HIGHER ORDER DISCONTINUOUS FINITE ELEMENT METHODS FOR DISCRETE ORDINATES THERMAL RADIATIVE TRANSFER

A Dissertation

by

PETER GREGORY MAGINOT

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

DOCTOR OF PHILOSOPHY

Chair of Committee,	Jean C. Ragusa
Co-Chair of Committee,	Jim E. Morel
Committee Members,	Marvin L. Adams
	Jean-Luc Guermond
Head of Department,	Yassin A. Hassan

August 2015

Major Subject: Nuclear Engineering

ABSTRACT

The linear discontinuous finite element method (LDFEM) is the current work horse of the radiation transport community. The popularity of LDFEM is a result of LDFEM (and its Q^1 multi-dimensional extensions) being both accurate and preserving the thick diffusion limit. In practice, the LDFEM equations must be "lumped" to mitigate negative radiation transport solutions. Negative solutions are non-physical, but are inherent to the mathematics of LDFEM and other spatial discretizations.

Ongoing changes in high performance computing (HPC) are dictating a preference for increased numbers of floating point operations (FLOPS) per unknown. Higher order discontinuous finite element methods (DFEM), those with polynomial trial spaces greater than linear, have been found to offer more accuracy per unknown than LDFEM. However, DFEM with higher degree trial spaces have received only limited attention due to their increased computational time per unknown, LDFEM's preservation of the thick diffusion limit, and the relative accuracy of LDFEM compared to other historical spatial discretizations. As solution methods evolve to make the most efficient use of HPC, it is possible that the increased computational work of higher order DFEM may become a strength rather than a hindrance.

For higher order DFEM to be useful in practice, lumping techniques must be developed to inhibit negative radiation transport solutions. We will show that traditional mass matrix lumping does not guarantee positive solutions and limits the overall accuracy of the DFEM scheme. To solve this problem, we propose a new, quadrature based, self-lumping technique. Our self-lumping technique does not limit solution order of convergence, improves solution positivity, and can be easily adapted to account for the within cell variation of interaction cross section. To test

and demonstrate the characteristics of our self-lumping methodology, we apply our schemes to several test problems: a homogeneous, source-free pure absorber; a pure absorber with spatially varying cross section; a model fuel depletion problem; and finally, we solve the grey thermal radiative transfer equations.

DEDICATION

To my wife, Kelli, for traveling with me along all the unforeseen curves of life.

To my mom and dad, for providing my foundation and inspiration.

ACKNOWLEDGEMENTS

I am thankful for the years of teaching and guidance provided by my co-chairs Dr. Jean Ragusa and Dr. Jim Morel; you convinced me long ago that a career in research was worth the extra challenges, costs, and years of work. I would also like to thank Dr. Marvin Adams for initially guiding me to the Computational Methods Group at Texas A&M and for encouraging me to consider Lawrence Livermore National Laboratory for post-graduation employment. Additionally, I thank I would like to thank Dr. Jean-Luc Guermond for your time and feedback on this dissertation and the master's thesis that preceeded it. Finally, I wish to thank the Department of Energy Computational Science Graduate Fellowship (administered by the Krell institute under grant number DE-FG02-97ER25308) for financial support during this work.

NOMENCLATURE

DFEM	Discontinuous finite element method
DSA	Diffusion synthetic acceleration
LDFEM	Linear discontinous finite element method
S2SA	S_2 synthetic acceleration
SL	Self-lumping
S_N	Discrete ordinates method
TL	Traditional lumping
TRT	Thermal radiative transfer
$\vec{\Omega}$	Particle direction
c	Speed of light
I	Photon radiation intensity
ψ	Neutron transport angular flux
T	Temperature
σ_t	Total interaction opacity
σ_s	Scattering interaction opacity
σ_a	Absorption opacity
B	Planck function
B_g	Planck function integrated across photon energy group g
$\underline{\mathrm{i}}$	Basis function i
Σ_t	Macroscopic neutron total interaction cross section
Σ_s	Macroscopic neutron scattering interaction cross section
Σ_a	Macroscopic neutron absorption interaction cross section
Σ_f	Macroscopic neuotrn fission interaction cross section

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

This dissertation is dedicated to the solution of the thermal radiative transfer (TRT) equations. The TRT equations:

$$\frac{1}{c}\frac{\partial I}{\partial t} + \vec{\Omega}\cdot\vec{\nabla}I + \sigma_t I = \int_0^\infty \int_{4\pi} \sigma_s(\vec{\Omega}' \to \vec{\Omega}, E' \to E) I d\vec{\Omega}' dE' + \sigma_a B \quad (1.1a)$$

$$C_v \frac{dT}{dt} = \int_0^\infty \sigma_a \left(\phi - 4\pi B\right) dE\,, \quad (1.1b)$$

are a nonlinear system of equations that describe the exchange of energy between a photon radiation field and a non-moving material. Radiation intensity, I, is a seven dimensional field dependent upon spatial location, $\vec{x}$; photon energy, E; photon direction of travel, $\vec{\Omega}$; and time t. c is the speed of light. Material opacities for all interactions, σ_t; absorption, σ_a; and scattering, σ_s are functions of photon energy and material temperature, T. Material heat capacity, C_v, is also a function of material temperature. The angle integrated radiation intensity, ϕ, is an integral over all photon directions of the the photon intensity and is a function of space and photon energy. Finally, the Planck function, B, is a function of photon energy and material temperature. While materials at all temperatures emit photon radiation, the radiation emission is proportional to T^4. Thus, solution of the radiative transfer equations is most important in situations where materials are very hot. Solving the thermal radiative transfer equations is an important component of the simulation of different scientific and engineering problems including astrophysics supernova explosions and high energy density physics laboratory experiments such as the ones conducted at the National Ignition Facility.

1.1 Simplifications of the Thermal Radiative Transfer Equations

In this dissertation, we make a number of simplifying assumptions to make solution of Eqs. (1.1) more tractable. First, we limit our focus to 1-D Cartesian (slab) geometry. The assumption of slab geometry is not required, but slab geometry radiation transport simulations require significantly less computational time. Further, any methods that have a possibility of being viable for radiation transport in multiple spatial dimensions must also work well in slab geometry.

Second, we approximate the continuous angle dependence of the intensity using the discrete ordinates (S_N) method. The S_N method approximates the true definition of the angle integrated intensity,

$$\phi(\vec{x}, E, t) = \int_{4\pi} I(\vec{x}, \vec{\Omega}, E, t) d\vec{\Omega}\,,$$

using quadrature integration,

$$\phi(\vec{x}, E, t) \approx \sum_{d=1}^{N_{dir}} w_d I(\vec{x}, \vec{\Omega}_d, E, t)\,. \tag{1.2}$$

In Eq. (1.2), $\{w_d, \vec{\Omega}_d\}_{d=1,\ldots N_{dir}}$ is the set of N_{dir} quadrature weights w_d and discrete directions, $\vec{\Omega}_d$ and corresponding intensities I_d.

Finally, we treat the photon energy dependence assuming a grey, or photon energy integrated model. The grey assumption assumes suitable opacities, σ_{grey} exist such

that

$$\int_0^\infty \sigma(E)I(E)dE = \sigma_{grey}I_{grey} \tag{1.3a}$$

$$\int_0^\infty \sigma(E)\phi(E)dE = \sigma_{grey}\phi_{grey} \tag{1.3b}$$

$$\int_0^\infty \sigma(E)B(E,T)dE = \sigma_{grey}B_{grey}(T)\,, \tag{1.3c}$$

where I_{grey}, ϕ_{grey}, and B_{grey} are photon energy integrated quantities,

$$I_{grey} = \int_0^\infty I(E)dE \tag{1.4}$$

$$\phi_{grey} = \int_0^\infty \phi(E)dE \tag{1.5}$$

$$B_{grey} = \int_0^\infty B(E,T)dE\,. \tag{1.6}$$

For the remainder of this work, we will forgo explicitly denoting quantities as grey, and unless otherwise noted, all quantities should be assumed to be photon energy integrated (grey). Though the assumption of Eqs. (1.3) does not hold unless all opacities are constant in energy, methods developed for the grey case are readily extensible to the multi-frequency treatment of photon energy dependence, and the multi-frequency approximation is by far the most common treatment of thermal radiative transfer photon energy dependence[1].

1.2 Spatial and Temporal Discretization

To complete a description of the approach we will take to solve Eqs. (1.1), we now describe how we will discretize the spatial and temporal variables.

1.2.1 Time Integration

The appearance of the speed of light in Eq. (1.1) results in the TRT equations being very stiff. To solve the such a stiff system of equations would require either an impractically small time step, or the use of implicit methods. We elect to use Diagonally Implicit Runge-Kutta (DIRK) methods to advance our TRT solution in time. The simplest of DIRK scheme is the first order implicit Euler scheme, but S-stable DIRK schemes with higher order convergence exist [2].

1.2.2 Spatial Discretization with Discontinuous Finite Elements

The linear discontinuous finite element method (LDFEM) has long been used to solve the discrete ordinates neutron transport equation [3]. Through manipulation, the thermal radiative transfer equations can be transformed into a form that is equivalent to the neutron transport equation with pseudo- scattering, fission, and fixed sources. This makes it possible to use the same methods and techniques developed for neutron transport to assist in solving the thermal radiative transfer equations. LDFEM has achieved wide spread acceptance in the neutron transport community because it is accurate [4] and highly damped. Because it possesses the thick diffusion limit [5], LDFEM has also been applied to the S_N TRT equations. Morel, Wareing, and Smith first considered the application of LDFEM to the S_N TRT equations in [6]. Mass matrix lumped LDFEM was shown to preserve the thick equilibrium diffusion limit [6]. This suggests that discontinuous finite element (DFEM) schemes can be used to accurately solve the TRT equations in both diffusive and transport effects dominated regions.

1.3 Progression Towards Higher Order DFEM Thermal Radiative Transfer

For higher order (quadratic and higher polynomial degree trial spaces) DFEM to be accurate and practical for solving Eqs. (1.1) we must demonstrate that higher order DFEM:

1. are "robust",
2. account for within cell spatial variation of opacity accurately, and
3. can be accelerated using appropriate iterative acceleration techniques.

By "robust", we mean that that calculated radiation outflow from a spatial cell is strictly positive for all cell widths and optical thicknesses.

In Chapter 2 we use a steady-state, mono-energetic, source-free pure absorber neutron transport problem with a cross section that is constant in space to examine the robustness of different radiation transport DFEM matrix lumping techniques. Next, we extend the techniques developed by Adams [7, 8], for a spatial discretization scheme related to LDFEM to address the within cell spatial variation of opacity, for higher order DFEM in Chapter 3 . Then, we examine iterative acceleration techniques compatible with higher order DFEM spatial discretizations in Chapter 4. In preparation for solving the coupled, non-linear TRT equations, in Chapter 5 we combine all of the strategies we have developed in Chapters 2-4 and apply them to a coupled system of linear equations to solve a two-group fuel depletion problem that uses explicit Euler time differencing. Finally, in Section 6 we solve the grey thermal radiative transfer equations using higher order DFEM, fully deriving the necessary equations, acceleration techniques, and providing numerical results that verify our newly developed DFEM methods and their implementation.

2. DISCONTINUOUS FINITE ELEMENTS FOR RADIATION TRANSPORT

In Chapter 1, we briefly mentioned that through manipulation, the thermal radiative transfer equations can be put into a form equivalent to the neutron transport equation with pseudo- scattering, fission, and fixed sources. We will fully demonstrate the linearization process in Chapter 6, but for now we take for granted that solving for the neutron transport equation's angular flux, ψ, is related to solving Eq. (1.1) for I. Additionally, we will assume that a steady-state, source-free, pure absorber neutron transport problem taxes DFEM schemes in a manner similar to the way DFEM schemes are tested in time-dependent thermal radiative transfer simulations, in particular Marshak wave type problems [9]. This chapter draws primarily from from our previously published work in [10] and [11].

2.1 History of DFEM for Neutron Transport

The linear discontinuous finite element method (LDFEM) for discrete ordinates neutron transport is widely used and has been extensively studied [12, 13, 14, 15]. However, the DFEM technique is not limited to linear trial spaces. Reed et al. [3] used arbitrary order DFEM S_N neutron transport in TRIPLET but, due to data storage limitations at the time, only LDFEM was computationally practical. As a result of these historical computing limitations, the accuracy of LDFEM [4], and LDFEM possessing the thick diffusion limit [5], the majority of reported DFEM radiation transport literature has focused on the LDFEM approximation. Higher order DFEM methods have received periodic attention; some older examples include the work of Walters [16] and Hennart and del Valle [17, 18]. More recent investigations of higher order DFEM trial spaces include those of Warsa and Prinja [19] and Wang and Ragusa [20, 21]. The primary focus of the work in [17, 18, 19, 20, 21] was the

convergence rate of arbitrary order DFEM schemes.

Negative angular flux solutions of the neutron transport equation obtained with LDFEM have been well documented in [13, 14, 15]. While these negativities do not affect the order of convergence and can be tolerated for certain applications [22], some nonlinear problems, particularly radiative transfer calculations, can diverge if the angular intensities are negative. As a result, several methods to eliminate or inhibit negative solutions have been developed and can be categorized into one of three categories: ad-hoc fix-ups [13], strictly non-negative solution representations [14], and matrix lumping [15]. The first two methods result in nonlinear systems of equations, while matrix lumping yields linear systems of equations. By definition, ad-hoc fix-ups and strictly non-negative solution representations yield non-negative outflows in 1-D, 2-D, and 3-D geometries, regardless of material properties. Mass matrix lumping (applied to LDFEM) yields strictly positive outflows only in 1-D geometries, though it does otherwise inhibit negativities [15].

Solution positivity of even degree unlumped DFEM methods for 1-D problems has been noted previously [16, 17, 18]. In comparing DFEM methods to nodal transport methods, Walters derived the quadratic DFEM scheme from the nodal transport equations using the Padé(2,3) approximation to the exponential term and noted that this approximation would result in a strictly positive outflow, regardless of cell optical thickness [16]. Hennart and del Valle then showed for slab geometry that all even P degree polynomial DFEM schemes approximate the cell outflow angular flux as a Padé$(P, P+1)$ function, which is a strictly positive approximation of the exponential [17, 18]. The positivity of even degree unlumped DFEM for 1-D problems was also shown in [19].

2.2 Mass Matrix Lumping Techniques

In this Section, we examine the idea of mass matrix lumping and its ability to ensure positive angular flux solutions of the neutron transport equation for arbitrary degree DFEM trial spaces in non-scattering 1-D slab geometries. We consider traditional lumping (TL), that constructs a diagonal mass matrix by collapsing all off-diagonal entries onto the main diagonal [15], and quadrature-based self-lumping (SL) methods [23], that yield a diagonal mass matrix by numerically integrating the DFEM equations using the DFEM interpolatory points as quadrature points. Restricting ourselves to equally-spaced interpolation points, self-lumping numerical integration with the greatest degree of accuracy is achieved through the use of closed Newton-Cotes formulae [24]. However, Newton-Cotes formulas with a large number of integration points are known to be oscillatory and are of relatively low-order accuracy, integrating polynomials at most of degree equal to the number of integration points. By considering solution representations that employ quadrature points as the interpolatory points, for example Gauss-Legendre (hereafter Gauss) or Lobatto-Gauss-Legendre (hereafter Lobatto) quadrature points [24], we wish to find methods that are self-lumping with a significantly higher accuracy. We analyze the combinations of Lagrange interpolatory points and numerical integration strategies given in Table 2.1 for positivity of the angular flux solution, local truncation error order, and spatial convergence order as a function of trial space polynomial degree. We limit the consideration of exact numerical integration schemes to those with equally-spaced interpolatory points, due to the fact that exact integration with any particular set of interpolatory points will always yield the same DFEM solution.

It has long been noted that traditional lumping (TL) with equally-spaced interpolatory points for 1-D LDFEM is equivalent to using the trapezoidal quadrature

Table 2.1: Nomenclature of numerical schemes.

Interpolation Point Type	Integration Strategy	Method Short Hand Name
Equally-Spaced	Exact spatial integration, with collapsing of mass matrix entries to the main diagonal	TL
Equally-Spaced	Numerical integration via Newton-Cotes quadrature restricted to interpolation points	SL Newton-Cotes
Gauss Quadrature	Numerical integration via Gauss quadrature restricted to interpolation points	SL Gauss
Lobatto Quadrature	Numerical integration via Lobatto quadrature restricted to interpolation points	SL Lobatto
Equally-Spaced	Exact spatial integration	Exact DFEM

rule to approximately integrate the mass matrix [25] while exactly integrating the gradient operator. Since the trapezoidal rule is identical to the closed Newton-Cotes formula with two points, we hypothesize that, for finite elements of arbitrary order using equally-spaced interpolatory points, traditional lumping is equivalent to using a closed Newton-Cotes formula to compute the mass matrix while exactly integrating the gradient operator. We demonstrate the equivalence between traditional lumping and closed Newton-Cotes formulae in the computation of the mass matrix.

Self-lumping (SL) based on Newton-Cotes formulae differs from traditional lumping in that SL Newton-Cotes generally does not exactly integrate the gradient operator. Coincidentally, the gradient operator is exactly integrated for linear/quadratic trial spaces using a 2-point/3-point Newton-Cotes formula, respectively. However, for higher degree polynomial trial spaces, the corresponding Newton-Cotes formula does not exactly integrate the gradient operator.

Self-lumping based on either Gauss or Lobatto quadratures exactly integrates the gradient operator in 1-D slab geometry for all degree of polynomial trial spaces; thus, there is no need to distinguish between exact integration and quadrature integration of the gradient operator for the SL Gauss and SL Lobatto schemes.

2.3 Lumping Techniques for the 1-D S_N Neutron Transport Equation with Arbitrary Order DFEM

We now derive the weak form of the 1-D S_N neutron transport equations discretized with DFEM and define the different mass matrix lumping techniques.

2.3.1 Weak Form Derivation

Consider the 1-D slab geometry S_N neutron transport equation:

$$\mu_d \frac{d\psi_d}{dx} + \Sigma_t \psi_d = Q_d \,, \tag{2.1}$$

where ψ_d is the angular flux $[1/[cm^2 - sec - ster]]$ in the μ_d direction, μ_d is the d'th directional cosine relative to the x-axis, Σ_t is the total interaction cross section $[cm^{-1}]$, and Q_d is a total source (fixed+scattering+fission) angular source in the direction of μ_d $[1/[cm^3 - sec - ster]]$. In all that follows, we consider only non-scattering, non-fissioning media (pure absorbers), thus Q_d will only be non-trivial if a fixed source is present in the problem. The scalar flux, ϕ $[n/cm^2 - sec]$, is defined as

$$\phi(x) = 2\pi \int_{-1}^{1} \psi(x, \mu_d)\ d\mu_d \,. \tag{2.2}$$

For simplicity, we derive the DFEM equations for a single-cell domain, with $x \in [x_L, x_R]$. A known angular flux, $\psi_{in,d}$, is defined on the incoming face of the domain for all μ_d. For $\mu_d > 0$, $\psi_{in,d}$ is defined only at x_L and for $\mu_d < 0$, $\psi_{in,d}$ is defined

at x_R. We begin our derivation by first transforming the physical geometry to a reference element, $s \in [-1, 1]$. This affine transformation is such that:

$$x = \bar{x} + \frac{\Delta x}{2} s \,, \tag{2.3a}$$

$$dx = \frac{\Delta x}{2} ds \,, \tag{2.3b}$$

with $\bar{x} = \frac{x_L + x_R}{2}$, and $\Delta x = x_R - x_L$. We seek a numerical approximation to the true angular flux ψ_d using Lagrange polynomials of degree P:

$$\psi_d(s) \approx \widetilde{\psi}_d(s) = \sum_{j=1}^{N_P} \psi_{j,d} b_j(s) \,, \tag{2.4}$$

where the $\widetilde{\psi}_d(s)$ denotes the numerical approximation. The basis functions b_j are the canonical Lagrange polynomials with interpolatory points s_j,

$$b_j(s) = \prod_{\substack{k=1 \\ k \neq j}}^{N_P} \frac{s - s_k}{s_j - s_k} \,, \tag{2.5}$$

and $N_P = P + 1$. To determine the N_P unknown coefficients of Eq. (2.4), we follow a standard discontinuous Galerkin procedure, successively multiplying Eq. (2.1) by weight function b_i and integrating by parts, hence generating N_P moment equations ($1 \leq i \leq N_P$). We assume that the cross sections are constant per cell. Inserting our solution representation $\widetilde{\psi}_d$, the i-th moment equation is given by:

$$\mu_d \left[b_i(1)\widetilde{\psi}_d(1) - b_i(-1)\widetilde{\psi}_d(-1) - \int_{-1}^{1} \widetilde{\psi}_d(s) \frac{db_i}{ds} \, ds \right] + \frac{\Delta x \Sigma_t}{2} \int_{-1}^{1} b_i(s) \widetilde{\psi}_d(s) \, ds = \frac{\Delta x}{2} \int_{-1}^{1} b_i(s) Q_d(s) \, ds \,. \tag{2.6}$$

We now introduce the upwind approximation to define the angular flux at the cell edges. For $\mu_d > 0$ the angular flux at the cell interfaces is

$$\widetilde{\psi}_d(-1) = \psi_{in,d} \text{ and} \tag{2.7a}$$

$$\widetilde{\psi}_d(1) = \sum_{j=1}^{N_P} \psi_{j,d} b_j(1)\,. \tag{2.7b}$$

Similarly for $\mu_d < 0$:

$$\widetilde{\psi}_d(-1) = \sum_{j=1}^{N_P} \psi_{j,d} b_j(-1) \text{ and} \tag{2.8a}$$

$$\widetilde{\psi}_d(1) = \psi_{in,d}\,. \tag{2.8b}$$

In Eq. (2.7a) and Eq. (2.8b), $\psi_{in,d}$ is either the known angular flux outflow from the upwind cell or a boundary condition. Inserting the definition of $\widetilde{\psi}_d(s)$, Eq. (2.6) becomes, for $\mu_d > 0$,

$$\mu_d \left[b_i(1) \left(\sum_{j=1}^{N_P} \psi_{j,d} b_j(1) \right) - b_i(-1)\psi_{in,d} - \int_{-1}^{1} \left(\sum_{j=1}^{N_P} \psi_{j,d} b_j(s) \right) \frac{db_i}{ds}\, ds \right] + \frac{\Delta x \Sigma_t}{2} \int_{-1}^{1} b_i(s) \left(\sum_{j=1}^{N_P} \psi_{j,d} b_j(s) \right) ds = \frac{\Delta x}{2} \int_{-1}^{1} b_i(s) Q_d(s)\, ds\,, \tag{2.9}$$

and, for $\mu_d < 0$,

$$\mu_d \left[b_i(1)\psi_{in,d} - b_i(-1) \left(\sum_{j=1}^{N_P} \psi_{j,d} b_j(-1) \right) - \int_{-1}^{1} \left(\sum_{j=1}^{N_P} \psi_{j,d} b_j(s) \right) \frac{db_i}{ds}\, ds \right] + \frac{\Delta x \Sigma_t}{2} \int_{-1}^{1} b_i(s) \left(\sum_{j=1}^{N_P} \psi_{j,d} b_j(s) \right) ds = \frac{\Delta x}{2} \int_{-1}^{1} b_i(s) Q_d(s) ds\,. \tag{2.10}$$

Considering all of the N_P moment equations at once we can write both Eq. (2.9) and Eq. (2.10) in a single matrix form:

$$(\mu_d \mathbf{G} + \Sigma_t \mathbf{M})\, \vec{\psi}_d = \vec{Q}_d + \mu_d \psi_{in,d} \vec{f}\,. \tag{2.11}$$

In Eq. (2.11) we have made use of the following definitions: the vector of unknowns is given by

$$\vec{\psi}_d = [\psi_{1,d} \ \ldots \ \psi_{N_P,d}]^T \ , \tag{2.12}$$

the mass matrix $\mathbf{M}$ is:

$$\mathbf{M}_{ij} = \frac{\Delta x}{2} \int_{-1}^{1} b_i(s) b_j(s)\ ds\,, \tag{2.13}$$

the fixed source moment vector, $\vec{Q}_d$, is a column vector of length N_P:

$$\vec{Q}_{d,i} = \frac{\Delta x}{2} \int_{-1}^{1} b_i(s) Q_d(s)\ ds\,, \tag{2.14}$$

and $\vec{f}$ is a column vector of length N_P:

$$\vec{f}_i = \begin{cases} b_i(-1) & \text{for } \mu_d > 0 \\ -b_i(1) & \text{for } \mu_d < 0 \end{cases} \ . \tag{2.15}$$

$\mathbf{G}$ is a $N_P \times N_P$ matrix which we refer to as the gradient operator. When $\mu_d > 0$, $\mathbf{G}$ is given by:

$$\mathbf{G}_{ij} = b_i(1) b_j(1) - \int_{-1}^{1} \frac{db_i}{ds} b_j(s)\ ds\,. \tag{2.16a}$$

For $\mu_d < 0$, $\mathbf{G}$ is:

$$\mathbf{G}_{ij} = -b_i(-1) b_j(-1) - \int_{-1}^{1} \frac{db_i}{ds} b_j(s)\ ds\,. \tag{2.16b}$$

When interpolatory points are not located at the cell interfaces (i.e., at $s = \pm 1$), it can be noted that

1. $\vec{f}$ has N_P non-zero entries and

2. $b_i(\pm 1)b_j(\pm 1) \neq 0$ for all $i, j = 1, \ldots, N_P$.

When a Lagrange interpolatory point exists on the cell edges, then $\vec{f}$ has only one non-zero entry and the product $b_i(\pm 1)b_j(\pm 1) \neq 0$ only when $i = j = N_P$ for $\mu_d > 0$ or when $i = j = 1$ for $\mu_d < 0$, as is the case when equally-spaced points or a Lobatto quadrature are used as interpolation points.

We evaluate the integrals of Eq. (2.13) and Eq. (2.16) using a numerical quadrature. A method exactly integrates a quantity when the quadrature rule used to evaluate the integral is accurate for polynomials of degree equal to or greater than the polynomial degree of the integrand. In general, the matrices are dense and their entries are computed as:

$$\mathbf{M}_{ij} \approx \frac{\Delta x}{2} \sum_{q=1}^{N_q} w_q b_i(s_q) b_j(s_q) \,, \tag{2.17}$$

$$\mathbf{G}_{ij} \approx sg(\mu_d) b_i(sg(\mu_d)) b_j(sg(\mu_d)) - \sum_{q=1}^{N_q} w_q \left. \frac{db_i}{ds} \right|_{s=s_q} b_j(s_q) \,, \tag{2.18}$$

where N_q is the number of quadrature points to be used, w_q are the weights associated with quadrature points s_q, and $sg(a)$ is the sign function defined as

$$sg(a) = \begin{cases} +1 & \text{if } a > 0 \\ -1 & \text{if } a < 0 \end{cases} \,. \tag{2.19}$$

2.3.2 Traditional Lumping

The traditional lumping (TL) scheme replaces $\mathbf{M}$ with $\widehat{\mathbf{M}}$, the latter being formed by collapsing row entries onto the main diagonal via the following formula [15]:

$$\widehat{\mathbf{M}}_{ij} = \begin{cases} \sum_{j=1}^{N_P} \mathbf{M}_{ij} & \text{for } \; i = j \\ 0 & \text{otherwise} \end{cases} . \tag{2.20}$$

2.3.3 Quadrature-Based Lumping

An alternative method of mass matrix lumping restricts the quadrature points to the interpolatory points where:

$$b_i(s_j) = \begin{cases} 1 & \text{if } s_i = s_j \\ 0 & \text{otherwise} \end{cases} , \; i = 1, \ldots, N_P \, , \tag{2.21}$$

and the quadrature integration of Eq. (2.17) reduces to:

$$\mathbf{M}_{ij} = \begin{cases} \frac{\Delta x}{2} w_i & i = j \\ 0 & \text{otherwise} \end{cases} . \tag{2.22}$$

As mentioned previously, we refer to the implicit lumping of Eq. (2.22) as self-lumping (SL). Self-lumping is a method to automatically generate a diagonal mass matrix. We note that self-lumping does not imply that the quadrature formula inexactly integrates the mass matrix.

2.3.4 Source Moment Evaluation

Historically, when discussing lumping techniques, the focus has been on matrix lumping [15] and little attention was paid to lumping source terms. For instance,

consider a δ-shaped volumetric sources (i.e., equal to 0 everywhere except at one given point). In such a case, the evaluation of $\vec{Q}_d$ using quadrature-based self-lumping schemes is an open question. Obviously, quadrature-based schemes cannot evaluate Eq. (2.14) for δ-sources. To address this, we expand the source on a Legendre polynomial basis:

$$\widehat{S}_d(s) = \sum_{n=0}^{P} S_n P_n(s) \tag{2.23a}$$

$$\text{with } S_n = \frac{2n+1}{2} \int_{-1}^{1} Q_d(s) P_n(s)\, ds\,, \tag{2.23b}$$

and evaluate $\vec{Q}_d$ as follows

$$\vec{Q}_{d,i} = \frac{\Delta x}{2} \int_{-1}^{1} b_i(s) \widehat{S}_d(s)\, ds\,. \tag{2.24}$$

Note that if the right-hand-side of Eq. (2.24) is exactly integrated, this is equivalent to exactly integrating Eq. (2.14).

2.4 Quadrature Point Selection

We now discuss the properties of different numerical quadratures as applied to the 1-D DFEM S_N neutron transport equations. We consider three different types of interpolatory points: equally-spaced, Gauss quadrature, and Lobatto quadrature. On the $[-1, 1]$ interval, the $N_P = P + 1$ equally spaced interpolation points for a degree P polynomial trial space are:

$$s_j = -1 + (j-1)\frac{2}{P}, \quad j = 1, \ldots, N_P\,. \tag{2.25}$$

Self-lumping using equally-spaced interpolation points requires numerical integration with closed Newton-Cotes quadrature formulae. The N_P weights, w_j, used for

Newton-Cotes numerical integration at the interpolation points do not follow a concise pattern, so we refer the reader to [24]. The Gauss quadrature points are the N_P roots of the Legendre polynomial, $P_{N_P}(s)$ [24]. The corresponding weights are:

$$w_j = \frac{2}{(1-s_j^2)}\left[P'_{N_P}(s_j)\right]^2 . \tag{2.26}$$

Lobatto quadrature points have fixed endpoints, $s_1 = -1$, $s_{N_P} = 1$. The remaining $N_P - 2$ points are the roots of $P'_{N_P-1}(s)$ [24], with corresponding weights:

$$w_j = \begin{cases} \frac{2}{N_P(N_P-1)} & j = 1,\ j = N_P \\ \frac{2}{N_P(N_P-1)\left[P_{N_P-1}(s_j)\right]^2} & \text{otherwise} \end{cases} . \tag{2.27}$$

The highest polynomial degree a particular self-lumping quadrature formula exactly integrates is given in Table 2.2 for Newton-Cotes, in Table 2.3 for Gauss, and Table 2.4 for Lobatto quadratures. Also listed in Table 2.2 - Table 2.4 is the maximum polynomial degree of the integrands present in the gradient and mass matrices.

Table 2.2: Accuracy of self-lumping closed Newton-Cotes quadratures for DFEM trial spaces of polynomial degree P.

Polynomial Degree of $\widetilde{\psi}$	$N_P = P+1$	Degree of **M** integrand	Degree of **G** integrand	Quadrature Accuracy
1	2	2	1	1
2	3	4	3	3
3	4	6	5	3
4	5	8	7	5
5	6	10	9	5
P	$P+1$	$2P$	$2P-1$	Odd $\widetilde{\psi}$: P Even $\widetilde{\psi}$: $P+1$

Table 2.3: Accuracy of self-lumping Gauss quadratures for DFEM trial spaces of polynomial degree P.

Polynomial Degree of $\widetilde{\psi}$	$N_P = P+1$	Degree of **M** integrand	Degree of **G** integrand	Quadrature Accuracy
1	2	2	1	3
2	3	4	3	5
3	4	6	5	7
4	5	8	7	9
5	6	10	9	11
P	$P+1$	$2P$	$2P-1$	$2P+1$

Table 2.4: Accuracy of self-lumping Lobatto quadratures for DFEM trial spaces of polynomial degree P.

Polynomial Degree of $\widetilde{\psi}$	$N_P = P+1$	Degree of **M** integrand	Degree of **G** integrand	Quadrature Accuracy
1	2	2	1	1
2	3	4	3	3
3	4	6	5	5
4	5	8	7	7
5	6	10	9	9
P	$P+1$	$2P$	$2P-1$	$2P-1$

Since the accuracy of an $N_P = P+1$ point Gauss quadrature integration exceeds the polynomial degree of the **M** and **G** integrands for a trial space of degree P, using the SL Gauss scheme will strictly yield the same numerical solution as any DFEM scheme that exactly integrates **M** and **G**. Thus, the SL Gauss scheme yields the same numerical solution as the Exact DFEM scheme.

For linear and quadratic trial spaces, self-lumping methods using either Lobatto or equally-spaced interpolation points will yield identical solutions. This is a direct result of the two- and three-point Lobatto quadrature formulae being identical to the two- and three-point closed Newton-Cotes quadratures. This equivalence does

not hold for higher degree polynomial trial spaces because the Lobatto quadrature points will no longer correspond to the equally-spaced quadrature points.

By definition, TL uses equally-spaced interpolation points and exactly integrates the gradient operator. For cell-wise constant cross sections, TL is equivalent to a numerical integration scheme that:

1. uses equally-spaced interpolation points,
2. integrates the gradient operator exactly, and
3. uses a Newton-Cotes quadrature restricted to the DFEM interpolation points to compute the mass matrix.

To prove the third point, consider the following. With traditional lumping, $\mathbf{M}_{ij}$ is exactly computed and then a row-sum operation is performed on the rows of $\mathbf{M}$; thus the entries of the diagonal mass matrix computed for TL are

$$\begin{aligned}\widehat{\mathbf{M}}_{ii} = \sum_{j=1}^{N_P} \frac{\Delta x}{2} \int_{-1}^{1} b_i(s) b_j(s)\, ds &= \frac{\Delta x}{2} \int_{-1}^{1} b_i(s) \left[\sum_{j=1}^{N_P} b_j(s)\right] ds \\ &= \frac{\Delta x}{2} \int_{-1}^{1} ds\, b_i(s) \qquad \forall i = 1, \ldots, N_P\,, \end{aligned} \tag{2.28}$$

because $\sum_j^{N_P} b_j(s) = 1 \;\forall\; s \in [-1, +1]$ by definition. The integral $\frac{\Delta x}{2}\int_{-1}^{1} b_i(s)\, ds$ is exactly integrated using a closed Newton-Cotes formula with $N_P = P+1$ points since $b_i(s)$ is a polynomial of degree P. Finally, when the b_i functions are defined using equally-spaced points, the use of a closed Newton-Cotes formula with N_P points yields

$$\widehat{\mathbf{M}}_{ii} = \frac{\Delta x}{2} \int_{-1}^{1} b_i(s)\, ds = \frac{\Delta x}{2} \sum_{q=1}^{N_P} w_q b_i(s_q) = \frac{\Delta x}{2} w_i\,, \tag{2.29}$$

because $b_i(s_q) = \delta_{iq}$. Thus, the diagonal mass matrix computed using TL contains the closed Newton-Cotes weights as diagonal entries and is equivalent to approximating

$\mathbf{M}$ using closed Newton-Cotes quadrature in Eq. (2.22). We also numerically verify this in Table 2.5 for polynomial degrees up to 4, assuming $\Delta x = 2$. For linear and

Table 2.5: Equivalence of traditional lumping and closed Newton-Cotes quadrature approximation of the mass matrix.

P	Exact Integration of $\mathbf{M}$	Row Sums of $\mathbf{M}$	Newton-Cotes w with $P+1$ points
1	$\begin{bmatrix} \frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{2}{3} \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$
2	$\begin{bmatrix} \frac{4}{15} & \frac{2}{15} & -\frac{1}{15} \\ \frac{2}{15} & \frac{16}{15} & \frac{1}{15} \\ -\frac{1}{15} & \frac{2}{15} & \frac{4}{15} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{3} \\ \frac{4}{3} \\ \frac{1}{3} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{3} \\ \frac{4}{3} \\ \frac{1}{3} \end{bmatrix}$
3	$\begin{bmatrix} \frac{16}{105} & \frac{38}{280} & -\frac{3}{70} & \frac{19}{840} \\ \frac{33}{280} & \frac{27}{35} & -\frac{27}{280} & -\frac{3}{70} \\ -\frac{3}{70} & -\frac{27}{280} & \frac{27}{35} & \frac{33}{280} \\ \frac{19}{840} & -\frac{3}{70} & \frac{38}{280} & \frac{16}{105} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{4} \\ \frac{3}{4} \\ \frac{3}{4} \\ \frac{1}{4} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{4} \\ \frac{3}{4} \\ \frac{3}{4} \\ \frac{1}{4} \end{bmatrix}$
4	$\begin{bmatrix} \frac{292}{2835} & \frac{296}{2835} & -\frac{58}{945} & \frac{8}{405} & -\frac{29}{2835} \\ \frac{296}{2835} & \frac{256}{405} & -\frac{128}{945} & \frac{256}{2835} & \frac{8}{405} \\ -\frac{58}{945} & -\frac{128}{945} & \frac{208}{315} & -\frac{128}{945} & -\frac{58}{945} \\ \frac{8}{405} & \frac{256}{2835} & -\frac{128}{945} & \frac{256}{405} & \frac{296}{2835} \\ -\frac{29}{2835} & \frac{8}{405} & -\frac{58}{945} & \frac{296}{2835} & \frac{292}{2835} \end{bmatrix}$	$\begin{bmatrix} \frac{7}{45} \\ \frac{32}{45} \\ \frac{4}{15} \\ \frac{32}{45} \\ \frac{7}{45} \end{bmatrix}$	$\begin{bmatrix} \frac{7}{45} \\ \frac{32}{45} \\ \frac{4}{15} \\ \frac{32}{45} \\ \frac{7}{45} \end{bmatrix}$

quadratic trial spaces, the 2-point and 3-point Newton-Cotes quadrature formulae exactly integrate the gradient operator, as shown in Table 2.2. Thus, for linear and quadratic trial spaces, schemes that use

1. equally-spaced interpolation points and traditional lumping,

2. equally-spaced interpolation points and self-lumping numerical integration, or

3. Lobatto quadrature interpolation points and self-lumping numerical integration,

will yield identical solutions.

2.5 Numerical Results

In this Section, we present numerical results for two 1-D slab problems. For the first problem, we consider a source-free pure absorber with vacuum boundary conditions on the right, a known angular flux $\psi_{in,d}$ incident on the left face, and a spatially constant total cross section Σ_t. The second problem consists of a slab with vacuum boundary conditions on both sides, no scattering, constant Σ_t, and a fixed δ-source.

For $\mu_d > 0$, the numerical approximations to the angular flux near the cell inflow and outflow are as follows:

$$\widetilde{\psi}_{in,d} = \sum_{j=1}^{N_P} \psi_{j,d} b_j(-1) \text{ and} \tag{2.30}$$

$$\widetilde{\psi}_{out,d} = \sum_{j=1}^{N_P} \psi_{j,d} b_j(1)\,. \tag{2.31}$$

Regardless of the sign of μ_d, the numerical approximation to the cell average angular flux is defined as:

$$\widetilde{\psi}_{A,d} = \frac{1}{2} \sum_{j=1}^{N_P} w_j \psi_{j,d}\,. \tag{2.32}$$

We used the following quadrature weight normalization: $\sum_{j=1}^{N_P} w_j = 2$. In Eq. (2.30), Eq. (2.31), and Eq. (2.32), $\psi_{j,d}$ are the components of $\vec{\psi}_d$, the numerical solution obtained by solving Eq. (2.11). Hence, the numerical angular flux solution of any of the previously discussed DFEM schemes can be obtained as a function of h, the number of mean free paths divided by μ_d,

$$h = \frac{\Sigma_t \Delta x}{\mu_d}\,. \tag{2.33}$$

2.5.1 Incident Flux Single-Cell Outflow Comparison

For the incident-flux problem, the analytical solution of Eq. (2.1) is:

$$\psi(x,\mu_d) = \begin{cases} \psi_{in,d} \exp\left[-\frac{\Sigma_t(x-x_L)}{\mu_d}\right] & \text{for } \mu_d > 0 \\ 0 & \text{for } \mu_d < 0 \end{cases} . \tag{2.34}$$

The analytic angular flux outflow, $\psi_{out,d} = \psi(x_R, \mu_d)$, is:

$$\psi_{out,d} = \psi_{in,d} \exp[-h] . \tag{2.35}$$

Similarly, the analytic average angular flux within the cell, $\psi_{A,d}$, is:

$$\psi_{A,d} = \frac{1}{\Delta x} \int_{x_L}^{x_R} \psi(x,\mu_d)\, dx = \frac{\psi_{in,d}}{h} \left(1 - \exp[-h]\right) . \tag{2.36}$$

The solution components are given by

$$\vec{\psi}_d = \psi_{in,d} \left(\mathbf{G} + \mathbf{M}\right)^{-1} \vec{f}, \tag{2.37}$$

and since $\mathbf{M} \propto h$, we can compare the various choices of interpolatory points and numerical integration strategies solely as a function of h.

Figures 2.1-2.4 show the numerically calculated cell outflow, $\widetilde{\psi}_{out,d}$, as a function of h for all methods considered. All methods converge to the analytical solution as $h \to 0$, thus we have zoomed in the range where the methods visually differ (i.e., $h \geq 1$). We observe that:

- SL Gauss yields strictly positive outflows for even degree polynomial trial spaces,

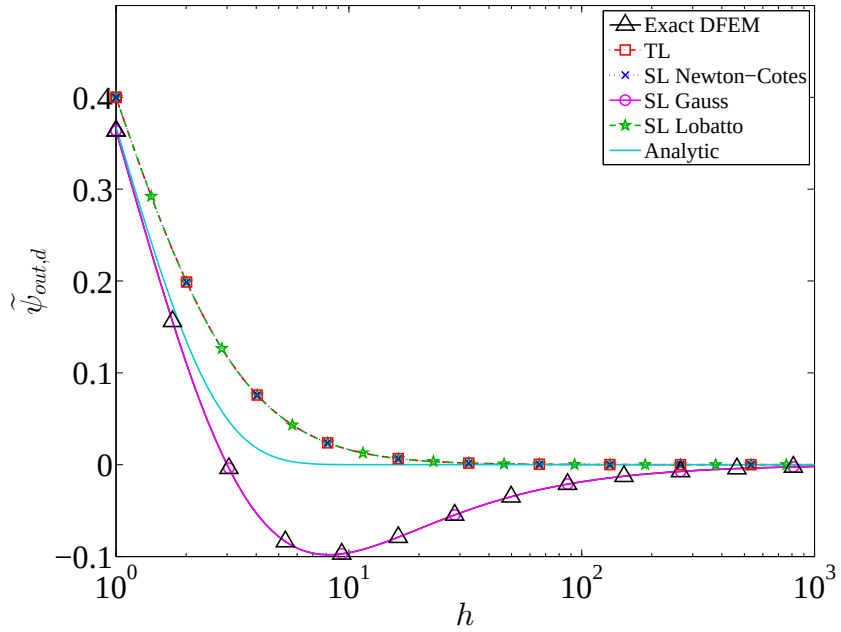

Figure 2.1: Numerical outflow values as a function of h, for a single cell homogeneous absorber problem with a linear DFEM trial space.

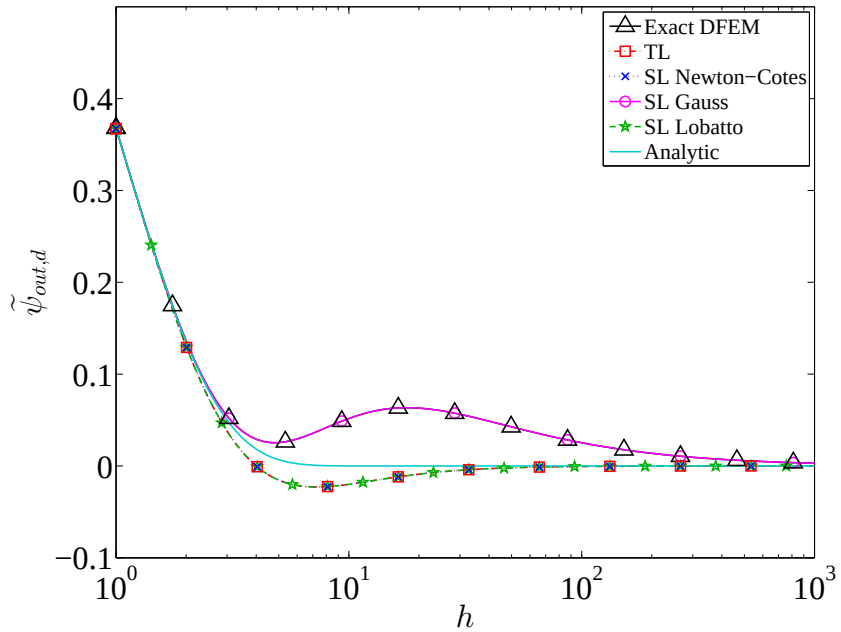

Figure 2.2: Numerical outflow values as a function of h, for a single cell homogeneous absorber problem with a quadratic DFEM trial space.

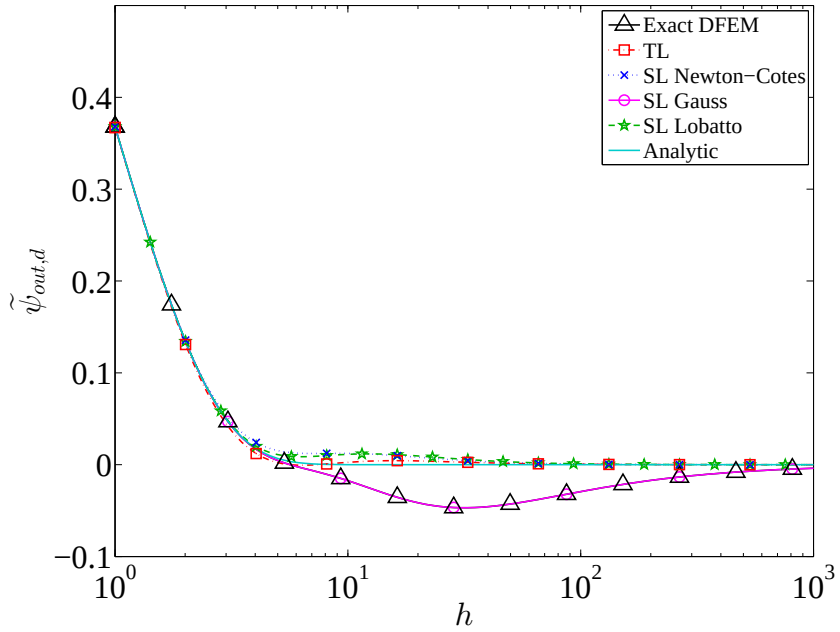

Figure 2.3: Numerical outflow values as a function of h, for a single cell homogeneous absorber problem with a cubic DFEM trial space.

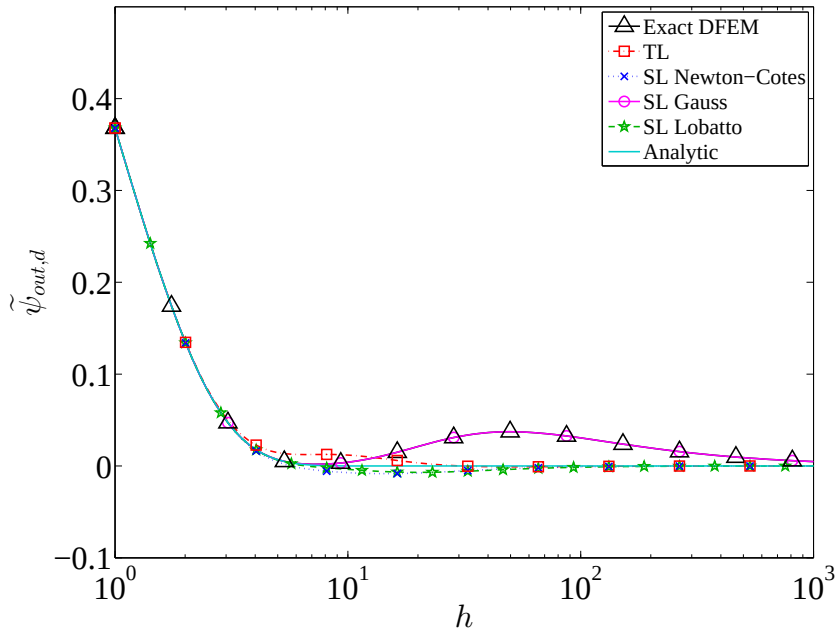

Figure 2.4: Numerical outflow values as a function of h, for a single cell homogeneous absorber problem with a quartic DFEM trial space.

- SL Lobatto and SL Newton-Cotes yield strictly positive outflows for odd degree polynomial trial spaces, and
- TL yields strictly positive outflows only for a linear trial space.

We also numerically verify the remarks made of Section 2.4, that is:

- SL Gauss is equivalent to Exact DFEM,
- SL Lobatto, SL Newton-Cotes, and TL are equivalent for linear and quadratic trial spaces, and
- for even degree trial spaces, the outflow value computed by SL Gauss is not monotonically decreasing as a function of h for cells of intermediate optical thickness (the same was noted in [19] for Exact DFEM).

2.5.2 Fixed Source Single-Cell Inflow Comparison

As noted in [14], it is possible for LDFEM to yield negative solutions near cell inflows for source driven problems. In this second problem, we use a δ-source:

$$Q_d(x) = \begin{cases} \delta(x - x_o) & \text{for } \mu_d > 0 \\ 0 & \text{for } \mu_d < 0 \end{cases}, \tag{2.38}$$

$x \in [-1, 1]$, and $-1 \leq x_o \leq 1$. The analytic solution to this problem for $\mu_d > 0$ is:

$$\psi(x, \mu_d) = \begin{cases} \exp\left[-\frac{\Sigma_t(x-x_o)}{\mu_d}\right] & x \geq x_o \\ 0 & x < x_o \end{cases}. \tag{2.39}$$

(For $\mu_d < 0$, $\psi(x, \mu_d) = 0$.) We now examine the numerical approximation to the angular flux near the cell inflow, $\widetilde{\psi}_{in,d}$, for various integration schemes, trial space degrees, and as a function of the ratio of the first Legendre moment of the source, S_1,

to the zero-th Legendre moment of the source, S_0. Note that the physical range of that ratio, $\frac{S_1}{S_0}$, is $[-3, 3]$, corresponding to a δ-source at the left cell edge ($\frac{S_1}{S_0} = -3$) or at the right edge ($\frac{S_1}{S_0} = 3$).

We first consider the case of a vacuum ($\Sigma_t = 0$), thus only testing the effect of quadrature accuracy in evaluating $\vec{Q}_d$ and $\mathbf{G}$. In Figs. 2.5-2.8, we plot $\widetilde{\psi}_{in,d}$ for three schemes:

1. Lobatto quadrature, which is exact for $\mathbf{G}$ and approximate for the source moments, Eq. (2.24) ,
2. Gauss quadrature: which is exact for both $\mathbf{G}$ and the source moments, and
3. Newton-Cotes quadrature: which is approximate for both $\mathbf{G}$ and the source moments.

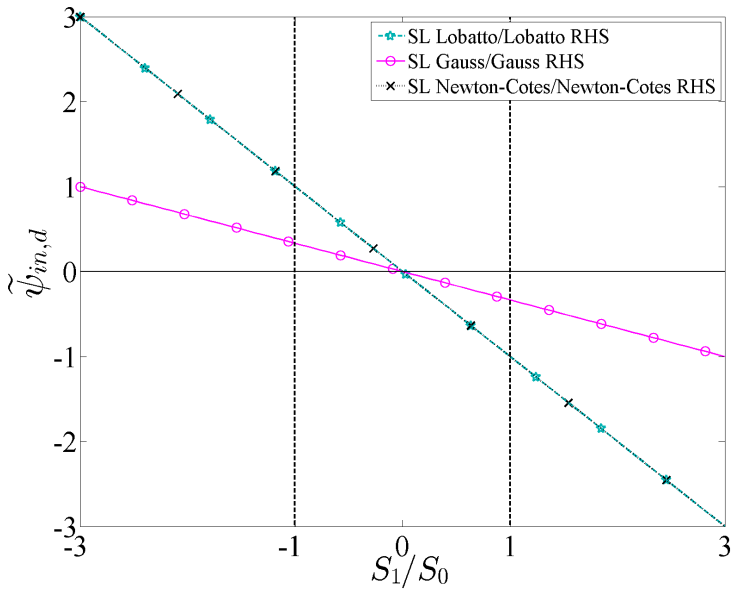

Figure 2.5: Numerical inflow values as a function of $\frac{S_1}{S_0}$ for a single cell (vacuum case) with a δ-shaped source, using linear DFEM.

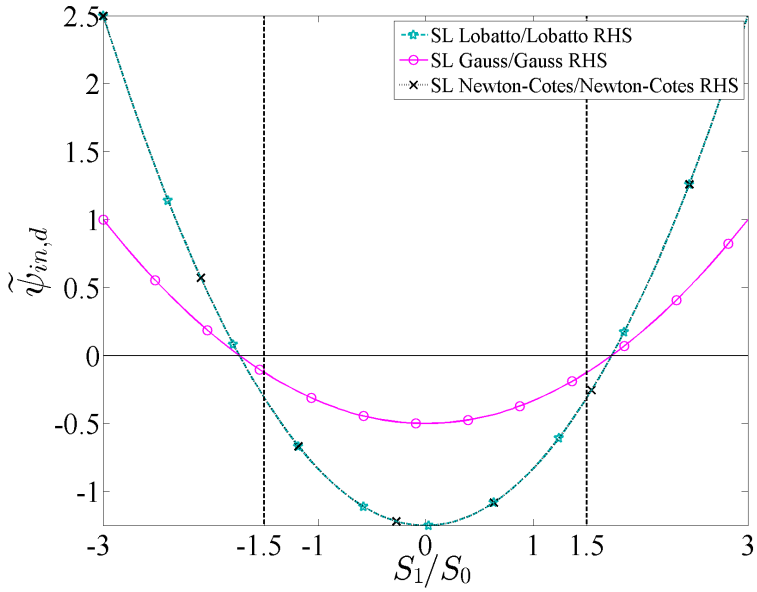

Figure 2.6: Numerical inflow values as a function of $\frac{S_1}{S_0}$ for a single cell (vacuum case) with a δ-shaped source, using quadratic DFEM.

The dotted vertical lines in Figs. 2.5-2.8 correspond to the extrema values of $\frac{S_1}{S_0}$ that yield a strictly positive polynomial source representation of degree P (indeed, the degree-P Legendre expansion of the δ-source is not everywhere positive for a wide range of possible $\frac{S_1}{S_0}$ that are physically realizable). For all trial space degrees, the Gauss scheme exhibits less negativity than either of the other two schemes. The dramatic difference between the Gauss scheme and the Lobatto scheme is solely due to the quadrature formula used to evaluate $\vec{Q}_d$ since both schemes exactly integrate $\mathbf{G}$. The Newton-Cotes scheme exhibits less severe negativities than the Lobatto scheme but is less robust than the Gauss scheme. Given the results shown in Figs. 2.5-2.8, we conclude that the most robust schemes exactly integrate the source moments, Eq. (2.24).

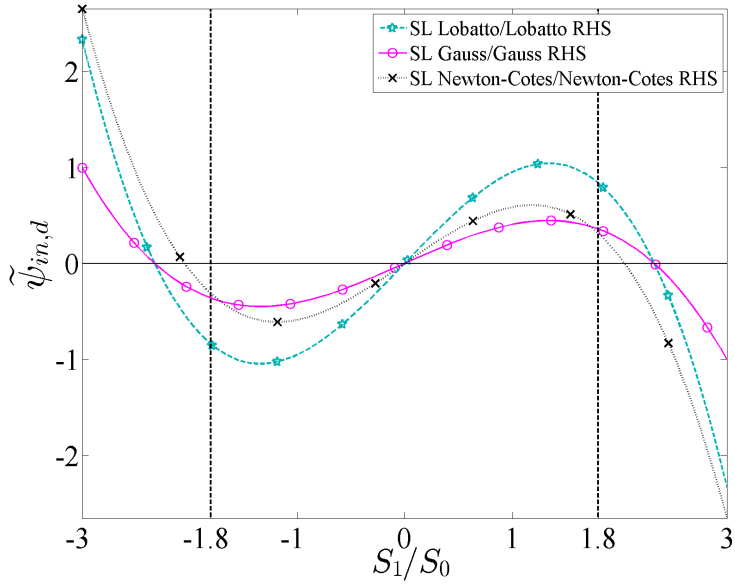

Figure 2.7: Numerical inflow values as a function of $\frac{S_1}{S_0}$ for a single cell (vacuum case) with a δ-shaped source, using cubic DFEM.

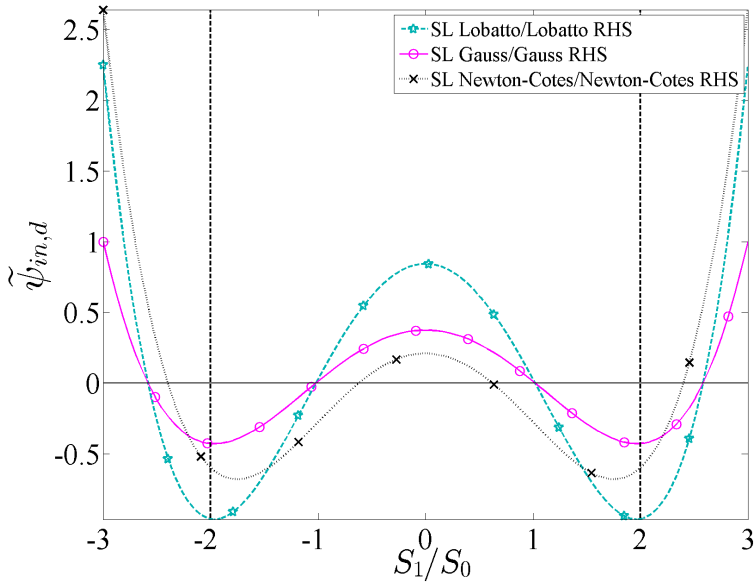

Figure 2.8: Numerical inflow values as a function of $\frac{S_1}{S_0}$ for a single cell (vacuum case) with a δ-shaped source, using quartic DFEM.

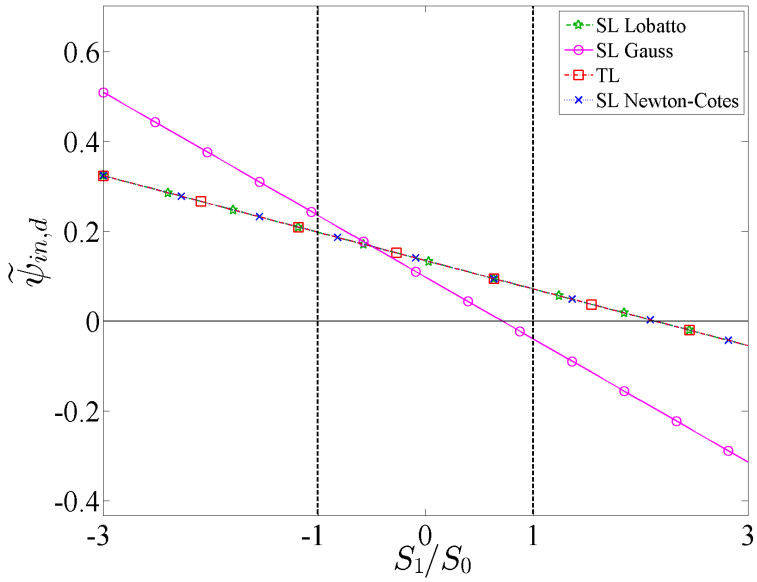

Figure 2.9: Numerical inflow values as a function of $\frac{S_1}{S_0}$, for a single cell (absorber case) with a δ-shaped source, using linear DFEM.

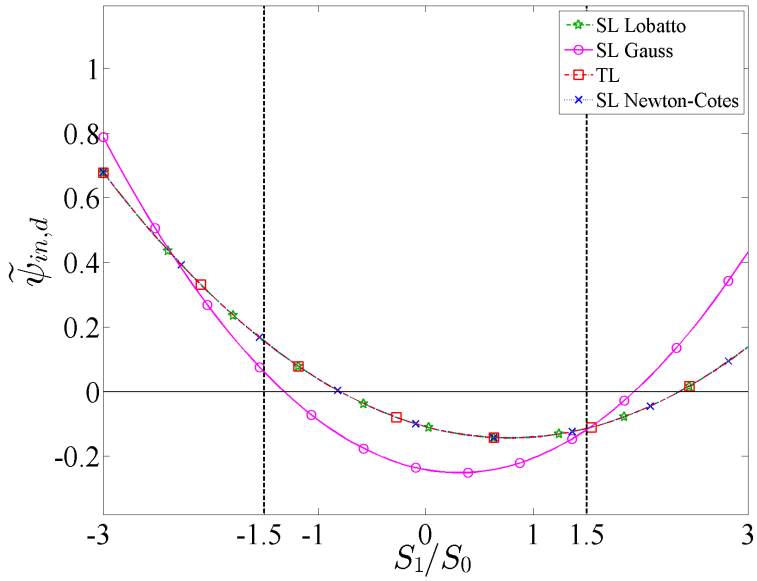

Figure 2.10: Numerical inflow values as a function of $\frac{S_1}{S_0}$, for a single cell (absorber case) with a δ-shaped source, using quadratic DFEM.

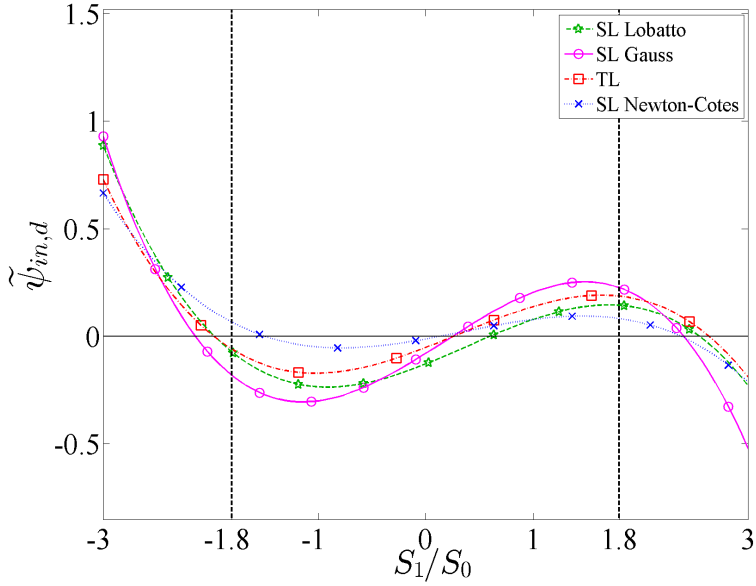

Figure 2.11: Numerical inflow values as a function of $\frac{S_1}{S_0}$, for a single cell (absorber case) with a δ-shaped source, using cubic DFEM.

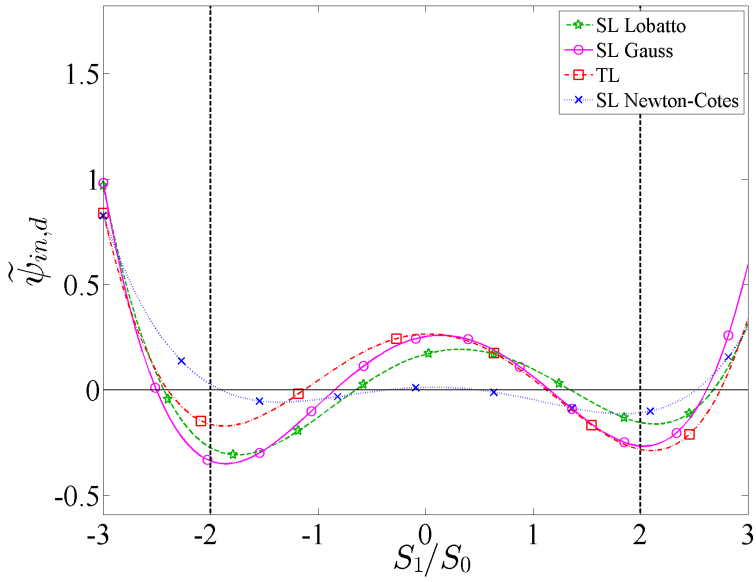

Figure 2.12: Numerical inflow values as a function of $\frac{S_1}{S_0}$, for a single cell (absorber case) with a δ-shaped source, using quartic DFEM.

In Figs. 2.9-2.12, we again examine the positivity of $\widetilde{\psi}_{in,d}$, but for a non-vacuum case. Total cell optical thickness was chosen to be 5 mean free paths in Figs. 2.9-2.12 because this value led to the clearest plots. The relative behaviors observed do not change with cell optical thickness, but using a thicker domain reduces the magnitude for the values of $\widetilde{\psi}_{in,d}$. All methods in Figs. 2.9-2.12 exactly integrate Eq. (2.24). Regardless of trial space chosen, all schemes exhibit some negativities, but the SL Gauss scheme exhibits the greatest negativities and oscillations. The SL Newton-Cotes scheme presents the least severe negativities.

2.5.3 Single-Cell Taylor Series Analysis

Next, we perform a local truncation error analysis by comparing the Taylor series expansions for the exact and numerical angular fluxes as a function of powers of h for the source-free, incident flux pure absorber problem. Matlab [26] has been employed to perform the symbolic Taylor series expansions about $h = 0$. We denote the Taylor-expanded quantities using the subscript T. The expansions for the analytical inflow, cell average, and outflow are given below:

$$\psi_{in,d,T} = \psi_{in,d} \tag{2.40a}$$

$$\psi_{A,d,T} = \psi_{in,d}\left(1 - \frac{h}{2} + \frac{h^2}{6} - \frac{h^3}{24} + \frac{h^4}{120} - \frac{h^5}{720}\ldots\right) \tag{2.40b}$$

$$\psi_{out,d,T} = \psi_{in,d}\left(1 - h + \frac{h^2}{2} - \frac{h^3}{6} + \frac{h^4}{24} - \frac{h^5}{120}\ldots\right). \tag{2.40c}$$

The Taylor expansions of the numerical analogues to the quantities in Eqs. (2.40) depend on the trial space polynomial degree, the choice of interpolatory points, and the numerical integration strategy. For brevity, we omit giving these numerical analogues. Table 2.6 gives the lowest order term for the difference between $\psi_{in,d,T}$ and the numerical analogs for the Exact DFEM and TL schemes. The same information

for the SL Newton-Cotes, SL Gauss, and SL Lobatto schemes is given in Table 2.7. The differences between $\psi_{A,d,T}$ and the respective numerical analogs are given in Table 2.8 for Exact DFEM and TL, and Table 2.9 for the SL Newton-Cotes, SL Gauss, and SL Lobatto schemes. Differences between $\psi out, d, T$ and the corresponding numerical analogs are given in Table 2.10 for Exact DFEM and TL and Table 2.11 gives the lowest order difference between the SL Newton-Cotes, SL Gauss, and SL Lobatto approximations of $\psi_{out,d,T}$. In Tables 2.6-2.11 all entries are listed as $q(C)$, to be read as "the difference between the analytic taylor expansion and the numeric analog is Ch^q with $h = \Sigma_t \Delta x/\mu$". Entries of "Machine Precision" in Tables 2.6-2.11 are meant to indicate that the difference between the analytic Taylor expansion and Taylor expansion of the numerical approximation was inconclusive due to all coefficients being within machine precision.

Table 2.6: Local truncation error analysis in $\widetilde{\psi}_{in,d}$ for a single cell problem with constant cross section, for Exact DFEM and TL.

Polynomial Degree of $\widetilde{\psi}$	Exact DFEM	TL
1	2 (2×10^{-1})	2 (5×10^{-1})
2	3 (2×10^{-2})	3 (4×10^{-2})
3	4 (1×10^{-3})	2 (7×10^{-2})
4	5 (7×10^{-5})	3 (1×10^{-2})
5	6 (3×10^{-6})	2 (5×10^{-2})
6	7 (1×10^{-7})	3 (1×10^{-2})
7	8 (4×10^{-9})	2 (5×10^{-2})

Table 2.7: Local truncation error analysis in $\widetilde{\psi}_{in,d}$ for a single cell problem with constant cross section, for SL Newton-Cotes, SL Gauss, and SL Lobatto.

Polynomial Degree of $\widetilde{\psi}$	SL Newton-Cotes	SL Gauss	SL Lobatto
1	2 (5×10^{-1})	2 (2×10^{-1})	2 (5×10^{-1})
2	3 (4×10^{-2})	3 (2×10^{-2})	3 (4×10^{-2})
3	2 (1×10^{-1})	4 (1×10^{-3})	4 (3×10^{-3})
4	3 (1×10^{-2})	5 (7×10^{-5})	5 (1×10^{-4})
5	2 (6×10^{-2})	6 (3×10^{-6})	6 (7×10^{-6})
6	3 (9×10^{-3})	7 (1×10^{-7})	7 (3×10^{-7})
7	2 (4×10^{-2})	8 (4×10^{-9})	8 (8×10^{-9})

Table 2.8: Local truncation error analysis in $\widetilde{\psi}_{A,d}$ for a single cell problem with constant cross section, for Exact DFEM and TL.

Polynomial Degree of $\widetilde{\psi}$	Exact DFEM	TL
1	3 (1×10^{-2})	2 (2×10^{-1})
2	5 (1×10^{-4})	4 (2×10^{-3})
3	7 (7×10^{-7})	3 (3×10^{-3})
4	9 (2×10^{-9})	5 (8×10^{-5})
5	11 (5×10^{-12})	3 (1×10^{-3})
6	13 (7×10^{-15})	5 (7×10^{-5})
7	Machine Precision	3 (1×10^{-3})

Table 2.9: Local truncation error analysis in $\widetilde{\psi}_{A,d}$ for a single cell problem with constant cross section, for SL Newton-Cotes, SL Gauss, and SL Lobatto

Polynomial Degree of $\widetilde{\psi}$	SL Newton-Cotes	SL Gauss	SL Lobatto
1	2 (2×10^{-1})	3 (1×10^{-2})	2 (2×10^{-1})
2	4 (2×10^{-3})	5 (1×10^{-4})	4 (2×10^{-3})
3	4 (6×10^{-4})	7 (7×10^{-7})	6 (1×10^{-5})
4	6 (8×10^{-6})	9 (2×10^{-9})	8 (5×10^{-8})
5	6 (2×10^{-6})	11 (5×10^{-12})	10 (1×10^{-10})
6	8 (2×10^{-8})	13 (7×10^{-15})	12 (2×10^{-13})
7	8 (3×10^{-9})	Machine Precision	Machine Precision

Table 2.10: Local truncation error analysis in $\widetilde{\psi}_{out,d}$ for a single cell with constant cross section, for Exact DFEM and TL.

Polynomial Degree of $\widetilde{\psi}$	Exact DFEM	TL
1	4 (1×10^{-2})	3 (2×10^{-1})
2	6 (1×10^{-4})	5 (2×10^{-3})
3	8 (7×10^{-7})	4 (3×10^{-3})
4	10 (2×10^{-9})	6 (1×10^{-2})
5	12 (5×10^{-12})	4 (1×10^{-3})
6	14 (7×10^{-15})	6 (7×10^{-5})
7	Machine Precision	4 (1×10^{-3})

Table 2.11: Local truncation error analysis in $\widetilde{\psi}_{out,d}$ for a single cell with constant cross section, for SL Newton-Cotes, SL Gauss, and SL Lobatto.

Polynomial Degree of $\widetilde{\psi}$	SL Newton-Cotes	SL Gauss	SL Lobatto
1	3 (2×10^{-1})	4 (1×10^{-2})	3 (2×10^{-1})
2	5 (2×10^{-3})	6 (1×10^{-4})	5 (2×10^{-3})
3	5 (6×10^{-4})	8 (7×10^{-7})	7 (1×10^{-5})
4	7 (8×10^{-6})	10 (2×10^{-9})	9 (5×10^{-8})
5	7 (2×10^{-6})	12 (5×10^{-12})	11 (1×10^{-10})
6	9 (2×10^{-8})	14 (7×10^{-15})	13 (2×10^{-13})
7	9 (3×10^{-9})	Machine Precision	Machine Precision

This local truncation error analysis illustrates the following.

1. Exact DFEM and SL Gauss, which are equivalent, exactly integrate the mass matrix, and are the most accurate,

2. TL does not guarantee increasing order of accuracy by using higher degree polynomial trial spaces,

3. TL converges at most third or fifth order for $\widetilde{\psi}_{A,d}$ and fourth or sixth order for $\widetilde{\psi}_{out,d}$ for odd or even polynomial trial spaces, respectively,

4. SL Newton-Cotes increases in accuracy with higher degree polynomial trial spaces, but only for $\widetilde{\psi}_{out,d}$ and $\widetilde{\psi}_{A,d}$,

5. TL and SL Newton-Cotes are at most second order or third order accurate for $\widetilde{\psi}_{in,d}$ for odd or even polynomial trial spaces, respectively,

6. SL Gauss is order $2P+1$ accurate in calculating $\widetilde{\psi}_{A,d}$ and order $2P+2$ accurate in calculating $\widetilde{\psi}_{out,d}$,

7. SL Lobatto is order $2P$ accurate in calculating $\widetilde{\psi}_{A,d}$ and order $2P+1$ in calculating $\widetilde{\psi}_{out,d}$,

8. SL Gauss, SL Lobatto, and Exact DFEM are accurate to order $P+1$ in calculating $\widetilde{\psi}_{in,d}$, and

9. SL Gauss is more accurate than SL Lobatto (smaller error constant) in computing $\widetilde{\psi}_{in,d}$, but not an order of h .

2.5.4 Convergence Rates for Spatially Discretized 1-D Domains

Here, we consider a homogeneous pure absorber material placed in a 1-D slab configuration and uniformly mesh the domain using N_{cells} cells. We use: $x \in [0, 10\ cm]$, $\Sigma_t = 1\ [cm^{-1}]$, no external sources, vacuum conditions on the right face of the slab, and a normally incident unit beam on the left face. The analytical solution to this problem is trivial to obtain:

$$\psi(x, \mu_d) = \begin{cases} \exp\left[-\Sigma_t x\right] & \mu_d = 1 \\ 0 & \text{otherwise} \end{cases} . \tag{2.41}$$

The L_2 norm of the error is:

$$E_\psi = \sqrt{\sum_{i=1}^{N_{cells}} \int_{x_{i-1/2}}^{x_{i+1/2}} \left(\psi(x,\mu_d) - \widetilde{\psi}_{d,i}(x)\right)^2 \, dx}\,, \tag{2.42}$$

where we recall that $\widetilde{\psi}_{d,i}(x)$ is the DFEM approximation of the angular flux in cell i. To evaluate the above integral, we use a high-order Gauss quadrature set ($x_{f,q}$, $w_{f,q}$) that employs a large number of quadrature points:

$$E_\psi \approx \sqrt{\sum_{i=1}^{N_{cells}} \frac{\Delta x_i}{2} \sum_{q=1}^{N_{qf}} w_{f,q} \left(\psi(x_{f,q},\mu_d) - \widetilde{\psi}_d(x_{f,q})\right)^2}\,. \tag{2.43}$$

Values of E_ψ shown here are calculated using $N_{qf} = 10$. In addition to the L_2 error, we also present the cell average angular flux error, E_{ψ_A}, defined as

$$E_{\psi_A} = \sqrt{\sum_{i=1}^{N_{cells}} \Delta x_i \left(\psi_{A,d,i} - \widetilde{\psi}_{A,d,i}\right)^2}\,, \tag{2.44}$$

and the cell outflow error, $E_{\psi_{out}}$, given by:

$$E_{\psi_{out}} = \sqrt{\sum_{i=1}^{N_{cells}} \Delta x_i \left(\psi(x_{i+1/2},\mu_d) - \widetilde{\psi}_{out,d,i}\right)^2}\,. \tag{2.45}$$

In Eq. (2.43), Eq. (2.44), and Eq. (2.45), Δx_i is the cell width of cell i and $\psi_{A,d,i}$ is the exact cell-averaged angular flux in cell i, which, for $\mu_d = 1$, is simply:

$$\psi_{A,d,i} = \exp[-\Sigma_t x_{i-1/2}] \frac{1}{\Delta x_i} \left(1 - \exp[-\Sigma_t \Delta x_i]\right)\,. \tag{2.46}$$

In the plots that follow, we omit plotting the errors of Exact DFEM since the Exact DFEM solution is identical to that of SL Gauss. For linear and quadratic polynomi-

als, we plot only SL Lobatto and omit plotting TL and SL Newton-Cotes since these methods yield identical solutions for linear and quadratic trial spaces. Figures

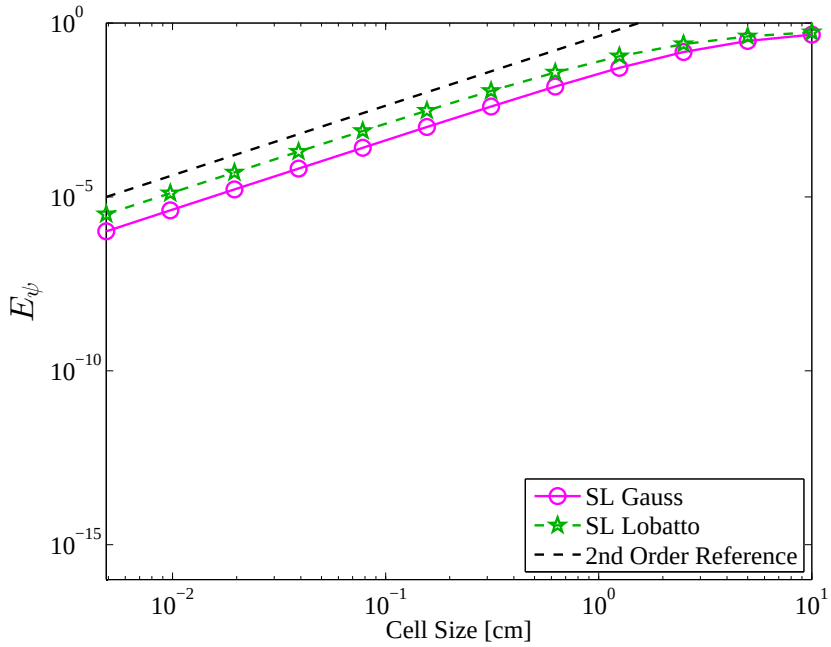

Figure 2.13: Convergence rate of the L_2 norm of the error, E_ψ, as a function of the mesh cell size for a pure absorber discretized with linear DFEM.

2.13-2.16 mirror the results of Table 2.6 and Table 2.7, which is expected since the convergence rate of E_ψ will be limited by the slowest converging local approximation which is $\widetilde{\psi}_{in,d}$. Similarly, Figs. 2.17-2.20 are the multiple-cell analogue of the local truncation error analysis of $\widetilde{\psi}_{A,d}$ given in Table 2.8 and Table 2.9. $E_{\psi_{out}}$, as shown in Figs. 2.21-2.24, does not converge at the local truncation error rates of Table 2.10 and Table 2.11.

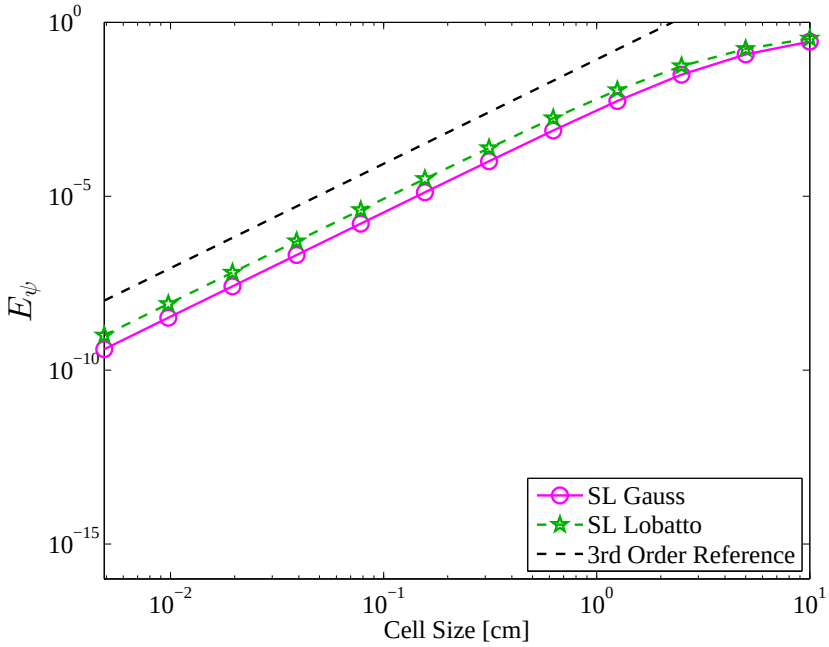

Figure 2.14: Convergence rate of the L_2 norm of the error, E_ψ, as a function of the mesh cell size for a pure absorber discretized with quadratic DFEM.

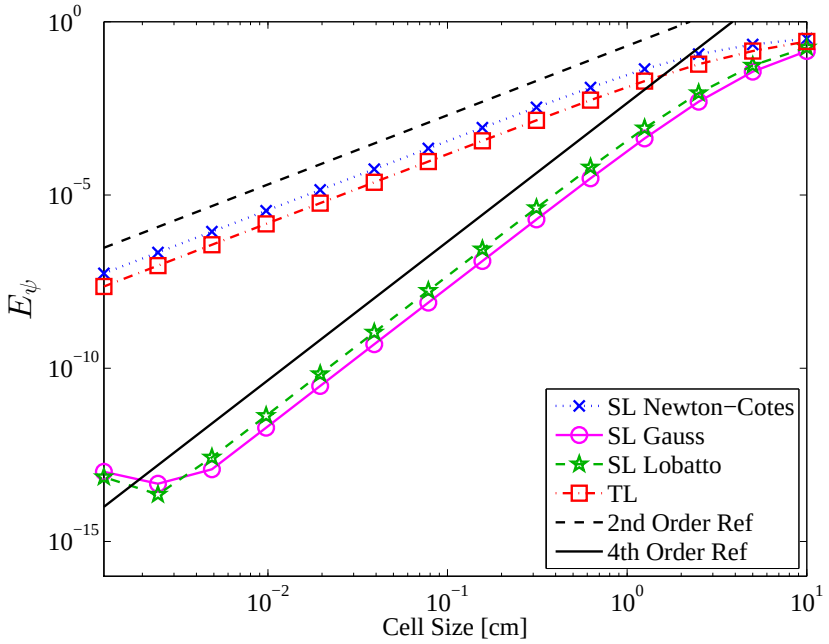

Figure 2.15: Convergence rate of the L_2 norm of the error, E_ψ, as a function of the mesh cell size for a pure absorber discretized with cubic DFEM.

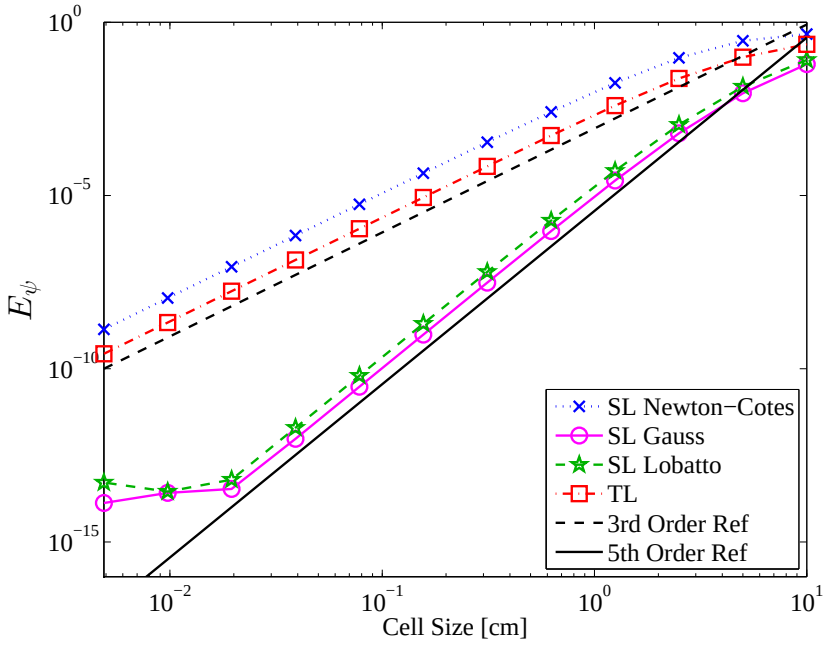

Figure 2.16: Convergence rate of the L_2 norm of the error, E_{ψ}, as a function of the mesh cell size for a pure absorber discretized with quartic DFEM.

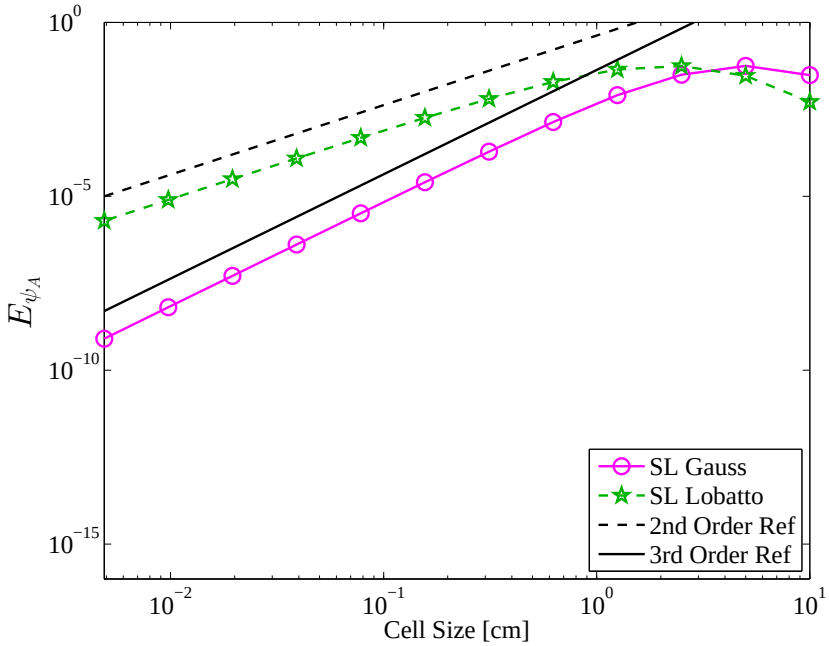

Figure 2.17: Convergence rate for $E_{\psi,A}$ as a function of the mesh cell size for a homogeneous pure absorber and linear DFEM.

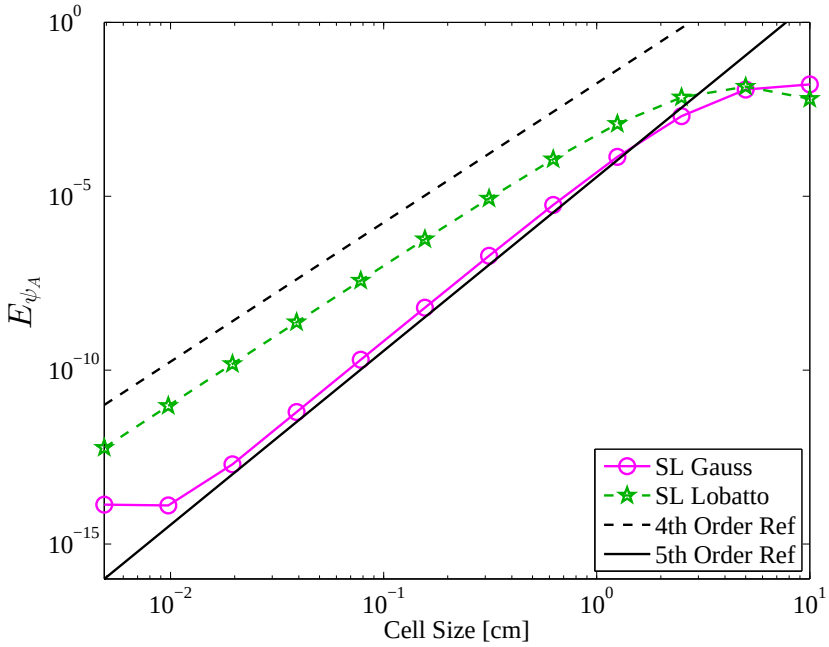

Figure 2.18: Convergence rate for $E_{\psi,A}$ as a function of the mesh cell size for a homogeneous pure absorber and quadratic DFEM.

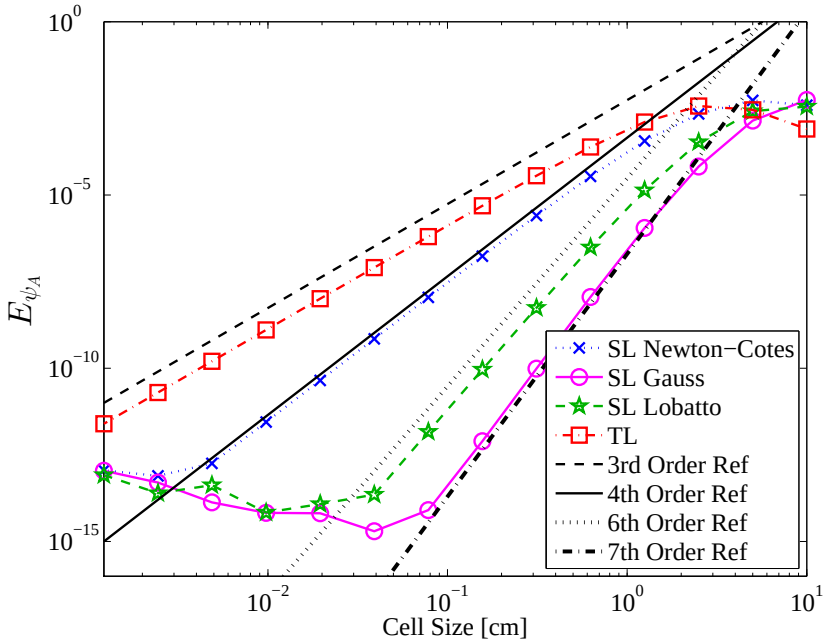

Figure 2.19: Convergence rate for $E_{\psi,A}$ as a function of the mesh cell size for a homogeneous pure absorber and cubic DFEM.

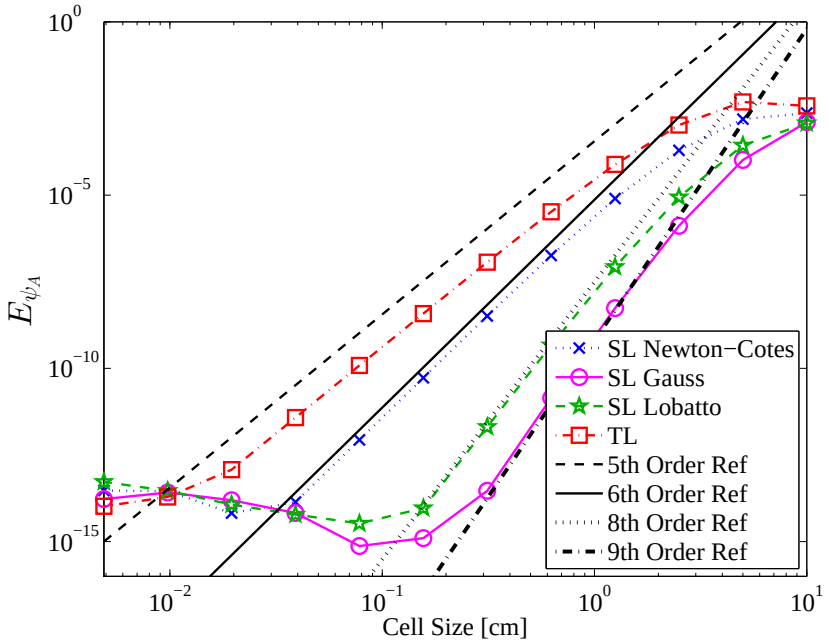

Figure 2.20: Convergence rate for $E_{\psi,A}$ as a function of the mesh cell size for a homogeneous pure absorber and quartic DFEM.

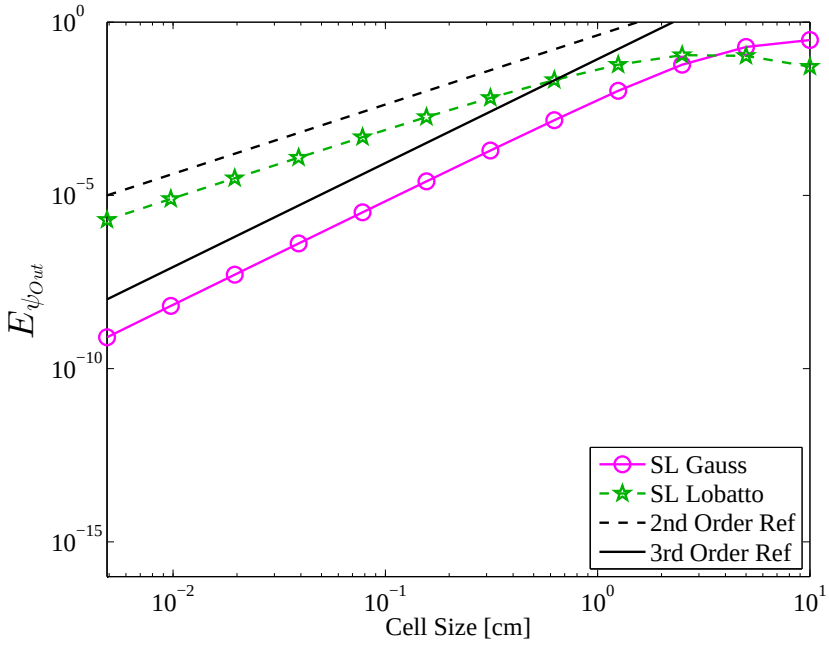

Figure 2.21: Convergence rate of $E_{\psi,out}$ as a function of the mesh cell size for a homogeneous pure absorber for linear DFEM.

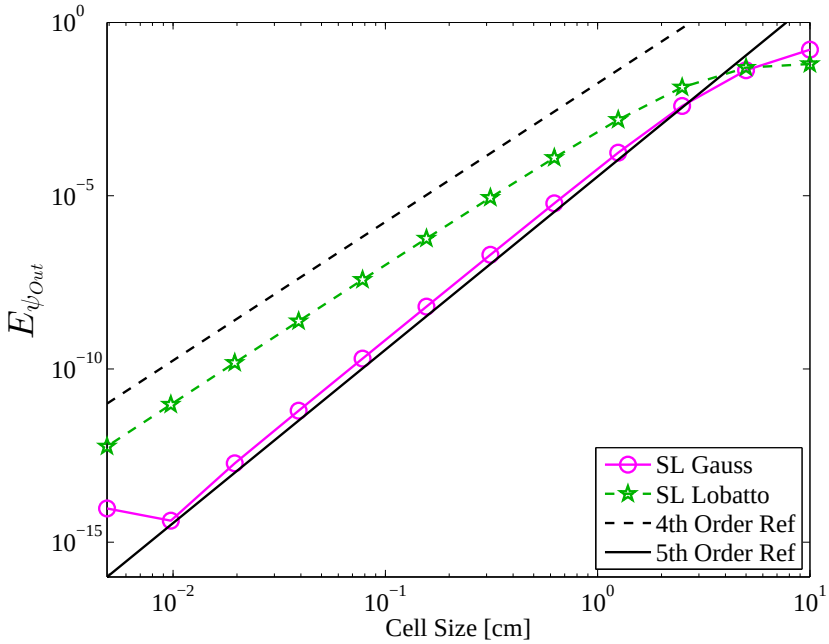

Figure 2.22: Convergence rate of $E_{\psi,out}$ as a function of the mesh cell size for a homogeneous pure absorber for quadratic DFEM.

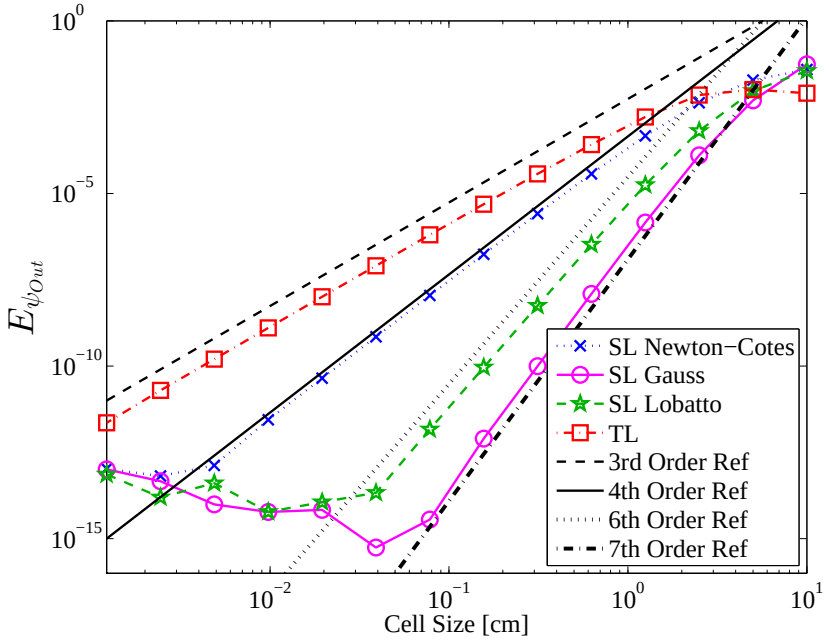

Figure 2.23: Convergence rate of $E_{\psi,out}$ as a function of the mesh cell size for a homogeneous pure absorber for cubic DFEM.

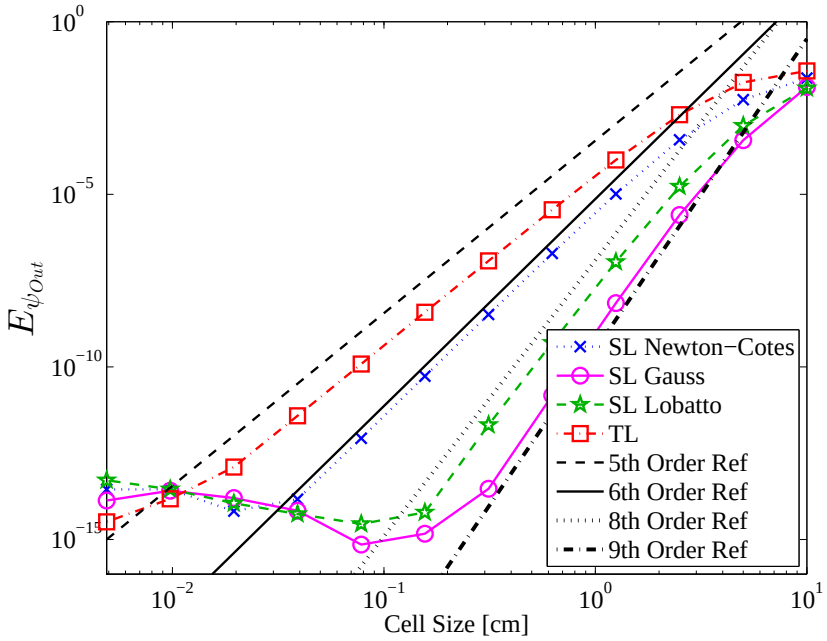

Figure 2.24: Convergence rate of $E_{\psi,out}$ as a function of the mesh cell size for a pure absorber for quartic DFEM.

The accumulation of errors in multiple-cell problems causes $E_{\psi_{out}}$ to globally converge one order of accuracy lower than the local truncation orders given in Table 2.10 and Table 2.11. It should be noted that the plateauing of errors E_{ψ}, E_{ψ_A}, and $E_{\psi_{out}}$ to values $\approx 10^{-14}$ in Figs. 2.13-2.16, Figs. 2.17-2.20, and Figs. 2.21-2.24, respectively, is simply a result of our numerical solutions being limited by machine precision (double precision).

2.6 Conclusions About Self-Lumping

We have shown that, for arbitrary degree polynomial trial space DFEM, a diagonal mass matrix does not necessarily ensure strictly positive angular flux outflow in a purely absorbing slab with spatially constant cross section. Indeed, the TL scheme was neither robust or accurate for polynomial trial space degree greater than linear. Also, we have shown that by using quadrature-based lumping schemes and choosing

DFEM interpolation points that are not equally spaced, robust, accurate polynomial DFEM schemes can be obtained. Based on the observed robustness, accuracy, and spatial convergence order results, we conclude that, for applications requiring robust solution techniques, the SL Lobatto scheme with odd degree polynomial trial space DFEM should be used to discretize the angular flux . If p-adaptivity is desired, software should be developed such that the ability to use either Lobatto (for odd trial space degrees) or Gauss (for even trial space degree) quadrature as the DFEM interpolation points is possible. However, given the non-monotonic behavior of the outflow angular flux as a function of the cell optical thickness when employing the SL Gauss scheme with even degree trial spaces for under-resolved problems, using SL Lobatto with an odd degree trial space would seem to be more accurate than using SL Gauss, despite SL Gauss being more accurate in the asymptotic (fine mesh) limit. Finally, though not as accurate SL Lobatto or SL Gauss, we will continue to consider SL Newton-Cotes due to its observed robustness.

3. DFEM METHODS FOR NEUTRON TRANSPORT FOR PROBLEMS WITH SPATIALLY VARYING CROSS SECTION

For many problems of interest to the nuclear science and engineering community, macroscopic cross sections in neutronics and opacities in radiative transfer calculations cannot accurately be described as piecewise constants in space. Cross sections and opacities are functions of continuously varying quantities such as temperature, density, burn-up history, etc. [27]. An example simulation that is not adequately described with cell-wise constant cross sections includes nuclear reactor isotopic depletion calculations. In thermal radiative transfer, interaction opacities can be rapidly varying functions of temperature. For example, consider Marshak wave problems and the canonical T^{-3} dependence [9] of absorption opacity, Across cells near the heated/cold material interface, opacity variations of several orders of magnitude are easily possible.

Historically, the neutron transport and thermal radiative transfer communities assumed interaction cross section and opacities, respectively, that were cell-wise constant [15, 1, 6]. Adams first described [7] and then presented computational results [8] for a "simple" corner balance (SCB) spatial discretization method that explicitly accounted for the spatial variation of opacity within individual spatial cells. The SCB scheme (which can be shown to be related to a LDFEM for certain geometries) accounts for opacity spatial variation within each cell via vertex-based quadrature evaluation. Similar strategies have been adapted to LDFEM radiative diffusion [9] and LDFEM TRT [28] calculations. For accurate TRT solutions, use of higher order DFEM will requires the development of corresponding higher order strategies for treating the within cell spatial variation of opacities. However, the majority of

neutron transport literature has only considered the case of cell-wise constant cross sections, see [15, 1, 29, 21]. The work of Kavenoky and Lautard [30] and more recently Santandrea and Bellier [31] are notable exceptions in neutron transport. In [30], continuous cubic finite element diffusion calculations that assume a linearly varying spatial cross section within each mesh cell were compared to results obtained using the same spatial discretization but with the assumption that cross sections are constant in each cell. Similarly, [31] compared the results of a linear characteristic scheme that assumes a linearly varying cross section in each spatial cell to those of a linear characteristic scheme that assumes a constant cross section in each cell.

In this chapter, we analyze the effects of cross-section spatial dependence on solution accuracy, as we did in our previously published work in [32]. Our work differs from [30, 31, 7, 8] by considering a discontinuous finite element (DFEM) spatial discretization of the slab geometry S_N transport equation using arbitrary degree polynomial finite element trial spaces. In addition, like [7] and [8] we do not make any approximation to the particular spatial shape of the cross-section spatial variation in each cell. We build on the quadrature integration ideas presented in Chapter 2 and employ a numerical quadrature to evaluate the mass matrix integrals that involve cross sections as a function of space. In general, the quadrature integration of the DFEM interaction term with arbitrary spatial cross section form will not be exact. However, we showed in Chapter 2 that exact computation of integrals appearing in the DFEM weak form, when cross sections are spatially constant, is not required to achieve high-order accuracy with high-order DFEM approximations. Building on this idea, we investigate the effects of using numerical quadratures to compute DFEM mass matrices, accounting for the spatial variation of cross section in space. As in Chapter 2 we use self-lumping numerical quadratures [23, 25], restricting quadrature integration points to the DFEM polynomial interpolation points. Results

are compared as a function of DFEM polynomial trial space degree and interpolation point type.

We demonstrate that assuming a piecewise constant cross section in each cell, when the cross section is not cell-wise constant in space, has several undesirable effects. Considering a source-free, purely absorbing medium, we show that DFEM schemes that assume a cell-wise constant cross section are at most second-order accurate for the angular flux solution and limited to at most first-order accuracy for the interaction rate solution, regardless of the DFEM polynomial trial space degree. We also show that assuming a piecewise constant cross section results in a highly discontinuous, non-monotonic spatial interaction rate. This phenomena has likely been present in published numerical results for problems with non-piecewise constant cross section but was not observed previously due to the choice of data presentation.

We then consider schemes that explicitly account for cross-section spatial variation within individual mesh cells. First, the positivity and robustness of different schemes are discussed using a source-free pure absorber problem. Next, we demonstrate that self-lumping schemes that evaluate the DFEM weak form integrals involving cross section with quadrature result in fully accurate schemes for arbitrary degree polynomial DFEM. By fully accurate we mean schemes that achieve the same order of convergence for problems with spatially varying and cell-wise constant cross section, for a given DFEM approximation order

3.1 Weak Form Derivation

We begin by repeating the DFEM neutron transport equation derived in Chapter 2:

$$\left(\mu_d \mathbf{G} + \mathbf{M}\right) \vec{\psi}_d = \vec{Q}_d + \mu_d \psi_{in,d} \vec{f}\,. \tag{3.1}$$

To account for the within cell variation of cross section, we need only make one change to Eq. (3.1). We introduce the concept of a reaction matrix, $\mathbf{R}_\Sigma$ where Σ is any interaction cross section or other material property:

$$(\mu_d \mathbf{G} + \mathbf{R}_{\Sigma_t})\,\vec{\psi}_d = \vec{Q}_d + \mu_d \psi_{in,d} \vec{f}\,. \tag{3.2}$$

The $N_P \times N_P$ reaction matrix, $\mathbf{R}_{\Sigma_t}$ is defined as:

$$\mathbf{R}_{\Sigma_t,ij} = \frac{\Delta x}{2} \int_{-1}^{1} \Sigma_t(s) b_i(s) b_j(s)\; ds\,. \tag{3.3}$$

Note that if Σ_t is indeed spatially constant within the mesh cell, there is no approximation in removing $\Sigma_t(s)$ from the integral of Eq. (3.3), giving

$$\mathbf{R}_{\Sigma_t,ij} = \frac{\Delta \Sigma_t}{2} \int_{-1}^{1} b_i(s) b_j(s)\; ds\,, \tag{3.4}$$

which is equivalent to

$$\mathbf{R}_{\Sigma_t} = \Sigma_t \mathbf{M}\,. \tag{3.5}$$

3.2 Numerical Schemes

We consider two classes of numerical methods in this paper. The first class uses exact spatial integration to evaluate the integrals that define $\mathbf{R}_{\Sigma_t}$. A second class of methods uses numerical quadrature to evaluate $\mathbf{R}_{\Sigma_t}$, $\mathbf{M}$, and $\mathbf{G}$. Specifically, we limit out discussion of quadrature-based integration to so called self-lumping methods [11]. Self-lumping methods, first discussed in [23, 25] for parabolic problems, use numerical quadrature restricted to the finite element interpolation points, and thus naturally yield diagonal mass matrices. A shorthand notation is given in Table 3.1 for all of the numerical methods considered in this chapter and described in detail in the

remainder of this section.

Table 3.1: Nomenclature of numerical schemes considered for the pure absorber problem with a spatially exponential cross section.

Interpolation Point Type	$\mathbf{R}_{\Sigma_t}$ Matrix Integration Strategy	Method Short Hand Name
Equally-Spaced	Exact Integration using true $\Sigma_t(x)$	EXS DFEM
Equally-Spaced	$\Sigma_t(x) \approx \widehat{\Sigma}_t$, $\mathbf{R}_{\Sigma_t} \approx \widehat{\Sigma}_t \mathbf{M}$ Exact Integration of $\mathbf{M}$	CXS DFEM
Equally-Spaced	Self-Lumping via Newton-Cotes Quadrature	SLXS Newton-Cotes
Lobatto Quadrature	Self-Lumping via Lobatto Quadrature	SLXS Lobatto
Gauss Quadrature	Self-Lumping via Gauss Quadrature	SLXS Gauss

3.2.1 Exact Spatial Integration

By exact spatial integration, we mean schemes that compute the entries of $\mathbf{M}$ and $\mathbf{G}$ exactly. Here, we achieve this by using equally-spaced interpolation points and employing a Gauss-Legendre quadrature rule [24] that exactly integrates the respective integrands of $\mathbf{M}$ and $\mathbf{G}$. Two schemes use exact spatial integration. One approximates the spatially varying cross section as a cell-wise constant cross section. The other uses the exact cross section when integrating the weak form DFEM quantities involving cross section. The scheme that assumes a cell-wise constant cross section represents the state of the practice in the neutron transport community,

while the second scheme represents the ideal scenario for DFEM transport schemes in problems with spatially varying cross sections.

3.2.1.1 Exact Cross Section

The exact cross section, exact spatial integration scheme (EXS DFEM) attempts to analytically integrate the full definition of $\mathbf{R}_{\Sigma_t}$. Note that since $\Sigma_t(x)$ can be an arbitrary function, analytic integration of $\mathbf{R}_{\Sigma_t}$ is in general impossible. Likewise, quadrature integration is unlikely to be exact. In our testing of the EXS DFEM scheme, we use a 20-point Gauss-Legendre quadrature to approximately integrate Eq. (3.3). Alternatively, adaptive quadrature, with a controllable tolerance, may be used such that the quadrature error in evaluating Eq. (3.3) could be reduced below some small tolerance.

3.2.1.2 Constant Cross Section

Historically, neutronics and some radiative transfer calculations have approximated spatially varying cross sections by assuming cell-wise constant cross sections [15, 1, 29, 6]. That is, some evaluation of the true $\Sigma_t(s)$ within a given cell is used to determine a constant value, $\hat{\Sigma}_t$, within each cell. Under this simplification, $\mathbf{R}_{\Sigma_t}$ is approximated as:

$$\mathbf{R}_{\Sigma_t} = \hat{\Sigma}_t \mathbf{M} \,. \tag{3.6a}$$

In our test problems, the constant cross section scheme (CXS DFEM) uses the volumetric average of $\Sigma_t(s)$ to generate $\hat{\Sigma}_t$:

$$\hat{\Sigma}_t = \frac{1}{2} \int_{-1}^{1} \Sigma_t(s) \; ds \,. \tag{3.7}$$

3.2.2 Self-Lumping Quadrature Integration

Schemes that are self-lumping evaluate the integrals of Eq. (3.3) using numerical quadrature. In Chapter 2 we showed that by definition, self-lumping schemes create diagonal mass matrices. Self-lumping schemes also create diagonal reaction matrices:

$$\mathbf{R}_{\Sigma_t,ij} = \begin{cases} w_i \frac{\Delta x}{2} \Sigma_t(s_i) & i = j \\ 0 & \text{otherwise} \end{cases} \tag{3.8}$$

Though the choice of interpolation points does not affect exact integration schemes, as shown in Chapter 2, the choice of interpolation points was shown to influence both the robustness and accuracy of self-lumping schemes. We consider equally-spaced closed Newton-Cotes, Lobatto-Gauss-Legendre, and Gauss-Legendre quadratures as interpolation points for self-lumping schemes. We do not expect any self-lumping scheme to exactly integrate $\mathbf{R}_{\Sigma_t}$, as the integrand defining $\mathbf{R}_{\Sigma_t}$ will generally not be a polynomial.

3.3 Pure Absorber Numerical Results

A beam of radiation, $\psi_{in}(\mu_d)$, is incident on the left face of the slab, the right face is a vacuum boundary, $x \in [0, x_R]$, and there are no fixed volumetric sources in the medium. We consider $\Sigma_t(x)$ to be of the form,

$$\Sigma_t(x) = c_1 e^{c_2 x}\,, \tag{3.9}$$

with c_1 and c_2 are constants $[cm^{-1}]$, with $c_1 > 0$ and $c_2 \neq 0$. The analytic angular flux solution for a source-free pure absorber with an exponentially varying cross section

is:

$$\psi(x,\mu) = \begin{cases} \psi_{in}(\mu)\exp\left[\frac{c_1}{\mu c_2}\left(1-e^{c_2 x}\right)\right] & \mu = \mu_d \\ 0 & \text{otherwise} \end{cases} . \tag{3.10}$$

By definition, the outflow angular flux from cell i, $\psi_{out,i}$ is $\psi(x_{i+1/2},\mu_d)$ and the average angular flux within cell i, $\psi_{A,i}$ as

$$\psi_{A,i} = \frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} \psi(x,\mu_d)\, dx\,, \tag{3.11}$$

with $\Sigma_t(x)$ defined as in Eq. (3.9). The analytical average flux value is:

$$\psi_{A,i} = \frac{\psi_{in}(\mu_d)}{\Delta x_i}\exp\left[\frac{c_1}{\mu_d c_2}\right]\left[E_1\left(\frac{c_1 e^{c_2 x_{i+1/2}}}{\mu_d c_2}\right) - E_1\left(\frac{c_1 e^{c_2 x_{i-1/2}}}{\mu_d c_2}\right)\right], \tag{3.12}$$

with E_1 the exponential integral [24].

3.3.1 Single Cell Outflow Comparisons

The only variable cross-section schemes that yields strictly positive angular outflows in a source-free pure absorber are the SLXS Lobatto and SLXS Newton-Cotes schemes using a linear trial space. For $\mu_d > 0$, consider a source-free, purely absorbing cell with known inflow, $\psi_{in}(\mu_d)$, of width Δx, and the total cross section at each interpolation point is $\Sigma_{t,j}$. Regardless of the actual functional form of the cross section within the cell, the linear DFEM SLXS Lobatto and SLXS Newton-Cotes schemes' numerical angular flux outflow, $\widetilde{\psi}_d(1)$, is:

$$\widetilde{\psi}_d(1) = \frac{2\mu_d^2\psi_{in}(\mu_d)}{2\mu_d^2 + \Delta x^2\Sigma_{t,1}\Sigma_{t,2} + \Delta x\mu_d\Sigma_{t,1} + \Delta x\mu_d\Sigma_{t,2}}\,. \tag{3.13}$$

Equation 3.13 is strictly positive when $\Sigma_t(x) \geq 0$, suggesting that the strictly positive outflow results observed in [11] might hold for an arbitrarily varying spatial cross sec-

tion. However, the results of [11] do not hold for higher-order DFEM approximations for spatially dependent cross sections.

To demonstrate that negative cell outflows are possible, we carry out the following test. In Figs. 3.1-3.4, we plot the angular flux outflow of each method as a function of trial space degree, and the parameter c_2. We hold the total cell optical thickness to 20 mean-free-path (MFP), vary $c_2 \in [1, 10]$, fix $x_R = 1$, and $\mu_d = 1$. With an exponential cross section, the cell optical thickness in MFP of a cell with $x \in [0, x_R]$ is:

$$MFP = \int_0^{x_R} \Sigma_t(x)\ dx = \frac{c_1}{c_2}\left(e^{c_2 x_R} - 1\right)\ . \tag{3.14}$$

To maintain a constant optical thickness in Figs. 3.1-3.4, c_1 is required to be:

$$c_1 = \frac{c_2\ MFP}{e^{c_2 x_R} - 1}\,. \tag{3.15}$$

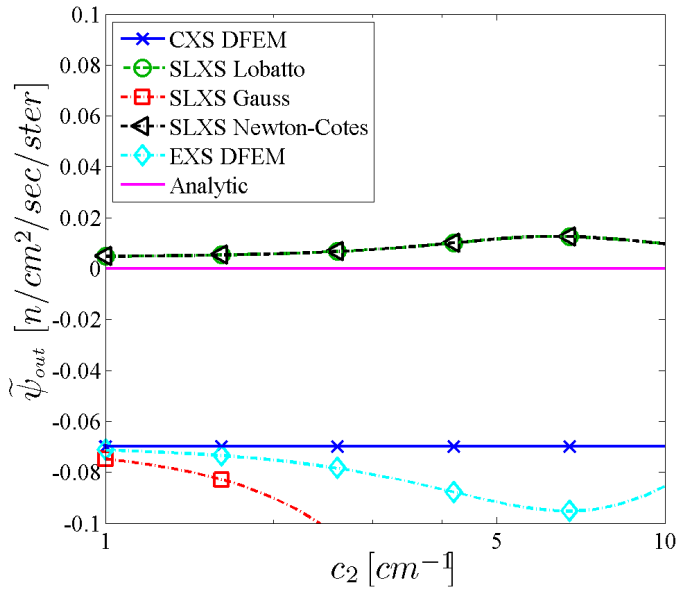

Figure 3.1: Numerical outflow from single cell pure absorber with $\Sigma_t(x) = c_1 e^{c_2 x}$, as a function of c_2 with constant optical thickness of twenty MFP, for a linear trial space.

Figures 3.1-3.4 confirms that SLXS Lobatto (and the equivalent SLXS Newton-Cotes scheme) with a linear trial space is the only scheme that explicitly accounts for the spatial variation of cross and maintains a strictly positive angular flux outflow regardless of the shape of $\Sigma_t(x)$. From Figs. 3.1-3.4 we also observe that $\widetilde{\psi}_{out}$ varies for every method as a function of the shape of $\Sigma_t(x)$, with the obvious exception of CXS DFEM. Considering that the analytic angular flux outflow is only a function of total cell MFP:

$$\psi_{out,i} = \psi_{in}(\mu_d) \exp\left[-\int_0^{x_{i+1/2}} \Sigma_t(x)\ dx/\mu_d\right] = \psi_{in}(\mu_d) \exp\left[-MFP\ /\mu_d\right] , \quad (3.16)$$

it is unphysical and undesirable that $\widetilde{\psi}_{out}$, for the SLXS Gauss, SLXS Lobatto, SLXS Newton-Cotes, and EXS DFEM schemes, depends on the spatial shape of $\Sigma_t(x)$.

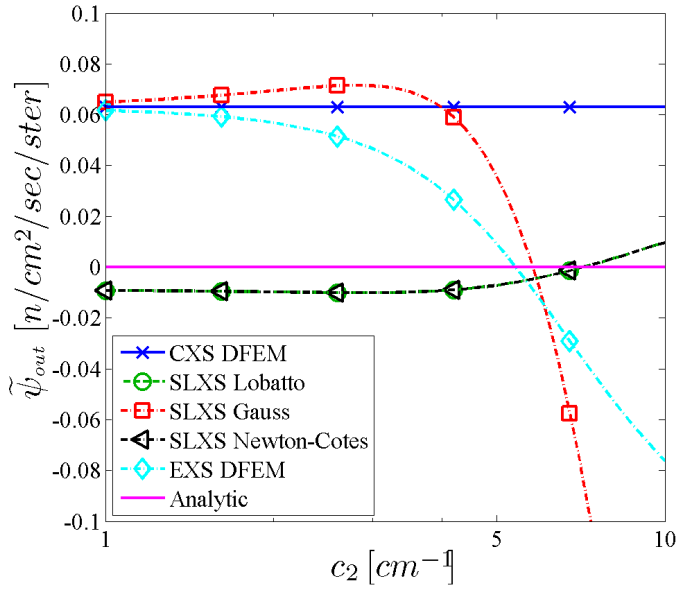

Figure 3.2: Numerical outflow from single cell pure absorber with $\Sigma_t(x) = c_1 e^{c_2 x}$, as a function of c_2 with constant optical thickness of twenty MFP, for a quadratic trial space.

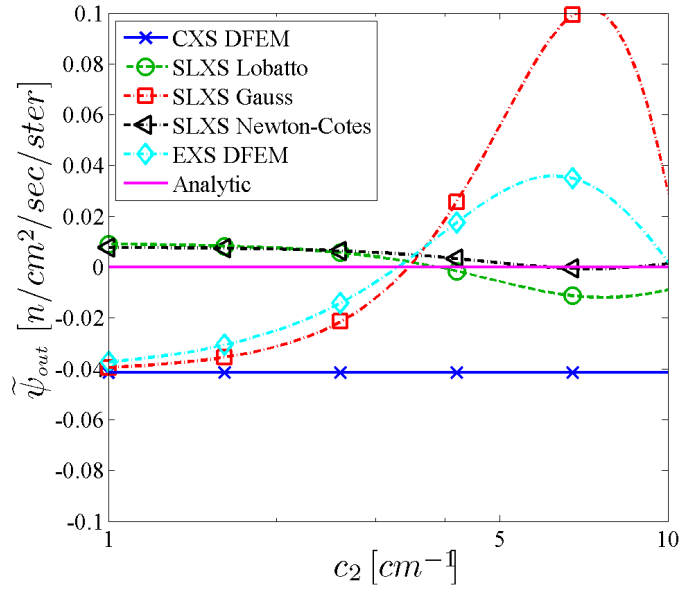

Figure 3.3: Numerical outflow from single cell pure absorber with $\Sigma_t(x) = c_1 e^{c_2 x}$, as a function of c_2 with constant optical thickness of twenty MFP, for different degree polynomial trial spaces.

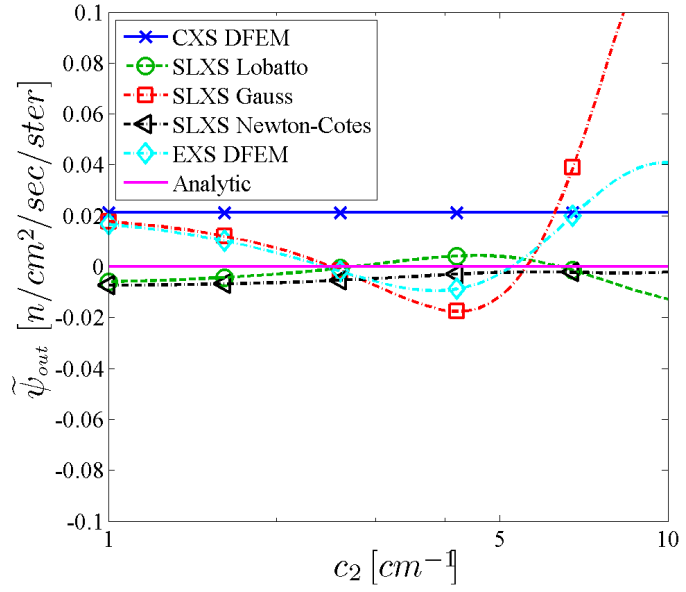

Figure 3.4: Numerical outflow from single cell pure absorber with $\Sigma_t(x) = c_1 e^{c_2 x}$, as a function of c_2 with constant optical thickness of twenty MFP, for different degree polynomial trial spaces.

3.3.2 Multiple Cell Spatial Convergence Rates

We now consider the order of spatial convergence for the following schemes: CXS DFEM, SLXS Gauss, SLXS Lobatto, and SLXS Newton-Cotes. Since exact integration of $\mathbf{R}_{\Sigma_t}$ is generally not feasible, we no longer consider the EXS DFEM scheme. Convergence results of the following angular flux errors as a function of the polynomial approximation order are presented:

$$E_{\psi} = \sqrt{\sum_{i=1}^{N_{cells}} \int_{x_{i-1/2}}^{x_{i+1/2}} \left(\widetilde{\psi}_d(x) - \psi(x,\mu_d)\right)^2 \, dx} \tag{3.17a}$$

$$E_{\psi_A} = \sqrt{\sum_{i=1}^{N_{cells}} \Delta x_i \left(\widetilde{\psi}_{A,i} - \psi_{A,i}\right)^2} \tag{3.17b}$$

$$E_{\psi_{out}} = \sqrt{\sum_{i=1}^{N_{cells}} \Delta x_i \left(\widetilde{\psi}_{out,i} - \psi(x_{i+1/2},\mu_d)\right)^2}. \tag{3.17c}$$

In Eqs. (3.17), Δx_i is the width of cell i, cell i spans $[x_{i-1/2}, x_{i+1/2}]$, $\widetilde{\psi}_d(x)$ is the DFEM numerical approximation, $\psi(x,\mu_d)$ is the analytic solution (see Eq. (3.10)). The problem is spatially discretized using N_{cells} spatial cells of equal width. We approximate the integrals defining the L^2 norm of the angular flux error, E_{ψ}, using a high-order Gauss quadrature set, $(w_{f,q}, s_{f,q})$, with N_{qf} points, such that:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \left(\widetilde{\psi}(x) - \psi(x,\mu_d)\right)^2 \, dx \approx \frac{\Delta x_i}{2} \sum_{q=1}^{N_{qf}} w_{f,q} \left(\widetilde{\psi}(s_{f,q}) - \psi(s_{f,q},\mu_d)\right)^2. \tag{3.18}$$

For the results that follow, $N_{qf} = 10$. We recall the definitions for the numerical approximations of the cell average angular flux, $\widetilde{\psi}_{A,i}$, and the outflow angular flux,

$\widetilde{\psi}_{out,i}$:

$$\widetilde{\psi}_{A,i} = \frac{1}{2}\sum_{j=1}^{N_P} w_j \psi_{i,j} \tag{3.19a}$$

$$\widetilde{\psi}_{out,i} = \sum_{j=1}^{N_P} \psi_{i,j} b_j(1)\,. \tag{3.19b}$$

We also consider the convergence of the numerical interaction rate, $\widetilde{IR}(x)$ to the true interaction rate $IR(x)$. First, we define the analytic reaction rate for our beam problem:

$$IR(x) = \Sigma_t(x)\psi(x,\mu_d)\,. \tag{3.20}$$

Similarly, we define a cell average interaction rate as:

$$IR_{A,i} = \frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} \Sigma_t(x)\psi(x,\mu_d)\; dx\,. \tag{3.21}$$

Defining a point-wise numerical approximation, $\widetilde{IR}(x)$, to the analytic interaction rate for the self-lumping schemes presents a unique problem, since only a a numerical quadrature is used to approximate the integrand of $\mathbf{R}$. Quadrature integration only requires point evaluations of $\Sigma_t(x)$, not knowledge of $\Sigma_t(x)$ in between quadrature points. However, for the purpose of plotting the SLXS schemes, we define:

$$\widetilde{IR}(s) = \sum_{j=1}^{N_P} b_j(s)\psi_{j,d}\Sigma_t(s_j)\,. \tag{3.22}$$

We approximate the cell average interaction rate in cell i as:

$$\widetilde{IR}_{A,i} = \frac{1}{2}\sum_{j=1}^{N_P} w_j \Sigma_t(s_j)\psi_{j,d}\,. \tag{3.23}$$

In Eq. (3.23), $\Sigma_t(s_j) = \hat{\Sigma}_t$ for the CXS DFEM scheme, and $\Sigma_t(s_j)$ is the point evaluation of the true cross section for all other schemes.

We consider two measures to assess the error of the DFEM schemes' approximation of the true interaction rate, $IR(x)$. The first, E_{IR} is an approximation of the L^2 norm of interaction rate error:

$$E_{IR} = \sqrt{\sum_{i=1}^{N_{cells}} \frac{\Delta x_i}{2} \sum_{q=1}^{N_P} w_q \left(IR(s_q) - \Sigma_t(s_q)\widetilde{\psi}(s_q, \mu_d) \right)^2}. \tag{3.24}$$

We reiterate that, for the self-lumping schemes, $\widetilde{IR}(s)$ is only truly defined at the DFEM interpolation points. E_{IR_A} measures the convergence of the average interaction rate:

$$E_{IR_A} = \sqrt{\sum_{i=1}^{N_{cells}} \Delta x_i (IR_{A,i} - \widetilde{IR}_{A,i})^2}. \tag{3.25}$$

For our convergence study, we consider a source-free purely absorbing slab with a cross section that varies exponentially in space as in Eq. (3.9) with $c_1 = 0.1$ and $c_2 = 2\ln(10)$. A beam of radiation is incident on the left face in the direction of $\mu_d = 1$, vacuum boundary conditions exist on the right face of the slab, and $x \in [0, 1]$. The convergence of the E_ψ, E_{ψ_A}, and $E_{\psi_{out}}$ as a function of the choice of numerical scheme for linear through quartic trial space polynomial degree are given in Figs. 3.5-3.8, Figs. 3.9-3.12, and Figs. 3.13-3.16, respectively.

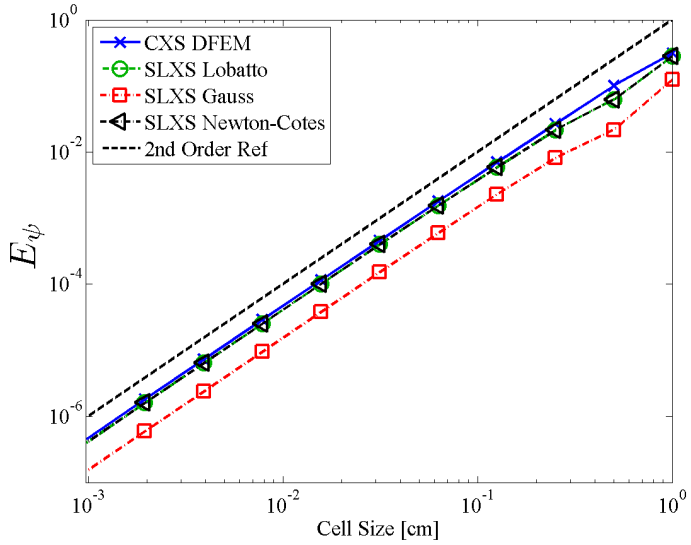

Figure 3.5: Convergence of E_ψ for a pure absorber with exponentially varying cross section discretized with linear DFEM.

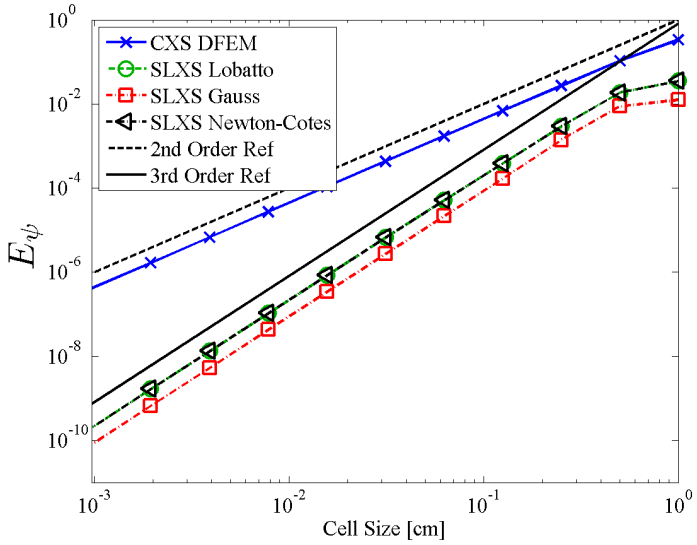

Figure 3.6: Convergence of E_ψ for a pure absorber with exponentially varying cross section discretized with quadratic DFEM.

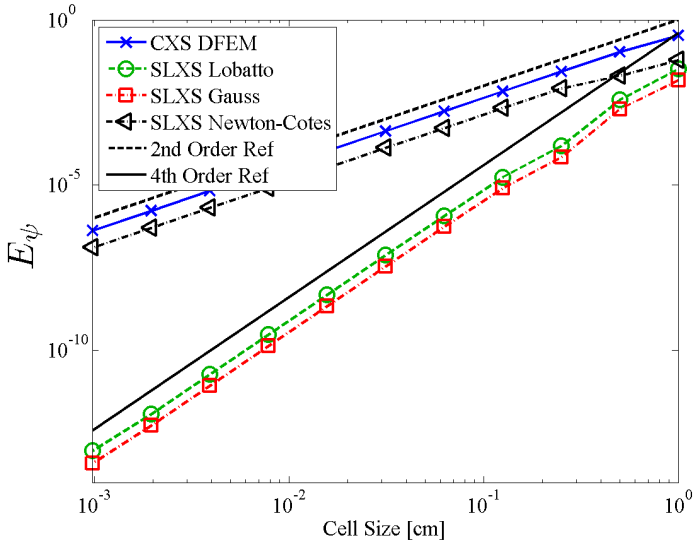

Figure 3.7: Convergence of E_ψ for a pure absorber with exponentially varying cross section discretized with cubic DFEM.

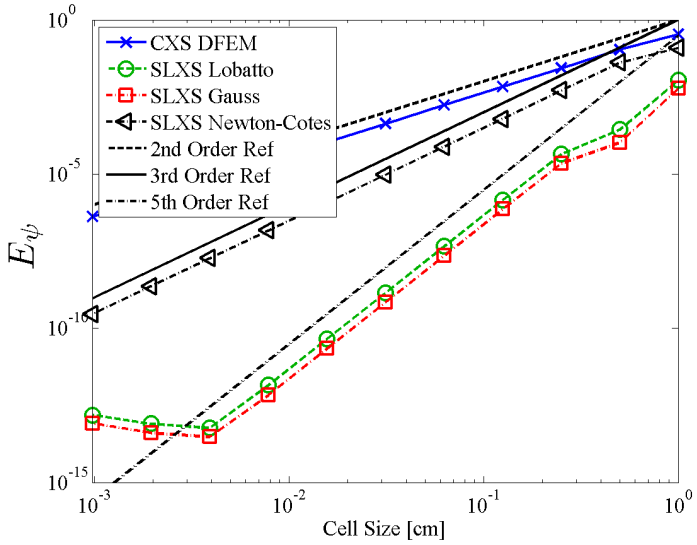

Figure 3.8: Convergence of E_ψ for a pure absorber with exponentially varying cross section discretized with quartic DFEM.

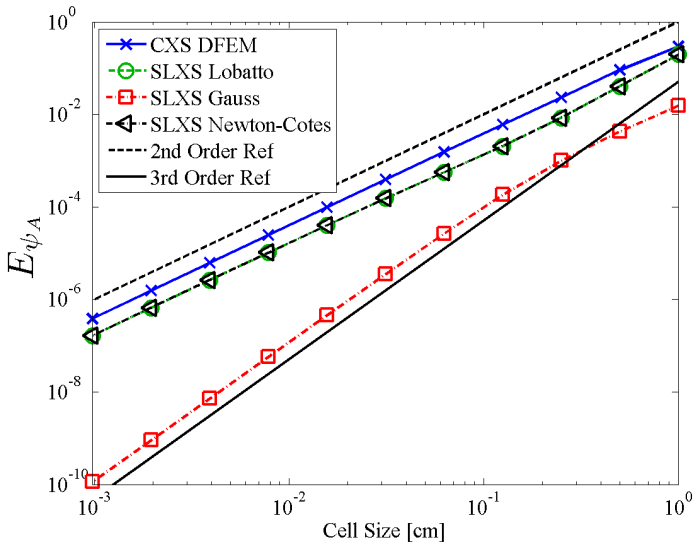

Figure 3.9: Convergence of E_{ψ_A} for a pure absorber with exponentially varying cross section discretized with linear DFEM.

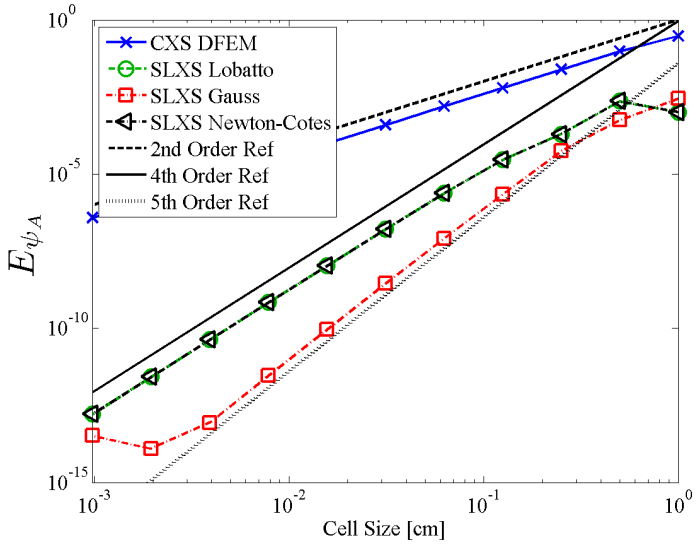

Figure 3.10: Convergence of E_{ψ_A} for a pure absorber with exponentially varying cross section discretized with quadratic DFEM.

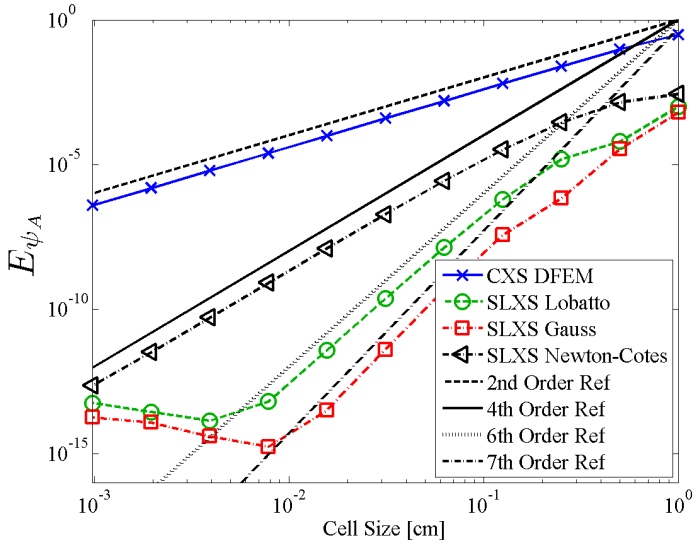

Figure 3.11: Convergence of E_{ψ_A} for a pure absorber with exponentially varying cross section discretized with cubic DFEM.

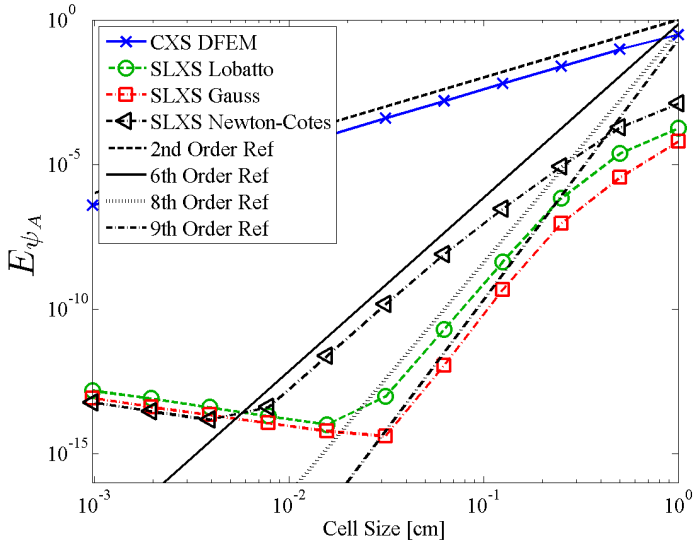

Figure 3.12: Convergence of E_{ψ_A} for a pure absorber with exponentially varying cross section discretized with quartic DFEM.

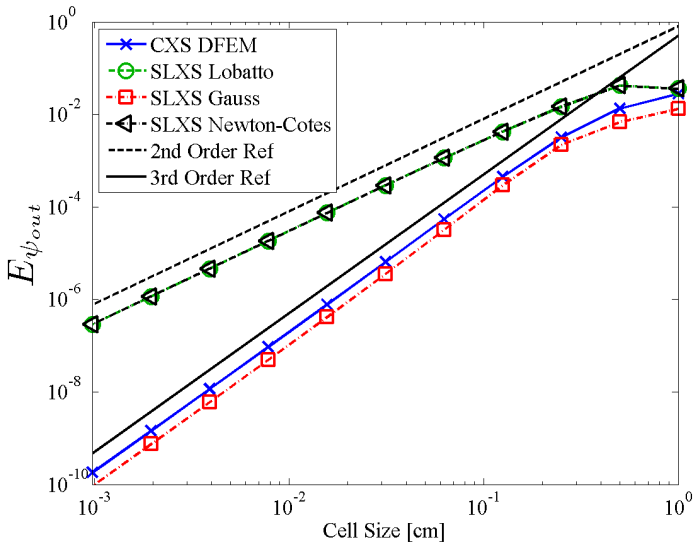

Figure 3.13: Convergence of $E_{\psi_{out}}$ for a pure absorber with exponentially varying cross section discretized with linear DFEM.

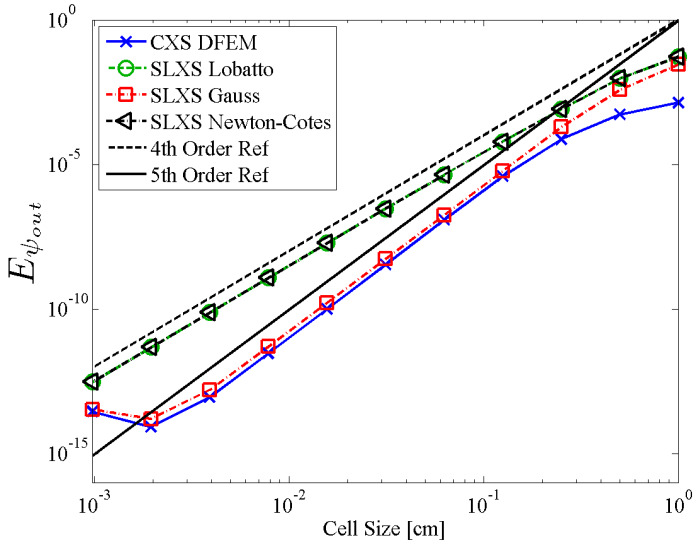

Figure 3.14: Convergence of $E_{\psi_{out}}$ for a pure absorber with exponentially varying cross section discretized with quadratic DFEM.

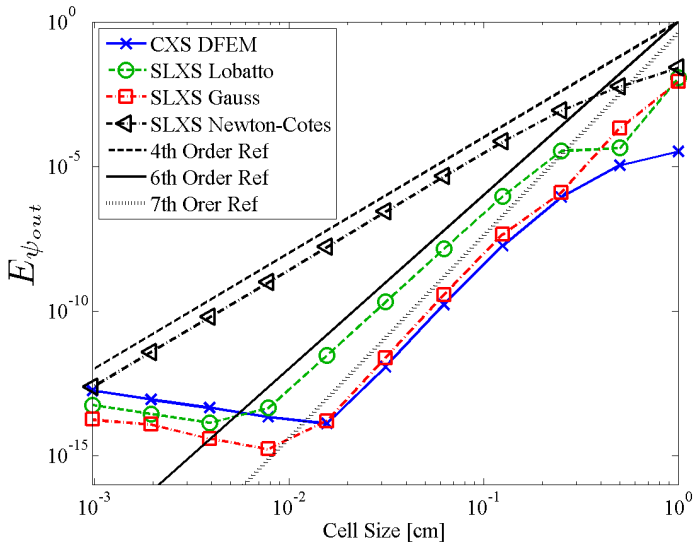

Figure 3.15: Convergence of $E_{\psi_{out}}$ for a pure absorber with exponentially varying cross section discretized with cubic DFEM.

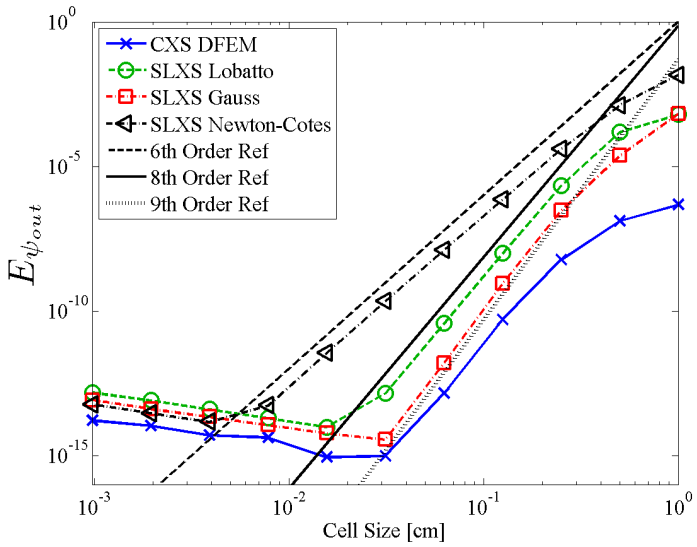

Figure 3.16: Convergence of $E_{\psi_{out}}$ for a pure absorber with exponentially varying cross section discretized with quartic DFEM.

The plateauing of numerical errors for various high-order methods using very small cell sizes in Figs. 3.5-3.16 is a consequence of having reached machine precision. The lines in Figs. 3.5-3.16 that extend to values smaller than machine precision are reference lines.

Figures 3.5-3.16 show that, for a linear angular flux trial space, CXS DFEM achieves the same orders of spatial convergence as observed with Exact DFEM in [11]. However, as the degree of the DFEM trial space is increased, the CXS DFEM scheme does not show an increase in the order of the spatial convergence rate of E_ψ and E_{ψ_A}; the convergence rate of CXS DFEM is limited to at most second order for both E_ψ and E_{ψ_A}, regardless of the trial space polynomial degree. The increase in order of convergence of CXS DFEM for $E_{\psi_{out}}$ as trial space is increased is a result of angular flux outflow in the CXS DFEM discretization being only a function of the cell optical thickness, which is preserved exactly by our definition of $\hat{\Sigma}_t$; see Eq. (3.7).

Of the self-lumping schemes, SLXS Newton-Cotes is the least accurate. SLXS Newton-Cotes convergence of E_ψ is limited to at most second order for odd degree polynomial trial spaces and third order for even degree trial spaces. Convergence of E_{ψ_A} and $E_{\psi_{out}}$ for the SLXS Newton-Cotes scheme generally increases with an increase in the DFEM polynomial trial space degree, but is only proportional to P. Both SLXS Lobatto and SLXS Gauss converge E_ψ, E_{ψ_A}, and $E_{\psi_{out}}$ similarly to the study carried out in [11] with a spatially constant cross section.

The spatial convergence of E_ψ for SLXS Lobatto and SLXS Gauss is order $P+1$. Though SLXS Lobatto and SLXS Gauss converge with the same order of spatial convergence for E_ψ, SLXS Gauss is more accurate than SLXS Lobatto by a constant. SLXS Gauss converges E_{ψ_A} and $E_{\psi_{out}} \propto 2P+1$, whereas SLXS Lobatto converges both $\propto 2P$. SLXS Gauss and SLXS Lobatto converge angular flux error quantities

for the case of a spatially varying cross section with the same rates of convergence as their constant cross-section analogs did in [11]. This suggests that exactly integrating the interaction term in the DFEM moment equations is not essential for developing arbitrarily high-order accuracy DFEM schemes for radiation transport.

Convergence of E_{IR} as function of numerical scheme for linear - quartic trial spaces is given in Figs. 3.17-3.20. We observe the detrimental effect of approxi-

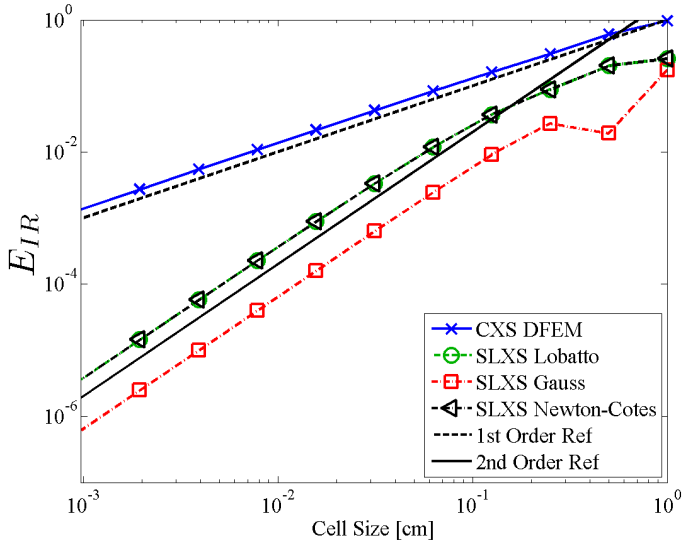

Figure 3.17: Convergence of E_{IR} for a pure absorber with exponentially varying cross section discretized with linear DFEM.

mating a spatially varying cross section with a constant in each spatial cell when we examine the L_2 convergence results for the interaction rate, E_{IR}, for the CXS DFEM scheme. Regardless of angular flux trial space polynomial degree, CXS DFEM converges E_{IR} to only first order in space. However, the self-lumping schemes exhibit the same trends in converging E_{IR} (in the L^2-norm sense) as exhibited in converging E_{ψ}:

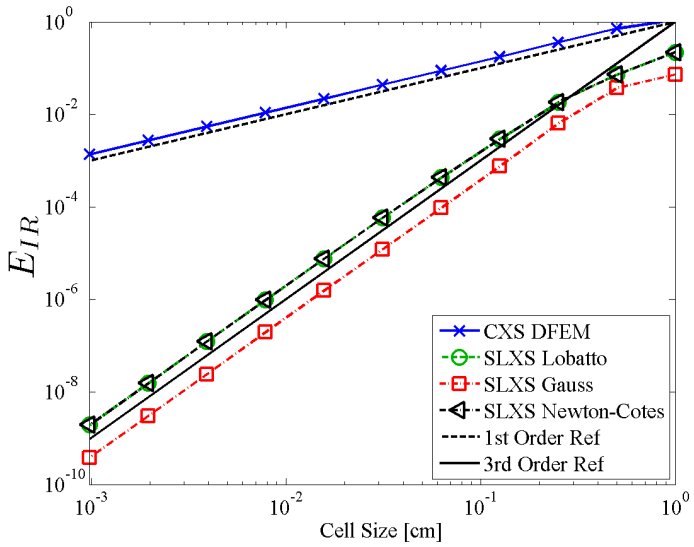

Figure 3.18: Convergence of E_{IR} for a pure absorber with exponentially varying cross section discretized with quadratic DFEM.

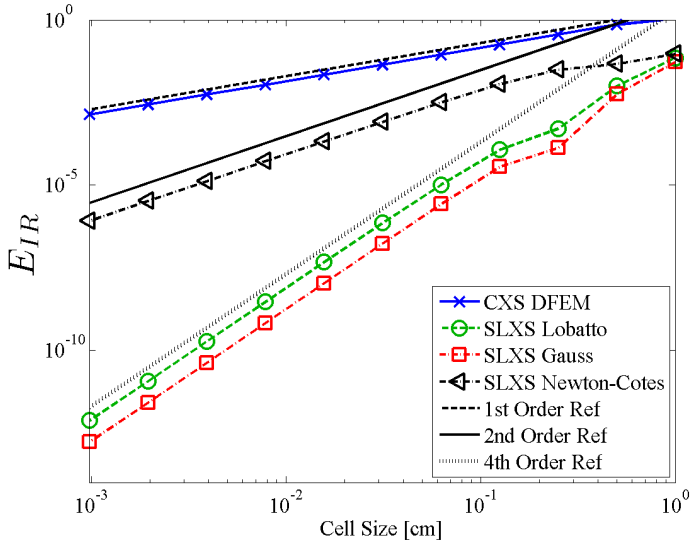

Figure 3.19: Convergence of E_{IR} for a pure absorber with exponentially varying cross section discretized with cubic DFEM.

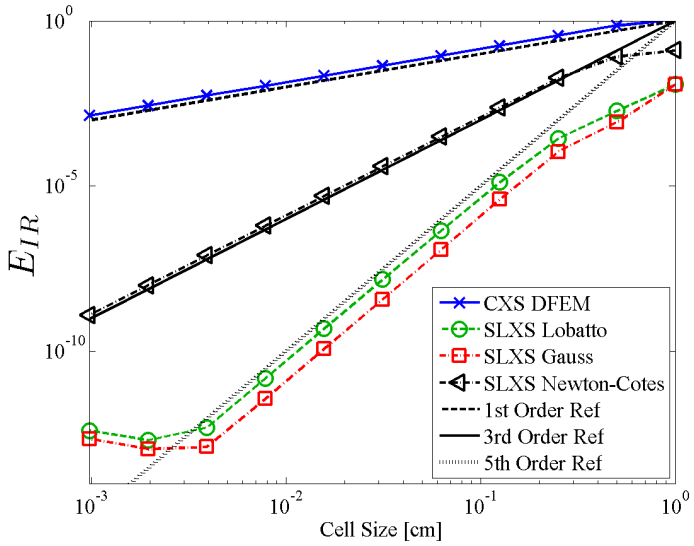

Figure 3.20: Convergence of E_{IR} for a pure absorber with exponentially varying cross section discretized with quartic DFEM.

- SLXS Lobatto and SLXS Gauss converge E_{IR} with order $P+1$,
- SLXS Gauss is more accurate than SLXS Lobatto by a constant, and
- SLXS Newton-Cotes converges E_{IR} second order in space for odd degree trial spaces and third order in space for even degree trial spaces.

Convergence data for E_{IR_A} as function of cell size for linear - quartic trial spaces is given in Figs. 3.21-3.24. The convergence results for the error in cell average interaction rate, E_{IR_A}, shown in Figs. 3.21-3.24, do not behave as intuitively as the convergence rates for E_{IR} observed in Figs. 3.17-3.20. Given the poor performance of CXS DFEM in converging E_{IR}, one would expect that CXS DFEM would converge E_{IR_A} poorly as well. However, this is not the case and CXS DFEM converges E_{IR_A} with the same order of convergence as the best performing self-lumping scheme considered.

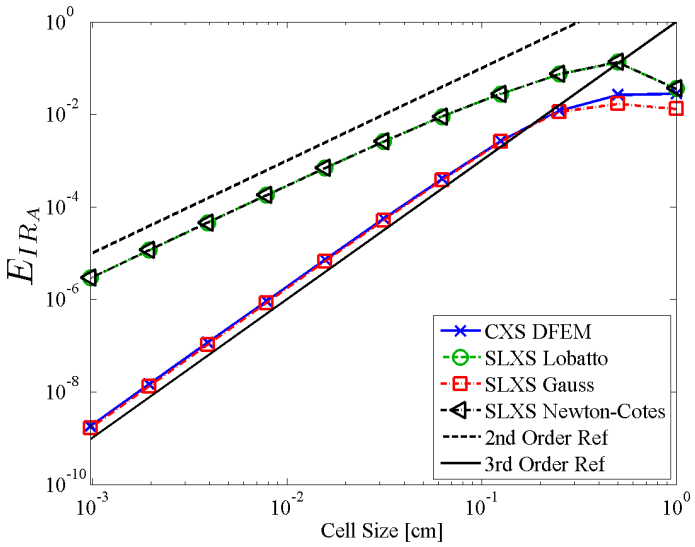

Figure 3.21: Convergence of E_{IR_A} for a pure absorber with exponentially varying cross section discretized with linear DFEM.

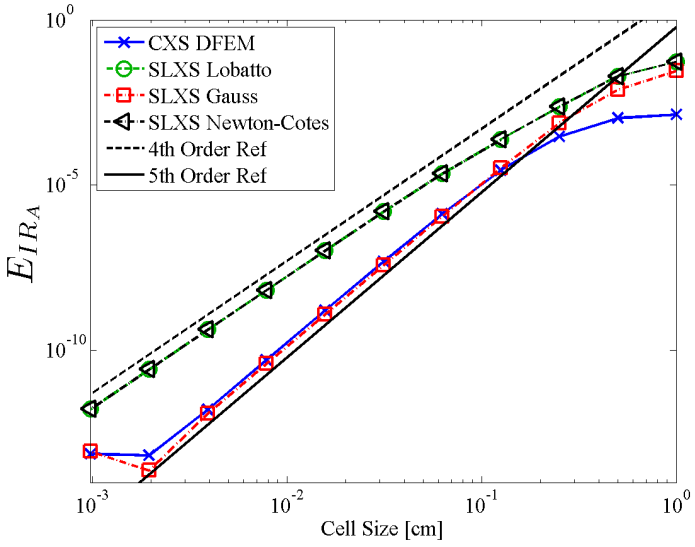

Figure 3.22: Convergence of E_{IR_A} for a pure absorber with exponentially varying cross section discretized with quadratic DFEM.

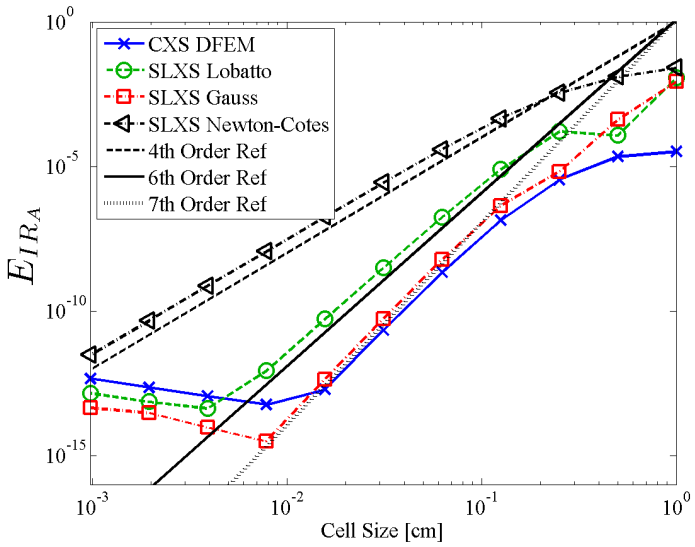

Figure 3.23: Convergence of E_{IR_A} for a pure absorber with exponentially varying cross section discretized with cubic DFEM.

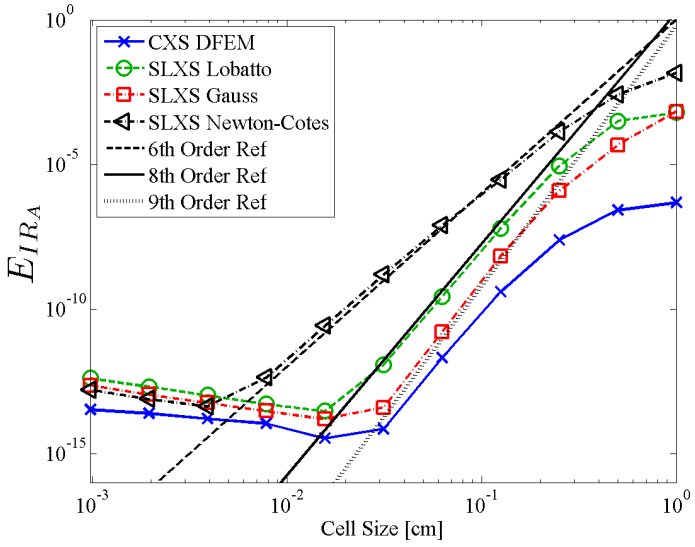

Figure 3.24: Convergence of E_{IR_A} for a pure absorber with exponentially varying cross section discretized with quartic DFEM.

CXS DFEM converges E_{IR_A} with a high-order of accuracy because of the locally conservative properties of DFEM approximations, that is:

$$\text{Particles Into Cell} - \text{Particles Out of Cell} = \text{Total Interactions in Cell}\,. \tag{3.26}$$

As shown in Figs. 3.13-3.16, CXS DFEM converges the quantities on the left hand side of Eq. (3.26) with the same order of accuracy as any self-lumping scheme considered; CXS DFEM is at least as accurate in calculating the particles into a cell (outflow from the previous cell) and out of the cell (outflow from the current cell) as any other scheme considered. Since Eq. (3.26) holds regardless of the numerical scheme considered, it follows that CXS DFEM converges E_{IR_A}, the term in the right hand side of Eq. (3.26) summed over all cells, with the maximum order of convergence displayed by any of the DFEM schemes we consider here. Figures 3.21-3.24 validate this conclusion. CXS DFEM and SLXS Gauss exhibit the highest order of spatial convergence, converging E_{IR_A} with order $\propto 2P+1$. SLXS Newton-Cotes and SLXS Lobatto converge E_{IR_A} with the same orders of convergence each method exhibits in converging E_{ψ_A}. For this problem SLXS Lobatto converges $E_{IR_A} \propto 2P$ and SLXS Newton-Cotes converged $E_{IR_A} \propto P$.

3.3.3 Consequences of Assuming a Cell-Wise Constant Cross Section

To more fully understand the poor convergence of point-wise error in angular flux and interaction rate, E_ψ and E_{IR}, associated with CXS DFEM we now examine more closely the CXS DFEM spatial approximations to $\psi(x,\mu_d)$ and $IR(x)$. We again consider a pure absorber with total absorption cross section that varies exponentially in space with $c_1 = 0.1$, and $c_2 = 2\ln(10)$. A beam of radiation is incident on the left face in the direction of $\mu_d = 1$, vacuum boundary conditions are applied on the right face of the slab, and $x \in [0,1]$.

In Fig. 3.25, we plot the exact $\psi(x)$ and in Fig. 3.26, we plot $IR(x)$. Additionally we plot the respective CXS DFEM numerical approximations, $\widetilde{\psi}(x)$ and $\widetilde{IR}(x)$, using $N_{cells} = 5$, and $\hat{\Sigma}_{t,i}$ as defined in Eq. (3.7). Also plotted in Fig. 3.25 and Fig. 3.26, we plot the analytic angular flux and reaction rate, respectively, one would obtain if the cell average cross section, $\hat{\Sigma}_{t,i}$, had been used instead of the true $\Sigma_t(x)$. We refer to these analytic solutions as $\psi_C(x)$ and $IR_C(x)$.

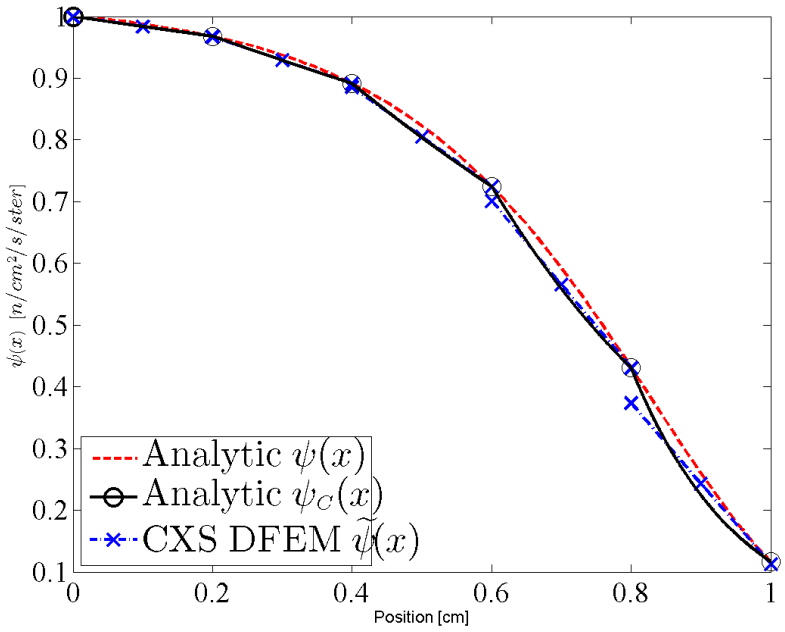

Figure 3.25: Plots of the analytic $\psi(x)$, CXS DFEM $\widetilde{\psi}(x)$, and $\psi_C(x)$ for the pure absorber with exponential cross section.

Since CXS DFEM is a discontinuous scheme, some discontinuity is expected in the plot of $\widetilde{\psi}$ in Fig. 3.25 and of $\widetilde{IR}$ in Fig. 3.26. However, the discontinuities present in Fig. 3.26 are highly disconcerting. The analytic $IR(x)$ is smooth and does not vary rapidly within individual mesh cells, yet there are significant, non-monotonic

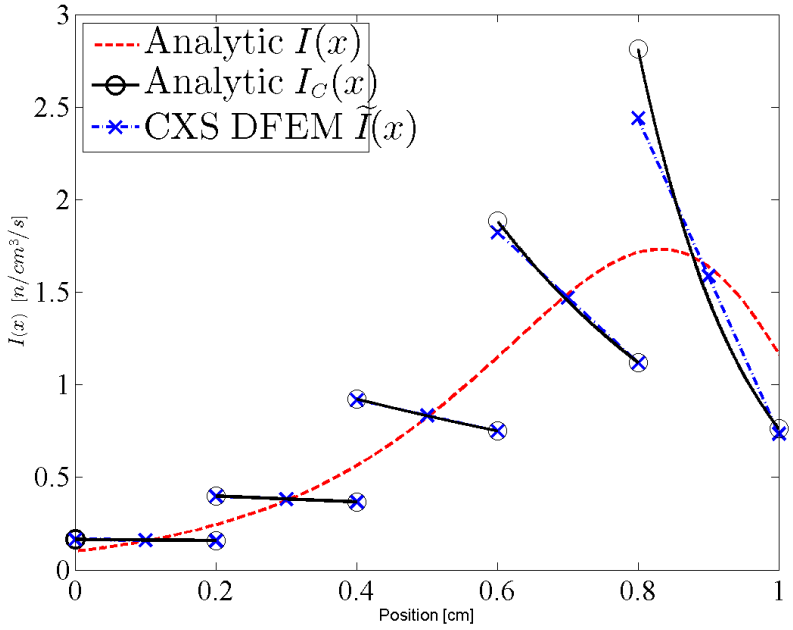

Figure 3.26: Plots of the analytic $IR(x)$, CXS DFEM $\widetilde{IR}(x)$, and $IR_C(x)$ for the pure absorber with exponential cross section.

discontinuities in the CXS DFEM interaction rate solution to the pure absorber problem with exponentially varying cross section. The noticeably poor behavior of $\widetilde{IR}(x)$ in Fig. 3.26 is inherent to the assumption of a cell-wise constant cross section. This inherent error of approximating a continuously varying cross section with cell-wise constants is clearly visible when comparing the plots of $IR_C(x)$ to $IR(x)$ in Fig. 3.26. The only difference and source of error in $IR_C(x)$ is the assumption of a cell-wise constant cross section. Figure 3.25 and Fig. 3.26 does not suggest that linear DFEM is unsuitable for use in problems with spatially varying cross sections. Rather, comparing the CXS DFEM $\widetilde{\psi}(x)$to $\psi_C(x)$ in Fig. 3.25 and $\widetilde{IR}(x)$ to $IR_C(x)$ in Fig. 3.26, we see that CXS DFEM is very accurate when compared to the analytic solution of the problem CXS DFEM is solving, a pure absorber with cell-wise constant cross section. Given the poor accuracy of CXS DFEM in approximating the true $\psi(x)$

and $IR(x)$, we wish to see how a scheme that does not assume a cell-wise constant cross section behaves. Consider $\widetilde{\psi}(x)$ and $\widetilde{IR}(x)$ obtained with SLXS Lobatto using a linear DFEM trial space and five spatial cells, shown in Fig. 3.27 and Fig. 3.28, for the same problem. In Fig. 3.27, the differences between the angular flux solutions obtained using (1) a cell-wise constant cross section (CXS DFEM) and (2) evaluating cross section values at quadrature points (SLXS Lobatto) are small.

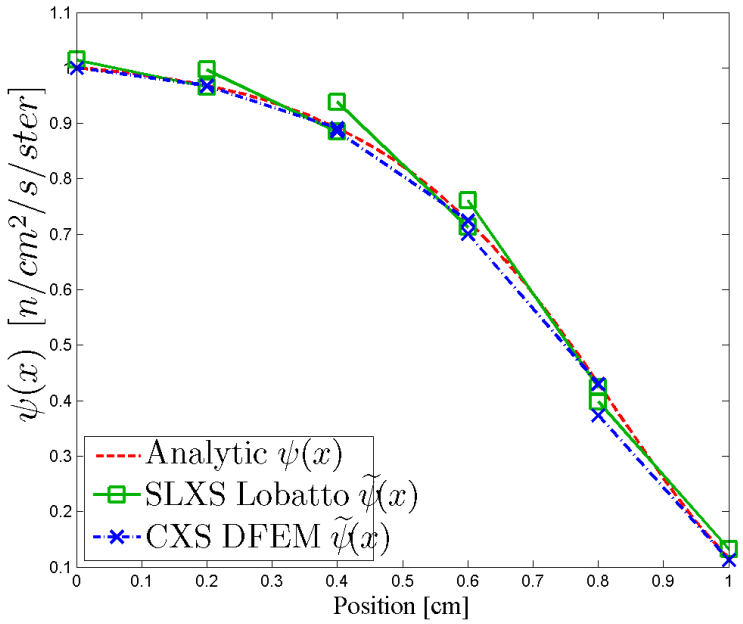

Figure 3.27: Plot of the linear trial space SLXS Lobatto and CXS DFEM approximations of $\widetilde{\psi}(x)$ for the pure absorber problem with exponentially varying cross section.

This is not the case when comparing the different approximations to the interaction rate, $\widetilde{IR}(x)$, in Fig. 3.28. Though there are discontinuities in the SLXS Lobatto $\widetilde{IR}(x)$, the discontinuities are smaller and the SLXS Lobatto $\widetilde{IR}(x)$ is monotonic unlike the CXS DFEM $\widetilde{IR}(x)$. The SLXS Lobatto $\widetilde{IR}(x)$ is clearly more accurate

than the CXS DFEM $\widetilde{IR}(x)$.

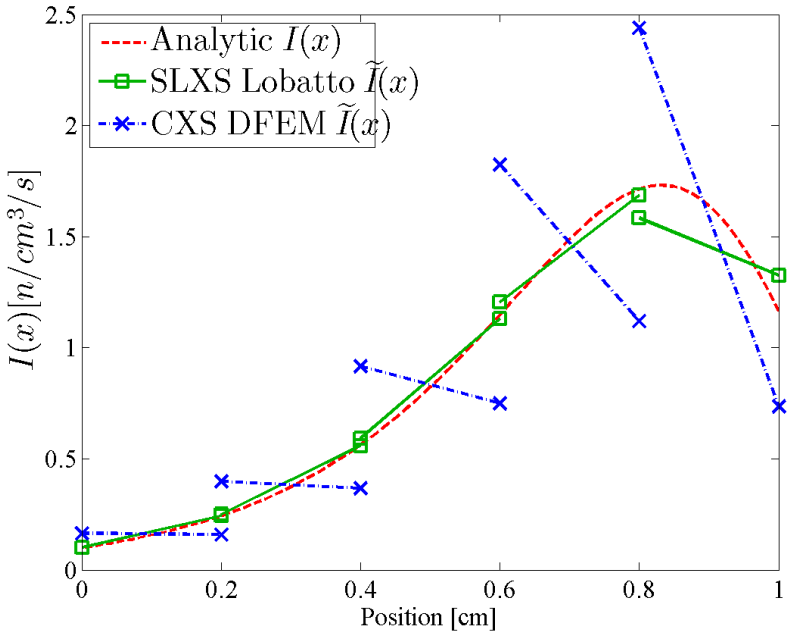

Figure 3.28: Plot of the linear trial space SLXS Lobatto and CXS DFEM approximations of $\widetilde{IR}(x)$ for the pure absorber problem with exponentially varying cross section.

In this problem, there are two possible sources of error that could cause a DFEM to be inaccurate: inexact matrix evaluation and not incorporating cross-section spatial variation into the scheme. By definition, CXS DFEM exactly integrates the mass matrix, and we showed in Chapter 2 that schemes that exactly integrate the mass matrix are more accurate than schemes that only approximately integrate the mass matrix, like SLXS Lobatto. Thus, the poor accuracy of CXS DFEM relative to SLXS Lobatto is entirely caused by the approximation of a spatially varying cross section with a cell-wise constant value.

The "blading" in $\widetilde{IR}(x)$ has not previously been reported in neutron transport

or thermal radiative transport community literature. We are likely not the first to have generated these large, non-monotonic discontinuities. In fact, we believe that blading has frequently been present in DFEM both neutron transport and thermal radiative transfer simulations but has likely gone unnoticed due to the prevalence of linear DFEM and simplified data visualization using cell midpoint values.

To demonstrate that a minor choice in data presentation can obscure the existence of blading, consider Fig. 3.29 and Fig. 3.30 that linearly interpolate between $\widetilde{\psi}_{A,i}$ and $\widetilde{IR}_{A,i}$ plotted at cell centers. Though Fig. 3.29 is visually indistinguishable from Fig. 3.25, the blading of $\widetilde{IR}(x)$ present in Fig. 3.26 is not at all visually present in Fig. 3.30. Interaction rate terms are present in other radiation transport physics. In particular, we think of the radiative transfer analog to the neutronics interaction rate, absorption rate density. We hypothesize here, and will show in Chapter 6 that assuming cell-wise constants opacities for problems with temperature dependent opacities introduces blading in thermal radiative transfer temperature solutions, via the material energy equation's absorption rate density term.

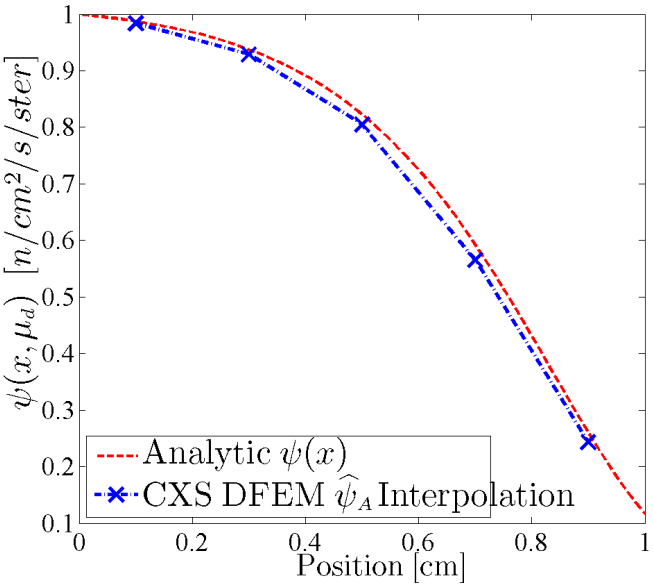

Figure 3.29: Plot of the CXS DFEM cell average angular flux at cell centers with linear interpolation for a pure absorber with exponentially varying spatial cross section.

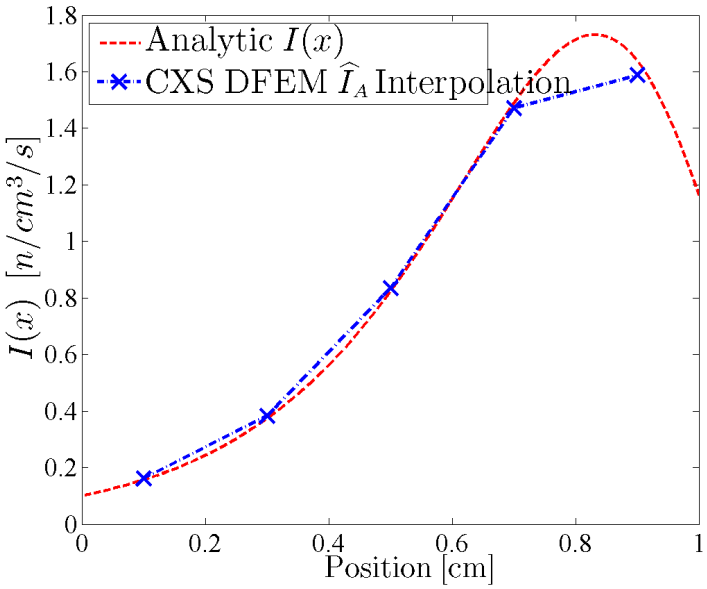

Figure 3.30: Plot of the CXS DFEM cell average interaction rate at cell centers with linear interpolation for a pure absorber with exponentially varying spatial cross section.

3.3.4 Flux-Weighting versus Volume-Averaged Cross Sections

In our results thus far, we have only considered volume-averaged cell-wise cross sections(the CXS DFEM scheme). However, in reactor physics problems, a flux-weighted cross section is often used to generate spatially averaged cross sections [33]. We now introduce the flux-weighted cell-wise constant cross section scheme (FW CXS), which differs from the CXS DFEM scheme only by how $\hat{\Sigma}$ is defined in each cell. For the FW CXS scheme, we define $\hat{\Sigma}_i$ as

$$\hat{\Sigma}_i = \frac{\int_{x_{i-1/2}}^{x_{i+1/2}} \Sigma(x)\psi(x,\mu_d)\ dx}{\int_{x_{i-1/2}}^{x_{i+1/2}} \psi(x,\mu_d)\ dx}\,. \tag{3.27}$$

In practice, flux-weighting is often done using the scalar flux in order not to have angle-dependent total cross section. However for the beam problem we consider here, $\psi(x,\mu_d)$ is proportional to the scalar flux.

We first compare the accuracy of FW CXS versus volume-averaged CXS DFEM for a cubic DFEM trial space, as shown in Figs. 3.31-3.34. In Figs. 3.31-3.34, we omit a plot of $E_{\psi_{out}}$ as we have already demonstrated that the accuracy of any method in calculating $E_{\psi_{out}}$ determines the method's accuracy in calculating E_{IR_A}. That is, if E_{IR_A} converges at a given rate, $E_{\psi_{out}}$ converges at the same rate.

Figures 3.31-3.33 show that FW CXS scheme is more accurate than CXS DFEM when comparing E_ψ, E_{ψ_A} and, at low resolutions, E_{IR}. However, though designed to preserve cell average interaction rates, FW CXS scheme is not only less accurate than CXS DFEM in calculating cell average interaction rates, it converges E_{IR_A} at most second order in space, whereas a volume-averaged cross section converges E_{IR_A} $\propto 2P+1$ for the pure absorber problem.

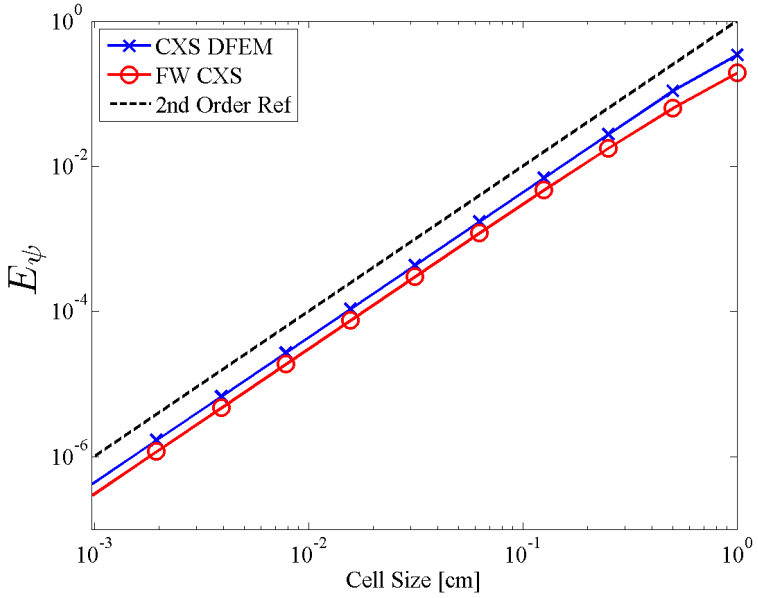

Figure 3.31: Accuracy comparison of flux weighted constant cross section scheme (FW CXS) to volume averaged cross section scheme (CXS DFEM) with a cubic angular flux trial space.

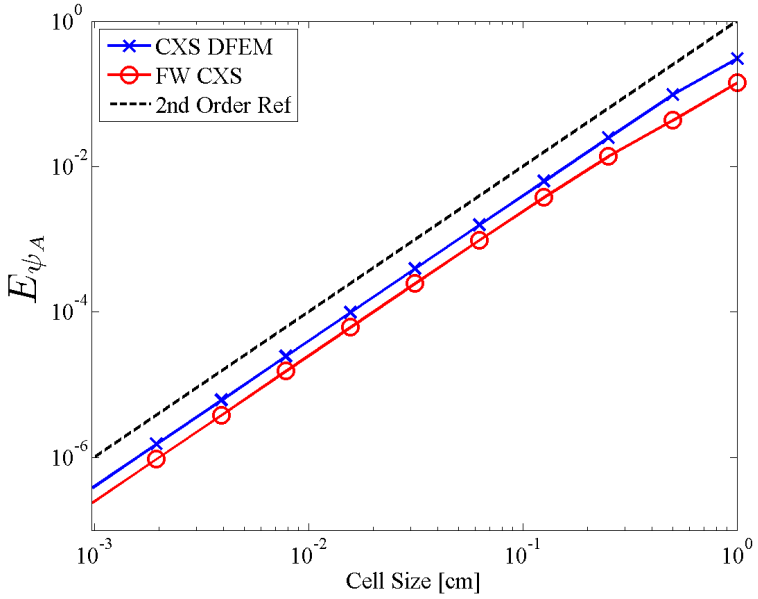

Figure 3.32: Accuracy comparison of flux weighted constant cross section scheme (FW CXS) to volume averaged cross section scheme (CXS DFEM) with a cubic angular flux trial space.

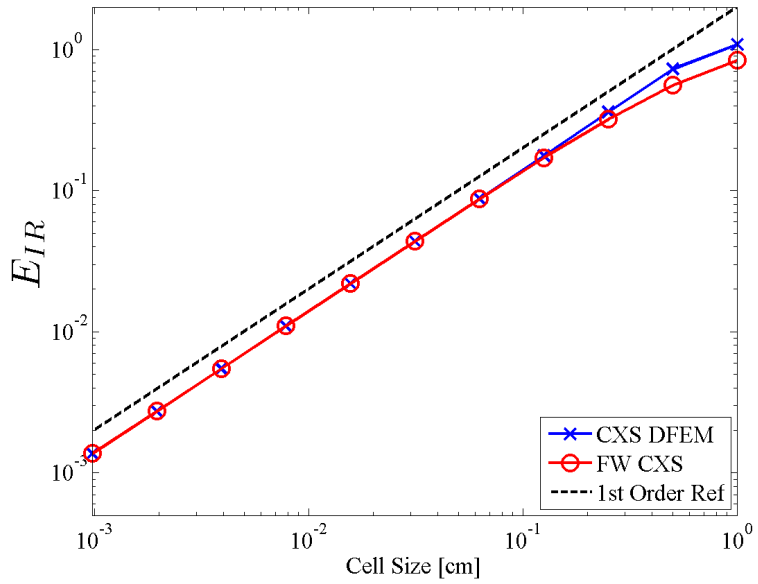

Figure 3.33: Accuracy comparison of flux weighted constant cross section scheme (FW CXS) to volume averaged cross section scheme (CXS DFEM) with a cubic angular flux trial space.

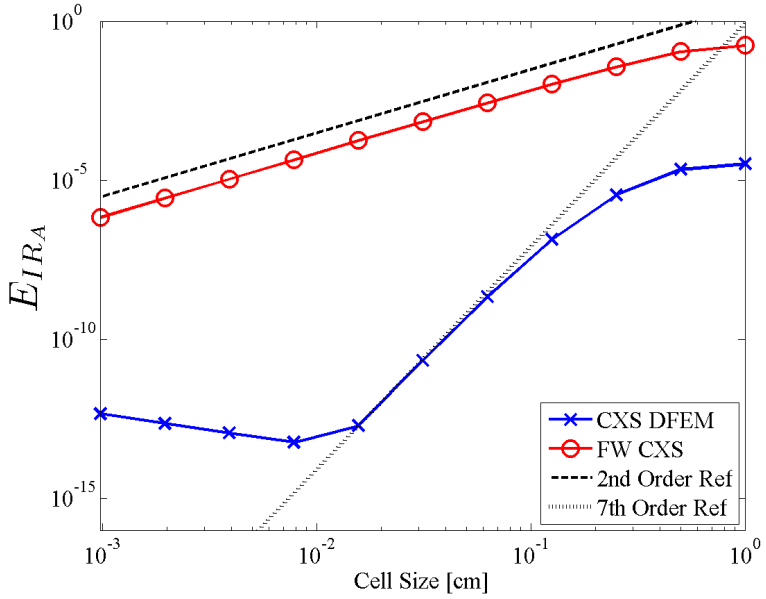

Figure 3.34: Accuracy comparison of flux weighted constant cross section scheme (FW CXS) to volume averaged cross section scheme (CXS DFEM) with a cubic angular flux trial space.

Finally, we consider the FW CXS scheme's $\widetilde{\psi}(x)$ in Fig. 3.35 and $\widetilde{IR}(x)$ in Fig. 3.36. In Fig. 3.35 and Fig. 3.36 the FW CXS and CXS DFEM schemes calculate slightly different solution representations. However, the FW CXS scheme exhibits the same interaction rate blading phenomena as the CXS DFEM scheme, reiterating that blading is a result of approximating a spatially varying cross section as a cell-wise constant. The choice of cell-wise cross section does not eliminate blading.

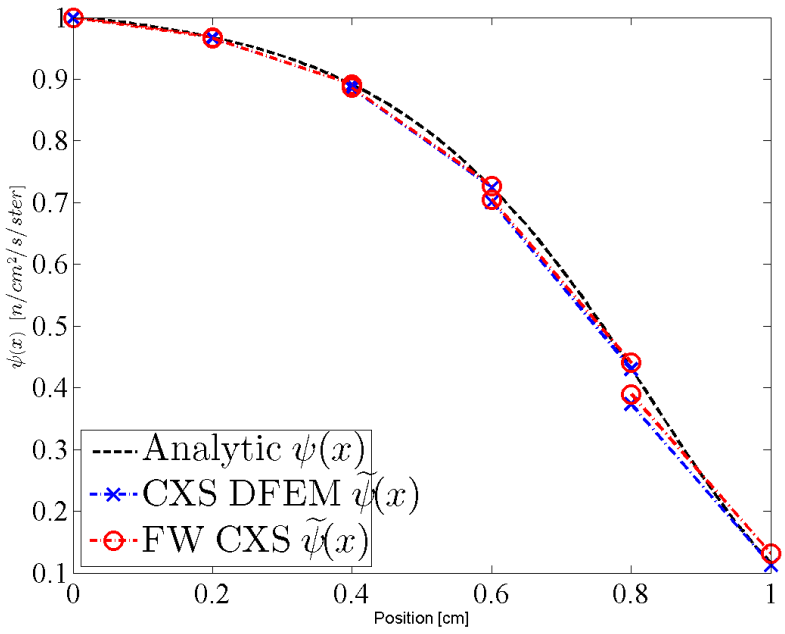

Figure 3.35: Plot of the linear trial space FW CXS and CXS DFEM approximations to $\widetilde{\psi}(x)$ for the pure absorber problem with exponentially varying spatial cross section.

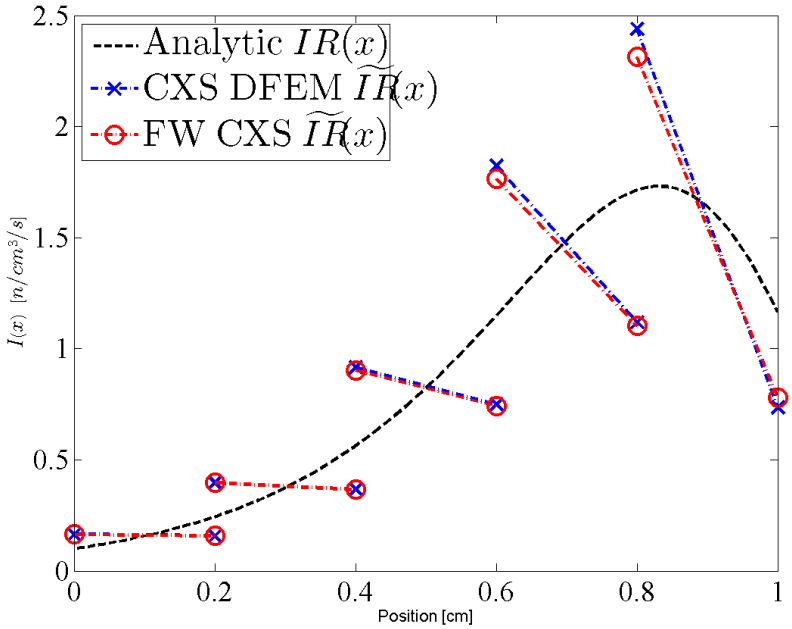

Figure 3.36: Plot of the linear trial space FW CXS and CXS DFEM approximations to $\widetilde{IR}(x)$ for the pure absorber problem with exponentially varying spatial cross section.

Table 3.2: Shorthand notation of different cell spacing schemes.

Spacing Label	Spacing Type
EQUAL	Equally-spaced cells
MFP	Constant optical thickness cells
LOG	Logarithmically spaced cells

3.3.5 Effects of Mesh Spacing

In practice, computational domains are not necessarily discretized with uniform grids; rather cells are concentrated in regions where the solution is known or assumed to vary rapidly. For the pure absorber problem, we compare two alternative methods of mesh spacing (logarithmic grids and equal optical thickness grids, see Table 3.2), to the results obtained with equally-spaced mesh cells.

We derive how to generate these meshes in Section 3.3.5.1 and give results in Section 3.3.5.2

3.3.5.1 Generating Improved Spatial Meshes

Two alternative meshing strategies are compared to equally-spaced meshed. In the following, we will use a shorthand notation, given in Table 3.2. With the MFP meshing strategy, we find each cell width by determining the width of each cell from $i = 1$ (leftmost cell) to $i = N_{cell}$ as outlined by Eq. (3.28): First, we determine the average cell optical thickness:

$$\bar{h} = \frac{\int_{x_{1/2}}^{x_{N_{cell}+1/2}} \Sigma_t(x)\ dx}{N_{cell}}\,. \tag{3.28a}$$

Then, we solve the following equation for $x_{i+1/2}$:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \Sigma_t(x)\ dx - \bar{h} = 0\,, \tag{3.28b}$$

yielding

$$x_{i+1/2} = \frac{1}{c_2} \log \left[\frac{c_2(\bar{h} + \Sigma_t(x_{i-1/2})}{c_1} \right] . \tag{3.28c}$$

There are several ways to specify LOG spacing, but we elected to set a ratio, 0.6, between adjacent cell sizes with the caveat that we would set a minimum cell size, Δx_{min}. In our convergence testing, at the first refinement when $\Delta x_{N_cell} < \Delta x_{min}$, the grid is "fixed" and all further refinements uniformly refine the "fixed" grid. $\Delta x_{N_{cell}}$ is the cell width for the right most cell where, for $R < 1$,

$$\Delta x_i = \Delta x_1 R^{i-1}, \ i \in [1, N_{cell}] . \tag{3.29}$$

Δx_1 is determined by requiring that the geometric series of cell widths completely fill the space:

$$\Delta x_1 = \left(x_{N_{cell}+1/2} - x_{1/2} \right) \frac{1 - R}{1 - R^{N_{cell}}} \tag{3.30}$$

The grid is "fixed" by resetting the width of every cell whose width, if set to the value required for a purely logarithmically spaced grid with R would be below Δx_{min}, to Δx_{min}. After imposing this, cell widths are determined by requiring the cells that were not reset to fill the problem space logarithmically using R. If there is no minimum cell width, at high mesh refinements, most cells will be infinitesimally small and the large cells will never be refined, causing error to stagnate. Logarithmic spacing represents the "smart" meshing strategy most likely to be employed in engineering practice as it requires the least amount of solution information prior to problem execution. For all of our calculations, we set $R = 0.6$ and $\Delta x_{min} = 10^{-3}$ $[cm]$.

3.3.5.2 Mesh Spacing Results

We begin our discussion of alternate mesh spacing by first noting that the choice of mesh spacing method does not alter asymptotic convergence rates, as shown in

Figs. 3.37-3.40. The convergence of the SLXS Lobatto scheme using a quadratic trial space for E_ψ is shown in Fig. 3.37, for E_{ψ_A} in Fig. 3.38, for E_{IR} in Fig. 3.39, and for E_{IR_A} in Fig. 3.40. Regardless of the choice of mesh spacing, all error quantities of interest converge at the same asymptotic rate. Plots showing other trial space degrees and DFEM schemes are omitted for brevity. We also omit showing the convergence of $E_{\psi_{out}}$ as we have already demonstrated that the convergence rate of $E_{\psi_{out}}$ and $E_{IR_A}$are related and identical.

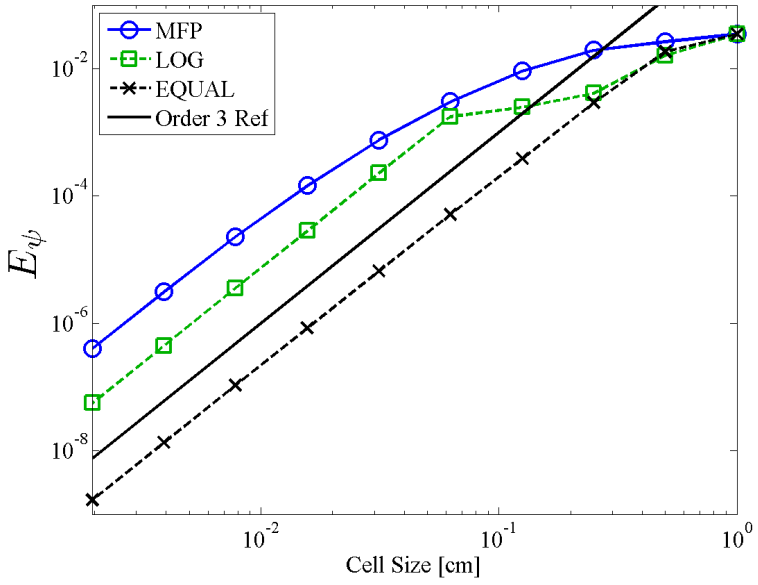

Figure 3.37: Asymptotic convergence of the SLXS Lobatto scheme using a quadratic trial space, for different mesh spacing methodologies.

At mesh refinements that are not in the asymptotic convergence regime, the intelligent meshing can result in a significant reduction in error. Consider the difference smart meshing has in reducing E_ψ, E_{ψ_A}, E_{IR}, and E_{IR_A} for the SLXS Lobatto scheme in Figs. 3.41-3.44, for the SLXS Gauss scheme in Figs. 3.45-3.48, and for CXS DFEM

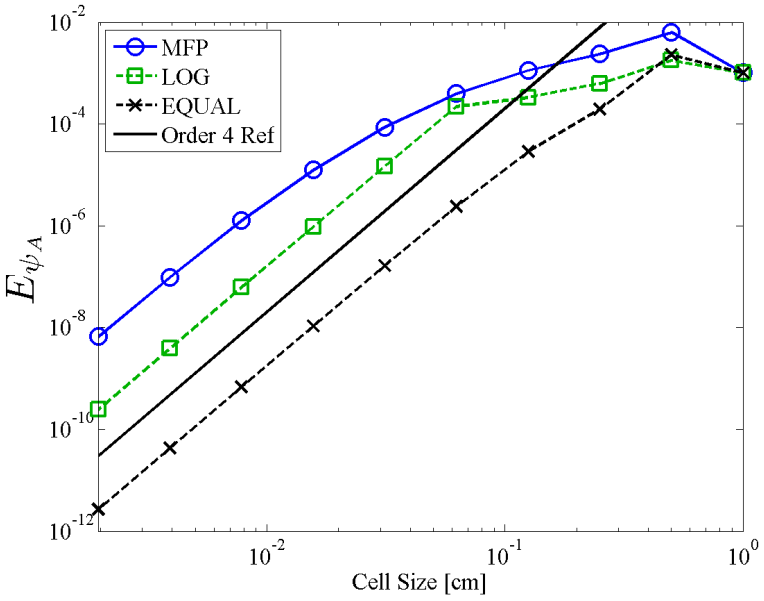

Figure 3.38: Asymptotic convergence of the SLXS Lobatto scheme using a quadratic trial space, for different mesh spacing methodologies.

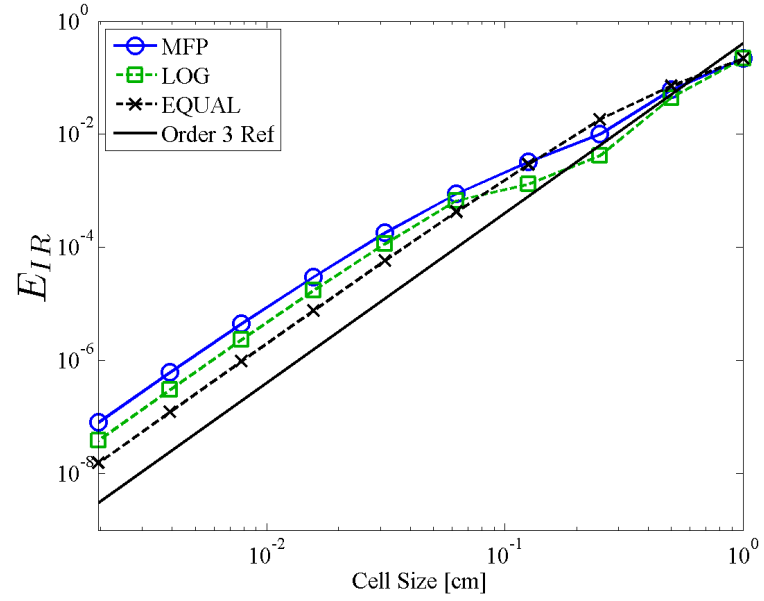

Figure 3.39: Asymptotic convergence of the SLXS Lobatto scheme using a quadratic trial space, for different mesh spacing methodologies.

in Figs. 3.49-3.52 when using a quadratic DFEM trial space at low cell resolutions.

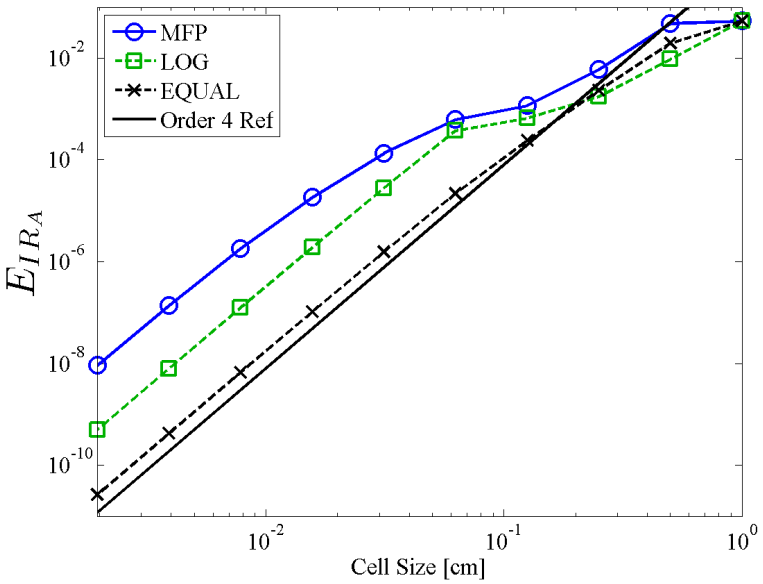

Figure 3.40: Asymptotic convergence of the SLXS Lobatto scheme using a quadratic trial space, for different mesh spacing methodologies.

First, we note that any particular mesh spacing methodology results in solutions that are more accurate for certain quantities, but not for all quantities. For example, with the self-lumping schemes, E_{ψ} is generally smaller when using an equally spaced mesh, but using the LOG mesh results in orders of magnitude improvement in E_{IR} and E_{IR_A} on coarse meshes. Figures 3.49-3.52, CXS DFEM again illustrates that LOG spacing is more accurate in calculating interaction rate quantities than an equally-spaced mesh and equally-spaced meshes are generally more accurate than other meshing strategies for calculating angular flux quantities. However, CXS DFEM shows a two order of magnitude reduction in calculating E_{IR_A} when using a mesh that has a uniform optical thickness in each cell. This is a direct result of CXS DFEM converging as $\hat{\sigma}\Delta x \to 0$.

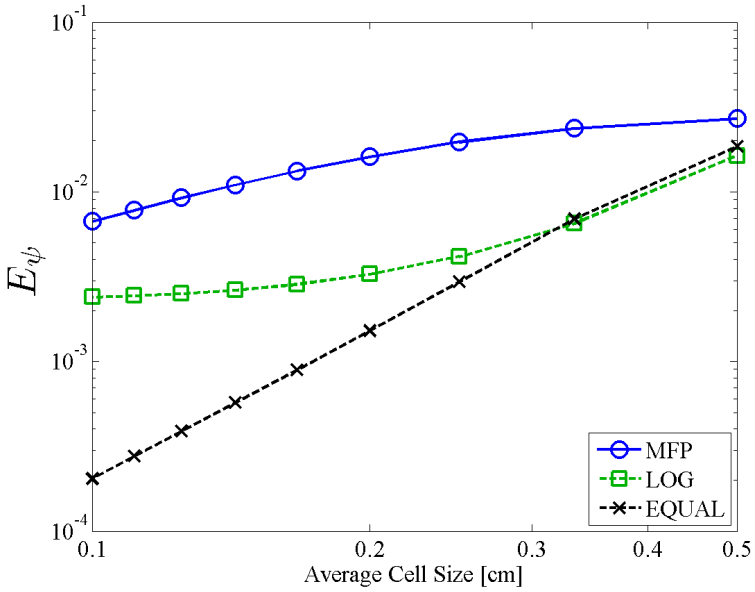

Figure 3.41: Errors associated with the SLXS Lobatto using a quadratic trial space scheme for different mesh spacing methodologies, at low resolutions.

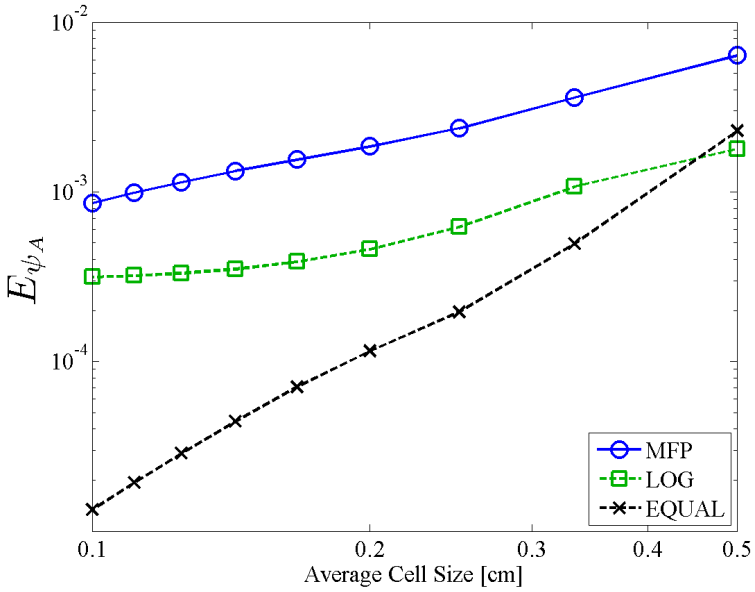

Figure 3.42: Errors associated with the SLXS Lobatto using a quadratic trial space scheme for different mesh spacing methodologies, at low resolutions.

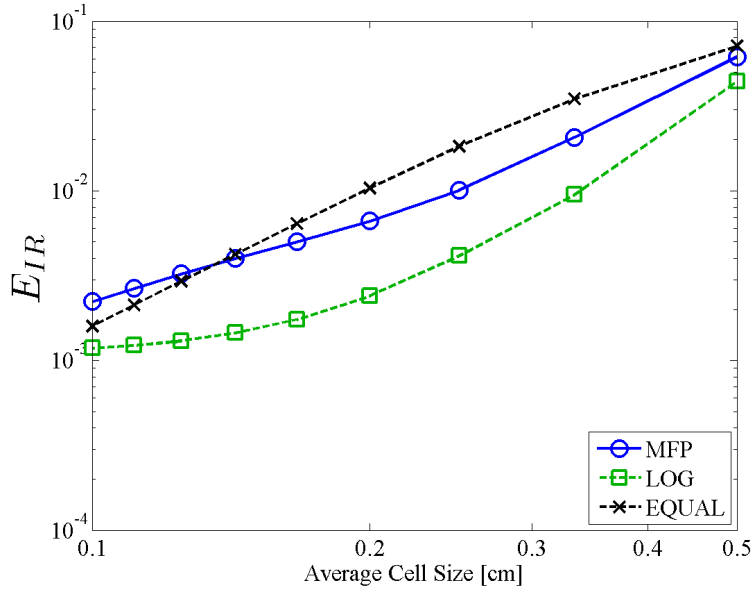

Figure 3.43: Errors associated with the SLXS Lobatto using a quadratic trial space scheme for different mesh spacing methodologies, at low resolutions.

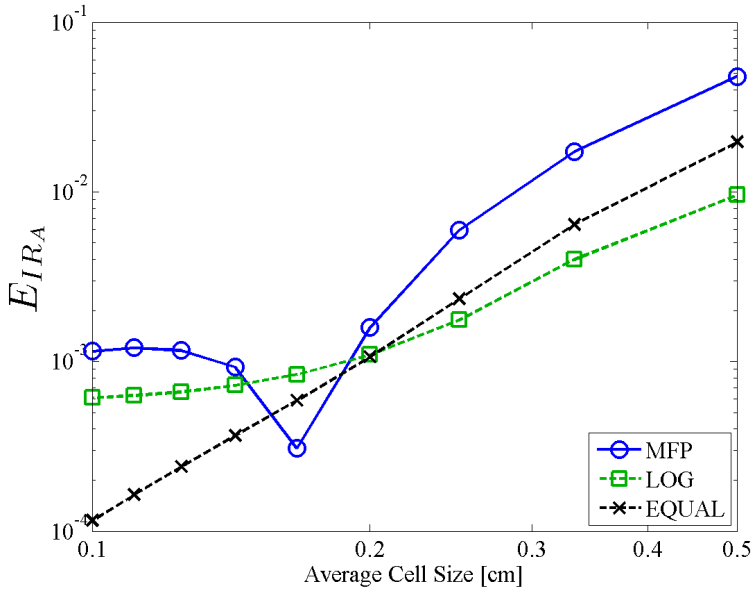

Figure 3.44: Errors associated with the SLXS Lobatto using a quadratic trial space scheme for different mesh spacing methodologies, at low resolutions.

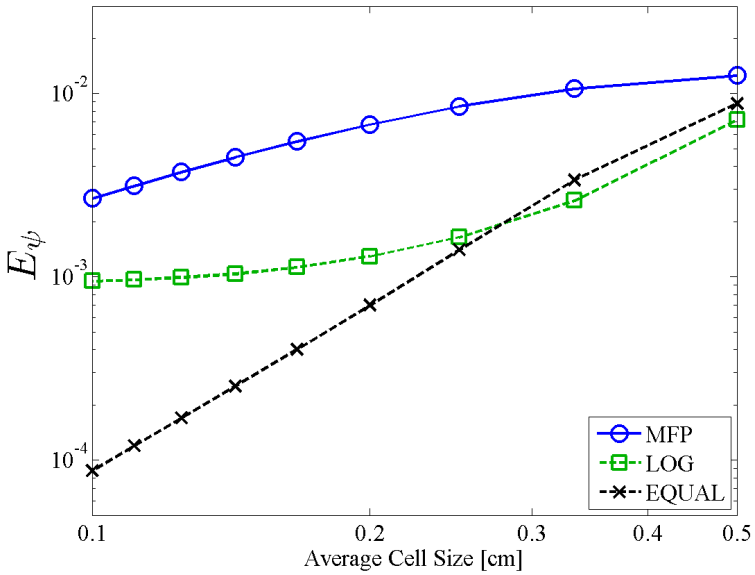

Figure 3.45: Errors associated with the SLXS Gauss using a quadratic trial space for different mesh spacing methodologies, at low resolutions.

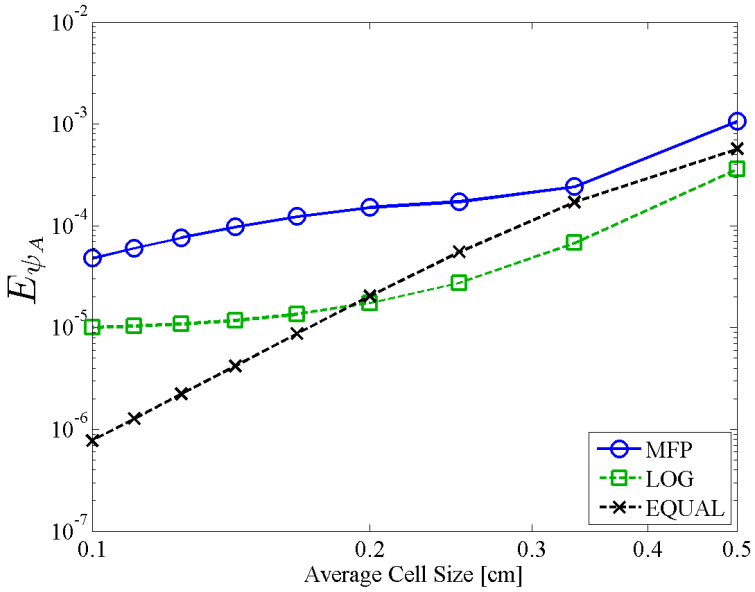

Figure 3.46: Errors associated with the SLXS Gauss using a quadratic trial space for different mesh spacing methodologies, at low resolutions.

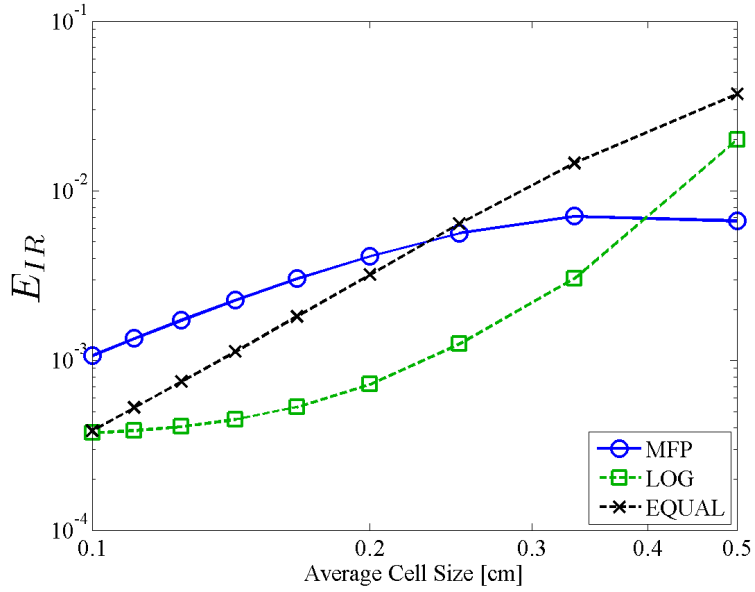

Figure 3.47: Errors associated with the SLXS Gauss using a quadratic trial space for different mesh spacing methodologies, at low resolutions.

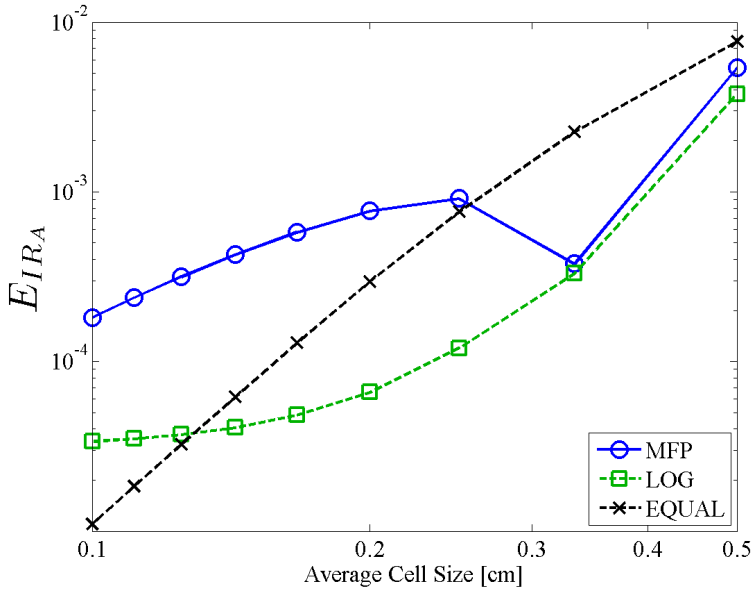

Figure 3.48: Errors associated with the SLXS Gauss using a quadratic trial space for different mesh spacing methodologies, at low resolutions.

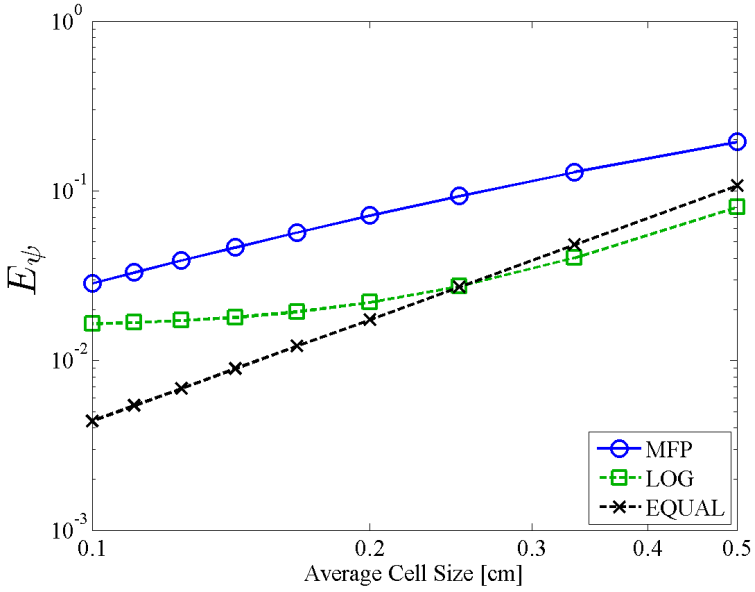

Figure 3.49: Errors associated with CXS DFEM using a quadratic trial space for different mesh spacing methodologies, at low resolutions.

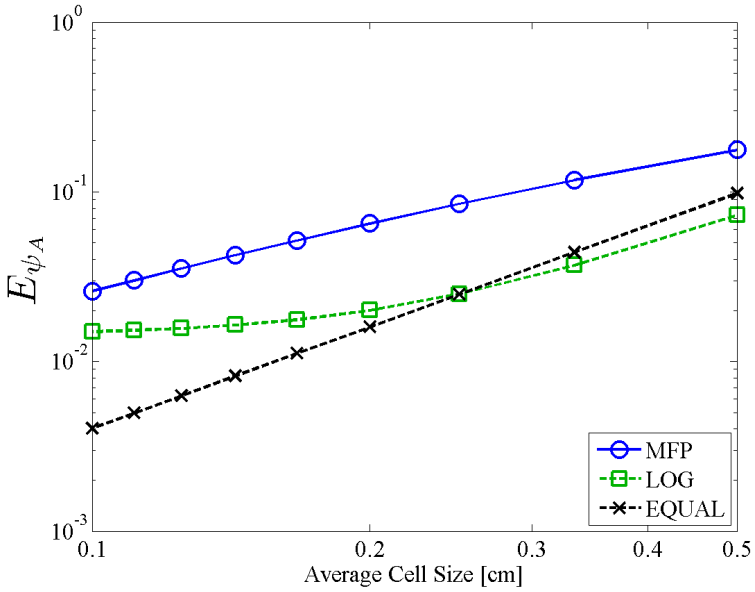

Figure 3.50: Errors associated with CXS DFEM using a quadratic trial space for different mesh spacing methodologies, at low resolutions.

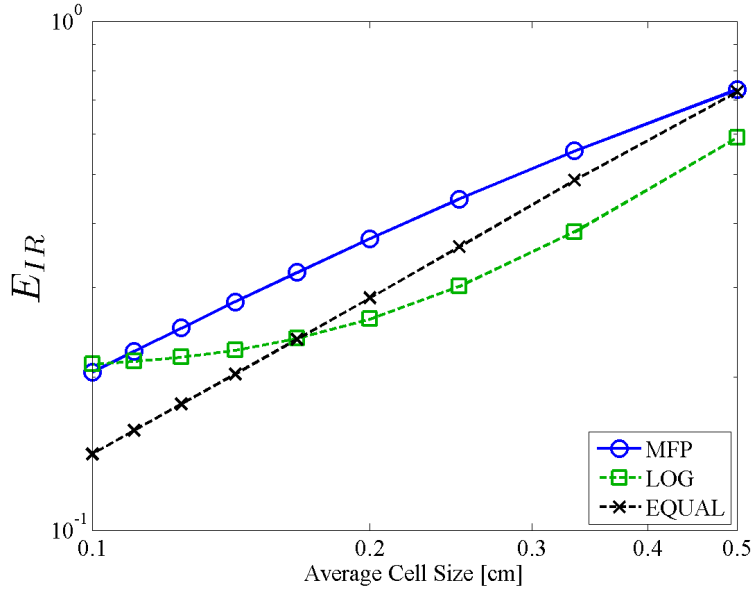

Figure 3.51: Errors associated with CXS DFEM using a quadratic trial space for different mesh spacing methodologies, at low resolutions.

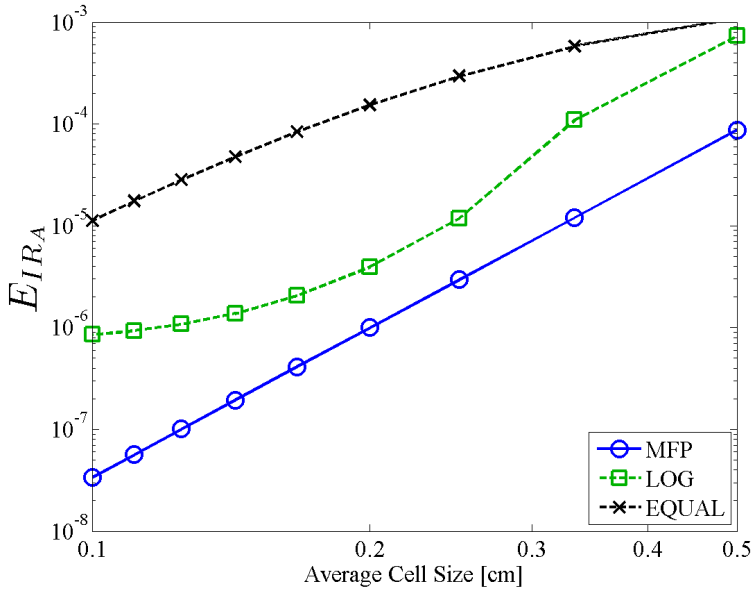

Figure 3.52: Errors associated with CXS DFEM using a quadratic trial space for different mesh spacing methodologies, at low resolutions.

3.4 Conclusions to Be Carried Forward

The SLXS Lobatto and SLXS Gauss schemes are accurate for problems S_N neutron transport problems that have within cell spatial interaction cross-section variation. Though no less accurate for the case of spatially varying cross sections that for cell-wise constant cross section, the SLXS Newton-Cotes scheme will no longer be considered. SLXS Newton-Cotes is not as asymptotically accurate as SLXS Gauss or SLXS Lobatto schemes, though the SLXS Lobatto scheme is equivalent to SLXS Newton-Cotes for linear and quadratic trial spaces. We continued to consider a self-lumping scheme using closed Newton-Cotes quadrature due to the robustness SL Newton-Cotes exhibited for the cell-wise constant cross section case. However, since no scheme other than linear SLXS Lobatto will yield strictly positive angular flux outflow for arbitrary within cell cross section variation, we conclude that SLXS

Lobatto has all of the strengths exhibited by SLXS Newton-Cotes, with the added benefit of increased accuracy.

We have also documented that the assumption of a cell-wise constant cross section will in general be less accurate than using a self-lumping scheme that explicitly accounts for within cell variation of cross section when constructing $\mathbf{R}$. Though the CXS DFEM was more accurate than self-lumping schemes in calculating E_{IR_A}, and $E_{\psi_{out}}$, when using the exact volume average cross section, this is more a function of our test problem being a pure absorber than an inherent benefit of CXS DFEM that will be true for all problems of interest.

4. ITERATIVE ACCELERATION FOR THE S_N NEUTRON TRANSPORT EQUATIONS

Thus far, we have omitted a discussion of how the global system of angular flux unknowns is solved, focusing instead on the solution of a single system of equations that describes the unknowns of a single spatial cell, for a single discrete direction. In Section 4.1, we explain the fundamental iterative techniques used to solve the neutron transport equation for all of the angular flux unknowns. In Section 4.2 we discuss two different synthetic acceleration techniques compatible with our chosen spatial discretizations of the transport equation. In Section 4.3 we derive the S_2 synthetic acceleration (S2SA) technique [34] and in Section 4.4 we derive a modified interior penalty (MIP) diffusion synthetic acceleration [35] (DSA) operator. Finally, in Section 4.5, we verify the implementation of each for a set of test problems with spatially constant and spatially varying interaction cross sections.

4.1 Iterative Solution of the Neutron Transport S_N Equations

To describe the iterative process by which the discrete ordinates neutron transport equations are solved, we re-visit the spatially analytic, steady-state, mono-energetic discrete ordinates neutron transport equation, but do not have a monolithic right hand side source. Rather, we treat the right hand side as having both an isotropic scattering component, and a fixed source component, as given in Eq. (4.1).

$$\mu \frac{d\psi_d}{dx} + \Sigma_t \psi_d = \frac{\Sigma_s}{4\pi}\phi + S_d \,. \tag{4.1}$$

The traditional practice is to solve Eq. (4.1) iteratively with Richardson iteration. Each Richardson iteration is referred to as a transport sweep, where for a fixed

right hand side, ψ_d is updated direction by direction, cell be cell, sweeping from each direction's incident boundary through the entire mesh with upwinding at cell interfaces. Introducing iteration index ℓ, this process can be written as:

$$\mu \frac{d\psi_d^{(\ell+1)}}{dx} + \Sigma_t \psi_d^{(\ell+1)} = \frac{\Sigma_s}{4\pi}\phi^{(\ell)} + S_d \,. \tag{4.2}$$

After we find, $\psi_d^{(\ell+1)}$, we update $\phi^{(\ell+1)}$ using the discrete ordinates definition:

$$\phi^{(\ell+1)} = 2\pi \sum_{d=1}^{N_{dir}} w_d \psi_d^{(\ell+1)} \,. \tag{4.3}$$

Though convergent, the source iteration process can converge arbitrarily slow, as shown by Larsen[36], when $\frac{\Sigma_s}{\Sigma_t} \to 1$. To accelerate the convergence of source iteration, the diffusion synthetic acceleration (DSA) technique was developed [37]. DSA is best explained through example.

To start, we consider to a single source iteration:

$$\mu \frac{d\psi_d^{(\ell+1/2)}}{dx} + \Sigma_t \psi_d^{(\ell+1/2)} = \frac{\Sigma_s}{4\pi}\phi^{(\ell)} + S_d \,. \tag{4.4}$$

Subtracting Eq. (4.1) from Eq. (4.4) yields:

$$\mu \frac{d\delta\psi_d^{(\ell+1/2)}}{dx} + \Sigma_t \delta\psi_d^{(\ell+1/2)} = \frac{\Sigma_s}{4\pi}\delta\phi^{(\ell)} \,, \tag{4.5}$$

where we have defined the iterative error of the angular flux, $\delta\psi_d^{(\ell+1)}$,

$$\psi_d = \psi_d^{(\ell+1)} + \delta\psi_d^{(\ell+1)} \,, \tag{4.6}$$

and scalar flux iterative error, $\delta\phi^{(\ell)}$

$$\phi = \phi^{(\ell)} + \delta\phi_d^{(\ell)} \,. \tag{4.7}$$

Subtracting $\frac{\Sigma_s}{4\pi}\delta\phi^{(\ell)}$ from both sides of Eq. (4.5), we see arrive at:

$$\mu\frac{d\delta\psi_d^{(\ell+1/2)}}{dx} + \Sigma_t\delta\psi_d^{(\ell+1/2)} - \frac{\Sigma_s}{4\pi}\delta\phi^{(\ell+1/2)} = \frac{\Sigma_s}{4\pi}\delta\phi^{(\ell)} - \frac{\Sigma_s}{4\pi}\delta\phi^{(\ell+1/2)} \,. \tag{4.8}$$

Recognizing:

$$\begin{aligned}
\phi &= \phi^{(\ell+1/2)} + \delta\phi^{(\ell+1/2)} \,, && \text{(4.9a)}\\
\phi &= \phi^{(\ell)} + \delta\phi^{(\ell)} \,, && \text{(4.9b)}\\
\phi^{(\ell+1/2)} - \phi^{(\ell)} &= \left(\phi - \delta\phi^{(\ell+1/2)}\right) - \left(\phi - \delta\phi^{(\ell)}\right) \,, \text{ and} && \text{(4.9c)}\\
\delta\phi^{(\ell)} - \delta\phi^{(\ell+1)} &= \phi^{(\ell+1/2)} - \phi^{(\ell)} \,, && \text{(4.9d)}
\end{aligned}$$

Eq. (4.8) becomes:

$$\mu\frac{d\delta\psi_d^{(\ell+1/2)}}{dx} + \Sigma_t\delta\psi_d^{(\ell+1/2)} = \frac{\Sigma_s}{4\pi}\delta\phi^{(\ell+1/2)} + \frac{\Sigma_s}{4\pi}\left(\phi^{(\ell+1/2)} - \phi^{(\ell)}\right) \,. \tag{4.10}$$

Equation 4.10 indicates that if we could solve a transport problem with a driving source equal to the difference between two scattering iterates,

$$\frac{\Sigma_s}{4\pi}\left(\phi^{(\ell+1/2)} - \phi^{(\ell)}\right) \,, \tag{4.11}$$

then we could get the iterative error of iteration $\delta\phi^{(\ell+1/2)}$, add to the $\phi^{(\ell+1/2)}$ we already have, and then have the exact solution, ϕ. However, solving Eq. (4.10) is as difficult as solving the original problem in Eq. (4.1). Alternatively, if we could solve

an approximation to Eq. (4.10), perhaps the result would satisfy:

$$\phi \approx \Delta\phi^{(\ell+1/2)} + \phi^{(\ell+1/2)}\,, \tag{4.12}$$

where $\Delta\phi^{(\ell+1/2)}$ comes from the lower order approximation to Eq. (4.10). The idea of using a lower order operator to approximately solve Eq. (4.10) is central to synthetic acceleration.

4.2 Qualitative Comparison of Different Synthetic Acceleration Techniques

We consider two synthetic acceleration techniques that have received significant attention in the neutron transport and thermal radiative transfer literature, S_2 synthetic acceleration (S2SA)[34] and diffusion synthetic acceleration (DSA)[37]. Both methods of synthetic acceleration has both advantages and disadvantages relating to the computational efficiency and iterative effectiveness of each method.

The S2SA method was shown to be iteratively effective in both slab and 1-D spherical geometries. Additionally, it is easily compatible with any DFEM spatial discretization of the S_N neutron transport equations. S2SA solves for ψ_+ and ψ_- using a single global matrix solve, rather than a direction by direction solve for the full transport ψ_d unknowns. The full transport scalar flux iterative update is then defined as,

$$\Delta\phi^{(\ell+1/2)} \approx 2\pi \left[w_+\psi_+ + w_-\psi_-\right]\,, \tag{4.13}$$

where $w_+,\ w_-$ correspond to the weights of a direction quadrature (typically Gauss) set with corresponding discrete direction μ_+ and μ_-, and $\psi_+,\ \psi_-$ are the fundamental unknowns of the S_2 discretization. Thus, S2SA uses the same local matrices of Eq. (3.2) as the full transport operator. However, the S2SA global matrix that must be inverted to solve for ψ_+ and ψ_- is extremely difficult to invert for multiple spatial

dimensions. This makes S2SA iteratively effective, but computationally expensive, limiting the extensibility of any techniques that require S2SA. But since our initial focus is on solving slab geometry neutron transport problems, we will continue to consider S2SA as it is readily adaptable to our new spatial discretization schemes. [34]

Gelbard and Hageman first showed in [37] that diffusion synthetic acceleration (DSA) could be used to accelerate the convergence of source iteration in neutron transport since the diffusion operator effectively attenuates the slowly varying error modes that hinder the convergence of source iteration. To be unconditionally effective, Larsen showed that DSA needed to be derived in a method consistent with the spatial and angular discretization of the transport equation [36]. Adams and Martin first showed that partially consistent diffusion discretizations could be used to effectively accelerate DFEM spatial discretizations of the neutron transport equation [38]. Though shown to be unconditionally stable for certain geometries the M4S DSA proposed in [38] has been shown to be unstable for unstructured multi-dimensional geometries [39]. To allow more general applicability, we wish to consider a more advanced DSA discretization. Alternative DSA discretizations that have been applied successfully to unstructured multi-dimensional geometries include: the partially consistent WLA DSA proposed in [40], the fully consistent DSA (FCDSA) proposed in [39], and the partially consistent MIP DSA proposed in [35]. WLA DSA produces a symmetric positive definite (SPD) diffusion matrix and is unconditionally stable, but the spectral radius of the WLA DSA scheme increases on distorted mesh cells and for optically thick cells with scattering ratios very close to unity [40, 39]. While the FCDSA scheme remains effective in optically thick cells, it creates a diffusion operator that is very difficult and costly to invert[39]. The MIP DSA discretization [35] of Wang and Ragusa generates a SPD diffusion operator, remains effective for all

cell optical thicknesses, has been successfully applied to high order DFEM S_N transport, and can be used with adaptive mesh refinement. Further, it was shown in [41] that the MIP DSA diffusion operator can be inverted very quickly using advanced preconditioners such as algebraic multi-grid. Thus, if we can show that a MIP DSA discretization is iteratively effective for neutron transport problems discretized with our higher order DFEM methods that account for the spatial variation of cross section within each cell, we will have found a scheme that is most likely to prove useful in meaningful (multiple spatial dimensions) thermal radiative transfer simulations.

We will derive an S2SA operator in Section 4.3 and MIP DSA operator in Section 4.4. The S2SA operator re-uses a lot of the transport sweep capability we have already developed with our high order DFEM of Chapter 3, but is computationally challenging to invert (in multiple spatial dimensions), and the MIP DSA operator we define in Section 4.4 requires significant derivation independent of the DFEM neutron transport methodology we have already derived, but is computationally efficient to invert.

4.3 S_2 Synthetic Acceleration

We begin our derivation by repeating the S_2, spatially analytic angular flux update equations from Morel[34] (Eqs. (12a) and (12b)), noting that we have elected to use ψ^+ and ψ^- instead of c^+ and c^-, Σ represents macroscopic interaction cross sections rather than σ, and we define $\phi = 2\pi \sum_d w_d \psi_d$:

$$\mu_+ \frac{d\psi^+}{dx} + \Sigma_t \psi^+ = \frac{\Sigma_s}{4\pi}\Delta\phi + \frac{\Sigma_s}{4\pi}\left(\phi^{(\ell+1/2)} - \phi^{(\ell)}\right) \tag{4.14a}$$

$$\mu_- \frac{d\psi^-}{dx} + \Sigma_t \psi^- = \frac{\Sigma_s}{4\pi}\Delta\phi + \frac{\Sigma_s}{4\pi}\left(\phi^{(\ell+1/2)} - \phi^{(\ell)}\right) . \tag{4.14b}$$

In Eqs. (4.14) we are assuming scattering is isotropic only. Spatially discretizing with a P degree DFEM as in Chapter 3, for an interior cell, c, we have:

$$\left(\mu_+ \mathbf{G}_+ + \mathbf{R}_{\Sigma_t}\right) \vec{\psi}_c^+ = \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\vec{\Delta\phi}_c + \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\left(\vec{\phi}_c^{(\ell+1/2)} - \vec{\phi}_c^{(\ell)}\right) + \mu_+\psi_{in,+}\vec{f}_+ \quad (4.15a)$$

$$\left(\mu_- \mathbf{G}_- + \mathbf{R}_{\Sigma_t}\right) \vec{\psi}_c^- = \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\vec{\Delta\phi}_c + \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\left(\vec{\phi}_c^{(\ell+1/2)} - \vec{\phi}_c^{(\ell)}\right) + \mu_-\psi_{in,-}\vec{f}_- \,. \quad (4.15b)$$

All reaction matrices are evaluated in cell c, unless otherwise noted with a subscript. In Eqs. (4.15), we note that $\mu_+ > 0$ and $\mu_- < 0$, and as such use the $\pm$ subscripts to define the appropriate $\mathbf{G}$ and $\vec{f}$, defined in Eqs. (4.16) and Eqs. (4.17), respectively:

$$\mathbf{G}_{+,i\ j} = b_i(1)b_j(1) - \int_{-1}^{1} \frac{db_i}{ds} b_j(s)\ ds \quad (4.16a)$$

$$\mathbf{G}_{-,i\ j} = -b_i(-1)b_j(-1) - \int_{-1}^{1} \frac{db_i}{ds} b_j(s)\ ds\,, \quad (4.16b)$$

$$\vec{f}_{+,i} = b_i(-1) \quad (4.17a)$$

$$\vec{f}_{-,i} = -b_i(1)\,. \quad (4.17b)$$

Noting that the inflow to cells on the interior is the outflow from the appropriate cell, for and using the definitions of Eqs. (2.7b) and Eqs. (2.8b), we define $\psi_{in,+}\vec{f}_+$ and $\psi_{in,-}\vec{f}_-$ entirely in terms of $\vec{\psi}_{c-1}^+$ and $\vec{\psi}_{c+1}^-$,

$$\psi_{in,+}\vec{f}_+ = \mathbf{U}_+\vec{\psi}_{c-1}^+ \quad (4.18a)$$

$$\psi_{in,-}\vec{f}_- = \mathbf{U}_-\vec{\psi}_{c+1}^-\,, \quad (4.18b)$$

where

$$\mathbf{U}_+ = \begin{bmatrix} b_1(-1) \\ \vdots \\ b_{N_P}(-1) \end{bmatrix} [b_1(1) \dots b_{N_P}(1)] \tag{4.19a}$$

$$\mathbf{U}_- = \begin{bmatrix} b_1(1) \\ \vdots \\ b_{N_P}(1) \end{bmatrix} [b_1(-1) \dots b_{N_P}(-1)] \, . \tag{4.19b}$$

Finally, assuming a symmetric angular quadrature,

$$\frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\vec{\Delta}\vec{\phi}_c = \frac{1}{2}\mathbf{R}_{\Sigma_s}\left(\vec{\psi}_c^+ + \vec{\psi}_c^-\right) \, , \tag{4.20}$$

we may write Eqs. (4.15) as:

$$\left(\mu_+\mathbf{G}_+ + \mathbf{R}_{\Sigma_t}\right)\vec{\psi}_c^+ - \frac{1}{2}\mathbf{R}_{\Sigma_s}\left(\vec{\psi}_c^+ + \vec{\psi}_c^-\right) - \mu_+\mathbf{U}_+\vec{\psi}_{c-1}^+ = \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\left(\vec{\phi}_c^{(\ell+1/2)} - \vec{\phi}_c^{(\ell)}\right) \tag{4.21a}$$

$$\left(\mu_-\mathbf{G}_- + \mathbf{R}_{\Sigma_t}\right)\vec{\psi}_c^- - \frac{1}{2}\mathbf{R}_{\Sigma_s}\left(\vec{\psi}_c^+ + \vec{\psi}_c^-\right) - \mu_-\mathbf{U}_-\vec{\psi}_{c+1}^- = \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\left(\vec{\phi}_c^{(\ell+1/2)} - \vec{\phi}_c^{(\ell)}\right) \, . \tag{4.21b}$$

Thus, the S2SA scheme uses all of the same matrices, in particular we think of $\mathbf{R}_{\Sigma_t}$ and $\mathbf{R}_{\Sigma_s}$, that we have already defined in our higher fidelity transport model. To find ψ^+ and ψ^-, we must then solve a system of $2 \times N_P \times N_{cell}$ linear equations with $2 \times N_P \times N_{cell}$ unknowns. It important to note that S2SA can accelerate not only the scalar flux, but also the first angular moment, J, of the S_N neutron transport equations

To complete our derivation of the S2SA scheme, we must now define appropriate boundary conditions. We will focus only on the leftmost boundary, though similar

equations for the right boundary can be defined analogously. It is sufficient for our purposes to consider problems only with specified incident flux boundary conditions and reflective boundaries. With incident flux conditions, we wish for the accelerated iterate to maintain the same inflow current as the specified boundary condition. Allowing for non-isotropic incident fluxes, the incident current, J^+ specified by our problem is:

$$\sum_{d=N_{dir}/2+1}^{N_{dir}} w_d \mu_d \psi_{in,d} \,. \tag{4.22}$$

Given the S2SA equations were derived via the assumption of a P_1 angular flux, the additive angular flux correction for direction d is:

$$\Delta\phi = 2\pi \left(\psi^+ + \psi^-\right) \tag{4.23a}$$

$$\Delta J = 2\pi \left(\mu_+ \psi^+ + \mu_- \psi^-\right) \tag{4.23b}$$

$$\Delta\psi_d = \frac{\Delta\phi}{4\pi} + \mu_d \frac{3\Delta J}{4\pi} \,. \tag{4.23c}$$

Wishing to maintain J^+, we have:

$$J^+ = 2\pi \sum_{d=N_{dir}/2+1}^{N_{dir}} w_d \mu_d \left[\psi_{in,d} + \frac{\Delta\phi}{4\pi} + \mu_d \frac{3\Delta J}{4\pi}\right], \tag{4.24}$$

which implies

$$0 = \sum_{d=N_{dir}/2+1}^{N_{dir}} w_d \mu_d \left[\frac{\Delta\phi}{4\pi} + \mu_d \frac{3\Delta J}{4\pi} \mu_d\right]. \tag{4.25}$$

Inserting the definitions of Eqs. (4.23), and allowing for DFEM interpolation points that do not exist at the left boundary:

$$0 = \sum_{d=N_{dir}/2+1}^{N_{dir}} w_d \mu_d \left[\frac{1}{2}\left(\psi_{in}^+ + \vec{L}\vec{\psi}_1^-\right) + \frac{3\mu_d}{2}\left(\mu_+ \psi_{in}^+ + \mu_- \vec{L}\vec{\psi}_1^-\right)\right], \tag{4.26}$$

where

$$\vec{L} = [b_1(-1) \dots b_{N_P}(-1)] \ . \tag{4.27}$$

Defining constants dependent on the S_N quadrature used,

$$\langle \mu^+ \rangle = \sum_{\mu_d > 0} w_d \mu_d \tag{4.28a}$$

$$\langle \mu^+ \rangle_2 = \sum_{\mu_d > 0} w_d \mu_d^2 \, , \tag{4.28b}$$

Eq. (4.26) becomes

$$0 = \frac{\langle \mu^+ \rangle}{2} \psi_{in}^+ + \frac{\langle \mu^+ \rangle}{2} \vec{L} \vec{\psi}_1^- + \frac{3}{2} \langle \mu^+ \rangle_2 \left(\mu_+ \psi_{in}^+ + \mu_- \vec{L} \vec{\psi}_1^- \right) \ . \tag{4.29}$$

Solving Eq. (4.29) for ψ_{in}^+,

$$\psi_{in}^+ = - \left(\frac{\langle \mu^+ \rangle}{2} + \frac{3}{2} \langle \mu^+ \rangle_2 \mu_+ \right)^{-1} \left(\frac{\langle \mu^+ \rangle}{2} + \frac{3}{2} \langle \mu^+ \rangle_2 \mu_- \right) \vec{L} \vec{\psi}_1^- \ . \tag{4.30}$$

Defining a constant, C_{inc},

$$C_{inc} = - \left(\frac{\langle \mu^+ \rangle}{2} + \frac{3}{2} \langle \mu^+ \rangle_2 \mu_+ \right)^{-1} \left(\frac{\langle \mu^+ \rangle}{2} + \frac{3}{2} \langle \mu^+ \rangle_2 \mu_- \right) \ , \tag{4.31}$$

and substituting into Eqs. (4.15), we have

$$\left(\mu_+ \mathbf{G}_+ + \mathbf{R}_{\Sigma_t} \right) \vec{\psi}_1^+ = \frac{1}{4\pi} \mathbf{R}_{\Sigma_s} \vec{\Delta \phi}_1 + \frac{1}{4\pi} \mathbf{R}_{\Sigma_s} \left(\vec{\phi}_1^{(\ell+1/2)} - \vec{\phi}_1^{(\ell)} \right) + \mu_+ C_{inc} \left(\vec{L} \vec{\psi}_1^- \right) \vec{f}_+ \tag{4.32a}$$

$$\left(\mu_- \mathbf{G}_- + \mathbf{R}_{\Sigma_t} \right) \vec{\psi}_1^- = \frac{1}{4\pi} \mathbf{R}_{\Sigma_s} \vec{\Delta \phi}_1 + \frac{1}{4\pi} \mathbf{R}_{\Sigma_s} \left(\vec{\phi}_1^{(\ell+1/2)} - \vec{\phi}_1^{(\ell)} \right) + \mu_- \psi_{in,-} \vec{f}_- \ . \tag{4.32b}$$

Noting that $\vec{f}_+ \vec{L}$ creates an $N_P \times N_P$ matrix, and inserting all of our other definitions,

we have the equations for cell 1 for incident angular flux boundary conditions

$$(\mu_+\mathbf{G}_+ + \mathbf{R}_{\Sigma_t})\,\vec{\psi}_1^+ - \frac{1}{2}\mathbf{R}_{\Sigma_s}\left(\vec{\psi}_1^+ + \vec{\psi}_1^-\right) - \mu_+ C_{inc}\vec{f}_+\vec{L}\vec{\psi}_1^- = \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\left(\vec{\phi}_1^{(\ell+1/2)} - \vec{\phi}_1^{(\ell)}\right) \tag{4.33a}$$

$$(\mu_-\mathbf{G}_- + \mathbf{R}_{\Sigma_t})\,\vec{\psi}_1^- - \frac{1}{2}\mathbf{R}_{\Sigma_s}\left(\vec{\psi}_1^+ + \vec{\psi}_1^-\right) - \mu_-\mathbf{U}_-\vec{\psi}_2^- = \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\left(\vec{\phi}_1^{(\ell+1/2)} - \vec{\phi}_1^{(\ell)}\right). \tag{4.33b}$$

For reflective conditions, we have a zero current on the left edge:

$$0 = 2\pi\sum_d w_d\mu_d\psi_d^{(\ell+1/2)} \tag{4.34}$$

$$0 = 2\pi\sum_d w_d\mu_d\left[\psi_d^{(\ell+1/2)} + \frac{\Delta\phi}{4\pi} + \mu_d\frac{3\Delta J}{2}\right]. \tag{4.35}$$

Equation 4.34 implies

$$0 = \sum_d w_d\mu_d\left[\frac{1}{2}\left(\psi_{in,+} + \vec{L}\vec{\psi}_1^-\right) + \frac{3\mu_d}{2}\left(\mu_+\psi_{in,+} + \mu_-\vec{L}\vec{\psi}_1^-\right)\right]. \tag{4.36}$$

Expanding our earlier quadrature definitions,

$$\langle\mu\rangle = \sum_d w_d\mu_d \tag{4.37a}$$

$$\langle\mu^2\rangle = \sum_d w_d\mu_d^2 \tag{4.37b}$$

we now solve for $\psi_{in,+}$ as a function of $\vec{\psi}_1^-$:

$$\psi_{in,+} = -\left(\frac{\langle\mu\rangle}{2} + \frac{3\mu_+}{2}\langle\mu^2\rangle\right)^{-1}\left(\frac{\langle\mu\rangle}{2} + \frac{3\mu_-}{2}\langle\mu^2\rangle\right)\vec{L}\vec{\psi}_1^-. \tag{4.38}$$

Defining another constant, C_{ref},

$$C_{ref} = -\left(\frac{\langle\mu\rangle}{2} + \frac{3\mu_+}{2}\langle\mu^2\rangle\right)^{-1}\left(\frac{\langle\mu\rangle}{2} + \frac{3\mu_-}{2}\langle\mu^2\rangle\right) \tag{4.39}$$

, the leftmost cell equations with a reflective boundary condition are:

$$\left(\mu_+\mathbf{G}_+ + \mathbf{R}_{\Sigma_t}\right)\vec{\psi}_1^+ - \frac{1}{2}\mathbf{R}_{\Sigma_s}\left(\vec{\psi}_1^+ + \vec{\psi}_1^-\right) - \mu_+ C_{ref}\vec{f}_+\vec{L}\vec{\psi}_1^- = \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\left(\vec{\phi}_1^{(\ell+1/2)} - \vec{\phi}_1^{(\ell)}\right) \tag{4.40a}$$

$$\left(\mu_-\mathbf{G}_- + \mathbf{R}_{\Sigma_t}\right)\vec{\psi}_1^- - \frac{1}{2}\mathbf{R}_{\Sigma_s}\left(\vec{\psi}_1^+ + \vec{\psi}_1^-\right) - \mu_-\mathbf{U}_-\vec{\psi}_2^- = \frac{1}{4\pi}\mathbf{R}_{\Sigma_s}\left(\vec{\phi}_1^{(\ell+1/2)} - \vec{\phi}_1^{(\ell)}\right). \tag{4.40b}$$

4.4 Modified Interior Penalty Diffusion Synthetic Acceleration

We now derive the modified interior penalty diffusion synthetic acceleration (MIP DSA) operator introduced by Ragusa and Wang [35], and also considered by Turcksin and Ragusa [41]. To accelerate the convergence of $N_P \times N_{cell}$ spatial unknowns of the scalar flux, we will need to solve a system of $N_P \times N_{cell}$ linear equations with $N_P \times N_{cell}$ unknowns. Adapted from [35], the MIP DSA update will attempt to solve the diffusion approximation of Eq. (4.10),

$$-\nabla\cdot D\nabla\Delta\phi + \Sigma_a\Delta\phi = \Sigma_s\left(\phi^{(\ell+1/2)} - \phi^{(\ell)}\right), \tag{4.41}$$

where we use the standard definitions [33],

$$D = \frac{1}{3\Sigma_t} \tag{4.42}$$

$$\Sigma_a = = \Sigma_t - \Sigma_s\,. \tag{4.43}$$

MIP DSA is presented in the standard finite element bilinear form

$$b_{MIP}(\Delta\phi, b_*) = l_{MIP}(b_*)\,, \tag{4.44}$$

with

$$\begin{aligned} b_{MIP}(\Delta\phi, b_*) = &(\Sigma_a \Delta\phi, b_*)_{\mathcal{D}} + \left(D\vec{\nabla}\Delta\phi, \vec{\nabla} b_*\right)_{\mathcal{D}} \\ &+ \left(\kappa_e \left[\!\left[\Delta\phi\right]\!\right], \left[\!\left[b_*\right]\!\right]\right)_{E_h^i} \\ &+ \left(\left[\!\left[\Delta\phi\right]\!\right], \left\{\!\left\{D\partial_n b_*\right\}\!\right\}\right)_{E_h^i} \\ &+ \left(\left\{\!\left\{D\partial_n \Delta\phi\right\}\!\right\}, \left[\!\left[b_*\right]\!\right]\right)_{E_h^i} \\ &+ (\kappa_e \Delta\phi, b_*)_{\partial\mathcal{D}^d} - \frac{1}{2}\left(\Delta\phi, D\partial_n b_*\right)_{\partial\mathcal{D}^d} \\ &- \frac{1}{2}\left(D\partial_n \Delta\phi, b_*\right)_{\partial\mathcal{D}^d}\,, \end{aligned} \tag{4.45}$$

and

$$l_{MIP}(b_*) = \left(\Sigma_s(\phi^{(\ell+1/2)} - \phi^{(\ell)}), b_*\right) + (J_{inc}, b_*)_{\partial\mathcal{D}^r}\,. \tag{4.46}$$

In Eqs. (4.44)-(4.46), b_* is any/every basis function (also referred to as test functions), the $(f,g)_{E_h^i}$ operator acting on quantities f and g is a sum over all cell interior edges:

$$(f,g)_{E_h^i} = \sum_{c=1}^{N_{cell}-1} (f,g)_{c+1/2}\,, \tag{4.47}$$

the jump operator, $\left[\!\left[f\right]\!\right]_{c+1/2}$, is defined as

$$\left[\!\left[f\right]\!\right]_{c+1/2} = f(x^+_{c+1/2}) - f(x^-_{c+1/2})\,, \tag{4.48}$$

where $x_{c+1/2}$ is the position of cell edge $c+1/2$, the average operator, $\{\!\{f\}\!\}_{c+1/2}$, is

$$\{\!\{f\}\!\}_{c+1/2} = \frac{1}{2}\left[f(x^+_{c+1/2}) + f(x^-_{c+1/2})\right], \tag{4.49}$$

and ∂_n is the edge directed normal dotted into the gradient operator. On the interior, the direction of ∂_n does not matter as long as it is consistent. We we assume the edge normal always pointing from left to right, thus $\partial_n = \frac{d}{dx}$. However, on the domain boundary, the edge directed normal must point outwards. Also in Eqs (4.45)-(4.46), the $(f,g)_{\mathcal{D}}$ operator is an integration of quantities f and g over the entire domain $\mathcal{D}$:

$$(f,g)_{\mathcal{D}} = \sum_{c=1}^{N_{cell}} (f,g)_c \tag{4.50}$$

$$(f,g)_c = \int_{x_{c-1/2}}^{x_{c+1/2}} f\ g\ dx\,. \tag{4.51}$$

Finally, κ_e is defined on edge $c-1/2$ as

$$\kappa_e = \kappa_{c-1/2} = \max\left(\frac{1}{4}, \kappa^{IP}_{c-1/2}\right), \tag{4.52}$$

and $\kappa^{IP}_{c-1/2}$ is defined as:

$$\kappa^{IP}_{c-1/2} = \frac{Z_{MIP}}{2} p_c(p_c+1)\frac{D_c}{\Delta x_c}\bigg|_{x^+_{c-1/2}} + \frac{Z_{MIP}}{2} p_{c-1}(p_{c-1}+1)\frac{D_{c-1}}{\Delta x_{c-1}}\bigg|_{x^-_{c-1/2}}, \tag{4.53}$$

where Z_{MIP} is a stability constant. Unless otherwise stated, we use $Z_{MIP} = 2$, as recommended by Wang and Ragusa[35]. Larger values of Z_{MIP} will increase scheme stability (by adding numerical diffusion), but if chosen too large, will affect the accuracy of the spatial discretization. Terms denoted by $\partial\mathcal{D}^d$ indicate Dirichlet boundary

terms (specified incident angular boundary condition), and terms applicable only for reflective (Neumann) boundary conditions are denoted with a $\partial\mathcal{D}^r$.

Focusing now on an interior mesh cell, c, and the N_P b_i that are non-zero in that cell, we now go about defining all of the terms of Eq. (4.45). First, we consider the volumetric integration terms, defining

$$(\Sigma_a \Delta\phi, b_*) = \mathbf{R}_{\Sigma_a} \vec{\Delta\phi}_c \,, \tag{4.54}$$

and

$$(D\vec{\nabla}\Delta\phi, \vec{\nabla} b_*) = \mathbf{S} \,. \tag{4.55}$$

In Eq. (4.55), we have defined:

$$\mathbf{S}_{ij} = \frac{2}{\Delta x_c} \int_{-1}^{1} \frac{1}{3\Sigma_t(s)} \frac{db_i}{ds} \frac{db_j}{ds} \, ds \,. \tag{4.56}$$

Now we treat the edge terms. We begin by defining $[\![\Delta\phi]\!]$ on each edge,

$$[\![\Delta\phi]\!]_{c-1/2} = \vec{L}\vec{\Delta\phi}_c - \vec{R}\vec{\Delta\phi}_{c-1} \,, \tag{4.57a}$$

$$[\![\Delta\phi]\!]_{c+1/2} = \vec{L}\vec{\Delta\phi}_{c+1} - \vec{R}\vec{\Delta\phi}_c \,, \tag{4.57b}$$

where we've defined

$$\vec{L} = [b_1(-1) \ldots b_{N_P}(-1)] \tag{4.58a}$$

$$\vec{R} = [b_1(1) \ldots b_{N_P}(1)] \,. \tag{4.58b}$$

Treating all test functions for cell c at once, we now define $[\![b_*]\!]$:

$$[\![b_*]\!]_{c-1/2} = \vec{L}^T - \vec{0} \tag{4.59a}$$

$$[\![b_*]\!]_{c+1/2} = \vec{0} - \vec{R}^T\,, \tag{4.59b}$$

where $\vec{0}$ is a length N_P column vector whose entries are all identically zero. Now we define the average operator on the edges $\{\!\{D\partial_n\Delta\phi\}\!\}$ and $\{\!\{D\partial_n b_*\}\!\}$. On edges $c-1/2$ and $c+1/2$,

$$\{\!\{D\partial_n\Delta\phi\}\!\}_{c-1/2} = \frac{D(x^+_{c-1/2})}{\Delta x_c}\vec{L}_s\vec{\Delta\phi}_c + \frac{D(x^-_{c-1/2})}{\Delta x_{c-1}}\vec{R}_s\vec{\Delta\phi}_{c-1} \tag{4.60a}$$

$$\{\!\{D\partial_n\Delta\phi\}\!\}_{c+1/2} = \frac{D(x^+_{c+1/2})}{\Delta x_{c+1}}\vec{L}_s\vec{\Delta\phi}_{c+1} + \frac{D(x^-_{c+1/2})}{\Delta x_c}\vec{R}_s\vec{\Delta\phi}_c\,, \tag{4.60b}$$

where

$$\vec{L}_s = \left[\left.\frac{db_1}{ds}\right|_{s=-1} \;\ldots\; \left.\frac{db_{N_P}}{ds}\right|_{s=-1}\right] \tag{4.61a}$$

$$\vec{R}_s = \left[\left.\frac{db_1}{ds}\right|_{s=1} \;\ldots\; \left.\frac{db_{N_P}}{ds}\right|_{s=1}\right]. \tag{4.61b}$$

On edges $c-1/2$ and $c+1/2$,

$$\{\!\{D\partial_n b_*\}\!\}_{c-1/2} = \frac{D(x^+_{c-1/2})}{\Delta x_c}\vec{L}_s^T \tag{4.62a}$$

$$\{\!\{D\partial_n b_*\}\!\}_{c+1/2} = \frac{D(x^-_{c+1/2})}{\Delta x_c}\vec{R}_s^T\,. \tag{4.62b}$$

Combining Eq. (4.54), Eq. (4.55), Eqs. (4.57), Eqs. (4.59), Eqs. (4.60), and Eqs.

(4.62), we have the left hand side of the cell c MIP DSA equations:

$$\begin{aligned}
&\mathbf{R}_{\Sigma_a}\vec{\Delta\phi}_c + \mathbf{S}\vec{\Delta\phi}_c \\
&\quad + \left\{ \left[\kappa_{c-1/2}\vec{L}^T \left(\vec{L}\vec{\Delta\phi}_c - \vec{R}\vec{\Delta\phi}_{c-1} \right) \right] - \left[\kappa_{c+1/2}\vec{R}^T \left(\vec{L}\vec{\Delta\phi}_{c+1} - \vec{R}\vec{\Delta\phi}_c \right) \right] \right\} \\
&\quad + \left\{ \frac{D(x^+_{c-1/2})}{\Delta x_c}\vec{L}_s^T \left(\vec{L}\vec{\Delta\phi}_c - \vec{R}\vec{\Delta\phi}_{c-1} \right) + \frac{D(x^-_{c+1/2})}{\Delta x_c}\vec{R}_s^T \left(\vec{L}\vec{\Delta\phi}_{c+1} - \vec{R}\vec{\Delta\phi}_c \right) \right\} \\
&\quad + \left\{ \vec{L}^T \left(\frac{D(x^+_{c-1/2})}{\Delta x_c}\vec{L}_s\vec{\Delta\phi}_c + \frac{D(x^-_{c-1/2})}{\Delta x_{c-1}}\vec{R}_s\vec{\Delta\phi}_{c-1} \right) \cdots \right. \\
&\quad \left. - \vec{R}^T \left(\frac{D(x^+_{c+1/2})}{\Delta x_{c+1}}\vec{L}_s\vec{\Delta\phi}_{c+1} + \frac{D(x^-_{c+1/2})}{\Delta x_c}\vec{R}_s\vec{\Delta\phi}_c \right) \right\}, \quad (4.63)
\end{aligned}$$

and the right hand side on the interior is obviously

$$\mathbf{R}_{\Sigma_s}\left(\vec{\phi}_c^{(\ell+1/2)} - \vec{\phi}_c^{(\ell)} \right). \quad (4.64)$$

In Eq. (4.63) and Eq. (4.64), $\mathbf{R}_{\Sigma_a}$, $\mathbf{R}_{\Sigma_s}$, and $\mathbf{S}$ are evaluated in cell c. Arranging Eq. (4.63) to isolate $\vec{\Delta\phi}_{c-1}$, $\vec{\Delta\phi}_c$, and $\vec{\Delta\phi}_{c+1}$, we have

$$\begin{aligned}
&\left\{ -\kappa_{c-1/2}\vec{L}^T\vec{R} - \frac{D(x^+_{c-1/2})}{\Delta x_c}\vec{L}_s^T\vec{R} + \frac{D(x^-_{c-1/2})}{\Delta x_{c-1}}\vec{L}^T\vec{R}_s \right\}\vec{\Delta\phi}_{c-1} \\
&\quad \left\{ \mathbf{R}_{\Sigma_a} + \mathbf{S} + \kappa_{c-1/2}\vec{L}^T\vec{L} + \kappa_{c+1/2}\vec{R}^T\vec{R} + \frac{D(x^+_{c-1/2})}{\Delta x_c}\vec{L}_s^T\vec{L} - \frac{D(x^-_{c+1/2})}{\Delta x_c}\vec{R}_s^T\vec{R} \right. \\
&\quad \left. + \frac{D(x^+_{c-1/2})}{\Delta x_c}\vec{L}^T\vec{L}_s - \frac{D(x^-_{c+1/2})}{\Delta x_c}\vec{R}^T\vec{R}_s \right\}\vec{\Delta\phi}_c \\
&\quad \left\{ -\kappa_{c+1/2}\vec{R}^T\vec{L} + \frac{D(x^-_{c+1/2})}{\Delta x_c}\vec{R}_s^T\vec{L} - \frac{D(x^+_{c+1/2})}{\Delta x_{c+1}}\vec{R}^T\vec{L}_s \right\}\vec{\Delta\phi}_{c+1} \quad (4.65)
\end{aligned}$$

We now consider the leftmost cell, so that we derive appropriate boundary conditions. We will derive equations for reflective and incident current boundary condi-

tions. Equations for the right boundary can be derived analogously. Starting with incident flux boundary conditions, we must define the $\left(\kappa_{1/2}\Delta\phi, b_*\right)$, $(\Delta\phi, D\partial_n b_*)$, and $(D\partial_n\Delta\phi, b_*)$ operators. They are:

$$\begin{aligned}
\left(\kappa_{1/2}\Delta\phi, b_*\right) &= \kappa_{1/2}\vec{L}^T\vec{L}\vec{\Delta}\vec{\phi}_1 && \text{(4.66a)}\\
(\Delta\phi, D\partial_n b_*) &= -\frac{2D(x^+_{1/2})}{\Delta x_1}\vec{L}_s^T\vec{L}\vec{\Delta}\vec{\phi}_1 && \text{(4.66b)}\\
(D\partial_n\Delta\phi, b_*) &= -\frac{2D(x^+_{1/2})}{\Delta x_1}\vec{L}^T\vec{L}_s\vec{\Delta}\vec{\phi}_1\,, && \text{(4.66c)}
\end{aligned}$$

where we remind the reader that on the left edge, $\partial_n = -\frac{\partial}{\partial x}$. Combing with the cell integral quantities, and the interior edge terms, for problems with an incident flux boundary condition, the leftmost cell equations are

$$\begin{aligned}
\mathbf{R}_{\Sigma_a}\vec{\Delta}\vec{\phi}_1 + \mathbf{S}\vec{\Delta}\vec{\phi}_1 + \kappa_{1/2}\vec{L}^T\vec{L}\vec{\Delta}\vec{\phi}_1 + \frac{D(x^+_{1/2})}{\Delta x_1}\vec{L}_s^T\vec{L}\vec{\Delta}\vec{\phi}_1 + \frac{D(x^+_{1/2})}{\Delta x_1}\vec{L}^T\vec{L}_s\vec{\Delta}\vec{\phi}_1 \\
- \kappa_{3/2}\vec{R}^T\left(\vec{L}\vec{\Delta}\vec{\phi}_2 - \vec{R}\vec{\Delta}\vec{\phi}_1\right) + \frac{D(x^-_{3/2})}{\Delta x_1}\vec{R}_s^T\left(\vec{L}\vec{\Delta}\vec{\phi}_2 - \vec{R}\vec{\Delta}\vec{\phi}_1\right) \\
- \vec{R}^T\left(\frac{D(x^-_{3/2})}{\Delta x_1}\vec{R}_s\vec{\Delta}\vec{\phi}_1 + \frac{D(x^+_{3/2})}{\Delta x_2}\vec{L}_s\vec{\Delta}\vec{\phi}_2\right) = \mathbf{R}_{\Sigma_s}\left(\vec{\phi}_1^{(\ell+1/2)} - \vec{\phi}_1^{(\ell)}\right)\,. \quad \text{(4.67)}
\end{aligned}$$

If the left edge satisfies a reflective condition, the leftmost cell equations are:

$$\begin{aligned}
\mathbf{R}_{\Sigma_a}\vec{\Delta}\vec{\phi}_1 + \mathbf{S}\vec{\Delta}\vec{\phi}_1 - \kappa_{3/2}\vec{R}^T\left(\vec{L}\vec{\Delta}\vec{\phi}_2 - \vec{R}\vec{\Delta}\vec{\phi}_1\right) + \\
\frac{D(x^-_{3/2})}{\Delta x_1}\vec{R}_s^T\left(\vec{L}\vec{\Delta}\vec{\phi}_2 - \vec{R}\vec{\Delta}\vec{\phi}_1\right) \\
- \vec{R}^T\left(\frac{D(x^-_{3/2})}{\Delta x_1}\vec{R}_s\vec{\Delta}\vec{\phi}_1 + \frac{D(x^+_{3/2})}{\Delta x_2}\vec{L}_s\vec{\Delta}\vec{\phi}_2\right) \\
= \mathbf{R}_{\Sigma_s}\left(\vec{\phi}_1^{(\ell+1/2)} - \vec{\phi}_1^{(\ell)}\right) + C_{MIP}\vec{L}^T\,, \quad \text{(4.68)}
\end{aligned}$$

where C_{MIP} is the incident partial current,

$$C_{MIP} = 2\pi \sum_{\mu_d > 0} w_d \mu_d \psi_d^{(\ell+1/2)} \,. \tag{4.69}$$

4.5 Numerical Results Verifying Implementation and Performance of S2SA and MIP DSA

We will consider two test problems to compare the effectiveness of S2SA and MIP DSA compared against source iteration, abbreviated as SI in the following results. To measure effectiveness, we will numerically compare the spectral radius, ρ of each scheme. The spectral radius of an iterative technique is the largest magnitude eigenvalue of the iterative operator. For an iteration scheme to be convergent, it must be true $d < 1$. After sufficiently many iterations, the following is true:

$$\left\| \delta\phi^{(\ell+1)} \right\| \leq \rho \left\| \delta\phi^{(ell)} \right\| \,, \tag{4.70}$$

where $\|\cdot\|$ is a valid norm and $\delta\phi^{(\ell)}$ is the error as defined in Eq. (4.7). We will numerically test our methods by considering a trivial solution problem. That is to say we will solve a problem with vacuum boundary conditions on all edges and no volumetric solutions. However, rather than initialize with a zero solution, we set the values of $\phi^{(0)}$ to be random numbers $\in [0, 1]$. We then use 150 iterations to estimate ρ, taking the last value to be the converged estimate of ρ. In Section 4.5.1 we present results for a scattering medium with a spatially constant cross section and in Section 4.5.2, we consider a scattering medium with spatially varying interaction cross section.

In the plots of Section 4.5.1 and Section 4.5.2 we will examine ρ as a function of scattering ratio, c, S_N order, DFEM trial space degree, iterative technique, DFEM

interpolation point type, and cell optical thickness, $\Sigma_t \Delta x$.

4.5.1 Spatially Constant Cross Section Scattering Problem

Our first test problem with is a 20 $[cm]$ slab with spatially homogeneous cross section. We use at least 10 cells in all simulations in Figs. 4.1-4.8. For $\Sigma_t \Delta x >= 2$, we increase Σ_t to the necessary value to obtain the desired $\Sigma_t \Delta x$, while holding $\Delta x = 2[cm]$ constant. For $\Sigma_t \Delta x < 2$, we hold $\Sigma_t = 1[cm^{-1}]$ constant, while increasing the number of cells to achieve the desired $\Sigma_t \Delta x$.

The purpose of the following section is to

1. verify that results from our S2SA implementation for linear SL Gauss are qualitatively similar to those presented in [34],

2. determine whether S2SA is an effective iterative acceleration technique, for SL Gauss and SL Lobatto DFEM with $P \in [1, 4]$,

3. verify that results from our MIP DSA implementation used in conjunction with SL Gauss, gives results similar to those presented in [35] for $P \in [1, 4]$, and

4. determine whether MIP DSA is an effective iterative acceleration technique for SL Lobatto DFEM.

In Fig. 4.1, we compare SI, S2SA, and MIP DSA for linear SL Gauss, with S_8 angular quadrature, and $c = 0.999$. We see that as expected[36], ρ for source iteration $\approx c$. Additionally, we see that MIP DSA achieves a ρ similar to that given in Figs. 6 and 7 of [35]. Likewise, our S2SA implementation estimates $\rho \leq \approx 0.2$, which agrees well with the Fourier analysis of [34] a true spectral radius of $0.2127c$. Since the analysis of [34] is for an infinite medium, we expect our estimate of ρ to be less. In Fig. 4.2, we compare the ρ estimates of SI, S2SA, and MIP DSA for linear SL Lobatto

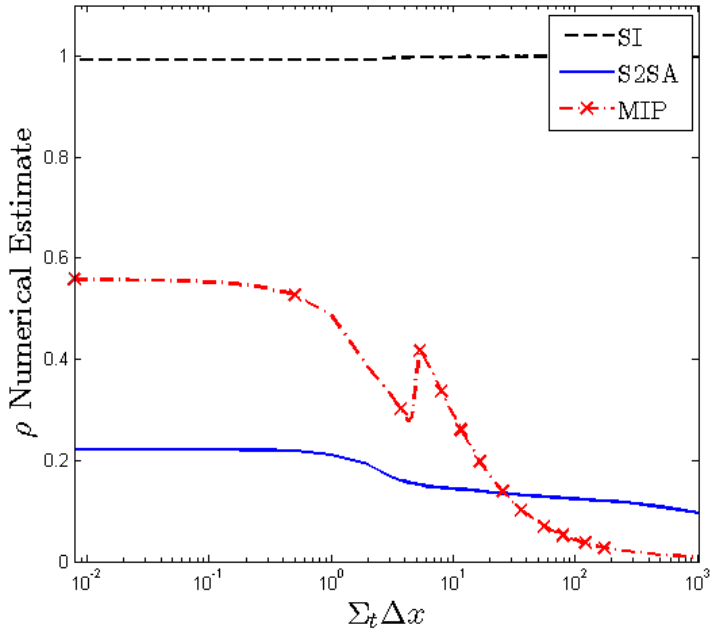

Figure 4.1: Estimates of ρ for different iterative techniques for S_8, $c = 0.999$, linear SL Gauss.

differencing. Results indicate that both S2SA and MIP DSA are compatible with SL Lobatto neutron transport. MIP DSA for linear SL Lobatto exhibits the same peaking as observed with linear SL Gauss, though the peak is slightly smoother. We now examine the sensitivity of S2SA and MIP DSA to different values of c, S_N, and DFEM trial space degree. From Fig. 4.1 and Fig. 4.2, we expect that the choice of DFEM interpolation point will change the estimate of ρ, but will not cause a method to become unstable.

4.5.1.1 S2SA Spectral Radius Sensitivity

In Fig. 4.3 and Fig. 4.4, we compute ρ for S2SA as a function of S_N order for linear SL Gauss and linear SL Lobatto, respectively. In both Figs. 4.3-4.4, the higher the S_N order, the larger ρ, across all cell optical thicknesses, though there is little

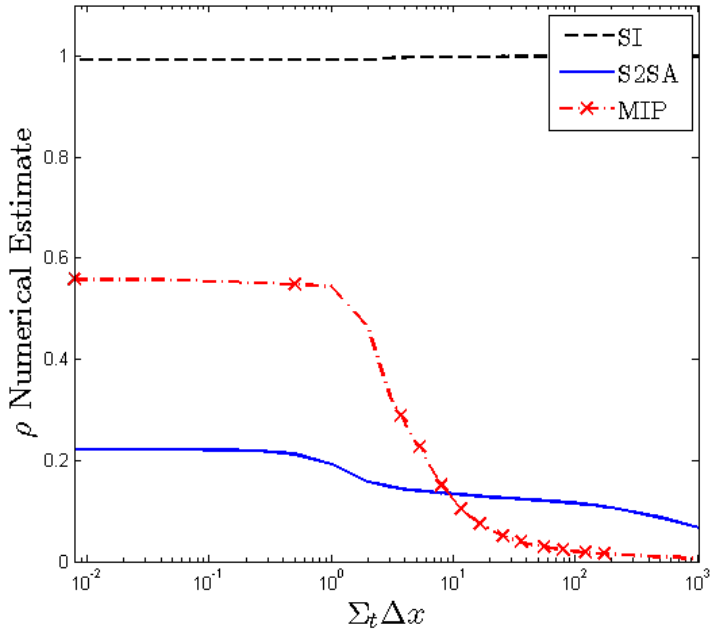

Figure 4.2: Estimates of ρ for different iterative techniques for S_8, $c = 0.999$, linear SL Lobatto.

difference between S_8 and S_{16}. Comparing our estimates of ρ of S2SA, as a function of c, for linear SL Gauss and SL Lobatto in Fig. 4.5 and Fig. 4.6, respectively, we conclude that the closer c is to unity, the larger, ρ, but there is negligible increase from $c = 0.999$ to $c = 0.9999$, except in very optically thick cells, but ρ is not largest with the S2SA scheme for large values of $\Sigma_t \Delta x$. Finally, in Fig. 4.7 and Fig. 4.8, we compare ρ for S_8, $c = 0.999$ S2SA with SL Gauss and SL Lobatto differencing, respectively, as a function of DFEM trial space degree. ρ remains nearer the optically thin value for larger values of $\Sigma_t \Delta x$ the higher the DFEM trial space degree, but S2SA remains stable for all P. Regardless of c, P, $\Sigma_t \Delta x$, and S_N order, we numerically verified that the S2SA matrix that must be inverted was neither symmetric or positive definite, requiring direct (Gaussian) elimination to invert the matrix.

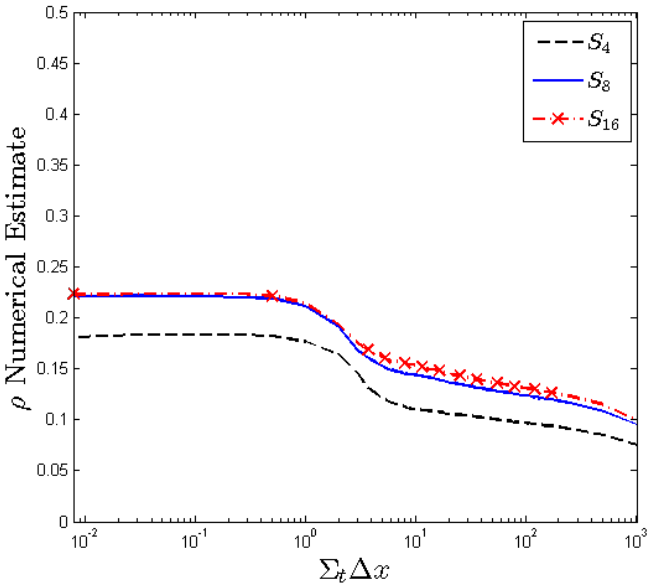

Figure 4.3: Estimate of ρ for S2SA as a function of S_N order for $c = 0.999$ for linear SL Gauss.

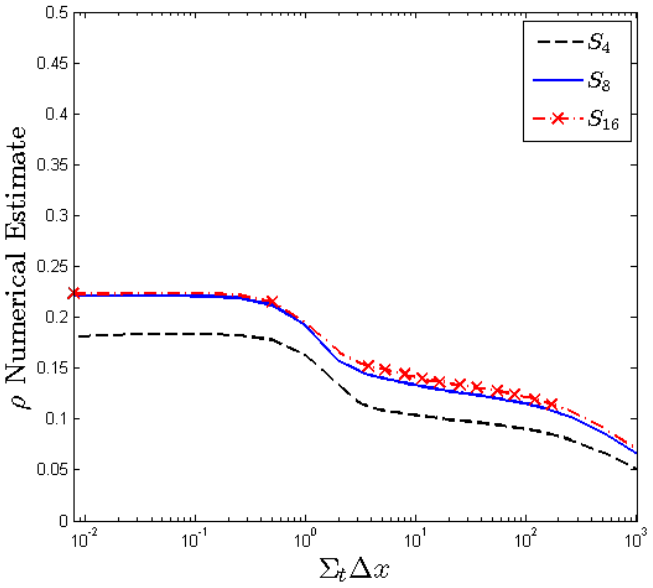

Figure 4.4: Estimate of ρ for S2SA as a function of S_N order for $c = 0.999$ for linear SL Lobatto.

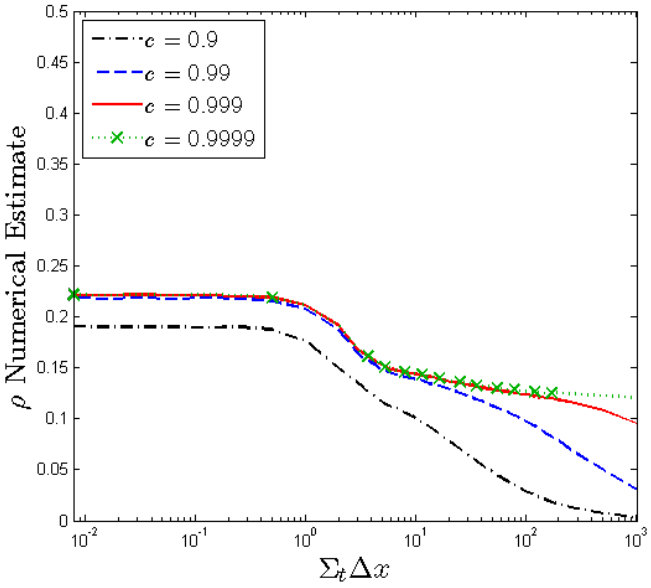

Figure 4.5: Estimates of ρ for S2SA as a function of c for S_8 linear SL Gauss.

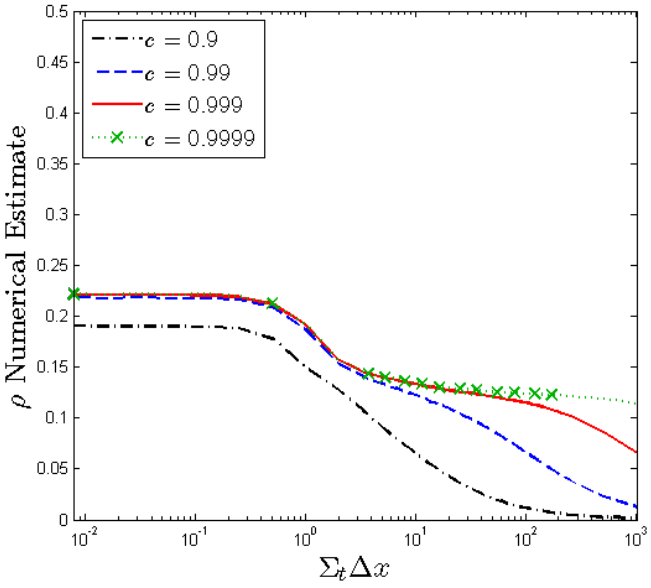

Figure 4.6: Estimates of ρ for S2SA as a function of c for S_8 linear SL Lobatto.

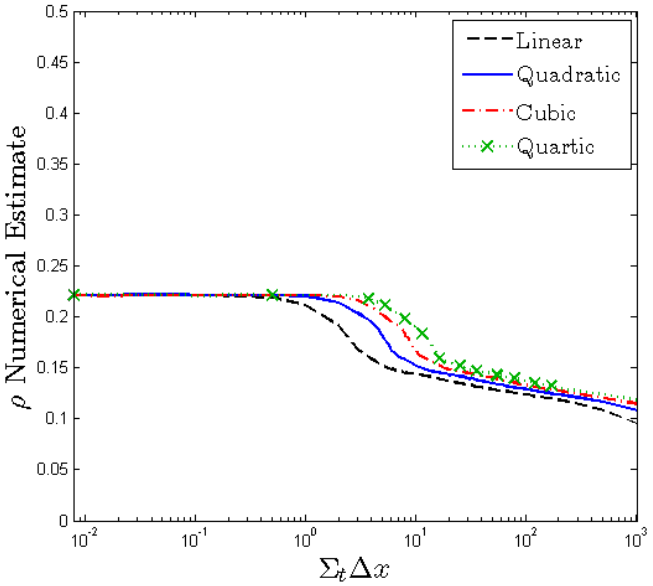

Figure 4.7: Estimates of ρ for S2SA as a function of S_8, $c = 0.999$, SL Gauss as a function of trial space degree.

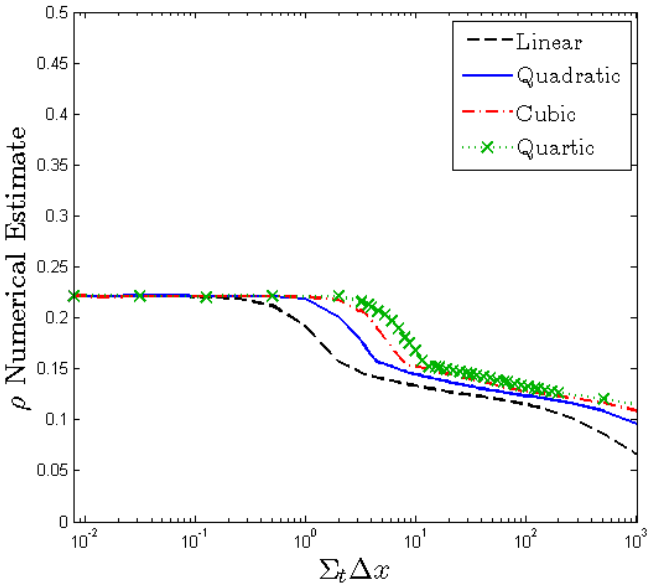

Figure 4.8: Estimates of ρ for S2SA as a function of S_8, $c = 0.999$, SL Lobatto as a function of trial space degree.

4.5.1.2 MIP DSA Spectral Radius Sensitivity

In Fig. 4.9 and Fig. 4.10 we compare ρ for MIP DSA as a function of S_N order, for linear SL Gauss and linear SL Lobatto, respectively, with $c = 0.999$. As with S2SA, the higher S_N order, the larger ρ is for the MIP DSA scheme. However, the value of ρ obtained with S_8 is only slightly smaller than the ρ of S_{16}, and as such we prefer the reduced computational work of S_8 simulations to S_{16} in examining iterative effectiveness.

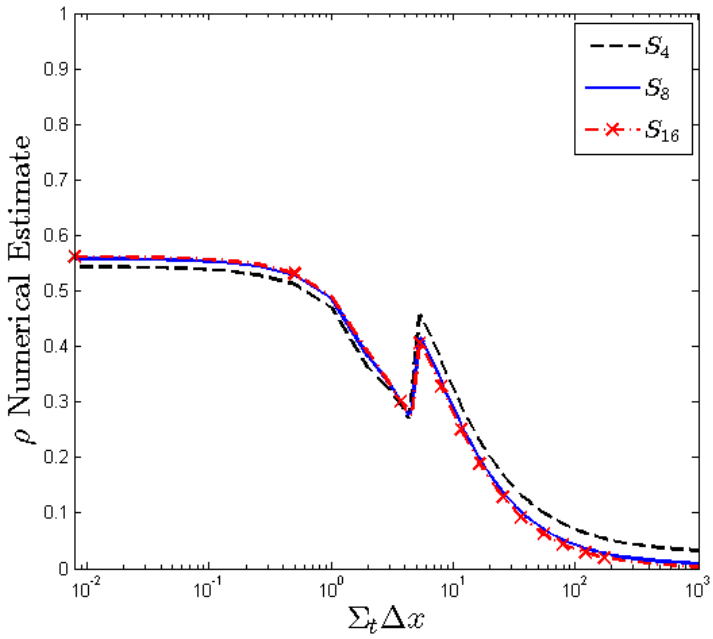

Figure 4.9: Estimate of ρ for MIP DSA as a function of S_N order for $c = 0.999$ linear SL Gauss.

Figure 4.11 and Fig. 4.12 examine the effect of c on ρ for MIP DSA with the linear SL Gauss and SL Lobatto schemes. The general trend is that as $c \to 1$, the larger the estimate of ρ, but both SL Gauss and SL Lobatto with MIP DSA remain

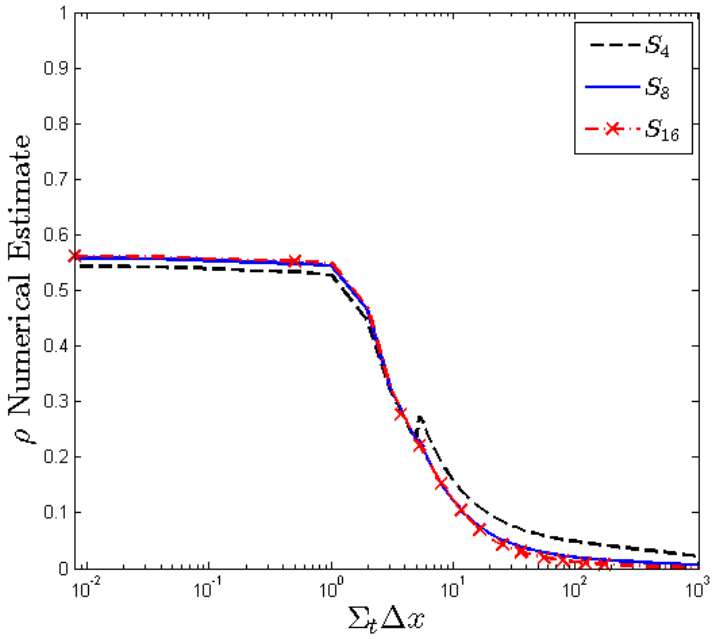

Figure 4.10: Estimate of ρ for MIP DSA as a function of S_N order for $c = 0.999$ linear SL Lobatto.

iteratively stable.

It is as this point that we remember that the MIP DSA scheme uses a constant, κ_{IP}, which depends on a chosen constant. In all of the results that have presented to this point, we have assumed that Z_{MIP} of Eq. (4.53), $Z_{MIP} = 2$. Just as Wang did in his dissertation[42], we now consider $Z_{MIP} = 4$. Estimates of ρ for MIP DSA as a function of c for SL Gauss and SL Lobatto, with $Z_{MIP} = 4$ are given in Fig. 4.13 and Fig. 4.14, respectively.

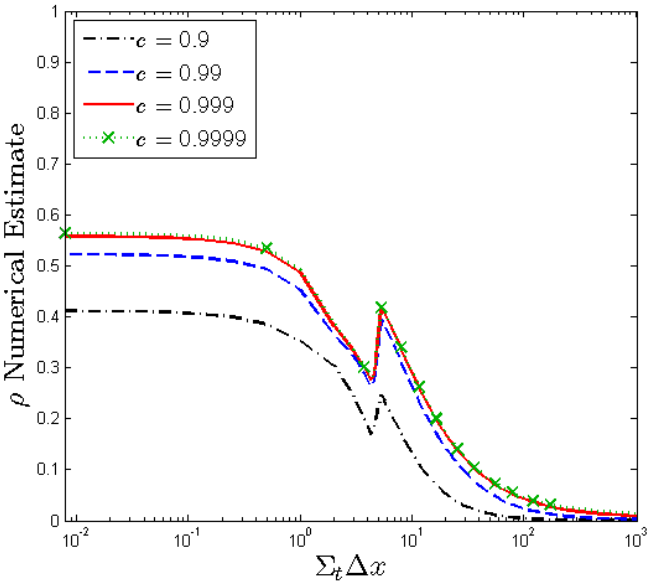

Figure 4.11: Estimate of ρ for MIP DSA as a function of c for S_8 linear SL Gauss.

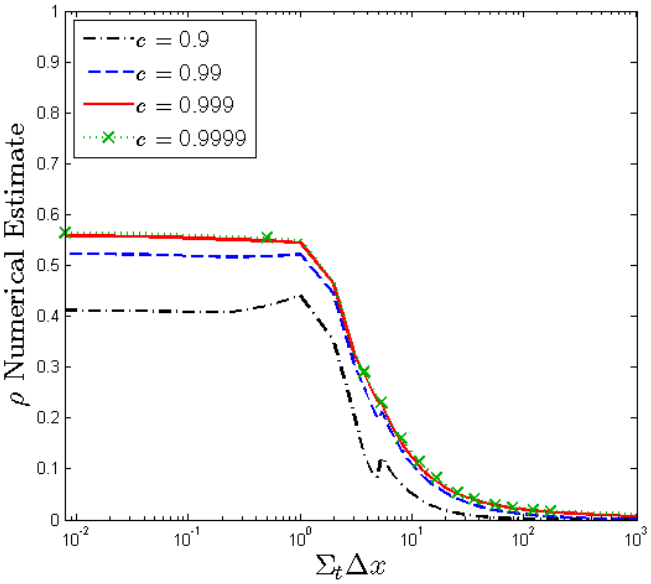

Figure 4.12: Estimate of ρ for MIP DSA as a function of c for S_8 linear SL Lobatto.

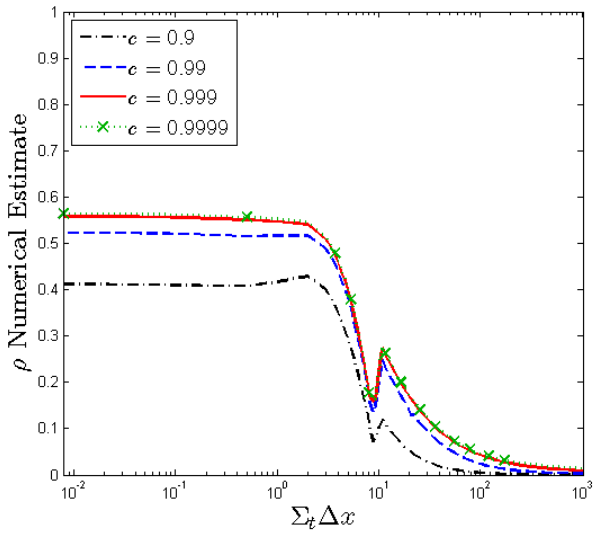

Figure 4.13: ρ for MIP DSA as a function of c for S_8 linear SL Gauss, $Z_{MIP} = 4$.

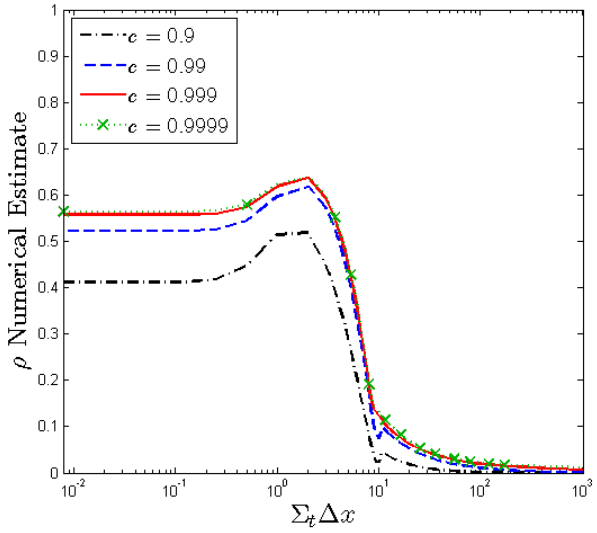

Figure 4.14: ρ for MIP DSA as a function of c for S_8 linear SL Lobatto, $Z_{MIP} = 4$.

There are small differences between Figs. 4.11-4.12 and Figs. 4.13-4.14 . With $Z_{MIP} = 4$ in Fig. 4.13 SL Gauss maintains a slightly higher value of ρ at small optical thicknesses than when $Z_{MIP} = 2$, and the the transitional drop from intermediate to larger rho at small optical thicknesses to smaller ρ for large cell optical thicknesses is more pronounced. With $Z_{MIP} = 4$, SL Lobatto displays a modest, smooth increase in ρ over a range of intermediate $\Sigma_t \Delta x$ and exhibits a sharper drop in ρ as well. Currently, the choice of $Z_{MIP} = 2$ or $Z_{MIP} = 4$ seems to have negligible effect on ρ, but we will retain adjustments to Z_{MIP} as a tool should any combination of spatial discretization and MIP DSA appear unstable.

Having determined that $c = 0.999$ and S_8 will yield nearly maximal ρ, we calculate ρ as a function of P for MIP DSA acceleration of SL Gauss and SL Lobatto transport spatial discretizations in Fig. 4.15 and Fig. 4.16, respectively, using $Z_{MIP} = 2$.

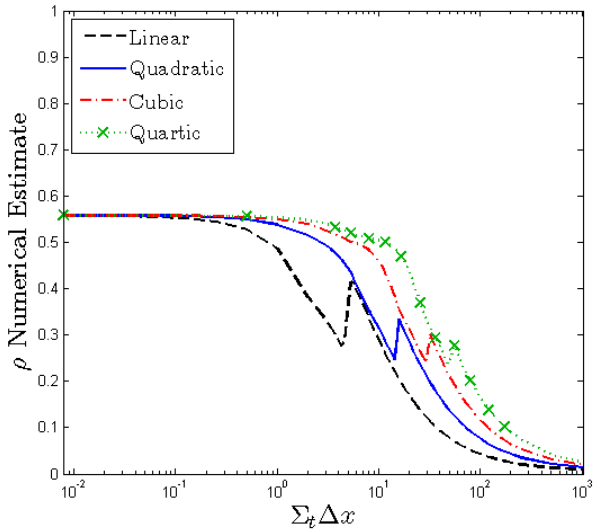

Figure 4.15: Estimate of ρ for MIP DSA as a function of P for S_8, $c = 0.999$, SL Gauss with $Z_{MIP} = 2$.

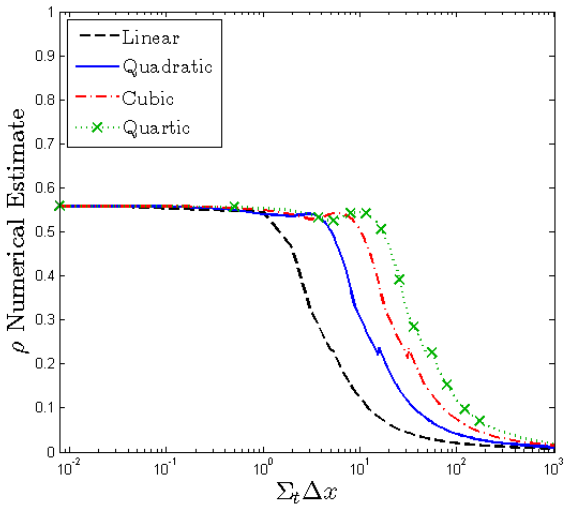

Figure 4.16: Estimate of ρ for MIP DSA as a function of P for S_8, $c = 0.999$, SL Lobatto with $Z_{MIP} = 2$.

As P increases, both SL Gauss and SL Lobatto accelerated with MIP DSA remain iteratively effecting. Similarly, if we consider $Z_{MIP} = 4$, both SL Gauss and SL Lobatto spatial differencing schemes remain stable.

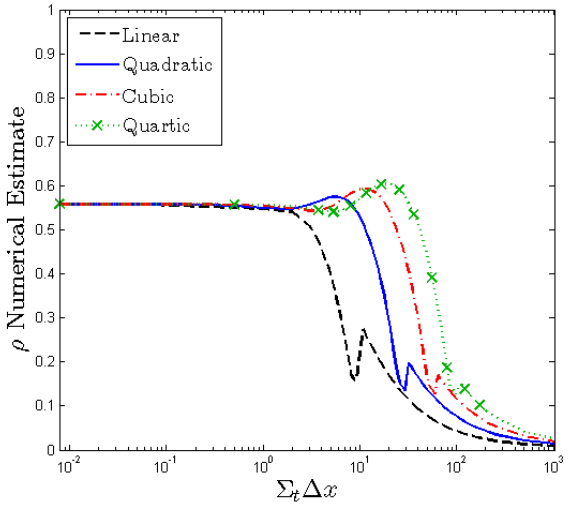

Figure 4.17: ρ for MIP DSA as a function of P for S_8, $c = 0.999$, SL Gauss with $Z_{MIP} = 4$.

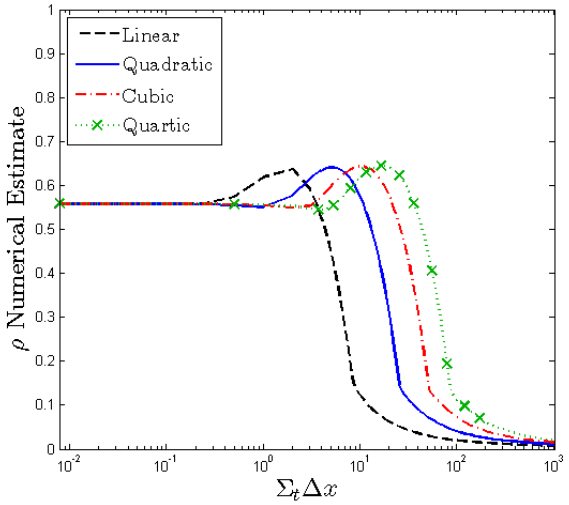

Figure 4.18: Estimates of ρ for MIP DSA as a function of P for S_8, $c = 0.999$, SL Lobatto with $Z_{MIP} = 4$.

4.5.2 Spatially Varying Cross Section Scattering Problem

To test the effectiveness of MIP DSA and S2SA for a problem with a spatially varying cross section, we again consider a slab $x \in [0, 20[cm]]$. We impose

$$\Sigma_t(x) = \Sigma_{t,0} \exp\left[\frac{|(10-x)|}{2}\right] . \tag{4.71}$$

We hold c constant in space. We will estimate ρ as a function of $\overline{\Sigma_t \Delta x}$, the average optical thickness of each mesh cell. For values of $\overline{\Sigma_t \Delta x} > 2$, we will use 10 mesh cells, and adjust $\Sigma_{t,0}$ to achieve the desired optical thickness. For values of $\overline{\Sigma_t \Delta x} < 2$, we will hold $\Sigma_{t,0}$ constant, and increase the number of mesh cells. We wish to maintain a total slab optical thickness of at least 20 mean free paths. Since,

$$\text{Total Mean Free Path} = 2\int_0^{10\ cm} \Sigma_{t,0} \exp\left[\frac{10-x}{2}\right]\ dx \tag{4.72a}$$

$$\text{Total Mean Free Path} = 4\Sigma_{t,0}\left(\exp[5]-1\right) , \tag{4.72b}$$

the minimum value of $\Sigma_{t,0}$ is:

$$\Sigma_{t,0} = \frac{5}{\exp[5]-1} . \tag{4.72c}$$

For values of $\overline{\Sigma_t \Delta x} > 2$,

$$10\overline{\Sigma_t \Delta x} = 4\Sigma_{t,0}\left(\exp[5]-1\right) \tag{4.72d}$$

$$\Sigma_{t,0} = \frac{5\overline{\Sigma_t \Delta x}}{2\left(\exp[5]-1\right)} . \tag{4.72e}$$

4.5.2.1 S2SA

In Fig. 4.19 and Fig. 4.20, we plot the estimate of ρ for the variable cross section problem, with $c = 0.999$ and S_8 angular quadrature for S2SA using SL Gauss and SL Lobatto schemes, respectively, as a function of trial space degree. S2SA remains

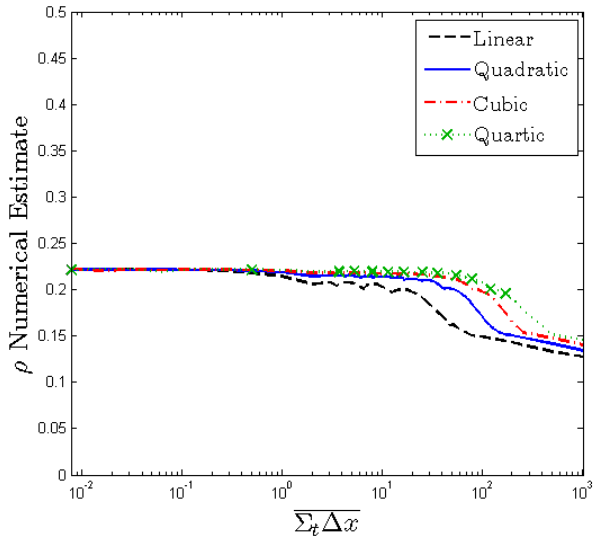

Figure 4.19: Estimate of ρ for S2SA as a function of $\overline{\Sigma_t \Delta x}$ and P, with SLXS Gauss.

unconditionally stable for the case of a spatially varying cross section, for all P considered, for both SLXS Gauss and SLXS Lobatto schemes.

4.5.2.2 MIP DSA

We now examine ρ for the MIP DSA scheme as a function of P for the variable cross section problem, with $c = 0.999$, S_8 angular quadrature, and $Z_{MIP} = 4$. Results for SLXS Gauss are given in Fig. 4.21 and ρ for MIP DSA with SLXS Lobatto scheme is given in Fig. 4.22. Since $\rho < 1$ in Figs. 4.21-4.22, we conclude that MIP DSA

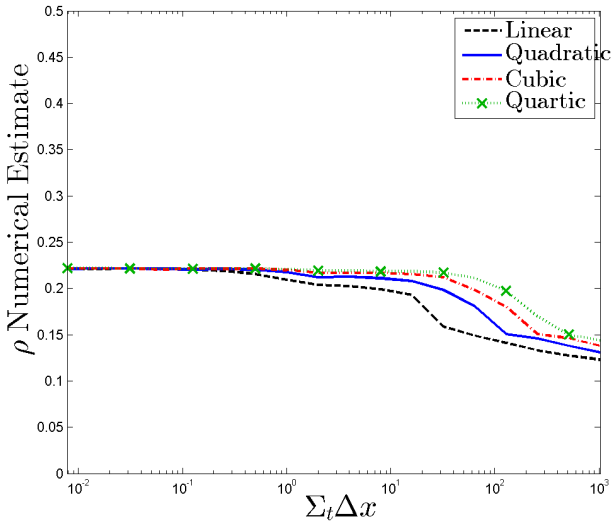

Figure 4.20: Estimate of ρ for S2SA as a function of $\overline{\Sigma_t \Delta x}$ and P, with SLXS Lobatto.

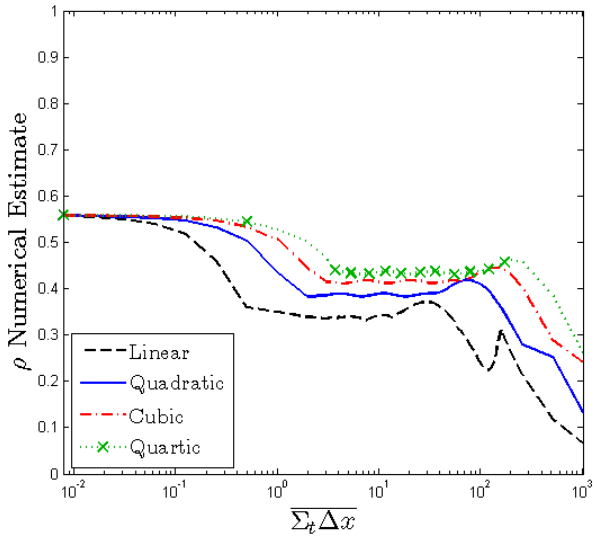

Figure 4.21: Estimate of ρ for MIP DSA with SLXS Gauss as a function of $\overline{\Sigma_t \Delta x}$ and P.

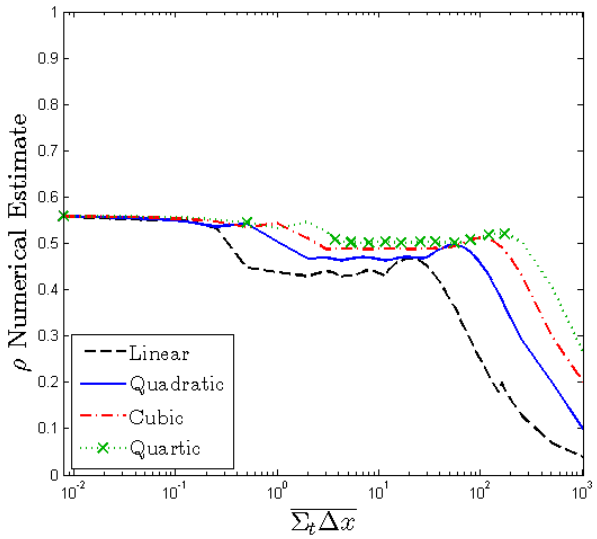

Figure 4.22: Estimate of ρ for MIP DSA with SLXS Lobatto as a function of $\overline{\Sigma_t \Delta x}$ and P.

remains a stable iterative scheme for SLXS Gauss and SLXS Lobatto, for all P considered when $Z_{MIP} = 2$ for problems with spatially varying cross sections.

4.6 Conclusions

Both S2SA and MIP DSA are stable iterative acceleration techniques for problems with spatially constant and spatially varying cross section. We have verified our implementations of S2SA and MIP DSA by comparing to published, spatially constant cross section results. While the S2SA technique is more effective than MIP DSA (smaller ρ), the S2SA matrix that must be inverted is not symmetric or positive definite. Thus, S2SA requires direct matrix inversion which in general is not feasible for large systems of equations. Conversely, though MIP DSA is not as effective as S2SA, the MIP DSA matrix is SPD, thus it can be inverted much more efficiently [faster] [41] than the S2SA matrix can be inverted. As such, to demonstrate that self-lumping schemes that account for within cell spatial variation of quadrature can be

used for more realistic problems, we first employ S2SA to accelerate neutron transport within a model fuel depletion problem in Chapter 5. However, since S2SA is really only practical for small, one dimensional problems, we will apply a variant of MIP DSA to accelerate the iterative process required for thermal radiative transfer problems in Chapter 6.

5. FUEL DEPLETION PROBLEM

We now consider a fuel depletion problem to illustrate that self-lumping DFEM schemes remain accurate for more complex physics than simply a purely absorbing medium. The depletion method we use [time stepping scheme, time step size, etc.] is chosen for its simplicity, not for its fidelity relative to state-of-the art depletion methodologies. Our goal is to assess the accuracy of *s*patial discretization methods for problems with spatially varying cross sections, not to propose a new depletion method. However, we stress that self-lumping methods can be implemented with any time depletion method or time stepping scheme since implementation of self-lumping only requires changes pertaining to the spatial discretization.

5.1 Problem Physical Description

We will consider a 1-D slab fuel depletion problem. The dimensions and layout of the fuel and moderator layers lattice is shown in Fig. 5.1.

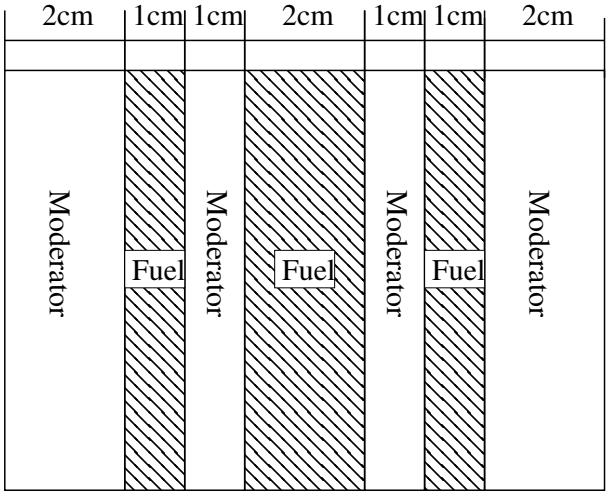

Figure 5.1: Depletion problem fuel/moderator lattice.

Initially, each fuel region is a homogeneous mixture containing only fissile ^{235}U and fertile ^{238}U nuclei, with compositions given in Table 5.1. As fuel depletion

Table 5.1: Fuel atom density data.

Fuel Density $[g/cm^3]$	10.97
Atom Fraction ^{235}U	0.05
Atom Fraction ^{238}U	0.95
Fuel Molecular Weight $[amu]$	270.03

progresses, the isotopic composition of fuel changes. The moderator is light water and its composition, given in Table 5.2, does not change with irradiation. We track

Table 5.2: Water atom density data.

Water Density $[g/cm^3]$	1
Atom Fraction ^{1}H	$\frac{2}{3}$
Atom Fraction ^{16}O	$\frac{1}{3}$
Water Molecular Weight $[amu]$	18.02

five nuclide types in the fuel during the fuel depletion problem: fissile, fertile, parasitic absorber fission product, scattering fission product, and inert, whose spatial nuclide densities, $[atom/cm^3]$, are respectively denoted as N_{FS}, N_{FT}, N_{FP-A}, N_{FP-S}, and N_I.

We use a two-energy-group approximation with the standard numbering convention- the lower the group number, the faster the neutron. All neutrons are born fast, there is no thermal upscattering, and we assume all scattering and fission is isotropic in angle. We also assume that if neutron absorption leads to isotopic transmutation,

the transmutation occurs at the time of absorption, there are no radioactive decay chains. All possible transmutation paths are shown in Fig. 5.2.

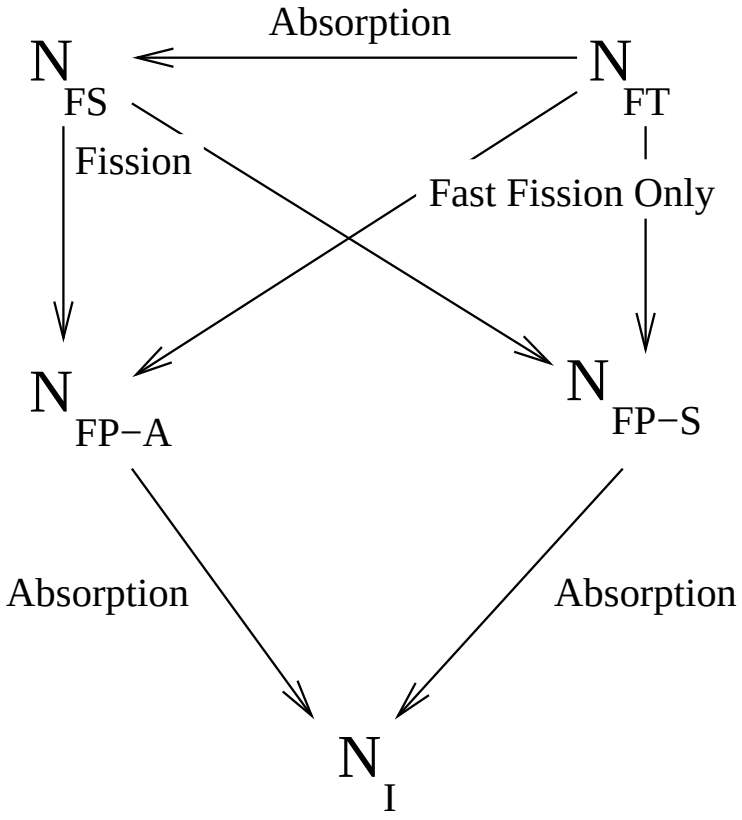

Figure 5.2: Depletion problem possible transmutation paths and mechanisms.

Under these assumptions, and given the transmutation paths given in Fig. 5.2, the fully analytic, nonlinear depletion equations are:

$$\mu\frac{\partial\psi_1}{\partial x}+\Sigma_{t,1}\psi_1=\frac{\Sigma_{s,1\to1}}{4\pi}\phi_1+\frac{1}{k}\left(\frac{\nu\Sigma_{f,1}}{4\pi}\phi_1+\frac{\nu\Sigma_{f,2}}{4\pi}\phi_2\right) \tag{5.1a}$$

$$\mu\frac{\partial\psi_2}{\partial x}+\Sigma_{t,2}\psi_2=\frac{\Sigma_{s,2\to2}}{4\pi}\phi_2+\frac{\Sigma_{s,1\to2}}{4\pi}\phi_1 \tag{5.1b}$$

$$\frac{\partial N_{FS}}{\partial t} = -N_{FS}\left[(1-\gamma_{FS,1})\sigma_{a,FS,1}\phi_1 + (1-\gamma_{FS,2})\sigma_{a,FS,2}\phi_2\right] + N_{FT}[\gamma_{FT,1}\sigma_{a,FT,1}\phi_1 + \gamma_{FT,2}\sigma_{a,FT,2}\phi_2] \quad (5.1c)$$

$$\frac{\partial N_{FT}}{\partial t} = -N_{FT}[\sigma_{a,FT,1}\phi_1 + \sigma_{a,FT,2}\phi_2] \quad (5.1d)$$

$$\frac{\partial N_{FP-S}}{\partial t} = N_{FS}\left[(1-p_{FS,1})y_{FS,1}\sigma_{a,FS,1}\phi_1 + (1-p_{FS,2})y_{FS,2}\sigma_{a,FS,2}\phi_2\right] + N_{FT}\left[(1-p_{FT,1})y_{FT}\sigma_{a,FT,1}\phi_1\right] - N_{FS}\left[\sigma_{a,FP-S,1}\phi_1 + \sigma_{a,FP-S,2}\phi_2\right] \quad (5.1e)$$

$$\frac{\partial N_{FP-A}}{\partial t} = N_{FS}\left[p_{FS,1}y_{FS,1}\sigma_{a,FS,1}\phi_1 + p_{FS,2}y_{FS,2}\sigma_{a,FS,2}\phi_2\right] + N_{FT}p_{FT,1}y_{FT,1}\sigma_{a,FT,1}\phi_1 - N_{FP-A}\left[(1-\xi_{FP-A,1})\sigma_{a,FP-A,1}\phi_1 + (1-\xi_{FP-A,2})\sigma_{a,FP-A,2}\phi_2\right] \quad (5.1f)$$

$$\frac{\partial N_I}{\partial t} = N_{FP-A}\left[\xi_{FP-A,1}\sigma_{a,FP-A,1}\phi_1 + \xi_{FP-A,2}\sigma_{a,FP-A,2}\phi_2\right] + N_{FP-S}\left[\sigma_{a,FP-S,1}\phi_1 + \sigma_{a,FP-S,2}\phi_2\right] . \quad (5.1g)$$

In Eq. (5.1a) and Eq. (5.1b) ϕ_g is the group g scalar flux $[n/cm^2/sec]$, $\Sigma_{t,g}$ is the total macroscopic cross section $[cm^{-1}]$ of group g, $\Sigma_{s,g\to g'}$ $[cm^{-1}]$ is the macroscopic cross section for neutrons scattering from group g to group g', k is the system multiplication factor, and $\nu\Sigma_{f,g}$ is the average number of neutrons released per fission (ν) for a fission induced by a neutron in group g, multiplied by the group g macroscopic fission cross section. The nuclide production destruction equations, Eqs. (5.1c) - (5.1g), use the following notation: $\gamma_{m,g}$ is the probability that a neutron absorption

in nuclide m results in the production of a fissile isotope, $y_{m,g}$ is the ratio of nuclide m's fission cross section to total cross section for group g, $p_{m,g}$ is the probability that the fission of nuclide m yields a parasitic absorber fission produce, and ξ_g is the probability that when a parasitic absorber fission production absorbs a neutron, another parasitic absorber fission product is produced. Though not explicitly noted in Eqs. (5.1), all scalar fluxes ϕ_g, macroscopic cross sections Σ_g, and nuclide densities N, are functions of position. We use Gauss-Legendre S_2 angular quadrature, with weights that sum to 2, to approximate the scalar fluxes.

Macroscopic cross sections are generated from nuclide density and microscopic cross section data. As an example, $\Sigma_{t,g}$ is calculated as shown in Eq. (5.2):

$$\Sigma_{t,g,i} = N_{FS}\sigma_{t,FS,g} + N_{FT}\sigma_{t,FT,g} + N_{FP-A}\sigma_{t,FP-A,g} + N_{FP-S}\sigma_{t,FP-S,g} + N_I\sigma_{t,I,g}\,. \quad (5.2)$$

Macroscopic fission cross section and average neutrons per fission products are found in a similar fashion, but we can limit our consideration to the fissile and fertile nuclide densities as shown in Eq. (5.3),

$$\nu\Sigma_{f,g,i} = N_{FS}\nu_{FS,g}\sigma_{f,FS,g} + N_{FT}\nu_{FT,g}\sigma_{f,FT,g}\,. \quad (5.3)$$

5.1.1 Microscopic Cross Section and Yield Data

We complete the specification of the depletion problem by giving the physical data used to solve the problem. Microscopic cross section data for the water is given in Table 5.3. Absorption and scattering cross sections for the fertile and fissile nuclides are given in Table 5.4, and fission cross sections and average neutrons per fission are given in Table 5.5. Radiative capture fractions and probability of an absorbed

neutron inducing fission are given in Table 5.6. Cross-section data for the fission products and inert nuclides are given in Table 5.7. Fission product yields and the parasitic absorber fission product regeneration fraction, ξ, are given in Table 5.8.

Table 5.3: Water microscopic cross section, in barns [10^{-24} cm^2]. Moderator is composed only of H_2O.

Nuclide	$\sigma_{a,1}$	$\sigma_{a,2}$	$\sigma_{s,1\to1}$	$\sigma_{s,2\to2}$	$\sigma_{s,1\to2}$
^{1}H	0	0.332	0	20.47	3.926
^{16}O	0	0	2.739	3.780	0

Table 5.4: Fuel microscopic cross sections, in barns [10^{-24} cm^2].

Nuclide	$\sigma_{a,1}$	$\sigma_{a,2}$	$\sigma_{s,1\to1}$	$\sigma_{s,2\to2}$	$\sigma_{s,1\to2}$
$^{235}_{92}$U	1.325	683.21	4.566	15.04	0
$^{238}_{92}$U	0.374	2.717	4.804	9.36	0

Table 5.5: Average neutron yield per fission, and fission microscopic cross section [10^{-24} cm^2].

Nuclide	ν_1	ν_2	$\sigma_{f,1}$	$\sigma_{f,2}$
$^{235}_{92}$U	2.6	2.4	1.235	584.4
$^{238}_{92}$U	2.8	N/A	0.308	0

Table 5.6: Radiative capture fraction, and fission probability for fissile and fertile nuclides.

Nuclide	γ_1	γ_2	y_1	y_2
$^{235}_{92}$U	$\frac{0.09}{1.325} = 0.068$	$\frac{98.81}{683.21} = 0.145$	$\frac{1.235}{1.325} = 0.932$	$\frac{584.4}{683.21} = 0.855$
$^{238}_{92}$U	$\frac{0.066}{0.374} = 0.177$	1	$\frac{0.308}{0.374} = 0.823$	0

Table 5.7: Parasitic absorber fission product, scattering fission product, and inert nuclide microscopic cross section data in barns $[10^{-24}\ cm^2]$.

Nuclide	$\sigma_{a,1}$	$\sigma_{a,2}$	$\sigma_{s,1}$	$\sigma_{s,2}$	$\sigma_{s,1\to 2}$
FP-A	15	1000	0.5	5	0
FP-S	0.5	5	15	100	0
Inert	1	5	1	5	0

Table 5.8: Fission product branch ratios and parasitic absorber fission product regeneration fraction.

$p_{FS,1}$	0.3
$p_{FS,2}$	0.3
$p_{FT,1}$	0.3
$p_{FT,2}$	0.3
$\xi_{FP-A,1}$	0.3
$\xi_{FP-A,2}$	0.5

5.1.2 Reactor Power Levels and Normalization

Vacuum boundary conditions are imposed on both sides of the slab. We normalize reactor scalar flux values so that the reactor produces a constant fission power level of 2000 $[W]$ for the duration of the burn-up cycle. The burn-up cycle length consists of 600 full-power days and we use a time step of 10 days to update the scalar fluxes. A typical beginning-of-cycle flux profile is shown in Fig. 5.3 and an end-of-cycle scalar flux profile is shown in Fig. 5.4.

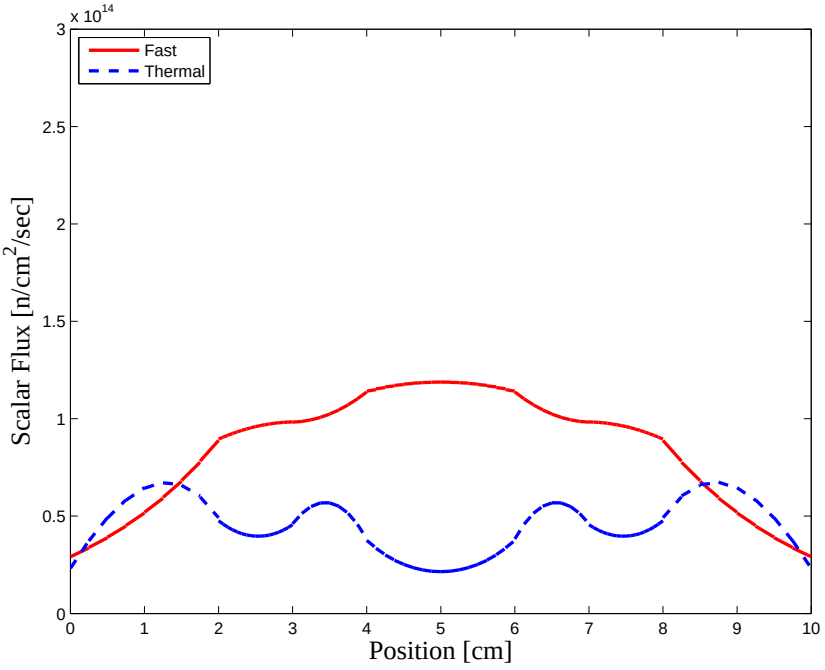

Figure 5.3: Example normalized scalar flux profiles at beginning and end of fuel burn-up cycle.

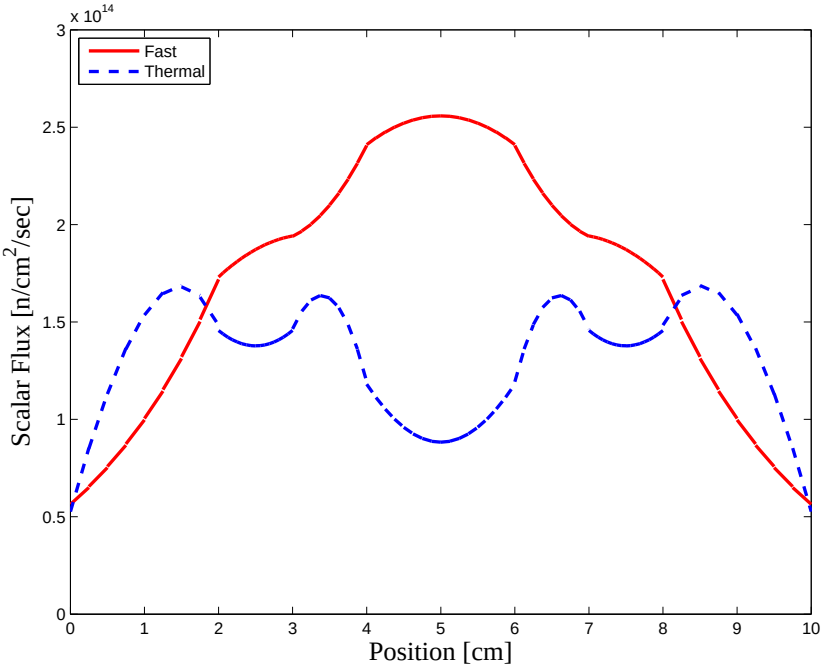

Figure 5.4: Example normalized scalar flux profiles at beginning and end of fuel burn-up cycle.

5.2 Spatial Discretization

We solve Eqs. (5.1) using a semi-static approach [33] assuming the flux distribution at the start of the time step remains constant throughout the time step. Nuclide densities are advanced in time using explicit Euler time differencing, then a corresponding radiation solution is found. We will consider three DFEM schemes to spatially discretize Eqs. (5.1).

1. AD DFEM: expands the angular flux in a P degree polynomial trial space using equally-spaced interpolation points, uses exact spatial integration, assumes cell-wise constant cross sections for solving the radiation equations, and tracks only cell average nuclide densities.

2. SL Collapse: expands both the angular flux and nuclide densities in a P degree

polynomial trial space, uses self-lumping quadrature to approximate integrals, and assumes cell-wise constant cross sections for solving the radiation equations. Lobatto quadrature is used as the DFEM interpolation points for odd degree trial spaces and Gauss quadrature as the DFEM interpolation points for even degree trial spaces.

3. SL Full: expands both the angular flux and nuclide densities in a P degree polynomial trial space, uses self-lumping quadrature to approximate integrals, and explicitly accounts for the variation of macroscopic cross section within each spatial cell. Lobatto quadrature is used as the DFEM interpolation points for odd degree trial spaces and Gauss quadrature as the DFEM interpolation points for even degree trial spaces.

5.2.1 Radiation Solution

Using the nomenclature developed in Chapter 2 and Chapter 3, the spatially discretized radiation equations are:

$$\mu_d \mathbf{G}\vec{\psi}_{d,1} - \mu_d \psi_{in,d,1}\vec{f} + \mathbf{R}_{\Sigma_{t,1}}\vec{\psi}_{d,1} = \frac{1}{4\pi}\mathbf{R}_{\Sigma_{s,1\to 1}}\vec{\phi}_1 + \frac{1}{k_\tau}\frac{1}{4\pi}\left(\mathbf{R}_{\nu\Sigma_{f,1}}\vec{\phi}_1 + \mathbf{R}_{\nu\Sigma_{f,2}}\vec{\phi}_2\right), \quad (5.4)$$

and

$$\mu_d \mathbf{G}\vec{\psi}_{d,2} - \mu_d \psi_{in,d,2}\vec{f} + \mathbf{R}_{\Sigma_{t,2}}\vec{\psi}_{d,2} = \frac{1}{4\pi}\left(\mathbf{R}_{\Sigma_{s,2\to 2}}\vec{\phi}_2 + \mathbf{R}_{\Sigma_{s,1\to 2}}\vec{\phi}_1\right). \quad (5.5)$$

In Eq. (5.4) and Eq. (5.5) k_τ is the multiplication factor at time index τ and $\mathbf{R}_{\Sigma_{s,1\to 1}}$, $\mathbf{R}_{\Sigma_{s,1\to 2}}$, $\mathbf{R}_{\Sigma_{s,2\to 2}}$, $\mathbf{R}_{\nu\Sigma_{f,1}}$, and $\mathbf{R}_{\nu\Sigma_{f,2}}$ are defined analogously to $\mathbf{R}_{\Sigma_t}$, as in Eq. (3.3), replacing $\Sigma_t(s)$ with $\Sigma_{s,1\to 1}(s)$, $\Sigma_{s,1\to 2}(s)$, $\Sigma_{s,2\to 2}(s)$, $\Sigma_{f,1}(s)$, and $\Sigma_{f,2}(s)$, respectively.

We solve Eq. (5.4) and Eq. (5.5) following the standard power iteration procedure, described for DFEM neutron transport in [43]. Convergence is checked after each power iteration. Using ℓ as the iteration index, convergence after the $\ell + 1$ iterate is said to occur when:

$$\delta_k = \left| \frac{k^{(\ell+1)} - k^{(\ell)}}{k^{(\ell)}} \right| < \epsilon_k \,, \tag{5.6}$$

and

$$\delta_\phi = \max_{g=1,2} \quad \max_{i=1,\ldots,N_{cell}} \quad \max_{j=1,\ldots,N_P} \quad \left| \frac{\phi_{g,i,j}^{(\ell+1)} - \phi_{g,i,j}^{(\ell)}}{\phi_{g,i,j}^{(\ell)}} \right| < \epsilon_\phi \,. \tag{5.7}$$

In our computational results we use $\epsilon_k = 10^{-12}$ and $\epsilon_\phi = 10^{-10}$. For each power iteration, the within group components of Eq. (5.4) and Eq. (5.5) are solved separately with a single transport sweep and S2SA iteration. Since we are using an S_2 angular quadrature, a single source iteration with a single S2SA step exactly solves a given within group neutron transport problem.

The converged scalar flux is normalized such that the desired power level of $P_{Total} = 2000\ [W/cm^2]$, is achieved. All fission energy is assumed to be deposited only in the fuel. For the SL Full scheme, we calculate the normalization factor, F_P, as:

$$F_P = E_f \sum_{g=1}^{2} \left[\sum_{i=1}^{N_{Fuel}} \frac{\Delta x_i}{2} \sum_{j=1}^{N_P} w_j \Sigma_{f,g,i,j} \phi_{g,i,j} \right], \tag{5.8}$$

where in E_f is the energy released per fission, assumed to be 200 $[MeV]$, w_j is the quadrature weight associated with the j-th DFEM interpolation point, and N_{Fuel} is the total number of spatial cells in the fuel region. The SL Collapse and AD DFEM schemes calculate F_P as

$$F_P = E_f \sum_{g=1}^{2} \left[\sum_{i=1}^{N_{Fuel}} \frac{\Delta x_i \widehat{\Sigma}_{f,g,i}}{2} \sum_{j=1}^{N_P} w_j \Sigma_{f,g,i,j} \phi_{g,i,j} \right]. \tag{5.9}$$

$\widetilde{\phi}^{(\ell+1)}$ is then scaled as:

$$\widetilde{\phi}_g^{(\ell+1)} \leftarrow \frac{P_{Total}}{F_P}\widetilde{\phi}_g^{(\ell+1)} \,. \tag{5.10}$$

5.2.2 Nuclide Density

We consider three spatial discretization schemes to solve the nuclide production/destruction components of Eqs. (5.1). The first, AD DFEM, tracks only cell average nuclide densities, approximating the true spatial distribution of nuclide m, $N_m(x,t)$, as being a constant in each cell, equal to the cell average density of nuclide m. Denoting the average nuclide density in cell i for nuclide m as $\bar{N}_{m,i}$ and the average group g scalar flux in cell i as $\bar{\phi}_{g,i}$, we give the fissile nuclide update equation for the AD DFEM scheme in Eq. (5.11):

$$\begin{aligned}\frac{\bar{N}_{FS,i}^{\tau+1} - \bar{N}_{FS,i}^{\tau}}{\Delta t} = \bar{N}_{FT,i}^{\tau}\left[\gamma_{FT,1}\sigma_{a,FT,1}\bar{\phi}_{1,i}^{\tau} + \gamma_{FT,2}\sigma_{a,FT,2}\bar{\phi}_{2,i}^{\tau}\right] \\ - \bar{N}_{FS,i}^{\tau}\left((1-\gamma_{FS,1})\sigma_{a,FS,1}\bar{\phi}_{1,i}^{\tau} + (1-\gamma_{FS,2})\sigma_{a,FS,2}\bar{\phi}_{2,i}^{\tau}\right) \,.\end{aligned} \tag{5.11}$$

In Eq. (5.11) superscript τ denotes time index τ quantities and Δt is the time step size. The update equations for N_{FT}, N_{FP-A}, N_{FP-S}, and N_I can be derived analogously to Eq. (5.11). AD DFEM calculates $\bar{\phi}_{g,i}$ using closed Newton-Cotes quadrature with the quadrature points limited to the DFEM interpolation points:

$$\bar{\phi}_{g,i} = \frac{1}{2}\sum_{j=1}^{N_P} w_j \phi_{g,j} \,. \tag{5.12}$$

The averaging in Eq. (5.12) is exact since an N_P point closed Newton-Cotes quadrature can exactly integrate any P degree polynomial. Equation (5.11) locally updates all average nuclide densities, $\bar{N}_{FS,i}$, $\bar{N}_{FT,i}$, $\bar{N}_{FP-A,i}$, $\bar{N}_{FP-S,i}$, and$\bar{N}_{I,i}$, simultaneously via a 5×5 matrix-vector multiply.

The SL Full and SL Collapse schemes approximate the true spatial density of nuclide m as a P degree Lagrange polynomial, $\widetilde{N}_m(s)$, in each cell:

$$\widetilde{N}_m(s) = \sum_{j=1}^{N_P} N_{m,j} b_j(s)\,, \tag{5.13}$$

where b_j are the $N_P = P + 1$ Lagrange interpolatory polynomials in the interval $s \in [-1, 1]$. We require the set of nuclide density DFEM interpolation points to be the same set of N_P points as the angular flux DFEM interpolation points. Following a Galerkin procedure, we multiply each production/destruction nuclide equation of Eqs. (5.1) by basis function b_j and integrate generating $5(P + 1)$ equations. The system of update equations for $\vec{N}_{FS,i}$ is shown in Eqs. (5.14):

$$\begin{aligned}\frac{1}{\Delta t}\mathbf{M}\left(\vec{N}^{\tau+1}_{FS,i} - \vec{N}^{\tau}_{FS,i}\right) = &-(1-\gamma_{FS,1})\sigma_{a,FS,1}\widehat{\mathbf{M}}_{\phi_{1,i},\tau}\vec{N}^{\tau}_{FS,i} \\ &- (1-\gamma_{FS,2})\sigma_{a,FS,2}\widehat{\mathbf{M}}_{\phi_{2,i},\tau}\vec{N}^{\tau}_{FS,i} \\ &+ \gamma_{FT,1}\sigma_{a,FT,1}\widehat{\mathbf{M}}_{\phi^{\tau}_{1,i}}\vec{N}^{\tau}_{FT,i} + \gamma_{FT,2}\sigma_{a,FT,2}\widehat{\mathbf{M}}_{\phi^{\tau}_{2,i}}\vec{N}^{\tau}_{FT,i}\,. \end{aligned} \tag{5.14}$$

The equations for $\vec{N}_{FT,i}$, $\vec{N}_{FP-A,i}$, $N_{FP-S,i}$, and $N_{I,i}$ are derived in a similar fashion to Eq. (5.14). In Eq. (5.14) we have defined:

$$\widehat{\mathbf{M}}_{\phi^{\tau}_{g,i},jk} = \frac{\Delta x}{2}\int_{-1}^{1} b_j(s) b_k(s) \widetilde{\phi}_{g,\tau,i}(s)\; ds \quad \text{, and} \tag{5.15}$$

$$\vec{N}^{\tau}_{m,i} = \begin{bmatrix} N^{\tau}_{m,1} \\ \vdots \\ N^{\tau}_{m,j} \\ \vdots \\ N^{\tau}_{m,P+1} \end{bmatrix}. \tag{5.16}$$

Given that we track five nuclides in the fuel region, each expanded in a P degree polynomial in each cell, there are $5(P+1)$ unknowns in each cell, thus Eq. (5.14) is a closed, $5(P+1) \times 5(P+1)$ system of linear equations for the $5(P+1)$ unknown nuclide densities, $N_{m,i,j}$ in each cell.

Using self-lumping quadrature to approximate Eq. (5.15) causes $\widehat{\mathbf{M}}_{\phi^{\tau}_{g,i}}$ to be a diagonal matrix. Recalling that $\widetilde{N}_m$ uses the same interpolation points as $\widetilde{\phi}_g$, approximating the integration of Eq. (5.15) with numerical quadrature restricted to the DFEM interpolating points, results in

$$\widehat{\mathbf{M}}_{\phi^{\tau}_{g,i},jk} = \begin{cases} w_j \frac{\Delta x}{2} \phi_{g,\tau,i,j} & j = k \\ 0 & \text{otherwise} \end{cases} . \tag{5.17}$$

Macroscopic cross sections are generated from nuclide density and microscopic cross section data. For the AD DFEM scheme, each cell has a single macroscopic cross section (per reaction type), and a single value of nuclide density for each nuclide type. Thus, interaction cross sections are easily tabulated. As an example, the cell average total interaction cross section in cell i is calculated in Eq. (5.18):

$$\begin{aligned} \widehat{\Sigma}_{t,g,i} = N_{FS,i}\sigma_{t,FS,g} + N_{FT,i}\sigma_{t,FT,g} + N_{FP-A,i}\sigma_{t,FP-A,g} \\ + N_{FP-S,i}\sigma_{t,FP-S,g} + N_{I,i}\sigma_{t,I,g} \,. \end{aligned} \tag{5.18}$$

Macroscopic fission cross section and average neutrons per fission products are found in a similar fashion, but we can limit our consideration to the fissile and fertile nuclide densities as shown in Eq. (5.19),

$$\widehat{\nu\Sigma}_{f,g,i} = N_{FS,i}\nu_{FS,g}\sigma_{f,FS,g} + N_{FT,i}\nu_{FT,g}\sigma_{f,FT,g} \,. \tag{5.19}$$

The SL Full and SL Collapse schemes calculate macroscopic cross sections in a similar fashion to Eq. (5.18) and Eq. (5.19), but instead of calculating a cell average, $\widehat{\Sigma}_{g,i}$, they calculate macroscopic values at each DFEM interpolation point. SL Collapse then averages the macroscopic cross section at each DFEM interpolation point to estimate the cell average cross section, as shown in Eq. (5.20) for $\widehat{\Sigma}_{t,i}$:

$$\widehat{\Sigma}_{t,i} = \frac{1}{2} \sum_{j=1}^{N_P} w_j \Sigma_{t,g,i,j} \, . \tag{5.20}$$

5.3 Numerical Results

Since an analytic solution to this depletion problem is not available we employ a fine spatial mesh to obtain the reference solution. We use a fine mesh of 10,240 cells and the SL Full scheme with a quartic polynomial trial space as our reference numerical solution. We present L_2 spatial error measures for

1. the total scalar flux (E_ϕ),
2. the fissile nuclide density ($E_{N_{FS}}$),
3. the fertile nuclide density ($E_{N_{FT}}$), and
4. the parasitic absorber fission product ($E_{N_{FP-A}}$).

To allow for easier comparison, we normalize each error to the reference solution quantity. We define E_ϕ as:

$$E_\phi = \frac{\sqrt{\sum_{g=1}^{2} \sum_{i=1}^{N_{ref}} \frac{\Delta x_i}{2} \sum_{q=1}^{N_{qf}} w_q \left(\widetilde{\phi}_{ref,i,g}(s_q) - \widetilde{\phi}_{num,i,g}(s_q) \right)^2}}{\sqrt{\sum_{g=1}^{2} \sum_{i=1}^{N_{ref}} \frac{\Delta x_i}{2} \sum_{q=1}^{N_{qf}} w_q \widetilde{\phi}_{ref,i,g}(s_q)^2}} \, , \tag{5.21}$$

where N_{ref} is the number of reference cells, $\widetilde{\phi}_{ref,i,g}(s)$ is the reference solution group g scalar flux in cell i, and $\widetilde{\phi}_{num,i,g}(s)$ is the coarse mesh numerical scheme's approximation of the group g scalar flux in cell i. Error measures for N_{FS}, N_{FT}, and N_{FP-A} are derived similarly.

Convergence of E_ϕ is shown in Figs. 5.5-5.8 as a function of DFEM trial space degree and DFEM scheme.

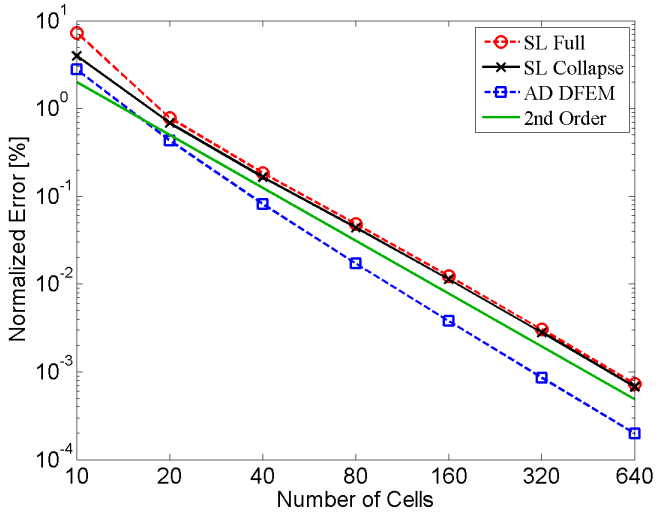

Figure 5.5: Normalized total scalar flux error for the depletion problem at end of cycle, for linear DFEM.

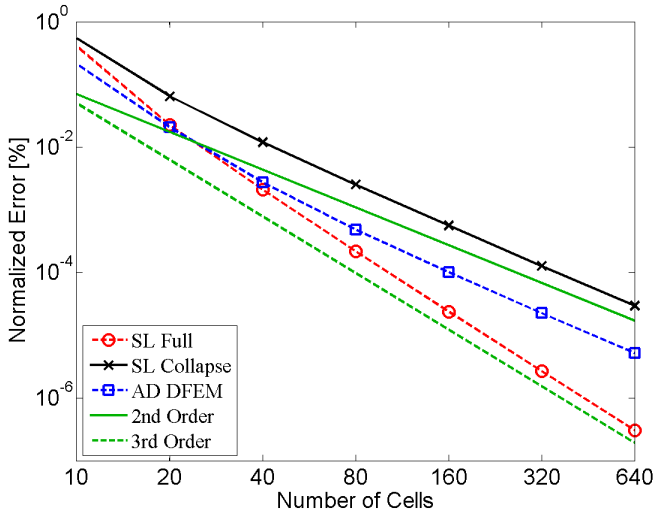

Figure 5.6: Normalized total scalar flux error for the depletion problem at end of cycle, for quadratic DFEM.

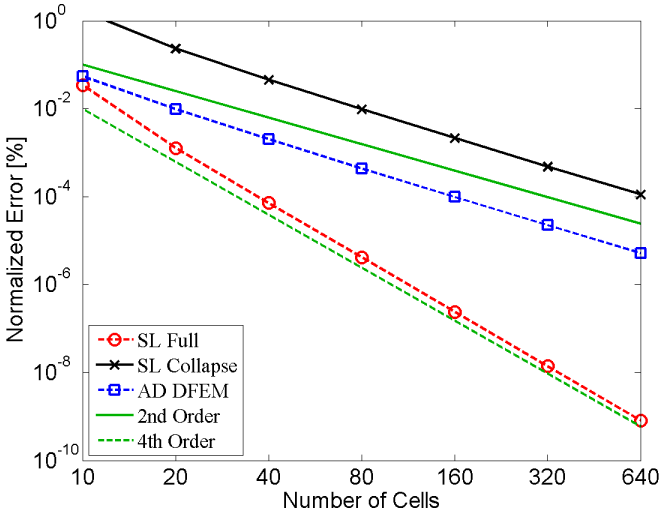

Figure 5.7: Normalized total scalar flux error for the depletion problem at end of cycle, for cubic DFEM.

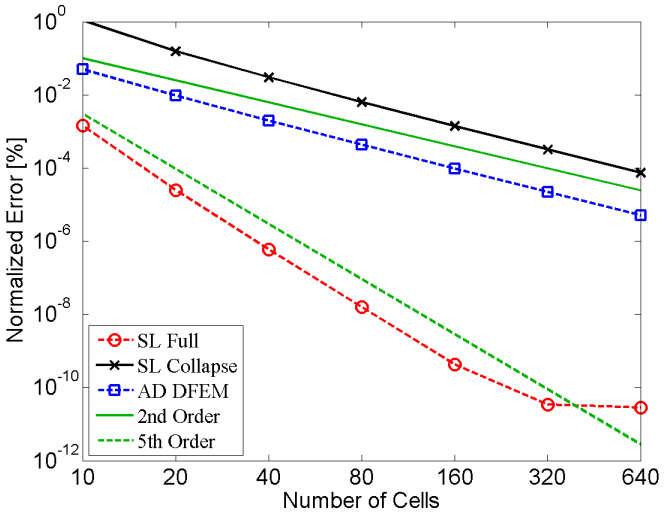

Figure 5.8: Normalized total scalar flux error for the depletion problem at end of cycle, for quartic DFEM.

Figures 5.5-5.8 re-emphasize two key results observed in the case of a pure absorber. First, when employing cell-wise constant cross sections, angular / scalar flux convergence is at most second order in space, regardless of the DFEM trial space polynomial degree. Second, exact integration of the interaction terms in the DFEM moment equations is not required to achieve high-order accuracy. In the depletion term, the DFEM interaction term is a degree $3P$ polynomial, and self-lumping schemes using Gauss or Lobatto quadrature only integrate $2P+1$ and $2P-1$ degree polynomials, respectively.

Convergence of $E_{N_{FS}}$, $E_{N_{FT}}$, and $E_{N_{FP-A}}$ are given in Figs. 5.9-5.12, Figs. 5.13-5.16, and Figs. 5.17-5.20, respectively for linear through quartic trial space degrees.

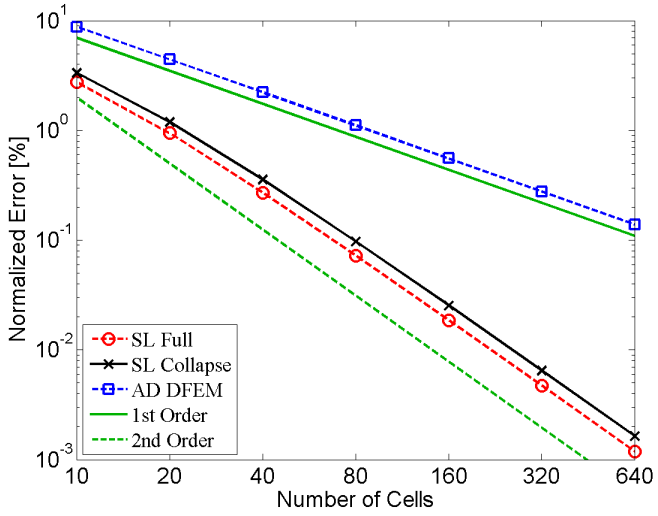

Figure 5.9: Normalized fissile nuclide density error for the depletion problem at end of cycle, for linear DFEM

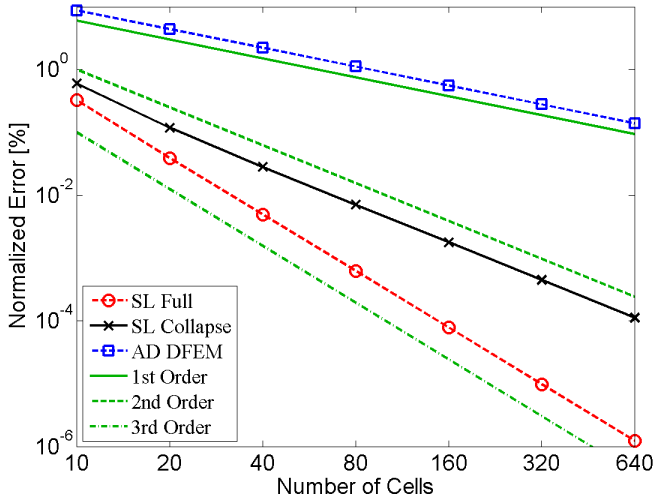

Figure 5.10: Normalized fissile nuclide density error for the depletion problem at end of cycle, for quadratic DFEM.

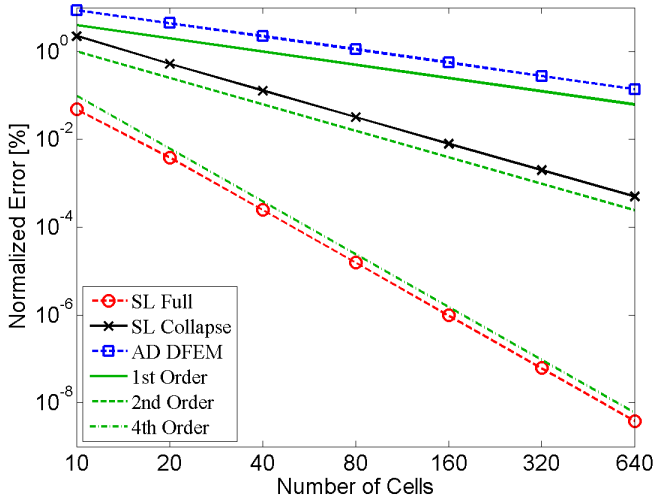

Figure 5.11: Normalized fissile nuclide density error for the depletion problem at end of cycle, for cubic DFEM.

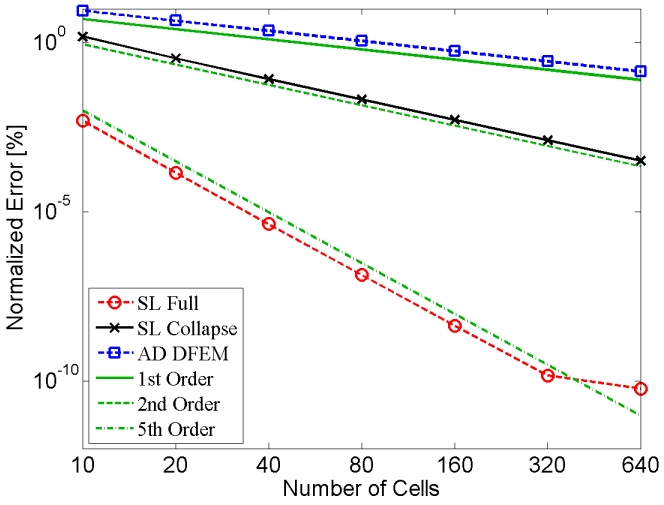

Figure 5.12: Normalized fissile nuclide density error for the depletion problem at end of cycle, for quartic DFEM.

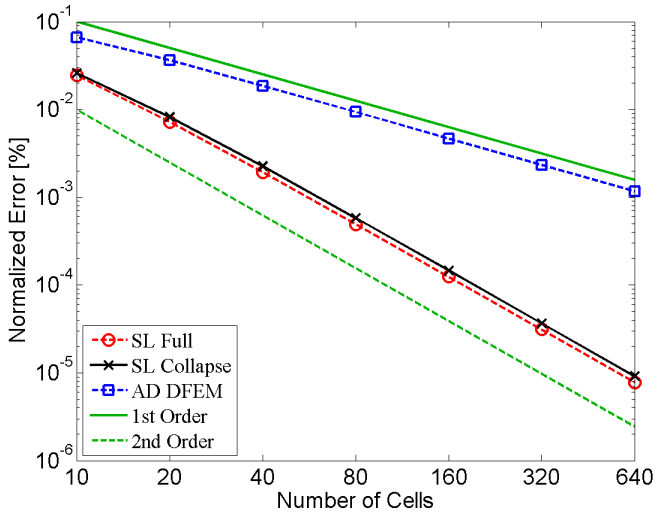

Figure 5.13: Normalized fertile nuclide density error for the depletion problem at end of cycle, for linear DFEM.

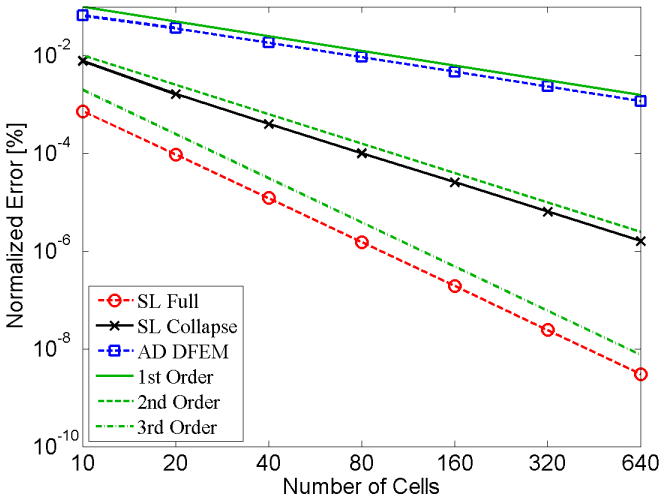

Figure 5.14: Normalized fertile nuclide density error for the depletion problem at end of cycle, for quadratic DFEM..

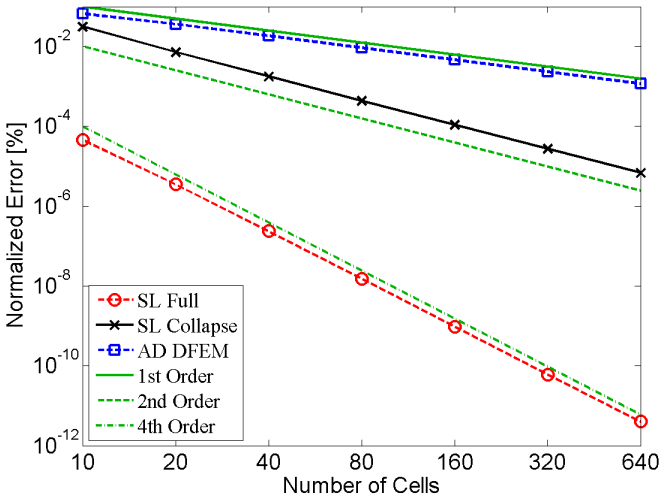

Figure 5.15: Normalized fertile nuclide density error for the depletion problem at end of cycle, for cubic DFEM.

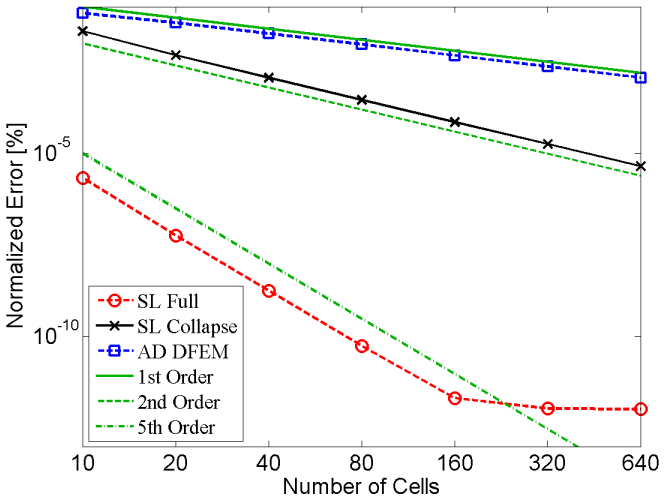

Figure 5.16: Normalized fertile nuclide density error for the depletion problem at end of cycle, for quartic DFEM.

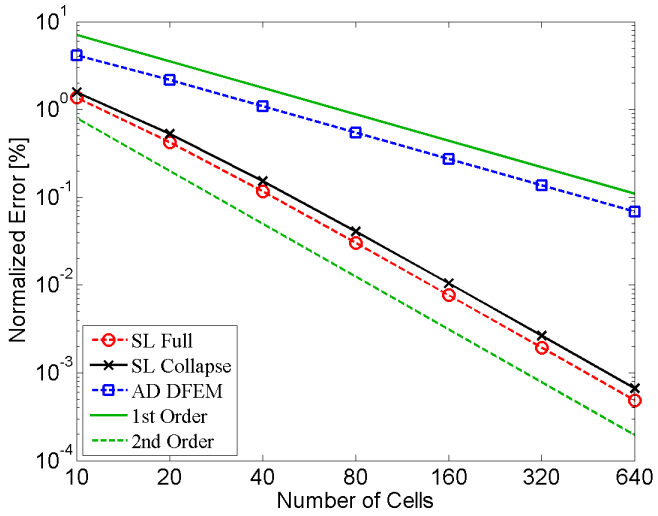

Figure 5.17: Normalized parasitic absorber fission product error for the depletion problem at end of cycle, for linear DFEM.

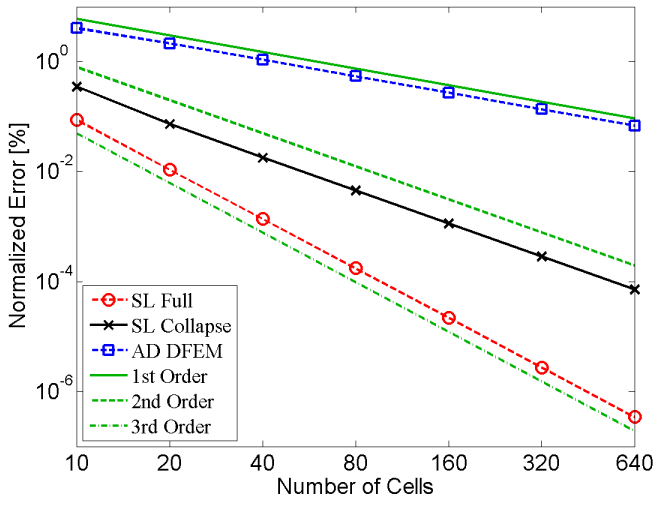

Figure 5.18: Normalized parasitic absorber fission product error for the depletion problem at end of cycle, for quadratic DFEM.

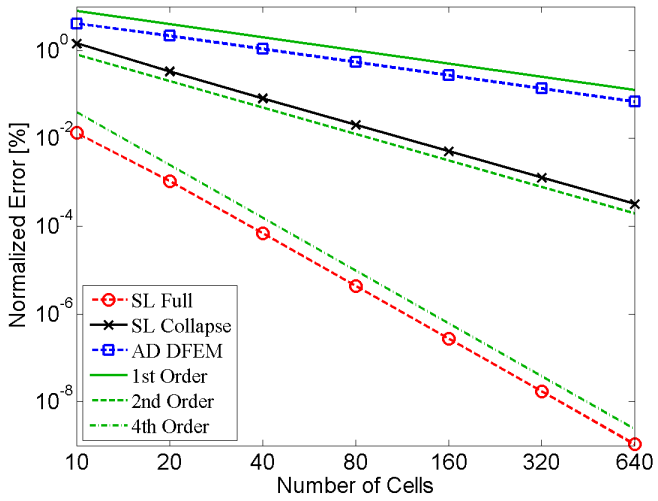

Figure 5.19: Normalized parasitic absorber fission product error for the depletion problem at end of cycle, for cubic DFEM.

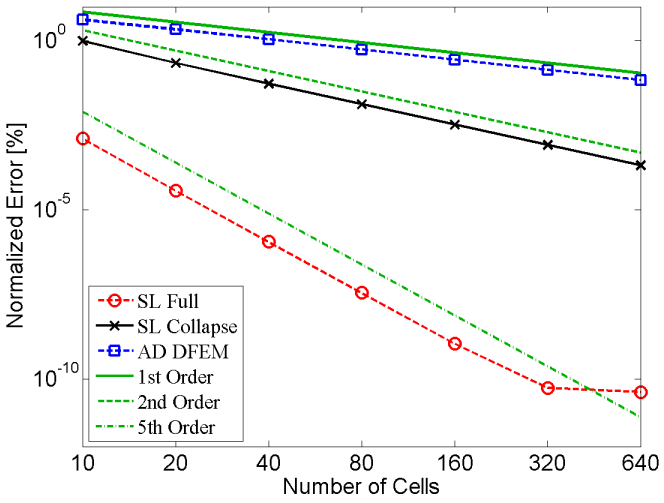

Figure 5.20: Normalized parasitic absorber fission product error for the depletion problem at end of cycle, for quartic DFEM.

Examining the spatial convergence in nuclide densities, we make several observations. First, we note that the AD DFEM scheme (cell-wise average cross section, cell average nuclide density) achieves at most first-order convergence for all spatial nuclide density errors, regardless of the angular flux trial space degree. The AD DFEM scheme is limited to at most first-order convergence of the error in the spatial distribution of nuclides because the scalar flux is updated using only a cell-wise average cross section and only the cell average nuclide density is tracked. Second, though the SL Collapse scheme expands nuclide density in a P degree polynomial DFEM trial space, it achieves at most second-order L^2 convergence of the error in nuclides spatial distribution, for all trial space polynomial degrees. SL Collapse is limited to at most second-order convergence of the spatial nuclide density solely because the scheme assumes a constant cross section in each cell when updating the scalar flux. The respective first-order and second-order convergence of the error in nuclide spatial distribution of the AD DFEM and SL Collapse scheme verifies the result observed in the pure absorber problem: assuming a cell-wise average cross section for coupled radiation transport problems limits the order of convergence of any quantity that depends on an interaction rate. SL Full achieves $P+1$ order convergence of the error in spatial nuclide density for the fuel depletion problem, showing that coupled systems of equations involving radiation transport can be solved with arbitrary order of accuracy using high-order DFEM polynomial trial spaces and self-lumping numerical quadrature that explicitly accounts for the spatial variation of cross section within each cell.

6. GREY THERMAL RADIATIVE TRANSFER

We now apply our self-lumping DFEM methodology to the grey thermal radiative transfer equations. Our framework shares important characteristics of the work presented by Morel, Wareing, and Smith [6]

1. linearization of the Planckian about an arbitrary temperature,
2. expansion of the angular intensity and temperature in a P degree trial space, and
3. expansion of the spatial dependence of the Planckian in a P degree trial space.

However, there are several differences between the work we present here and that of [6]. First, we derive our method for arbitrary DFEM polynomial trial space degree, not only a linear polynomial trial space. Second, [6] used a DFEM scheme equivalent to traditional lumping, whereas we primarily consider quadrature based self-lumping discretizations. However, the equations we derive are applicable to any DFEM scheme that uses polynomial trial space and test functions. Additionally, we consider arbitrary order (and stage count) SDIRK time integration, not only implicit Euler time integration. The two most important differences between our work here and that of [6] are:

1. we assume opacity and heat capacity can vary within each spatial cell and
2. we will fully converge the Planckian linearization in temperature.

As shown by Larsen, Kumar, and Morel, failure to fully converge the Planckian linearization in temperature can result in non-physical solutions that violate the maximum principle [44].

The remainder of this section is divided into three parts. Section 6.1 describes our linearization and discretization of the grey thermal radiative transfer (TRT) equations. We describe the time marching and nested iteration process we follow while solving the TRT in Section 6.2 In Section 6.3, we briefly describe the features and code implementation of the discretized TRT equations. Finally, we present numerical results that verify and demonstrate the capabilities of our TRT solver in Section 6.4.

6.1 Linearization and Discretization of Grey TRT Equations

We begin our discussion of how to linearize the Planckian of the grey thermal radiative transfer in temperature by first briefly outlining SDIRK temporal integration in Section 6.1.1. A more complete explanation can be found in [2]

6.1.1 SDIRK Time Integration

SDIRK time integration schemes are one of many options available for solving initial value problems of the form:

$$g(0) = G_0 \tag{6.1}$$

$$f(t,g) = \frac{\partial g}{\partial t}\,, \tag{6.2}$$

where t is time, and G_0 is the initial value of g at time $t = 0$. Depending on the literary source, SDIRK stands for Single-Diagonally Implicit Runge-Kutta, S-stable Diagonally Implicit Runge-Kutta, or one of many other expansions of the SDIRK acronym. The coefficients, a_i, b_i, and c_i that describe any Runge-Kutta time integration are typically given in formatted tables called Butcher tableaux. Due to the stiff nature of the TRT equations, we will limit ourselves to SDIRK time integration schemes. The Butcher tableaux of an SDIRK scheme with N_{stage} stages

is given in Eq. (6.3)

Stage	c_i	a			
1	c_1	a_{11}	0	$\dots$	0
2	c_2	a_{21}	a_{22}	0	$\vdots$
i	c_i	a_{i1}	a_{i2}	$\ddots$	0
N_{stage}	$c_{N_{stage}}$	$a_{N_{stage}1}$	$a_{N_{stage}2}$	$\dots$	$a_{N_{stage}N_{stage}}$
		b_1	b_2	$\dots$	$b_{N_{stage}}$

. (6.3)

To illustrate how SDIRK is used to advance time-dependent quantities, let us consider a time-dependent scalar function, $g(t)$. Given an initial value at time (or time step) t^n, $g(t^n) = g_n$, then $g(t^{n+1})$ is:

$$g_{n+1} = g_n + \Delta t \sum_{i=1}^{N_{stage}} b_i k_i \,, \tag{6.4}$$

where $\Delta t = t^{n+1} - t^n$, and k_i is defined as:

$$k_i = f\left(t_n + c_i \Delta t \;,\; g_n + \Delta t \sum_{j=1}^{i} a_{ij} k_j \right) . \tag{6.5}$$

Equation 6.5 can also be interpreted as meaning:

$$g_i = g_n + \Delta t \sum_{j=1}^{i} a_{ij} f\left(t_n + \Delta t c_j, g_j\right) , \tag{6.6}$$

where g_i is the intermediate value of g at the time of stage i, $t_i = t^n + \Delta t c_i$.

6.1.2 Spatially Analytic Linearization

We now linearize Planckian of the spatially analytic, 1-D slab, grey, discrete ordinates TRT equations with SDIRK time integration. The spatially and temporally

analytic grey discrete ordinates TRT are given in Eqs. (6.7),

$$\frac{1}{c}\frac{\partial I}{\partial t} + \mu_d \frac{\partial I}{\partial x} + \sigma_t I = \frac{1}{4\pi}\sigma_s \phi + \sigma_a B + S_I \tag{6.7a}$$

$$C_v \frac{\partial T}{\partial t} = \sigma_a \left(\phi - 4\pi B\right) + S_T \,. \tag{6.7b}$$

In Eqs. (6.7), we have assumed all scattering and material photon emission is isotropic, defined that I is the photon intensity with directional cosine μ_d (relative to the x-axis), S_I is a driving radiation source intensity source in the direction of μ_d, S_T is a driving temperature source, and the frequency integrated (grey) Planck, B, is:

$$B(T) = \frac{1}{4\pi} acT^4 \,, \tag{6.8}$$

where c is the speed of light and a is the Planck radiation constant. To use SDIRK to advance I and T in time, we must first define the time derivatives of I and T:

$$\frac{\partial I}{\partial t} = c\left[\frac{1}{4\pi}\sigma_s \phi + \sigma_a B + S_I - \mu_d \frac{\partial I}{\partial x} - \sigma_t I\right] \tag{6.9}$$

and

$$\frac{\partial T}{\partial t} = \frac{1}{C_v}\left[\sigma_a \left(\phi - 4\pi B\right) + S_T\right] \,. \tag{6.10}$$

We evaluate $k_{I,s}$ and $k_{T,s}$, the SDIRK k values for intensity and temperature for stage s as:

$$k_{I,s} = c\left[\frac{1}{4\pi}\sigma_s(T_s)\phi_s + \sigma_a(T_s)B(T_s) + S_I(t_s) - \mu_d \frac{\partial I_s}{\partial x} - \sigma_t(T_s) I_s\right] \tag{6.11}$$

and

$$k_{T,s} = \frac{1}{C_v(T_s)}\left[\sigma_a(T_s)\left(\phi_s - 4\pi B(T_s)\right) + S_T(t_s)\right] \,, \tag{6.12}$$

where ϕ_s, I_s, and T_s are the angle integrated intensity, angular intensity, and temperature at time t_s, and t_s is the time of stage s.

6.1.2.1 SDIRK Stage 1

With the definitions of Eq. (6.6), Eq. (6.11), and Eq. (6.12), we now seek to find I_1

$$I_1 = I_n + a_{11}\Delta t k_{I,1}\,. \tag{6.13}$$

Substituting in the definition of Eq. (6.11) into Eq. (6.13), for the intensity in stage 1, we have,

$$I_1 = I_n + a_{11}\Delta t c\left[\frac{1}{4\pi}\sigma_s\phi_1 + \sigma_a B + S_I - \mu_d\frac{\partial I_1}{\partial x} - \sigma_t I_1\right]\,. \tag{6.14}$$

Similarly, for T_1, we have

$$T_1 = T_n + \frac{a_{11}\Delta t}{C_v}\left[\sigma_a\left(\phi_1 - 4\pi B\right) + S_T\right]\,. \tag{6.15}$$

In Eq. (6.14) and Eq. (6.15), we have assumed that unless otherwise noted, all material properties and sources are evaluated at time t_s and temperature T_s.

We now introduce the linearization of the Planckian in temperature. For an arbitrary temperature iterate, T_*, we approximate $B(T_s)$ as:

$$B(T) \approx B(T_*) + \left.\frac{\partial B}{\partial T}\right|_{T=T_*}(T - T_*) \tag{6.16a}$$

$$B(T) \approx B_* + D_*\left(T - T_*\right) \tag{6.16b}$$

$$B_* = \frac{1}{4\pi}acT_*^4 \tag{6.16c}$$

$$D_* = \frac{1}{\pi}acT_*^3\,. \tag{6.16d}$$

Equation 6.14 has a strong non-linear dependence on T_1 due to the Planckian term, $\sigma_a B$, and a weak non-linear dependence on T_1 if the material opacities are temperature dependent. If we could remove the dependence of T_1 from Eq. (6.14), we could solve Eq. (6.14) using the same techniques that have been developed to solve the discrete ordinates neutron transport equation. We will attempt to remove the strong non-linear dependence on T_1 from Eq. (6.14) by linearizing the Planckian term. We neglect the non-linear dependence material opacities and heat capacities on temperature. To linearize the Planckian we first apply the linearization of Eq. (6.16) to Eq. (6.15) and manipulate. Inserting the linearization, we begin with Eq. (6.17)

$$T_1 = T_n + \frac{a_{11}\Delta t}{C_v}\left[\sigma_a\left(\phi_1 - 4\pi\left(B_* + D_*\left(T_1 - T_*\right)\right)\right) + S_T\right], \tag{6.17}$$

then move all T_1 terms to the left hand side,

$$\left(1 + \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right) T_1 = T_n + \frac{a_{11}\Delta t}{C_v}\left[\sigma_a\left(\phi_1 - 4\pi B_* + 4\pi D_* T_*\right) + S_T\right]. \tag{6.18}$$

In Eq. (6.17), we have made the assumption that all material properties are evaluated at T_*, but will neglect to denote this for streamlined notation. Next, we divide by the coefficient in front of T_1 on the left hand side:

$$\begin{aligned} T_1 = \left(1 + \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1} T_n + \ldots \\ \left(1 + \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1} \frac{a_{11}\Delta t}{C_v}\left[\sigma_a\left(\phi_1 - 4\pi B_*\right) + S_T\right] + \ldots \\ \left(1 + \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1} \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_* T_*, \end{aligned} \tag{6.19}$$

and then add "nothing",

$$\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}\left(T_*-T_*\right)\,, \tag{6.20}$$

to the right hand side,

$$\begin{aligned} T_1 = &\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1} T_n + \dots \\ &\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}\frac{a_{11}\Delta t}{C_v}\left[\sigma_a\left(\phi_1-4\pi B_*\right)+S_T\right]+\dots \\ &\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)T_*\dots \\ &-\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}T_*\,. \end{aligned} \tag{6.21}$$

Noting that

$$\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)=1\,, \tag{6.22}$$

and condensing, we finally have:

$$T_1 = T_* + \left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}\left(T_n - T_* + \frac{a_{11}\Delta t}{C_v}\left[\sigma_a\left(\phi_1-4\pi B_*\right)+S_T\right]\right)\,. \tag{6.23}$$

We occasionally refer to Eq. (6.23), and subsequent, similar equations, as a temperature update, as Eq. (6.23) is the non-linear iteration for temperature we must converge in order to solve the grey TRT.

We now linearize the Planck term of Eq. (6.14) in temperature,

$$I_1 = I_n + a_{11}\Delta t c\left[\frac{1}{4\pi}\sigma_s\phi_1+\sigma_a\left(B_*+D_*(T_1-T_*)\right)+S_I-\mu_d\frac{\partial I_1}{\partial x}-\sigma_t I_1\right]\,. \tag{6.24}$$

We then insert Eq. (6.23) into Eq. (6.24),

$$I_1 = I_n + a_{11}\Delta tc\left[\frac{1}{4\pi}\sigma_s\phi_1 + S_I - \mu_d\frac{\partial I_1}{\partial x} - \sigma_t I_1\right.$$
$$\left.+\sigma_a\left(B_* + D_*\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}\left(T_n - T_* + \frac{a_{11}\Delta t}{C_v}\left[\sigma_a\left(\phi_1 - 4\pi B_*\right) + S_T\right]\right)\right)\right]. \tag{6.25}$$

Next, we divide by $a_{11}\Delta tc$, and move the interaction and gradient terms to the left hand side:

$$\mu_d\frac{\partial I_1}{\partial x} + \sigma_t I_1 + \frac{1}{a_{11}\Delta tc}I_1 = \frac{1}{a_{11}\Delta tc}I_n + \frac{1}{4\pi}\sigma_s\phi_1 + S_I \ldots$$
$$+\sigma_a\left(B_* + D_*\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}\left(T_n - T_* + \frac{a_{11}\Delta t}{C_v}\left[\sigma_a\left(\phi_1 - 4\pi B_*\right) + S_T\right]\right)\right). \tag{6.26}$$

We now manipulate the right hand side,

$$\mu_d\frac{\partial I_1}{\partial x} + \sigma_t I_1 + \frac{1}{a_{11}\Delta tc}I_1 = \frac{1}{a_{11}\Delta tc}I_n + \frac{1}{4\pi}\sigma_s\phi_1 \ldots$$
$$+\sigma_a D_*\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}\frac{a_{11}\Delta t}{C_v}\sigma_a\phi_1 \ldots$$
$$+ S_I + \sigma_a B_* + \sigma_a D_*\left(1+\frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1}\left(T_n - T_* + \frac{a_{11}\Delta t}{C_v}\left[S_T - 4\pi\sigma_a B_*\right]\right), \tag{6.27}$$

and define

$$\sigma_{\tau,1} = \sigma_t + \frac{1}{a_{11}\Delta tc}, \tag{6.28}$$

giving

$$\mu_d \frac{\partial I_1}{\partial x} + \sigma_{\tau,1} I_1 = \frac{1}{a_{11}\Delta t c} I_n + \frac{1}{4\pi}\sigma_s \phi_1 \ldots + \sigma_a D_* \left(1 + \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1} \frac{a_{11}\Delta t}{C_v}\sigma_a \phi_1 \ldots + S_I + \sigma_a B_* + \sigma_a D_* \left(1 + \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1} \left(T_n - T_* + \frac{a_{11}\Delta t}{C_v}\left[S_T - 4\pi\sigma_a B_*\right]\right) . \tag{6.29}$$

Focusing on the second ϕ_1 term on the right hand side,

$$\sigma_a D_* \left(1 + \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1} \frac{a_{11}\Delta t}{C_v}\sigma_a \phi_1 \,, \tag{6.30}$$

we first simplify the terms in front of $\sigma_a \phi_1$

$$\sigma_a D_* \left(1 + \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1} \frac{a_{11}\Delta t}{C_v} = \frac{\sigma_a D_* a_{11}\Delta t}{C_v + 4\pi a_{11}\Delta t \sigma_a D_*} \,, \tag{6.31}$$

then multiply by $\frac{4\pi}{4\pi}$ and arrange, yielding:

$$\frac{1}{4\pi}\frac{\sigma_a D_* a_{11}\Delta t}{C_v + 4\pi a_{11}\Delta t \sigma_a D_*}\sigma_a \phi_1 \,. \tag{6.32}$$

Defining a constant, ν_1,

$$\nu_1 = \left(4\pi a_{11}\Delta t \sigma_a D_*\right)\left(C_v + 4\pi a_{11}\Delta t \sigma_a D_*\right)^{-1} \,, \tag{6.33}$$

we arrive at an equation for intensity I_1 that is similar in form to the discrete ordinates neutron transport equations with isotropic scattering, fission, and a fixed

source:

$$\mu_d \frac{\partial I_1}{\partial x} + \sigma_{\tau,1} I_1 = \frac{1}{4\pi}\sigma_s \phi_1 + \frac{1}{4\pi}\nu_1 \sigma_a \phi_1 + \xi_1 \,, \tag{6.34}$$

where

$$\begin{aligned} \xi_1 = & \frac{1}{a_{11}\Delta t c} I_n + S_I + \sigma_a B_* + \ldots \\ & \sigma_a D_* \left(1 + \frac{4\pi a_{11}\Delta t}{C_v}\sigma_a D_*\right)^{-1} \left(T_n - T_* + \frac{a_{11}\Delta t}{C_v}\left[S_T - 4\pi\sigma_a B_*\right]\right) . \end{aligned} \tag{6.35}$$

The similarity of Eq. (6.34) to the neutron transport equation is especially apparent if we define pseudo total interaction and scattering cross sections, $\widetilde{\Sigma_t}$ and $\widetilde{\Sigma_s}$, to be

$$\widetilde{\Sigma_t} = \sigma_{\tau,1} \tag{6.36}$$

$$\widetilde{\Sigma_s} = \sigma_s + \nu_1 \sigma_a \,, \tag{6.37}$$

giving:

$$\mu_d \frac{\partial I_1}{\partial x} + \widetilde{\Sigma_t} I_1 = \frac{1}{4\pi}\widetilde{\Sigma_s}\phi_1 + \xi_1 \,. \tag{6.38}$$

If we define a pseudo absorption interaction cross section as,

$$\widetilde{\Sigma_a} = \widetilde{\Sigma_t} - \widetilde{\Sigma_s} \,, \tag{6.39}$$

then it appears that all of the methodology, including DSA, developed for discrete ordinates neutron transport might be applicable for solving the discrete ordinates thermal radiative transfer equations using SDIRK time integration. The next step in examining the possibility of applying neutron transport methods to solve the TRT equations is to attempt to form a pseudo neutron transport equation for stage i of any SDIRK time integration scheme.

6.1.2.2 SDIRK Stage i

The intensity at stage i, is given by:

$$I_i = I_n + \Delta t \sum_{j=1}^{i-1} a_{ij} k_{I,j} + a_{ii} \Delta t c \left[\frac{1}{4\pi} \sigma_s \phi_i + \sigma_a \left[B_* + D_* (T_i - T_*) \right] + S_I - \mu_d \frac{\partial I_i}{\partial x} - \sigma_t I_i \right] . \tag{6.40}$$

Likewise for T_i, we have

$$T_i = T_n + \Delta t \sum_{j=1}^{i-1} a_{ij} k_{T,j} + \frac{a_{ii} \Delta t}{C_v} \left[\sigma_a \left(\phi_i - 4\pi \left[B_* + D_* (T_i - T_*) \right] \right) + S_T \right] . \tag{6.41}$$

The only difference between Eq. (6.40) and Eq. (6.24) is the presence of

$$\Delta t \sum_{j=1}^{i-1} a_{ij} k_{I,j} . \tag{6.42}$$

Likewise, the only difference between Eq. (6.41) and Eq. (6.17) is

$$\Delta t \sum_{j=1}^{i-1} a_{ij} k_{T,j} . \tag{6.43}$$

For brevity, we omit the manipulation of Eq. (6.41) to form a temperature update equation, and instead give the final result:

$$T_i = T_* + \left(1 + \frac{4\pi a_{ii} \Delta t}{C_v} \sigma_a D_* \right)^{-1} \left(T_n - T_* + \Delta t \sum_{j=1}^{i-1} a_{ij} k_{T,j} + \frac{a_{ii} \Delta t}{C_v} \left[\sigma_a \left(\phi_i - 4\pi B_* \right) + S_T \right] \right) . \tag{6.44}$$

Similarly, we omit the manipulation of Eq. (6.40) into a pseudo-fission form, and

instead give the final result,

$$\mu_d \frac{\partial I_i}{\partial x} + \sigma_{\tau,i} = \frac{1}{4\pi}\sigma_s \phi_i + \frac{1}{4\pi}\nu_i \sigma_a \phi_i + \xi_i \,, \tag{6.45}$$

where we have defined the following:

$$\nu_i = 4\pi\sigma_a D_* \left(1 + \frac{4\pi a_{ii}\Delta t}{C_v}\sigma_a D_*\right)^{-1} \frac{a_{ii}\Delta t}{C_v} = \frac{4\pi a_{ii}\Delta t \sigma_a D_*}{C_v + 4\pi a_{ii}\Delta t \sigma_a D_*} \,, \tag{6.46a}$$

$$\sigma_{\tau,i} = \frac{1}{a_{ii}\Delta t c} + \sigma_t \ \text{,and} \tag{6.46b}$$

$$\begin{aligned} \xi_i = {} & \sigma_a B_* + S_I + \frac{1}{a_{ii}\Delta t c} I_n + \frac{1}{a_{ii} c} \sum_{j=1}^{i-1} a_{ij} k_{I,j} + \ldots \\ & \sigma_a D_* \left(1 + \frac{4\pi a_{ii}\Delta t}{C_v}\sigma_a D_*\right)^{-1} \left(T_n - T_* + \Delta t \sum_{j=1}^{i-1} a_{ij} k_{T,j} + \frac{a_{ii}\Delta t}{C_v}\left(S_T - 4\pi\sigma_a B_*\right)\right) . \end{aligned} \tag{6.46c}$$

In the spatially analytic case, the grey TRT equations, with Planck linearization and arbitrary stage count SDIRK time integration can be put into a pseudo neutron transport form. This suggests that we may techniques such as spatial discretization and acceleration methods, can be used to solve the grey TRT equations. To verify this, we go through the linearization procedure again with the spatially discretized grey TRT equations that explicitly account for the spatial variation of material properties in the next section.

If we could solve the spatially discretized equations efficiently through source iteration alone, it would be redundant to go through the entire linearization process with the spatially analytic and spatially discretized TRT equations. However, iterative acceleration is essential for efficient solution of the TRT equations due to the Planckian

absorption/re-emission terms in the linearized radiation intensity equation creating a situation analogous to a scattering dominated medium in neutron transport. As we will see, the problem with accelerating the spatially discretized TRT equations is that we will need a pseudo diffusion coefficient, $\widetilde{D}$, both at quadrature integration points and cell edges for MIP DSA acceleration, but do not have a physical definition of that quantity.

6.1.3 Spatially Discretized Linearization

Now we attempt to derive a pseudo fission form of the spatially discretized grey TRT equations. First, we must define a spatially discretized k_I and k_T. To do this, we apply the standard Galerkin procedure outlined in Chapter 2 and Chapter 3 to the spatially analytic forms of k_I and k_T given in Eq. (6.11) and Eq. (6.12), respectively. For $\vec{k}_I$, we have:

$$\vec{k}_I = c\mathbf{M}^{-1}\left[\frac{1}{4\pi}\mathbf{R}_{\sigma_s}\vec{\phi} + \mathbf{R}_{\sigma_a}\vec{B} - \mathbf{R}_{\sigma_t}\vec{I} - \mu_d\mathbf{G}\vec{I} + \mu_d I_{in}\vec{f} + \vec{S}_I\right], \tag{6.47}$$

where we have approximated the true angular intensity, angle integrated intensities, and temperatures in every spatial cell as P degree polynomials:

$$\begin{aligned}
I(s) &\approx \widetilde{I}(s) && (6.48)\\
\widetilde{I}(s) &= \sum_{i=1}^{N_P} I_i b_i(s) && (6.49)\\
\phi(s) &\approx \widetilde{\phi(s)} && (6.50)\\
\widetilde{\phi}(s) &= \sum_{i=1}^{N_P} \phi_i b_i(s) && (6.51)\\
T(s) &\approx \widetilde{T}(s) && (6.52)\\
\widetilde{I}(s) &= \sum_{i=1}^{N_P} T_i b_i(s)\,, && (6.53)
\end{aligned}$$

with

$$\vec{I} = [I_1 \dots I_{N_P}]^T \tag{6.54}$$

$$\vec{\phi} = [\phi_1 \dots \phi_{N_P}]^T \tag{6.55}$$

$$\vec{T} = [T_1 \dots T_{N_P}]^T \tag{6.56}$$

$$\vec{B} = \frac{1}{4\pi} [B(T_1) \dots B(T_{N_P})]^T \,, \tag{6.57}$$

and

$$\vec{S}_{I,j} = \frac{\Delta x}{2} \int_{-1}^{1} b_j(s) S_I(s) \; ds \,. \tag{6.58}$$

Additionally, $\mathbf{R}_{\sigma_t}$, $\mathbf{R}_{\sigma_a}$, and $\mathbf{R}_{\sigma_s}$, are defined analogously to the $\mathbf{R}_\Sigma$ defined for neutron transport, as are $\mathbf{G}$, $\mathbf{M}$, and $\vec{f}$. Defining $\vec{k}_T$:

$$\vec{k}_T = \mathbf{R}_{C_v}^{-1} \left[\mathbf{R}_{\sigma_a} \left(\vec{\phi} - 4\pi \vec{B} \right) + \vec{S}_T \right] \,, \tag{6.59}$$

with

$$\vec{S}_{T,j} = \frac{\Delta x}{2} \int_{-1}^{1} b_j(s) S_T(s) \; ds \,, \tag{6.60}$$

and

$$\mathbf{R}_{C_v,ij} = \frac{\Delta x}{2} \int_{-1}^{1} C_v(s) b_i(s) b_j(s) \; ds \,. \tag{6.61}$$

Before proceeding, it is critical to note that in Eq. (6.47) and Eq. (6.59) we approximate the "true" spatial dependence of the Planckian,

$$B(s) = B(\widetilde{T}(s)) \,, \tag{6.62}$$

as

$$B(s) \approx \sum_{i=1}^{N_P} b_i(s) B(T_i) \,. \tag{6.63}$$

Additionally, since the we assume the Planckian is a P degree polynomial in space, it follows that:

$$B(\widetilde{T}(s)) \approx \sum_{i=1}^{N_P} b_i(s) \left[B(T_{i,*}) + (T_i - T_{i,*}) \frac{dB}{dT}\bigg|_{T=T_{i,*}} \right], \tag{6.64}$$

where $T_{i,*}$ is an arbitrary temperature at DFEM interpolation point i.

6.1.3.1 SDIRK Stage i

As we showed with the spatially analytic case, the i-th SDIRK stage is only slightly different than the first SDRIK stage. Thus, we will omit the first stage derivation for the spatially discretized case, and begin with SDIRK stage i, first writing the equation for $\vec{I}_i$ and $\vec{T}_i$:

$$\vec{I}_i = \vec{I}_n + \Delta t \sum_{j=1}^{i-1} a_{ij} k_{I,j} +$$
$$\Delta t a_{ii} c\mathbf{M}^{-1} \left[\frac{1}{4\pi} \mathbf{R}_{\sigma_s} \vec{\phi}_i + \mathbf{R}_{\sigma_a} \left(\vec{B}_* + \mathbf{D}_* \left(\vec{T}_i - \vec{T}_* \right) \right) - \mathbf{R}_{\sigma_t} \vec{I}_i - \mu_d \mathbf{G} \vec{I}_i + \mu_d I_{in,i} \vec{f} + \vec{S}_I \right] \tag{6.65}$$

$$\vec{T}_i = \vec{T}_n + \Delta t \sum_{j=1}^{i-1} a_{ij} k_{T,j} + \Delta t a_{ii} \mathbf{R}_{C_v}^{-1} \left[\mathbf{R}_{\sigma_a} \left(\vec{\phi}_i - 4\pi \vec{B}_* - 4\pi \mathbf{D}_* \left(\vec{T}_i - \vec{T}_* \right) \right) + \vec{S}_T \right]. \tag{6.66}$$

We now solve Eq. (6.66) for $\vec{T}_i$, by first moving all $\vec{T}_i$ terms to the left hand side:

$$\vec{T}_i + 4\pi \Delta t a_{ii} \mathbf{R}_{C_v}^{-1} \mathbf{R}_{\sigma_a} \mathbf{D}_* \vec{T}_i =$$
$$\vec{T}_n + \Delta t \sum_{j=1}^{i-1} a_{ij} k_{T,j} + \Delta t a_{ii} \mathbf{R}_{C_v}^{-1} \left[\mathbf{R}_{\sigma_a} \left(\vec{\phi}_i - 4\pi \vec{B}_* + 4\pi \mathbf{D}_* \vec{T}_* \right) + \vec{S}_T \right], \tag{6.67}$$

and consolidate the $\vec{T}_i$ coefficient matrices,

$$\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]\vec{T}_i = \vec{T}_n + \Delta t\sum_{j=1}^{i-1} a_{ij}k_{T,j} + \Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\left[\mathbf{R}_{\sigma_a}\left(\vec{\phi}_i - 4\pi\vec{B}_* + 4\pi\mathbf{D}_*\vec{T}_*\right) + \vec{S}_T\right] . \quad (6.68)$$

Multiplying the inverse of the coefficient matrix,

$$\vec{T}_i = \left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left[\vec{T}_n + \Delta t\sum_{j=1}^{i-1} a_{ij}k_{T,j}\right]\ldots + \Delta t a_{ii}\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1}\left[\mathbf{R}_{\sigma_a}\left(\vec{\phi}_i - 4\pi\vec{B}_* + 4\pi\mathbf{D}_*\vec{T}_*\right) + \vec{S}_T\right] , \quad (6.69)$$

Isolating the $\vec{T}_*$ term on the right hand side,

$$\vec{T}_i = \left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left[\vec{T}_n + \Delta t\sum_{j=1}^{i-1} a_{ij}k_{T,j}\right]\ldots + \Delta t a_{ii}\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1}\left[\mathbf{R}_{\sigma_a}\left(\vec{\phi}_i - 4\pi\vec{B}_*\right) + \vec{S}_T\right]\ldots + 4\pi\Delta t a_{ii}\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\vec{T}_* , \quad (6.70)$$

adding nothing,

$$\vec{T}_i = \left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left[\vec{T}_n + \Delta t\sum_{j=1}^{i-1} a_{ij}k_{T,j}\right]\ldots + \Delta t a_{ii}\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1}\left[\mathbf{R}_{\sigma_a}\left(\vec{\phi}_i - 4\pi\vec{B}_*\right) + \vec{S}_T\right]\ldots + \left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left[4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\vec{T}_*\right]\ldots + \left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left(\vec{T}_* - \vec{T}_*\right) , \quad (6.71)$$

and noting that

$$\left[4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\vec{T}^*\right]^{-1}\left[4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\vec{T}^*\right]=\mathbf{I}\,, \tag{6.72}$$

we reach the stage i temperature update equation,

$$\begin{aligned}\vec{T}_i=\vec{T}_*+\left[\mathbf{I}+4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left[\vec{T}_n-\vec{T}_*+\Delta t\sum_{j=1}^{i-1}a_{ij}k_{T,j}\right]\ldots\\ +\Delta t a_{ii}\left[\mathbf{I}+4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1}\left[\mathbf{R}_{\sigma_a}\left(\vec{\phi}_i-4\pi\vec{B}_*\right)+\vec{S}_T\right]\,.\end{aligned} \tag{6.73}$$

Multiplying Eq. (6.65) by $\frac{1}{c\Delta t a_{ii}}\mathbf{M}$:

$$\begin{aligned}\frac{1}{c\Delta t a_{ii}}\mathbf{M}\vec{I}_i=\frac{1}{c\Delta t a_{ii}}\mathbf{M}\vec{I}_n+\frac{1}{ca_{ii}}\mathbf{M}\sum_{j=1}^{i-1}a_{ij}k_{I,j}\ldots\\ +\frac{1}{4\pi}\mathbf{R}_{\sigma_s}\vec{\phi}_i+\mathbf{R}_{\sigma_a}\left(\vec{B}_*+\mathbf{D}_*\left(\vec{T}_i-\vec{T}_*\right)\right)-\mathbf{R}_{\sigma_t}\vec{I}_i-\mu_d\mathbf{G}\vec{I}_i+\mu_d I_{in,i}\vec{f}+\vec{S}_I\,,\end{aligned} \tag{6.74}$$

moving the gradient terms and $\frac{1}{a_{ii}c\Delta}\mathbf{M}\vec{I}_i$ to the left hand side and inserting the result of Eq. (6.73) into Eq. (6.74), we have:

$$\begin{aligned}\mu_d\mathbf{G}\vec{I}_i+\left(\frac{1}{c\Delta t a_{ii}}\mathbf{M}+\mathbf{R}_{\sigma_t}\right)\vec{I}_i=\frac{1}{c\Delta t a_{ii}}\mathbf{M}\vec{I}_n+\frac{1}{ca_{ii}}\mathbf{M}\sum_{j=1}^{i-1}a_{ij}k_{I,j}+\frac{1}{4\pi}\mathbf{R}_{\sigma_s}\vec{\phi}_i+\mathbf{R}_{\sigma_a}\vec{B}_*\ldots\\ +\mathbf{R}_{\sigma_a}\mathbf{D}_*\left\{\left[\mathbf{I}+4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left[\vec{T}_n-\vec{T}_*+\Delta t\sum_{j=1}^{i-1}a_{ij}k_{T,j}\right]\ldots\right.\\ \left.+\Delta t a_{ii}\left[\mathbf{I}+4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1}\left[\mathbf{R}_{\sigma_a}\left(\vec{\phi}_i-4\pi\vec{B}_*\right)+\vec{S}_T\right]\right\}+\mu_d I_{in,i}\vec{f}+\vec{S}_I\,.\end{aligned} \tag{6.75}$$

Re-arranging Eq. (6.75) to isolate $\vec{\phi}_i$ terms on the right hand side, we have:

$$\begin{aligned}\mu_d \mathbf{G}\vec{I}_i + \left(\frac{1}{c\Delta t a_{ii}}\mathbf{M} + \mathbf{R}_{\sigma_t}\right)\vec{I}_i = \frac{1}{4\pi}\mathbf{R}_{\sigma_s}\vec{\phi}_i \ldots \\ + \Delta t a_{ii}\mathbf{R}_{\sigma_a}\mathbf{D}_*\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\vec{\phi}_i \ldots \\ + \mu_d\vec{f}I_{in,i} + \vec{S}_I + \frac{1}{c\Delta t a_{ii}}\mathbf{M}\vec{I}_n + \frac{1}{ca_{ii}}\mathbf{M}\sum_{j=1}^{i-1}a_{ij}k_{I,j} + \mathbf{R}_{\sigma_a}\vec{B}_* \ldots \\ + \mathbf{R}_{\sigma_a}\mathbf{D}_*\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left[\vec{T}_n - \vec{T}_* + \Delta t\sum_{j=1}^{i-1}a_{ij}k_{T,j}\right] \ldots \\ + \Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left[\vec{S}_T - 4\pi\mathbf{R}_{\sigma_a}\vec{B}_*\right]. \quad (6.76)\end{aligned}$$

Making the following definitions:

$$\begin{aligned}\bar{\bar{\xi}}_{d,i} = \frac{1}{c\Delta t a_{ii}}\mathbf{M}\vec{I}_n + \frac{1}{ca_{ii}}\mathbf{M}\sum_{j=1}^{i-1}a_{ij}k_{I,j} + \mathbf{R}_{\sigma_a}\vec{B}_* + \vec{S}_I \ldots \\ + \mathbf{R}_{\sigma_a}\mathbf{D}_*\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left\{\vec{T}_n - \vec{T}_* + \Delta t\sum_{j=1}^{i-1}a_{ij}k_{T,j}\right. \\ \left. + \Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\left[\vec{S}_T - 4\pi\mathbf{R}_{\sigma_a}\vec{B}_*\right]\right\} \quad (6.77a)\end{aligned}$$

$$\bar{\bar{\nu}}_i = 4\pi\Delta t a_{ii}\mathbf{R}_{\sigma_a}\mathbf{D}_*\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1} \quad (6.77b)$$

$$\bar{\bar{\mathbf{R}}}_{\sigma_\tau,i} = \mathbf{R}_{\sigma_t} + \frac{1}{c\Delta t a_{ii}}\mathbf{M}, \quad (6.77c)$$

and inserting into Eq. (6.77a) gives our final equation for the radiation intensity:

$$\mu_d\mathbf{G}\vec{I}_i + \bar{\bar{\mathbf{R}}}_{\sigma_\tau,i}\vec{I}_i = \frac{1}{4\pi}\mathbf{R}_{\sigma_s}\vec{\phi}_i + \frac{1}{4\pi}\bar{\bar{\nu}}_i\mathbf{R}_{\sigma_a}\vec{\phi}_i + \bar{\bar{\xi}}_{d,i} + \mu_d\vec{f}I_{in,i}. \quad (6.78)$$

Given the form of Eq. (6.78), it appears that we can apply MIP DSA to accelerate the iterative solution of Eq. (6.78), but we still wish to verify that definition of $\widetilde{\Sigma}_t$ and $\widetilde{\Sigma}_a$ is consistent between both the spatially discretized and spatially analytic cases.

6.1.4 Consistency of Pseudo Cross Sections

In Chapter 4, we used MIP DSA to accelerate a spatially analytic problem of the form

$$\mu_d \frac{d\psi}{dx} + \Sigma_t \psi = \frac{1}{4\pi} \Sigma_s \phi + Q_d \,, \tag{6.79}$$

with spatially discretized analog

$$\mu_d \frac{d\psi}{dx} \mathbf{R}_{\Sigma_t} \psi = \frac{1}{4\pi} \mathbf{R}_{\Sigma_s} \phi + \vec{Q}_d \,. \tag{6.80}$$

In the neutron transport problem, Σ_t, Σ_s, and Σ_a are physically meaningful quantities, the total macroscopic interaction cross section, macroscopic scattering cross section, and absorption macroscopic cross section are inherent physical properties of a material, and by definition

$$\Sigma_a + \Sigma_s = \Sigma_t \,. \tag{6.81}$$

Likewise for the spatially discretized case, it is true that

$$\mathbf{R}_{\Sigma_a} = \mathbf{R}_{\Sigma_t} - \mathbf{R}_{\Sigma_s} \,, \tag{6.82}$$

since we have defined

$$\mathbf{R}_{\Sigma_a,ij} = \frac{\Delta x}{2}\int_{-1}^{1}\Sigma_a(s)b_i(s)b_j(s)\;ds \tag{6.83a}$$

$$\mathbf{R}_{\Sigma_s,ij} = \frac{\Delta x}{2}\int_{-1}^{1}\Sigma_s(s)b_i(s)b_j(s)\;ds \tag{6.83b}$$

$$\mathbf{R}_{\Sigma_t,ij} = \frac{\Delta x}{2}\int_{-1}^{1}\Sigma_t(s)b_i(s)b_j(s)\;ds\,. \tag{6.83c}$$

Equation 6.82 can be verified by inserting the definitions of Eqs. (6.83) into Eq. (6.82),

$$\mathbf{R}_{\Sigma_a,ij} \stackrel{?}{=} \frac{\Delta x}{2}\int_{-1}^{1}\Sigma_t(s)b_i(s)b_j(s)\;ds - \frac{\Delta x}{2}\int_{-1}^{1}\Sigma_s(s)b_i(s)b_j(s)\;ds\,, \tag{6.84}$$

and obviously,

$$\mathbf{R}_{\Sigma_a,ij} = \frac{\Delta x}{2}\int_{-1}^{1}\left(\Sigma_t(s)-\Sigma_s(s)\right)b_i(s)b_j(s)\;ds = \frac{\Delta x}{2}\int_{-1}^{1}\Sigma_a(s)b_i(s)b_j(s)\;ds\,. \tag{6.85}$$

However, though the forms of Eq. (6.45) and Eq. (6.78) appear analogous to neutron transport, it is not obvious that the linearized, pseudo interaction cross sections (Eq. (6.45)) and their spatially discretized analogs, (Eq. (6.78)) are equivalent. That is, if we started with the spatially analytic linearization, Eq. (6.45), re-cast as,

$$\mu_d\frac{dI}{dx} + \widetilde{\Sigma}_t I = \frac{1}{4\pi}\widetilde{\Sigma}_s\phi + \xi_{d,i} \tag{6.86}$$

with

$$\widetilde{\Sigma_t} = \sigma_{\tau,i} \tag{6.87}$$

$$\widetilde{\Sigma_s} = \sigma_s + \nu_i\sigma_a\,, \tag{6.88}$$

then applied the Galerkin procedure, yielding,

$$\mu_d \mathbf{G}\vec{I} + \mathbf{R}_{\widetilde{\Sigma}_t}\vec{I} = \frac{1}{4\pi}\mathbf{R}_{\widetilde{\Sigma}_s}\vec{\phi} + \vec{\vec{\xi}}_{d,i}\,, \tag{6.89}$$

with

$$\mathbf{R}_{\widetilde{\Sigma}_t,jk} = \frac{\Delta x}{2}\int_{-1}^{1} \widetilde{\Sigma}_t(s) b_j(s) b_k(s)\; ds \tag{6.90a}$$

$$\mathbf{R}_{\widetilde{\Sigma}_s,jk} = \frac{\Delta x}{2}\int_{-1}^{1} \widetilde{\Sigma}_s(s) b_j(s) b_k(s)\; ds \tag{6.90b}$$

$$\vec{\vec{\xi}}_{d,j} = \frac{\Delta x}{2}\int_{-1}^{1} b_j(s)\xi_{d,i}\; ds\,, \tag{6.90c}$$

would the definitions of Eqs. (6.90) be equivalent to

$$\mathbf{R}_{\widetilde{\Sigma}_t} \stackrel{?}{=} \overline{\overline{\mathbf{R}}}_{\sigma_{\tau,i}} \tag{6.91a}$$

$$\mathbf{R}_{\widetilde{\Sigma}_s} \stackrel{?}{=} \mathbf{R}_{\sigma_s} + \overline{\overline{\nu}}_i \mathbf{R}_{\sigma_a}? \tag{6.91b}$$

In Eqs. (6.90), we use i subscript to denote SDIRK stage i, j to denote matrix row, and k to denote matrix column. We will continue this notation for the remainder of this section.

We first consider the equivalence of $\overline{\overline{\mathbf{R}}}_{\sigma_{\tau,i}}$ to $\mathbf{R}_{\widetilde{\Sigma}_t}$. By definition,

$$\overline{\overline{\mathbf{R}}}_{\sigma_{\tau,i}} = \frac{1}{a_{ii}c\Delta t}\mathbf{M} + \mathbf{R}_{\sigma_t}\,. \tag{6.92}$$

It follows that,

$$\overline{\overline{\mathbf{R}}}_{\sigma_{\tau,i},jk} = \frac{1}{a_{ii}c\Delta t}\frac{\Delta x}{2}\int_{-1}^{1} b_j(s)b_k(s)\; ds + \frac{\Delta x}{2}\int_{-1}^{1}\sigma_t(s) b_j(s) b_k(s)\; ds \tag{6.93}$$

$$\overline{\overline{\mathbf{R}}}_{\sigma_{\tau,i},jk} = \frac{\Delta x}{2}\int_{-1}^{1}\left(\frac{1}{a_{ii}c\Delta t} + \sigma_t(s)\right) b_j(s) b_k(s)\; ds\,. \tag{6.94}$$

From Eqs. (6.90),

$$\mathbf{R}_{\widetilde{\Sigma}_t,jk} = \frac{\Delta x}{2}\int_{-1}^{1}\left(\frac{1}{a_{ii}\Delta t c}+\sigma_t(s)\right)b_j(s)b_k(s)\;ds\,, \tag{6.95}$$

and we conclude that $\overline{\overline{\mathbf{R}}}_{\sigma_{\tau,i}} = \mathbf{R}_{\widetilde{\Sigma}_t}$. This is a very important result because it implies that we have a consistent definition of $\widetilde{\Sigma}_t$, that in turn allows us to define a diffusion coefficient for MIP DSA,

$$D(s) = \frac{1}{3\widetilde{\Sigma}_t(s)} = \frac{1}{3\left(\frac{1}{a_{ii}c\Delta t}+\sigma_t(s)\right)}\,. \tag{6.96}$$

We now attempt to determine if

$$\mathbf{R}_{\widetilde{\Sigma}_s} = \mathbf{R}_{\sigma_s} + \overline{\overline{\nu}}_i\mathbf{R}_{\sigma_a}\,. \tag{6.97}$$

First, we expand the definition of $\mathbf{R}_{\widetilde{\Sigma}_s}$ from Eqs. (6.90), with the definition of TRT pseudo cross sections,

$$\mathbf{R}_{\widetilde{\Sigma}_s,jk} = \frac{\Delta x}{2}\int_{-1}^{1} b_j(s)b_k(s)\;ds\left(\sigma_s+\frac{4\pi a_{ii}\Delta t\sigma_a D_*}{C_v+4\pi a_{ii}\Delta t\sigma_a D_*}\sigma_a\right). \tag{6.98}$$

Obviously, we may split Eq. (6.98) into separate components:

$$\mathbf{R}_{\widetilde{\Sigma}_s,jk} = \frac{\Delta x}{2}\int_{-1}^{1}\sigma_s b_j(s)b_k(s)\;ds + \ldots \\ \frac{\Delta x}{2}\int_{-1}^{1} b_j(s)b_k\left(\frac{4\pi a_{ii}\Delta t\sigma_a D_*}{C_v+4\pi a_{ii}\Delta t\sigma_a D_*}\sigma_a\right)ds\,. \tag{6.99}$$

Clearly

$$\frac{\Delta x}{2}\int_{-1}^{1}\sigma_s b_j(s)b_k(s)\;ds = \mathbf{R}_{\sigma_s}\,, \tag{6.100}$$

so we are left to determine whether

$$\frac{\Delta x}{2}\int_{-1}^{1} b_j(s)b_k\left(\frac{4\pi a_{ii}\Delta t\sigma_a D_*}{C_v+4\pi a_{ii}\Delta t\sigma_a D_*}\sigma_a\right)ds = \overline{\overline{\nu}}_i\mathbf{R}_{\sigma_a}\,. \tag{6.101}$$

Using the definition of $\overline{\overline{\nu}}_i$ from Eqs. (6.77), we expand $\overline{\overline{\nu}}_i\mathbf{R}_{\sigma_a}$,

$$\overline{\overline{\nu}}_i\mathbf{R}_{\sigma_a} = 4\pi\Delta t a_{ii}\mathbf{R}_{\sigma_a}\mathbf{D}_*\left[\mathbf{I}+4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\,. \tag{6.102}$$

In the general case, when $\mathbf{R}_{\sigma_a}$ and $\mathbf{R}_{C_v}$ are dense, Eq. (6.102) will not yield a matrix with elements

$$\left[\overline{\overline{\nu}}_i\mathbf{R}_{\sigma_a}\right]_{jk} \neq \frac{\Delta x}{2}\int_{-1}^{1} b_j(s)b_k\left(\frac{4\pi a_{ii}\sigma_a D_*}{C_v+4\pi a_{ii}\Delta t\sigma_a D_*}\sigma_a\right)ds\,. \tag{6.103}$$

Though we have showed in the general case, $\mathbf{R}_{\sigma_s}+\overline{\overline{\nu}}_i\mathbf{R}_{\sigma_a} \neq \mathbf{R}_{\widetilde{\Sigma}_s}$, we wish to investigage whether self-lumping DFEM schemes may be better suited for the solution of the TRT equations. Consider an arbitrary, linear DFEM self-lumping scheme,

with the following matrices,

$$\mathbf{R}_{C_v} = \begin{bmatrix} \frac{\Delta x}{2} w_l C_{v,l} & 0 \\ 0 & \frac{\Delta x}{2} w_r C_{v,r} \end{bmatrix} \tag{6.104a}$$

$$\mathbf{R}_{\sigma_a} = \begin{bmatrix} \frac{\Delta x}{2} w_l \sigma_{a,l} & 0 \\ 0 & \frac{\Delta x}{2} w_r \sigma_{a,r} \end{bmatrix} \tag{6.104b}$$

$$\mathbf{D}_* = \begin{bmatrix} D_{*,l} & 0 \\ 0 & D_{*,r} \end{bmatrix} \tag{6.104c}$$

$$D_{*,l} = \left.\frac{dB}{dT}\right|_{T=\widetilde{T}_{*,l}} \tag{6.104d}$$

$$D_{*,r} = \left.\frac{dB}{dT}\right|_{T=\widetilde{T}_{*,r}} \tag{6.104e}$$

with the l and r subscripts denoting values at the two quadrature points, s_l and s_q Since $\mathbf{R}_{C_v}$ is diagonal,

$$\mathbf{R}_{C_v}^{-1} = \begin{bmatrix} \frac{2}{\Delta x w_l C_{v,l}} & 0 \\ 0 & \frac{2}{\Delta x w_r C_{v,r}} \end{bmatrix}, \tag{6.105}$$

we have

$$\left[\mathbf{I} + 4\pi\Delta t a_{ii} \mathbf{R}_{C_v}^{-1} \mathbf{R}_{\sigma_a} \mathbf{D}_*\right]_{11} = 1 + 4\pi\Delta t a_{ii} \left(\frac{2}{\Delta x w_l C_{v,l}}\right) \left(\frac{\Delta x w_l \sigma_{a,l}}{2}\right) D_{*,l} \tag{6.106}$$

$$\left[\mathbf{I} + 4\pi\Delta t a_{ii} \mathbf{R}_{C_v}^{-1} \mathbf{R}_{\sigma_a} \mathbf{D}_*\right]_{22} = 1 + 4\pi\Delta t a_{ii} \left(\frac{2}{\Delta x w_r C_{v,r}}\right) \left(\frac{\Delta x w_r \sigma_{a,r}}{2}\right) D_{*,r} \tag{6.107}$$

$$\left[\mathbf{I} + 4\pi\Delta t a_{ii} \mathbf{R}_{C_v}^{-1} \mathbf{R}_{\sigma_a} \mathbf{D}_*\right]_{12} = \left[\mathbf{I} + 4\pi\Delta t a_{ii} \mathbf{R}_{C_v}^{-1} \mathbf{R}_{\sigma_a} \mathbf{D}_*\right]_{21} = 0\,. \tag{6.108}$$

Completing the definition of $\overline{\overline{\nu}}_i \mathbf{R}_{\sigma_a}$,

$$\overline{\overline{\nu}}_i \mathbf{R}_{\sigma_a} = 4\pi a_{ii}\Delta t \begin{bmatrix} \frac{\Delta x w_l \sigma_{a,l}}{2} & 0 \\ 0 & \frac{\Delta x w_l \sigma_{a,r}}{2} \end{bmatrix} \begin{bmatrix} D_{*,l} & 0 \\ 0 & D_{*,r} \end{bmatrix} \times ... \\ \begin{bmatrix} \frac{1}{1+4\pi\Delta a_{ii}\frac{\sigma_{a,l}}{C_{v,l}}} & 0 \\ 0 & \frac{1}{1+4\pi\Delta a_{ii}\frac{\sigma_{a,r}}{C_{v,r}}} \end{bmatrix} \begin{bmatrix} \frac{2}{\Delta x w_l C_{v,l}} & 0 \\ 0 & \frac{2}{\Delta x w_r C_{v,r}} \end{bmatrix} \begin{bmatrix} \frac{\Delta x w_l \sigma_{a,l}}{2} & 0 \\ 0 & \frac{\Delta x w_l \sigma_{a,r}}{2} \end{bmatrix} \tag{6.109}$$

$$\overline{\overline{\nu}}_i \mathbf{R}_{\sigma_a} = 4\pi a_{ii}\Delta t \frac{\Delta x}{2} \begin{bmatrix} \frac{w_l \sigma_{a,l} D_{*,l}}{1+4\pi\Delta a_{ii}\frac{\sigma_{a,l}}{C_{v,l}}} \frac{\sigma_{a,l}}{C_{v,l}} & 0 \\ 0 & \frac{w_r \sigma_{a,r} D_{*,r}}{1+4\pi\Delta a_{ii}\frac{\sigma_{a,r}}{C_{v,r}}} \frac{\sigma_{a,r}}{C_{v,r}} \end{bmatrix} \tag{6.110}$$

For some quantity f, we note that $\mathbf{R}_f$ with our linear, self-lumping quadrature would be:

$$\mathbf{R}_f = \begin{bmatrix} \frac{\Delta x}{2} w_l f_l & 0 \\ 0 & \frac{\Delta x}{2} w_r f_r \end{bmatrix} . \tag{6.111}$$

Considering Eq. (6.110), and thinking about $\overline{\overline{\nu}}_i \mathbf{R}_{\sigma_a} = \mathbf{R}_f$

$$f_l = 4\pi a_{ii}\Delta t \frac{\sigma_{a,l} D_{*,l}}{C_{v,l} + 4\pi\Delta a_{ii}\sigma_{a,l}} \sigma_{a,l} \tag{6.112}$$

$$f_r = 4\pi a_{ii}\Delta t \frac{\sigma_{a,r} D_{*,r}}{C_{v,r} + 4\pi\Delta a_{ii}\sigma_{a,r}} \sigma_{a,r} \tag{6.113}$$

$$f = 4\pi a_{ii}\Delta t \frac{\sigma_a D_*}{C_v + 4\pi\Delta a_{ii}\sigma_a} \sigma_a = \nu_i \sigma_a \tag{6.114}$$

$$\mathbf{R}_{\nu_i \sigma_a} = \overline{\overline{\nu}}_i \mathbf{R}_{\sigma_a} . \tag{6.115}$$

Thus, when accounting for the within cell spatial variation of material properties, only when using self-lumping DFEM schemes would it be equivalent to

1. linearize the Planckian of the spatially analytic grey TRT and then spatially

discretize, or

2. spatially discretize the grey TRT equations then linear the Planckian.

Given this information, in the most general case, when using MIP DSA to accelerate grey radiative transfer, we will define the integration in cell k (Eq. (4.54)) as:

$$(\Sigma_a \Delta\phi, b_*) = \mathbf{R}_{\sigma_{\tau,i}} - \left(\mathbf{R}_{\sigma_s} + \overline{\overline{\nu}}_i \mathbf{R}_{\Sigma_a}\right) \vec{\Delta\phi}_k \,, \tag{6.116}$$

where $\Delta\phi$ is the low order approximation of the angle integrated intensity error, not the neutron transport scalar flux error. Likewise, we will define the MIP DSA driving source, the difference between two successive iterates, as

$$\left(\Sigma_s(\phi^{(\ell+1/2)} - \phi^{(\ell)}), b_*\right) = \left(\mathbf{R}_{\sigma_s} + \overline{\overline{\nu}}_i \mathbf{R}_{\Sigma_a}\right) \left(\vec{\phi}^{(\ell+1)} - \vec{\phi}^{(\ell)}\right) . \tag{6.117}$$

Since we have shown that

$$\mathbf{R}_{\sigma_{\tau,i},jk} = \frac{\Delta x}{2} \int_{-1}^{1} ds \left(\frac{1}{a_{ii} c \Delta t} + \sigma_t\right) b_j(s) b_k(s) \,, \tag{6.118}$$

we will use the point-wise definition of Eq. (6.96) to evaluate $\mathbf{S}$ and MIP DSA edge integrals requiring knowledge of a point-wise diffusion coefficient.

6.2 Iterative Solution Process

To solve the time-dependent, grey TRT equations, we use a time marching loop with two levels of nested iteration, as detailed through the pseudo-code shown in Listing 6.1.

Listing 6.1: TRT iteration pseudo-code

```
time = t_start; t_step = 0
while !end_of_time
{
  t_step += 1; dt = calculate_dt(t_step, dt_old,time,t_end)
  t_star = t_old
  for stage = 1:1:n_stage
  {
    time_stage = time + dt*c_sdirk[stage]
    damping = 1; thermal_iter = 0; thermal_converged = false
    while !thermal_converged
    {
      thermal_iter += 1; intensity_converged = false; phi_new = 0
      while !intensity_converged
      {
        phi_new = calculate_new_intensity_iterate(t_star)
        change_phi = normalized_diff(phi_new,phi_old)
        intensity_converged = change_phi < epsilon_phi
        phi_old = phi_new
      }
      [t_star,change_t] = update_temperature(t_star,phi_new,damping)
      thermal_converged = change_t < epsilon_temperature
      damping = check_if_damping_needed(thermal_iter)
    }
    k_I[stage] = calculate_k_I(t_star,phi_new)
    k_T[stage] = calculate_k_T(t_star,phi_new)
  }
  advance_intensity(i_old,k_I)
  advance_temperature(t_old,k_T)
  time += dt; end_of_time = time < t_end
}
```

We terminate the thermal iteration using a point-wise relative change criterion. However, we only store two objects of temperature unknowns, each of size $N_{cell} \times N_P$, corresponding to $\vec{T}_n$ and $\vec{T}_*$ $\vec{T}_n$ is modified only after the completion of a time step. This is not a problem, as we already have the point-wise change in temperature, $\Delta\vec{T} = \vec{T}_*^{(\ell+1)} - \vec{T}_*^{(\ell)}$, from Eq. (6.73):

$$\Delta\vec{T} = \left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\left[\vec{T}_n - \vec{T}_* + \Delta t\sum_{j=1}^{i-1} a_{ij}k_{T,j}\right]\ldots$$
$$+ \Delta t a_{ii}\left[\mathbf{I} + 4\pi\Delta t a_{ii}\mathbf{R}_{C_v}^{-1}\mathbf{R}_{\sigma_a}\mathbf{D}_*\right]^{-1}\mathbf{R}_{C_v}^{-1}\left[\mathbf{R}_{\sigma_a}\left(\vec{\phi}_i - 4\pi\vec{B}_*\right) + \vec{S}_T\right], \quad (6.119)$$

and

$$\text{change\_t} = \max_{c=1}^{N_{cell}}\left[\max_{j=1}^{N_P}\left[\left|\frac{\Delta T_j}{T_{j,*}}\right|\right]\right]. \quad (6.120)$$

Likewise, we terminate the intensity update iteration using a point-wise relative change criterion:

$$\text{change\_phi} = \max_{c=1}^{N_{cell}}\left[\max_{j=1}^{N_P}\left[\left|\frac{\phi_{c,j}^{(\ell+1)} - \phi_{c,j}^{(\ell)}}{\phi_{c,j}^{(\ell+1)}}\right|\right]\right]. \quad (6.121)$$

Since MIP DSA requires two iterates of the angle integrated radiation energy density, $\phi^{(\ell+1)}$ and $\phi^{(\ell)}$, both of size $N_{cell} \times N_P$, we store at least two objects of the same dimensionality as ϕ. We use one of these objects, `phi_new` in Listing 6.1, to update $\vec{T}^*$, while the other is local only to the intensity update. Our code is designed to use only a single intensity data object of size $N_{cell} \times N_P \times N_{dir}$, $\vec{I}_N$. Limiting ourselves to a single `i_old` requires our intensity update convergence to be based upon the angle integrated intensity and to perform one additional sweep while calculating $\vec{k}_I$. The SDIRK time integration technique requires that we save data objects k_T and k_I with, $N_{stage} \times N_{cell} \times N_P$ and $N_{stage} \times N_{cell} \times N_P \times N_{dir}$ unknowns, respectively. For

all significant values of N_{dir}, the need to store $\vec{I}_n$ and $\vec{k}_I$ will dominate the memory footprint of our implementation, and a reasonable upper bound on memory usage is $(1 + N_{stage}) \times N_{dir} \times N_{cell} \times N_P$.

6.3 Software Implementation

We have implemented our grey radiative transfer equations in a C++ 11 computer code. All attempts have been made to incorporate the best practices of modern, C++ programming and software design[45, 46]. We have made extensive use of the object-oriented programming paradigm of virtual base classes with concrete instantiations. For example, we have compared the effects of different methods of DFEM mass matrix construction. Regardless of whether we use exact matrix construction, traditional lumping, or self lumping integration, solution of the radiative transfer equations requires access to a mass matrix, for example when calculating $\vec{k}_I$ in Eq. (6.47). To shelter other portions of our code and programming logic from the particulars of a given simulation's choice of mass matrix construction, in our radiative transfer implementation, we declared a virtual base class, `V_Matrix_Construction` with pure virtual member function `construct_dimensionless_mass_matrix()` . The concrete instantiations of `V_Matrix_Construction`: `Matrix_Construction_Exact`, `Matrix_Construction_Trad_Lumping`, and `Matrix_Construction_Self_Lumping`, each exhibit the object-oriented programming inheritance "is a" relationship with base class `V_Matrix_Construction`. Then, through the use of C++ 11's smart pointers, in particular the `std::shared_ptr`, at program run-time we declare, once during the entire program's execution, the particular instantiation of base class `V_Matrix_Construction` we wish our smart pointer, named `matrix_construction`, to point to/possess. From this point forward, anytime a mass matrix is needed, we simply call `matrix_construction->construct_dimensionless_mass_matrix()`, and,

regardless of our choice of DFEM integration strategy, the appropriate mass matrix is returned.

Where possible we have used third party software to prevent duplication of efforts. All Gauss-Legendre, Gauss-Lobatto, and closed Newton-Cotes quadrature functions are derived from the `QUADRULE` [47] package, with minor modifications including: use C++ `std::vector`s rather than arrays and encapsulation of the `QUADRULE` functions into a `QUADRULE` class to limit access and contamination of the global name space. We have directly implemented all of the matrix/vector based equations in this chapter, as written, using the `Eigen` linear algebra package [48]. To invert the MIP DSA matrix used to accelerate the iterative solution of the grey TRT equations, we use the `PETSc` package[49] preconditioned with algebraic multigrid[41] via the `BoomerAMG` package of `Hypre` [50]. We document our code using inline comments and `Doxygen` [51]. To build and test the components of our grey thermal radiative transfer code, we use the CMake and CTest packages from Kitware[52]. By using CTest to verify and test the code, we have created a set (greater than 50) of unit tests that can be performed every time the code is changed or compiled. Tests range in size from single component to full simulations using the method of manufactured solutions[53] testing for convergence. Though requiring more additional work than simply using `std::cout` to test components as added then commenting out the output statements, CTest allows for continuous testing to find bugs that the programmer would not otherwise suspect. Finally, input parameters for the code are input an `XML` file, which we read using `TinyXML` [54].

6.4 Numerical Results

We now consider several test problems to verify and demonstrate the capability of the self-lumping DFEM schemes we have developed to solve the grey TRT equations.

6.4.1 Method of Manufactured Solutions

The method of manufactured solutions, MMS, consists of choosing an analytic solution, then defining the driving source necessary to achieve that solution [53]. Though MMS solutions are derived, they are excellent for testing the convergence of numerical methods devised to solve complex physics phenomena. While analytic solutions that do not use exotic source terms are more appealing for verification purposes, these types of solutions are difficult to devise for complex physics phenomena, especially for coupled physical phenomena like radiative transfer. Or if an analytic solution exists, solutions are really semi-analytic, and cannot be used effectively as a reference solution for a convergence study because the semi-analytic solution has numerical errors of the same magnitude (or greater) than the numerical simulations we are attempting to verify. Similarly experimental results will contain measurement errors, and will almost certainly be of low resolution, either measuring only integral quantities, or providing data at only a few points in space, and thus do not permit convergence studies.

We elect to choose separable MMS of the form:

$$
\begin{aligned}
I_d(x, \mu_d, t) &= M(\mu_d) F(t) W_I(x) && \text{(6.122a)} \\
T(x) &= F(t) W_T(x) && \text{(6.122b)} \\
\phi(x) &= C_M F(t) W_I(x) && \text{(6.122c)} \\
C_M &= \sum_{d=1}^{N_{dir}} w_d M(\mu_d) \,. && \text{(6.122d)}
\end{aligned}
$$

where $M(\mu_d)$ is as angular component of the intensity solution desired, $I_d(x, \mu_d, t)$; $F(t)$ is the time component of the MMS, chosen to be the same for intensity, temperature, and angle integrated intensity; $W_I(x)$ is the spatial component of the intensity,

and $W_T(x)$ is the spatial component of the temperature solutions. $W_I(x)$ being the spatial component of ϕ, and the definition of C_M given in Eqs. (6.122) is required for the S_N approximation to be true.

6.4.1.1 Linear in Time - Trigonometric in Space

In our first method of manufactured solutions problem, MMS1, we attempt to reproduce a TRT problem whose radiation and temperature solutions is a cosine in space and linear in time. We use a non-dimensional form of the TRT, assuming that $a = c = 1$, and for this problem we assume opacities are constant functions of space and temperature. We will impose the solution:

$$M(\mu_d) = \frac{1}{4\pi} \tag{6.123}$$

$$W_I(x) = 10\cos\left(\frac{\pi x}{10} - \frac{\pi}{2}\right) + 15 \tag{6.124}$$

$$W_T(x) = 25\cos\left(\frac{\pi x}{10} - \frac{\pi}{2}\right) + 30 \tag{6.125}$$

$$F(t) = 1 + .02x\,. \tag{6.126}$$

Physically, $x \in [0, 10]$, $C_v = 0.1$, $\sigma_a = 100.0$, $\sigma_s = 0.5$, we start the simulation at $t_0 = 0$, and use 100 equal time steps of length $\Delta t = 0.01$, with Alexander's two stage, second order SDIRK method, which we will refer to as 2-2 [2]. The Butcher tableaux for this and all other SDIRK schemes considered is given in Section 6.4.1.4. We evaluate $\vec{S}_I$ and $\vec{S}_T$ using the spatially analytic definitions of S_I and S_T respectively, and numerically integrate the source moments using a N_q Gauss-Legendre quadrature points, with $N_q = 2N_P + 5$, and N_P is the number of DFEM interpolation points, as defined previously.

We consider three DFEM schemes,

1. TL: traditional mass matrix lumping using equally-spaced interpolation points,

2. SL Lobatto: self-lumping quadrature using Lobatto interpolation points,

3. SL Gauss: self-lumping quadrature using Gauss interpolation points, and

compare each method's accuracy in calculating the discrete L_2 norm of the error in angle integrated intensity, $E_\phi = \left\| \widetilde{\phi}(x) - \phi(x) \right\|_{L^2}$, and the L_2 error in temperature, $E_T = \left\| \widetilde{T}(x) - T(x) \right\|_{L^2}$. E_ϕ and ET are calculated analogously to the definition of E_ψ in Eq. (2.43). Figure 6.1, Fig. 6.2, and Fig. 6.3 plot the order of convergence of E_ϕ for the TL, SL Lobatto, and SL Gauss schemes, respectively, as a function of trial space degree and cell size. Analogous data for E_T is given in Figs. 6.4-6.6.

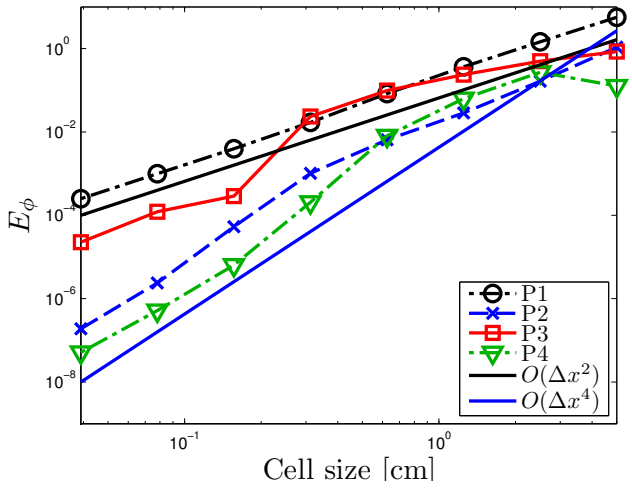

Figure 6.1: Convergence of E_ϕ for the TL scheme in problem MMS1.

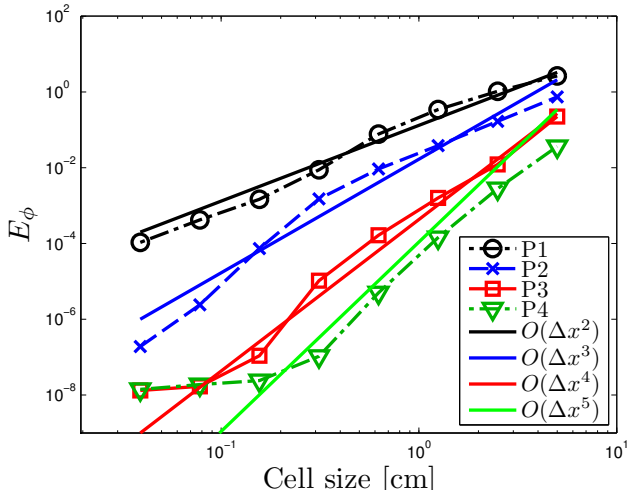

Figure 6.2: Convergence of E_ϕ for the SL Lobatto scheme in problem MMS1.

We make several observations regarding Figs. 6.1-6.6. First, the TL scheme does not necessarily increase in accuracy with an increase in trial space degree, is limited to

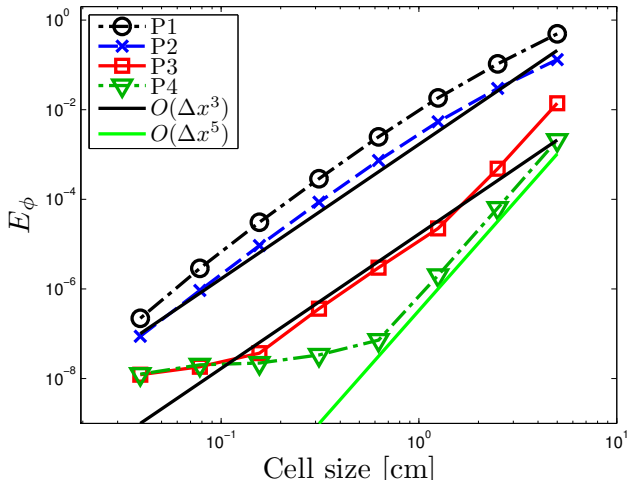

Figure 6.3: Convergence of E_ϕ for the SL Gauss scheme in problem MMS1.

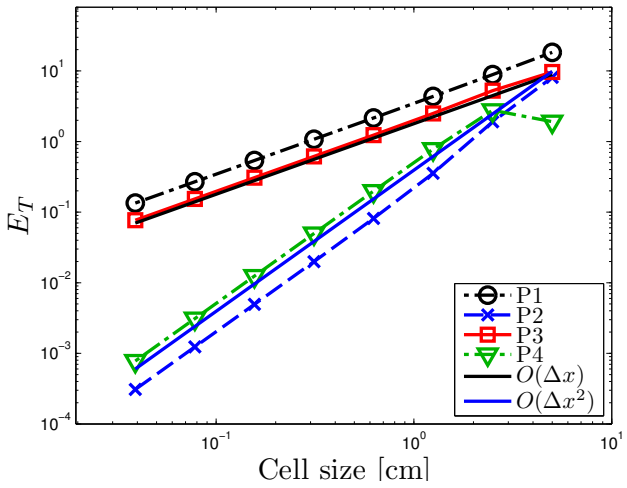

Figure 6.4: Convergence of E_T for the TL scheme in problem MMS1.

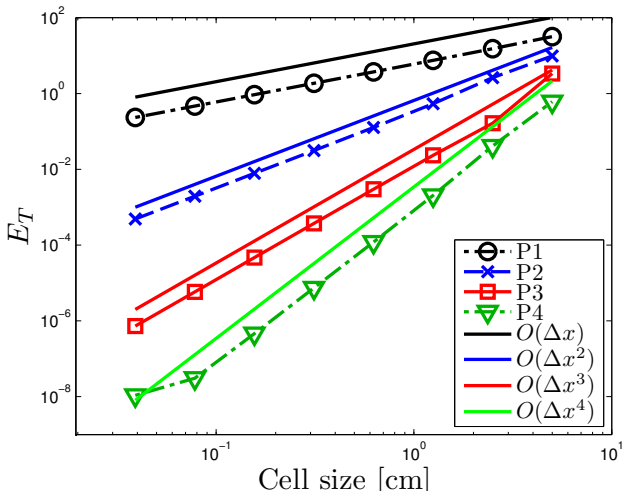

Figure 6.5: Convergence of E_T for the SL Lobatto scheme in problem MMS1.

second order spatial convergence for odd degree trial space DFEM, and is limited to their order spatial convergence for even degree trial space schemes, behavior identical

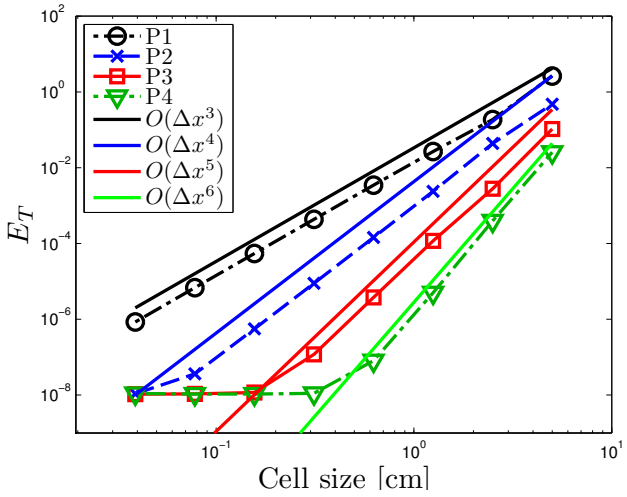

Figure 6.6: Convergence of E_T for the SL Gauss scheme in problem MMS1.

to that demonstrated in our neutron transport testing. Second, SL Lobatto converges the L^2 norm of the radiation energy density $\propto P+1$, the same order of convergence achieved in our neutron transport test problems for the convergence of L^2 error of the angular and scalar fluxes. However, SL Lobatto only converges $E_T \propto P$, not $P+1$, as was the case for the neutron transport interaction rate. Finally, SL Gauss converges $E_\phi \propto P+1$, the same order of convergence as for the neutron transport analog, and exhibits super convergence in converging $E_T \propto P+2$.

To understand the behaviors of SL Lobatto and SL Gauss completely, we now consider the L^2 like norm of cell average radiation energy density error, E_{ϕ_A} and cell average temperature error, E_{T_A}. E_{ϕ_A} and E_{T_A} are calculated analogously to Eq. (2.44). Convergence of E_{ϕ_A} is given in Fig. 6.7 for SL Lobatto and Fig. 6.8 for SL Gauss. Convergence of E_{T_A} is given in Figs. 6.9-6.10 for SL Lobatto and SL Gauss, respectively. While no clear pattern emerges in the convergence of E_{ϕ_A} as a function of P for SL Lobatto in Fig. 6.7 when considering linear through quartic polynomial

trial spaces, it is clear that the orders of convergence exhibited are less than orders of convergence exhibited by SL Lobatto in Figs. 2.17-2.20 for the analogous neutron transport quantity E_{ψ_A}. Similarly, the radiative transfer variant of SL Gauss does not converge E_{ϕ_A} at as high a degree as SL Gauss converged E_{ψ_A} for a pure absorber in neutron transport. However, comparing Fig. 6.8 to Figs. 2.17-2.20, it appears that SL Gauss converges E_{ϕ_A} for TRT at most one degree less than SL Gauss converges E_{ψ_A} in neutron transport. Interestingly, the observed decreases in the convergence of E_{ϕ_A} for TRT relative to neutron transport convergence of E_{ψ_A} does not hold when considering the convergence ofE_{T_A}. SL Lobatto converges E_{T_A} for TRT with the same order as SL Lobatto converges E_{ψ_A} and E_{IR_A} for neutron transport. SL Gauss actually exhibits an E_{T_A} order of convergence that exceeds its neutron transport analogs.

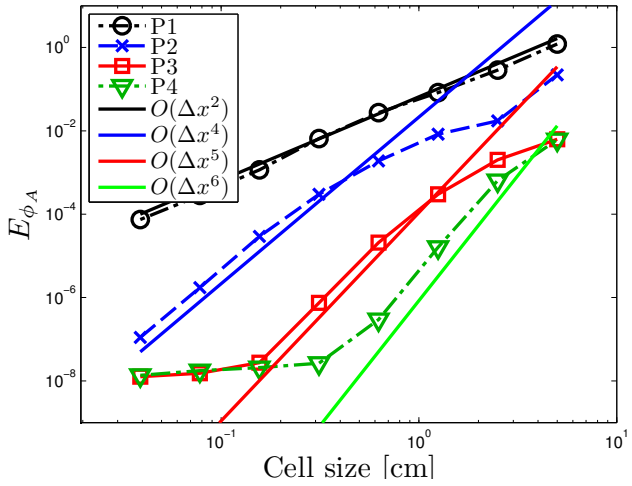

Figure 6.7: Convergence of E_{ϕ_A} for the SL Lobatto scheme in problem MMS1.

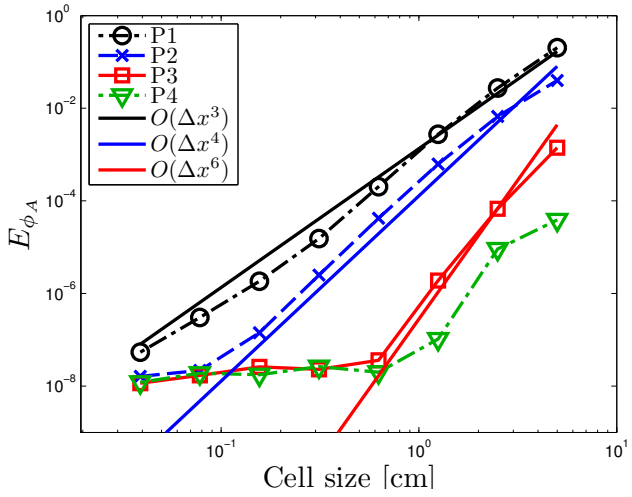

Figure 6.8: Convergence of E_{ϕ_A} for the SL Gauss scheme in problem MMS1.

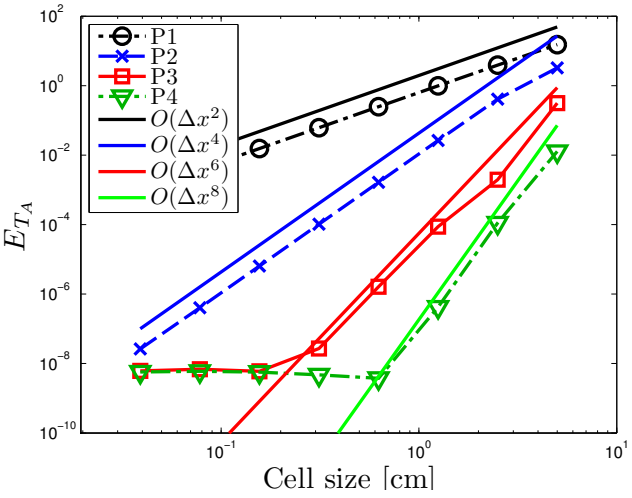

Figure 6.9: Convergence of E_{T_A} for the SL Lobatto scheme in problem MMS1.

It is our hypothesis that the apparent super convergence demonstrated by SL Gauss in converging E_T is related to how we expand the Planckian in the same trial space as the DFEM temperature and radiation solutions. That is to say that we suspect the N_P quadrature points of a given trial space degree SL Gauss scheme are significantly more accurate at integrating

$$b_i(s)B(\widetilde{T})\,, \tag{6.127}$$

a degree $5P$ polynomial, than an N_P Lobatto quadrature. Though the N_P point Gauss quadrature only exactly integrates polynomials of degree $2(P+1)-1$, it may be that the remaining terms are coincidentally very small in magnitude for a degree $5P$ polynomial of the particular nature of Eq. (6.127).

This plateauing of errors seen in the above plots is a result of two things. First, the nested nature of our TRT solution algorithm. Due to the nested iterations, we

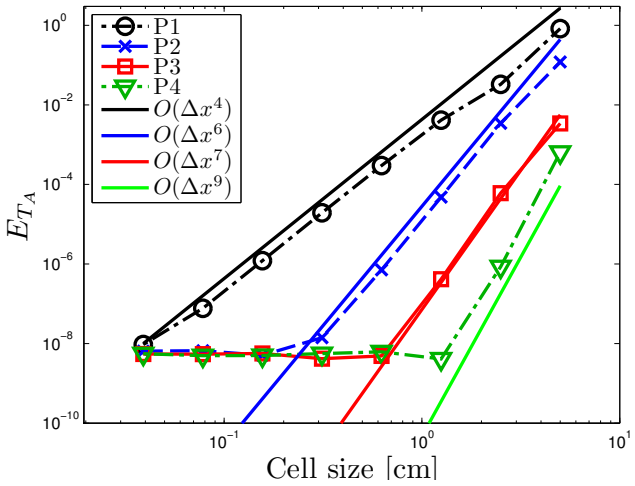

Figure 6.10: Convergence of E_{T_A} for the SL Gauss scheme in problem MMS1.

must use the tightest tolerance on the innermost solve, the intensity update. We terminate the innermost solve after reaching a certain point-wise change tolerance, ϵ_{inner}, of the zero-th angular moment of the angle integrated intensity:

$$\max_j \left| \frac{\phi_j^{(\ell+1)} - \phi_j^{(\ell)}}{\phi_j^{(\ell+1)}} \right| < \epsilon_{inner} \,. \tag{6.128}$$

Similarly, the thermal iteration of SDIRK stage s is terminated when every temperature solution at the DFEM interpolation points, T_j, changes less than $\epsilon_{thermal}$,

$$\max_j \left| \frac{T_{j,s}^{(\ell+1)} - T_j^{(\ell)}}{T_j^{(\ell+1)}} \right| < \epsilon_{thermal} \,. \tag{6.129}$$

For this and all other MMS problems, $\epsilon_{inner} = 10^{-12}$ and $\epsilon_{thermal} = 10^{-10}$. Secondly, as the TRT equations are functions of space and time, it possible that the error of a given spatial discretization is smaller than the temporal error of the problem, thus a plateauing of the solution error as a function of spatial mesh refinement occurs. Thus, truly accurate solutions require the refinement of both spatial mesh and time step size.

6.4.1.2 Trigonometric in Space - Linear in Time - with Temperature Dependent Material Properties

We now consider a problem with temperature dependent material properties, MMS2. We impose the following solution:

$$\begin{aligned} M(\mu_d) &= \frac{1}{4\pi} && (6.130)\\ W_I(x) &= 9\cos\left(\frac{\pi x}{10} - \frac{\pi}{2}\right) + 3\,, && (6.131)\\ W_T(x) &= 5\cos\left(\frac{\pi x}{10} - \frac{\pi}{2}\right) + 5\,, && (6.132)\\ F(t) &= 1 + .02t\,, && (6.133) \end{aligned}$$

and define the following material properties:

$$C_v = 0.2 + 0.01T^3 \tag{6.134}$$

$$\sigma_a = \frac{10^4}{T^3} \tag{6.135}$$

$$\sigma_s = 0.5\,. \tag{6.136}$$

In our problem, $x \in [0, 10]$, $t \in [0, 2]$, we use the three stage, third order accurate SDRIK method of Alexander [2] with $\Delta t = 0.001$. In total , we will consider four different DFEM schemes for this problem, SL Lobatto, SL Gauss, SLXS Lobatto, and SLXS Gauss. The methods denoted SL will assume cell-wise constant opacities and heat capacities, equal to the cell-wise volumetric average of that quantity. The SLXS schemes will explicitly account for the within cell variation of temperature dependent material properties by evaluating $\mathbf{R}$ as in Eq. (3.8). E_ϕ for the SL Lobatto and SL Gauss schemes is plotted in Fig. 6.11 and Fig. 6.12, respectively.

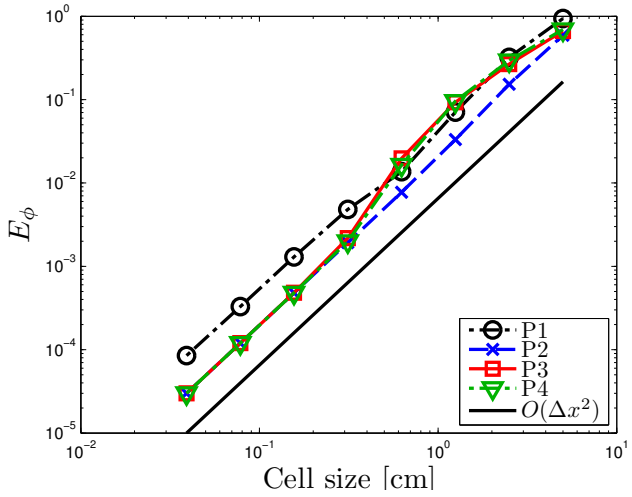

Figure 6.11: Convergence of E_ϕ for the SL Lobatto scheme in problem MMS2.

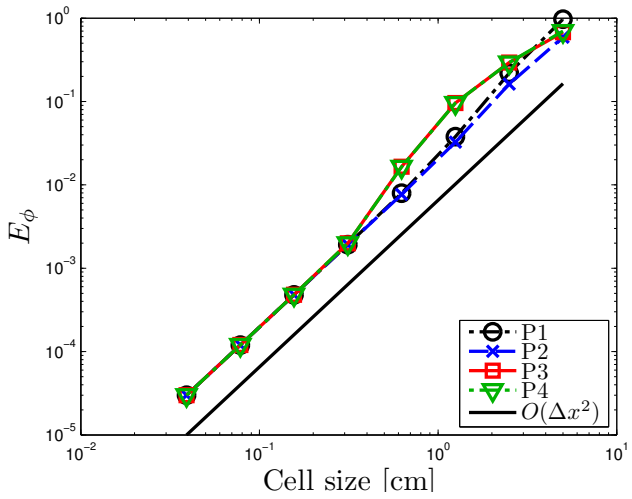

Figure 6.12: Convergence of E_ϕ for the SL Gauss scheme in problem MMS2.

Regardless of DFEM interpolation point type or trial space degree, assuming cell-wise constant material properties limits spatial convergence of E_ϕ to at most second order. Confirming our suspicion that assuming cell-wise constant material properties limit the convergence of E_T as well, we plot the convergence of E_T for the SL Lobatto scheme in Fig. 6.13 and for the SL Gauss scheme in Fig. 6.14. Figures 6.13-6.14 verify the hypothesis we developed while discussing our neutron transport results, assuming a cell wise constant opacity limits L^2 convergence of temperature to at most first order in space, regardless of trial space degree. Wishing to verify that the high order of convergence observed for E_{IR_A} in Section 3 does not translate to a high rate of spatial convergence for E_{T_A}, we consider Fig. 6.15 and Fig. 6.16. Figure 6.15 verifies that regardless of trial space degree, the SL Lobatto scheme assuming a cell-wise constant opacity for a problem with temperature or spatially varying opacities converges E_{T_A} second order in space. Likewise, Fig. 6.16 demonstrates the same is

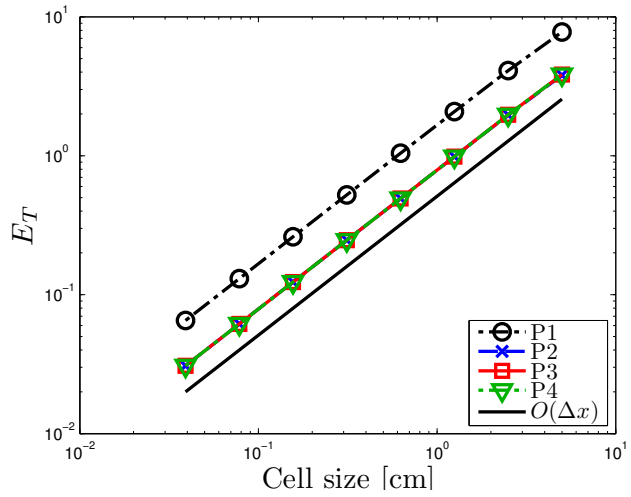

Figure 6.13: Convergence of E_T for the SL Lobatto scheme in problem MMS2.

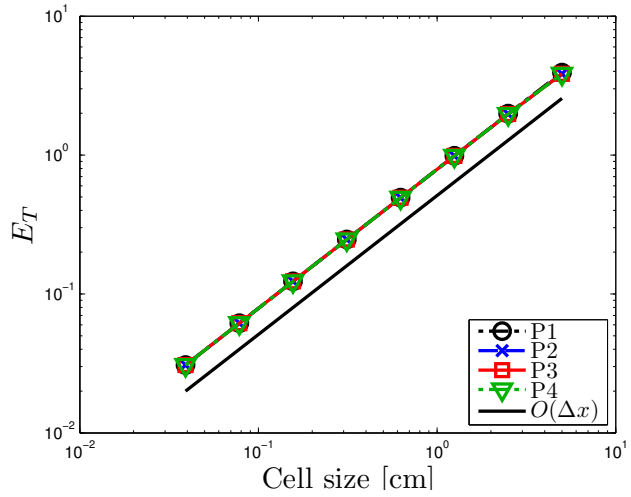

Figure 6.14: Convergence of E_T for the SL Gauss scheme in problem MMS2.

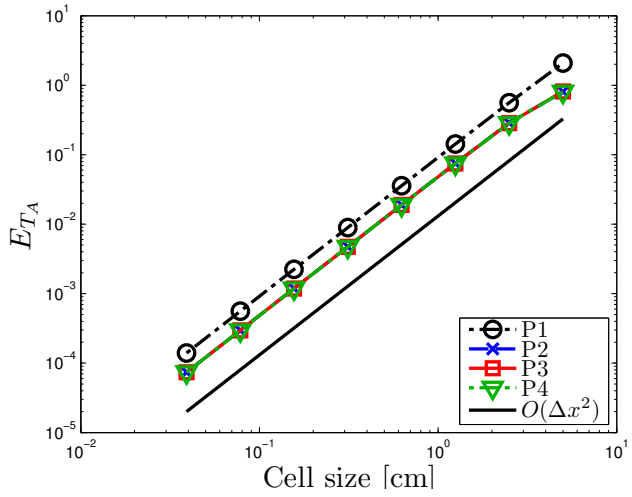

Figure 6.15: Convergence of E_{T_A} for the SL Lobatto scheme in problem MMS2.

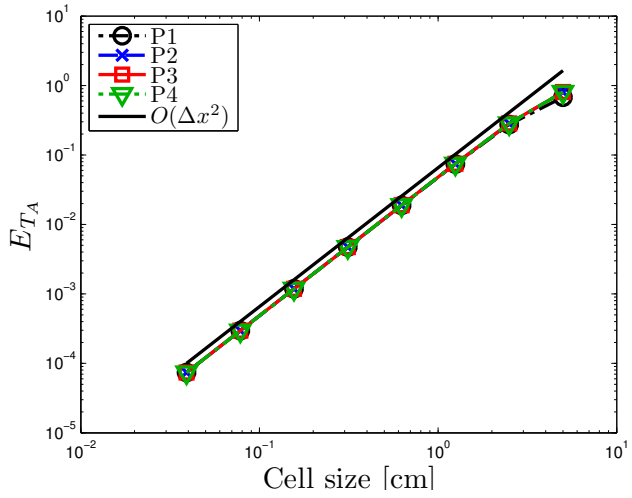

Figure 6.16: Convergence of E_{T_A} for the SL Gauss scheme in problem MMS2.

true for the SL Gauss scheme.

We now consider the convergence of SLXS Lobatto and SLXS Gauss. First, we see that in Fig. 6.17, SLXS Lobatto converges $E_\phi \propto P+1$, and in Fig. 6.18, SLXS Gauss converges E_ϕ at least $\propto P+1$.

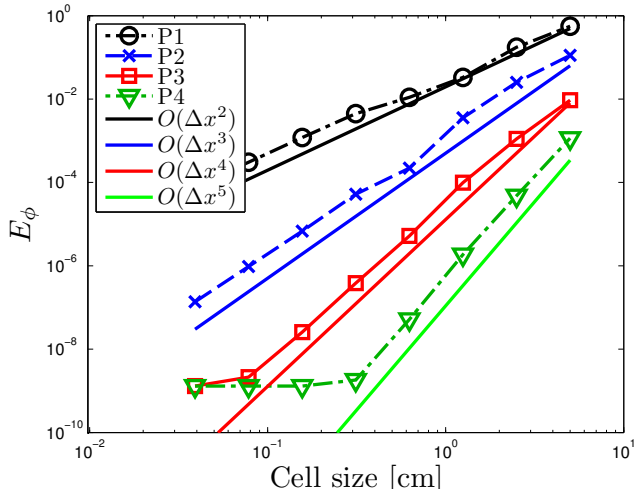

Figure 6.17: Convergence of E_ϕ for the SLXS Lobatto scheme in problem MMS2.

Moving on to the convergence of E_T, in Fig. 6.19 SLXS Lobatto converges $E_T \propto P$, the same rate SL Lobatto converged E_T for the TRT problem with constant material properties. In Fig. 6.20 SLXS Gauss converges E_T very rapidly, and only the asymptotic convergence rate of linear and quadratic DFEM can be estimated, though both suggest SLXS Gauss converges $E_T \propto P+2$.

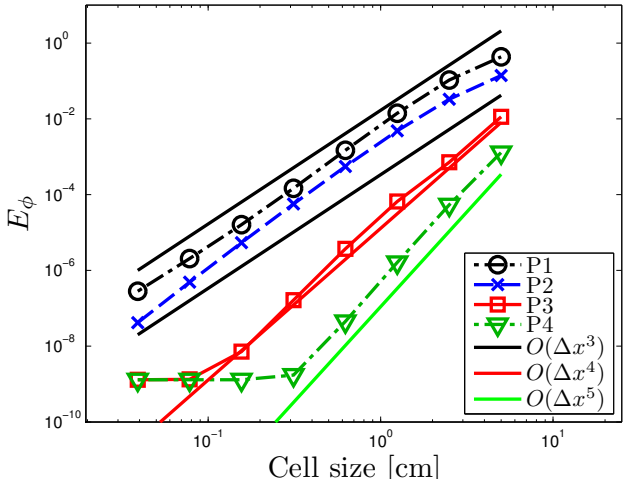

Figure 6.18: Convergence of E_ϕ for the SLXS Gauss scheme in problem MMS2.

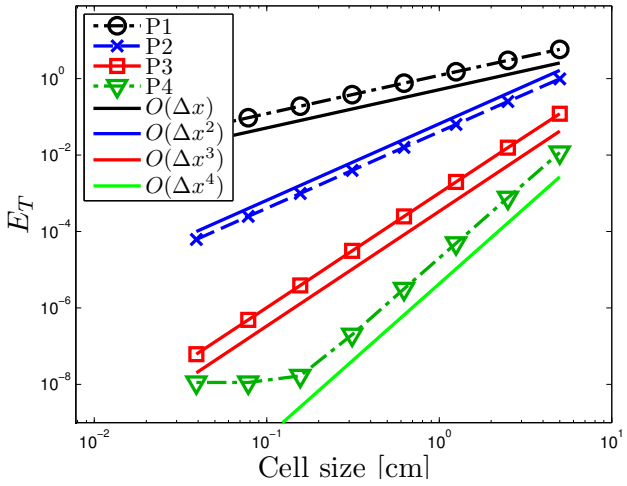

Figure 6.19: Convergence of E_T for the SLXS Lobatto scheme in problem MMS2.

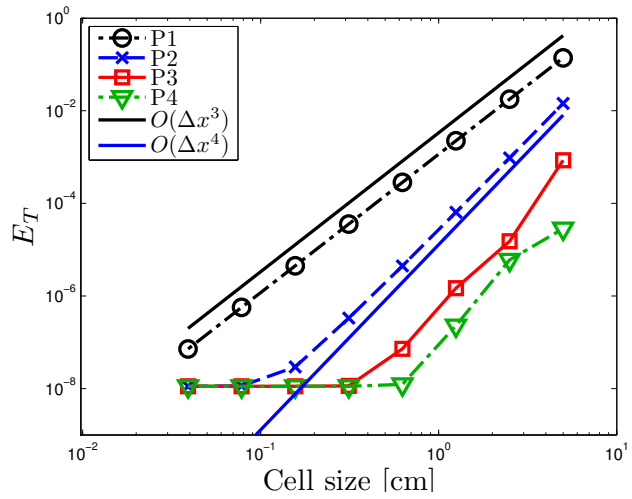

Figure 6.20: Convergence of E_T for the SLXS Gauss scheme in problem MMS2.

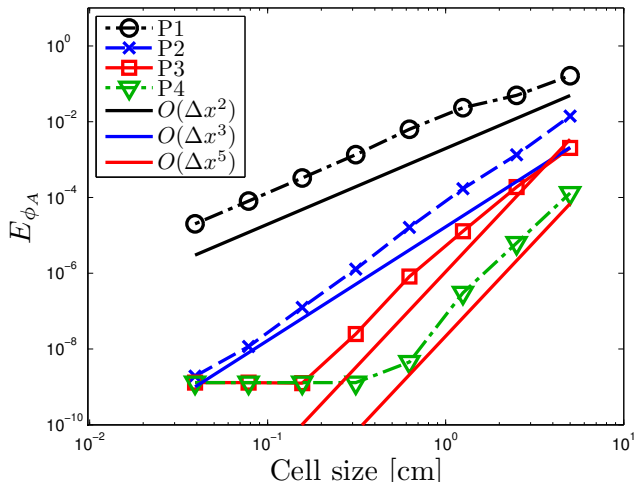

Figure 6.21: Convergence of E_{ϕ_A} for the SLXS Lobatto scheme in problem MMS2.

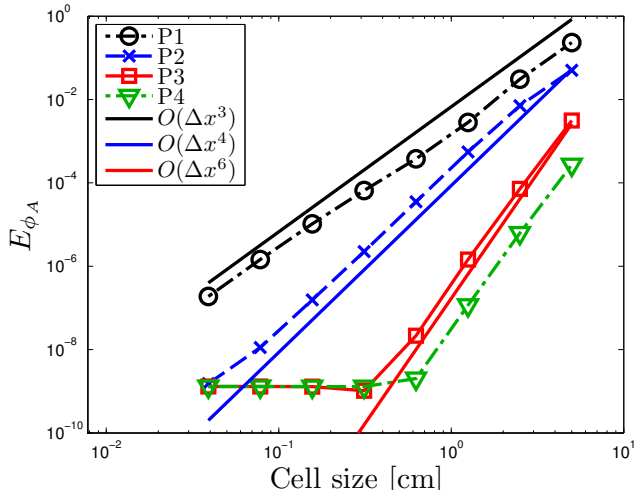

Figure 6.22: Convergence of E_{ϕ_A} for the SLXS Gauss scheme in problem MMS2.

Focusing now on the convergence of cell average error quantities, we first consider E_{ϕ_A} for SLXS Lobatto and SLXS Gauss in Fig. 6.21 and Fig. 6.22, respectively. Due to the high accuracy of both of both SLXS Lobatto and SLXS Gauss, it is difficult to draw many conclusions from Fig. 6.21 and Fig. 6.22. All we can say for certain in that both converge E_{ϕ_A} at an order less than or at most equal to the order their respective neutron transport analogs converged E_{ψ_A}. Reference order of convergence lines are given, but only linear DFEM in both figures appears to have reached a truly asymptotic order of convergence. For completeness, we include convergence plots of E_{T_A} for SLXS Lobatto and SLXS Gauss respectively in Fig. 6.23 and Fig. 6.24. All we can conclude from Figs. 6.23-6.24 is that both SLXS Lobatto and SLXS Gauss experience increases in convergence of E_{T_A} with increases in trial space degree.

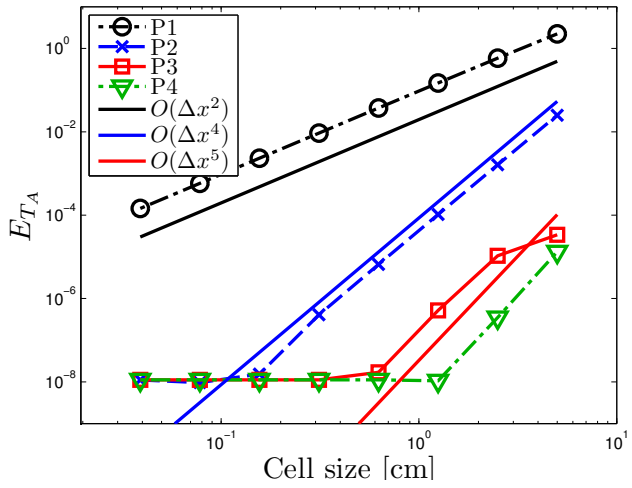

Figure 6.23: Convergence of E_{T_A} for the SLXS Lobatto scheme in problem MMS2.

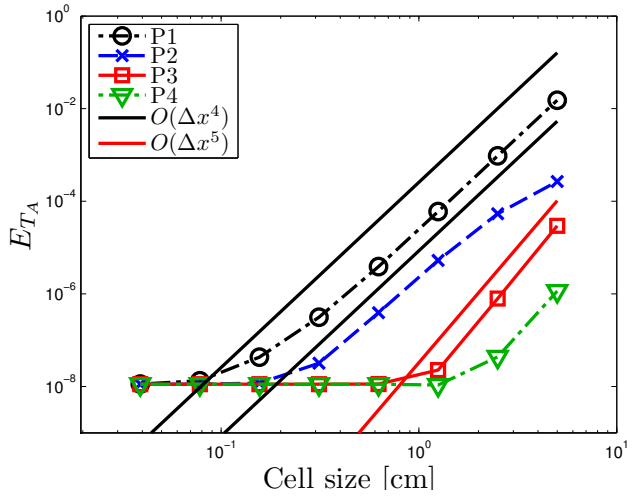

Figure 6.24: Convergence of E_{T_A} for the SLXS Gauss scheme in problem MMS2.

6.4.1.3 Constant in Time - Trigonometric in Space - with Temperature Dependent Material Properties

We now consider a problem that is constant in time and varies as a cosine in space with temperature dependent material properties. Though very similar to MMS2, we hope that by considering a truly steady state problem, we can study the spatial error of our DFEM schemes without temporal error interference. We impose a solution of the form:

$$M(\mu_d) = \frac{1}{4\pi} \tag{6.137}$$

$$W_I(x) = 19\cos\left(\frac{\pi x}{2}\right) + 20\,, \tag{6.138}$$

$$W_T(x) = 15\cos\left(\frac{\pi x}{2}\right) + 20\,, \tag{6.139}$$

$$F(t) = 10 \tag{6.140}$$

and define the following material properties:

$$C_v = 0.1 + 0.2T^2 \tag{6.141}$$

$$\sigma_a = \frac{5}{T^2} \tag{6.142}$$

$$\sigma_s = 0.01\,. \tag{6.143}$$

Repeating as we have before, we first consider the convergence of E_ϕ for SLXS Lobatto and SLXS Gauss in Fig. 6.25 and Fig. 6.26. Eliminating the temporal error makes it clear that for TRT problems with temperature dependent quantities, SLXS Lobatto converges $E_\phi \propto P + 1$, and SLXS Gauss converges $E_\phi \propto P + 2$. Likewise, the stead-state problem makes it clear that SLXS Lobatto converges $E_T \propto P$ in Fig. 6.27 and SLXS Gauss converges $E_T \propto P + 2$ in Fig. 6.28.

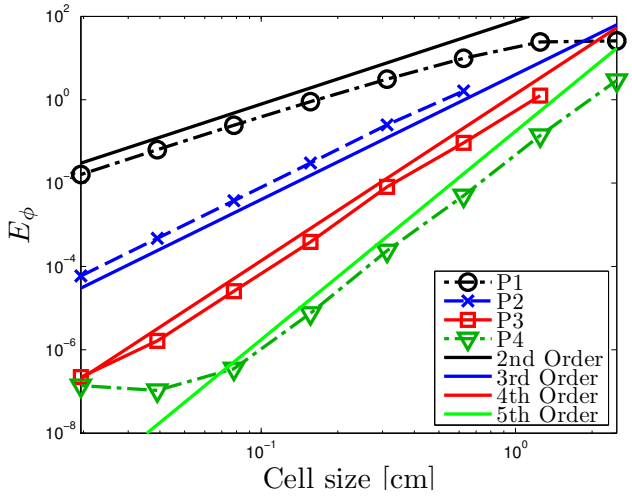

Figure 6.25: Convergence of E_ϕ for SLXS Lobatto scheme for steady state test problem.

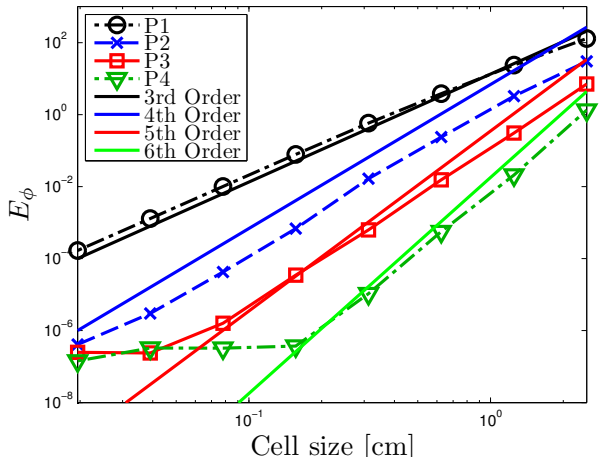

Figure 6.26: Convergence of E_ϕ for SLXS Gauss scheme, for steady state test problem.

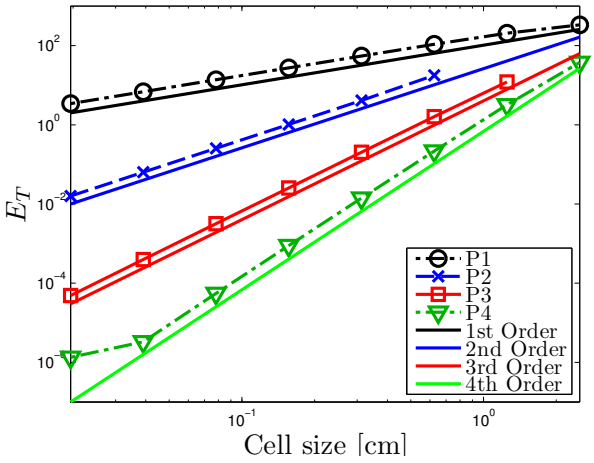

Figure 6.27: Convergence of E_T for SLXS Lobatto scheme, for steady state test problem.

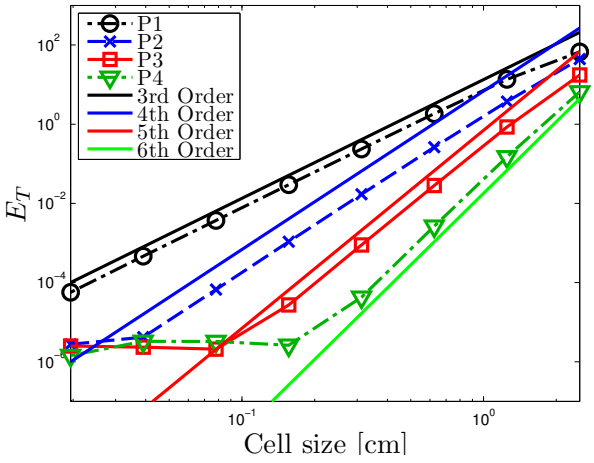

Figure 6.28: Convergence of E_T for SLXS Gauss scheme, for steady state test problem.

We now examine the convergence of E_{ϕ_A}. The orders of convergence estimated by Fig. 6.29 for SLXS Lobatto, are not identical to those estimated in MMS2 shown in Fig. 6.21. Even with the steady problem, the error converges quickly enough that solver tolerance is quickly reached, preventing an obvious development of asymptotic behavior. Similarly the convergence of E_{ϕ_A} for SLXS Gauss in Fig. 6.30 is not identical to the estimates of Fig. 6.22.

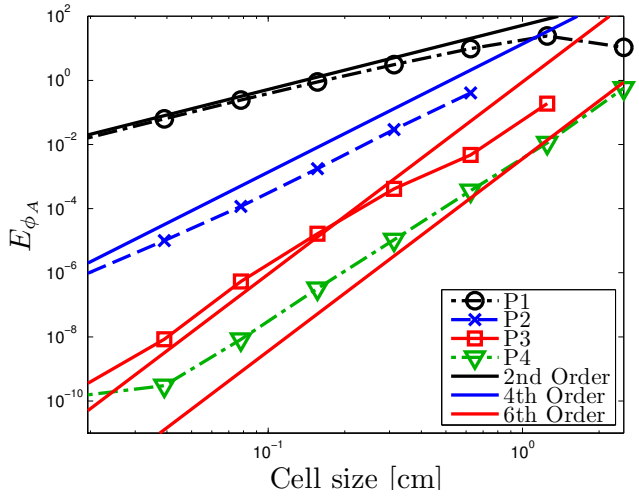

Figure 6.29: Convergence of E_{ϕ_A} for SLXS Lobatto scheme, for steady state test problem.

Finally, we consider the convergence of E_{T_A}. In Fig. 6.31 the SLXS Lobatto TRT convergence of E_{T_A} appears to be $\propto 2P$, the same order of convergence exhibited by SLXS Lobatto in neutron transport for converging E_{IR_A}. SLXS Gauss converges $E_{T_A} \propto 2P + 2$ in Fig. 6.32, whereas SLXS Gauss for neutron transport converges $E_{IR_A} \propto 2P + 1$ as seen in Figs. 3.21-3.24.

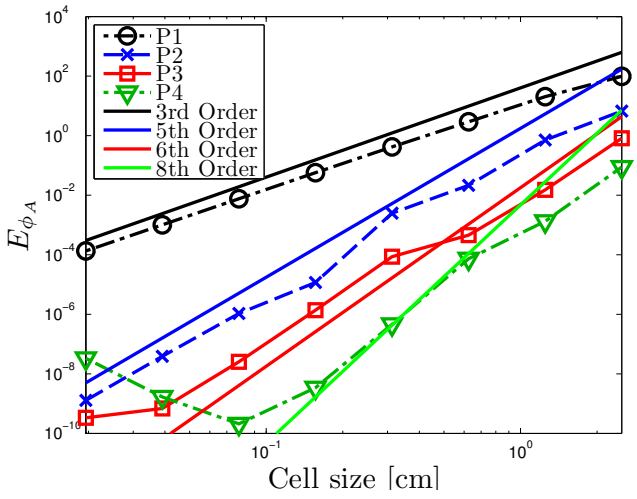

Figure 6.30: Convergence of E_{ϕ_A} for SLXS Gauss scheme, for steady state test problem.

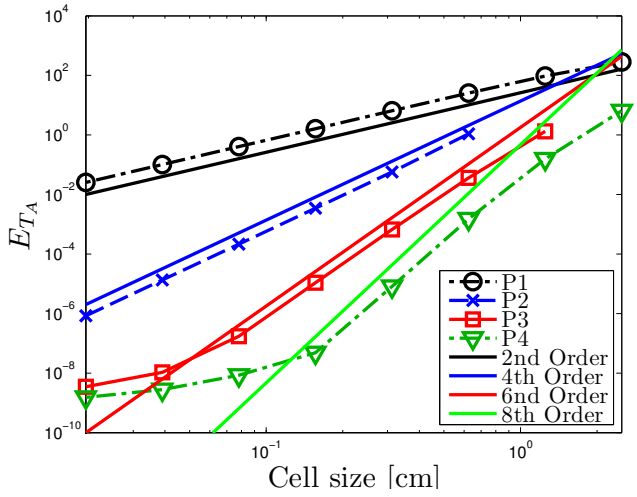

Figure 6.31: Convergence of E_{T_A} for SLXS Lobatto scheme, for steady state test problem.

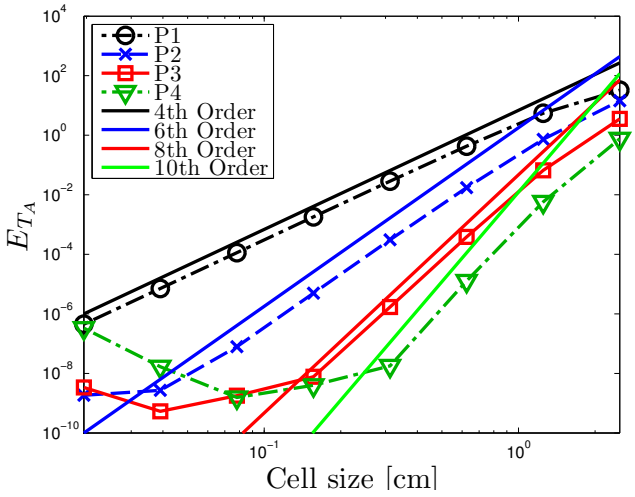

Figure 6.32: Convergence of E_{T_A} for SLXS Gauss scheme, for steady state test problem.

6.4.1.4 Constant in Space - Trigonometric in Time

We consider three different SDIRK schemes in our work, implicit Euler (IE); a two stage, second order S-stable scheme (2-2); and a three stage, third order S-stable scheme (3-3). Both multi-stage schemes were taken from [2]. The Butcher tableaux of the implicit Euler scheme can be written as:

$$\begin{array}{c|c|c} 1 & 1 & 1 \\ \hline & & 1 \end{array}. \tag{6.144}$$

The 2-2 scheme of Alexander has the following Butcher tableaux,

$$\begin{array}{c|cc|cc} 1 & 1-\frac{\sqrt{2}}{2} & & 1-\frac{\sqrt{2}}{2} & 0 \\ 2 & 1 & & \frac{\sqrt{2}}{2} & 1-\frac{\sqrt{2}}{2} \\ \hline & & & \frac{\sqrt{2}}{2} & 1-\frac{\sqrt{2}}{2} \end{array}, \tag{6.145}$$

and the 3-3 scheme's Butcher tableaux is:

$$\begin{array}{c|c|ccc} 1 & \gamma & \gamma & & \\ 2 & \frac{1+\gamma}{2} & \frac{1-\gamma}{2} & \gamma & \\ 3 & 1 & \delta & \beta & \gamma \\ \hline & & \delta & \beta & \gamma \end{array}, \tag{6.146}$$

with

$$\gamma = 0.435866521508459 \tag{6.147}$$

$$\delta = \frac{-6\gamma^2 + 16\gamma - 1}{4} \tag{6.148}$$

$$\beta = \frac{6\gamma^2 - 20\gamma + 5}{4}. \tag{6.149}$$

We now verify the asymptotic order of convergence of each SDIRK scheme. To simplify the process, we consider a problem with constant spatial dependence:

$$M(\mu_d) = \frac{1}{4\pi} \tag{6.150}$$

$$W_I(x) = \frac{10}{4\pi} \tag{6.151}$$

$$W_T(x) = 10 \tag{6.152}$$

$$F(t) = 45\cos(\pi t) + 46\,, \tag{6.153}$$

$t \in [0, 1]$, $\sigma_s = 0.1$, $\sigma_a = 2.5$, $C_v = 0.2$, $x \in [0, 10]$ discretized with 10 equally spaced cells. Convergence of E_ϕ as a function of Δt for the IE, 2-2, and 3-3 time differencing schemes are given in Fig. 6.33. As expected, the IE converges 1st order in time, the 2-2 scheme converges second order, and the Alexander 3-3 scheme converges third order in time. The same data for temperature is given in Figs. 6.34-6.35. Surprisingly,

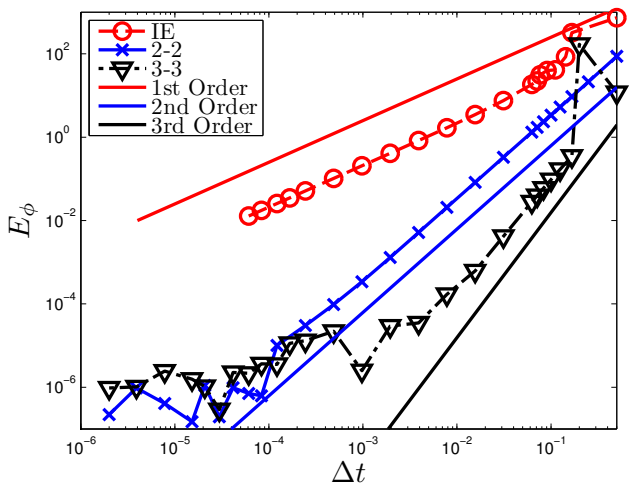

Figure 6.33: Convergence of E_ϕ for different time integrators as a function of Δt.

it took a very large number of time steps for the 2-2 and 3-3 schemes to become asymptotic. Figure 6.34 shows the convergence of E_T for larger time steps, and Fig. 6.35 shows the convergence of E_T for smaller time steps, required for the 2-2 and 3-3 schemes to demonstrate their respective asymptotic orders of convergence.

6.4.2 Marshak Wave Problem

We now consider our most realistic test problem, a Marshak wave problem presented by Ober and Shadid for radiative diffusion [9]. We use the S_2 discrete ordinates approximation with a Gauss quadrature in angle, which is equivalent to a P_1 approximation in slab geometry [34] as compared to the P_0 approximation used in [9]. Like most thermal radiative transfer problems, no analytic solution exists. As such, we will focus on qualitative comparisons of how DFEM trial space degree, mesh refinement, and time step refinement affect the radiation energy density and material temperature solutions.

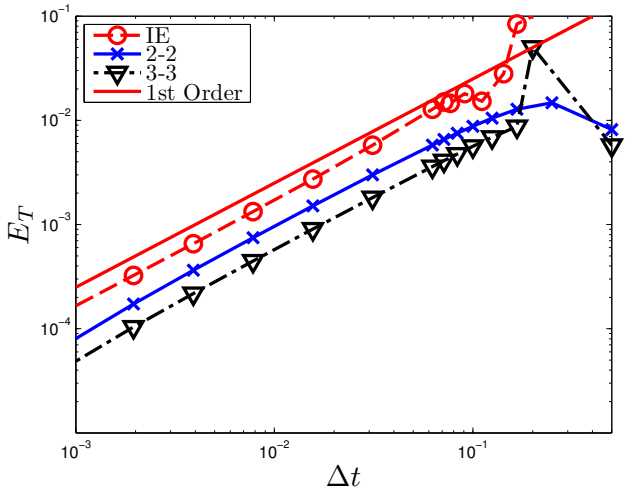

Figure 6.34: Convergence of E_T for different time integrators as a function of Δt for larger Δt.

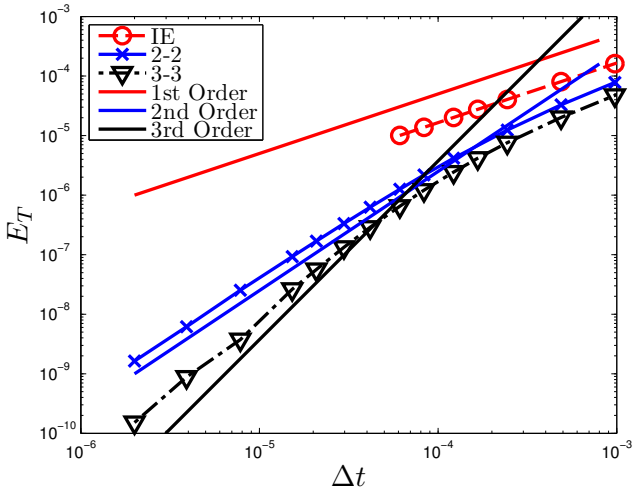

Figure 6.35: Convergence of E_T for different time integrators as a function of Δt for small Δt.

The Marshak wave problem consists of an initially cold slab that is heated from the left by an incident radiation intensity, with a vacuum boundary condition on the right. Again, the problem is dimensionless, $a = c = C_v = 1$; $x \in [0, 1]$; and we advance the solution from $t = 0$ to $t = 1$. Initially, the slab is in thermal equilibrium, and $T = (10^{-5})^{1/4}$. There is no scattering, $\sigma_s = 0$, and the absorption opacity is temperature dependent, $\sigma_a = \frac{1}{T^3}$.

We will use an initial mesh of 20 cells, but will also consider results using meshes of 80, 320, and 1280 cells. For all simulations we use the 2-2 Alexander scheme. We begin with a minimum time step of $\Delta t = 5 \times 10^{-4}$ and increase the time step size by 10% until we reach a maximum time step size of $\Delta t = 10^{-2}$. For our time refinement studies, we will divide the minimum and maximum time step sizes both by a factor of 4, 16, or 64. We will consider linear, quadratic, cubic and quartic SLXS Lobatto and SLXS Gauss DFEM schemes. To demonstrate that assuming a cell-wise constant opacity results in a bladed radiative transfer temperature solution, we will also consider linear SL Lobatto with a cell-wise volumetric average opacity.

We first investigate the effect, if any of assuming a cell-wise constant opacity. Figure 6.36 shows the linear SL Lobatto radiation energy density solution at $t = 1.0$. Except for the effects of a very coarse spatial mesh when using 20 cells, the radiation energy density solution is effectively smooth, and comparable to the results published in [9]. The full temperature solution is shown in Fig. 6.37. Clearly, the large, non-monotonic discontinuities (blading) observed in the neutron transport interaction rate profile are present in the TRT temperature profile. In Fig. 6.37, mesh refinement reduces the magnitude of the temperature solution blading but does not eliminate blading. To emphasize that blading is not eliminated with mesh refinement, consider Fig. 6.38, that zooms in on the material temperature solution near the Marshak wavefront. Clearly, Figs. 6.36-6.38, in addition to the limited order of convergence

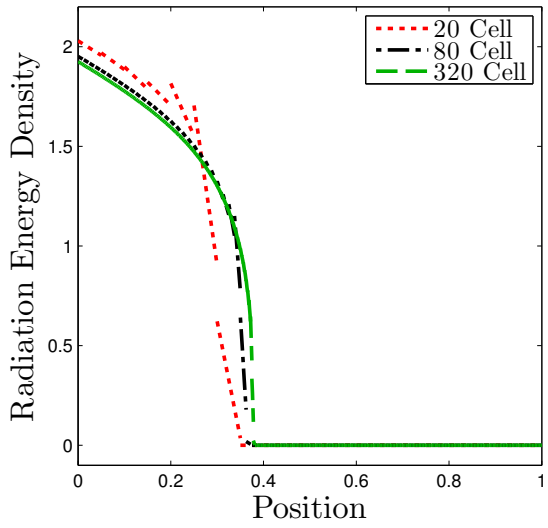

Figure 6.36: SL Lobatto radiation solution for the Marshak wave problem assuming cell-wise constant volumetric averaged opacities.

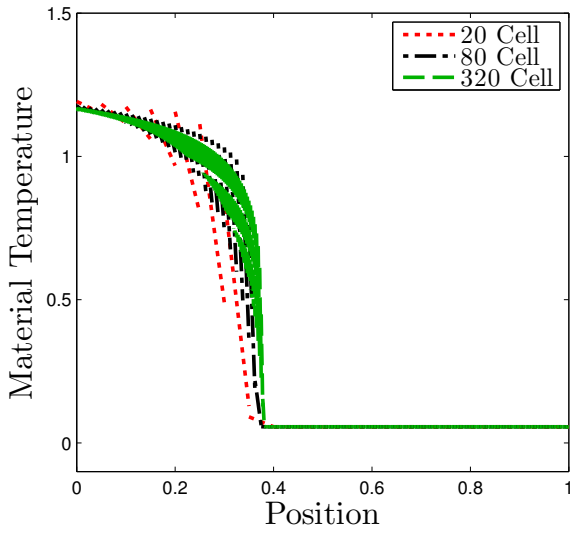

Figure 6.37: SL Lobatto temperature solution for the Marshak wave problem assuming cell-wise constant volumetric averaged opacities.

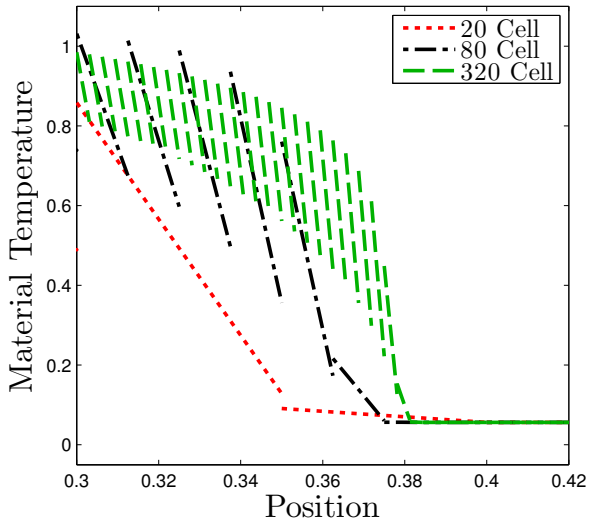

Figure 6.38: SL Lobatto temperature solution for the Marshak wave problem near the wavefront.

results given Fig. 6.13 and Fig. 6.14 demonstrate that the assumption of a cell-wise constant opacity is not appropriate for thermal radiative transfer simulations with temperature dependent material properties.

We now consider the effects of spatial mesh refinement on TRT solutions. We first consider the linear SLXS Lobatto scheme, looking at a zoom in near the wavefront of the radiation profile in Fig. 6.39 and the material temperature profile in Fig. 6.40 Though the changes are subtle, we can see that mesh refinement actually changes the location of both the radiation and temperature profile discontinuity, with the changes more prominent in the material temperature profile, Fig. 6.40. The changes in the material temperature profile are more pronounced, even when moving from 320 to 1280 mesh cells, than in the radiation profile most likely due to SLXS Lobatto's first order convergence of E_T. Plots similar to Fig. 6.39 and Fig. 6.40 are provided

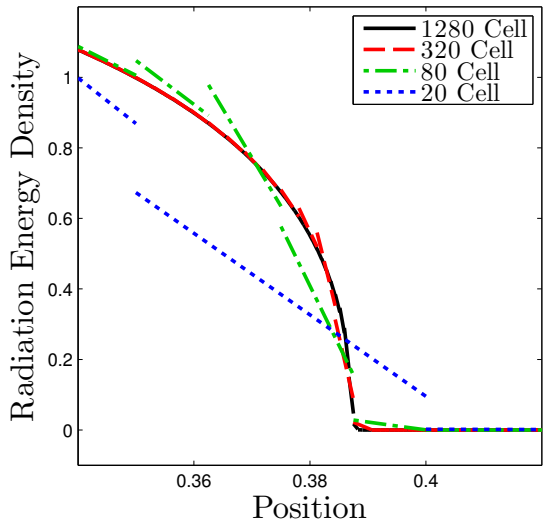

Figure 6.39: Linear SLXS Lobatto radiation solution near wavefront with increasing spatial mesh refinement.

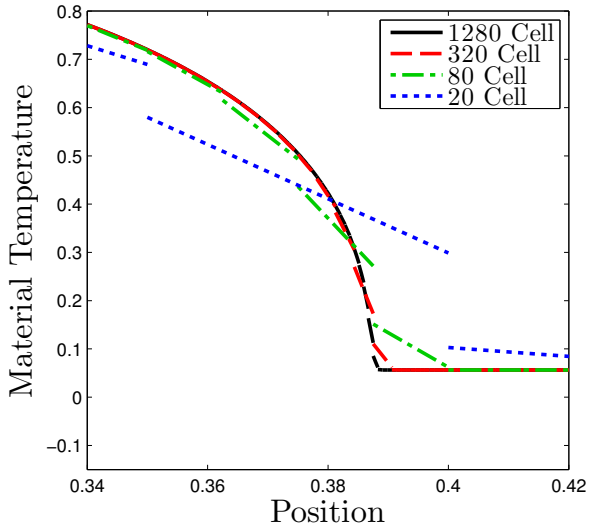

Figure 6.40: Linear SLXS Lobatto temperature solution near wavefront with increasing spatial mesh refinement.

in Fig. 6.41 and Fig. 6.42, respectively for the quartic SLXS Gauss scheme. Due to

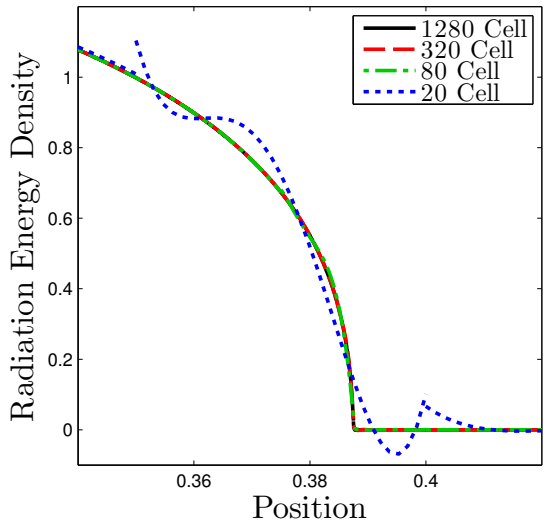

Figure 6.41: Quartic SLXS Gauss radiation solution near wavefront with increasing spatial mesh refinement.

the SLXS Gauss' high order of spatial convergence, few changes are noticeable with mesh refinement, except when moving from 20 to 80 cells. However, when moving from 20 to 80 cells, the Gibbs' phenomena near the solution discontinuity are no longer visible, except for a very small negativity in the temperature solution. All solutions for this spatial mesh refinement study used the finest time step sizes.

We now examine the effect of increasing DFEM trial space degree, so called P refinement, on a fixed mesh of 320 spatial cells, we examine how the SLXS Lobatto solution profile changes with different choice in P for the radiation profile in Fig. 6.43 and temperature profile in Fig. 6.44, when near the Marshak wavefront.

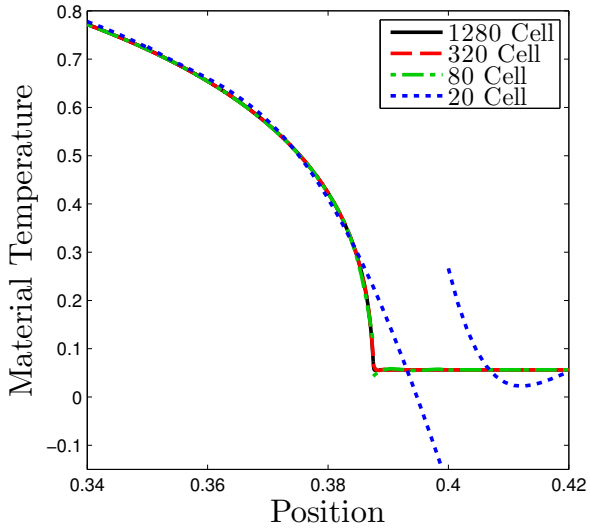

Figure 6.42: Quartic SLXS Gauss temperature solution near wavefront with increasing spatial mesh refinement.

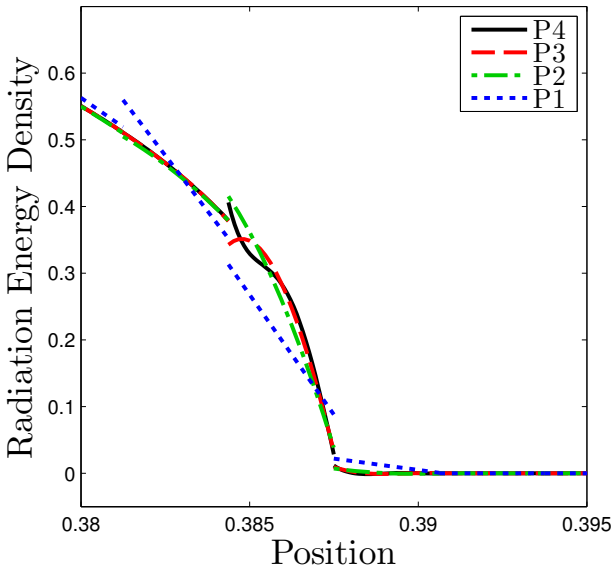

Figure 6.43: SLXS Lobatto radiation energy density solution with 320 cells for different P near Marshak wavefront.

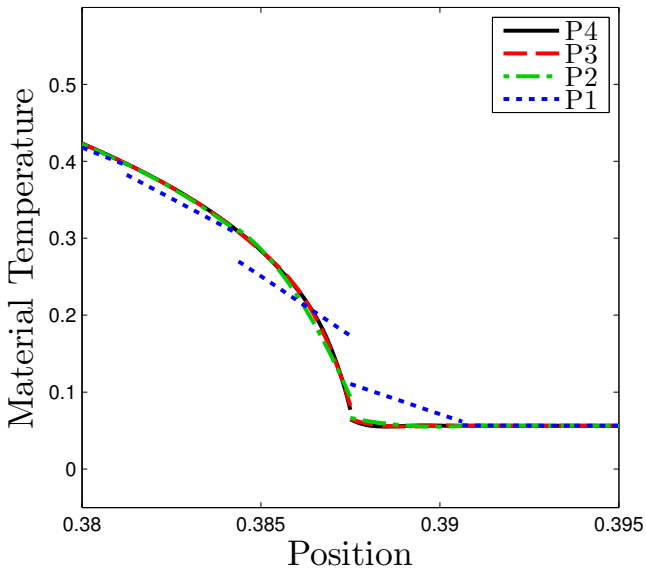

Figure 6.44: SLXS Lobatto material temperature solution with 320 cells for different P near Marshak wavefront.

Finally, we consider the effect of time step refinement in Fig. 6.45 for the radiation profile and in Fig. 6.46. Both Fig. 6.45 and Fig. 6.46 use a quartic SLXS Gauss spatial discretization with 1280 cells. The effects of decreasing time step size are non-trivial near the wavefront. As seen in Fig. 6.45, at lower time resolutions (Δt and $\frac{\Delta t}{4}$, the wavefront is visibly not uniformly concave down, resulting in several "wiggles" in the radiation profile in the heated region of the slab. Additionally, increased temporal resolution causes the discontinuity at the leading edge of the wavefront to sharpen. In Fig. 6.46, the effects of increased time resolution are the same as in Fig. 6.45. However, increased time resolution more noticeably sharpens the discontinuity in the temperature profile than it eliminates wavefront wiggles, though in Fig. 6.46 the Δt curve is not strictly concave down in the heated region of the slab near the wavefront.

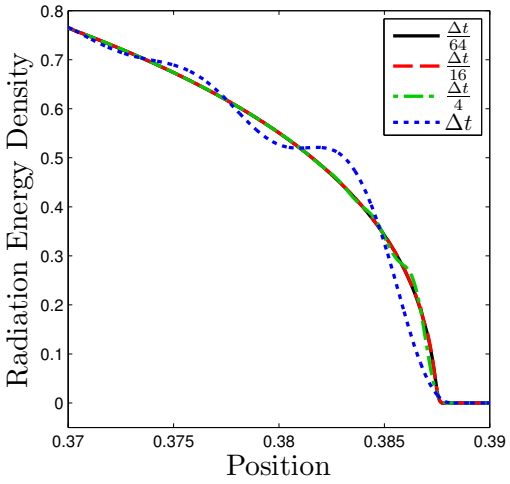

Figure 6.45: Quartic SLXS Lobatto radiation energy density solution with 1280 cells for different time refinements near the Marshak wavefront.

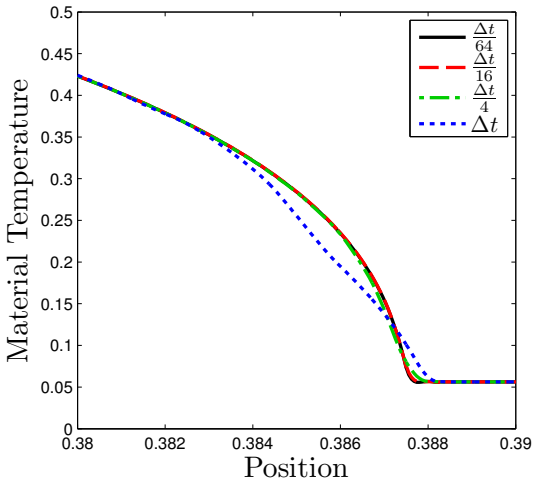

Figure 6.46: Quartic SLXS Lobatto temperature solution with 1280 cells for different time refinements near the Marshak wavefront.

We conclude our discussion of the Marshak wave problem by considering the results of a very fine spatial mesh calculation. Our hope is that with sufficient spatial resolution and higher order DFEM, we are able to resolve transport boundary layers. Our highest resolution simulation uses ten thousand spatial mesh cells. Given the initial, cold temperature of the slab is roughly $T = 0.056$, $\sigma_t = \sigma_a = \frac{1}{T^3}$, then the total slab optical thickness is roughly 5700 MFP thick, and when using ten though cells, each mesh cell is roughly 0.57 MFP thick. As noted by Larsen, Morel, and Miller [55], this type of mesh spacing is neither optically thick nor thin. To answer whether ten thousand mesh cells is sufficient, we first consider Fig. 6.47 where we compare the results of cubic SLXS Lobatto schemes that use

1. ten thousand spatial cells, with ten thousand time steps and the 3-3 SDIRK scheme,

2. ten thousand spatial cells, with one thousand time steps and the 3-3 SDIRK scheme, and

3. twelve-hundred spatial cells, with six thousand time steps of the 3-3 SDIRK scheme.

Missing' segments in Fig. 6.47 are caused by negative radiation energy densities. The twelve-hundred cell simulation has only 1 negative node. The ten thousand cell simulation that uses one thousand time steps has a total of 8 negative nodes; one entire cell has a negative radiation energy density, and two other cells have at least one node with a negative radiation energy density. Since the ten thousand cell simulation with ten thousand time steps does not have any negative radiation energy densities, it is clear then that temporal resolution is also required. Unfortunately, noting that the radiation energy density jumps four orders of magnitude for $x \in$

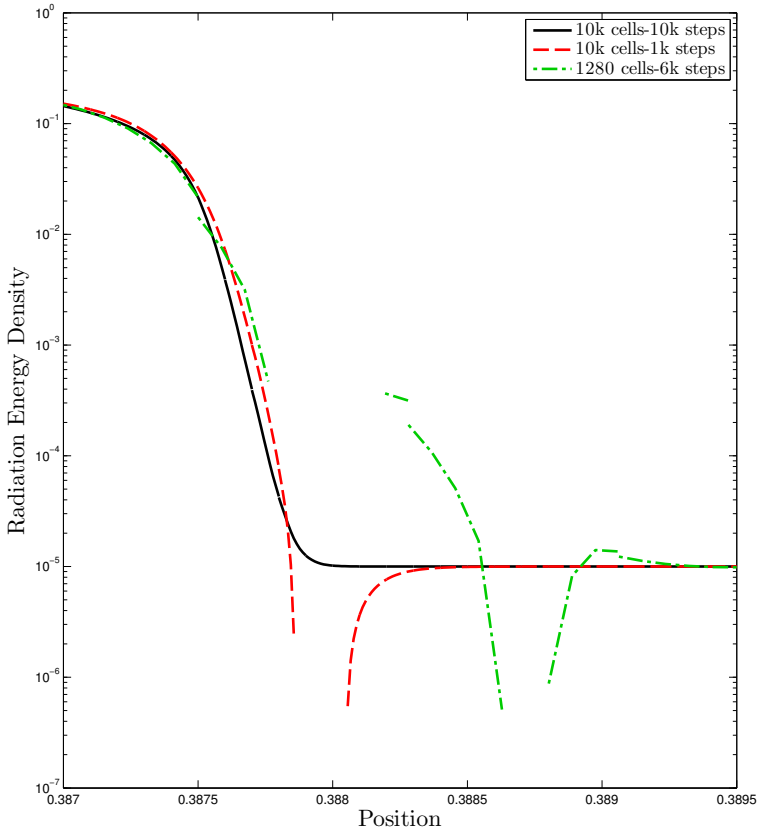

Figure 6.47: Plot of the radiation energy density on a logarithmic scale for different high resolution simulations near the Marshak wavefront.

[0.3870.388], our highest resolution run only had a total of ten cells available to resolve the transport boundary layer. In Fig. 6.48, we plot the angular intensities for $\mu = \pm\frac{1}{\sqrt{3}}$ of the ten thousand cell, ten thousand time step simulation. Clearly the

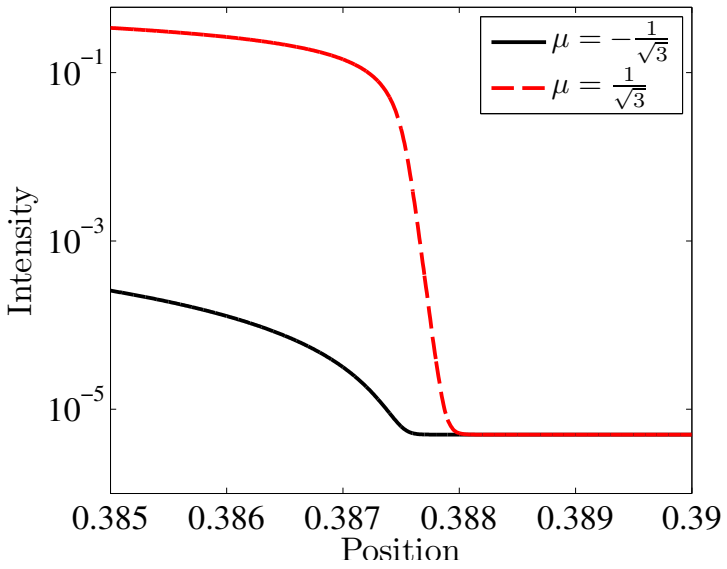

Figure 6.48: Plot of the angular intensity on a logarithmic scale near the transport boundary layer at the Marshak wave front.

radiation traveling from the hot to cold region, $\mu = \frac{1}{\sqrt{3}}$ has the largest boundary layer, but even with less than ideal spatial resolution, the rapid rise in angular intensity appears to be smooth, suggesting we have resolved the radiation boundary layer.

6.4.3 Su-Olson Problem

The Su-Olson problem [56] is an analytic benchmark that consists of an initially cold (initial radiation energy density and temperature conditions are identically zero)

half-space slab, heated for a finite amount of time by a volumetric, isotropic radiation source. The slab's scattering opacity is constant in space and temperature, absorption opacity is constant in space and temperature, and $C_v \propto T^3$. Assuming $C_v = \alpha T^3$ is critical; as Long, et al. [57] noted, the assumption regarding C_v is not physical, but is required to make the thermal radiative transfer equations linear in I and T^4, or conversely linear in I and material energy density. After a series of transformations, Su and Olson derived an analytic solution to the thermal radiative transfer equations. The solution is more accurately described as being semi-analytic. While the radiation energy density and material temperature at every point can be expressed as a closed form integral, evaluation of each integral requires numerical estimation. Further, the integral is a 2-D, indefinite integral (in both variables) of a trigonometric function with a slowly decaying exponential argument. However, [56] provides several radiation energy density and material energy density points in space, and is beneficial as a benchmark problem to verify the physics of a given radiative transfer implementation.

Given the initial temperature condition is explicitly zero, this implies the initial C_v is also zero. This is problematic when solving explicitly for temperature and not material energy. A near-zero heat capacity would result in the material rapidly heating, but a zero heat capacity implies a material that cannot accept heat, and thus can never be heated up. To prevent this problem, we modify the definition of C_v:

$$C_v = 10^{-8} + \alpha T^3 \,. \tag{6.154}$$

Alternatively, we could set an initial temperature to be a non absolute zero value.

Since [56] presents results in a non-dimensional format, we elect to define $a = c = 1$, $\sigma_a = 1$, $\sigma_s = 0$, $\alpha = 4$, truncate the full half space to $x \in [0, 5]$, and define

reference temperature $T_H = 1$. We solve the problem using 200 spatial mesh cells, linear SLXS Lobatto DFEM, the Alexander 2-2 time differencing scheme, an initial time step size of $\Delta t = 10^{-5}$, and increase the time step size by a factor of 1.1 until a maximum time step size of $\Delta t = 10^{-3}$ is reached.

In Fig. 6.49 we present the radiation energy density solution, $W(x)$ (in the notation on [56]), for S_2 angular differencing plotted against the analytic diffusion and transport solutions. Likewise, in Fig. 6.50, we plot the material energy density, $V(x)$ (in the notation on [56]) for S_2 angular differencing. Solutions at various times, $\tau = 1$ and $\tau = 10$ are given in both plots. As expected, the S_2 solution is nearly identical to diffusion, but skews slightly in the direction of the full transport solution.

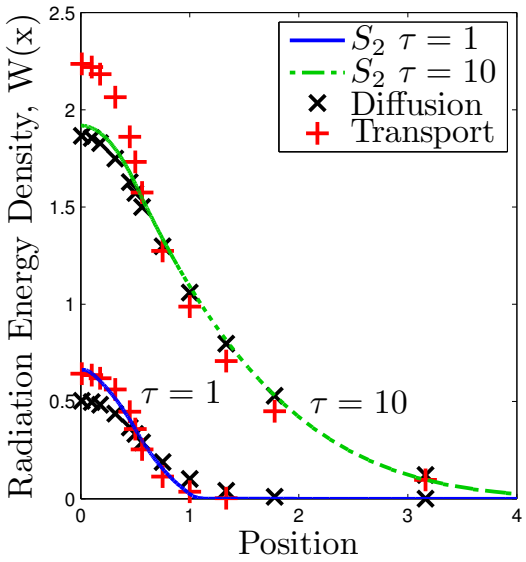

Figure 6.49: S_2 radiation energy density profile for Su-Olson problem.

Increasing the number of discrete ordinates, the radiation energy density profile and material energy density profiles are given in Fig. 6.51 and Fig. 6.52 for S_8 angular differencing. By adding a few discrete directions to the quadrature set, our numerical solution becomes indistinguishable from published results of [56], and we conclude that our TRT equation implementation is valid.

6.4.4 Effectiveness of MIP DSA for Radiative Transfer

As of yet, we have failed to have any discussion of the iterative performance of MIP DSA applied to the grey thermal radiative transfer equations. Though the problems we have considered are not necessarily optically thick, we have used MIP DSA to solve all problems. A large number of the problems we have considered are not optically thick, in part because we were interested in spatial error convergence. In

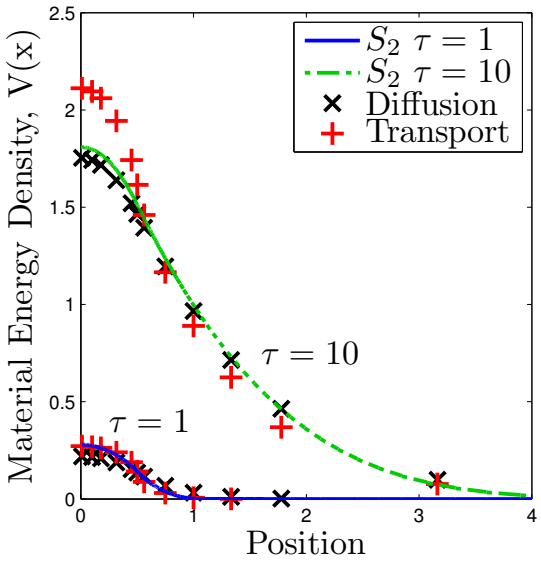

Figure 6.50: S_2 material energy density profile for Su-Olson problem.

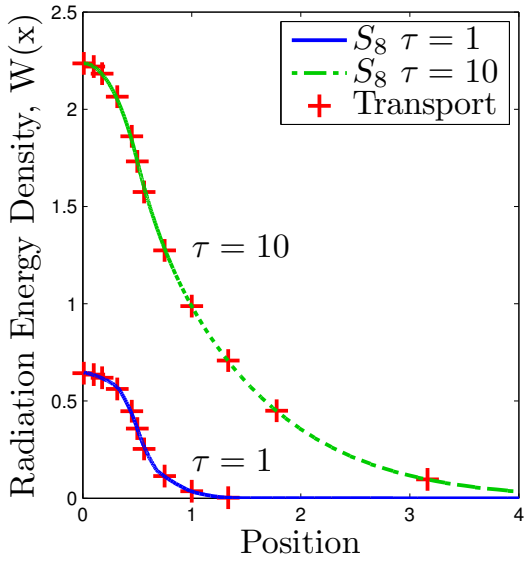

Figure 6.51: S_8 radiation energy density profile for Su-Olson problem.

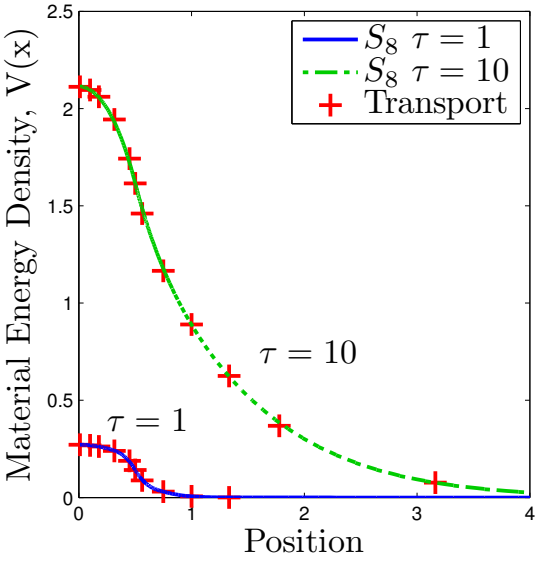

Figure 6.52: S_8 material energy density profile for Su-Olson problem.

Table 6.1, we give a sampling of the average number of iterations required to update the intensity for a given thermal iteration for the problems we have considered thus far.

Several observations can be made regarding the data in Table 6.1. Most importantly, MIP DSA applied to the grey TRT is a stable iterative scheme and at worst requires as many iterations as source iteration alone. Also, the number of iterations for MIP DSA and SI are nearly equal only for most of the problems we have considered, but for those where SI requires larger number of iterations, the ratio of SI+DSA iterations to SI alone iterations grows. Finally, MIP DSA is compatible with the self-lumping DFEM schemes we have developed that explicitly account for the within cell variation of opacity and heat capacity.

To demonstrate the iterative effectiveness of MIP DSA we now present a problem designed solely to be optically thick. We again define a dimensionless problem, $a = c = 1$. We assume a constant $C_v = 0.05$, define $x \in [0, 100]$, $t \in [0, 5]$, $\sigma_s = 0$, and $\sigma_a = \frac{5000}{T^2}$. Initially, the slab is in thermodynamic equilibrium at a temperature of $T = 0.5$, and is heated with an incident current of 100 on the left hand side of the slab. We difference the problem with linear SLXS Lobatto using 50 spatial cells, implicit Euler in time differencing, and a maximum time step size of $\Delta t_{max} = 0.1$. The average number of transport iterations per thermal iteration is given in Table 6.2. Clearly, MIP DSA can significantly reduce the iterative work required to solve the grey TRT equations, but the majority of problems we have considered are not very optically thick.

Table 6.1: Iteration count for different TRT model problems.

Problem Description	Scheme	Average DSA+SI Iterations	Average SI Iterations
MMS Constant Time 4 cells	Linear SLXS Lobatto	1.4	2.4
MMS Constant Time 8 cells	Cubic SLXS Lobatto	1.6	2.3
MMS Constant Time 128 cells	Cubic SLXS Lobatto	1.8	1.8
MMS1 2 cells	Quadratic SLXS Gauss	2.0	13.5
MMS1 32 cells	Quadratic SLXS Gauss	3.0	13.6
MMS1 128 cells	Quadratic SLXS Gauss	4.0	13.5
MMS1 256 cells	Quadratic SLXS Gauss	4.2	13.5
MMS2 2 cells	Linear SLXS Gauss	1.0	2.7
MMS Constant Space Alexander 3-3, $\Delta t = 1$	Quartic SLXS Lobatto	17.0	39.0
MMS Constant Space Alexander 3-3, $\Delta t = \frac{1}{8}$	Quartic SLXS Lobatto	6.6	11.7
MMS Constant Space Alexander 3-3, $\Delta t = \frac{1}{128}$	Quartic SLXS Lobatto	2.3	4.9
Marshak Wave 20 cells, largest Δt	Linear SLXS Lobatto	2.1	2.9

Table 6.2: Iteration count for a very optically thick TRT problem.

Intensity Iterative Strategy	Average Intensity Iterations Per Thermal Iteration
SI+DSA	2.5
SI	4460.7

7. SUMMARY AND CONCLUSIONS

7.1 Summary

In this dissertation we have developed a new family of interpolatory DFEM spatial discretizations for discrete ordinates radiation transport. This set of DFEM schemes is unique in that it is not limited to equally-spaced DFEM interpolation points, automatically generates diagonal mass matrices, can be used with DFEM trial spaces of arbitrary degree, and can explicitly account for the within cell variation of material properties such as interaction cross section. We have tested this new family of DFEM techniques in a variety of slab geometry radiation transport applications including steady-state neutron transport, criticality, and time dependent thermal radiative transfer simulations.

Additionally, we have demonstrated that some canonical methods of discrete ordinates radiation transport do not always behave as expected. First, we showed that the traditional method of mass matrix lumping limits solution accuracy for higher degree polynomial trial spaces and is robust with a linear DFEM trial space. Later, we demonstrated that the usual assumption of a cell-wise constant interaction cross section has several negative effects in problems with spatially varying interaction cross sections. Across all application areas considered, the assumption of a cell-wise constant cross section for problems that had cross section variation within individual mesh cells resulted in a fundamental limit on spatial order of convergence, and generated non-smooth interaction rates. The non-smooth neutron transport interaction rates manifested themselves in our thermal radiative transfer results as a material temperature solution that contained large, non-monotonic discontinuities. Mesh refinement can reduce the severity of the thermal radiative transfer tempera-

ture solution discontinuities, but cannot eliminate the discontinuities. Not only are the discontinuities non-physical and a sign of limited spatial order of convergence, they also complicate and can inhibit the non-linear temperature iteration required to solve the thermal radiative transfer equations.

7.2 Conclusions

There are two main conclusions to be made from this dissertation.

1. Self-lumping DFEM schemes using Gauss-Legendre (SL Gauss) or Gauss-Lobatto-Legendre (SL Lobatto) quadrature as the DFEM interpolation points are well suited to discrete ordinates radiation transport calculations.

2. Self-lumping schemes are easily modified to explicitly account for the within cell variation of material properties, resulting in methods that are significantly more accurate for problems with spatially varying material properties than those that assume cell-wise constant material properties.

In neutron transport, criticality, and thermal radiative transfer simulations, both SL Gauss and SL Lobatto converge the L^2 norm of the angular flux error $\propto P+1$ for problems with cell-wise constant cross section. SL Lobatto is robust for all odd degree DFEM trial spaces, and SL Gauss is robust for all even degree trial spaces, assuming cell-wise constant cross section. Further, SL Lobatto is equivalent to traditional lumping for linear DFEM, but unlike traditional lumping DFEM schemes, SL Lobatto increases in spatial order of convergence with increased trial space degree.

Self-lumping DFEM schemes easily account for the variation of interaction cross or other material properties. A P degree self-lumping scheme using Gauss or Lobatto quadrature only requires $P+1$ material property evaluations to obtain schemes that converge the L^2 norm of the angular flux or radiation intensity $\propto P+1$. This is

in contrast to DFEM schemes that assume a cell-wise constant cross yielding only second order spatial convergence, regardless of trial space degree, for problems with spatially varying interaction cross sections. In our neutron transport test problems, SLXS Lobatto and SLXS Gauss (the variants of SL Lobatto and SL Gauss that explicitly account for within cell variation of material properties), converged interaction rate and interaction rate dependent quantities $\propto toP + 1$. However, in our thermal radiative transfer MMS testing SLXS Lobatto converged the L^2 error of temperature, a quantity driven by an interaction rate, $\propto P$. Though not $P + 1$ as we had hoped, given the non-linear nature of the thermal radiative transfer equations, and necessity to integrate much higher order polynomials, e.g. the Planckian term that is a P^4 degree polynomial, order P convergence is still promising, as it still allows for increased accuracy with increasing DFEM trial space degree. More surprising is that SLXS Gauss applied to the grey TRT appears to converge the L^2 error of temperature $\propto P + 2$. It should be noted though that the orders of convergence we have given here are effectively only experimental observations, and there was some disagreement between the apparent orders of convergence for certain error quantities between test problems.

7.2.1 Future Work

There are several exciting avenues for continued study and advancement if the topics and methods covered in this dissertation. Clearly the extension of this slab geometry work to multiple spatial dimensions is required for problems of greater scientific and engineering interest and complexity. Additionally, MIP DSA appeared to be an effective iterative acceleration technique for the grey TRT equations, and as such we would like to see how it performs as the diffusion operator for linear multi-frequency grey acceleration of the multi-group/multi-frequency thermal radiative

transfer equations.

Topics of research beyond simply extending our methodology are abundant as well. A partial listing includes:

1. developing a theory to explain and apriori predict whether a given matrix lumping technique will yield a robust solution,

2. explaining the apparent super convergence of SLXS Gauss for the TRT temperature solution,

3. conducting a diffusion limit analysis of higher order trial space DFEM, and

4. finding more TRT manufactured solutions that challenge spatial discretization more completely and are closer to real-world applications in both nature and scaling.

REFERENCES

[1] E. E. Lewis and W. F. Miller. *Computational Methods of Neutron Transport*. American Nuclear Society, La Grange Park, IL, 1993.

[2] R. Alexander. Diagonally implicit Runge-Kutta methods for stiff O.D.E.'s. *SIAM Journal of Numerical Analysis*, 14(6):1006–1021, 1977.

[3] W. H. Reed, T. R. Hill, F. W. Brinkley, and K. D. Lathrop. TRIPLET: A two-dimensional, multigroup, triangular mesh, planar geometry, explicit transport code. Technical Report LA-5428-MS, Los Alamos Scientific Lab, 1973.

[4] E. W. Larsen and P. Nelson. Finite-difference approximations and superconvergence for the discrete ordinates equations in slab geometry. *SIAM Journal of Numerical Analysis*, 19(2):334–348, 1982.

[5] E. W. Larsen and J. E. Morel. Asymptotic solutions of numerical transport problems in optically thick, diffusive regimes II. *Journal of Computational Physics*, 83:212–236, 1989.

[6] J. E. Morel, T. A. Wareing, and K. Smith. A linear-discontinuous spatial differencing scheme for S_N radiative transfer calculations. *Journal of Computational Physics*, 128:445–462, 1996.

[7] M. L. Adams. Subcell balance methods for radiative transfer on arbitrary grids. *Transport Theory and Statistical Physics*, 26(4 &5):385–431, 1997.

[8] M. L. Adams and P. F. Nowak. Asymptotic analysis of a computational method for time- and frequency- dependent radiative transfer. *Journal of Computational Physics*, 46:366–403, 1998.

[9] C. C. Ober and J. N. Shadid. Studies on the accuracy of time-integration methods for the radiation-diffusion equations. *Journal of Computational Physics*, 195:743–772, 2004.

[10] P. G. Maginot, J. C. Ragusa, and J. E. Morel. Characterization of high order spatial discretizations and lumping techniques for discontinuous finite element S_N transport. In *International Conference on Mathematics and Computational methods, Applied to Nuclear Science and Engineering*, Sun Valley, Idaho, May 2013.

[11] P. G. Maginot, J. C. Ragusa, and J. E. Morel. Lumping techniques for DFEM transport in S_N transport in slab geometry. *Nuclear Science and Engineering*, 179(2):148–163, 2015.

[12] E. W. Larsen and W. F. Miller. Convergence rates of spatial difference equations for the discrete-ordinates neutron transport equations in slab geometry. *Nuclear Science and Engineering*, 73:76–83, 1980.

[13] S. Hamilton, M. Benzi, and J. S. Warsa. Negative flux fixups in discontinuous finite elements S_N transport. In *International Conference on Mathematics, Computational Methods & Reactor Physics*, Saratoga Springs, New York, May 2009.

[14] P. G. Maginot, J. E. Morel, and J. C. Ragusa. A non-negative moment preserving spatial discretization scheme for the linearized boltzmann transport equation in 1-D and 2-D cartesian geometries. *Journal of Computational Physics*, 231(20):6801–6826, 2012.

[15] M. L. Adams. Discontinuous finite element transport solutions in thick diffusive problems. *Nuclear Science and Engineering*, 137:298–333, 2001.

[16] W. F. Walters. The relation between finite element methods and nodal methods in transport theory. *Progress in Nuclear Energy*, 18:21–26, 1986.

[17] J. P. Hennart and E. del Valle. A generalize nodal finite element formalism for discrete ordinate equations in slab geometry: Part II theory in the discontinuous moment case. *Transport Theory and Statistical Physics*, 24:479–504, 1995.

[18] J. P. Hennart and E. del Valle. A generalize nodal finite element formalism for discrete ordinate equations in slab geometry: Part III numerical results. *Transport Theory and Statistical Physics*, 24:505–533, 1995.

[19] J. S. Warsa and A. K. Prinja. p-adaptive numerical methods for particle transport. *Transport Theory and Statistical Physics*, 28(3):229–270, 1999.

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

[21] Y. Wang and J. C. Ragusa. A high-order discontinuous Galerkin method for the S_N transport equations on 2D unstructured triangular meshes. *Annals of Nuclear Energy*, 36(7):931–939, 2009.

[22] K. D. Lathrop. Spatial differencing of the transport equation: Positivity vs. accuracy. *Journal of Computational Physics*, 4:475–498, 1969.

[23] P. A. Raviart. The use of numerical integration in finite element methods for solving parabolic equations. In *Conference on Numerical Analysis, RIANA 1972*, pages 233–264, August 1972.

[24] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. United States Department of Commerce, Washington, D.C., 1972.

[25] V. Thomee. *Galerkin Finite Element Methods for Parabolic Problems*. Springer, New York, 1997.

[26] The MathWorks. MATLAB 2011b, 2011.

[27] W. M. Stacey. *Nuclear Reactor Physics*. John-Wiley & Sons Inc., 2001.

[28] J. E. Morel, T.-Y. B. Yang, and J. S. Warsa. Linear multifrequency-grey acceleration recast for preconditioned Krylov iterations. *Journal of Computational Physics*, 227:244–264, 2007.

[29] 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. *Nuclear Science and Engineering*, 147(3):218–248, 2004.

[30] A. Kavenoky and J. Lautard. A finite element depletion diffusion calculation method with space-dependent cross section. *Nuclear Science and Engineering*, 64(2):563–575, 1977.

[31] S. Santandrea and P. Bellier. An unstructured characteristics scheme with a linear expansion for both fluxes and cross sections. In *Proceedings of the Joint International Topical Meeting on Mathematics & Computation and Supercomputing in Nuclear Applications (M&C + SNA 2007)*, Monterey, California, April 2007.

[32] P. G. Maginot, J. C. Ragusa, and J. E. Morel. Accurate methods for dfem S_N transport in slab geometry for non-piecewise constant cross section problems. *Annals of Nuclear Energy*, 73:506–526, 2014.

[33] G. Bell and S. Glasstone. *Nuclear Reactor Theory*. Van Nostrand Rienhold, Inc., New York, NY, 1970.

[34] L. J. Lorence, J. E. Morel, and E. W. Larsen. An S_2 synthetic accleration scheme for the one-dimensional S_N equations with linear discontinuous spatial differencing. *Nuclear Science and Engineering*, 101:341–351, 1989.

[35] Y. Wang and J. C. Ragusa. Diffusion synthetic acceleration for high-order discontinuous finite element S_N transport schemes and application to locally refined unstructured meshes. *Nuclear Science and Engineering*, 166:145–166, 2010.

[36] E. W. Larsen. Unconditionally stable diffusion-synthetic acceleration methods for the slab geometry discrete ordinates equations. *Nuclear Science and Engineering*, 82:47–63, 1982.

[37] E. M. Gelbard and L. A. Hageman. Synthetic methods as applied to S_N equations. *Nuclear Science and Engineering*, 37(2):288–298, 1969.

[38] M. L. Adams and W. R. Martin. Diffusion synthetic acceleration of discontinuous finite element transport iterations. *Nuclear Science and Engineering*, 111:145–167, 1992.

[39] J. S. Warsa, T. A. Wareing, and J. E. Morel. Fully consistent diffusion synthetics acceleration of linear discontinuous S_N transport discretization on unstructured tetrahedral meshes. *Nuclear Science and Engineering*, 141:236–251, 2002.

[40] T. A. Wareing, E. W. Larsen, and M. L. Adams. Diffusion accelerated discontinuous finite element schemes for the S_N equations in slab and X-Y geometries. In *Advances in Mathematics, Computations, and Reactor Physics*, Pittsburgh, Pennsylvania, April 28-May 2 1991.

[41] B. Turcksin and J. C. Ragusa. A diffusion synthetic acceleration scheme for rectangular geometries based on bilinear discontinuous finite elements. In *In-*

ternational Conference on Mathematics and Computational methods, Applied to Nuclear Science and Engineering, Sun Vlley, Idaho, May 2013.

[42] Y. Wang. *Adaptive Mesh Refinement Solution Techniques for the Multigroup S_N Transport Equations Using a Higher-Order Discontinuous Finite Element Method*. PhD thesis, Texas A&M University, 2009.

[43] J. S. Warsa, T. A. Wareing, J. E. Morel, J. M. McGhee, and R. B. Lehoucq. Krylov subspace iterations for deterministic k-eigenvalue calculations. *Nuclear Science and Engineering*, 147:26–42, 2004.

[44] E. W. Larsen, A. Kumar, and J. E. Morel. Properties of the implicitly time-differenced equations of thermal radiation transport. *Journal of Computational Physics*, 238:82–96, 2013.

[45] B. Stroustrup. *Programming: Principles and Practice Using C++*. Pearson Education Inc., Upper Saddle River, New Jersey, 2nd edition, 2014.

[46] S. Meyers. *Effective C++*. Pearson Education Inc., Upper Saddle River, New Jersey, 3rd edition, 2005.

[47] J. Burkhardt. QUADRULE. `http://people.sc.fsu.edu/~jburkardt/cpp_src/quadrule/quadrule.html`, 2011.

[48] Gaël Guennebaud, Benoît Jacob, et al. Eigen v3. `http://eigen.tuxfamily.org`, 2010.

[49] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, and H. Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 3.5, Argonne National Laboratory, 2014.

[50] V.E. Henson and U.M. Yang. BoomerAMG: a parallel algebraic multigrid solver and preconditioner. *Applied Numerical Mathematics*, 41:155–177, 2002.

[51] D. van Hessch. Doxygen manual. `http://www.stack.nl/~dimitri/doxygen/index.html`, 2015.

[52] Kitware. CMake, a cross-platform open-source build system. `http://www.cmake.org`, 2003.

[53] K. Salari and P. Knupp. Code verification by the method of manufactured solutions. Technical Report SAND2000-1444, Sandia National Labs, 2000.

[54] L. Thomason and A. Ellerton Y. Berquin. TinyXml. `www.sourceforge.net/projects/tinyxml`, 2010. v2.6.0.

[55] E. W. Larsen, J. E. Morel, and W. F. Miller. Asymptotic solutions of numerical transport problems in optically thick, diffusive regimes. *Journal of Computational Physics*, 69:283–324, 1987.

[56] B. Su and G. L. Olson. An analytical benchmark for non-equilibrium radiative transfer in an isotropically scattering medium. *Annals of Nuclear Engergy*, 24(13):1035–1055, 1997.

[57] A. R. Long, N. A. Gentile, and T. S. Palmer. The iterative thermal emission method: A more implicit modification of IMC. *Journal of Computational Physics*, 277:228–247, 2014.