computational cost of evaluating for these special points, robust pseudo arc-length tracing can become untenable for large scale scientific problems.

Algorithm 2 Pseudo Arc-Length Continuation Method

Given c_0 and $H(x, \lambda)$

$\lambda_0 = 0$

$\vec{t_0} = h$

while $\lambda < 1$ **do**

Predict $\omega_{n+1} = c_n + h\vec{t}(c_n)$

Correct for c_{n+1} by minimizing $\begin{pmatrix} H(\omega_{n+1}) + \dot{H}(\omega_{n+1}) \\ \vec{t}^* \end{pmatrix} \delta c$, calculate $\tilde{p}$

if $\tilde{p} == 0.5$ **then**

Adapt Step Length $h = h * \tilde{p}$

Break corrector iteration and re-predict

end if

if special point detected **then**

Perturb system

end if

Calculate Tangent $t_{n+1} = \frac{c_{n+1} - c_n}{\|c_{n+1} - c_n\|}$

Adapt Step Length $h = h * \tilde{p}$

end while

2.1.5 Jacobian-Free Newton Krylov

The well known Newton iteration process [4] is outlined as follows

$$F(u^k) + \delta u^k J^k \approx 0, \tag{2.23a}$$

$$J^k \delta u^k = -F(u^k), \tag{2.23b}$$

$$u^{k+1} = u^k + \delta u^k. \tag{2.23c}$$

Here p is the Newton iteration index and J is the Jacobian matrix. In the JFNK method [31], we use a Krylov non-stationary iterative solver to solve the linear system

for the correction step, δu. We also do not explicitly form the Jacobian matrix. We instead use a finite difference scheme to approximate the action of the Jacobian operator on δu. A first order finite differencing is given as

$$Jv \approx \frac{F(u+\epsilon v) - F(u)}{\epsilon}. \tag{2.24}$$

The choice of ϵ is important. If it is too large, then the approximation to the Jacobian is too coarse. If ϵ is too small, machine error can cause instabilities in the finite differencing. Different schemes exist to calculate an appropriate ϵ. We present one such scheme as follows from Knoll and Keyes [31]

$$\epsilon = \frac{1}{N\|v\|_2} \sum_{i=1}^{N} (a|u_i^k| + a), \tag{2.25}$$

where v is the Krylov vector, N is the system dimension, u_i^p is the system parameter at the current iterate, and a is a constant with a magnitude close to the square root of machine precision. In this dissertation, we simply specify ϵ to be the inner tolerance given for the Krylov solver unless it is otherwise noted.

Approximating the action of the Jacobian on a vector requires that two system vectors be calculated—one for the solution at the current iteration and one for a perturbed solution at the current iteration. The Newton iteration is continued until the desired norm of the residual vector is smaller than a specified tolerance. In this study the L_2 norm is used for the purposes of determining the error of the system.

2.2 Polynomial Rootfinding Tutorial

We present a homotopy continuation example using a simple second degree polynomial problem. We begin by establishing our 'hard' problem that we wish to minimize (i.e. find the roots)

$$F(x) = x^2 - 3x + 2. \tag{2.26}$$

We know by factoring the problem that there are two real roots at $r_1 = 2.0$ and $r_2 = 1.0$. However, we will assume for the purpose of this example that we do not know the solution to this problem. We will construct a polynomial system that we know the roots for a priori, $\tilde{r_1} = 3.0$ and $\tilde{r_2} = -1.0$,

$$G(x) = (x-3)(x+1) = x^2 - 2x - 3. \tag{2.27}$$

Now we construct an homotopy using the convex artificial parameter formulation

$$H(x,\lambda) = (1-\lambda)G(x) + \lambda F(x), \tag{2.28}$$

where $\lambda \in [0,1]$. In order to illustrate pseudo-arclength continuation, we show how the tangent vector for the solution curve is calculated:

$$\frac{\partial H(x,\lambda)}{\partial x}\frac{\partial x}{\partial \lambda} = -\frac{\partial H(x,\lambda)}{\partial \lambda}, \tag{2.29}$$

with the tangent vector defined as $\frac{\partial x}{\partial \lambda}$. After performing the necessary derivatives, we solve for the tangent vector,

$$\frac{\partial x}{\partial \lambda} = \frac{x-5}{2x-2-\lambda}. \tag{2.30}$$

Clearly, this method will only work if the denominator does not equal zero. If we were working in a multivariate system, this would require the Jacobian to remain non-singular over the necessary domains of x and λ.

Now we perform the predictor step with a simple Forward Euler numerical integration step:

$$x^{k+\frac{1}{2}} = x^k + \Delta\lambda \left(\frac{\partial x}{\partial \lambda}\right)^k, \tag{2.31}$$

where k is the homotopy interval index such that $\lambda^{k+1} - \lambda^k = \Delta\lambda$. It is assumed here that the homotopy intervals are uniformly spaced. Now a corrector step is applied. For this simple problem, we will correct with the Trapezoidal method:

$$x^{k+1} = x^k + \frac{1}{2}\Delta\lambda \left[\left(\frac{\partial x^k}{\partial \lambda^k}\right) + \left(\frac{\partial x^{k+\frac{1}{2}}}{\partial \lambda^{k+1}}\right)\right]. \tag{2.32}$$

With this answer for x^{k+1}, we repeat the continuation method to predict the solution at the next step, correct the prediction, and repeat until we arrive at $\lambda = 1$. In this example, each root is individually traced through the homotopy.

Table 2.1: Tabulated results of homotopy continuation with ten homotopy intervals for the system, $F(x) = x^2 - 3x + 2$ and $G(x) = x^2 - 2x - 3$.

Interval	λ	Root 1	Root 2
0	0.0	3.00	-1.00
1	0.1	2.95	-8.48
2	0.2	2.89	-6.92
3	0.3	2.83	-5.30
4	0.4	2.76	-3.62
5	0.5	2.69	-1.86
6	0.6	2.60	-2.99×10^{-05}
7	0.7	2.50	0.20
8	0.8	2.38	0.42
9	9.0	2.23	0.67
10	10.0	2.00	0.97

Table 2.2: Tabulated results of homotopy continuation using PSARC for the system, $F(x) = x^2 - 3x + 2$ and $G(x) = x^2 - 2x - 3$.

Root 1		Root 2	
λ	x	λ	x
0.0	3.00	0.0	-1.00
0.089	2.95	2.95	-0.917
0.262	2.85	0.164	-0.749
0.580	2.62	0.373	-0.408
1.0	2.00	0.745	0.297
		1.0	1.00

We present results for this simple tutorial problem in Table 2.1 and Figure 2.1 for a direct continuation implementation. Only explicit predictor-corrector methods were used. It can be seen that the answers are converging to the true roots to the system, $r_1 = 2.00$ and $r_2 = 1.00$. However, since only ten homotopy intervals were

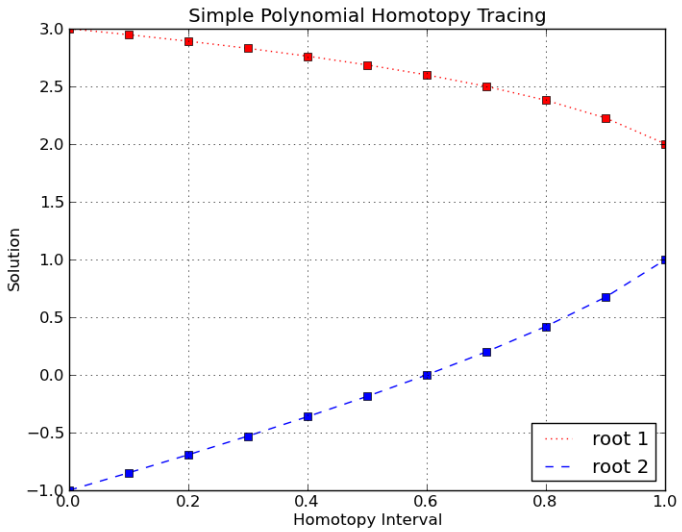

Figure 2.1: Plot of homotopy continuation with ten homotopy intervals for the system, $F(x) = x^2 - 3x + 2$ and $G(x) = x^2 - 2x - 3$.

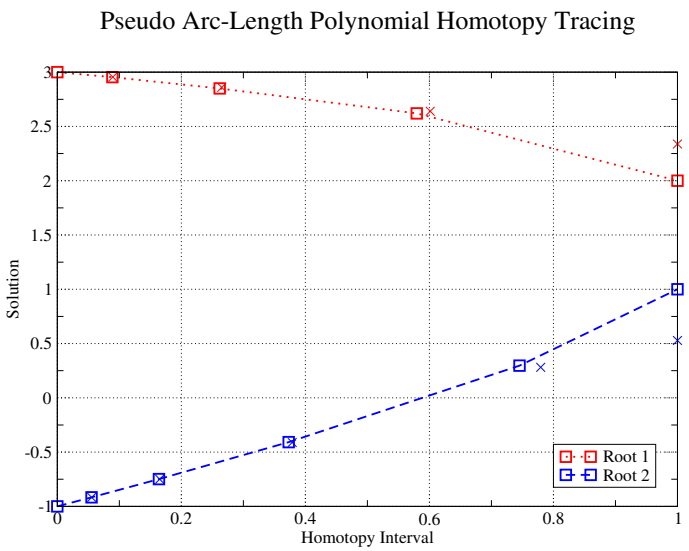

Figure 2.2: Plot of homotopy continuation using PSARC for the system, $F(x) = x^2 - 3x + 2$ and $G(x) = x^2 - 2x - 3$.

used and only explicit methods were used, there is a slight error with the second root only reporting as $r_2 = 0.97$.

Table 2.2 and Figure 2.2 show results using the more robust pseudo-arclength method. We use Newton's method as a corrector and directly invert the Jacobian using the LAPACK GESV package [3]. We converge each corrector (Newton) step to a tolerance of 1.0×10^{-08}. We implement the adaptive steplength algorithm specified in (2.20). The tracing algorithm has no problem following the solution curves of the two roots and correctly converges the appropriate solutions for $F(x)$.

Chapter 3

Transport Model

3.1 Overview

In this chapter we present the neutron transport model that concerns this research. We discuss both the fixed internal source problem and the k-eigenvalue problem. We first examine the continuous theoretical model and then describe the process of numerical discretization. Finally, we outline some of the common acceleration and preconditioning methods used by the transport community to improve convergence.

3.2 Neutral Particle Transport Equation

The radiation transport model of interest is derived from the Boltzmann equation. It is often referred to as the linearized Boltzmann equation. The steady state, general

geometry transport equation with a fixed internal source can be expressed as

$$\vec{\Omega} \cdot \nabla \psi(\vec{r}, \vec{\Omega}, E) + \sigma_t(\vec{r}, E)\psi(\vec{r}, \vec{\Omega}, E) = \tag{3.1}$$
$$\int_0^\infty \int_{4\pi} d\Omega' dE' \sigma_s(\vec{r}, \Omega' \to \Omega, E' \to E)\psi(\vec{r}, \Omega', E') + Q(\vec{r}, \Omega, E)$$

,

where the individual components are defined in the following manner

$$\vec{\Omega} \cdot \nabla \psi(\vec{r}, \vec{\Omega}, E) \equiv \text{Streaming Losses}, \tag{3.2a}$$
$$\sigma_t(\vec{r}, E)\psi(\vec{r}, \vec{\Omega}, E) \equiv \text{Total Interaction Losses}, \tag{3.2b}$$
$$\int_0^\infty \int_{4\pi} d\Omega' dE' \sigma_s(\vec{r}, \Omega' \to \Omega, E' \to E)\psi(\vec{r}, \Omega', E') \equiv \text{Scattering Source}, \tag{3.2c}$$
$$Q(\vec{r}, \Omega, E) \equiv \text{Fixed Internal Source}. \tag{3.2d}$$

The steady state, general geometry k-eigenvalue transport problem can be expressed as

$$\vec{\Omega} \cdot \nabla \psi(\vec{r}, \vec{\Omega}, E) + \sigma_t(\vec{r}, E)\psi(\vec{r}, \vec{\Omega}, E) = \tag{3.3}$$
$$\int_0^\infty \int_{4\pi} d\Omega' dE' \sigma_s(\vec{r}, \Omega' \to \Omega, E' \to E)\psi(\vec{r}, \Omega', E') +$$
$$\frac{1}{k} \int_0^\infty dE' \nu(E)\sigma_f(\vec{r}, E' \to E) \int_{4\pi} d\Omega' \psi(\vec{r}, \Omega', E'),$$

where

$$\int_0^\infty dE' \nu(E)\sigma_f(\vec{r}, E' \to E) \int_{4\pi} d\Omega' \psi(\vec{r}, \Omega', E') \equiv \text{Fission Source}, \tag{3.4}$$

and k is the eigenvalue for the system; it represents the ratio of new neutral particles created in a generation to the number of neutral particles in the previous generation.

Equations (3.1) and (3.3) describe the average angular flux field, $\psi(\vec{r}, \Omega, E)$, of neutrally charged particles in the phase space $(\vec{r}, \Omega, E)$. These equation are representative of a linearized Boltzmann Equation for the fixed internal source and

k-eigenvalue problems. There are six dimensions to this equation: three in space, two in angle, and one in energy. In this dissertation we have neglected the time component of the linearized Boltzmann equation because our problems are all steady state.

We may express the fixed source equation in operator notation as follows

$$L\psi = MSD\psi + Q, \tag{3.5}$$

and the k-eigenvalue problem in operator notation as

$$L\psi = MSD\psi + \frac{1}{k}MFD\psi, \tag{3.6}$$

where L is the transport streaming operator, M is the moment-to-discrete operator, D is the discrete-to-moment operator, S is the scattering operator, and F is the fission source distribution operator.

In order to deterministically solve this equation for more general and complex systems, we discretize the equation over the phase space. The discretized problem is then solved with an iterative numerical scheme on a computer. We outline our discretization methods in the following subsections.

3.2.1 Energy Discretization

The energy phase space is discretized using the well known multi-group method [16] represented in Figure 3.1. Energy space is divided into groups that represent a range of energy. Angular flux values are then calculated as averages of these groups. Applying this energy discretization to Equation (3.1) yields the following result for the fixed source problem

$$\vec{\Omega} \cdot \nabla \psi_g(\vec{r}, \vec{\Omega}) + \sigma_{t_g}(\vec{r})\psi_g(\vec{r}, \vec{\Omega}) = \tag{3.7}$$
$$\sum_{g'=0}^{G} \int_{4\pi} d\Omega' \sigma_{s_{g' \to g}}(\vec{r}, \Omega' \to \Omega)\psi_{g'}(\vec{r}, \Omega') + Q_g(\vec{r}, \Omega),$$

and the following for the k-eigenvalue problem

$$\vec{\Omega} \cdot \nabla \psi_g(\vec{r}, \vec{\Omega}) + \sigma_{t_g}(\vec{r}) \psi_g(\vec{r}, \vec{\Omega}) = \tag{3.8}$$
$$\sum_{g'=0}^{G} \int_{4\pi} d\Omega' \sigma_{s_{g' \to g}}(\vec{r}, \Omega' \to \Omega) \psi_{g'}(\vec{r}, \Omega') +$$
$$\frac{1}{k} \sum_{g'=0}^{G} \nu_g \sigma_{f_{g' \to g}}(\vec{r}) \int_{4\pi} d\Omega' \psi_{g'}(\vec{r}, \Omega'),$$

where the subscript, g, is the associated energy group index for the equation and G represents the maximum number of energy groups chosen. The size of the numerical problem that must be solved is increased by a factor of G.

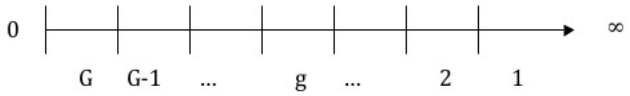

Figure 3.1: Multigroup Example Discretization of Continuous Energy Spectrum

The energy group cross sections are defined as flux weighted averages over the defined energy group. This is given as follows:

$$\sigma_g(\vec{r}, \Omega) = \frac{\int_{E_{g+\frac{1}{2}}}^{E_{g-\frac{1}{2}}} dE' \phi(\vec{r}, \Omega, E') \sigma(\vec{r}, \Omega, E')}{\int_{E_{g+\frac{1}{2}}}^{E_{g-\frac{1}{2}}} dE' \phi(\vec{r}, \Omega, E')}. \tag{3.9}$$

Because we often are trying to solve for ϕ, the actual flux weighting utilizes spectral or distribution shape approximations to the true flux, ϕ. In this research, all group cross sectional data will be defined explicitly for the problem, or taken from the Nuclear Data Interface (NDI) database provided by Los Alamos National Laboratory. The NDI data is averaged according to a Maxwellian distribution.

3.2.2 Angular Discretization

The angular phase space is discretized using the discrete ordinates method [1]. The unit sphere is divided into discrete ordinates, or directions, with appropriate weights.

We now represent the scalar flux as a quadrature integration of the angular fluxes

$$\phi_g(\vec{r}) = \sum_{n=1}^{N} \omega_n \psi_{g,n}(\vec{r}). \tag{3.10}$$

We use Gauss-Legendre quadrature points for our 1D problems and level symmetric [32] quadrature points for our 2D problems. An example of 1D Gauss-Legendre is represented in Figure 3.2. An example of level symmetric quadrature is shown in Figure 3.3. Level symmetric quadrature points have the advantage of being rotationally invariant for 90°, but are disadvantaged by only having one degree of freedom associated with the arrangement of latitudes.

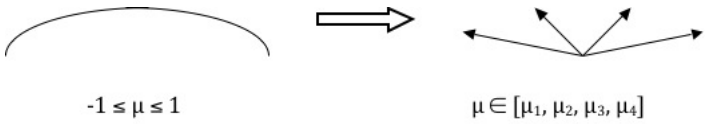

Figure 3.2: 1D Discrete Ordinate Gauss-Legendre Example

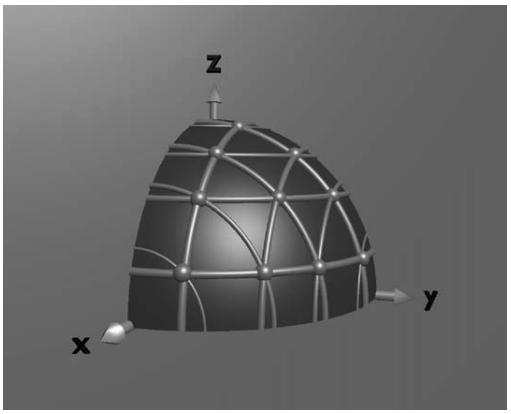

Figure 3.3: 2D Level Symmetric Example [32]

The resulting discretized transport equation, with assumed isotropic angular dis-

tribution, is written as follows for the fixed source problem

$$\vec{\Omega_n} \cdot \nabla \psi_{g,n}(\vec{r}) + \sigma_{t_g}(\vec{r})\psi_{g,n}(\vec{r}) = \tag{3.11}$$
$$\frac{1}{4\pi} \sum_{g'=0}^{G} \sigma_{s_{0_{g'\to g}}}(\vec{r}) \sum_{n=1}^{N} \omega_n \psi_{g',n}(\vec{r}) + \frac{1}{4\pi} Q_g(\vec{r}),$$

and for the k-eigenvalue problem as

$$\vec{\Omega_n} \cdot \nabla \psi_{g,n}(\vec{r}) + \sigma_{t_g}(\vec{r})\psi_{g,n}(\vec{r}) = \tag{3.12}$$
$$\frac{1}{4\pi} \sum_{g'=0}^{G} \sigma_{s_{0_{g'\to g}}}(\vec{r}) \sum_{n=1}^{N} \omega_n \psi_{g',n}(\vec{r}) +$$
$$\frac{1}{4\pi k} \sum_{g'=0}^{G} \nu_g \sigma_{f_{g'\to g}}(\vec{r}) \sum_{n=1}^{N} \omega_n \psi_{g',n}(\vec{r}),$$

where n is the ordinate subscript and ω_n is the associated quadrature weight.

3.2.3 Spatial Discretization

Spatial discretization is handled using the linear discontinuous finite element method applied to structured and unstructured meshes. The linear discontinuous finite element method (LDFEM) represents the scalar flux and angular flux in each element k as a sum of linearly independent basis functions. It also allows for discontinuity at the volume element boundaries. A 1D example of this is shown in Figure 3.4.

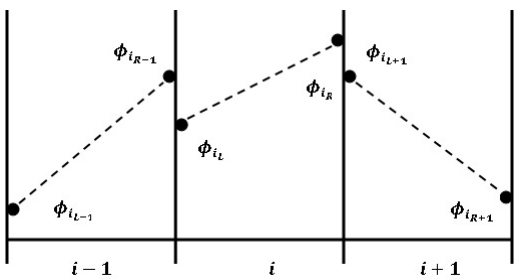

Figure 3.4: 1D Linear Discontinuous Finite Element Method Example

We will start the description of the LDFEM spatial discretization in a general spatial coordinate system by writing the S_N equations in the form

$$\hat{\Omega}_m \cdot \nabla \psi_m + \sigma_t \psi_m(\vec{r}) = S_m(\vec{r}) \tag{3.13}$$

where $S_m(\vec{r})$ represents the scattering source plus inhomogeneous sources.

The angular flux is first expanded $\psi_m(\vec{r})$ in a nodal basis on an p–node element $\mathbb{E}$

$$\widehat{\psi}_m(\vec{r}) = \sum_{j=1}^{p} \psi_{m,j} B_j(\vec{r}), \quad \vec{r} \in \mathbb{E} \backslash \partial\mathbb{E}. \tag{3.14}$$

To introduce the upwind discontinuous approximation, we first define the indexing function for the k^{th} "face" of an element, $\partial\mathbb{E}_k$,

$$v(k) = \{j \mid \vec{r}_j \in \partial\mathbb{E}_k\}, \tag{3.15}$$

that is, $v(k)$ is the set of all vertices on $\partial\mathbb{E}_k$. The angular flux on k is upwinded

$$\widetilde{\psi}_m^{(k)}(\vec{r}) = \sum_{v(k)} \begin{cases} \psi^{\text{B}}_{m,v(k)} B_{v(k)}(\vec{r}), & \vec{r} \in \Gamma \backslash \partial\Gamma \\ \psi^{\text{INC}}_{m,v(k)} B_{v(k)}(\vec{r}), & \vec{r} \in \partial\Gamma \end{cases} \tag{3.16a}$$

where

$$\psi^{\text{B}}_{m,i} = \begin{cases} \psi_{m,i}, & (\hat{n}_k \cdot \hat{\Omega}_m) > 0, \\ \psi_{m,i(k)}, & (\hat{n}_k \cdot \hat{\Omega}_m) < 0, \end{cases} \tag{3.16b}$$

and $\psi^{\text{INC}}_{m,i}$ is specified by boundary conditions for the problem domain Γ at vertex i, to define the discontinuous approximation. The expression $i(k)$ refers to the index of the vertex in the element sharing face k across from vertex i. With N_f being the number of faces of an element such that

$$\partial\mathbb{E} = \bigcup_{k=1} \partial\mathbb{E}_k.$$

the weak form of (3.13) is then constructed for each basis function $B_i(\vec{r})$

$$\sum_{k=1}^{N_f} \int_{\partial \mathbb{E}_k} (\hat{n}_k \cdot \hat{\Omega}_m) B_i(\vec{r}) \widetilde{\psi}_m^{(k)}(\vec{r}) dA - \int_{\mathbb{E}} \left(\hat{\Omega}_m \cdot \nabla B_i(\vec{r}) \right) \widehat{\psi}_m(\vec{r}, \hat{\Omega}) dV \\ + \sigma_t(\vec{r}) \int_{\mathbb{E}} B_i(\vec{r}) \widehat{\psi}_m(\vec{r}) dV = \int_{\mathbb{E}} B_i(\vec{r}) \widehat{S}_m(\vec{r}) dV, \tag{3.17}$$

where $S_m(\vec{r})$ has been expanded in the same nodal basis as $\psi_m(\vec{r})$, that is,

$$\widehat{S}_m(\vec{r}) = \sum_{j=1}^{p} \widehat{S}_{m,j} B_j(\vec{r}). \tag{3.18}$$

We can now write the fully discrete equations corresponding to (3.17) for Carteisian coordinates in matrix form for the vector of angular fluxes Ψ_m in an element $\mathbb{E}$, and the angular fluxes on face k, $\widetilde{\Psi}_m^{(k)}$, which may be known from from neighboring cells sharing face k or from the boundary conditions depending on the upwinding relation (3.16), and the vector of source coefficients S as follows.

$$\sum_{k=1}^{N_f} \left(\hat{\Omega}_m \cdot \overline{\mathbf{N}}^{(k)} \right) \widetilde{\Psi}_m^{(k)} - \left(\hat{\Omega}_m \cdot \overline{\mathbf{L}} \right) \Psi_m + \sigma_t \mathbf{M} \Psi_m = \mathbf{M} S. \tag{3.19}$$

where σ_t is the total cross section in the element. Note that the bar over $\overline{\mathbf{N}}^{(k)}$ and $\overline{\mathbf{L}}$ is to indicate that they have components in each geometric dimension.

Altogether we can write the $(p \times p)$ operators appearing in both (3.19) as follows. They are given for a given row i and column j, or column $v(k)$ for face k, by

$$\overline{\mathbf{N}}_{i,v(k)}^{(k)} = \int_{\partial \mathbb{E}_k} \hat{n}_k B_i(\vec{r}) B_{v(k)}(\vec{r}) dA, \tag{3.20}$$

$$\overline{\mathbf{L}}_{i,j} = \int_{\mathbb{E}} \left[\nabla B_i(\vec{r}) \right] B_j(\vec{r}) dV, \tag{3.21}$$

and

$$\mathbf{M}_{i,j} = \int_{\mathbb{E}} B_i(\vec{r}) B_j(\vec{r}) dV. \tag{3.22}$$

The LDFEM spatial discretization is third order accurate for 1D problems and special cases of higher dimensional problems. LDFEM is more stable in handling sharp derivatives of the scalar flux because it solves the problem in the weak sense. A disadvantage with respect to central differencing methods is that the memory requirement is increased by the number of basis function nodes required for the specified finite element. Even the simplest 1D bar element requires a doubling in the number of spatial solution points. In this dissertation, we use a uniform structured bar mesh for 1D geometry. When we run problems in 2D geometry, we use an unstructured mesh with either triangle, quadrilateral, or hexagon elements.

3.3 Numerical Solution Methods

Once the transport model has been discretized, the scalar flux (and k-effective for eigenvalue problems) are computed iteratively. In operator notation, the fixed source problem is

$$L\psi = MSD\psi + Q, \tag{3.23a}$$

$$\psi = L^{-1}MSD\psi + L^{-1}Q, \tag{3.23b}$$

$$D\psi = DL^{-1}MSD\psi + DL^{-1}Q, \tag{3.23c}$$

$$\phi = DL^{-1}MS\phi + DL^{-1}Q, \tag{3.23d}$$

$$[I - DL^{-1}MS]\phi = DL^{-1}Q, \tag{3.23e}$$

$$\tag{3.23f}$$

and the k-eigenvalue problem is

$$L\psi = MSD\psi + \tfrac{1}{k}MFD\psi, \tag{3.24a}$$
$$\psi = L^{-1}MSD\psi + \tfrac{1}{k}L^{-1}MFD\psi, \tag{3.24b}$$
$$D\psi = DL^{-1}MSD\psi + \tfrac{1}{k}DL^{-1}MFD\psi, \tag{3.24c}$$
$$\phi = DL^{-1}MS\phi + \tfrac{1}{k}DL^{-1}MF\phi, \tag{3.24d}$$
$$[I - DL^{-1}MS]\phi = \tfrac{1}{k}DL^{-1}MF\phi, \tag{3.24e}$$

where $\phi = D\psi$. For large scale scientific problems, exact inversion would be impractical. Direct matrix inversion methods, such as Gaussian elimination, are historically of the order N^3 in computation cost (where N is the dimension of the matrix) [50]. Recent research in direct sparse solvers uses hierarchically applied condensation of internal degrees of freedom for finite element problems to reduce the order of operations substantially [10]. These newer direct sparse solvers have a setup cost of order $N^{\frac{3}{2}}$, a memory storage cost of order $N\log(N)$ and a solve cost of $N\log(N)$.

In this dissertation, we use matrix-free iterative solution methods. Iterative solvers can reduce the computational cost to as low as order N through proper preconditioning and taking advantage of matrix sparsity because only the action of an operator is required. They are also the more widely used solution methods for computing a solution for the scalar flux from the radiation transport model.

In the following subsections we will outline the iterative methods associated with the fixed internal source transport problem and the k-eigenvalue problems. The simplest technique for both problems is the well known linear fixed point iteration, known in the nuclear engineering community as source iteration [1]. Other popular methods are Krylov methods [53] and Newton type methods [21, 42].

3.3.1 Fixed Internal Source Problem

This subsection outlines two iterative algorithms used to converge a scalar flux solution to the radiation transport problem with a fixed internal source and no fission

$$Ł\psi = MSD\psi + Q. \tag{3.25}$$

The first is the well known source iteration method. The second method outlines the use of a Krylov solver to invert the transport operator.

Source Iteration

Source iteration is the most widely known method of computing a scalar flux solution to the transport equation. It is very simple to implement, but has slow convergence in the thick-diffusive regime. In thick-diffusive regimes, false convergence becomes a real issue [1]. Basic source iteration is shown in operator notation as

$$\phi^{k+1} = DL^{-1}MS\phi^{k} + DL^{-1}Q, \tag{3.26}$$

where Q is the prescribed fixed internal source and L^{-1} represents a transport sweep. A transport sweep works by starting at a known boundary value of the angular flux and then solving the finite element system of equations in each cell in each discrete direction Ω until all angular fluxes have been calculated. No direct matrix inversion is required. This source iteration is guaranteed to converge linearly when the largest eigenvalue of $[DL^{-1}MS] < 1$.

Krylov Solver Fixed Point Iteration

An alternative approach is to use a Krylov iterative solver. When used to solve a linear system of the form $Ax = b$, an approximate solution is constructed from a

linear combination of vectors forming the Krylov space [28]

$$\kappa(A, c) \equiv span\{r_0, Ar_0, A^2 r_0, \ldots, A^{N-1} r_0\}, \tag{3.27}$$

where $r_0 = b - Ax_0$. The dimension of the Krylov subspace is equal to the degree of the minimal polynomial of A [28]. If the minimal polynomial of A is small, then Krylov methods will quickly converge a solution. Krylov methods are particularly useful when the matrix A cannot be explicitly represented, or is very expensive to compute. Instead of requiring an explicit formulation of A, only the action of A on a vector is required. This can be readily accomplished in computer codes via functions/methods that represent the action of the operator.

In this dissertation we use restarted GMRES [47] and restarted GCRODR [43] as our Krylov solvers. GCRODR is a Krylov subspace projection method based on GMRES that is intended for solving a sequence of related linear systems. By recycling portions of selected subspaces, GCRODR should be more efficient for our algorithm than traditional GMRES, because our linear systems are related through a homotopy parameter λ. Restarted-GCRODR also recycles a portion of the subspaces generated during the inner iterations between restarts as well, leading to further improvement in overall efficiency compared to restarted-GMRES.

The convergence rate of both restarted GMRES is related to the minimal polynomial and the eigenvalue clusters of the matrix A. Campbell et. al. [12] show that the convergence rate is based on the size of the relative radius of the eigenvalue clusters. In particular, if there is one single dominant cluster of eigenvalues with few outliers, the convergence factor is the relative radius of the primary cluster. If there are multiple eigenvalue clusters, then GMRES treats them as a single large cluster with a convergence factor equal to this compound cluster radius. Eigenvalues near zero have a large relative radius and are more difficult to converge.

By representing the individual operators from Eqs. (3.23) and (3.24) as matrix-

vector products, we can formulate the Boltzmann operator $[I - DL^{-1}MS]$ and use a Krylov solver to invert against the right hand side. An alternative formulation involves splitting the scattering operator into lower, diagonal, and upper portions and re-formulating our transport problems in the following way

$$[I - DL^{-1}MS_{LD}]\phi = DL^{-1}M(S_U)\phi + DL^{-1}Q, \tag{3.28}$$

and

$$[I - DL^{-1}MS_{LD}]\phi = DL^{-1}M(S_U)\phi + \frac{1}{k}DL^{-1}MF\phi, \tag{3.29}$$

where the up-scattering component is lagged in the iteration. Eq. (3.29) is traditional power iteration in the nuclear engineering community. We can also move everything to the right hand side and lag all scattering contribution

$$\phi = DL^{-1}MS\phi + \frac{1}{k}DL^{-1}MF\phi. \tag{3.30}$$

This is called the flattened formulation [21] and is the default formulation for k-eigenvalue problems in the Capsaicin code project developed at Los Alamos National Laboratory.

A Krylov iterative solution can be accelerated through the use of preconditioning. This is accomplished by choosing an operator, M, that is an approximation to A. Then we invert it against both sides of the linear system

$$M^{-1}Ax = M^{-1}b, \tag{3.31}$$

with the goal that $M^{-1}A \approx I$. The previously mentioned acceleration methods for source iteration can be recast as operators and applied with these types of solvers.

3.4 k-Eigenvalue Formulation

A k-eigenvalue problem is a steady state radiation transport problem where there is a self multiplying material present and no fixed internal source. We represent this

eigenvalue problem in operator notation as

$$A\phi = \frac{1}{k}B\phi, \tag{3.32}$$

where B represents the composite fission source operator with moment-to-discrete (M) and discrete-to moment (D) already incorporated, and A is all the other physics (diffusion, scattering, etc.). A typical k-eigenvalue search algorithm involves first guessing an initial eigenpair, (ϕ^0, k^0). Then the eigenvector for the n^{th} iterate is converged in the following way

$$\phi^{n+1} = \frac{1}{k^n}A^{-1}B\phi^n. \tag{3.33}$$

Once a scalar flux is converged, the k-eigenvalue is updated commonly using the fission rate update

$$k^{n+1} = k^n \frac{\int d^3rF\phi^{n+1}}{\int d^3rF\phi^n}, \tag{3.34}$$

where F is the fission operator from Eq. (3.4).

The eigenpair is iterated on until a prescribed convergence criteria is met. This is known in the nuclear engineering community as inverse power iteration. The iteration is slow to converge when the dominance ratio of the problem is close to 1.0. The dominance ratio, σ is defined as:

$$\sigma = \frac{\|k_2\|}{\|k_1\|}, \tag{3.35}$$

where $\|k_2\|$ and $\|k_1\|$ are the two greatest eigenvalues for the k-eigenvalue problem and $\|k_1\| \geq \|k_2\|$. There are a variety of methods used to accelerate convergence of the k-eigenvalue problem when a high dominance ratio is present, including the Wieland shift method [46], Chebyshev acceleration [45], nonlinear krylov acceleration [11], and a nonlinear method developed by Park et. al. [42] that combines nonlinear diffusion acceleration [30, 48] and nonlinear criticality acceleration [20, 29].

3.4.1 Newton Method

Recently the Newton-Raphson method has been used to solve the k-eigenvalue problem as a fully coupled nonlinear problem [21, 42]. We define the k-eigenvalue transport residual as

$$R(\phi, k) = \phi - P(k)\phi. \tag{3.36}$$

The coupled system is represented as

$$\Xi(\phi, k) = \begin{pmatrix} R(\phi, k) \\ \kappa(\phi, k) \end{pmatrix}, \tag{3.37}$$

where $P(k)$ is the chosen operator formulation for the k-eigenvalue transport model. In this dissertation we use the flattened formulation, $P(k) = [DL^{-1}MS + \frac{1}{k}DL^{-1}MF]$. Various constraint equation are discussed by Gill et. al. [21]. We choose the constraint equation $\kappa(\phi, k)$ to be the the fission rate update from Equation (3.34)

$$\kappa(\phi, k) = k - k\frac{E^T F P(k)\phi}{E^T F \phi}, \tag{3.38}$$

where E^T is a vector of ones to represent the integral over the volume domain on our discretized mesh. The Jacobian for this problem is

$$J = \begin{pmatrix} R_\phi(\phi, k) & R_k(\phi, k) \\ \kappa_\phi(\phi, k) & \kappa_k(\phi, k) \end{pmatrix}, \tag{3.39}$$

where

$$R_\phi(\phi, k) = I - P(k), \tag{3.40a}$$

$$R_k(\phi, k) = -\tfrac{1}{k^2}DL^{-1}MF\phi, \tag{3.40b}$$

$$\kappa_\phi(\phi, k) = \tfrac{(E^T F\phi)(E^T FP(k)) - (E^T FP(k)\phi)(E^T F)}{(E^T F)^2}, \tag{3.40c}$$

$$\kappa_k(\phi, k) = 1 - \tfrac{E^T FDL^{-1}MS\phi}{E^T F\phi}. \tag{3.40d}$$

The Newton algorithm begins by choosing an initial guess, (ϕ_0, k_0), and inverting the Jacobian matrix against the residual to compute a step correction

$$\begin{pmatrix} R_\phi(\phi^i, k^i) & R_k(\phi^i, k^i) \\ \kappa_\phi(\phi^i, k^i) & \kappa_k(\phi^i, k^i) \end{pmatrix} \begin{pmatrix} \delta\phi^i \\ \delta k^i \end{pmatrix} = - \begin{pmatrix} R(\phi^i, k^i) = \phi - P(k^i)\phi^i \\ \kappa(\phi^i, k^i) \end{pmatrix}, \tag{3.41}$$

where the index, i, is the Newton iteration index. The solution is updated in the Newton iteration

$$\phi^{i+1} = \phi^i + \delta\phi^i, \tag{3.42a}$$

$$k^{i+1} = k^i + \delta k^i, \tag{3.42b}$$

and is continued until the L_2-norm of $\Xi(\phi^{i+1}, k^{i+1})$ is less than the prescribed tolerance for the problem. For practical large scale scientific problems it is untenable to compute and store the Jacobian matrix (3.39) explicitly. Instead we use the Jacobian-Free Newton Krylov method as outlined in Section 2.1.5.

Chapter 4

Homotopy in Thick-Diffusive Fixed Source Problems

4.1 Overview

In this section we show the derivation of the Diffusion Length Preserving Continuation (DLPC) and show how the respective natural parameter homotopy algorithm is constructed. The method is an inverted asymptotic scaling from that derived by Larsen [34, 35] and is similar to the stretch-filtered Transport Synthetic Accleration in Hanshaw, et, al. [23, 24]. In this dissertation, the DLPC method is applied to problems that are thick-diffusive. We also detail how to construct artificial homotopy formulations using diffusion and S_2 isotropic imbeddings. These imbeddings are revisited in the k-eigenvalue problems of chapter 5. All problems are simulated through the Capsaicin code project developed at Los Alamos National Laboratory.

4.2 Diffusion Length Preserving Continuation

We begin our homotopy derivation by stating the definition of the diffusion length

$$L \equiv \sqrt{(D/\sigma_a)}, \tag{4.1}$$

where D is the diffusion coefficient. The diffusion coefficient is defined as

$$D = [3(\sigma_t - \bar{\mu_0}\sigma_s)]^{-1}, \tag{4.2}$$

where $\bar{\mu_0}$ is the average cosine of the scattering angle in a neutron collision and σ represents the total and scattering cross sectional data. In a thick-diffusive regime, we can assume a largely isotropic scattering distribution such that

$$D = [3(\sigma_t]^{-1}. \tag{4.3}$$

This approximation loses accuracy near sharp heterogenous material boundaries. In a thick-diffusive regime, our diffusion length becomes

$$L = \sqrt{\frac{1}{3\sigma_t\sigma_a}}. \tag{4.4}$$

We now introducing our asymptotic scaling parameter, η, and scale our cross sectional data and internal source in the following manner

$$\tilde{\sigma}_t = \eta\sigma_t, \tag{4.5}$$

$$\tilde{\sigma}_a = \tfrac{\sigma_a}{\eta}, \tag{4.6}$$

$$\tilde{Q} = \tfrac{Q}{\eta}, \tag{4.7}$$

where σ denotes the cross sectional data and Q is the internal source. We scale the total group scattering cross section as

$$\tilde{\sigma}_s = \tilde{\sigma}_t - \tilde{\sigma}_a. \tag{4.8}$$

For multi-group problems, the group-to-group scattering cross sections are scaled respective to the original ratios of the total group scattering cross section

$$\tilde{\sigma}_{s_{g'\to g}} = \sigma_{s_{g'\to g}} \frac{\tilde{\sigma}_{t_g} - \tilde{\sigma}_{a_g}}{\sigma_{t_g} - \sigma_{a_g}}. \tag{4.9}$$

We construct our homotopy mapping as a natural parameter continuation where the natural parameters being scaled are the cross sectional data. Our initial state is chosen to be a scaled purely absorbing problem. The corresponding η for this is $\eta^* = \frac{\sqrt{\sigma_a \sigma_t}}{\sigma_t}$. This homotopy formulation preserves the thick-diffusive diffusion length throughout the entire deformation.

We define our Diffusion Length Preserving Continuation (DLPC) as a natural parameter homotopy of the following form

$$H(\phi, \eta) = A(\eta)\phi - b(\eta), \tag{4.10}$$

where $\eta \in [\eta^*, 1]$, $A(\eta) \equiv [I - DL_\eta^{-1} M S_\eta]$, and $b(\eta) \equiv DL_\eta^{-1} M Q_\eta$. When $\eta = \eta^*$ the problem is purely absorbing and can be solved without iteration, requiring only a single transport sweep. When $\eta = 1$ we recover the original difficult (thick-diffusive) problem. We show a 1D numerical DLPC mapping using MATLAB in figure 4.1. The mapping is smooth, except on the boundaries where the diffusion length is no longer preserved.

Typically, one will use the uncollided flux as the initial guess for the iterative solution process. We label our "reference solution" as that which uses the uncollided flux as the initial guess and does not utilize homotopy. In this chapter, we measure effectiveness of our homotopy methods in terms of the total number of global function evaluations. Unless otherwise specified, all homotopy formulations are direct tracings with respect to the homotopy parameter. We also use the numerical secant predictor from Eq. (2.19) in all problems unless otherwise specified. The DLPC homotopy algorithm is given in Algorithm 3.

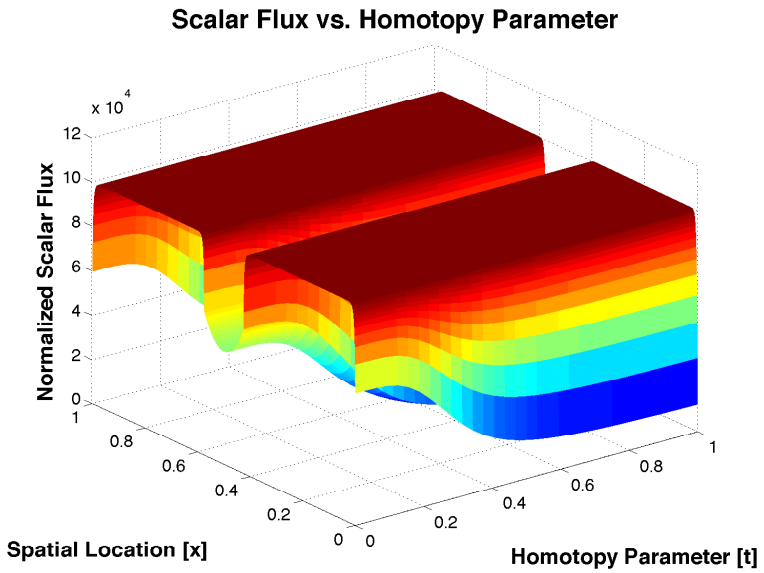

Figure 4.1: DLPC 1D Slab Mapping for 2-group problem

Algorithm 3 Diffusion Length Preserving Continuation

Given ϕ_0 and $H(\phi, \eta(t))$

for $n = 0, \ldots, N$ **do**

$t_n = \frac{n}{N}$

$\eta_n(t) = (1 - t_n)\eta^* + t_n$

Deform Data: $[\sigma(\eta), Q(\eta)]$

Scale Tolerance: Tol $= (1 - t^4)(Itol) + t^4(Ftol)$

Correct: minimize $H(\phi_n)$ with initial guess ϕ_0

Calculate Tangent: $\vec{t} = \frac{\phi_n - \phi_{n-1}}{\eta_n - \eta_{n-1}}$

Predict: $\tilde{\phi}_n = \phi_n + \Delta\eta_n\vec{t}$

$\phi_0 = \tilde{\phi}_n$

end for

4.3 Artificial Parameter Continuation

We also construct an artificial homotopy parameter formulation. The generalized expression of the artificial parameter homotopy is given as

$$H(x, \lambda) = (1 - \lambda)G(\phi) + \lambda F(\phi), \tag{4.4}$$

where $\lambda \in [0,1]$ and $F(\phi)$ is our original thick-diffusive problem. We present two imbeddings for G, a diffusion and an S_2 coherent isotropic approximation of our original problem, F. The specific algorithm for this artificial parameter formulation is presented in Algorithm 4.

Algorithm 4 Artificial Parameter Homotopy Continuation

Given ϕ_0, $G(\phi)$, and $F(\phi)$

for $n = 0, \ldots, N$ **do**

$\lambda_n = \frac{n}{N}$

Scale Tolerance: Tol $= (1-\lambda_n^4)(Itol) + \lambda_n^4(Ftol)$

Correct: minimize $H(\phi_n) = (1-\lambda_n)G(\phi_n) + \lambda_n F(\phi_n)$ with initial guess ϕ_0

Calculate Tangent: $\vec{t} = \frac{\phi_n - \phi_{n-1}}{\lambda_n - \lambda_{n-1}}$

Predict: $\tilde{\phi}_n = \phi_n + \Delta\lambda_n \vec{t}$

$\phi_0 = \tilde{\phi}_n$

end for

4.3.1 Diffusion Imbedding

In our diffusion imbedding, we solve the steady state multi-group diffusion equation for the given problem specifications. This diffusion equation is expressed as

$$G(\phi) = -\nabla \cdot D_g(\vec{r})\nabla\phi_g(\vec{r}) + \sigma_{t_g}(\vec{r})\phi_g(\vec{r}) - \sum_{g'=1}^{G} \sigma_{s_{g'\to g}}(\vec{r})\phi_{g'}(\vec{r}) + Q_g(\vec{r}) = 0, \quad (4.12)$$

where D_g is our multi-group diffusion coefficient. Extra interpolation and extrapolation steps need to be performed when passing the scalar flux between the diffusion imbedding and the LDFE transport problem. This is due to the diffusion solution being solved on the cell average rather than at the LDFE corner nodes. An example figure showing the different solution locations is given in Figure 4.2. The diffusion imbedding presents an added complexity to computational cost because the

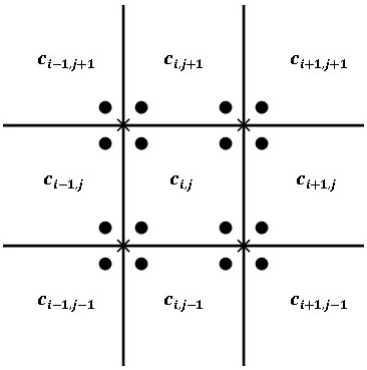

Figure 4.2: Example of Diffusion [×] vs. Transport [•] Solution Locations

diffusion operator needs to be inverted with a Krylov solver. Because we desire a trivial problem solution, we restrict our Krylov tolerance to 1.0E-02 when inverting the diffusion operator. For all instances of the diffusion imbedding, we use GMRES as our Krylov solver.

4.3.2 S_2 Coherent Isotropic

We construct a reduced quadrature formulation for our transport problem where we use an S_2 quadrature. We also extract the coherent isotropic scattering data from the full problem and use only the within group scattering contribution for our reduced quadrature problem. This leads to a scattering matrix that is diagonal. We call this the S_2-Coherent Isotropic imbedding where the embedded problem is

$$G(\phi) = L_{S_2}\psi - M\tilde{S}D\psi + Q = 0, \tag{4.13}$$

where L_{S_2} represents our transport operator with the associated discrete ordinates for S_2 and $\tilde{S}$ represents the coherent isotropic scattering matrix. In our results, we often denote this imbedding as S_2-Coherent.

4.4 Results for 1D-Slab Problems

The first problem we examine is a non-dimensionalized 1D-slab with one energy group. We examine both a homogeneous and heterogeneous case. The homogeneous case uses only material 1 (thick material). The heterogeneous case inserts a portion of material 2 (thin material) from $x = 0.4$ to $x = 0.6$.

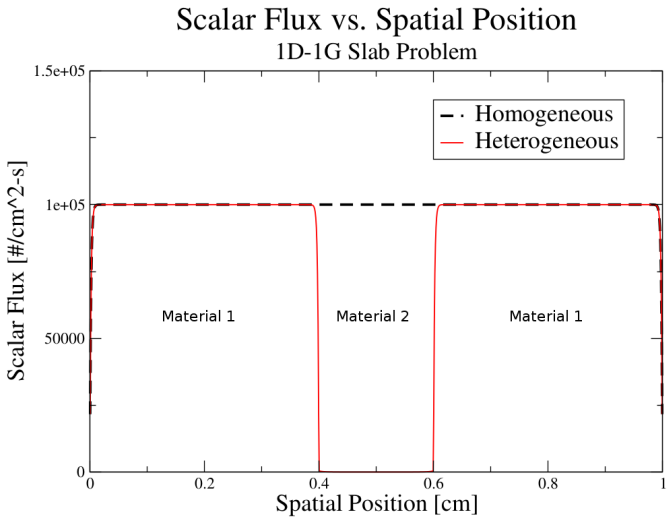

Figure 4.3: Scalar Flux Results for 1D Slab Problem using Homotopy Continuation

The solution profiles for these problems are shown in Fig. 4.6. We give the numerical problem specifications in Table 4.1. In all our numerical problems we utilize the linear discontinuous finite element method (LDFE). Because the problem is non-dimensional, the thickness of the slab is determined by the length, the material density, and the scattering ratio. The scattering ratio, c, is defined as $c = \sigma_s/\sigma_t$.

The problem is solved both without preconditioning and with diffusion synthetic acceleration (DSA) preconditioning using three different homotopy formulations: DLPC, S2-Coherent, and Diffusion. The results are shown in figure 4.5. The DSA

version we use only operates within the respective energy groups and doesn't couple the whole spectrum together. Our reference cases uses the uncollided flux as the initial guess for the Krylov solver. We set the initial tolerance to be looser than the desired final tolerance. The tolerance is scaled as the continuation progresses in the following way

$$\text{tol}(t) = (1 - t^4)(\text{tol}_0) + t^4(\text{tol}_f), \tag{4.14}$$

where tol_0 is the prescribed initial tolerance and tol_f is the desired final tolerance. This is plotted in Figure 4.4 with respect to η for a problem with a scattering ratio of 1.0 at $t = 1.0$.

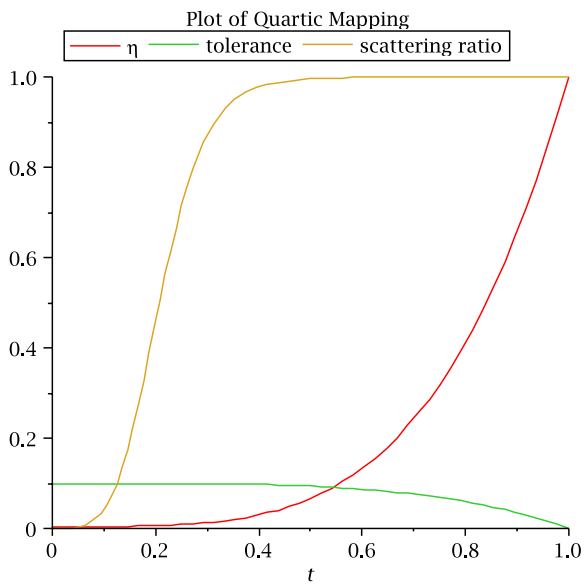

Figure 4.4: Tolerance Path Mappings

We use direct homotopy parameter tracing where we *a priori* discretize our solution path into intervals respective to the homotopy parameter. In the DLPC formulation, η is discretized uniformly from $\eta = \eta^*$ to $\eta = 1.0$. The artificial parameter construction is discretized uniformly with respect to λ, where we begin with

$\lambda = 0$ and end with $\lambda = 1$. Throughout this dissertation we will refer to a homotopy interval. We define a homotopy interval as the interval between discretized points of λ. Thus, one homotopy interval corresponds to using homotopy continuation simply as an alternative initial guess.

We observe that homotopy continuation results in a slight reduction in iteration count for the un-preconditioned solution. However, DSA is far more effective for improving convergence. Even coupling DSA and DLPC together does not yield better results than DSA on its own. DLPC performs better as a homotopy for the heterogeneous problem while S2-Coherent and diffusion perform better for the homogeneous case.

Table 4.1: 1D-Slab Geometric and Angular Discretization for Fixed Source Problem

Parameter	Value
Length	1.0
Cells	1000
Nodes	1001
Regions	1
Boundaries	Vacuum
Spatial Discretization	LDFEM
Quadrature	Gauss-Legendre S_8

Table 4.2: 1D-Slab Numerical Solver Parameters for Fixed Source Problem

Parameter	Value
Solver	GCRODR Belos
Restart	30
Max Iterations	10000
Final Tolerance	1.0×10^{-8}
Initial Tolerance	1.0×10^{-2}
Predictor	SECANT

Table 4.3: 1D-Slab Material Data for Fixed Source Problem

Parameter	Material 1	Material 2
σ_t	1.0	1.0
c	0.99999	0.5
ρ	1.0×10^5	1.0×10^2
Q	1.0×10^5	1.0×10^2

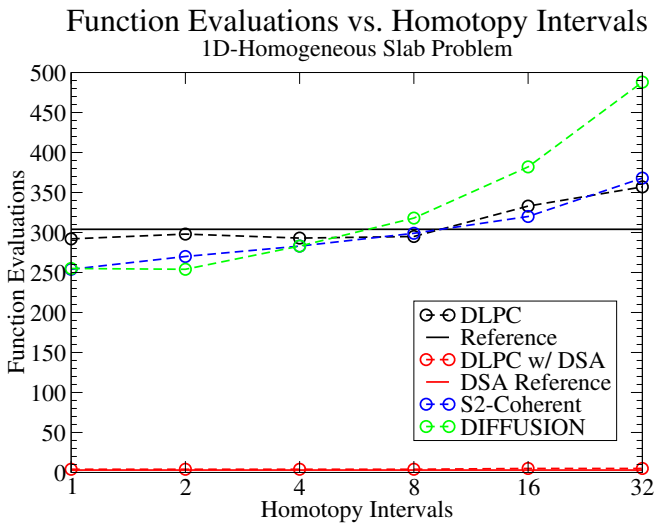

Figure 4.5: Function Evaluation Results for 1D-Homogeneous Slab Problem using Homotopy

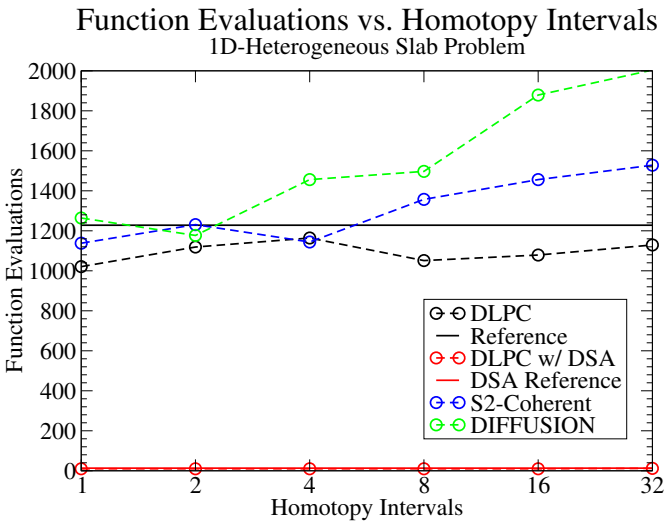

Figure 4.6: Function Evaluation Results for 1D-Heterogeneous Slab Problem using Homotopy

4.4.1 Results for 2-Group 2D Problem

We investigate a 2D heterogeneous problem as depicted in Figure 4.7. We use two energy groups and two different materials. On material is defined at the center of the problem and the other is defined in the surrounding box. We define our thickest material to be the surrounding box and our thinner material to be the central portion. Function evaluation results are reported for the DLPC homotopy method with and without DSA preconditioning. We also show results for the S2-Coherent Isotropic and Diffusion artificial homotopy formulations.

The random cross section data is generated using the algorithm by Rosa, et, al. [44] to achieve a high scattering ratio. The resulting cross sections are given in Table 4.4. The spatial discretization is an unstructured hexagonal linear discontinuous finite element mesh. The problem is solved in parallel with 48 processors using the Moonlight cluster at Los Alamos National Laboratory. Once again, the problem is non-dimensionalized and we measure effectiveness in terms of functional evaluations.

Table 4.4: Cross Section Data for 2D-Heterogeneous Fixed Source Problem

Parameter	Material 1	Material 2
Group 1 velocity	2.2×10^5	2.2×10^5
Group 2 velocity	4.7×10^8	4.7×10^8
Group 1 Energy Bin	0.0-2.4 [MeV]	0.0-2.4 [MeV]
Group 2 Energy Bin	2.4-17.0 [MeV]	2.4-17.0 [MeV]
σ_{t_1}	1.5454	1.3766992
σ_{t_2}	4.5468×10^{-1}	6.433007×10^{-1}
$\sigma_{s_{1\to1}}$	6.1789×10^{-1}	8.65153×10^{-1}
$\sigma_{s_{1\to2}}$	9.2747×10^{-1}	4.979156×10^{-1}
$\sigma_{s_{2\to1}}$	3.8211×10^{-1}	1.34847×10^{-1}
$\sigma_{s_{2\to2}}$	7.2534×10^{-2}	5.020844×10^{-1}
ρ	1.0×10^5	1.0×10^2

Table 4.5: Geometric Parameters for 2D-Heterogeneous Fixed Source Problem

Parameter	Value
Side Lengths	1.0
Cells	2529
Nodes	4864
Regions	2
Boundaries	Vacuum
Spatial Discretization	Unstructured Hexagon LDFEM
Quadrature	Level Symmetric S_8

In Figure 4.8 we distribute a fixed internal source of $Q_1 = 1.0 \times 10^5$ uniformly throughout the outer material (green) and a fixed internal source of $Q_2 = 1.0 \times 10^2$ uniformly throughout the central material (purple). While this is a problem where DSA is not helping, neither are any of the homotopy formulations. The best results were achieved when solving without DSA and without any homotopy application. In Figure 4.9 we switch the internal source distribution such that the outer material

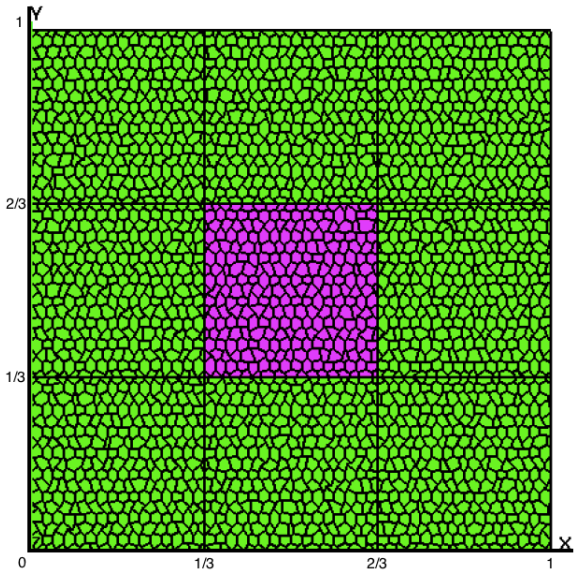

Figure 4.7: Problem Geometry for 2D Heterogeneous Problems

Table 4.6: Numerical Solver Parameters for 2D-Heterogeneous Fixed Source Problem

Parameter	Value
Solver	GCRODR Belos
Restart	30
Max Iterations	10000
Final Tolerance	1.0×10^{-8}
Initial Tolerance	1.0×10^{-1}
Predictor	SECANT

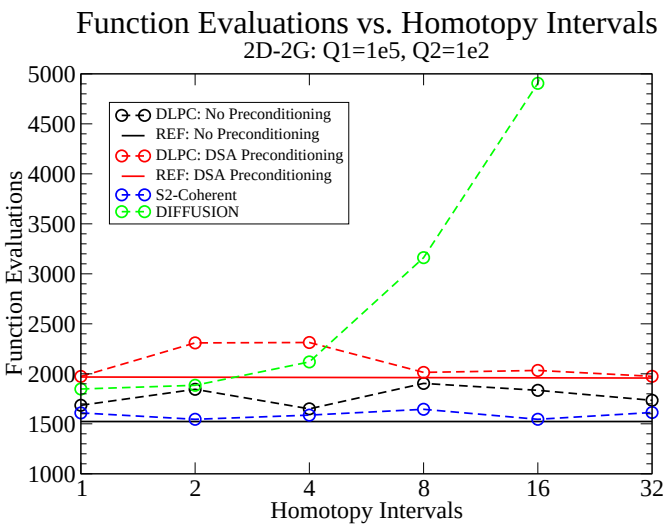

Figure 4.8: Convergence Results for Internal Sources of Q1=1.0×10^5 and Q2=1.0×10^2

internal source is $Q_1 = 1.0 \times 10^2$ and the inner material internal source is $Q_2 = 1.0 \times 10^5$. We observe that DSA again degrades for this problem specification, but now homotopy continuation is improving convergence speed by a slight amount. Using only one homotopy interval, which can be viewed simply as an initial guess, the S2-Coherent Isotropic and DLPC homotopy formulations converge slightly faster than the reference case. DLPC with and without DSA continues to be effective even for 32 homotopy intervals. T In Figure 4.10 we fix both internal sources for the two

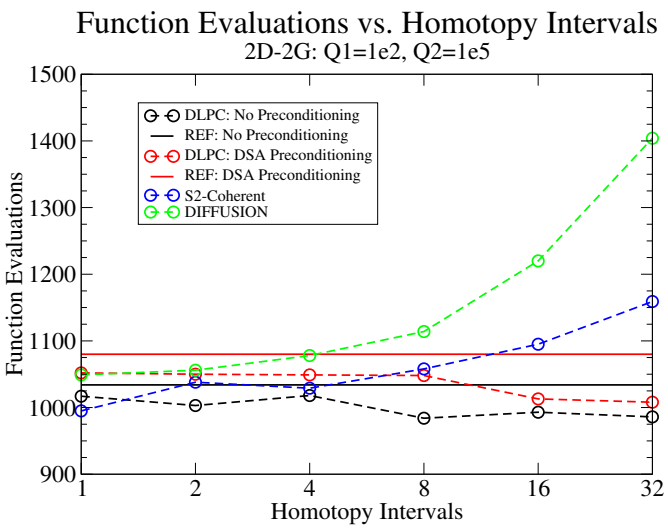

Figure 4.9: Convergence Results for Internal Sources of Q1=1.0×10^2 and Q2=1.0×10^5

materials to be uniformly distributed $Q_1 = Q_2 = 1.0 \times 10^5$. Once more we see the degraded effectiveness of DSA. As with Figure 4.8, the homotopy formulations do not provide any gain in convergence speed beyond the reference case.

Homotopy continuation appears to be beneficial for problems that have a strong internal source in thin regions and a weaker internal source in thick regions. A practical physical case where this occurs is when simulating the thermal flux of a reactor core. Most of the thermal neutrons are in the water moderator, which is thin with respect to the fuel elements. Of the three homotopy formulations, the diffusion imbedding performed the worst. It rapidly degraded in effectiveness as the number of homotopy intervals increased. This is likely due to compounding error from the rough inversion of the diffusion operator that is being performed at each step in the homotopy deformation.

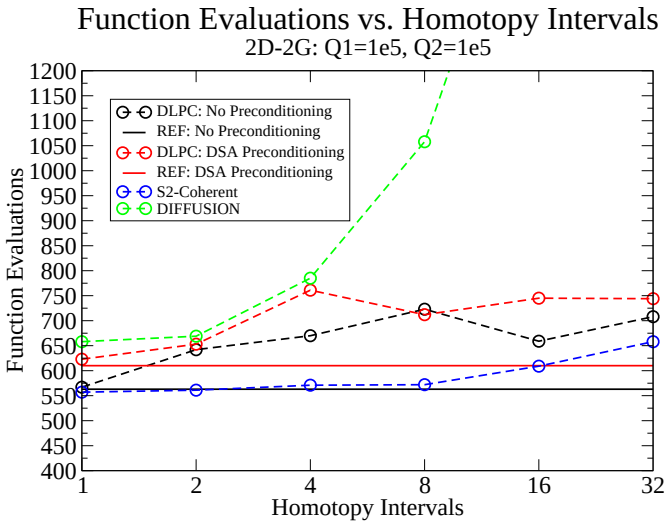

Figure 4.10: Convergence Results for Internal Sources of Q1=1.0×10^5 and Q2=1.0×10^5

4.4.2 Results for 30-Group 2D Problem

We now examine a thirty energy group problem using our same 2D geometry. We generate the scattering data using the same algorithm from Rosa, et, al. [44], keeping the same degree of difficulty, but increasing the number of energy groups over which to generate the data. We compute the solution in parallel on the moonlight computing cluster at Los Alamos National Laboratory with 54 processors. All of our other numerical parameters remain the same as in Table 4.6.

We first examine the problem configuration where the density of the outer (green) material of Figure 4.7 has a density of 1.0×10^5 and the internal material (purple) has a density of 1.0×10^2. We assign a fixed internal source of 1.0×10^5 to the outer material and an internal source of 1.0×10^2 to the internal material. We again measure effectiveness in terms of functional evaluations. The numerical results for this problem are in Figure 4.11. We observe that the DLPC and Diffusion formulation

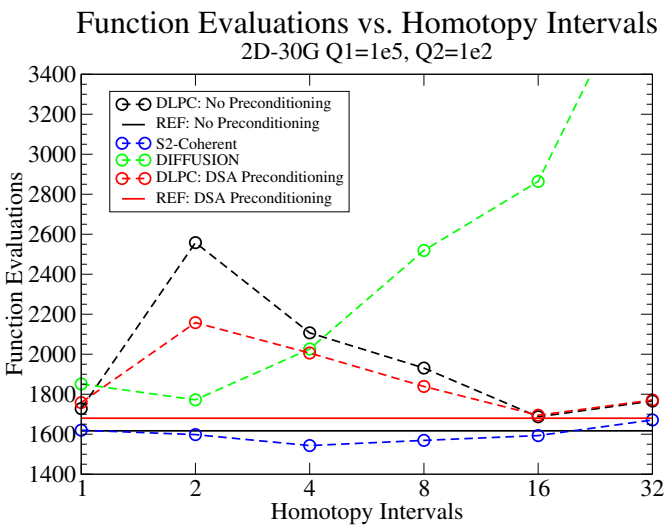

Figure 4.11: Convergence Results for Internal Sources of Q1=1.0×10^5 and Q2=1.0×10^2

provide no benefit over the reference case. The S2-Coherent Isotropic imbedding provides a slight benefit over the reference case, but degrades as the number of homotopy intervals increases.

We next examine the problem configuration where the density of the outer material and inner material are unchanged. We change the outer fixed internal source to 1.0×10^2 and the internal fixed internal source to 1.0×10^5. The numerical results for this problem are in Figure 4.12. We observe that the DLPC formulation essentially provides no benefit over the reference case. However, both the S2-Coherent Isotropic and Diffusion imbeddings exhibit usefulness, particularly for low numbers of homotopy intervals. The S2-Coherent Isotropic imbedding improves the convergence speed by about 7% when using 2-16 homotopy intervals. This is not a large improvement, but it is better than using DSA preconditioning. All homotopy formulations appear to degrade as the number of homotopy intervals increases.

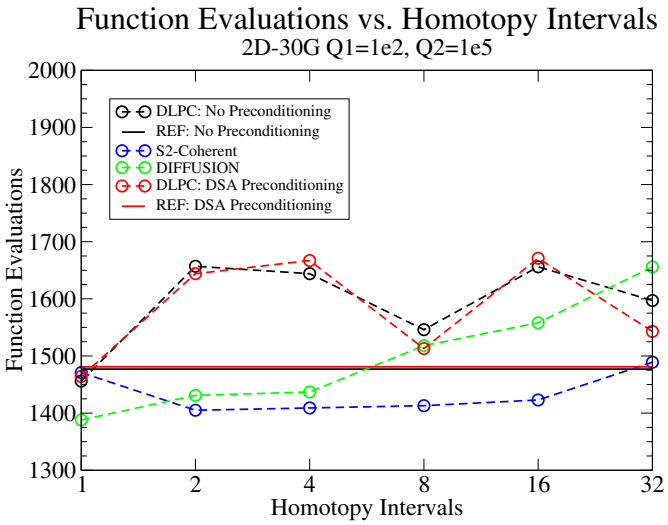

Figure 4.12: Convergence Results for Internal Sources of Q1=1.0×10^2 and Q2=1.0×10^5

Chapter 5

k-Eigenvalue Problem

5.1 Overview

In this chapter we develop the homotopy formulations that we use for the k-eigenvalue problem. We examine a one dimensional slab problem with a high dominance ratio as well as two dimensional problems that are more representative of nuclear fuel elements using a thirty-group structure. We investigate whether homotopy continuation can improve convergence speed of these problems as well as provide a stable initial guess for problems where a high dominance ratio is present. Lastly, we implement pseudo-arclength continuation for the one dimensional slab problem with a high dominance ratio. We particularly investigate where problem instabilities may arise for the preferred imbedding (ABLOCK).

5.2 Homotopy Formulations

We construct all of our homotopy formulations for the k-eigenvalue problem using an artificial parameter to scale from an imbedded easier problem to the difficult/complex original problem

$$H(\phi, k, \lambda) = (1 - \lambda)G(\phi, k) + \lambda F(\phi, k), \tag{5.1}$$

where $\lambda \in [0, 1]$, $G(\phi, k)$ is our imbedded system, and $F(\phi, k)$ is our original system. We formulate three different imbeddings for $G(\phi, k)$: ABLOCK, S2-Coherent Isotropic, and diffusion. The algorithm for our implementation of this homotopy formulation is given in Algorithm 5.

Algorithm 5 Artificial Parameter Homotopy Continuation

Given (ϕ_0, k_0), $G(\phi, k)$, and $F(\phi, k)$

for $n = 0, \ldots, N$ **do**

$\lambda_n = \frac{n}{N}$

Scale Tolerance: Tol $= (1 - \lambda_n^4)(Itol) + \lambda_n^4(Ftol)$

Correct: minimize $H(\phi_n, k_n) = (1-\lambda_n)G(\phi_n, k_n)+\lambda_n F(\phi_n, k_n)$ with initial guess (ϕ_0, k_0)

Calculate Tangent: $\vec{t}_n = \frac{(\phi_n, k_n)-(\phi_{n-1}, k_{n-1})}{\lambda_n - \lambda_{n-1}}$

Predict: $\tilde{\phi}_n = \phi_n + \Delta\lambda_n \vec{t}_n$

$\phi_0 = \tilde{\phi}_n$

end for

5.2.1 ABLOCK

Carstensen et. al. [13] recommend a symmetric real initial imbedding matrix even for nonsymmetric real eigenvalue problems. A symmetric real matrix that is in the neighborhood of the original system should help keep the eigenpaths from crossing.

If we represent the k-eigenvalue problem very simply in operator notation as

$$A\phi = \frac{1}{k}B\phi, \tag{5.2}$$

then we seek an imbedding where the operator A is nearly diagonal (and symmetric) for easy inversion. The operator, A, is representative of the following physical operators for our problem

$$A = I - DL^{-1}MS. \tag{5.3}$$

We formulate a block-diagonal operator for A by decoupling the system with respect to all elements of the phase space. The only coupling that we keep is the internal nodal coupling of the spatial finite element space. We will call this imbedding, ABLOCK, due to the block-diagonal nature of the given matrix, A.

Because this is an imbedding that is mathematically inspired, there is a concern that the non-physical nature of the imbedding could result in a disjointed path tracing (singularities and turning points). We numerically model the solution in Figure 5.1 for the 1D slab problem from section 5.3 to show that the path tracing is generally smooth and continuous, except near the very beginning of the tracing. Figures 5.2 and 5.3 show that the pathing may experience jumping issues when using a loose initial tolerance (1e-3); although, if it is extremely loose (1e-1) then the path jumping seems to disappear.

5.2.2 S2-Coherent Isotropic

Just as is in Chapter 4, we create a reduced quadrature formulation for our k-eigenvalue problem where we use an S_2 quadrature. We also extract the coherent isotropic scattering data from the full problem and use only the within group scattering contribution for our reduced quadrature problem. This leads to a scattering matrix that is diagonal. We do not alter the fission distribution matrix.

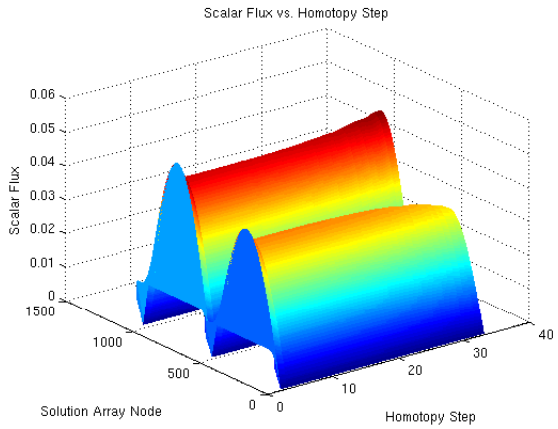

Figure 5.1: ABLOCK Mapping for Initial Tolerance $= 1 \times 10^{-6}$

We again call this the S_2-Coherent Isotropic imbedding where the embedded problem that represents G is

$$G(\phi, k) = \phi - [DL_{S_2}^{-1} M\tilde{S} + \frac{1}{k} DL_{S_2}^{-1} MF]\phi = 0, \tag{5.4}$$

where L_{S_2} represents our transport operator with the associated discrete ordinates for S_2 and $\tilde{S}$ represents the coherent isotropic scattering matrix. In our results, we

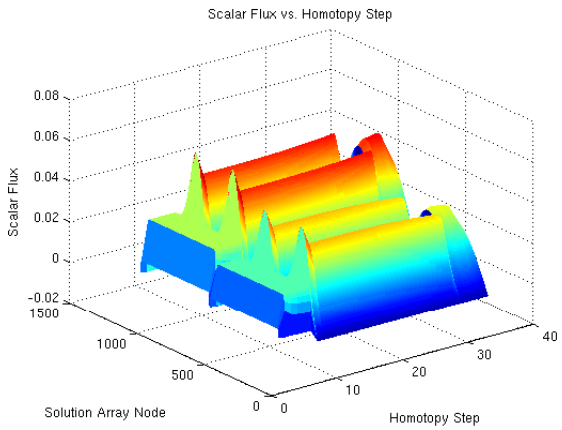

Figure 5.2: ABLOCK Mapping for Initial Tolerance $= 1 \times 10^{-3}$

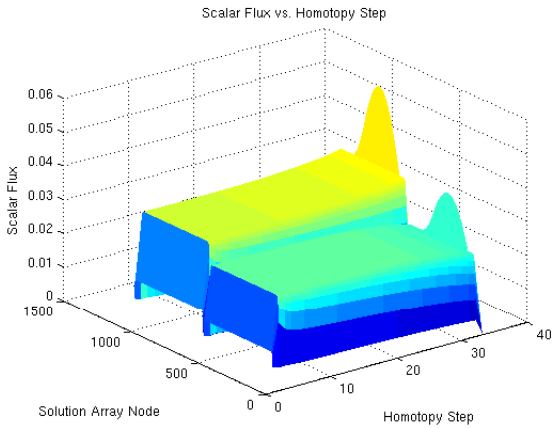

Figure 5.3: ABLOCK Mapping for Initial Tolerance = 1×10^{-1}

will again denote this imbedding as S_2-Coherent.

5.2.3 Diffusion

In our diffusion imbedding, we solve the steady state multi-group k-eigenvalue diffusion equation for the given problem specifications. This equation is expressed as

$$G(\phi, k) = -\nabla \cdot D_g(\vec{r})\nabla\phi_g(\vec{r}) + \sigma_{t_g}(\vec{r})\phi_g(\vec{r}) - \sum_{g'=1}^{G} \sigma_{s_{g'\to g}}(\vec{r})\phi_{g'}(\vec{r}) + \frac{1}{k}\chi_g \sum_{g'=1}^{G} \nu'_g \sigma_{f_{g'}}\phi'_g(\vec{r}) = 0, \tag{5.5}$$

where D_g is our multi-group diffusion coefficient, χ_g is the distribution fraction of fission neutrons into the energy group, ν_g is the average number of neutrons emitted per fission of the material, and $\sigma_{f_{g'}}$ is the probability of a fission occurring in group g'.

Just as in the thick-diffusive problem, extra interpolation and extrapolation steps need to be performed when passing the scalar flux between the diffusion imbedding and the LDFEM transport. We again restrict our Krylov tolerance to 1.0E-02 when

inverting the diffusion operator within each function evaluation to keep the computational cost small with respect to the transport sweep. For all instances of the diffusion imbedding, we use GMRES as our Krylov solver.

5.3 Iteration Results for 1D - 2 Group Problem

In this section we examine the iteration results for the k-eigenvalue problem with 2 energy groups. We first look at a manufactured problem that was first presented in [21]. This is a one dimensional homogeneous slab problem that has been artificially scaled such that the dominance ratio is very large. We will examine the case when the dominance ratio factor (drf) is 32, which corresponds to a dominance ratio of 0.9989.

We apply three numerical solvers to the problem, a fixed point iteration (FPI) without any acceleration, fixed point iteration with nonlinear krylov acceleration (NKA), and unpreconditioned Jacobian-Free Newton Krylov (JFNK). The Krylov solver for JFNK is restarted GCRODR Belos with restart 30. We use a structured bar element for our finite element method. Further geometric specifications for the problem are listed in Table 5.1 and cross section data is specified in Table 5.3.

5.3.1 Convergence Acceleration

We first examine the effectiveness of using homotopy continuation to improve convergence for our various numerical solvers (FPI, NKA, JFNK). We use the JFNK solver from the NOX Trilinos package supported by Sandia National LAboratory. We use two different JFNK-NOX solvers with different forcing function specifications. NOX0 uses the internal Krylov solver tolerance as the ϵ for the Jacobian-Free approximation. NOX1 uses the packages adaptive procedure for the forcing term

flag set to 1.

We measure effectiveness of the method in terms of function evaluations from $G(\phi, k)$ and $F(\phi, k)$ because most of the computational effort is performed in the transport sweep operation (L^{-1}). Except for the diffusion imbedding, both G and F perform one transport sweep per function evaluation. For each function In each case, we show what the reference number of function evaluations would cost if given the initial guess for the k-eigenvalue (k_0) that is associated with the chosen imbedding. The k-eigenvalue that we are converging to is 3.1606739.

Table 5.1: 1D-Slab Geometric and Angular Discretization for k-Eigenvalue Problem

Parameter	Value
Length	769.632 [cm]
Cells	256
Nodes	512
Regions	1
Boundaries	Vacuum
Spatial Discretization	LDFEM
Quadrature	Gauss-Legendre S_8

Table 5.2: 1D-Slab Numerical Solver Parameters for k-Eigenvalue Problem

Parameter	Value
Krylov Solver	GCRODR Belos, restart 30
Max Iterations	1000
Final Tolerance	1.0e-8
Initial Tolerance	1.0e-2
Predictor	SECANT

In Figure 5.4 we observe that homotopy continuation helps standardize the number of function evaluations for the fixed point iteration to be on the lower end of what is achievable for various initial guesses for k_0. This occurs when many homotopy intervals are used. The diffusion imbedding appears to be the most reliable and

Table 5.3: Cross Section Data for 1D-Slab k-Eigenvalue Problem

Parameter	Group (g=1)	Group (g=2)
Energy Bins	2.4-17.0 [MeV]	0.0-2.4 [MeV]
σ_{t_g}	$2.16 \times 10^{-1}[\text{cm}^{-1}]$	$3.456 \times 10^{-1}[\text{cm}^{-1}]$
$\sigma_{s_{1\to g}}$	$7.824 \times 10^{-2}[\text{cm}^{-1}]$	$3.60 \times 10^{-2}[\text{cm}^{-1}]$
$\sigma_{s_{2\to g}}$	$7.20 \times 10^{-2}[\text{cm}^{-1}]$	$2.6304 \times 10^{-1}[\text{cm}^{-1}]$
σ_{f_g}	$1.67 \times 10^{-1}[\text{cm}^{-1}]$	$1.728 \times 10^{-1}[\text{cm}^{-1}]$
χ_g	0.575	0.425
ρ	$1.0[\text{g/cm}^3]$	

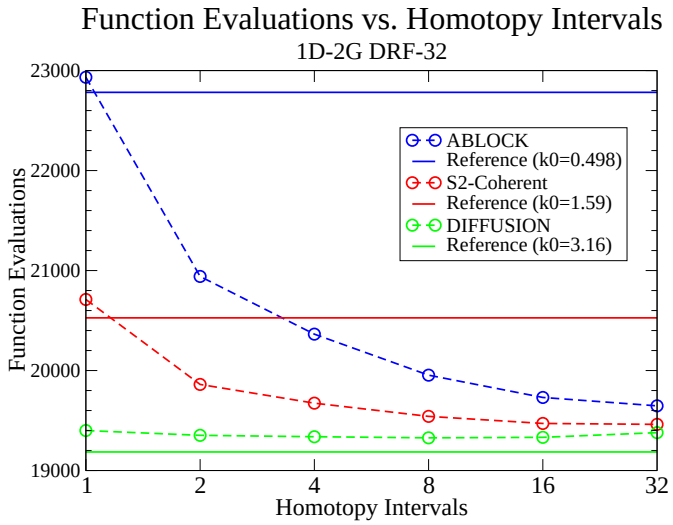

Figure 5.4: 1D Slab Results with FPI

performs better than the ABLOCK and S2-Coherent Isotropic imbeddings. This is not surprising since the problem is a homogeneous slab.

In Figure 5.5 we observe that homotopy continuation sometimes improves convergence rate for NKA, but not according to any systematic pattern. It appears that all imbeddings perform well for 8 homotopy intervals. The best use of homotopy in this problem occurs when using S2-Coherent Isotropic with one homotopy interval.

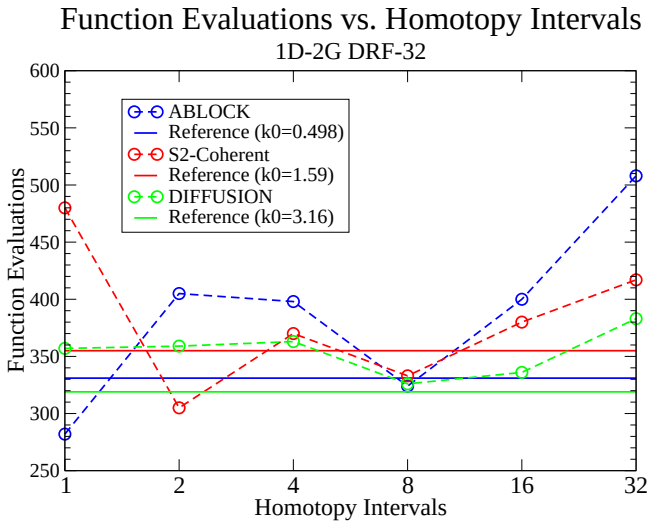

Figure 5.5: 1D Slab Results with NKA

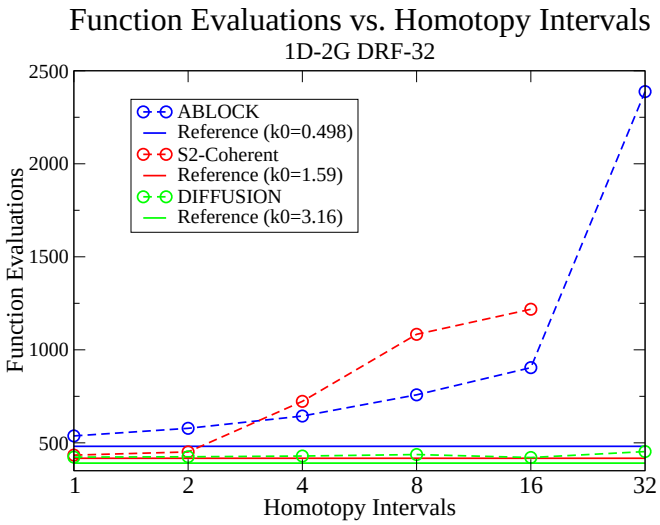

Figure 5.6: 1D Slab Results with JFNK-NOX0

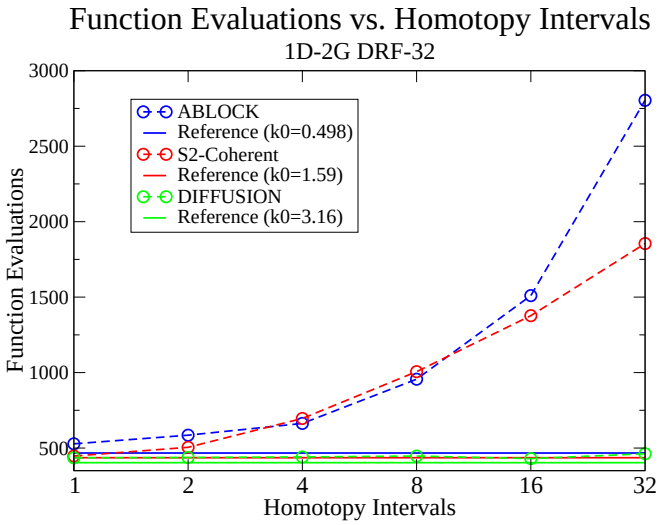

Figure 5.7: 1D Slab Results with JFNK-NOX1

In Figures 5.6 and 5.7 we observe that homotopy continuation generally not improving convergence speed. The best performance is with the diffusion imbedding, but it is not much different than any of the reference cases. We note that the NOX0 solver (fixed ϵ) does not converge the correct eigenmode for the S2-Coherent Isotropic case when 32 homotopy intervals are used.

5.3.2 Robustness of Newton Type Formulation

We next examine the use of homotopy continuation for the purpose of providing a stable initial guess for the Newton type formulation of the k-eigenvalue problem. We run the problem using the built in JFNK solver for capsaicin with initial guesses ranging from 0.1 to 4.0 in increments of 0.1. The dominant eigenvalue is 3.1606739. In our reference case without homotopy continuation we observe in Table 5.4 that there are four initial guesses for k-eff that converge to a wrong (non-dominant) eigenvalue.

Table 5.4: Failed Initial Guesses for Reference Case

k_0	Converged Eigenvalue
2.4	3.1514721
3.0	3.1514721
3.1	3.1514721
3.2	3.1514721

Table 5.5: Failed Initial Guesses with S2-Coherent Isotropic over 1 Homotopy Interval

k_0	Converged Eigenvalue
1.6	—
2.5	3.1332529
3.3	3.1514721
3.4	3.1514721
3.5	—
3.6	—

These correspond to the second eigenmode of the problem.

We first formulate our homotopy with an initial tolerance equal to the final tolerance, 1e-08. We run the simulation with only one homotopy interval. This corresponds to an alternative initial guess. We observed that the ABLOCK imbedding converged to the correct eigenvalue for every initial guess of k_0. The S2-Coherent Isotropic and diffusion imbeddings did not always converge to the correct eigenvalue. We show the tabulated data for these failed initial guesses in Tables 5.5 and 5.6. The "−" in the tables indicates that the simulation diverged.

We next formulate a homotopy path with 32 intervals and an initial tolerance of 1e-2 that is scaled as shown in Eq. (4.14). We again observe that the ABLOCK imbedding converges to the correct eigenvalue for every initial guess. The S2-Coherent Isotropic and diffusion imbeddings continue to converge to the wrong eigenvalue, and even diverge. The failed initial guesses for these imbeddings are show in Tables 5.7 and 5.8.

Table 5.6: Failed Initial Guesses with Diffusion over 1 Homotopy Interval

k_0	Converged Eigenvalue
3.7	3.1332529

Table 5.7: Failed Initial Guesses with S2-Coherent Isotropic over 32 Homotopy Intervals

k_0	Converged Eigenvalue
0.2	—
1.6	—

Table 5.8: Failed Initial Guesses with Diffusion over 32 Homotopy Intervals

k_0	Converged Eigenvalue
3.0	3.1514721
3.1	3.1514721
3.2	3.1514721
3.3	—

5.4 Iteration Results for 2D - 30 Group Problems

In this section we investigate the use of homotopy continuation applied in two dimensional geometry with multiple materials. We use a multigroup energy structure that is thirty energy groups large. The material data is provided by the Nuclear Data Interface (NDI) database supported by Los Alamos National Laboratory. We first examine a problem that has cruciform geometry with materials representative of highly enriched uranium and light water. The second problem is a 2D slab formation composed of the same materials, but artificially thickened to achieve a higher dominance ratio.

5.4.1 Cruciform Uranium-Water Problem

We use highly enriched uranium (93.71% U235) as one of our materials and water as our second material. We specify a temperature of 4.0×10^{-4} [MeV]. We build our geometry such that the uranium is distributed in a cruciform manner, with water filling the corner regions. This is visually depicted in Figure 5.8. We simulate the problem with and without anisotropic scattering. The first set of results correspond to isotropic scattering. The second set of results correspond to a P_3 scattering cross section expansion for anisotropic moments. Further geometric and numerical specifications for the problem are given in Tables 5.9 and 5.10. The problems dimensions are aligned with the data provided by NDI (centimeters, grams, seconds).

This is a hard problem in the sense that it is computationally intensive due to the large number of energy groups. With thirty energy groups, there is up-scatter present in the thermal range. We are interested in whether homotopy continuation can provide a benefit for such difficult problems regardless of whether a high dominance ratio is present.

Table 5.9: Geometric and Angular Discretization for 2D-Heterogeneous Cruciform Eigenvalue Problem

Parameter	Value
Side Lengths	12 [cm]
Cells	900
Regions	2
Boundaries	Vacuum
Spatial Discretization	Structured Box LDFEM
Quadrature	Gauss-Legendre S_8

Table 5.10: Numerical Solver Parameters for 2D-Heterogeneous Cruciform Eigenvalue Problem

Parameter	Value
Krylov Solver	GCRODR Belos, restart 30
Max Iterations	1000
Final Tolerance	1.0×10^{-8}
Initial Tolerance	1.0×10^{-2}
Predictor	SECANT

We observe in Figure 5.9 that homotopy continuation provides marginal benefit for the fixed point iteration solution method with no cross section expansion. The benefits are only for 1 or 2 homotopy intervals. All imbeddings degrade in effectiveness as the number of homotopy intervals increases. The best results were observed with the diffusion imbedding.

In Figures 5.10 and 5.11 we observe that no homotopy formulation provides any improvement in convergence compared to the reference cases. Although, in the JFNK-NOX1 case the reference solution corresponding to the ABLOCK imbedding, $k_0 = 0.0382$, did not converge to the correct eigenmode. The ABLOCK imbedding did not converge the correct eigenmode when 2 and 4 intervals were used. This suggests that stability of the nonlinear solver is an issue, but that ABLOCK is not the best choice for this type of problem.

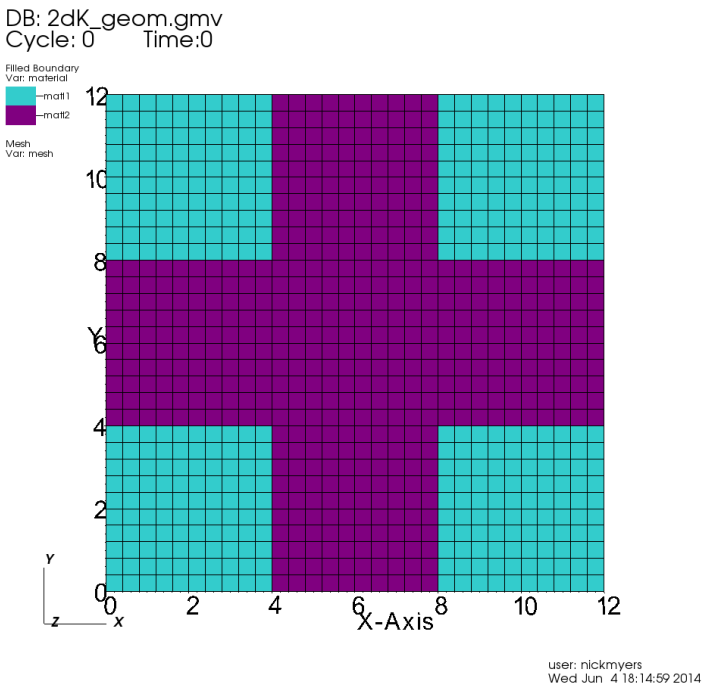

Figure 5.8: Problem Geometry for 2D-Heterogeneous Cruciform K-Eigenvalue Problem

In Figures 5.12, 5.13, and 5.14 we show that our homotopy continuation formulations provide essentially no benefit towards improving convergence speed in the problems where we specify a P_3 scattering cross section expansion.

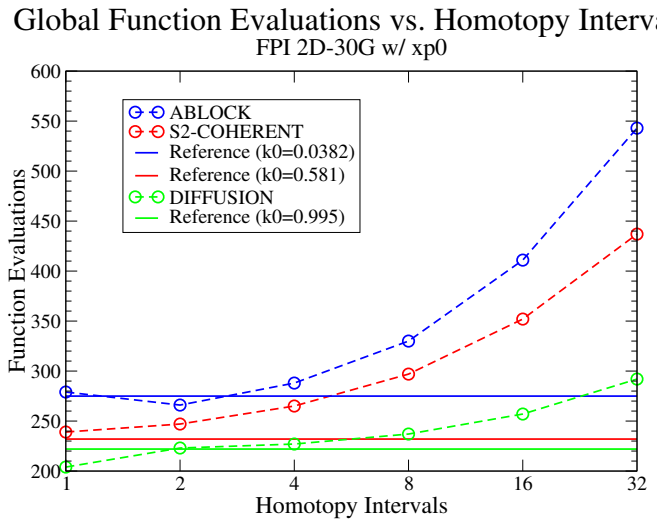

Figure 5.9: Results for 2D-30G Cruciform Problem using FPI and Cross Section expansion order 0

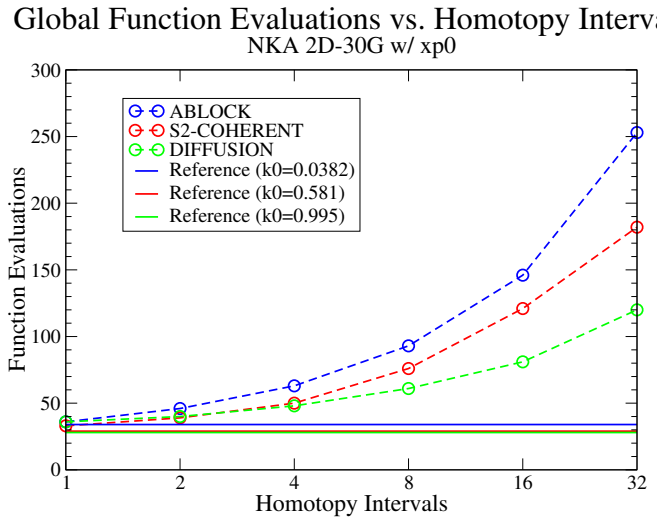

Figure 5.10: Results for 2D-30G Cruciform Problem using NKA

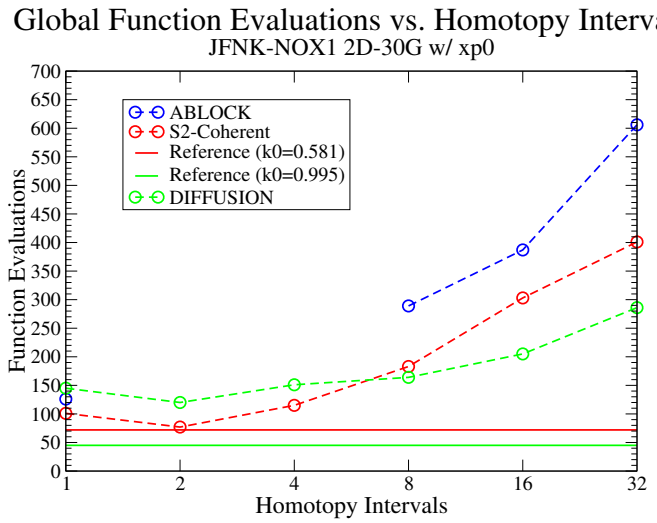

Figure 5.11: Results for 2D-30G Cruciform Problem using JFNK-NOX1

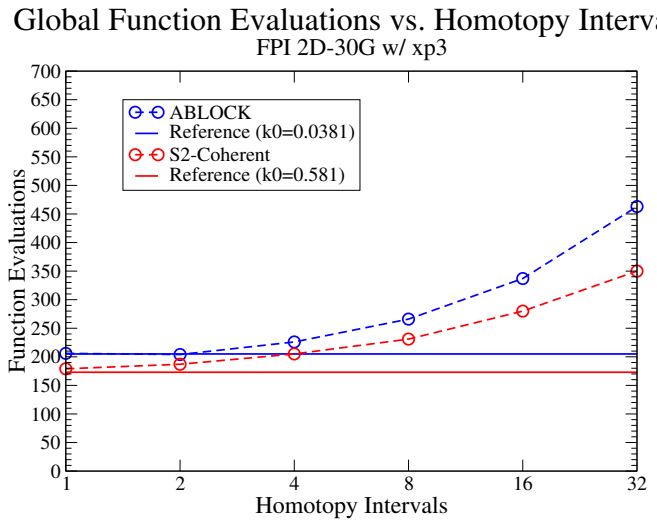

Figure 5.12: Results for 2D-30G Cruciform Problem using FPI and Cross Section expansion order 3

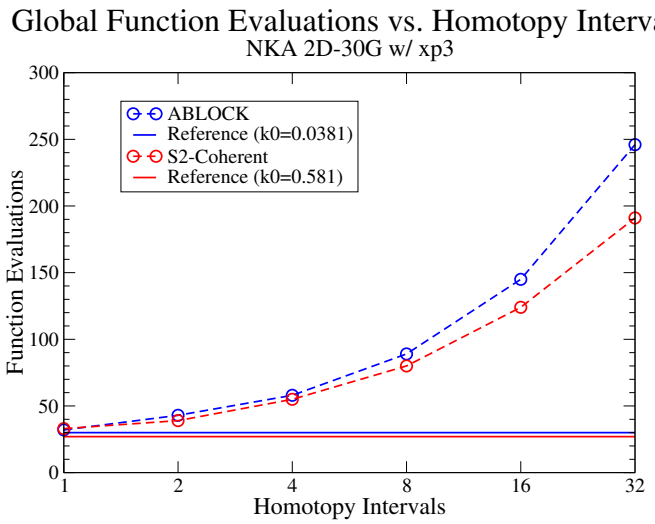

Figure 5.13: Results for 2D-30G Cruciform Problem using NKA and Cross Section expansion order 3

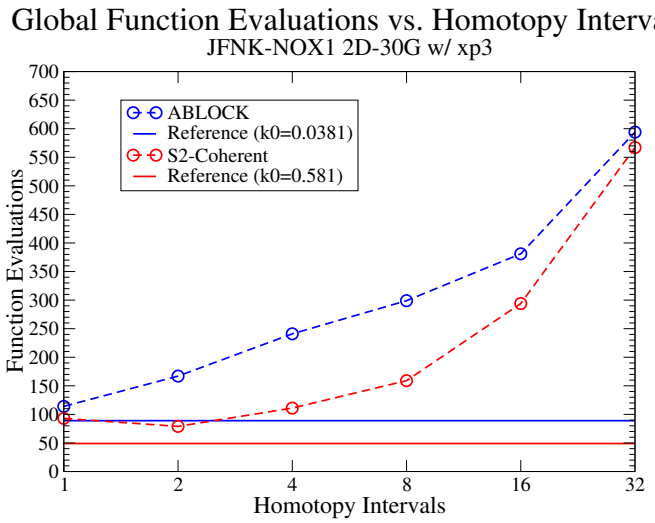

Figure 5.14: Results for 2D-30G Cruciform Problem using JFNK-NOX1 and Cross Section expansion order 3

5.4.2 2D-Slab Problem

We now examine a second two dimensional spatial problem that has been artificially scaled to achieve a higher dominance ratio. We use the same materials as in the previous 2D problem, but scale the densities of the Uranium and Water materials by a factor of 1000. We also increase the number of cells to 3879 for greater resolution. Figure 5.15 shows the geometry for the problem. We prescribe reflecting boundary conditions on the top and bottom boundaries.

We examine using homotopy continuation with three typical numerical solvers: fixed point iteration (FPI), FPI with nonlinear krylov acceleration (NKA), and unpreconditioned Jacobian-Free Newton Krylov (JFNK). We again use the Trilinos NOX package as our JFNK solver. We measure effectiveness in terms of function evaluations required to converge the dominant eigenmode. We apply our three imbeddings (ABLOCK, S2-Coherent Isotropic, and Diffusion) with varying resolution in the number of homotopy intervals in the path tracing. The results for this problem are shown in Figures 5.16, 5.17, and 5.18. The diffusion imbedding results are not shown because it diverged with the FPI solver and stagnated with the NKA and JFNK solvers.

We observe that homotopy continuation using the ABLOCK and S2-Coherent imbeddings provides convergence improvement with the FPI solver, but not NKA and JFNK. The method improves with the resolution of the path tracing when using FPI, but the opposite is true when using NKA and JFNK. Because this effectiveness with the FPI solver was observed in the 1D-slab problem (with a high dominance ratio) but not in the previous 2D-cruciform problem (without high dominance ratio), we conclude that homotopy continuation is useful for problems with a high dominance ratio that use FPI as a solver.

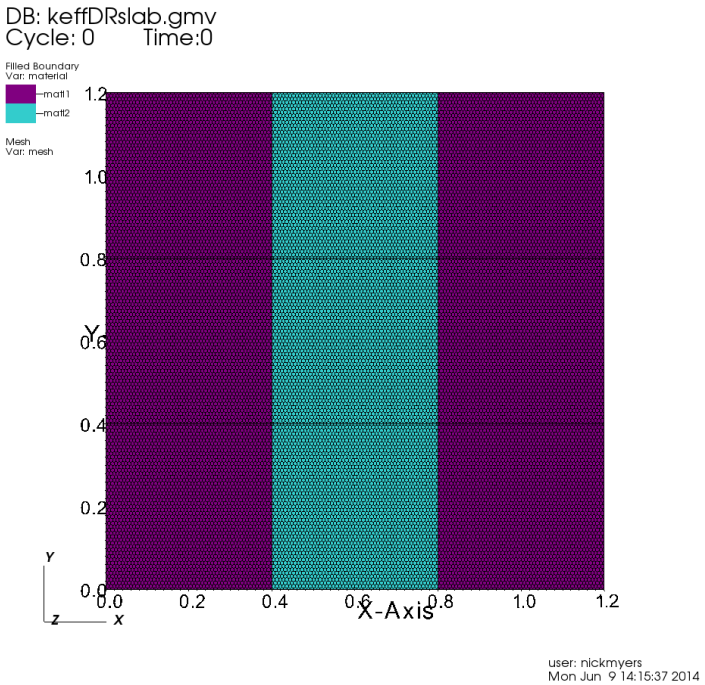

Figure 5.15: Geometry for 2D Thick Slab K-Eigenvalue Problem

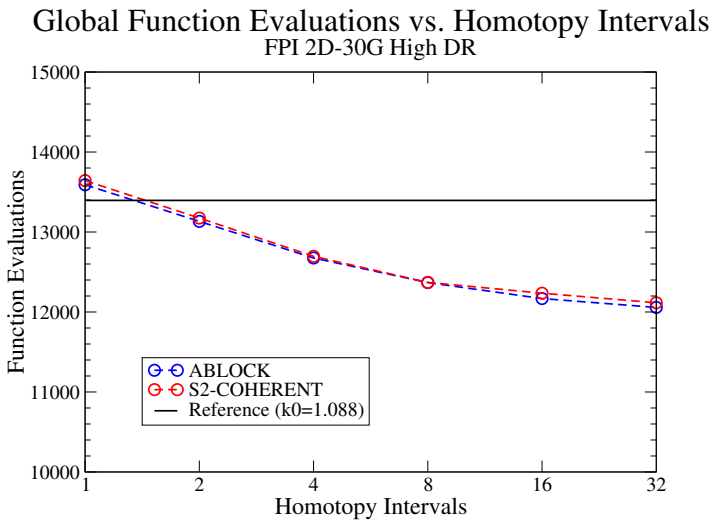

Figure 5.16: K-Eigenvalue FPI Results for 2D Slab K-Eigenvalue Problem

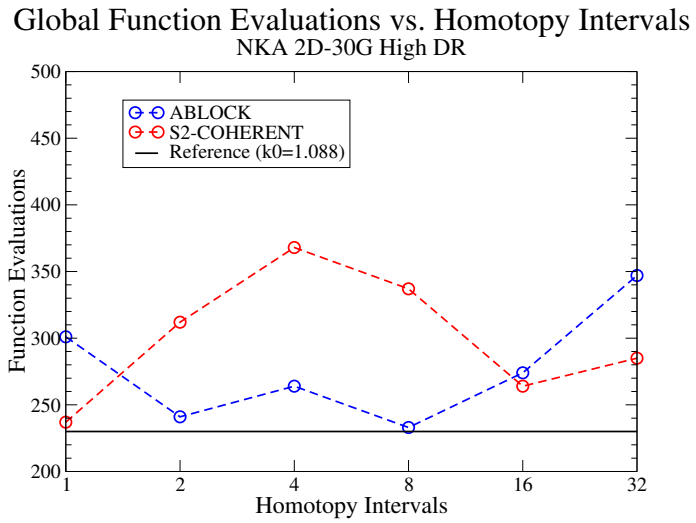

Figure 5.17: K-Eigenvalue NKA Results for 2D Slab K-Eigenvalue Problem

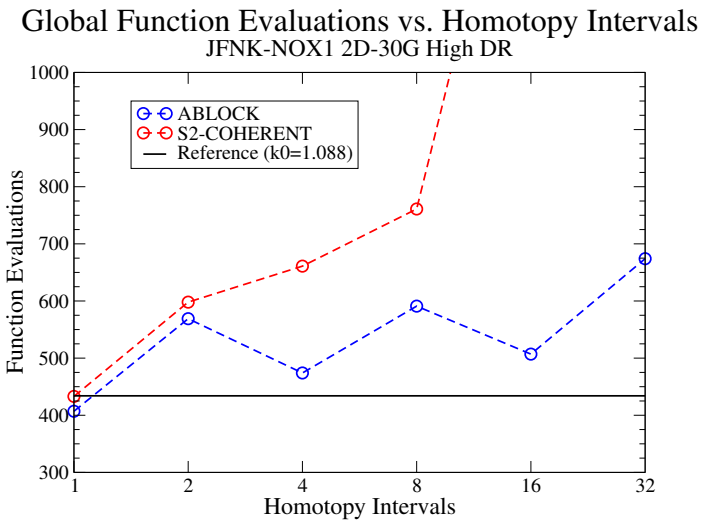

Figure 5.18: K-Eigenvalue JFNK-NOX1 Results for 2D Slab K-Eigenvalue Problem

5.5 Pseudo-Arclength Tracing

We implement the pseudo-arclength continuation (PSARC) algorithm in MATLAB for the 1D-slab problem with a dominance ratio of 0.6397 (DRF1) and 0.9989 (DRF32) [21]. We investigate whether PSARC provides more stability and requires fewer transport sweeps than the direct continuation method. We solve the k-eigenvalue neutral particle transport problem with both an unpreconditioned Jacobian-Free Newton Krylov implementation and an analytic Newton Method implementation. A plot of the eigenvalue spectrum for the problem is given in Figure 5.19

5.5.1 Tracing Derivation and Parameters

We use the convex artificial homotopy parameter formulation. Our nonlinear residual with the added rank for the PSARC formulation is

$$\Xi(\phi, k, \lambda) = \begin{pmatrix} (1-\lambda)(\phi - \tilde{P}(k)\phi) + \lambda(\phi - P(k)\phi) \\ k - k\frac{E^T F[(1-\lambda)\tilde{P}(k)+\lambda P(k)]\phi}{E^T F\phi} \\ t^* \delta u \end{pmatrix} = 0, \tag{5.6}$$

where $\tilde{P(k)}$ represents the initial imbedding physics, t is the calculated tangent vector at the current step, and δu is the step correction to the solution vector during the Newton iteration. This additional rank to the residual is a constraint that requires the step correction to be normal to the tangent vector. The Jacobian for this problem is calculated directly as follows for the analytic Newton Method

$$J = \begin{pmatrix} J_{11} & J_{12} & J_{13} \\ J_{12} & J_{22} & J_{23} \\ J_{13} & J_{32} & J_{33} \end{pmatrix}, \tag{5.7}$$

where the individual elements are defined as

$$J_{11} = I - [1-\lambda)\tilde{P(k)} + \lambda P(k)], \tag{5.8}$$

$$J_{12} = (1-\lambda)(\frac{1}{k^2}\tilde{D}\tilde{L}^{-1}\tilde{M}\tilde{F}\phi) + \lambda(\frac{1}{k^2}DL^{-1}MF\phi); \tag{5.9}$$

$$J_{13} = P(k)\phi - \tilde{P}(k)\phi, \tag{5.10}$$

$$\begin{aligned} J_{21} = \quad & -k\frac{(E^TF\phi)(E^TF[(1-\lambda)\tilde{P}(k)+\lambda P(k)]}{E^TF\phi)(E^TF\phi)} \\ & -\frac{(E^TF(1-\lambda)\tilde{P}(k)+\lambda P(k)\phi)(E^TF)}{E^TF\phi)(E^TF\phi)} \end{aligned} \tag{5.11}$$

$$J_{22} = 1.0 - \frac{E^TF[(1-\lambda)(\tilde{D}\tilde{L}^{-1}\tilde{M}\tilde{S}*\phi + \lambda DL^{-1}MS\phi]}{E^TF\phi} \tag{5.12}$$

$$J_{23} = k\frac{E^TF\left[\tilde{P}(k)\phi - P(k)\phi\right]}{E^TF\phi} \tag{5.13}$$

$$J_{31} = \left(\frac{\partial\phi}{\partial s}\right)^T \tag{5.14}$$

$$J_{32} = \frac{\partial k}{\partial s} \tag{5.15}$$

$$J_{33} = \frac{\partial\lambda}{\partial s} \tag{5.16}$$

We use two different predictor steps, the numerical secant and the exact Jacobian inversion. We use the numerical secant method when using JFNK as our corrector. We use the exact Jacobian inversion when we use the direct Newton method (NM) as our corrector. We seek to show the differences between an analytic direct tracing and the matrix-free tracing methods.

We use the adaptive step length algorithm from Eqs. (2.20a)-(2.20d). We set $\kappa_0 = 0.25$ as our reference contraction rate for the Newton iteration. We initially specify $h = 0.1$.

We check for special points after the predictor step, but before entering the corrector step. We monitor for turning points by calculating the Schur complement for

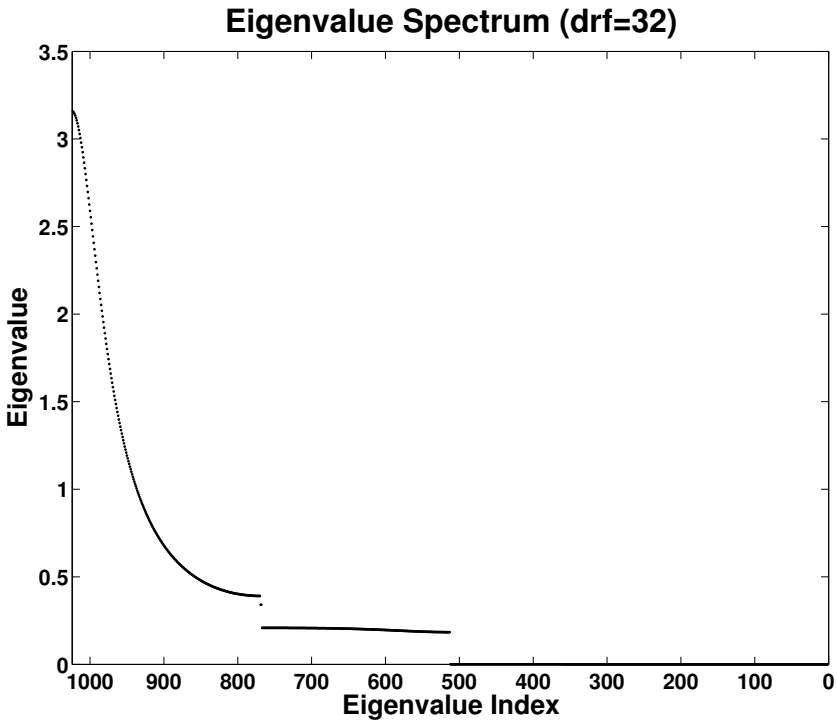

Figure 5.19: Eigenvalue Spectrum for DRF=32 1D-Slab Problem

the corner element of the system Jacobian that corresponds to the change in λ. The Schur complement for λ is calculated as

$$\frac{J_{31}\left(J_{31}^{*}J_{33}-J_{31}^{*}J_{31}J_{11}^{-1}J_{13}\right)}{J_{31}\left(J_{31}^{*}J_{32}-J_{31}^{*}J_{31}J_{11}^{-1}J_{12}\right)}-\frac{J_{21}\left(J_{21}^{*}J_{23}-J_{21}^{*}J_{21}J_{11}^{-1}J_{13}\right)}{J_{21}\left(J_{21}^{*}J_{22}-J_{21}^{*}J_{21}J_{11}^{-1}J_{12}\right)}. \tag{5.17}$$

When this scalar value evaluates to nearly zero, we know that a turning point is being approached. We monitor for bifurcation points by observing when the sign of the determinant of the system Jacobian changes. When a special point is detected, we perturb our predicted values by integrating an additional step length along the solution curve in order to try and jump over the problem area.

We also monitor for turning points in the Newton iteration. If a turning point is detected, we break out of the Newton iteration and perturb our predictor vector by

Table 5.11: PSARC Geometric and Angular Discretization

Parameter	Value
Length	769.632
Cells	256
Nodes	512
Regions	1
Boundaries	Vacuum
Spatial Discretization	LDFEM
Quadrature	Gauss-Legendre S_8

Table 5.12: PSARC Numerical Solver Parameters

Parameter	JFNK Values	NM Values
Inner Solver	GMRES Belos, restart 30	LU Gaussian Elimination
Max Outer Iterations	1000	1000
Final Tolerance	1.0×10^{-8}	1.0×10^{-8}
Initial Tolerance	1.0×10^{-2}	1.0×10^{-2}
Predictor	SECANT	EXACT

artificially incrementing λ forward to jump over the special point. The geometric and numerical parameter specifications for the problem are given in Tables 5.11 and 5.12. We prescribe a maximum number of homotopy steps of 1000 for each simulation. We end the tracing by flagging when the corrector step corrects to a state where $\lambda > 1$. We then fix $\lambda = 1$ and correct for the end state problem. The algorithm for this method is given in Algorithm 6.

Algorithm 6 Pseudo Arc-Length Continuation for k-Eigenvalue Problem

Given $c_0 = (\phi_0, k_0, \lambda_0)$, $G(\phi, k)$, and $F(\phi, k)$

$\lambda_0 = 0$

$\vec{t}_0 = h$

while $\lambda < 1$ AND iterations $<$ max iterations **do**

Calculate Tangent: $\begin{pmatrix} H_x & H_\lambda \\ (\partial x/\partial s)^* & (\partial\lambda/\partial s)^* \end{pmatrix}_n \begin{pmatrix} \partial x/\partial s \\ \partial\lambda/\partial s \end{pmatrix}_n = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

Predict: $\omega_{n+1} = c_n + h\vec{t}(c_n)$

Correct: converge c_{n+1} by minimizing $\begin{pmatrix} H(\omega_{n+1}) + \dot{H}(\omega_{n+1}) \\ \vec{t}^* \end{pmatrix} \delta c$

In Corrector: calculate $\tilde{p}$

if $\tilde{p} == 0.5$ **then**

Adapt Step Length $h = h * \tilde{p}$

Break corrector iteration and re-predict

end if

if special point **then**

Perturb ω_{n+1} and re-predict

end if

Adapt Step Length $h = h * \tilde{p}$

end while

5.5.2 Low Dominance Ratio Problem

First we investigate the problem with a low dominance ratio of 0.6397 where the largest eigenvalue is 2.5237. We trace the paths corresponding to four different initial guesses ($k_0 = 0.5, 1.5, 2.5, 3.5$). Figure 5.20 shows the path tracing for the ABLOCK imbedding using the JFNK corrector with a numerical secant predictor. All of the paths lead to the dominant eigenvalue of 2.5237, although the path that corresponds to $k_0 = 2.5$ initially corrects to a slight negative λ state before moving positive along the path.

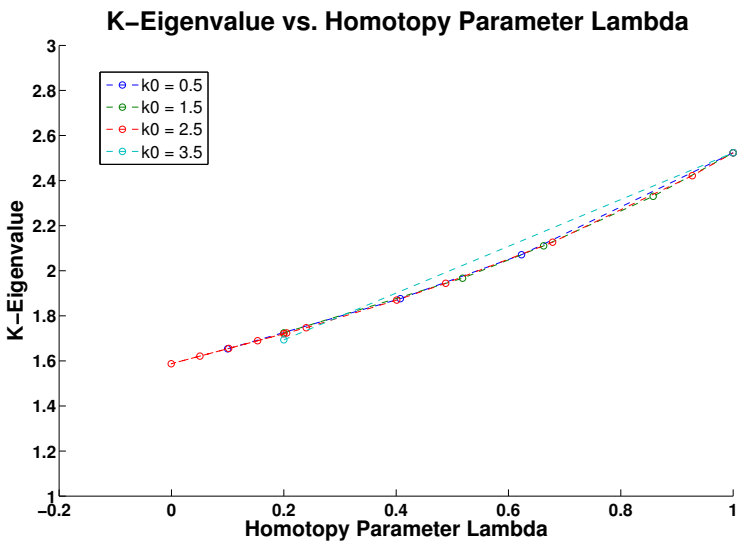

Figure 5.20: DRF 1: PSARC with JFNK and ABLOCK imbedding

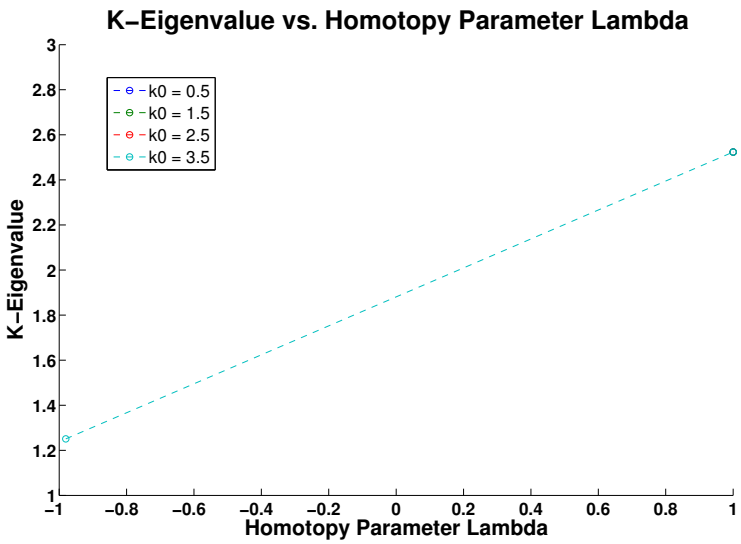

Figure 5.21: DRF 1: PSARC with NM and ABLOCK imbedding

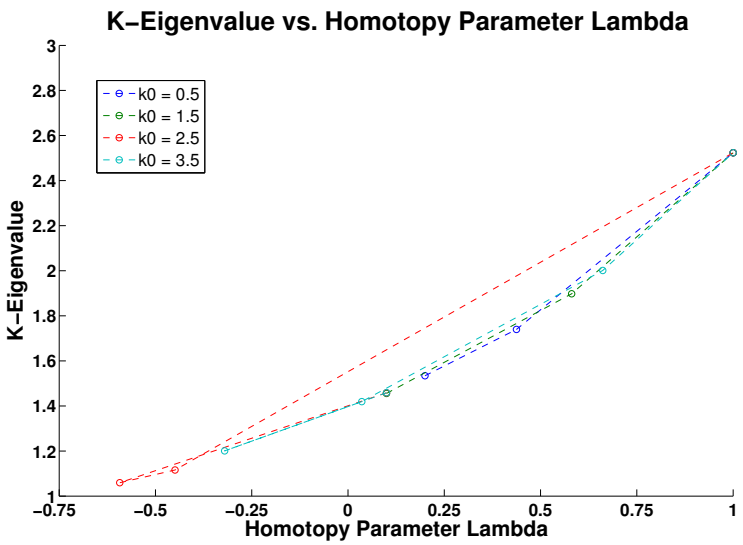

Figure 5.22: DRF 1: PSARC with JFNK and S2 imbedding

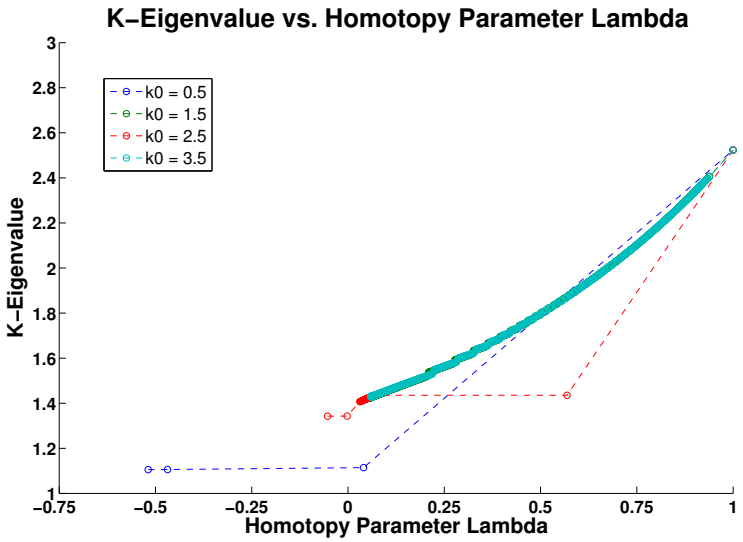

Figure 5.23: DRF 1: PSARC with NM and S2 imbedding

Figure 5.21 shows the path tracing for the ABLOCK imbedding using the NM corrector and exact Jacobian inversion for the predictor step calculation of a tangent vector. All initial guesses except $k_0 = 3.5$ immediately correct to a $\lambda > 1$ state and trigger the ending condition flag. The $k_0 = 3.5$ path initially corrects to a strongly negative λ state before jumping back to the end state. This seems to indicate that there may be a mirror attraction zone in the nonphysical negative λ space. All initial guesses lead to the dominant eigenvalue.

In Figures 5.22 and 5.23 we observe the path tracing for the S2-Coherent Isotropic imbedding using JFNK and NM respectively. Once again, the JFNK tracing appears to be generally well conditioned, excepting the initial negative domain excursions for $k_0 = 2.5$ and $k_0 = 3.5$. The NM method with the S2-Coherent Isotropic imbedding exhibits actual path tracing, though a bit erratic. The $k_0 = 1.5$ and $k_0 = 3.5$ paths trigger the adaptive step length to trace very tightly while the other two paths both suffer negative domain excursions and what appears to be a jump to a lower eigenvalue path. The $k_0 = 2.5$ path diverges instead of correcting to any eigenvalue. The tabulated data for the low dominance ratio problem is given in Tables 5.13 and 5.14.

Table 5.13: DRF1 Tabulated Results for PSARC with ABLOCK Imbedding

	JFNK-SECANT		NM-EXACT	
k_0	Transport Sweeps	k-eff	Transport Sweeps	k-eff
0.5	3136	2.5237	736	2.5237
1.5	4720	2.5237	832	2.5237
2.5	5632	2.5237	1040	NaN
3.5	2288	2.5237	1200	2.5237

Table 5.14: DRF1 Tabulated Results for Direct Tracing with S2-Coherent Imbedding

	JFNK-SECANT		NM-EXACT	
k_0	Transport Sweeps	k-eff	Transport Sweeps	k-eff
0.5	2272	2.5237	880	2.5237
1.5	2032	2.5237	22272	2.5237
2.5	2496	2.5237	3632	2.5237
3.5	2288	2.5237	27040	2.5237

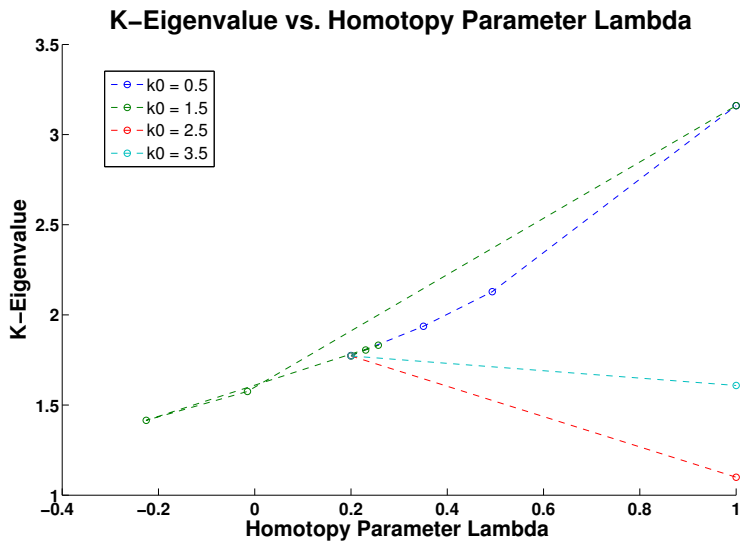

Figure 5.24: DRF 32: PSARC with JFNK and ABLOCK imbedding

5.5.3 High Dominance Ratio Problem

We next examine the problem with a high dominance ratio of 0.9989 where the largest eigenvalue is 3.16067. Figure 5.24 shows the path tracing for the JFNK corrector with the ABLOCK imbedding. We observe a change in the sign of the determinant of the system Jacobian at $\lambda = 0.2$ where the path bifurcates along different eigenvalue paths. Both the $k_0 = 0.5$ and $k_0 = 1.5$ paths trace to the dominant eigenvalue while the other two trace to much lower eigenvalues.

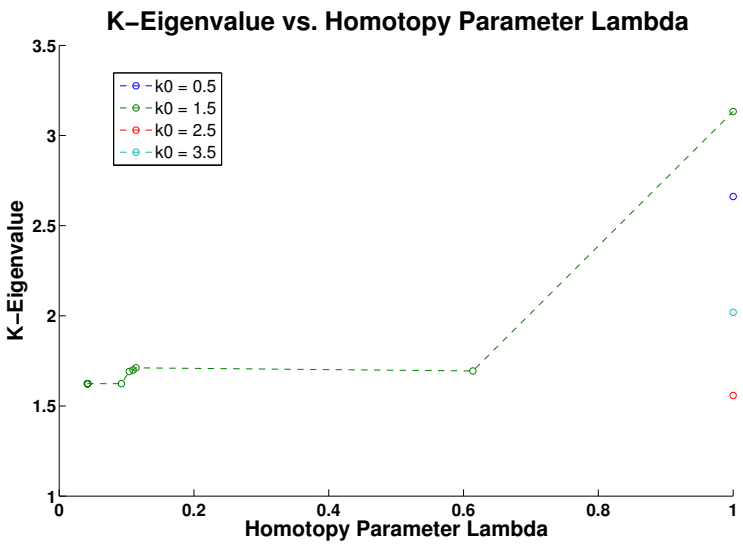

Figure 5.25: DRF 32: PSARC with NM and ABLOCK imbedding

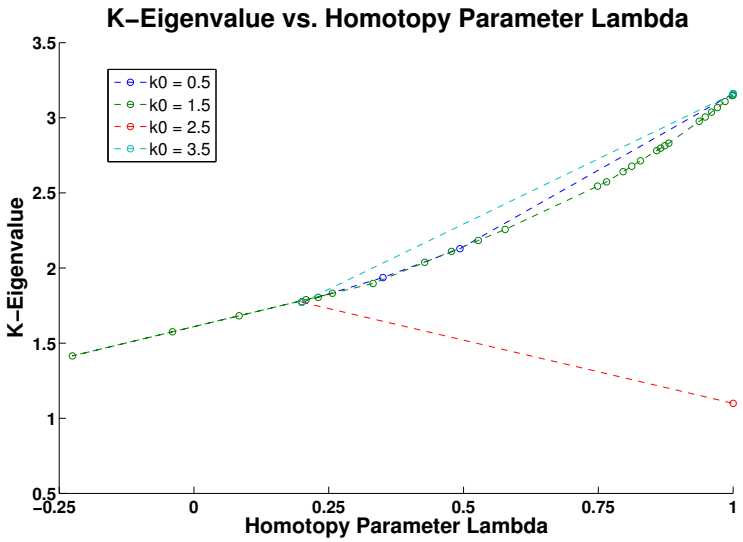

Figure 5.26: DRF 32: PSARC with JFNK and S2 imbedding

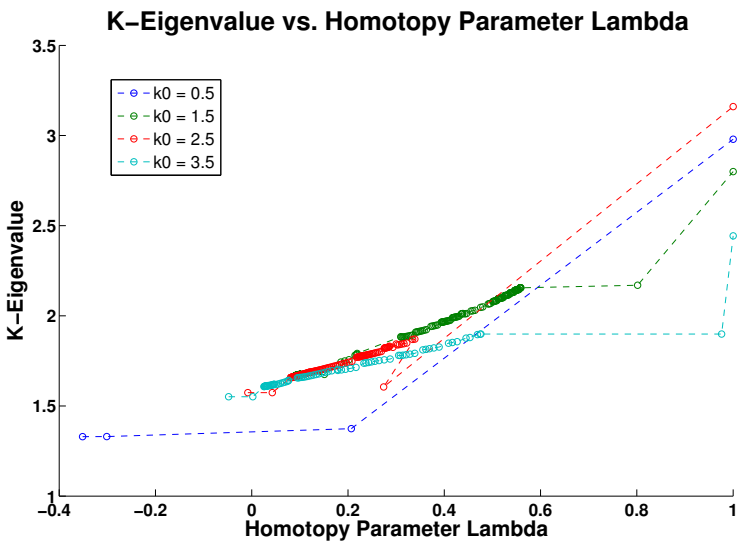

Figure 5.27: DRF 32: PSARC with NM and S2 imbedding

In Figure 5.25 we observe the path tracing using the NM corrector with the ABLOCK imbedding. Similar to the lower dominance ratio problem, three of the paths immediately trigger the $\lambda > 1$ flag and are forced to correct to the solution at $\lambda = 1$. Unlike the lower dominance ratio problem, only the path that actually is traced through intermediate steps is able to converge to the dominant eigenvalue.

In Figures 5.26 and 5.27 we observe the path tracing for the S2-Coherent Isotropic imbedding in the high dominance ratio problem. We again observe a similar bifurcation point around 0.2 with the JFNK corrector, but only the path corresponding to $k_0 = 2.5$ bifurcates to a lower eigenvalue. The NM corrector triggers the adaptive step length to adapt to very small values, just as in the lower dominance ratio problem. Similarly, many steps are taken before the paths jump to lower eigenvalue paths. Only the path corresponding to $k_0 = 2.5$ traces to the dominant eigenvalue with the NM corrector. The tabulated data for the high dominance ratio problem is given in Tables 5.15 and 5.16.

Table 5.15: DRF 32: Tabulated Results for PSARC with ABLOCK Imbedding

	JFNK-SECANT		NM-EXACT	
k_0	Transport Sweeps	k-eff	Transport Sweeps	k-eff
0.5	33744	3.16067	704	2.66184
1.5	35040	3.16067	800	3.13325
2.5	61264	1.09929	352	1.55817
3.5	47616	1.60829	672	2.01930

Table 5.16: DRF 32: Tabulated Results for PSARC with S2-Coherent Imbedding

	JFNK-SECANT		NM-EXACT	
k_0	Transport Sweeps	k-eff	Transport Sweeps	k-eff
0.5	35408	3.16067	704	2.97960
1.5	14720	3.15147	640	2.79981
2.5	121968	1.09929	1232	3.16067
3.5	25888	3.16067	672	2.44339

We observe that the unpreconditioned JFNK corrector is generally more stable than the unpreconditioned Newton Method. We monitored the condition number of the Jacobian throughout the tracing and observed that the condition number was often quite large. This seems to indicate that preconditioning of the system is required for robust path tracing. The approximation error of the JFNK method acts as perturbation that helps in avoiding correcting directly onto a special point.

5.5.4 Direct Parameter Tracing Comparison

We compare using the pseudo-arclength tracing method against a direct homotopy parameter tracing with the high dominance ratio problem (DRF32). We simulate the direct parameter tracing with our MATLAB code to be consistent with the PSARC results and use all the same numerical parameters as specified previously. We use both the ABLOCK and S2-Coherent Isotropic imbeddings for the problem. We fix

the number of homotopy intervals to be 8 uniformly distributed sections along the solution path.

We observe that the JFNK corrector with the direct parameter tracing method is more reliable in converging the dominant eigenvalue. The NM corrector also exhibits improved stability, but still has some paths converge to lower eigenvalues at $\lambda = 1$. We observe again that unpreconditioned JFNK is generally more stable of a corrector than the direct unpreconditioned Newton Method.

The number of sweeps required to converge the answer is not much different than what is required with the PSARC method. This leads us to conclude that the direct parameter tracing method is preferable to the PSARC method for neutral particle transport problems.

Table 5.17: Tabulated Results for Direct Tracing with ABLOCK Imbedding

	JFNK-SECANT		NM-EXACT	
k_0	Transport Sweeps	k-eff	Transport Sweeps	k-eff
0.5	24944	3.16067	1280	3.16067
1.5	32384	3.16067	1280	3.16067
2.5	32256	3.16067	1280	3.13325
3.5	24944	3.16067	1280	3.16067

Table 5.18: Tabulated Results for Direct Tracing with S2-Coherent Imbedding

	JFNK-SECANT		NM-EXACT	
k_0	Transport Sweeps	k-eff	Transport Sweeps	k-eff
0.5	35584	3.160674	1280	2.370071
1.5	25216	3.160674	1280	3.133254
2.5	22192	3.160674	1280	3.151472
3.5	22096	3.160674	1280	3.160674

5.5.5 Eigenvalue Sensitivity

We consider that the instabilities observed at $\lambda = 0$ may be due to eigenvalue sensitivity issues. Eigenvalue perturbation sensitivity has been well developed by Stewart [49]. Stewart explains that an ill-disposed eigenvalue is ill-conditioned and may spread that ill-conditioning to the other eigenvalues of the system.

Given our generalized eigenvalue problem

$$k\phi = A^{-1}B\phi, \tag{5.18}$$

we are interested in how a perturbation in the matrix $[A^{-1}B]$ affects the k-eigenvalues. From Moler [39], roundoff error can be thought of as a perturbation in the matrix $[A^{-1}B]$. Because we begin our tracing algorithm with a very loose tolerance at $\lambda = 0$, we check the eigenvalue condition numbers for our imbedding, $G(\phi, k, \lambda)$. The imbedding corresponds to an altered eigenvalue problem, $[\tilde{A}^{-1}\tilde{B}]$

We calculate the eigenvalue condition numbers using the *condeig*(A) function from MATLAB. The *condeig*(A) function returns a vector of condition numbers (β) corresponding to the eigenvalues of the matrix A. Large condition numbers magnify the perturbation error from $[A^{-1}B]$ in solving for the associated k-eigenvalue.

Table 5.19 shows the maximum calculated eigenvalue condition numbers associated with the ABLOCK, S2-Coherent Isotropic, and the original reference problem $[A^{-1}B]$. We observe that the ABLOCK imbedding has extremely ill-conditioned eigenvalues. This may explain why the analytic Newton Method struggled to converge the initial solution at $\lambda = 0$. While the S2-Coherent Isotropic imbedding does not have such ill-conditioned eigenvalues, they are more sensitive than the original reference system. Sensitive eigenvalues are not ideal for an imbedding choice, but the results suggest that the issues with the pseudo-arclength tracing may be more related to the sensitivity of the problem than the homotopy algorithm itself.

Table 5.19: Maximum Eigenvalue Condition Number

$H(\phi, k, \lambda)$	$\max(\beta)$
ABLOCK	4.593×10^{11}
S2	4.525×10^{2}
Original	1.662×10^{2}

Chapter 6

Conclusions and Future Development

6.1 Thick-Diffusive Problem Conclusions

We find that homotopy continuation provided some benefit for specific thick-diffusive problems. While convergence improvement was observed in the 1D geometric cases, the benefits were not sufficient to warrant implementation when compared with the power of diffusion synthetic acceleration (DSA). However, in the 2D geometric cases, we find that homotopy continuation improves convergence in problems where a large internal source is located in the thin materials of a heterogeneous problem. These are problems where DSA provides little or no benefit in accelerating convergence. Physically, these situations might represent the simulation of the thermal neutron flux in a reactor (where most neutrons are located in the moderator instead of the fuel elements).

Of the different imbeddings that we applied, we found that the S2-Coherent Isotropic imbedding performed well where homotopy continuation is useful. The

diffusion length preserving continuation formulation performed well when the number of energy groups was low (one or two), but degraded when the number of energy groups increased to thirty. This might be due to the non-asymptotic preserving nature of the group-to-group scattering cross sections.

6.2 k-Eigenvalue Problem Conclusions

We examined the use of homotopy continuation applied to the k-eigenvalue problem of radiation transport. We investigated both the direct and pseudo-arclength continuation methods to determine whether homotopy continuation would improve convergence. We measured improvement in reduced function evaluation count as well as stability in converging the dominant eigenmode of the nonlinear formulation of the k-eigenvalue problem.

We found that homotopy continuation exhibited usefulness in improving convergence speed with a fixed point iterative solver for problems with a high dominance ratio–both in the 1D and 2D geometric cases. This requires many homotopy intervals to bring the number of iterations down to the floor of what is achievable with a great initial guess.

Between the three different imbeddings, we found that the ABLOCK imbedding provided the most stability. In particular, we found that the ABLOCK imbedding improved stability of the JFNK nonlinear formulation of the k-eigenvalue problem in the 1D spatial geometry problem. We were able to converge the dominant eigenmode of the problem for each initial guess. This is an improvement over the reference case where four initial guesses failed.

The diffusion imbedding is very accurate for the 1D geometric problems with high dominance ratios, but degrades when higher spatial dimensional problems are used.

The S2-Coherent Isotropic imbedding was most useful in the 30-group 2D spatial geometry problem where a high dominance ratio is not present.

6.3 Pseudo-Arclength Continuation

We find that pseudo-arclength continuation has limited use in high dominance ratio problems. Both the ABLOCK and S2-Coherent Isotropic imbeddings experience severe instabilities early on in the path tracing. Unlike the direct parameter tracing, there is little guarantee that the correct eigenmode will be converged with pseudo-arclength continuation.

The JFNK corrector formulation is preferable to the direct Newton method. Not only is a Jacobian matrix expensive and sometimes impossible to compute explicitly for large scale scientific problems, but the approximation error of the JFNK method helps provide a suitable perturbation to avoid special points. However, when the solution paths very near one another (as is the case in a high dominance ratio problem) the path tracing can bifurcate due to jumping onto the neighboring solution curves.

We find that pseudo-arclength continuation generally requires more functional evaluations than the direct continuation formulations of the same problem. Pseudo-arclength continuation experienced more instability in the path tracing than the direct homotopy tracing, even with implementation of special point handling. However, preliminary eigenvalue sensitivity analysis indicates that the issue is likely caused by the imbedding eigenvalue sensitivity instead of the pseudo-arclength implementation.

The implementation of pseudo-arclength continuation is much more complex than that of direct parameter tracing. Because of the advantages of direct parameter tracing versus pseudo-arclength continuation, we recommend that direct parameter tracing be used for large scale neutral particle transport problems.

6.4 Future Development

The idea behind homotopy continuation is useful, even if the implementation of the concept in this dissertation only yielded positive results for specific problems. As a means of providing a better initial guess, homotopy continuation is able to be coupled readily with existing solvers and acceleration/preconditioning methods.

Future work with homotopy continuation might involve adaptive physics based preconditioning throughout solution curve tracing. We did not apply any matrix preconditioning with the nonlinear k-eigenvalue problem formulation. Future development of the pseudo-arclength continuation algorithm might require careful preconditioning to help damp the numerical instabilities observed in the correcting step.

According to the literature, homotopy continuation is generally best applied to highly nonlinear problems. While we have applied homotopy continuation to the nonlinear formulation of the k-eigenvalue problem, a more highly nonlinear problem might be suitable. Future development would include applying the homotopy continuation concept to multi-physics problems that are highly nonlinear–such as photon transport with matter coupling.

In this research we restricted ourselves to the real plane where we have applied the pseudo-arclength tracing algorithm. Future development of tracing algorithms could allow for tracing into the complex plane. This might allow for the method to circumvent special points that occur in the real domain.

We took a very experimental approach to measuring the usefulness of homotopy continuation in the problems examined. Future development of the method would benefit greatly from rigorous mathematical analysis to guide development of imbeddings and path tracing, such as that touched on with our preliminary eigenvalue sensitivity analysis.

References

[1] M. L. Adams and E. W. Larsen, *Fast Iterative Methods for Discrete-Ordinates Particle Transport Calculations*, Prog. Nucl. Energy **40** (2002), 3–159.

[2] E. L. Allgower and K. Georg, *Introduction to Numerical Continuation Methods*, Society for Industrial Mathematics, Philadelphia, PA, 2003.

[3] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK users' guide*, Third, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1999.

[4] K. Atkinson and W. Han, *Elementary numerical analysis*, Third Edition, John Wiley and Sons Inc. 2004.

[5] Y. Azmy, *Impossibility of Unconditional Stability and Robustness of Diffusive Accleration Schemes*, 1998 American Nuclear Society Radiation Protection and Shielding Division Topical Meeting, Nashville, TN, 1998, April, pp. 480.

[6] Y. Azmy, *Unconditionally Stable and Robust Adjacent-Cell Diffusive Preconditioning of Weighted-Difference Particle Transport Methods is Impossible*, J. Comp. Phys. **182** (2002), 213–233.

[7] Y. Azmy, T. Wareing, and J. Morel, *Effect of Material Heterogeneity on the Performance of DSA for Even-Parity S_N Methods*, International Conference on Mathematics and Computation, Reactor Physics, and Environmental Analysis in Nuclear Applications, 1999September, pp. 55–63.

[8] R. Bellman and R. Kalaba, *Invariant Imbedding and Mathematical Physics. I. Particle Processes*, Journal of Mathematical Physics **1** (1960), no. 4.

[9] R. Blomquist, M. Armishaw, D. Hanlon, N. Smith, Y. Naito, J. Yang, Y. Mioshi, T. Yamamoto, O. Jacquet, and J. Miss, *Source Convergence in Criticality Safety Analyses*, Technical Report 5431, OECD Nuclear Energy Agency, 2006.

[10] T. Vejchodsky, *A Direct Solver for Finite Element Matrices Requireing O(NlogN) Memory Places* (J. Brandts, S. Korotov, M. Krizek, J. Sistek, and T. Vejchodsky, eds.) Institute of Mathematics, 2013.

[11] M. Calef, E. Fichtl, J. Warsa, M. Berndt, and N. Carlson, *Nonlinear krylov acceleration applied to a discrete ordinates formulation of the k-eigenvalue problem*, Journal of Computational Physics **238** (2013), 188–209.

[12] S Campbell, I. Ipsen, C. Kelley, and C. Meyer, *Gmres and the minimal polynomial*, North Carolina State University, Center for Research and Scientific Computation, 1996.

[13] C. Carstensen, J. Gedicke, and V. Mehrmann, *An adaptive homotopy approach for non-selfadjoint eigenvalue problems*, Numerische Mathematik **119** (2011), 557–583.

[14] J. Chang and M. Adams, *Analysis of Transport Synthetic Acceleration for Highly Heterogeneous Problems*, Proceedings of the 2003 Nuclear Mathematical and Computational Sciences Conference, 2003.

[15] M. Chu, *A Simple Application of the Homotopy Method to Symmetric Eigenvalue Problems*, Linear Algebra and its Applications **59** (1984), 85–90.

[16] J. Duderstadt and L. Hamilton, *Nuclear Reactor Analysis*, John Wiley and Sons Inc. 1976.

[17] Nuclear Energy Institute, *Nuclear Energy's Economic Benefits - Current and Future* (2013).

[18] B. Ganapol, *Analytical Benchmarks for Nuclear Engineering Applications*, Nuclear Energy Agency, 2008.

[19] K. Georg, *Matrix-Free Numerical Continuation and Bifurcation*, Numerical Functional Analysis and Optimization **22** (2001), no. 3-4.

[20] D. Gill and Y. Azmy, *Newton's Method for the Computation of k-Eigenvalues in Neutron Diffusion Theory*, Nuclear Science and Engineering **167** (2011).

[21] D. Gill, Y. Azmy, J. Warsa, and J. Densmore, *Newton's Method for the Computation of k-Eigenvalues in SN Transport Applications*, Nucl. Sci. Eng. **168** (2011), 36–58.

[22] V. Y. Gol'din, *A quasi-diffusion method for solving the kinetic equation*, USSR Comp. Math. and Math. Phys. **4** (1967), no. 136.

[23] H. Hanshaw, P. F. Nowak, and E. W. Larsen, *Stretched and Filtered Transport Synthetic Acceleration of S_N Problems. Part 2: Heterogeneous Media*, Trans. Am. Nucl. Soc. 2003.

[24] H. L. Hanshaw, P. F. Nowak, and E. W. Larsen, *Stretched and Filtered Transport Synthetic Acceleration of s_n problems. Part 1: Homogeneous Media*, Trans. Am. Nucl. Soc. 2003.

[25] J. He, *Homotopy perturbation technique*, Computer Methods in Applied Mechanics and Engineering **178** (1999), 257–262.

[26] Ji Huan He, *Comparison of homotopy perturbation method and homotopy analysis method*, Applied Mathematics and Computation **156** (2004), 527–539.

[27] Nuclear Energy Institute, *Nuclear Energy Just the Facts* (2012).

[28] I. Ipsen and C. Meyer, *The idea behind krylove methods*, American Mathematical Monthly **105** (1997), 889–899.

[29] D. Knoll, H. Park, and C. Newman, *Acceleration of k-eigenvalue / criticality calculations using the jacobian-free newton-krylov method*, Nuclear Science and Engineering **167** (2011).

[30] D. Knoll, H. Park, and K. Smith, *Application of the Jacobian-Free Newton-Krylov Method in Computational Reactor Physics*, Proceedings of the joint international conference on mathematics, computational methods and reactor physics, Saratoga Springs, New York, USA, 2009 May.

[31] D.A. Knoll and D.E. Keyes, *Jacobian-free newton–krylov methods: a survey of approaches and applications*, Journal of Computational Physics **193** (2004), 357–397.

[32] R. Koch and R. Becker, *Evaluation of Quadrature Schemes for the Discrete Ordinates Method*, Journal of Quantitative Spectroscopy and Radiative Transfer **84** (2004), 423–435.

[33] H. J. Kopp, *Synthetic method solution of the transport equation*, Nuclear Science and Engineering **17** (1963), no. 65.

[34] E. W. Larsen and J. E. Morel, *Asymptotic Solutions of Numerical Transport Problems in Optically Thick, Diffusive Regimes II*, J. Comp. Phys. **83** (1989), 212–236.

[35] E. W. Larsen, J. E. Morel, and Jr. W. F. Miller, *Asymptotic Solutions of Numerical Transport Problems in Optically Thick, Diffusive Regimes I*, J. Comp. Phys. **69** (1987), 283–324.

[36] E. Lewis and W. Miller, *Computational Methods of Neutron Transport*, American Nuclear Society, 1993.

[37] T. Li and Z. Zeng, *Homotopy-Determinant Algorithm for Solving Nonsymmetric Eigenvalue Problems*, Mathematics of Computation **59** (1992), no. 200, 483–502.

[38] O. Martin, *Boundary Problems for Stationary Neutron Transport Equations Solved by Homotopy Perturbation Method*, WSEAS Transactions on Mathematics **12** (2013May), no. 5.

[39] C. Moler, *Numerical computing with matlab*, Society for Industrial and Applied Mathematics, 2004.

[40] United Nations, *Comprehensive Nuclear-Test-Ban Tready* (1996).

[41] M. Oettli, *The Homotopy Method Applied to Eigenvalue Problems*, Eidgenoessische Technische Hochschule, Zurich, 1993.

[42] H. Park, D. Knoll, and C. Newman, *Nonlinear Acceleration of Transport Criticality Problems*, International Conference on Mathematics and Computational Methods, 2011May.

[43] M. Parks, E. de Sturler, G. Mackey, D. Johnson, and S. Maiti, *Recycling Krylov Subspaces for Sequences of Linear Systems*, SIAM Journal on Scientific Computing **28** (2006), 1651–1674.

[44] M. Rosa, J. Warsa, and M. Perks, *Implementation of a Cell-Wise Block-Gauss-Seidel Iterative Method for SN Transport on a Hybrid Parallel Computer Architecture*, Proceedings of the International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2011), 2011May.

[45] Y. Saad, *Chebyshev Acceleration Technniques for Solving Nonsymmetric Eigenvalue Problems*, Mathematics of Computation **42** (1984), 567–588.

[46] Y. Saad, *Numerical Methods fo Large Eigenvalue Problems*, Halstead Press, New York, NY, 1992.

[47] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM Journal on Scientific Computing **7** (1986).

[48] K. Smith and J. Rhodes III, *Full-core, 2-d LWR core calculations with CASMO-4E*, Physor 2002, 2002.

[49] G.W. Stewart, *On the sensitivity of the eigenvalue problem* $ax = \lambda bx$, Journal on Numerical Analysis **9** (1972Dec), no. 4, 669–686.

[50] L. N. Trefethen and D. Bau III, *Numerical Linear Algebra*, Society for Industrial and Applied Mathematics, 1997.

[51] J. Warsa, T. Wareing, and J. Morel, *On the Degraded Effectiveness of Diffusion Synthetic Acceleration for Multidimensional SN Calculations in the Presence of Material Discontinuities*, Proceedings of the 2003 Nuclear Mathematical and Computational Sciences Conference, 2003.

[52] J. S. Warsa, T. A. Wareing, and J. E. Morel, *Krylov iterative methods and the degraded effectiveness of diffusion synthetic acceleration for multidimensional s_n calculations in problems with material discontinuities*, Nucl. Sci. Engr. **147** (2004), no. 3, 218–248.

[53] J. S. Warsa, T. A. Wareing, J. E. Morel, and J. M. McGhee, *Krylov Subspace Iterations for the Calculation of k-Eigenvalues with S_N Transport Codes*, Nuclear Mathematical and Computational Sciences: A Century in Review, A Century Anew, Gatlinburg, TN, 2003, April.

[54] E. Wasserstrom, *Numerical Solutions by the Continuation Method*, Society of Industrial and Applied Mathematics Review **15** (1973), no. 1, 89–119.