ABSTRACT

WIESELQUIST, WILLIAM A. The Quasidiffusion Method for Transport Problems on Unstructured Meshes. (Under the direction of Dmitriy Y. Anistratov.)

In this work, we develop a quasidiffusion (QD) method for solving radiation transport problems on unstructured quadrilateral meshes in 2D Cartesian geometry, for example hanging-node meshes from adaptive mesh refinement (AMR) applications or skewed quadrilateral meshes from radiation hydrodynamics with Lagrangian meshing. The main result of the work is a new low-order quasidiffusion (LOQD) discretization on arbitrary quadrilaterals and a strategy for the efficient iterative solution which uses Krylov methods and incomplete LU factorization (ILU) preconditioning. The LOQD equations are a non-symmetric set of first-order PDEs that in second-order form resembles convection-diffusion with a diffusion tensor, with the difference that the LOQD equations contain extra cross-derivative terms. Our finite volume (FV) discretization of the LOQD equations is compared with three LOQD discretizations from literature.

We then present a conservative, short characteristics discretization based on subcell balances (SCSB) that uses polynomial exponential moments to achieve robust behavior in various limits (e.g. small cells and voids) and is second-order accurate in space. A linear representation of the isotropic component of the scattering source based on face-average and cell-average scalar fluxes is also proposed and shown to be effective in some problems.

In numerical tests, our QD method with linear scattering source representation shows some advantages compared to other transport methods. We conclude with avenues for future research and note that this QD method may easily be extended to arbitrary meshes in 3D Cartesian geometry.

The Quasidiffusion Method for Transport Problems on Unstructured Meshes

by
William A. Wieselquist

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Nuclear Engineering

Raleigh, North Carolina

2009

APPROVED BY:

Dr. Paul J. Turinsky	Dr. Yousry Y. Azmy
Dr. Robin P. Gardner	Dr. Semyon V. Tsynkov

Dr. Dmitriy Y. Anistratov
Chair of Advisory Committee

DEDICATION

I would like to dedicate this to my family, both new and old, for their encouragement and support during these years. In particular, my brother Andrew for always being there to distract me from my problems and my best friend Elizabeth (whom I finally married in 2006) for always being there to help me through them.

"Prediction is difficult, especially the future."

- Niels Bohr

BIOGRAPHY

I, William Adam Wieselquist, was born in 1979 to Roy Walter Wieselquist and Katherine Dashiell Rouse Wieselquist in Washington, D.C. I grew up in central North Carolina, with brother, Andrew Greco, born in 1982, and sister, Grace Dashiell, born in 1985. I attended middle school and high school in Winston-Salem, NC, with a brief stint at the North Carolina School of Science and Mathematics in 2001 in Durham, NC. In 1998, I began college at North Carolina State University in Raleigh, NC, graduating in 2002 with a B.S. in nuclear engineering, followed by an M.S. in 2005.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor, Dr. Dmitriy Y. Anistratov, for the years of guidance and patience as I have grown both as a scientist and as a person. I would also like to thank other faculty in the nuclear engineering department at North Carolina State University, with whom I have spent the last 10 years, namely R.M. Mayo, J.M. Doster, P.J. Turinsky, and R.P. Gardner, from whom I have learned many things, both in and outside the classroom.

I would like to thank Richard M. Schultz for my first national lab experience at Idaho National Laboratory, Michael Todosow for my experience with Brookhaven National Laboratory, and finally Jae H. Chang for my experience at Los Alamos National Laboratory.

This work was partially supported by the Nuclear Engineering Education and Research (NEER) Program of the US Department of Energy under the grant No. DE-FG07-03ID14496.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

In this research we present a new method for solving radiation transport problems using the quasidiffusion (QD) method [4] on arbitrary spatial meshes of quadrilaterals in Cartesian XY geometry. In this introduction, we discuss the motivation for radiation transport simulations, give some necessary background information on radiation transport while reviewing the relevant literature, and then briefly outline our advancements.

1.1 Motivation

Simulation of physics on digital computers has become an integral part of the design process, especially in the preliminary stages (e.g. scoping studies), when the numerous degrees of freedom make performing actual experiments prohibitively expensive. Simulation of radiation transport is no exception, playing a vital role in many natural and artificial processes. Radiation transport simulation is a valuable tool in: nuclear reactor design, radiation safety, astrophysics, medical imaging, radiotherapy, fuel transport/storage, shielding design, oil exploration, fire simu-

lation, combustion engineering, and electronics.

1.1.1 When is full transport warranted?

There are many physical processes in which modelling radiation and it's interaction with matter is simple. For example, consider photon interactions in a system at local thermodynamic equilibrium (LTE). In LTE, the specific radiation intensity (so-called angular flux ψ) is a simple function of material temperature, $\psi \propto T^4$ [28]. As another example, consider neutron attenuation in a purely absorbing shield, where the mean radiation intensity (so-called scalar flux ϕ) obeys an exponential decay law, $\phi(x) = \phi_0 e^{-\sigma_a x}$, where σ_a is the macroscopic absorption cross-section. As a final example, consider nuclear reactors where the largeness of the reactor system, static geometry, and homogenization methods allow the system to be modelled using a few-group diffusion approximation.

However, it is easy to think of situations where the above assumptions are not valid, and thus the approximation most certainly isn't—e.g. a system which has not reached LTE, a shielding problem in highly scattering media, or a small modular reactor. In these cases and many more, one must resort to transport methods capable of capturing all physics of radiation transport. Traditionally these "full" transport methods are categorized as either *deterministic* or *stochastic*.

In this work, we concern ourselves with deterministic transport methods, [1] which center on the discretization and solution of the linearized Boltzmann transport equation [5], an integro-differential equation that is integral in angle ($\vec{\Omega}$) and energy (E) and differential in space ($\vec{r}$) and

[1] Specifically, we concern ourselves with the first-order form of the transport equation as in Eq. (1.1). Second-order forms exist and have the advantage of being self-adjoint, therefore amenable to SPD discretizations and fast iterative solution via conjugate gradients (CG). They also have an advantage in developing consistently discretized synthetic acceleration equations (see Sec.1.3.1) because the second-order forms have a natural similarity to diffusion. However, second-forms suffer from problems with void regions where $\sigma_t \to 0$ and cannot take advantage of the efficient transport sweep [88], as described in Sec.1.3.1.1.

time (t),

$$\frac{1}{v}\frac{\partial\psi}{\partial t} + \vec{\Omega}\cdot\vec{\nabla}\psi + \sigma_t\psi = \int_{4\pi} d\Omega' \int_0^\infty dE'\sigma_s(\vec{\Omega}'\rightarrow\vec{\Omega}, E'\rightarrow E)\psi(\vec{r},\vec{\Omega}', E', t) + Q_{ext}. \tag{1.1}$$

The primary unknown is the so-called "angular flux", $\psi(\vec{r},\vec{\Omega},E,t)$, a convenient grouping of particle speed $v(E)$ and particle distribution function $n(\vec{r},\vec{\Omega},E,t)$, i.e. $\psi = v\,n$. The total interaction cross section is $\sigma_t(\vec{r},E,t)$ and double differential scattering cross section is $\sigma_s(\vec{r},\vec{\Omega}'\rightarrow\vec{\Omega},E'\rightarrow E,t)$. The direction of particle travel is the unit vector $\vec{\Omega} = (\Omega_x,\Omega_y,\Omega_z)$, with $\Omega_x = \cos\gamma\sin\theta$, $\Omega_y = \sin\gamma\sin\theta$, and $\Omega_z = \cos\theta$, where $\gamma\in[0,2\pi]$ is the azimuthal angle and $\theta\in[-\pi/2,\pi/2]$ is the polar angle. Integration over an angular dependent quantity $f(\vec{\Omega})$ is represented in the shorthand,

$$\int_{4\pi} d\Omega\, f(\vec{\Omega}) = \int_0^{2\pi} d\gamma \int_{-\pi/2}^{\pi/2} \sin\theta d\theta f(\theta,\gamma). \tag{1.2}$$

Solving deterministic transport problems is difficult because of the high dimensionality of the problem (7D in the most general case), and because the particle transport phenomenon may exhibit different behaviors depending on properties of the physical system. To paraphrase from an excellent 2002 radiation transport review paper [88], "in void-like regions of a physical system, the transport equation behaves like a hyperbolic wave equation; in highly scattering regions it behaves like an elliptic (steady-state) or parabolic (time-dependent) diffusion equation; and in regions with highly forward-peaked scattering, it can behave like a parabolic equation." It is extremely difficult to find discretization methods that are accurate over this wide range of behavior.

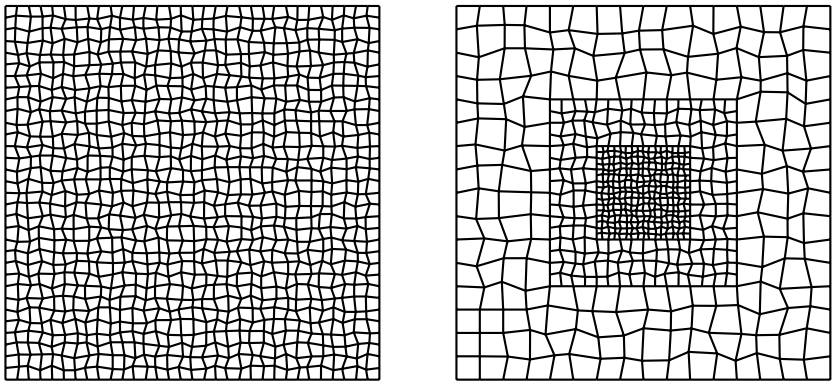
1.1.2 Why use quasidiffusion?

The quasidiffusion method (QD), developed by Gold'in in 1964 [4], is a method to solve Eq. (1.1), the transport equation. QD utilizes a special "low-order problem" (low-order in the sense

of angular dependence) which one derives by integrating the transport equation over all angle using Eq. (1.2), with respect to $f(\vec{\Omega}) = 1$, Ω_x, Ω_y, and Ω_z. Quasidiffusion has a number of qualities which will be illuminated in subsequent sections. In brief, those qualities are:

- **versatility** because the high-order (transport) and low-order quasidiffusion discretizations may be chosen independently of one another,
- **acceleration** of transport iterations in highly-scattering media and for nonlinear problems, and
- **simplified coupling** to other physics.

1.1.3 Why consider unstructured meshes?



(a) logically rectangular (b) hanging-node

Figure 1.1: Some arbitrary quadrilateral meshes.

Deterministic transport methods require a discretization of Eq. (1.1) in space, energy, angle, and time for the most general problem. In energy, angle, and time, there are standard and accepted discretizations which we shall assume so that we may focus on developing spatial

discretizations of the transport equation on arbitrary meshes. Typical reasons for solving transport problems on arbitrary meshes are to

- resolve **curved surfaces** and **material interfaces** with more accuracy,
- allow **adaptation** of meshes to minimize some aspects of the error, and
- **increase generality**, which is useful in multi-physics applications where the transport method cannot dictate the mesh.

We focus on arbitrary quadrilateral meshes as in Fig. 1.1. Both types of meshes have useful applications. Hanging-node meshes commonly arise in adaptive mesh refinement (AMR) applications [35], and although we do not perform such adaptation of the mesh to minimize the error (usually with respect to some local norm), our methods may be used on such meshes. Logically rectangular meshes are our other target mesh and can be described by perturbations of an orthogonal (rectangular) mesh that do not create any new cells (or concave cells.) These meshes are common in radiation hydrodynamics with a Lagrangian reference frame. Increased generality is also useful in radiation hydrodynamics because of it's multi-physics nature. Hydrodynamics dictates the mesh, and although the radiation transport could use a different mesh and map the necessary information to and from the hydrodynamics mesh, a more efficient solution is probably to have a transport component that can compute directly on the hydrodynamics mesh, and thus the multi-physics simulation need maintain only one spatial mesh. Our methods are also valid on arbitrary quadrilateral meshes created from

1. triangular meshes by dividing each triangle into 3 quadrilaterals and/or merging 2 neighboring triangles into one quadrilateral or

2. polygonal meshes by dividing each N-sided polygon into N quadrilaterals.

1.2 Preliminaries

In this section, we will present the Quasidiffusion (QD) equations in continuous space and then remark on aspects of QD for the reader that is not familiar with the method.

1.2.1 Formulation of the Quasidiffusion Method

The Quasidiffusion (QD) method [4] for steady-state one-group transport problems with isotropic scattering[2] and source in Cartesian geometry is defined by the following system of equations.

Transport Equation

$$\vec{\Omega} \cdot \vec{\nabla}\psi + \sigma_t \psi = \frac{\sigma_s}{4\pi}\phi + \frac{1}{4\pi} q_{ext} \qquad \text{in domain } G, \tag{1.3a}$$

$$\psi = \psi_{IN} \qquad \text{for incoming directions } \vec{\Omega} \cdot \vec{n} < 0 \text{ on boundary } \partial G, \tag{1.3b}$$

Low-Order Quasidiffusion (LOQD) Equations

$$\vec{\nabla} \cdot \vec{J} + (\sigma_t - \sigma_s)\phi = q_{ext} \qquad \text{in domain } G, \tag{1.4a}$$

$$\sum_{\beta=x,y,z} \frac{\partial}{\partial \beta}(\phi E_{\alpha\beta}) + \sigma_t J_\alpha = 0 \qquad \text{for } \alpha = x, y, z \text{ in domain } G, \tag{1.4b}$$

$$\vec{n} \cdot \vec{J} + J_{IN} = C(\phi - \phi_{IN}) \qquad \text{on boundary } \partial G, \tag{1.4c}$$

where G is the domain of the problem, ∂G is the boundary surface of G, $\vec{n}$ is the outward normal to ∂G, $\phi = \int_{4\pi} \psi \, d\Omega$ is the scalar flux, $\vec{J} = \int_{4\pi} \vec{\Omega}\psi \, d\Omega$ is the net current, $J_{IN} = \int_{\vec{n}\cdot\vec{\Omega}<0} |\vec{n} \cdot \vec{\Omega}| \psi_{IN} \, d\Omega$

[2] We only assume isotropic scattering for simplicity—the QD method is effective in problems with highly anisotropic scattering as well [65].

is the incoming partial current, $\phi_{IN} = \int_{\vec{n}\cdot\vec{\Omega}<0} \psi_{IN}\ d\Omega$, is the incoming partial scalar flux, $E_{\alpha\beta}$ is the $\alpha\beta$-th component of the QD (aka "Eddington") tensor, and C is the boundary QD factor. The nonlinear $E_{\alpha\beta}$ and C factors of the angular flux ψ are given as

$$E_{\alpha\beta}[\psi] = \frac{\int_{4\pi}\ \Omega_\alpha \Omega_\beta \psi\ d\Omega}{\int_{4\pi}\ \psi\ d\Omega}, \tag{1.5a}$$

$$C[\psi] = \frac{\int_{\vec{n}\cdot\vec{\Omega}>0}\ \vec{n}\cdot\vec{\Omega}\ \psi\ d\Omega}{\int_{\vec{n}\cdot\vec{\Omega}>0}\ \psi\ d\Omega}. \tag{1.5b}$$

The iteration scheme used to solve the QD equations of Eq. (1.3) and Eq. (1.4) is shown in Algorithm 1.

Algorithm 1: Quasidiffusion Iteration Process.

while *Convergence criteria not met* **do**

  if *first iteration* **then**

    Assume diffusion approximation for low-order factors: $E_{\alpha\beta} = \frac{1}{3}\delta_{\alpha\beta}$ and $C = \frac{1}{2}$.

  else

    Calculate low-order factors $E_{\alpha\beta}[\psi]$ and $C[\psi]$ via Eq. (1.5).

  end

  Solve the LOQD equations of Eq. (1.4) for ϕ and $\vec{J}$.

  Update the RHS scattering source of the transport equation (e.g. $\frac{1}{4\pi}\sigma_s\phi$.)

  Solve the transport equation of Eq. (1.3) for ψ.

end

1.2.1.1 Remarks about QD

The QD Factors, $E_{\alpha\beta}[\psi]$ and $C[\psi]$ depend only weakly on the angular flux ψ, possessing a small Frechet derivative [47]. This leads to fast convergence for a wide range of linear (and nonlinear) transport problems. The LOQD equations of Eq. (1.4) are a non-symmetric, elliptic

system with a tensor divergence term. It is convenient to write Eq. (1.4b) in tensor form,

$$\vec{\nabla} \cdot (\phi \mathrm{E}) + \sigma_t \vec{J} = 0. \tag{1.6}$$

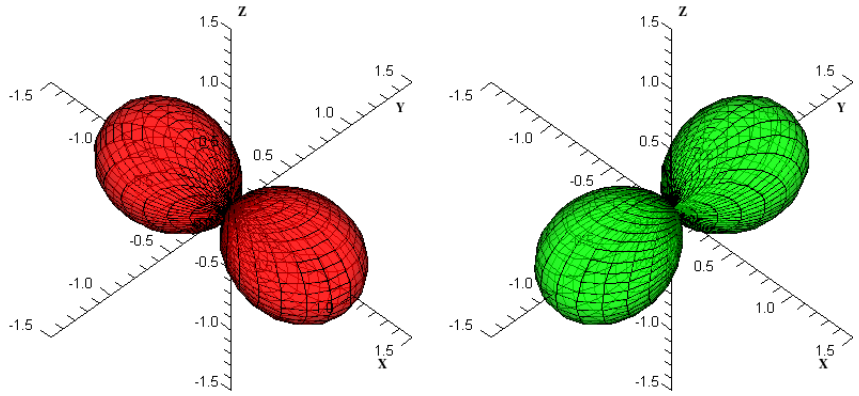
Note that the QD tensor E appears with the scalar flux ϕ under the divergence operator. (This makes discretization of the LOQD equations much more difficult than if E was outside.) The QD (aka Eddington) tensor is a symmetric, positive definite, rank-2 tensor that may be visualized as the matrix,

$$\mathrm{E} = \begin{pmatrix} E_{xx} & E_{xy} & E_{xz} \\ E_{xy} & E_{yy} & E_{yz} \\ E_{xz} & E_{yz} & E_{zz} \end{pmatrix}. \tag{1.7}$$

Note in 2D, only the components E_{xx}, E_{yy}, E_{xy} are needed. The $\Omega_\alpha \Omega_\beta$ weights used to construct E are shown in Fig. 1.2. Based on the fact that $\vec{\Omega}$ is a unit vector, the QD tensor E has diagonal components ($\alpha = \beta$) that satisfy $E_{\alpha\alpha} \in [0, 1]$ and off-diagonal components ($\alpha \neq \beta$) that satisfy $E_{\alpha\beta} \in [-\frac{1}{2}, \frac{1}{2}]$.

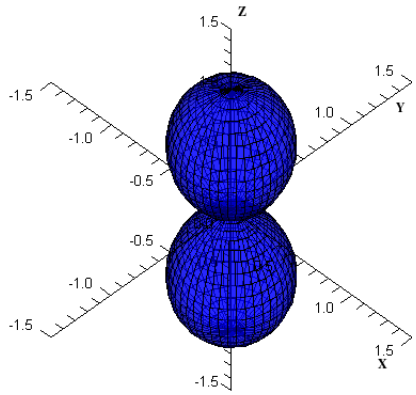
As a PDE, the steady-state LOQD equations of Eq. (1.4) have similar terms to steady-state convection-diffusion equations [90]. When solving the LOQD equations, we prefer to keep Eq. (1.4a) and Eq. (1.4b) separate and not eliminate the current $\vec{J}$ by substituting Eq. (1.4b) into the balance equation Eq. (1.4a). However, in order to compare LOQD to convection-diffusion, let us do just that, which yields

$$-\frac{1}{\sigma_t}\left(E_{xx}\frac{\partial^2 \phi}{\partial x^2} + 2E_{xy}\frac{\partial^2 \phi}{\partial x \partial y} + E_{yy}\frac{\partial^2 \phi}{\partial y^2}\right) + \vec{b} \cdot \vec{\nabla}\phi + c\,\phi = q_{ext}, \tag{1.8a}$$

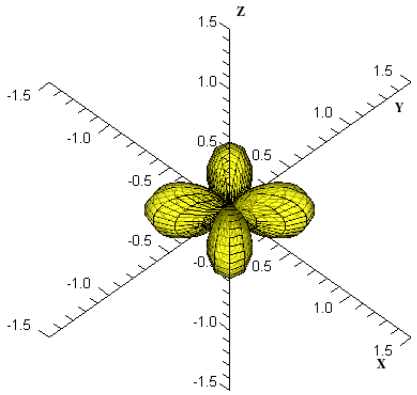


(a) $\mathbf{\Omega_x^2}$

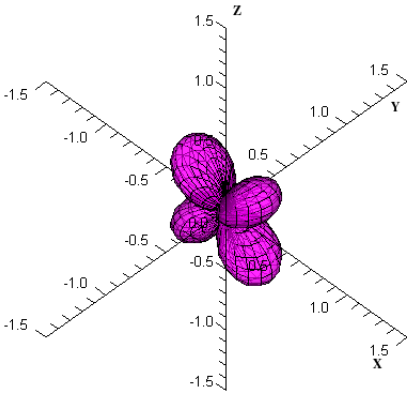
(b) $\mathbf{\Omega_y^2}$



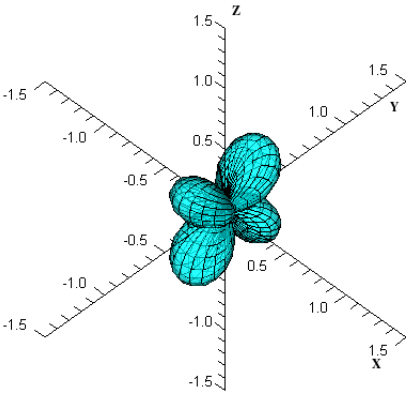
(c) $\mathbf{\Omega_z^2}$



(d) $\mathbf{\Omega_x\Omega_y}$



(e) $\mathbf{\Omega_x\Omega_z}$



(f) $\mathbf{\Omega_y\Omega_z}$

Figure 1.2: QD Factor Angular Weight Functions, $\Omega_\alpha\Omega_\beta$

with

$$\vec{b} = -\frac{1}{\sigma_t}\begin{pmatrix} 2\frac{\partial E_{xx}}{\partial x} + 2\frac{\partial E_{xy}}{\partial x} \\ 2\frac{\partial E_{xy}}{\partial y} + 2\frac{\partial E_{yy}}{\partial y} \end{pmatrix}, \tag{1.8b}$$

$$c = -\frac{1}{\sigma_t}\left(\frac{\partial^2 E_{xx}}{\partial x^2} + 2\frac{\partial^2 E_{xy}}{\partial x \partial y} + \frac{\partial^2 E_{yy}}{\partial y^2}\right) + \sigma_a, \tag{1.8c}$$

where we have assumed 2D and constant total cross section σ_t. Compared to the second-order form of the LOQD equations in Eq. (1.8), the model convection-diffusion equation in [90] contains much simpler second-order terms, $D\partial^2\phi/\partial x^2 + D\partial^2\phi/\partial y^2$. The second-order term in the LOQD equations more closely resembles that found in tensor diffusion which has $\vec{J} \propto \mathrm{D}\vec{\nabla}\phi$, where D is a symmetric, positive definite diffusion tensor [97]. However, because the diffusion tensor, D, appears outside the derivative term, QD has second-order cross-terms, $\partial^2 E_{xy}/\partial x \partial y$, that tensor diffusion does not.

1.2.2 Angle, Energy, and Time Discretizations

The QD equations were presented in Eq. (1.3) and Eq. (1.4) for the steady-state, single-group (monoenergetic) case. A discretization in angle and space must still be specified. To discretize the angular domain, we use the discrete ordinates (S_N) approximation[3], which defines a quadrature rule to evaluate integrals over angle as in Eq. (1.2),

$$\int_{4\pi} d\Omega\, f(\vec{\Omega}) \approx \sum_{m=1}^{N_M} w_m\, f(\vec{\Omega}_m)\,, \tag{1.9}$$

where the set of directions (or ordinates) is $\{\vec{\Omega}_m | m = 1, ..., N_M\}$, with corresponding weights, $\{w_m | m = 1, ..., N_M\}$.

[3]See [30, 79] and references therein for more information on the S_N approximation and [107] for an angular truncation error analysis.

The spatial discretization of the QD equations is a focus of this work. However, the spatial discretization we develop is not restricted to the steady-state single-group case. Extending a given spatial discretization from the one-group case to the multi-group case is generally straightforward [30]. To solve time-dependent equations, simple discretizations in time are fairly straightforward to implement as a sequence of "steady-state" calculations.

1.3 History of Transport Methods

Here we will discuss the quasidiffusion method as it relates to other transport methods. This discussion is centered about two features of Quasidiffusion.

1. The LOQD equations may be used to accelerate convergence of transport iterations. The attitude is that an iterative solution to Eq. (1.3) is desired and the inclusion of the LOQD equations Eq. (1.4) speeds convergence of the iterations.

2. The LOQD representation of the transport problem facilitates coupling to other physics. The attitude here is remarkably different in that Eq. (1.4) is treated as the central equation, where the QD factor data is updated periodically by some means (e.g. Eq. (1.3)).

Depending on the application, one may find particular advantage in one of the features above.

1.3.1 QD as an Acceleration Method

In order to discuss QD as an acceleration method we introduce the following operator form of the transport equation of Eq. (1.3)

$$L\psi = \frac{\sigma_s}{4\pi}\phi + \frac{1}{4\pi}q_{ext} \qquad \text{in domain } G, \tag{1.10a}$$

$$\ell\psi = \psi_{IN} \qquad \text{on boundary } \partial G, \tag{1.10b}$$

where the transport operator consists of interior operator L and boundary operator ℓ. Utilizing the S_N approximation we get (for the interior),

$$L_m\psi_m = \frac{\sigma_s}{4\pi}\phi + \frac{1}{4\pi}q_{ext} \tag{1.11}$$

to solve for each direction in the quadrature set, $\vec{\Omega}_m$, where the scalar flux is given by approximate integration via quadrature set $\phi = \sum_m w_m\psi_m$. In practice, Eq. (1.11) is solved iteratively by assuming scattering source $\sigma_s\phi$ from a previous iteration or from the solution of *acceleration equations*.

1.3.1.1 The Transport Sweep

The major advantage of using the first-order form of the transport equation Eq. (1.1) and the discrete ordinates approximation[4] is that the L_m operator is easily inverted. On orthogonal meshes in Cartesian geometry, the L_m operator of Eq. (1.11) under most discretizations is triangular and may be inverted by simple backsubstitution, starting with known boundary conditions. This can be visualized as starting from boundary conditions and moving in the direction $\vec{\Omega}_m$, visiting

[4]In Cartesian geometry, the discrete ordinates transport operator L_m is identical to the operator for the method of characteristics. In curvilinear geometry, where $\vec{\Omega}\cdot\vec{\nabla}\psi$ has angular derivatives, the characteristics and discrete ordinates treatments result in different L_m operators.

each cell in a sequence such that upwind information is known. This way of inverting L_m for all m is known as a *transport sweep*. On unstructured meshes, the proper ordering of cells for the transport sweep is more complicated. One must consider the dependency graph of cells to their upwind neighbors, as dictated by which faces are incoming and outgoing for a given direction. On unstructured meshes in 2D, the dependency graph is known to be a Directed Acyclic Graph (DAG) and there are simple algorithms for traversing the DAG using stacks [60, 84]. In graph terminology, cells should be visited in the postorder (or reverse order) of the depth-first traversal of the DAG. On unstructured meshes in 3D, the dependency graph may be cyclic, which indicates a non-triangular structure for L_m. With cycles and in parallel environments, more complicated sweep algorithms are used [84].

1.3.1.2 Transport Iterations

The simplest way to solve Eq. (1.11) is to use the transport sweep for the action of L_m^{-1} and source iterations (SI) for the scattering,

$$\psi_m^{k+1/2} = L_m^{-1}\left(\frac{\sigma_s}{4\pi}\phi^k + \frac{1}{4\pi}q_{ext}\right) \quad \text{for each direction } m, \qquad \phi^{k+1} = \sum_m w_m \psi_m^{k+1/2}, \tag{1.12}$$

where k is the index for transport iterations. (The reason for the $k+1/2$ indices will become apparent later when acceleration methods are introduced.) The well-known drawback of SI (aka Richardson iterations) is that the iteration process converges very slowly for highly scattering problems. The spectral radius[5] of SI is determined from Fourier analysis in an infinite homogeneous medium to be

[5] The spectral radius ρ is the largest absolute value of eigenvalue ω over all wave numbers λ,

$$\rho = \max_{-\infty \le \lambda \le \infty} |\omega(\lambda)|. \tag{1.13}$$

$\rho^{\text{SI}} = c$, where the scattering ratio, $c \stackrel{\text{def}}{=} \frac{\sigma_s}{\sigma_t}$. Using SI leads to an unacceptable number of transport iterations in many practical problems, for example, in nuclear reactor models with large thermal neutron energy groups. This leads us to methods for accelerating transport iterations.

Acceleration methods solve the following system of equations,

$$\psi_m^{k+1/2} = L_m^{-1}\left(\frac{\sigma_s}{4\pi}\phi^k + \frac{1}{4\pi}q_{ext}\right) \quad \text{for each direction } m, \tag{1.14a}$$

$$\phi^{k+1} = F_{k+1/2}^{-1} f^{k+1/2}, \tag{1.14b}$$

where $F_{k+1/2}$ is the operator of the acceleration equations, ϕ^{k+1} is the *accelerated* flux, and $f^{k+1/2}$ is the RHS of the acceleration equations. Both F and f may depend on the solution $\psi_m^{k+1/2}$. In acceleration methods, Eq. (1.14a) is the *high-order problem* and Eq. (1.14b) is the *low-order problem*. The terms high-order and low-order refer to the *angular dimensionality* of the problem, not for example, the spatial order of accuracy for the method. Similar naming conventions have been adopted with QD: the transport equation of Eq. (1.3) is the high-order problem and the LOQD equations of Eq. (1.4) is the low-order problem. In order to proceed with our review of acceleration methods, let us divide acceleration methods into two categories: linear and nonlinear acceleration [47].

1.3.1.3 Linear Acceleration Methods

In the case of linear or *synthetic acceleration methods*, the accelerated flux ϕ^{k+1} is an additive correction to the unaccelerated flux $\phi^{k+1/2}$,

$$\psi_m^{k+1/2} = L_m^{-1}\left(\frac{\sigma_s}{4\pi}\phi^k + \frac{1}{4\pi}q_{ext}\right) \quad \text{for each direction } m, \tag{1.15a}$$

$$\phi^{k+1/2} = \sum_m w_m \psi_m^{k+1/2},$$

$$\Delta\phi^{k+1} = F_{k+1/2}^{-1} f^{k+1/2}, \tag{1.15b}$$

$$\phi^{k+1} = \phi^{k+1/2} + \Delta\phi^{k+1}, \tag{1.15c}$$

where the correction satisfies $\Delta\phi^{k+1} \to 0$ as the solution converges. The downside of acceleration via this type of correction is that in order to have stability, the low-order system Eq. (1.15b) and high-order system Eq. (1.15a) must be consistently discretized, in some sense [33, 42, 70]. Synthetic acceleration methods may be interpretted as *preconditioned Richardson iterations* [88] with the operator F of Eq. (1.15b) related to the preconditioner.

The most popular synthetic acceleration methods are Diffusion Synthetic Acceleration (DSA) [22, 23, 29, 59] and Transport Synthetic Acceleration (TSA) [62].

In DSA, the low-order problem is pure diffusion and effectively accelerates the diffusive ($\lambda = 0$) mode of the solution, which dominates in many problems with high scattering. With DSA, the spectral radius of transport iterations is substantially reduced from that of source iterations, $\rho^{\text{DSA}} \approx 0.22c$. Developing consistent discretizations for DSA with efficient solvers is difficult, especially in multi-dimensional geometry. As an example, consider that the first stable and effective DSA method in 2D Cartesian geometry is due to Wareing, et. al. in 1991 [41], where they used a bi-linear discontinuous (BLD) finite element discretization on rectangles for the transport and DSA

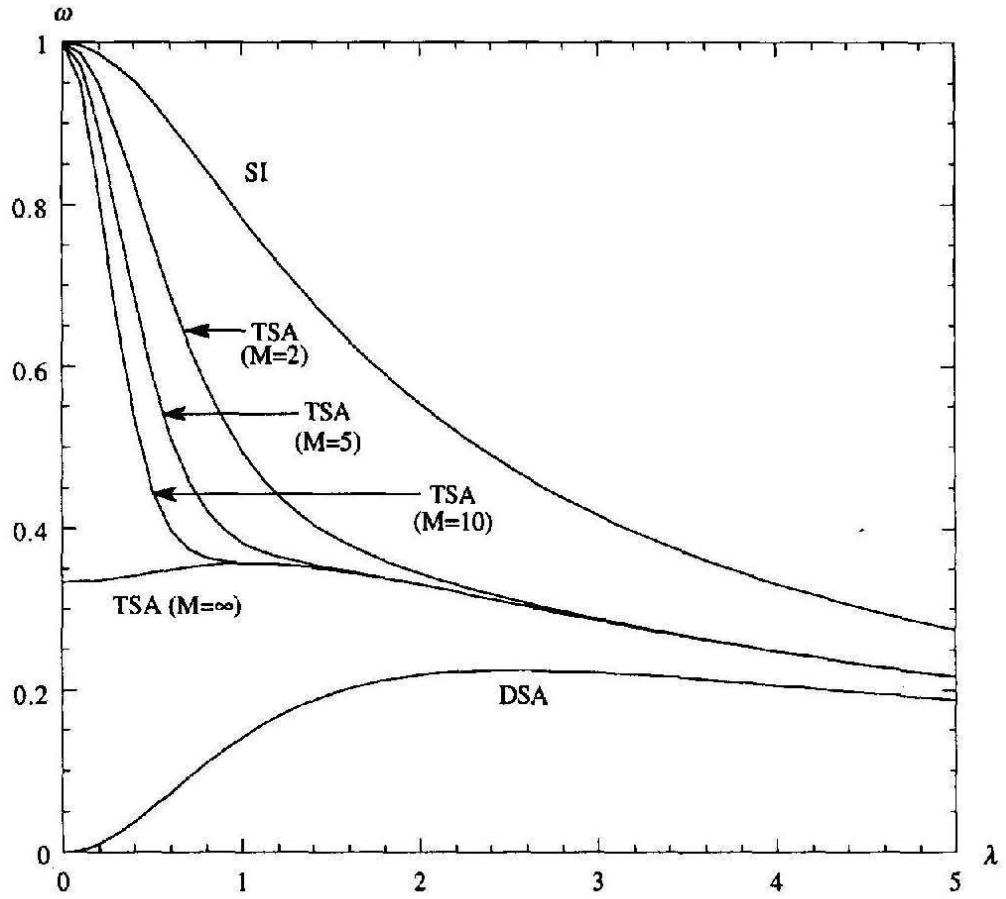


Figure 1.3: Comparison of eigenvalues ω versus wave number λ for TSA (with $\beta = 1/3$), DSA, and SI with scattering ratio $c = 1$. Figure taken from [88].

equations. However, an efficient solver was not available until 1993, when multigrid techniques were applied to the DSA equations by Morel, et. al. [55].

With TSA, a transport equation with a reduced order quadrature set (usually S_2) and modified cross sections (via free parameter $\beta \in [0, 1]$) is used to accelerate the transport iterations. Because the low-order equation of TSA is the transport equation there is no issue with a consistent discretization. The TSA equations are solved iteratively, just like the transport equation, so the transport iterations have "inner iterations" inside each acceleration update. Typically, the TSA equations are not iterated to convergence but to some maximum number of iterations M [88].

Although TSA may lead to large reduction in spectral radius over SI, the optimal values for free parameter β and maximum number of "inner iterations" M are problem-dependent. Also, for optically thick problems with very small absorption, the spectral radius of the TSA scheme (with fixed M) still approaches unity [88], shown in Fig. 1.3 with DSA for comparison.

Also worth mentioning are the adjacent-cell preconditioner (AP) methods of Azmy [70,87] which address consistency by matching spectral properties of the high-order and low-order problems.

1.3.1.4 Nonlinear Acceleration

As an accelerator, QD behaves similarly to DSA in diffusive problems, accelerating the diffusive mode of the transport solution with $\rho^{\text{QD}} \approx 0.22c$. However, unlike DSA, QD is 1) nonlinear, 2) stable under *independent discretizations* of the low-order and high-order problems, 3) the low-order problem is not an approximate transport method, [6] and 4) results in a multiplicative correction to the high-order problem [33, 47],

$$\psi_m^{k+1/2} = L_m^{-1}\left(\frac{\sigma_s}{4\pi}\phi^k + \frac{1}{4\pi}q_{ext}\right) \quad \text{for each direction } m, \tag{1.16a}$$

$$\phi^{k+1} = F_{k+1/2}^{-1}[\psi^{k+1/2}]f^{k+1/2}. \tag{1.16b}$$

The multiplicative correction described by Eq. (1.16a) is actually common to a class of methods called nonlinear projective iteration (NPI) methods [24, 47], also known as projected discrete ordinates (PDO) methods [33]. Just like QD, the NPI methods are defined by a system of *nonlinearly coupled* high-order and low-order problems that are equivalent to the original, linear transport problem of Eq. (1.11). Also just like QD, the equations of NPI methods are closed by a defining set

[6] In DSA, the low-order problem is a diffusion equation and only under certain asymptotic conditions is a solution to the transport equation. In QD, the low-order problem (the LOQD equations) is capable of representing the transport equation exactly.

of linear-fractional factors and usually converge fast because the factors are weakly dependent on the solution, as evidenced by a small Frechet derivative. The acceleration effectiveness of QD can also be explained by interpretting QD as a 2-level angular multigrid for the transport equation [39]. All NPI methods have the versatility of allowing independent discretizations of the high-order and low-order problems. Other nonlinear acceleration methods are discussed in [47, 78, 88].

One interesting feature of independent discretizations is that the coverged high-order and low-order solutions are not necessarily equal,

$$\lim_{k\to\infty}(\phi_h^{k+1/2} - \phi_h^k) = O(h^p),$$

differing by discretization-dependent truncation error, $O(h^p)$, where h is some measure of the size (length) of a spatial cell and p satisfies $p > 0$ for a consistent method. Naturally, we only work with consistent methods for which the difference between converged high-order and low-order solutions is guaranteed to approach zero as the mesh is refined ($h \to 0$).

1.3.1.5 Krylov Iterations for Transport Methods

Another linear method for acceleration of transport iterations deserves its own category as it is not based on Richardson iterations but on Krylov iterations. In 2004, Warsa, et. al. showed that DSA can break down in highly heterogeneous media for discontinuous finite element (DFEM) discretizations of the transport equation, but by using *Krylov methods with diffusion preconditioning*, robustness was recovered [96]. In order to be amenable to Krylov iteration, the

transport equation in Eq. (1.11) may be cast in the following form [7],

$$\left(I - DL_m^{-1}\frac{\sigma_s}{4\pi}\right)\phi = DL_m^{-1}\left(\frac{1}{4\pi}q_{ext}\right), \tag{1.17}$$

where I is the identity operator and D is the so-called discrete-to-moment operator, describing how angular fluxes ψ_m are converted to scalar fluxes ϕ, i.e. the quadrature integration. The linear system $A\phi = b$ may be solved iteratively via Krylov methods [91], where iteration matrix $A = (I - DL_m^{-1}\frac{\sigma_s}{4\pi})$ and the RHS is $b = DL_m^{-1}\left(\frac{1}{4\pi}q_{ext}\right)$, which may be interpretted as the uncollided flux. Note, the efficient transport sweep is still utilized when computing the action of A, as needed by a particular Krylov method. In [96,102], GMRES is used to solve Eq. (1.17) with a DFEM transport discretization, preconditioned via a partially consistent (continuous FEM) diffusion equation. Additionally, a discontinuous update procedure is used to map the continuous unknowns of the diffusion preconditioner to the discontinuous unknowns of the DFEM transport discretization.

For practical applications, one disadvantage of Krylov methods is that, because it is a linear method, one has trouble applying nonlinear negative-flux fixup techniques [109]. (In our notation, negative flux fixups result in a nonlinear transport sweep $L_m^{-1} \rightarrow L_m^{-1}[\psi_m)]$.) With many discretizations, including DFEM, negative fluxes may be calculated in cells that are many mean-free paths thick ($\sigma_t h \gg 1$) or near interfaces between unlike materials. Negative fluxes are, of course, physically not meaningful but may also lead to instabilities when the transport solution is coupled to other physics [109]. Introducing negative-flux fixups into Krylov iterations is difficult for two reasons: 1) introducing any non-linearity into the system destroys the theory of convergence and 2) a negative flux-fixup (even a linear one) must be applied to a physical quantity (e.g. the angular

[7] We show the iterative system for the scalar flux ϕ, but in general one can use the hybrid methods of Morel [36] to solve for any number of moments (in the spherical harmonic sense) of the angular flux or solve for the angular flux itself ψ_m.

flux ψ), not to subspace vectors specific to the Krylov method.

1.3.2 QD as a Low-Order Transport Method

For applications with time-dependent, highly non-linear, coupled physics equations (e.g. radiative transfer and radiation hydrodynamics[8]), the transport method must be relatively cheap and robust, which has led researchers to seek approximations which allow one to solve a set of PDEs of reduced complexity and cost compared to the full transport equation of Eq. (1.1). Another strong motivation for low-order transport methods is that in many applications, the most important radiation-related quantities are the particle/media reaction rates and the particle flow rates, proportional to the scalar flux ϕ and current $\vec{J}$, respectively. The complete distribution of particles in space and angle described by the angular flux ψ is unnecessary.

1.3.2.1 P_N Equations

The P_N equations are derived by taking spherical harmonics moments of the transport equation Eq. (1.1) [30]. They have an advantage over S_N methods in not producing ray effects as all angular basis functions are continuous over the entire unit sphere. However, imposing transport boundary conditions, ψ_{IN}, for incoming directions, is difficult with P_N methods, especially in multi-dimensional geometry. The P_N equations also need closure as the first N equations will contain $N+1$ moments—usually the highest order moment, Φ_{N+1}, is assumed constant, i.e. all derivative terms with Φ_{N+1} are zero [30]. Of the P_N family, the P_1 equations receive the most attention as low-order transport methods—without time-dependence the P_1 equations are identical to the diffusion equations [92].

[8] An excellent text for introductory radiation hydrodynamics and current methodologies is [98].

1.3.2.2 Flux-Limited Diffusion

A physical law can be derived that says $|\vec{J}| \leq \phi$, that is the magnitude of the net current $|\vec{J}|$ must be less than the scalar flux ϕ [15]. In astrophysics (and most physics for that matter), the quantities $\vec{J}$ and ϕ are called the "flux" and "intensity", respectively, which we will use for the remainder of this flux-limited diffusion section. Conversely, $|\vec{J}| > \phi$ is a violation of particle conservation[9] but may arise quite easily in a diffusion discretization where $\vec{J} = -\frac{1}{3\sigma_t}\vec{\nabla}\phi$. The technique of "flux-limiting" was created to force diffusion (or P_1) discretizations to conserve particles.

1.3.2.3 "Eddington Methods"

There is a host of low-order transport methods with "Eddington" somewhere in the name. They all introduce a *closure relation* into the system of equations formed by taking the 0^{th} and 1^{st} angular moments of the transport equation, i.e. the P_1 equations. In our QD notation from Eq. (1.6), the "Eddington Approximation" [1] is the approximation $\mathrm{E} \approx \frac{1}{3}\mathrm{I}$, where I is the identity tensor— it is simply the diffusion approximation. An extension to the Eddington approximation is presented in [10] in which the extended equations closely resemble the LOQD equations, but approximate closures are formulated which do not rely on a high-order transport sweep. The "Variable Eddington Factor" (VEF) methods follow naturally, where new approximate closures are presented that are allowed to vary, $\mathrm{E}(\vec{r}) \approx \mathrm{E}(\eta)$ [16], where η is the ratio of magnitude of the flux to the intensity, $\eta(\vec{r}) = |\vec{J}(\vec{r})|/\phi(\vec{r})$. Thus the VEF equations are a closed system for $\vec{J}$ and ϕ. The closure relations, $\mathrm{E}(\eta)$, are asymptotically correct as $\eta \to 0$ (isotropic) and $\eta \to 1$ (pure streaming) but intermediate values differ significantly by choice of factor [82]. For example,

[9] In the flux-limiter literature, the notation $\vec{F}$ is used for the flux, $\vec{F} = \int_{4\pi} d\Omega\, \vec{\Omega}\psi$, and E for the energy density, $E = \frac{1}{c}\int_{4\pi} d\Omega\, \psi$, where c is the speed of light. Related to our notation, $\vec{F} = \vec{J}$ and $\phi = E/c$. So in this notation, the "flux limit" is $|\vec{F}| \leq cE$, and the violation is viewed as allowing particles to move faster than the speed of light c.

consider the Minerbo closure [16], derived from statistical considerations in 1D,

$$E_{xx} = 1 - \frac{2\eta}{Z},$$

where Z is a parameter related to η by

$$\eta = \coth Z - \frac{1}{Z}.$$

Eddington factors E(η) may also be cast as flux limiters, as described in [27], where it is shown that flux limiters derived from Eddington factors lead to a well-behaved theory whereas using the actual Eddington factors can produce anomalous results. A wealth of information on flux-limiters in the general context of conservation laws may be found in [43]. Currently, the preferred strategy for calculating the Variable Eddington Factors is by transport sweep, just as in QD [86,98].

1.3.2.4 LOQD Equations

The QD Method (or the VEF Method with factors calculated from ψ) is not a low-order transport method, because one must still solve the transport equation Eq. (1.11) in order to get factors $E_{\alpha\beta}[\psi]$ and $C[\psi]$. However, one can still see an advantage when solving fully time-dependent and multi-group radiation transport problems with coupling to other physics. Under these conditions, the LOQD equations are fully discretized in time and energy and the factors $E_{\alpha\beta}[\psi]$ and $C[\psi]$ may simply be treated as nonlinear data that must be updated periodically via snapshot-in-time solutions of the transport equation [86].

1.3.3 Applications of the QD Method

The point of sections entitled "QD as a Low-Order Transport Method" and "QD as an Acceleration Method" was to give the reader insight into the relation of the QD Method to other

radiation transport methods. At this point, one thing should be clear to the reader, that there is no simple solution to transport problems as all methods have advantages and disadvantages. In particular with the QD Method, the low-order quasidiffusion (LOQD) equations is useful as both an acceleration method and as a low-order representation of the transport problem, the main impediment to their use may be they are difficult to discretize and solve. However, QD has already been applied successfully in many areas, including radiation hydrodynamics and reactor physics, and properties of the method have been investigated by numerous authors [18, 24, 31, 33, 39, 40, 47, 51, 95]. We shall remark on some in the following sections.

1.4 History of QD Discretizations

A QD discretization requires both a discretization of the transport equation Eq. (1.3) and the LOQD equations Eq. (1.4). Therefore it makes sense to discuss both transport and LOQD discretizations, beginning with LOQD.

1.4.1 LOQD Spatial Discretizations

As discussed in Sec.1.2.1.1, mathematically the LOQD equations are similar to convection-diffusion equations, with two important exceptions, the LOQD equations have second-order cross-derivative terms, $\partial^2\phi/\partial x\partial y$, and the "diffusion coefficients" are not isotropic, $E_{xx} \neq E_{yy}$. Also, we desire discretizations for the first-order form of the LOQD equations, shown in Eq. (1.4), not the second-order form shown in Eq. (1.8). For these reasons, many of the spatial discretizations from convection-diffusion and tensor diffusion literature are difficult to apply to the LOQD equations. However, our LOQD system does represent a conservation law and the desireable qualities for

modelling conservation laws, as laid out by Leveque [43], apply to us as well. As is often done with conservation laws for elliptic equations, we focus on finite volume discretizations which are locally conservative and have an appropriate definition of the net current $\vec{J}$ on faces, thus guaranteeing conservation globally.

In the LOQD literature, the most advanced discretization is for time-dependent, convex quadrilaterals in Cylindrical RZ geometry, due to Aristova, Gol'din, and Kolpakov [68]. In Cartesian XY geometry, there are the LOQD discretizations of Gol'din, Gol'dina, and Kolpakov [24] and Aristova and Kolpakov [54], developed for skewed quadrilaterals and used for problems with shocks. For nuclear reactor assembly-level calculations, a coarse-mesh finite element discretization was developed by Hiruta and Anistratov [95, 103].

Another difficulty arises because the LOQD operator is non-self-adjoint in multi-dimensions, so that the discretization of the LOQD equations generally results in a non-symmetric linear system, which is difficult to solve. Larsen and Miften devised a symmetric version of quasidiffusion, "symmetrized Quasidiffusion" (SQD) [50, 51] by replacing the solution of the non-self-adjoint operator with two symmetric ones. An SQD discretization for rectangles was presented in [51] and structured meshes of triangles in [56].

1.4.2 Transport Spatial Discretizations

Although the QD method imposes no restriction on discretizations for the high-order and low-order equations, there is a strategy in choosing a high-order (transport) spatial discretization which compliments the LOQD equations. Particularly, the $E_{\alpha\beta}$ factors should satisfy some smoothness requirements because they appear in derivatives. Typically, this is ensured with a monotonic

transport method. However, this is not the only possibility. As an alternative, the angular flux or the QD factors may be monotonized (or smoothed) before using them in the LOQD equations. Negative angular fluxes are especially harmful with QD as the calculated factors may be completely unphysical, i.e. outside the bounds of $E_{xx}, E_{yy} \in [0, 1]$ and $E_{xy} \in [-1/2, 1/2]$. Historically with QD [4, 18, 26, 54], the transport equation was solved with a method of short characteristics with monotonization.

The result of monotonization is a nonlinear characteristic method [3], which are similar to the idea of ENO (essentially non-oscillatory) methods [32]. It is important to note that the short characteristics schemes used do not need to be conservative because the LOQD equations of Eq. (1.4) have explicit conservation [47]. The most important thing for the transport discretization is that it approximates the "shape" of the solution ψ well, embedded in the $E_{\alpha\beta}[\psi]$ and $C[\psi]$ factors. We also desire a transport local discretization, that is the transport equation may be solved cell by cell via transport sweep, where incoming fluxes ψ_{IN} are known from upwind cells and the cell-local discretization is used to calculate outgoing fluxes ψ_{OUT}. These reasons, and the fact that characteristic methods are relatively easily extended to arbitrary meshes leads us to consider them for our transport discretization for arbitrary meshes.

1.4.2.1 Characteristic Methods

Characteristic methods for deterministic transport [21] may be divided into two groups: short and long. The long characteristic method is widely used for nuclear reactor assembly-level calculations [20] and has its advantages, notably no interpolation error compared to short characteristics. However long characteristics has error associated with approximating volume-average

quantities from line-average contributions;, for details see [72] and the references therein. We concentrate on short characteristic methods, in particular the vertex-based methods as in [6, 26] and *conservative short characteristics methods*, described below.

Conservative short characteristic methods were originally formulated in the Russian work [34], we shall call subcell balance methods. The subcell balance discretizations operate on directionally-dependent slices through cells, or *subcells*. Because the subcells are geometrically simple, subcell balance methods are fairly simple to derive for unstructured meshes. The original subcell balance method due to Bakirova, V.Ya. Karpov, and M.I. Mukhina [34] was for transport on rectangles. In 1993, Voronkov and Sychugova produced the "Characteristic Discrete Ordinates Method" [52]. The subcell balance concept was investigated later by the following researchers:

- DeHart in 1992 produced "Extended Step Characteristics" [46],
- Azmy in 1992 produced "Arbitrarily High Order Transport" [45],
- Grove in 1995 produced the "Slice Balance Method" [60],
- Miller and Mathews in 1996 produced "Split-Cell Methods" [61], and
- Walters, Wareing, and Marr in 1995 produced "Nonlinear Characteristics" [58].

Below we shall briefly discuss these methods. Note that all these methods involve equations along characteristics, and the first such characteristic transport method was step characteristics (SC) due to Lathrop in 1969 [9].

Extended Step Characteristics

The Extended Step Characteristics (ESC) method of DeHart [46, 57] was developed for general,

polygonal geometries. ESC[10] has face-average and cell-average angular flux unknowns, and assumes a flat source distribution within a cell and flat incoming flux distribution on faces. The method is strictly positive and $O(h)$.

Arbitrarily High-Order Transport

The Arbitrarily High-Order Transport (AHOT) method of Azmy [45] was developed on rectangular meshes and uses short characteristics to develop a system of equations for 0^{th} through N^{th} spatial moments of the angular flux. Results were presented which showed $O(h^N)$ convergence. An AHOT method for tetrahedral meshes is developed in [77].

Characteristic Discrete Ordinates Method

The Characteristic Discrete Ordinates Method (CDS_N) of Voronkov and Sychugova [52] was developed on rectangular meshes. Three methods were developed: the $CDS_{N\,1}$ method of $O(h)$ and CDS_N and $CDS_{N\,2}$ methods of $O(h^2)$. They exploit the linearity of the transport L operator and use different discretizations for the incoming flux part and source part of the solution within a subcell. They then compare their methods to diamond differencing (DD) and linear characteristics (LC), noting good performance on coarse meshes, the expected convergence orders, and fewer negative fluxes than DD. However, they note overall LC seems superior, producing smaller errors consistently than the CDS_N methods.

Slice Balance

The Slice Balance method of Grove [60] presents a set of equations for $O(h^N)$ transport on arbitrary polygonal meshes. Thus, to some degree, it is an extension of DeHart's ESC method [46], in that it develops equations for the 0^{th} through N^{th} spatial moments—DeHart considered subcell balances

[10]It is currently used in the radiation transport method of Triton, the module used for nuclear reactor assembly level calculations in SCALE [111], a code package developed at Oak Ridge National Laboratory.

on only the O^{th} moment. Also, to some degree, it is an extension of Azmy's AHOT-C method [45], in that it develops equations for polygonal meshes, whereas Azmy's $O(h^N)$ method only applied to rectangular meshes. Grove discusses the advantages of decomposition of arbitrary polyhedral cells into "slices" (subcells), highlighting the fact that a slice is like a slab, therefore the slice balance method may be used to extend 1D methods into multi-D. Also, even concave cells may be handled by the slice balance method, if one is willing to globally decompose the mesh into slices, because slices are guaranteed to be convex. The resulting transport problem on slices permits a face-based transport sweep, which may allow more freedom in processor scheduling (for parallel applications) than a cell-based sweep, as well leading to fewer (or no) cycles. In the dissertation [60], slice balance discretizations were presented for step-characteristics (SB-SC), linear-characteristics (SB-LC), and a diamond difference-like method (SB-DDL). The SB-SC method is identical to the ESC method of DeHart.

Split-Cell Methods

The Split-Cell Methods of Miller and Mathews [61] take a different approach to transport solutions on unstructured meshes, preferring to consider meshes of triangles instead of polygons. Their split-cell discretizations, which they call "quadratures", express moments of the outgoing and interior angular flux in a triangular cell in terms of moments of the incoming angular flux and source in the cell. They decompose each triangular cell into 2 triangular subcells and balance the 0^{th} and 1^{st} moments, thus defining the Split-Cell LC method (S-LC). The S-LC method demonstrates good performance and the expected $O(h^2)$. In 2000, Mathews and Miller [69] extended the S-LC method to tetrahedral meshes, where each tetrahedron is split into 4 tetrahedral subcells. Split-cell methods have also been formulated that have exponential representations of the incoming angular

fluxes and sources (S-ExpC). In 1997, Mathews and Brennan formulated the S-ExpC method for triangles [63], and in 2001 by Brennan, Miller, and Mathews formulated the S-ExpC method on tetrahedra [76].

Nonlinear Characteristics

The Nonlinear Characteristic (NC) scheme of Walters, Wareing, and Marr [58] is formulated for meshes of rectangles with an exponential representation of the source and spatial moments of the angular flux on faces. The NC method is strictly positive and very similar to exponential split-cell methods (S-ExpC) of Miller and Mathews, but derived completely differently with arguments from information theory.

1.4.2.2 Other Transport Discretizations

Other common S_N transport discretizations utilize discontinuous finite elements and corner balances.

Discontinuous Finite Elements

On orthogonal meshes, one can construct a discontinuous FEM (DFEM) for transport on quadrilaterals using bi-linear discontinuous basis functions, called BLD. On non-orthogonal quadrilateral meshes, in order for this BLD to satisfy the thick diffusion limit, one must construct the FEM with bi-linear basis functions $\{1, x, y, xy\}$ on rectangles and map the result to the arbitrary quadrilateral [11]. On arbitrary triangular (and tetrahedral) meshes using linear-discontinuous basis functions the FEM is called LD. Additionally, Adams and Stone have developed a transport discretization on polyhedrons [94] which uses a piece-wise linear (PWL) finite element method (FEM).

[11] In mapping the bi-linear basis from rectangle to the arbitrary quadrilateral, the basis on arbitrary quadrilaterals is recognized as ratios of bi-linear basis functions, known as the Wachspress basis functions [80]. This is related to the fact that the inverse of a bi-linear transformation is not a bi-linear transformation [37].

Corner Balances

Corner balance methods are based on decomposition of the cell into corners with balance and auxiliary equations that allow the methods to satisfy certain asymptotics [64]. They have an advantage over DFEM methods in that an FEM matrix need not be inverted in each cell, outgoing fluxes are algebraically related to incoming ones.

1.4.2.3 Asymptotics

One final important aspect of a transport discretization is that it preserve some asymptotics. For example, a transport discretization should reduce to a reasonable approximation of the exponential in a source free attenuating medium and behave well on optically thick spatial cells of many mean-free paths (mfp), where $1mfp = 1/\sigma_t$. Another important asymptotic limit is the thick diffusion limit, where cells are thick compared to the diffusion length $1/\sqrt{3\sigma_t\sigma_a}$, and it is important that the transport discretization reduce to a reasonable diffusion discretization [38]. With DFEM transport discretizations, lumping is commonly used to increase robustness and better handle asymptotic behavior [80]. The thick diffusion limit for characteristic methods is discussed in [66].

1.5 Advancement

The *ideal transport method* would discretize (and solve) the transport equation in such a way to

- conserve particles,
- preserve the asymptotic diffusion limit,

- converge rapidly for all types of problems (e.g. highly scattering media, heterogeneous media, etc.), and
- converge rapidly for non-linear problems.

The Quasidiffusion method has been shown to satisfy these criteria. However, radiation transport practitioners demand more of discretizations. In particular, they should

- be valid on unstructured meshes,
- have robust and efficient solutions, and
- be at least second-order accurate in space.

This work seeks to address the second set of requirements while showing we have kept the first list intact, with the exception that we do not investigate non-linear problems here.

The main result of this research is the new low-order quasidiffusion (LOQD) discretization which is accurate on arbitrary meshes of quadrilaterals in XY geometry. We present this new discretization in Chapter 2, along with a general FV framework for LOQD discretizations and new interface conditions which allow a discretization of the LOQD equations to be used on hanging-node meshes. In Chapter 3, we present two transport discretizations which are compatible with the LOQD discretization presented in Chapter 2, and the manner in which QD factors should be calculated, as well as a new spatial representation of the scattering source. In Chapter 4, we also present details on the solution of the LOQD equations via iterative methods. In Chapter 5 we present some numerical results using our new QD method.

The results of this research have been presented by the candidate in the following venues:

1. American Nuclear Society (ANS) Summer Meeting in Boston, MA, June 24-28, 2007,

2. ANS Winter Meeting in Washington, D.C. November 11-15, 2007, and

3. NC State University, Nuclear Engineering Department seminar in Fall 2007.

Results have been published in ANS conference transactions [105, 106] and a full conference paper was accepted for publication in the peer-reviewed proceedings of the ANS 2009 Topical Meeting of the Mathematics and Computation (M&C) Division [110].

1.6 Arbitrary Mesh Terminology

In describing an arbitrary 2D mesh, the following indices will be used to denote the part of the mesh to which a quantity belongs: c for cell, f for faces, v for vertices, f_b for boundary faces (on the domain boundary), and i for interfaces.

The most basic mesh we use is the *simple orthogonal mesh*, described by the number of cells in the x and y directions, $N_C = N_x \times N_y$. An orthogonal hanging-node (or multi-level [97]) mesh may be created by refining a region of the simple mesh, by dividing all the quadrilateral cells in that region into four smaller quadrilaterals. Refining a single region leads to a two-level mesh. Further refinement (of a subregion) leads to a three-level mesh, etc.

Algorithm 2: Randomized mesh generation algorithm.

1. Generate an $N_x \times N_y$ mesh with square cells of width h.

2. At each interior vertex, sample a random angle in $\theta \in [0, 2\pi]$ and perturbation fraction $p \in [0, p_{max}]$.

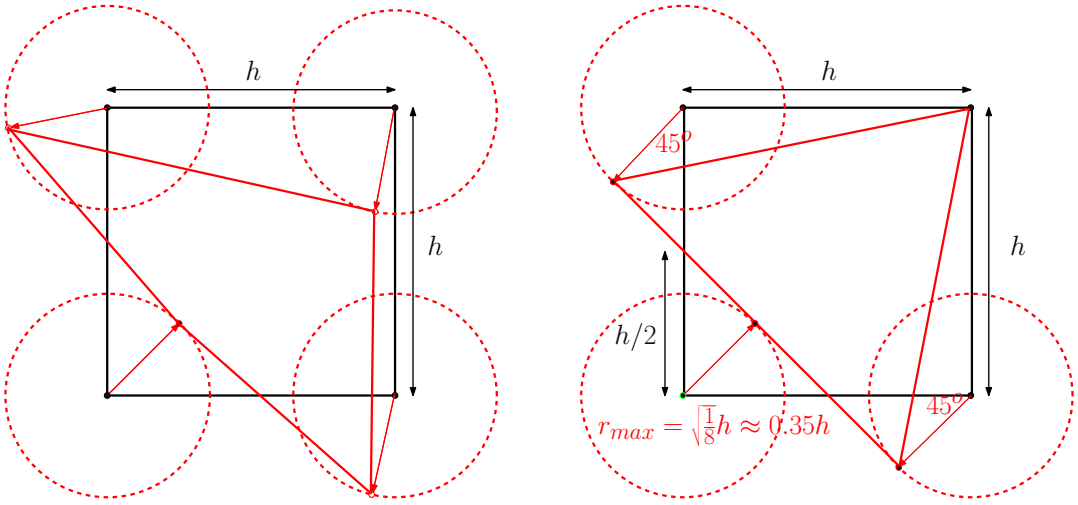
3. Move vertices from original location (x^v, y^v) to new location,

$$\begin{aligned} x^{v'} &= x^v + p\ h\ \cos\theta, \\ y^{v'} &= y^v + p\ h\ \sin\theta. \end{aligned}$$

4. Scale the mesh by factors H_x and H_y.

$$\begin{aligned} x^{v''} &= H_x x^{v'} \\ y^{v''} &= H_y y^{v'} \end{aligned}$$

In order to generate arbitrary quadrilateral meshes on which to test our new transport methods, we perturb vertices of an orthogonal mesh. We compare results on orthogonal and *randomized meshes* for problems which are well-suited to orthogonal meshes to demonstrate the relative invariance of our methods with respect to the mesh. Preserving the mean cell-width of a mesh allows us to analyze convergence on randomized meshes. The randomization that preserves the mean cell-width of a mesh is described in Alg. 2. The maximum perturbation is $p_{max} \approx 0.35$ such that cells remain convex (see Fig. 1.4.) The cell-width of a cell c is calculated as $h^c = (V^c)^{1/N_{dim}}$, where N_{dim} is the number of dimensions in the problem and V^c is the volume of the cell. In 2D, we have $h^c = \sqrt{V^c}$ and the "volume" may be calculated as the area of a polygon by the winding



(a) near-maximum perturbation

(b) maximum perturbation

Figure 1.4: Perturbation of vertices to construct randomized meshes.

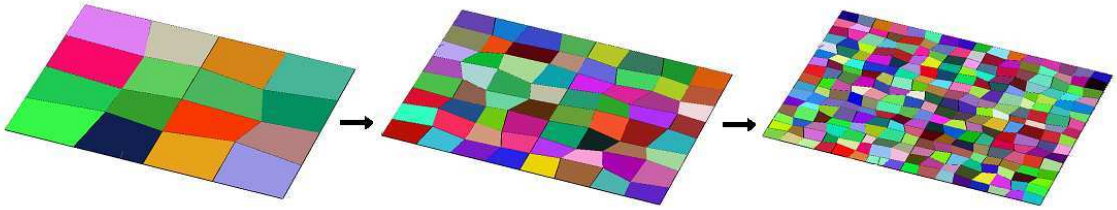
formula,

$$V^c = \frac{1}{2} \sum_{v \in c} \left(x^{v-1}\, y^v \; - \; x^v\, y^{v-1} \right), \tag{1.18}$$

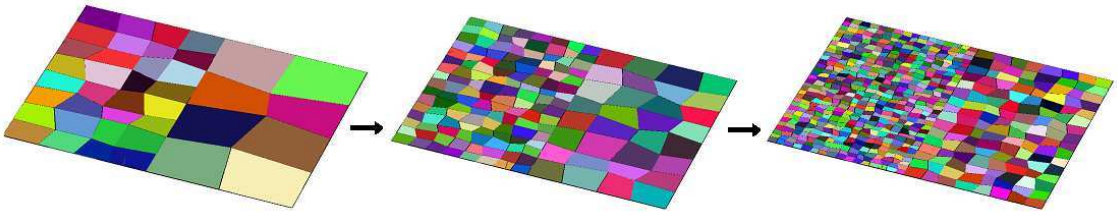
where $v \in c$ indicates summation over all N_V^c vertices in cell c and the convention is used that $v - 1 = 0 = N$. The mean cell-width of a randomized mesh may be expressed as $\bar{h} = \sqrt{\bar{V}}$, where the mean cell volume is $\bar{V} = \frac{1}{N_C} \sum V^c$. With the perturbations described in Alg. 2, the mean cell-width of a randomized mesh $\bar{h}$ defined this way is identically equal to the original h of the orthogonal mesh because the mean cell volume is unaltered by this type of perturbation. This allows us to analyze convergence as described in the next section.

1.7 Numerical Analysis Tools Used

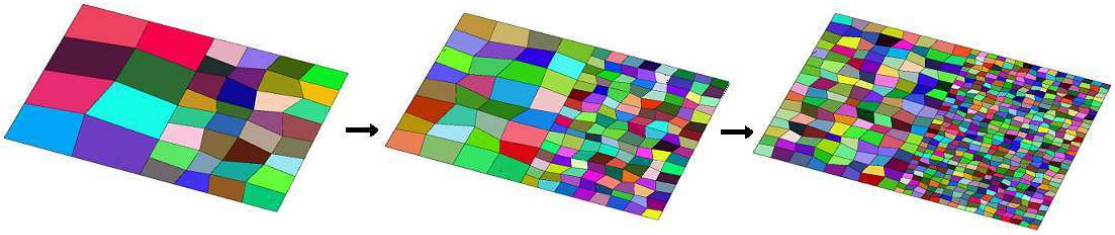
Our general strategy is to show the accuracy and spatial convergence of our new methods on orthogonal and randomized meshes. Typical sequences of orthogonal and randomized meshes



(a) logically-rectangular sequence



(b) left-refined sequence



(c) right-refined sequence

Figure 1.5: Examples of the randomized meshes used in numerical convergence studies. (Mesh cells colored randomly for contrast.)

on which we analyze convergence are shown in Fig. 1.5. Note that the successively refined randomized meshes are not generated from a previous randomized mesh by splitting cells but from new randomizations of orthogonal meshes.

1.7.1 Numerical Convergence Analysis with Known Exact Solution

In tests where we have an exact solution, we analyze convergence in the ℓ_2 and ∞ (or c) norms, shown below for the scalar flux ϕ, where we have a reference scalar flux solution, ϕ_{exact},

$$\|\phi^c - \phi^c_{exact}\|_{\ell_2} = \sqrt{\sum_c (\phi^c - \phi^c_{exact})^2 V^c}, \tag{1.19}$$

$$\|\phi^c - \phi^c_{exact}\|_{\infty} = \max_c |\phi^c - \phi^c_{exact}|, \tag{1.20}$$

where ϕ^c_{exact} is the cell-average exact solution, ϕ^c is the approximate cell-average, and V^c is the cell volume (area in 2D). The convergence ratio *Ratio* in a given norm is then the ratio of terms on successively refined meshes (h and $h/2$, respectively),

$$Ratio = \frac{\|\phi^c_h - \phi^c_{exact}\|}{\|\phi^c_{h/2} - \phi^c_{exact}\|}, \tag{1.21}$$

where h is the mean cell-width of the mesh. The convergence order *Order* is given as

$$Order = \frac{\log Ratio}{\log 2}, \tag{1.22}$$

where it is assumed we always refine the mesh by decreasing the mean cell-width h by a factor of 2.

1.7.2 Numerical Convergence Analysis without Exact Solution

In tests where we do not know the solution, we will calculate a region-averaged quantities and analyze the convergence of this property with the Aitken (aka Richardson) process. For example, in many tests we calculate the average scalar flux in a subdomain of the problem and analyze the convergence of this quantity.

With the Aitken process, we require approximate quantities on three successively refined meshes, a_h, $a_{h/2}$, and $a_{h/4}$,

$$Ratio = \frac{a_h - a_{h/2}}{a_{h/2} - a_{h/4}}. \tag{1.23}$$

If the *Ratio* calculated as per Eq. (1.23) is negative, it indicates the approximate solution is oscillating about the actual solution. That is, in order to have asymptotic convergence Eq. (1.23) must be positive. The convergence order may then be calculated as in Eq. (1.22).

Chapter 2

A FINITE VOLUME LOQD DISCRETIZATION

First we describe a general finite volume (FV) framework for discretizing the low-order quasidiffusion (LOQD) equations. The framework uses a cell-local discretization of the balance equation, Eq. (1.4a), and the LOQD first moment equations, Eq. (1.4b), plus global boundary conditions, Eq. (1.4c), and interface conditions. Proper interface conditions are determined for both single-level and hanging-node meshes. Then, using the general FV framework, we present a new LOQD discretization for arbitrary quadrilateral meshes. Finally, we present numerical results for both the new interface conditions and new LOQD discretization.

2.1 General Finite Volume LOQD Methodology

Let us introduce a general finite volume framework for discretizing the LOQD equations which has cell-average scalar flux unknowns, ϕ^c, face-average scalar flux unknowns, ϕ^f, and face-

average normal current unknowns, $J^f = \vec{J}^f \cdot \vec{n}^f$, where $\vec{n}^f$ is the outward normal of face f. For convenience, consider the LOQD equations in the following form,

$$\vec{\nabla} \cdot \vec{J} + (\sigma_t - \sigma_s)\phi = q_{ext} \qquad \text{in domain } G, \tag{2.1a}$$

$$\vec{\nabla} \cdot (\phi \mathrm{E}) + \sigma_t \vec{J} = 0 \qquad \text{in domain } G, \tag{2.1b}$$

$$\vec{n} \cdot \vec{J} + J_{IN} = C(\phi - \phi_{IN}) \qquad \text{on boundary } \partial G, \tag{2.1c}$$

where we have used the notation $\vec{\nabla} \cdot (\phi \mathrm{E})$ for the tensor divergence explicitly shown in Eq. (1.4b). In order to derive the general FV framework, we proceed as follows.

1. Integrate the scalar QD balance equation of Eq. (2.1a) over each cell c in the mesh.
2. Integrate the vector QD first moment equation of Eq. (2.1b) over each face f of cell c and then project the result onto the face outward normal $\vec{n}^f$.
3. Integrate boundary conditions of Eq. (2.1c) over each boundary face.
4. Specify interface conditions.

The first two steps of this process yield $N_F^c + 1$ equations for each cell, where N_F^c is the number of faces on cell c,

$$\sum_{f \in c} J^f A^f + \sigma_a \phi^c V^c = q_{ext}^c V^c \quad \text{for each cell } c, \tag{2.2a}$$

$$\vec{\nabla} \cdot (\phi \mathrm{E})\Big|_f \cdot \vec{n}^f A^f + \sigma_t J^f A^f = 0 \quad \text{for each face } f \text{ in cell } c. \tag{2.2b}$$

Per cell, there is one cell-average scalar flux ϕ^c, N_F^c face-average scalar-fluxes, and N_F^c face-average normal currents, for a total of $2N_F^c + 1$ unknowns per cell. A specific FV discretization is defined by the approximation used for the face-average tensor divergence term, $\vec{\nabla} \cdot (\phi \mathrm{E})\Big|_f$ in Eq. (2.2), which

will be deferred momentarily. The third step of the process yields discrete boundary conditions for each boundary face f_b of N_{BF} boundary faces total,

$$J^{f_b} + J^{f_b}_{IN} = C^{f_b}(\phi^{f_b} - \phi^{f_b}_{IN}) \quad \text{for each boundary face } f_b, \tag{2.2c}$$

The fourth and final step of specifying interface conditions that provides the remaining equations is discussed in the next section. The proper interface conditions are shown to be

$$\begin{aligned} \phi^{if_1} &= \phi^{if_2}, \\ J^{if_1} &= -J^{if_2}, \end{aligned}$$

for standard interfaces i of 2 faces (see Fig. 2.1(a)) and

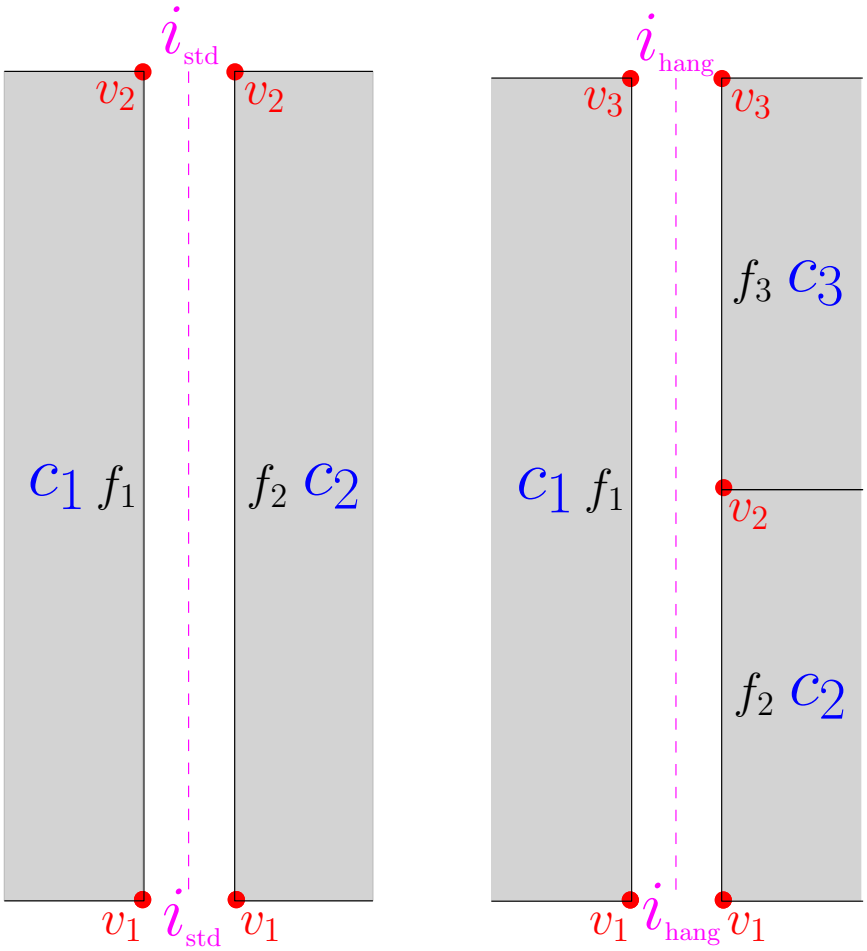
$$\begin{aligned} A^{if_1}\phi^{if_1} &= A^{if_2}\phi^{if_2} + A^{if_3}\phi^{if_3}, \\ J^{if_1} &= -J^{if_2}, \\ J^{if_1} &= -J^{if_3}, \end{aligned}$$

for interfaces i of 3 (or more) faces—see Fig. 2.1(b).

2.1.1 Interface Conditions

To determine the interface conditions, we look at the remaining number of equations after considering the discretized balance equation, first moment equations, and boundary conditions in Eq. (2.2a), Eq. (2.2b), and Eq. (2.2c), respectively. Globally, we have a number of equations

$$N_{eqn} = \sum_{c=1}^{N_C}(N_F^c + 1) + \sum_{i=1}^{N_I} N_{IC}^i + N_{BF}, \tag{2.3}$$



(a) standard interface i_{std} (b) hanging node interface i_{hang}

Figure 2.1: Interfaces considered: (a) standard i_{std} and (b) hanging node i_{hang}.

where N_I is the number of interfaces and N_{IC}^i is the number of interface conditions on interface i. The number of unknowns is simply

$$N_{unk} = \sum_{c=1}^{N_C} (2N_F^c + 1). \tag{2.4}$$

We shall use these results to show how many interface condition equations we are allowed. A necessary condition for a unique solution is $N_{eqn} = N_{unk}$, or equivalently

$$\sum_{i=1}^{N_I} N_{IC}^i + N_{BF} = \sum_{c=1}^{N_C} N_F^c.$$

The quantity $\sum_{c=1}^{N_C} N_F^c$ is simply the total number of faces, N_F, in the mesh. Splitting the number of faces in the mesh into N_{BF} boundary faces and N_{IF} interior faces, $N_F = N_{BF} + N_{IF}$, we arrive at the condition

$$\sum_{i=1}^{N_I} N_{IC}^i = N_{IF}, \tag{2.5}$$

which says that (globally) we must have a number of interface conditions equal to the total number of interior faces in the mesh. If we represent N_{IF} in terms of each interface, we have

$$N_{IF} = \sum_{i=1}^{N_I} N_{IF}^i \tag{2.6}$$

Thus, by Eq. (2.5) and Eq. (2.6), the only appropriate choice is to have a number of interface conditions at each interface equal to the number of faces of the interface, $N_{IC}^i = N_{IF}^i$. All the results up to this point are valid in 2D or 3D.

2.1.1.1 Example Unknown Counts

As an example, consider an $N \times N$ logically rectangular mesh which has $N_F^c = 4$ faces per cell and $N_F^i = 2$ faces per interface. There are $N_C = N^2$ cells and $N_F = 4N^2$ faces in the mesh. Of the $4N^2$ faces, $N_{BF} = 4N$ are boundary faces and $N_{IF} = 4N(N-1)$ are interior faces. There are 2 faces per interface, so there are $N_I = 2N(N-1)$ interfaces. Thus there are

$N_{unk} = N_C + 8N_C = 9N^2$ unknowns (9 per cell) and an equal number of equations:

$$\begin{aligned} N_{eqn} &= \sum_{c=1}^{N_C}(N_F^c + 1) + \sum_{i=1}^{N_I} N_{IF}^i + N_{BF}, \\ &= 5N_C + 2N_I + N_{BF}, \\ &= N^2 + 4N^2 + 4N(N-1) + 4N, \\ &= 9N^2. \end{aligned}$$

In the previous section, we showed that the number of interface condition equations required are the same as the number of faces on an interface, N_F^i. Thus an interface with 2 faces has 2 interface conditions, an interface with 3 faces has 3 interface conditions, and so on. In the following sections, we present and analyze new interface conditions for hanging node interfaces of three faces.

2.1.2 Standard Interface Conditions

On logically rectangular meshes (or any single level mesh) there are $N_F^i = 2$ faces per interface, so we are allowed 2 interface condition equations per interface. It is thus possible to demand continuity of the face-average scalar flux and face-average normal current, respectively,

$$\phi^{if_1} = \phi^{if_2}, \tag{2.7a}$$

$$J^{if_1} = -J^{if_2}, \tag{2.7b}$$

for the pairs of faces if_1 and if_2 of each interface i. Note the negative sign in Eq. (2.7b), because if_1 and if_2 have outward normals pointing in opposite directions, $\vec{n}^{if_1} = -\vec{n}^{if_2}$. We call these *standard interface conditions.*

2.1.3 New Hanging-Node Interface Conditions

Now we consider hanging-node meshes [97,105] with interfaces as in Fig. 2.1(b) which may have only a single hanging node at the midpoint of the larger, master face if_1. The two subfaces are denoted if_2 and if_3. Such interfaces have 3 faces and thus require 3 interface conditions at each interface, i.e. the standard interface conditions of Eq. (2.7) cannot be used.[1] Three possible sets of interface conditions are given below.

2.1.3.1 Strong Current / Weak Scalar Flux Interface Conditions

The strong current interface condition specifies that the current at each subface is equal to the current at the master face. The weak scalar flux interface condition specifies the scalar flux on the master face is equal to an area-weighted sum of scalar fluxes on subfaces.

$$A^{if_1}\phi^{if_1} = A^{if_2}\phi^{if_2} + A^{if_3}\phi^{if_3}, \tag{2.8a}$$

$$J^{if_1} = -J^{if_2}, \tag{2.8b}$$

$$J^{if_1} = -J^{if_3}. \tag{2.8c}$$

[1] It is possible to consider the master face, if_1, as two separate faces of a 5-sided polygon and enforce the standard interface conditions—the disadvantage is more unknowns are introduced into the system. We pursue interface conditions that do not introduce more unknowns into the system by imposing weaker conditions.

2.1.3.2 Strong Scalar Flux / Weak Current Interface Conditions

This condition swaps the scalar flux and current conditions, using a strong interface condition for the scalar flux and a weak one for the current.

$$A^{if_1} J^{if_1} = -A^{if_2} J^{if_2} - A^{if_3} J^{if_3}, \tag{2.9a}$$

$$\phi^{if_1} = \phi^{if_2}, \tag{2.9b}$$

$$\phi^{if_1} = \phi^{if_3}. \tag{2.9c}$$

2.1.3.3 Strong Current / Weak Factor-Weighted Scalar Flux Interface Conditions

This condition is similar to Eq. (2.8a) but for vertical faces (aligned with the $y-$axis) we add E_{xx} QD factors to the weighting and for horizontal faces (aligned with the $x-$axis) we add E_{yy}. This interface condition of Eq. (2.10a) is only valid for faces aligned with the $x-$axis or $y-$axis.

$$E_{xx}^{if_1} A^{if_1} \phi^{if_1} = E_{xx}^{if_2} A^{if_2} \phi^{if_2} + E_{xx}^{if_3} A^{if_3} \phi^{if_3} \quad \text{for vertical faces,} \tag{2.10a}$$

$$E_{yy}^{if_1} A^{if_1} \phi^{if_1} = E_{yy}^{if_2} A^{if_2} \phi^{if_2} + E_{yy}^{if_3} A^{if_3} \phi^{if_3} \quad \text{for horizontal faces,} \tag{2.10b}$$

$$J^{if_1} = -J^{if_2}, \tag{2.10c}$$

$$J^{if_1} = -J^{if_3}. \tag{2.10d}$$

2.2 New Arbitrary Mesh LOQD FV Discretization

In this section, we will present a new discretization of the LOQD equations via a finite volume method. The discretization is based on representing average values at the appropriate centers and then applying a point-wise gradient approximation using a corresponding reciprocal basis [89].

2.2.1 Point Values vs. Average Values

It is essential to show that it is an $O(h^2)$ approximation to consider cell-average and face-average values as point values at cell and face-centers, respectively. Consider the following Taylor series expansion of a variable $\xi(x, y)$,

$$\begin{aligned}\xi(x,y) &= \xi_{0,0} + \left.\frac{\partial \xi}{\partial x}\right|_{0,0} x + \left.\frac{\partial \xi}{\partial y}\right|_{0,0} y + \\ &\quad \left.\frac{\partial^2 \xi}{\partial x \partial y}\right|_{0,0} xy + \frac{1}{2}\left.\frac{\partial^2 \xi}{\partial x^2}\right|_{0,0} x^2 + \frac{1}{2}\left.\frac{\partial^2 \xi}{\partial y^2}\right|_{0,0} y^2 + O(h^3). \end{aligned} \tag{2.11}$$

Without loss of generality, let us shift our coordinate system so the origin is the cell center, $(x^c, y^c) = (0, 0)$, where

$$\begin{aligned} x^c &= \frac{1}{V^c}\iint_c x\, dxdy, \\ y^c &= \frac{1}{V^c}\iint_c y\, dxdy. \end{aligned} \tag{2.12}$$

Applying the averaging operator, $\frac{1}{V^c}\iint_c \circ dxdy$, to Eq. (2.11) over cell c yields

$$\begin{aligned}\xi^c &= \xi_{0,0} + \left.\frac{\partial \xi}{\partial x}\right|_{0,0} x^c + \left.\frac{\partial \xi}{\partial y}\right|_{0,0} y^c + \\ &\quad \left.\frac{\partial^2 \xi}{\partial x \partial y}\right|_{0,0} \langle xy\rangle^c + \frac{1}{2}\left.\frac{\partial^2 \xi}{\partial x^2}\right|_{0,0} \langle x^2\rangle^c + \frac{1}{2}\left.\frac{\partial^2 \xi}{\partial y^2}\right|_{0,0} \langle y^2\rangle^c + O(h^3). \end{aligned} \tag{2.13}$$

Because $(x^c, y^c) = (0, 0)$, first-order terms disappear and we have the following expression for the leading order error in the difference between the average value and the center value $\xi_{0,0}$,

$$\xi^c - \xi_{0,0} = \left.\frac{\partial^2 \xi}{\partial x \partial y}\right|_{0,0} \langle xy\rangle^c + \frac{1}{2}\left.\frac{\partial^2 \xi}{\partial x^2}\right|_{0,0} \langle x^2\rangle^c + \frac{1}{2}\left.\frac{\partial^2 \xi}{\partial y^2}\right|_{0,0} \langle y^2\rangle^c. \tag{2.14}$$

The second-order terms are defined as

$$\langle xy\rangle^c = \frac{1}{V^c}\iint_c x\,y\;dxdy, \tag{2.15a}$$

$$\langle x^2\rangle^c = \frac{1}{V^c}\iint_c x^2\;dxdy, \tag{2.15b}$$

$$\langle y^2\rangle^c = \frac{1}{V^c}\iint_c y^2\;dxdy. \tag{2.15c}$$

Note that on rectangles, the term $\langle xy\rangle^c$ is identically zero. For non-orthogonal quadrilaterals, the term $\langle xy\rangle^c$ may be positive or negative. The other terms are positive, $\langle x^2\rangle^c > 0$ and $\langle y^2\rangle^c > 0$, on any quadrilateral.

It is possible to transform x and y according to $x = h^c\eta/2$ and $y = h^c\nu/2$ where η and ν are not mesh-dependent and the mean cell width is $h^c = \sqrt{V^c}$. This transformation leads to Eq. (2.15c) represented as

$$\langle xy\rangle^c = (h^c)^2\iint_c \eta\,\nu\;d\eta d\nu = O(h^2), \tag{2.16a}$$

$$\langle x^2\rangle^c = (h^c)^2\iint_c \eta^2\;d\eta d\nu = O(h^2), \tag{2.16b}$$

$$\langle y^2\rangle^c = (h^c)^2\iint_c \nu^2\;d\eta d\nu = O(h^2). \tag{2.16c}$$

Thus Eq. (2.14) reduces to

$$\xi^c - \xi_{0,0} = O(h^2), \tag{2.17}$$

that is, cell-average and cell-center values are identical to $O(h^2)$. The same argument may be made for face-average and face-center values by writing another Taylor series centered about the face. This result also extends trivially to 3D.

2.2.2 Discretization Details

We make the second-order in space $O(h^2)$ approximation that the face-average value of the tensor divergence terms $\frac{\partial}{\partial\beta}\phi E_{\alpha\beta}$ may be evaluated at the face-center. Using the corresponding reciprocal basis as shown in [89] for a diffusion discretization, we can write a differencing expression for the derivative in face-average quantity ξ^f in terms of other cell-local values, i.e. the cell-average ξ^c and all other face-average $\xi^{f'}$ in the cell,

$$\frac{\partial}{\partial\beta}\xi^f = (\xi^f - \xi^c)G^f_\beta + \sum_{f'\in c}\xi^{f'}H^{ff'}_\beta + O(h^2), \tag{2.18}$$

where G^f_β and $H^{ff'}_\beta$ are the β-th component of purely geometric factors, most easily defined in vector notation as follows,

$$\vec{G}^f = \frac{\vec{A}^f}{\vec{A}^f\cdot\vec{R}^f}, \tag{2.19a}$$

$$\vec{H}^{ff'} = \frac{1}{V^c}\left(\vec{A}^{f'} - \vec{A}^f\frac{\vec{R}^f\cdot\vec{A}^{f'}}{\vec{R}^f\cdot\vec{A}^f}\right), \tag{2.19b}$$

where $\vec{A}^f$ are face area vectors, $\vec{A}^f = \vec{n}^f A^f$, and $\vec{R}^f$ are cell-to-face vectors, $\vec{R}^f = \vec{r}^c - \vec{r}^f$, where $\vec{r}^c$ and $\vec{r}^f$ are the coordinates of the cell-center and face-center, respectively. The approximation of Eq. (2.18) may be interpretted as using the difference between face-average value ξ^f and cell-average value ξ^c to approximate the projection of the gradient at face f in the direction of $\vec{R}^f$. Values at other faces $\xi^{f'}$ are used to estimate the projection of the gradient at face f in the transverse directions. The expression simplifies considerably when face area vectors $\vec{A}^f$ and cell-to-face vectors $\vec{R}^f$ are aligned, for example, with parallelograms. In our case, we have $\xi^f = E^f_{\alpha\beta}\phi^f$

and our new cell-local discretization is given as

$$\sum_{f\in c} J^f A^f + \sigma_a \phi^c V^c = q^c_{ext} V^c \tag{2.20a}$$

for each cell c,

$$\sum_{\alpha=x,y} n^f_\alpha \sum_{\beta=x,y} \left((\phi^f E^f_{\alpha\beta} - \phi^c E^c_{\alpha\beta}) G^f_\beta + \sum_{f'\in c} \phi^{f'} E^{f'}_{\alpha\beta} H^{ff'}_\beta \right) + \sigma_t J^f = 0 \tag{2.20b}$$

for each face $f \in$ c.

2.2.3 LOQD Discretizations from Literature

In this section we present some LOQD discretizations from the literature, cast into our general FV framework to facilitate comparison.

2.2.3.1 GGK Discretization on Orthogonal Meshes

The discretization due to Gol'din, Gol'dina, and Kolpakov (GGK) [24] is derived by dividing the quadrilateral into 4 half-cells, and integrating the vector first moment Eq. (1.4b) equation over each half-cell. The four faces of a rectangular cell are denoted $f = B, R, T, L$ for the bottom, right, top, and left sides respectively. The cell size is h_x by h_y in the x-direction and y-directions, respectively. The tensor divergence term then is evaluated in terms of line integrals over the edges of the half-cell. Consider the right R half cell for example. One proceeds to evaluate the integrals counterclockwise over face R for a distance h_y, then half of face T for a distance $h_x/2$, through the center of the cell for a distance h_y, then finally over half of face B for a distance $h_x/2$. Finally, the approximation is made that the half-cell average current is the face-average current. As an

alternative, we use Gauss's divergence theorem for tensors over the right half-cell, denoted c^R,

$$\begin{aligned} \phi^R\Big(\mathrm{E}^R\cdot\vec{A}^R\Big) &+ \frac{1}{2}\phi^{RT}\Big(\mathrm{E}^{RT}\cdot\vec{A}^T\Big)+ \\ \phi^{RC}\Big(\mathrm{E}^{RC}\cdot\vec{A}^{RC}\Big) &+ \frac{1}{2}\phi^{RB}\Big(\mathrm{E}^{RB}\cdot\vec{A}^B\Big)+\sigma_t\vec{J}^{c^R}V^{c^R}=0, \end{aligned} \tag{2.21}$$

where ϕ^R is the face-average scalar flux on the right face, E^R is the appropriately defined QD tensor on the right face, $\vec{A}^R$ is the area vector of the right face, ϕ^{RT} is the average scalar flux over the right half of the top face, and E^{RT} is the appropriately defined average QD tensor of the right half of the top face. Similarly, RC denotes the average quantities through the center (C) of the cell which joins top and bottom faces, RB denotes the right half of the bottom face quantities, RT denotes the right half of the top face quantities. The volume of the right half-cell is $V^{c^R}=h_xh_y/2$ and $\vec{J}^{c^R}$ denotes the right half-cell average current. The following approximations are made in [24] in order to arrive at the set of unknowns:

1. The right half-cell average current is equal to the right face-average current, $\vec{J}^{c^R}\approx\vec{J}^R$.

2. The portions of the right-top RT and right-bottom RB quantities may be taken to be equal to the average quantities for the entire top and bottom faces, $\mathrm{E}^{RT}\approx\mathrm{E}^T$, $\mathrm{E}^{RB}\approx\mathrm{E}^B$, $\phi^{RT}\approx\phi^T$, $\phi^{RB}\approx\phi^B$.

3. The right half-cell center RC quantities may be taken to be cell-average c quantities, $\mathrm{E}^{RC}\approx\mathrm{E}^c$, $\phi^{RC}\approx\phi^c$.

Utilizing these approximations and projecting the result onto the right face normal, $\vec{n}^R$, we get

$$\begin{aligned} \phi^R\Big(\mathrm{E}^R\cdot\vec{A}^R\Big)\cdot\vec{n}^R &+ \frac{1}{2}\phi^T\Big(\mathrm{E}^T\cdot\vec{A}^T\Big)\cdot\vec{n}^R+ \\ \phi^c\Big(\mathrm{E}^c\cdot\vec{A}^{RC}\Big)\cdot\vec{n}^R &+ \frac{1}{2}\phi^B\Big(\mathrm{E}^B\cdot\vec{A}^B\Big)\cdot\vec{n}^R+\sigma_tJ^R\frac{h_xh_y}{2}=0. \end{aligned}$$

In the orthogonal case, $\vec{n}^R = (1,0)^t$, $\vec{A}^R = \vec{n}^R\ h_y$, $\vec{A}^T = (0,1)^t\ h_x$, $\vec{A}^{RC} = (-1,0)^t\ h_y$, $\vec{A}^B = (0,-1)^t\ h_x$, where t denotes the transpose to avoid confusion with T for top, which leads to

$$\phi^R\Big(E^R_{xx}h_y\Big) + \phi^T\Big(E^T_{xy}\frac{h_x}{2}\Big) - \phi^c\Big(E^c_{xx}h_y\Big) - \phi^B\Big(E^B_{xy}\frac{h_x}{2}\Big) + \sigma_t J^R\frac{h_x h_y}{2} = 0.$$

Division by $h_x h_y/2$ leads us to the "GGK" approximation of right face first moment equation for orthogonal cells,

$$\frac{2}{h_x}(\phi^c E^c_{xx} - \phi^R E^R_{xx}) + \frac{1}{h_y}(\phi^T E^T_{xy} - \phi^B E^B_{xy}) + \sigma_t J^R = 0. \tag{2.22}$$

2.2.3.2 GGK Discretization for Skewed Quadrilateral Meshes

A discretization for skewed quadrilaterals can be extended from the GGK discretization for orthogonal quadrilaterals by using the appropriate half-cell volumes and center area-vectors (e.g. V^{RC} and $\vec{A}^{RC}$, respectively for the right face) for a skewed quadrilateral. If the top face-center has coordinates $\vec{r}^T = (x^T, y^T)^t$ and the bottom face-center has coordinates $\vec{r}^B = (x^B, y^B)^t$, then the area vector which represents the completion of the half-cell boundary integration for the right half cell by joining points $\vec{r}^T$ and $\vec{r}^B$ is

$$\vec{A}^{RC} = \begin{pmatrix} y^B - y^T \\ x^T - x^B \end{pmatrix}. \tag{2.23}$$

The volume of the right half-cell V^{c^R} is calculated by the winding formula in Eq. (1.18) with the following sequence of points: $\vec{r}^{TR}$, $\vec{r}^T$, $\vec{r}^B$, and $\vec{r}^{BR}$. After taking the dot-product with the face-normal $\vec{n}^R$, the resulting first moment equation for the skewed quadrilateral GGK discretization

is

$$\left(\phi^R\left(\mathrm{E}^R\cdot\vec{A}^R\right) \;+\; \frac{1}{2}\phi^T\left(\mathrm{E}^T\cdot\vec{A}^T\right)+\right.$$
$$\left.\phi^c\left(\mathrm{E}^c\cdot\vec{A}^{RC}\right) \;+\; \frac{1}{2}\phi^B\left(\mathrm{E}^B\cdot\vec{A}^B\right)\right)\cdot\frac{\vec{n}^R}{V^{c^R}} \;+\; \sigma_t J^R = 0. \tag{2.24}$$

Another option exists for calculating the area vector and half-cell volume. If one considers the half-cell integral as proceeding from top face-center, to cell-center, to bottom face-center, one has the following approximation,

$$\phi^R\left(\mathrm{E}^R\cdot\vec{A}^R\right) \;+\; \frac{1}{2}\phi^T\left(\mathrm{E}^T\cdot\vec{A}^T\right)+$$
$$\phi^c\left(\mathrm{E}^c\cdot\vec{A}^{RC1}\right) \;+\; \phi^c\left(\mathrm{E}^c\cdot\vec{A}^{RC2}\right)+\frac{1}{2}\phi^B\left(\mathrm{E}^B\cdot\vec{A}^B\right)+\sigma_t\vec{J}^R V^{c^R}=0,$$

with two area vectors now,

$$\vec{A}^{RC1} \;=\; \begin{pmatrix} y^C-y^T \\ x^T-x^C \end{pmatrix},$$

$$\vec{A}^{RC2} \;=\; \begin{pmatrix} y^B-y^C \\ x^C-x^B \end{pmatrix},$$

and the half-cell volume calculated from Eq. (1.18) and the following sequence of points that includes the cell-center: $\vec{r}^{TR}$, $\vec{r}^T$, $\vec{r}^c$, $\vec{r}^B$, and $\vec{r}^{BR}$. The two area vectors actually lead to the same approximation, $\vec{A}^{RC}=\vec{A}^{RC1}+\vec{A}^{RC2}$, so the alternate GGK discretization only has a different V^{c^R}. We have observed through numerical experiments that the definition of half-cell volume *which does not include the center of the cell* yields superior accuracy, so we will use it for the remainder.

2.2.3.3 AK Discretization for Skewed Quadrilateral Meshes

The discretization of Aristova and Kolpakov (AK) [68] uses the same half cells as the GGK discretization in its derivation but, considering the right half-cell again, expresses the integration of ϕ over the top (T) and bottom (B) faces as the integration of a linear function based on ϕ^R and ϕ^c, instead of using ϕ^T and ϕ^B as in the GGK discretization. The authors report increased accuracy and robustness over the GGK discretization for problems with shocks and other features that lead to large discontinuities and/or gradients in the solution [68]. The first moment equation with the AK discretization is as follows,

$$\begin{aligned}\Bigg(\phi^R\Big(\mathrm{E}^R\cdot\vec{A}^R\Big) \quad &+ \quad \frac{1}{4}(\phi^R+\phi^c)\Big(\mathrm{E}^B\cdot\vec{A}^B\Big)+\\ \phi^c\Big(\mathrm{E}^c\cdot\vec{A}^{RC}\Big) \quad &+ \quad \frac{1}{4}(\phi^R+\phi^c)\Big(\mathrm{E}^T\cdot\vec{A}^T\Big)\Bigg)\cdot\frac{\vec{n}^R}{V^{c^R}} \quad + \quad \sigma_t J^R=0. \end{aligned} \tag{2.25}$$

The balance equation is the same as in Eq. (2.2a) but obviously, the first moment equation in Eq. (2.25) is different. As opposed to Eq. (2.22), it leads to a first-moment equation on each face which is only dependent on one face-average scalar flux (e.g. ϕ^R), one face-average normal current (e.g. J^R), and the cell-average scalar flux ϕ^c.

2.2.4 Discretization Comparisons

JM discretization geometry factors for an (orthogonal) rectangular cell are shown in Table 2.1 and Table 2.2. When substituted into the moment equation of Eq. (2.20b), these geometry factors for the right face, $f=R$, lead to

$$\frac{2}{h_x}(\phi^c E^C_{xx}-\phi^R E^R_{xx})+\frac{1}{h_y}(\phi^T E^T_{xy}-\phi^B E^B_{xy})+\sigma_t J^R=0, \tag{2.26}$$

which is the same expression as in the GGK discretization. Thus, for rectangular cells, the half-cell discretization described in 2.2.3.1 [24] may be interpretted as our FV framework with the tensor divergence approximated as in the JM discretization. As opposed to the multiple approximations necessary to arrive at the GGK approximation of Eq. (2.26) with the half-cell framework, we only need Eq. (2.18) on rectangles. If the cross terms are zero, i.e. $E_{xy} = 0$, both the GGK and JM

Table 2.1: Rectangular Cell $\vec{G}^f$ Geometry Factors.

f	$\vec{G}^f$
R	$(\,2/h_x\,,\,0\,)^t$
T	$(\,0\,,\,2/h_y\,)^t$
L	$(\,-2/h_x\,,\,0\,)^t$
B	$(\,0\,,\,-2/h_y\,)^t$

Table 2.2: Rectangular Cell $\vec{H}^{ff'}$ Geometry Factors.

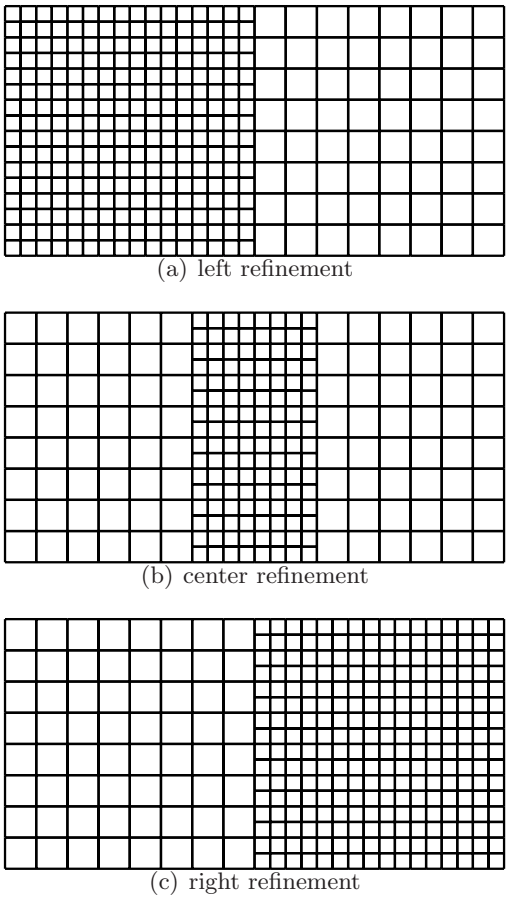
	$\vec{H}^{ff'}$			
f/f'	R	T	L	B
R	$(\,0\,,\,0\,)^t$	$(\,0\,,\,1/h_y\,)^t$	$(\,0\,,\,0\,)^t$	$(\,0\,,\,-1/h_y\,)^t$
T	$(\,1/h_x\,,\,0\,)^t$	$(\,0\,,\,0\,)^t$	$(\,-1/h_x\,,\,0\,)^t$	$(\,0\,,\,0\,)^t$
L	$(\,0\,,\,0\,)^t$	$(\,0\,,\,1/h_y\,)^t$	$(\,0\,,\,0\,)^t$	$(\,0\,,\,-1/h_y\,)^t$
B	$(\,1/h_x\,,\,0\,)^t$	$(\,0\,,\,0\,)^t$	$(\,-1/h_x\,,\,0\,)^t$	$(\,0\,,\,0\,)^t$

discretizations on orthogonal meshes correspond to a five-point finite-difference diffusion stencil.

2.3 Results

Here we present tests of new interface conditions and our new JM discretization for the LOQD equations. We present analytic tests via manufactured solutions as well as numerical tests in which we must generate the QD factors, $E_{\alpha\beta}[\psi]$ and $C[\psi]$. Note that generating the QD factors requires a transport solver for arbitrary quadrilateral meshes which we will defer discussion of to Chapter 3.

2.3.1 Discontinuous Media Test



(a) left refinement

(b) center refinement

(c) right refinement

Figure 2.2: Discontinuous Media Test: Meshes.

The domain is a 8cm × 4cm rectangle with discontinuous total cross sections for the left and right sides: $\sigma_t = 1\text{cm}^{-1}$ for $x \leq 4\text{cm}$ and $\sigma_t = 10\text{cm}^{-1}$ for $x \geq 4\text{cm}$. There is no scattering, vacuum BC on all sides, $J_{IN} = \phi_{IN} = 0$, and the external source is $q_{ext} = 1^{\text{n}}/_{\text{cm}^3\text{s}}$ in a band, $2\text{cm} \leq x \leq 3\text{cm}$.

Hanging-node meshes are generated by applying refinement to a region of an single-level orthogonal mesh: left-side refinement as in Fig. 2.2(a), right-side refinement as in Fig. 2.2(c), and center refinement as in Fig. 2.2(b). Right and left refinement increases the number of cells by a factor of 2.5, center refinement increases the number of cells by a factor of 1.75, the standard refinement increases the number of cells by a factor of 4.

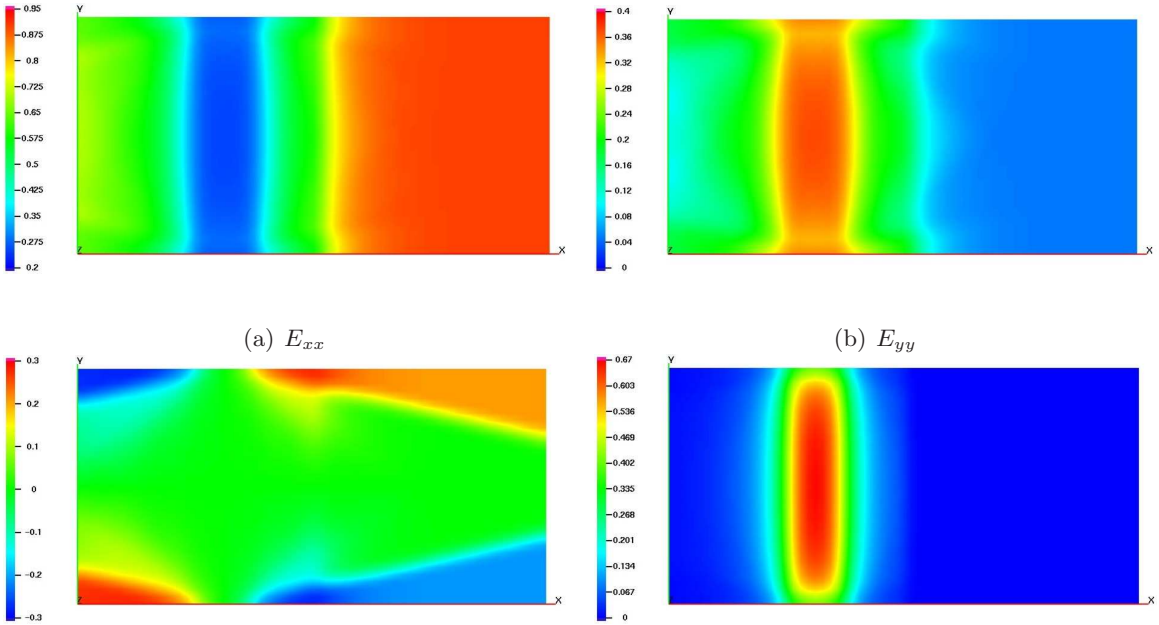
This test is used to compare the Strong Current / Weak Scalar Flux interface conditions in Eq. (2.8a) (denoted WF) and the Strong Scalar Flux / Weak Current interface conditions in Eq. (2.9a) (denoted WC). (The Strong Current / Weak Factor-Weighted Scalar Flux Interface Conditions led to very poor results and will not be considered further.)

2.3.1.1 Data

The reference solution and QD factors, as shown in Fig. 2.3, are from a fine mesh solution generated with SCSB, which is very accurate for problems with no scattering (see Chapter 3). We use the SCSB method described in Chapter 3 for the high-order QD problem in this test.

The scalar flux at $x = 4\text{cm}$ along the discontinuous media is shown in Fig. 2.4-Fig. 2.9 for all possible combinations of interface conditions (WC or WF) and meshes (Left, Right, and Center Refinement), with single-level meshes which use standard interface conditions (Std) for comparison. In the figures, refined meshes have the same symbol (e.g. triangle) as the single-level orthogonal

mesh they are created from. The number of cells is reported to easily compare various meshes. The scalar flux along the midplane, $y = 2$cm is shown in Fig. 2.10 and around the peak in Fig. 2.11 on a 64×64 orthogonal mesh, comparing the LOQD solutions to bilinear discontinuous (BLD) and short characteristics with subcell balances (SCSB) transport discretizations.



(a) E_{xx} (b) E_{yy}

(c) E_{xy} (d) ϕ

Figure 2.3: Discontinuous Media Test: QD Factors and fine-mesh scalar flux solution.

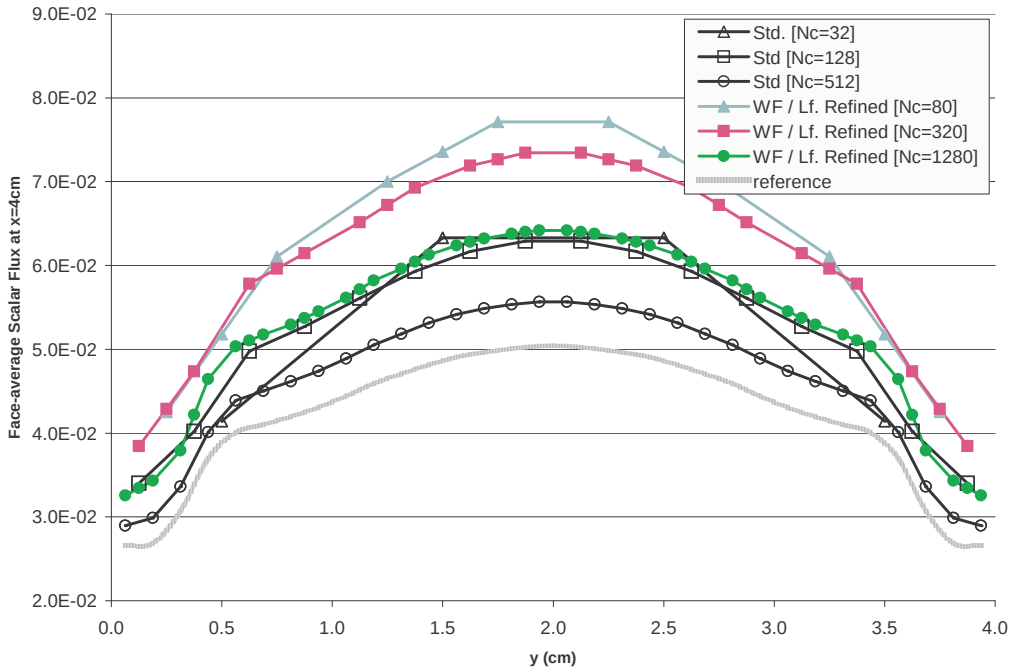


Figure 2.4: Discontinuous Media Test: Scalar flux along centerline $x = 4$cm for Strong Current / Weak Scalar Flux (WF) conditions for meshes with left refinement.

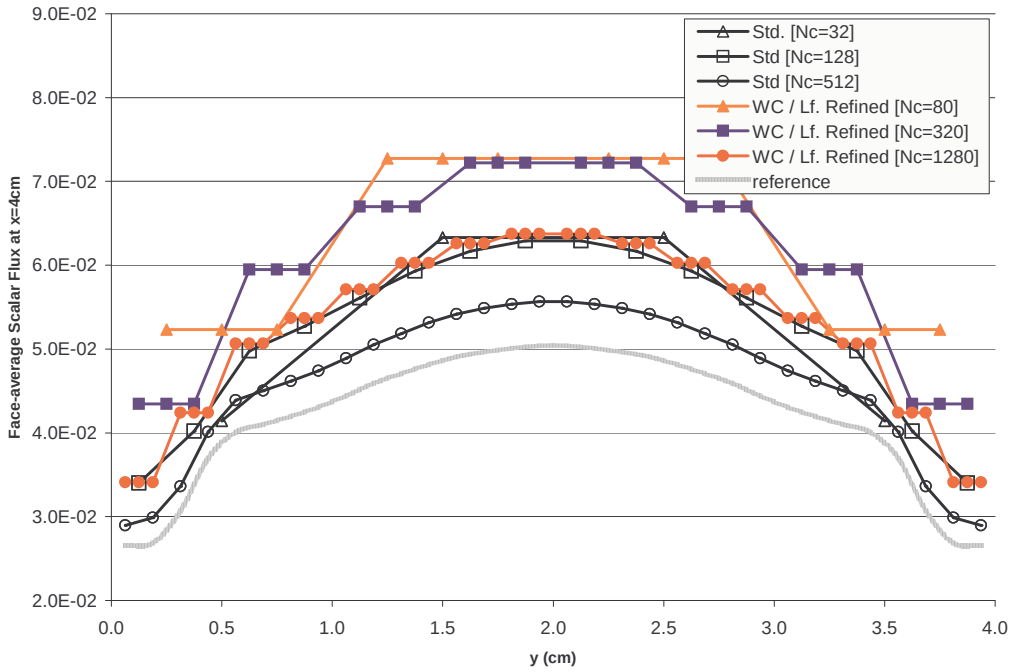


Figure 2.5: Discontinuous Media Test: Scalar flux along centerline $x = 4$cm for Strong Scalar Flux / Weak Current (WC) conditions for meshes with left refinement.

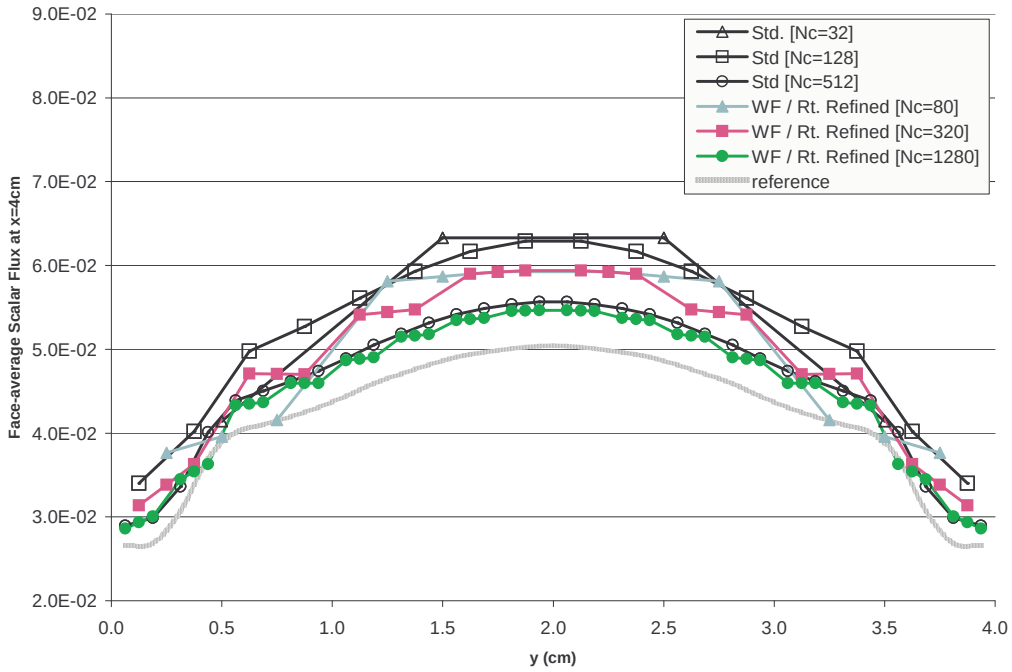


Figure 2.6: Discontinuous Media Test: Scalar flux along centerline $x = 4$cm for Strong Current / Weak Scalar Flux (WF) conditions for meshes with right refinement.

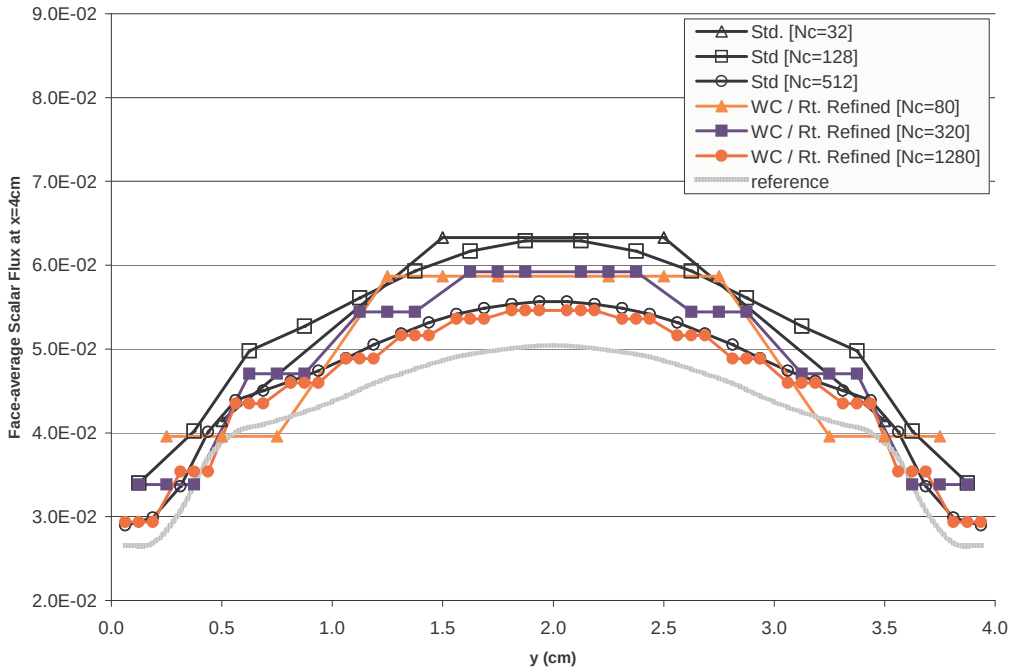


Figure 2.7: Discontinuous Media Test: Scalar flux along centerline $x = 4$cm for Strong Scalar Flux / Weak Current (WC) conditions for meshes with right refinement.

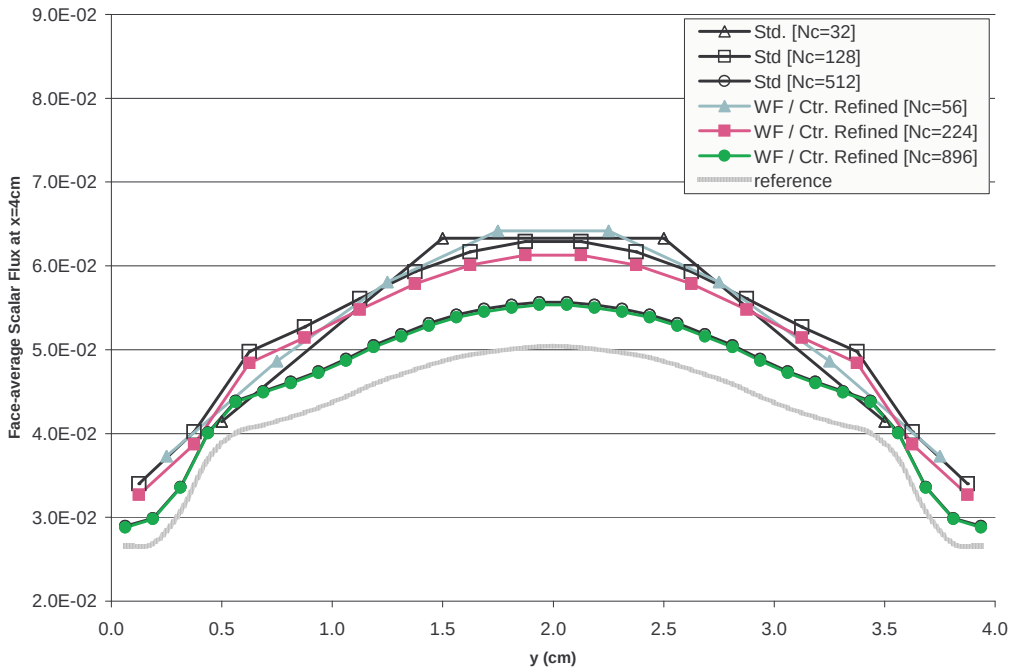


Figure 2.8: Discontinuous Media Test: The scalar flux along centerline $x = 4$cm for Strong Current / Weak Scalar Flux (WF) conditions for meshes with center refinement.

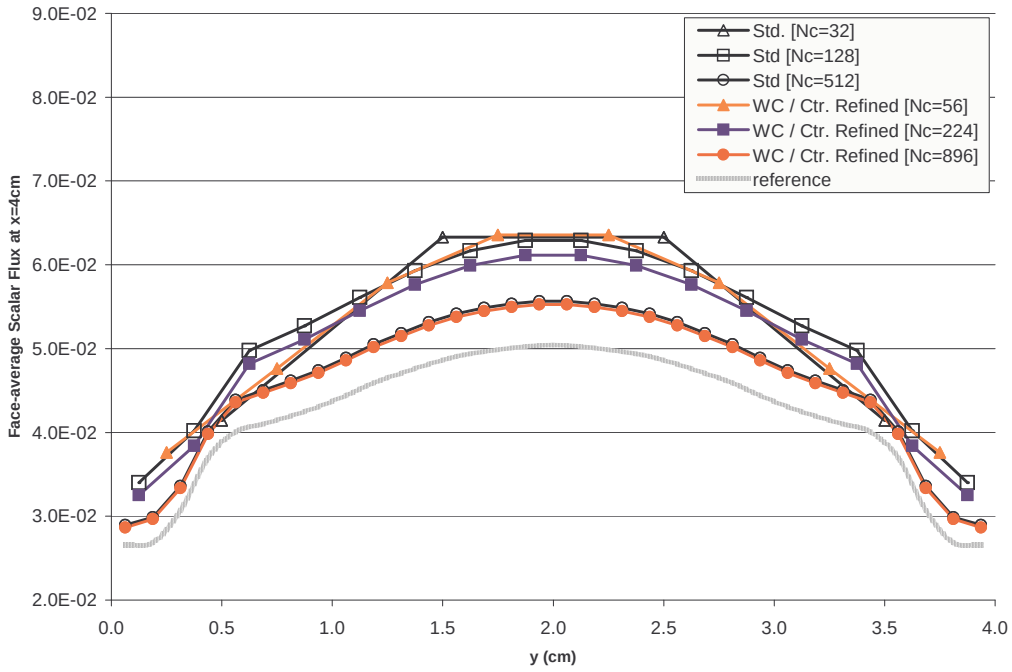


Figure 2.9: Discontinuous Media Test: The scalar flux along centerline $x = 4$cm for Strong Scalar Flux / Weak Current (WC) conditions for meshes with center refinement.

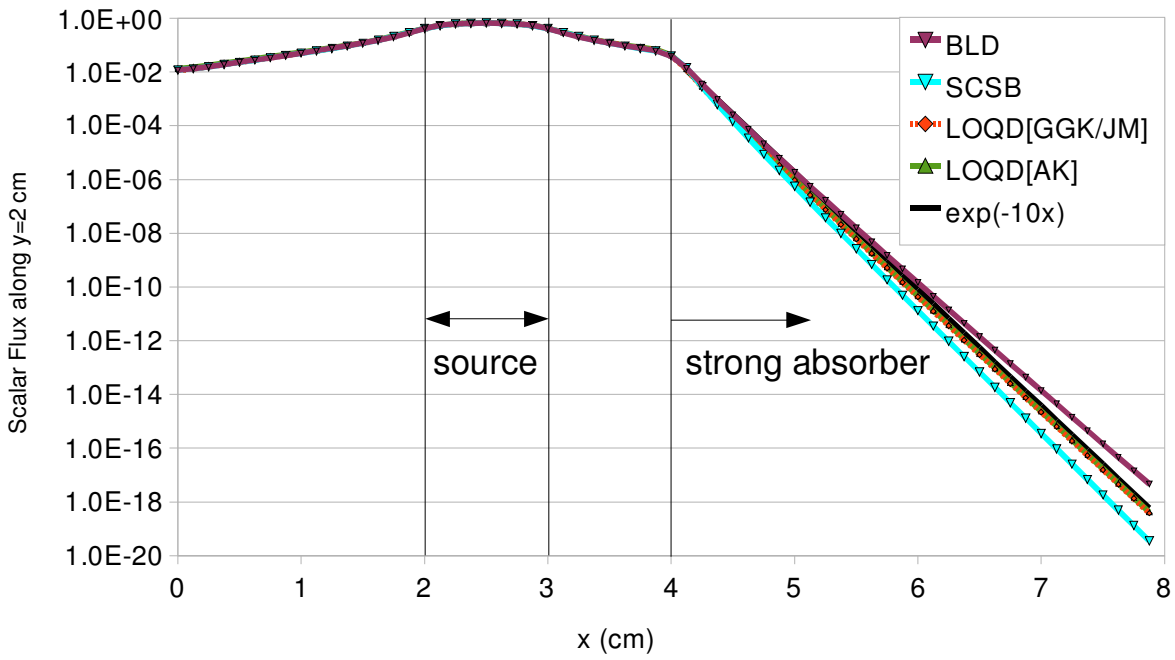


Figure 2.10: Discontinuous Media Test: Scalar flux along midplane, $y = 2$cm with 64×64 orthogonal mesh.

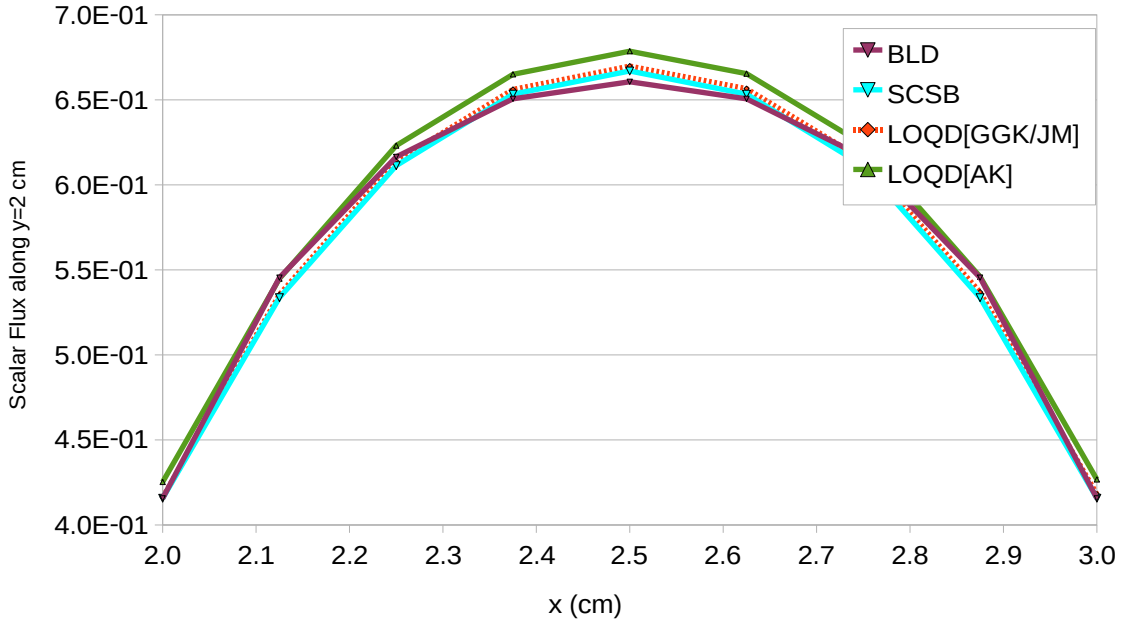


Figure 2.11: Discontinuous Media Test: Scalar flux around peak 2cm $\leq x \leq$ 3cm with 64×64 orthogonal mesh.

2.3.1.2 Analysis

First let us compare the various interface conditions on hanging-node meshes. On left-refined meshes as in Fig. 2.2(a) the scalar flux along the material discontinuity at $x = 4$cm in Fig. 2.4 and Fig. 2.5 shows that left refinement leads to no improvement in the solution. One can also see the weak scalar flux conditions (WF) in Fig. 2.4 lead to scalar fluxes on interfaces that have a linear shape whereas the weak current conditions (WC) shown in Fig. 2.5 lead to flat scalar flux distributions for each interface.

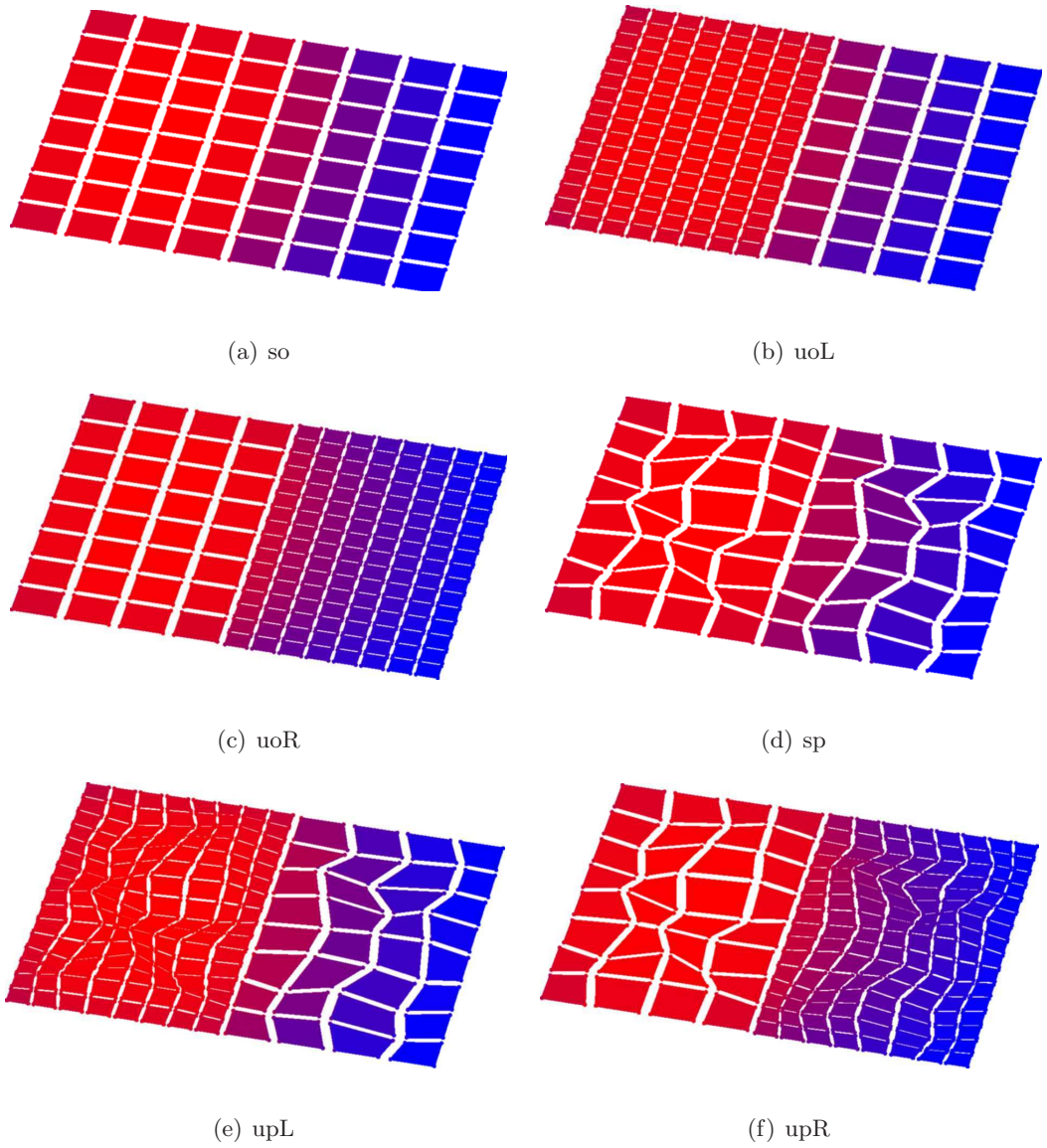
On right-refined meshes, where cells are refined in the optically thick right-side of the problem, one sees reduction in error from refinement up to the finest mesh shown in Fig. 2.6 and Fig. 2.7. Specifically, the right-refined mesh has error close to the error on a mesh with refinement of both sides. On the finest meshes shown, the coarseness of the source region is probably limiting the accuracy and would need to be refined as well. Because there is no scattering, information strictly propagates from the source on the left, through coarse cells across the interface to finer cells. Under these conditions (information propagating from coarse to fine cells), a much flatter representation of the scalar flux is observed than in the left-refined case, so the WF and WC conditions do not show much difference.

Finally, we consider center-refined meshes, where cells are refined along the material discontinuity, but not in the source region. Solutions on these meshes, as shown in Fig. 2.8 and Fig. 2.9 are similar to the right-refined meshes, with advantage of refinement shown for the two coarser meshes. Away from hanging-node interfaces, both the WF and WC conditions appear appropriate. Overall, the Strong Current / Weak Scalar Flux (WF) conditions of Eq. (2.8a) lead to much better shapes of the scalar flux for the types of refinements tested.

In Fig. 2.10, the BLD solution is shown to underapproximate the attenuation in the strongly absorbing region. Approximating strong attenuation on coarse meshes is difficult for a linear discontinuous method, but is a task where characteristic methods excel. The LOQD solution, with factors calculated from SCV, also underapproximates the attenuation, showing close to $\exp(-10x)$ attenuation—which would occur in a slab with uniform incident flux and no scattering. As opposed to a slab, this problem is finite in the $\pm y$ directions, which means there is leakage and the scalar flux should decrease faster than $\exp(-10x)$. Point-wise long characteristics calculations (which are exact with no scattering) confirm that the SCSB solution decreases at the correct rate. All methods perform similarly in the peak region Fig. 2.11.

2.3.2 Discontinuous Source Test

This test has a discontinuous external source on a 5cm $\times$ 3cm rectangular domain with homogeneous material properties, $\sigma_t = 1\text{cm}^{-1}$ with scattering, $\sigma_s = 0.5\text{cm}^{-1}$. There are vacuum BC on all sides, $\phi_{IN} = J_{IN} = 0$. The external source on the left is $q_{ext} = 0.5^{\text{n}}/_{\text{cm}^3\text{s}}$ for $x \leq 2.5$cm and elsewhere $q_{ext} = 0$. We present results for a variety of meshes: the mesh naming scheme is as follows: single-level orthogonal is so; single-level perturbed (by randomizations) is sp; unstructured, orthogonal hanging-node is uoL and uoR for left and right refinement, respectively; and upL and upR are the perturbed versions of uoL and upL, respectively. Fig. 2.12 shows the meshes derived from a base 8×8 mesh. Mesh randomizations are 20%. The scalar flux is converged to a relative tolerance of $\epsilon_\phi = 10^{-8}$. With this test, we compare three FV LOQD discretizations: the GGK and AK from literature, and our new JM discretization. For the transport discretization, we use a short characteristics discretization with unknowns at vertices and min/max monotonization (SCV)



(a) so (b) uoL

(c) uoR (d) sp

(e) upL (f) upR

Figure 2.12: Discontinuous Source Test: Meshes derived from a base 8×8 mesh.

2.3.2.1 Data

For each of the three FV variants (GGK, AK, and JM), we calculate numerical convergence orders for the left-side scalar flux via Eq. (2.27a), right-side scalar flux via Eq. (2.27b), and average exiting current at the top right corner via Eq. (2.27c).

$$\phi_{left} = \int_{0.0}^{2.5} dx \int_{0.0}^{3.0} dy \;\; \phi(x,y) \tag{2.27a}$$

$$\phi_{right} = \int_{2.5}^{5.0} dx \int_{0.0}^{3.0} dy \;\; \phi(x,y) \tag{2.27b}$$

$$J_{ur} = \frac{1}{4.0}\left(\int_{2.5}^{5.0} dx \; J(x,3.0) + \int_{1.5}^{3.0} dy \; J(5.0,y)\right) \tag{2.27c}$$

The results for ϕ_{left}, ϕ_{right}, and J_{ur} are shown for various meshes in Table 2.4, Table 2.5, and Table 2.6, respectively. For hanging-node meshes, the mean cell-widths for the left and right sides are shown as h_L/h_R. The respective convergence orders are presented in Table 2.3, calculated from the Aitken process as in Eq. (1.23). The scalar flux at the discontinuous source interface at $x = 2.5$cm is shown in Fig. 2.13, Fig. 2.14, and Fig. 2.15.

Table 2.3: Discontinuous Source Test: Numerical convergence orders.

	GGK			AK			JM		
Mesh	J_{ur}	ϕ_{left}	ϕ_{right}	J_{ur}	ϕ_{left}	ϕ_{right}	J_{ur}	ϕ_{left}	ϕ_{right}
so	1.9	2.3	2.3	2.1	2.3	2.2	1.9	2.3	2.3
sp	$*$	1.2	1.3	2.4	0.7	0.5	2.8	1.9	1.9
uoL	1.5	1.9	2.0	1.8	2.0	1.9	1.5	1.9	2.0
uoR	1.4	2.2	1.3	1.6	2.2	2.2	1.4	2.2	1.3
upL	0.8	1.5	1.3	1.3	2.6	1.9	2.0	1.4	1.4
upR	0.9	2.2	2.3	1.2	2.8	2.2	1.3	1.8	1.8

$*$ Aitken process failed

Table 2.4: Discontinuous Source Test: Results for ϕ_{left}.

Mesh Type	h(cm)	GGK	AK	JM
so	1.936	9.905	10.097	9.905
	0.968	8.909	9.118	8.909
	0.484	8.413	8.564	8.413
	0.242	8.266	8.396	8.266
	0.121	8.231	8.356	8.231
	0.061	8.224	8.347	8.224
sp	1.936	9.905	10.097	9.905
	0.968	8.887	9.132	8.912
	0.484	8.382	8.531	8.447
	0.242	8.184	8.290	8.284
	0.121	8.129	8.249	8.233
	0.061	8.105	8.223	8.223
uoL	0.968/1.936	9.105	9.374	9.105
	0.484/0.968	8.515	8.692	8.515
	0.242/0.484	8.304	8.442	8.304
	0.121/0.242	8.242	8.369	8.242
	0.061/0.121	8.226	8.350	8.226
uoR	1.936/0.968	9.751	9.912	9.751
	0.968/0.484	8.826	9.011	8.826
	0.468/0.242	8.376	8.517	8.376
	0.242/0.121	8.254	8.381	8.254
	0.121/0.061	8.228	8.353	8.228
upL	0.968/1.936	9.105	9.374	9.105
	0.484/0.968	8.488	8.690	8.512
	0.242/0.484	8.283	8.394	8.326
	0.121/0.242	8.201	8.256	8.254
	0.061/0.121	8.171	8.232	8.229
upR	1.936/0.968	9.751	9.912	9.751
	0.968/0.484	8.820	9.031	8.833
	0.468/0.242	8.352	8.483	8.395
	0.242/0.121	8.191	8.278	8.264
	0.121/0.061	8.156	8.250	8.228

Table 2.5: Discontinuous Source Test: Results for ϕ_{right}.

Mesh Type	h(cm)	GGK	AK	JM
so	1.936	0.593	0.612	0.593
	0.968	0.893	0.903	0.893
	0.484	1.100	1.124	1.100
	0.242	1.173	1.205	1.173
	0.121	1.192	1.227	1.192
	0.061	1.195	1.231	1.195
sp	1.936	0.593	0.612	0.593
	0.968	0.904	0.892	0.914
	0.484	1.110	1.134	1.076
	0.242	1.210	1.259	1.157
	0.121	1.242	1.291	1.186
	0.061	1.255	1.314	1.194
uoL	0.968/1.936	0.713	0.640	0.713
	0.484/0.968	0.996	0.986	0.996
	0.242/0.484	1.135	1.155	1.135
	0.121/0.242	1.181	1.213	1.181
	0.061/0.121	1.193	1.228	1.193
uoR	1.936/0.968	0.736	0.802	0.736
	0.968/0.484	0.984	1.018	0.984
	0.468/0.242	1.139	1.172	1.139
	0.242/0.121	1.185	1.219	1.185
	0.121/0.061	1.194	1.230	1.194
upL	0.968/1.936	0.713	0.640	0.713
	0.484/0.968	1.016	0.987	1.022
	0.242/0.484	1.145	1.170	1.118
	0.121/0.242	1.205	1.264	1.169
	0.061/0.121	1.228	1.289	1.188
upR	1.936/0.968	0.736	0.802	0.736
	0.968/0.484	0.981	1.001	0.986
	0.468/0.242	1.144	1.184	1.126
	0.242/0.121	1.209	1.273	1.178
	0.121/0.061	1.222	1.293	1.192

Table 2.6: Discontinuous Source Test: Results for J_{ur}.

Mesh Type	h(cm)	GGK	AK	JM
so	1.936	0.02237	0.01436	0.02237
	0.968	0.02705	0.01884	0.02705
	0.484	0.02998	0.02198	0.02998
	0.242	0.03141	0.02349	0.03141
	0.121	0.03194	0.02398	0.03194
	0.061	0.03209	0.02409	0.03209
sp	1.936	0.02237	0.01436	0.02237
	0.968	0.02787	0.01804	0.02574
	0.484	0.03077	0.02258	0.02872
	0.242	0.03283	0.02545	0.03124
	0.121	0.03410	0.02660	0.03205
	0.061	0.03408	0.02682	0.03216
uoL	0.968/1.936	0.02275	0.01474	0.02275
	0.484/0.968	0.02850	0.02058	0.02850
	0.242/0.484	0.03071	0.02282	0.03071
	0.121/0.242	0.03167	0.02378	0.03167
	0.061/0.121	0.03201	0.02405	0.03201
uoR	1.936/0.968	0.02610	0.01873	0.02610
	0.968/0.484	0.02829	0.02071	0.02829
	0.468/0.242	0.03044	0.02274	0.03044
	0.242/0.121	0.03155	0.02368	0.03155
	0.121/0.061	0.03197	0.02400	0.03197
upL	0.968/1.936	0.02275	0.01474	0.02275
	0.484/0.968	0.02926	0.01992	0.02733
	0.242/0.484	0.03138	0.02340	0.02959
	0.121/0.242	0.03275	0.02572	0.03156
	0.061/0.121	0.03356	0.02664	0.03203
upR	1.936/0.968	0.02610	0.01873	0.02610
	0.968/0.484	0.02888	0.01953	0.02807
	0.468/0.242	0.03086	0.02310	0.03020
	0.242/0.121	0.03229	0.02549	0.03150
	0.121/0.061	0.03304	0.02657	0.03203

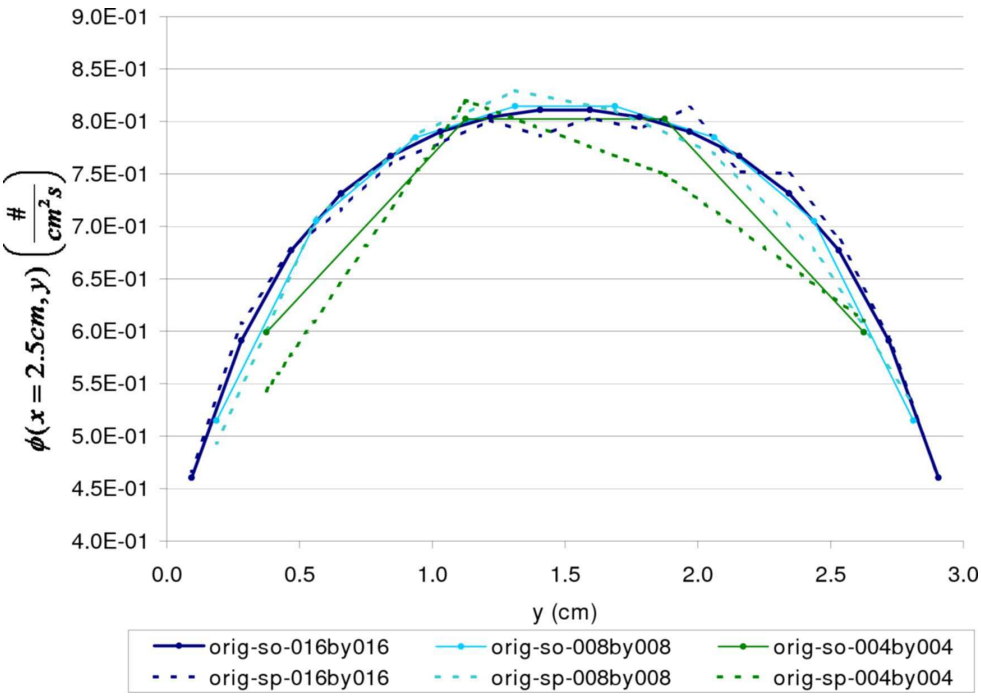


Figure 2.13: Discontinuous Source Test: Face-average scalar fluxes along center interface $x = 2.5$cm for GGK FV discretization, denoted "orig".

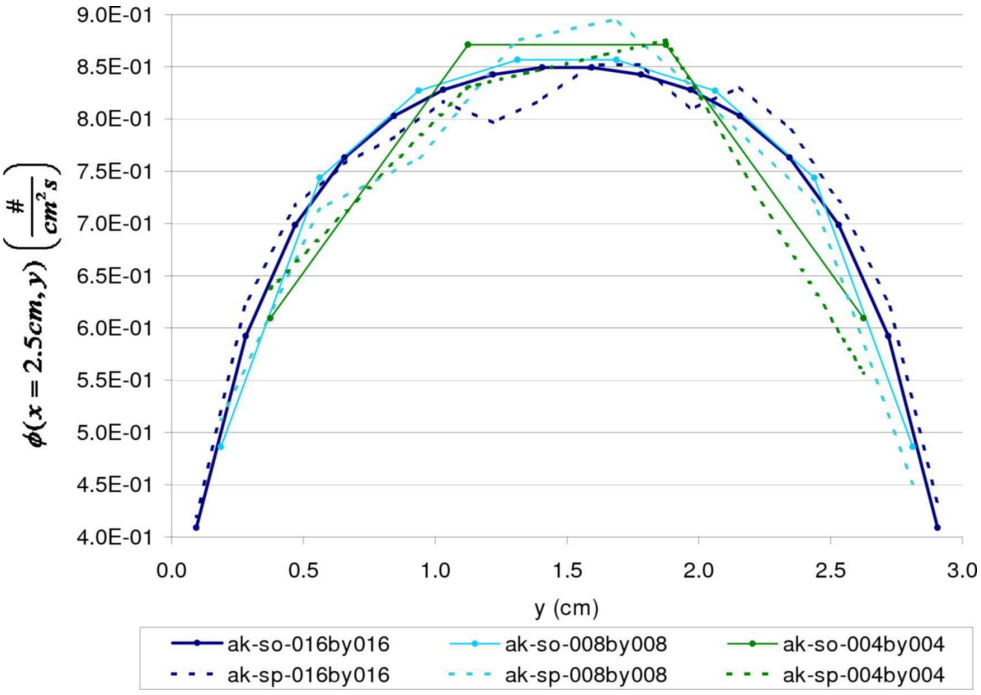


Figure 2.14: Discontinuous Source Test: Face-average scalar fluxes along center interface $x = 2.5$cm for AK FV discretization, denoted "ak".

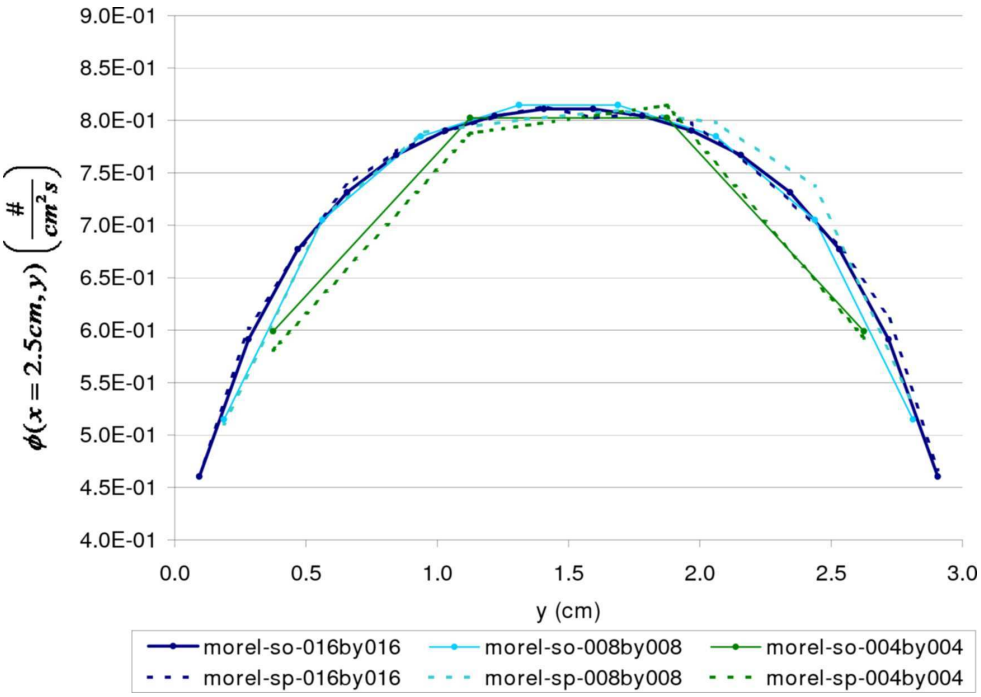


Figure 2.15: Discontinuous Source Test: Face-average scalar fluxes along center interface $x = 2.5$cm for JM FV discretization, denoted "morel".

2.3.2.2 Analysis

All discretizations behave well on the single-level orthogonal (so) meshes, exhibiting second-order convergence, see Table 2.3. The orthogonal hanging-node uoL and uoR meshes also exhibit second-order convergence of the scalar flux, except for ϕ_{right} on the uoR mesh. The reason is most likely related to the fact that information is propagating from left to right, from coarse to fine cells, and this type of redistribution requires a much smaller h to see asymptotic convergence. Numerical convergence of the current J_{ur} is approximately $O(h^{1.5})$, because the interface conditions are not guaranteed to preserve second-order for the current.

On single-level randomized (sp) meshes, the GKK and AK discretizations exhibit reduced numerical convergence orders. GGK exhibits orders of 1.2 and 1.3 for the left-side and right-side scalar fluxes, respectively. AK exhibits 0.7 and 0.9. Our JM discretization exhibits numerical convergence orders of 1.9 and 1.9.

Comparing the discretizations qualitatively by looking at Fig. 2.13, Fig. 2.14, and Fig. 2.15, one sees that our new JM discretization exhibits much less deviation about the orthogonal solution than either the AK or GGK on the coarse meshes shown. Finally, consider the actual accuracy (not order) displayed in the results in Table 2.4, Table 2.5, and Table 2.6. Compare practically any orthogonal scalar flux result on the finest mesh for orthogonal and randomized meshes. With our JM discretization, the relative error is less than 0.1%. With the GGK and AK, scalar fluxes may be different by up to 10%. The AK discretization produces currents which differ from the GGK and JM currents by up to 25%.

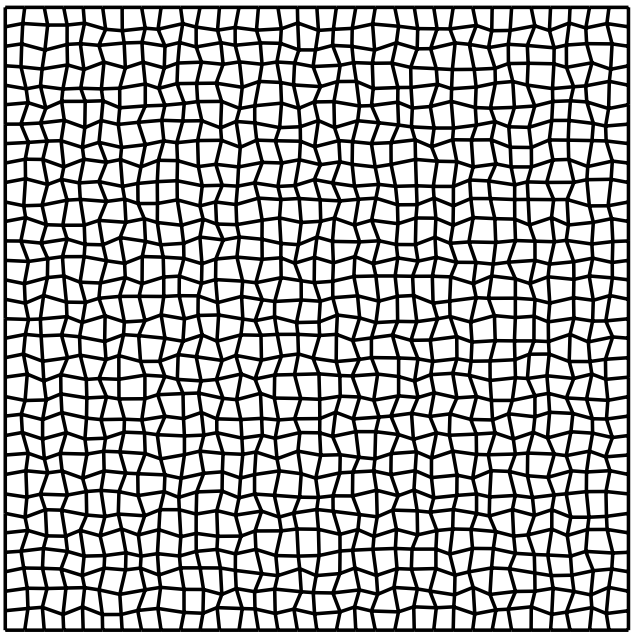


Figure 2.16: Analytic Hump Test: 32×32 logically-rectangular mesh with 20% randomizations of vertices.

2.3.3 Analytic Hump Test

This test uses manufactured solutions on a 1cm $\times$ 1cm domain with $\sigma_t = 1\text{cm}^{-1}$ and scattering $\sigma_s = 0.5\text{cm}^{-1}$. The boundary conditions are vacuum, $\phi_{IN} = J_{IN} = 0$. The exact solution is

$$\phi_{exact}(x,y) = 5 - \tanh\left[5\left(x-\frac{1}{2}\right)^2 + 5\left(y-\frac{1}{2}\right)^2\right], \tag{2.28}$$

with QD factors specified as

$$E_{xx}(x,y) = \frac{1}{3} + \frac{1}{3}\left(x-\frac{1}{2}\right)^2 + \frac{1}{3}\left(y-\frac{1}{2}\right)^2 \tag{2.29a}$$

$$E_{yy}(x,y) = \frac{1}{3} + \frac{1}{3}\left(x-\frac{1}{2}\right)^2 + \frac{1}{3}\left(y-\frac{1}{2}\right)^2 \tag{2.29b}$$

$$E_{xy}(x,y) = -\left(x-\frac{1}{2}\right)\left(y-\frac{1}{2}\right). \tag{2.29c}$$

Using Eq. (2.28) and Eq. (2.29) one can determine the analytic current $\vec{J}_{exact}(x,y)$ by the first moment equations Eq. (1.4b) and the analytic boundary factor C by definition Eq. (1.5). The analytic source $q_{exact}(x,y)$ is then determined by the balance equation Eq. (1.4a). We consider orthogonal and randomized $N \times N$ logically rectangular meshes in this test, where $N = 2, 4, 8, 16, 32, 64$ (see sample mesh in Fig. 2.16.) Randomized meshes have 20% perturbations of vertices. Because this is an analytic test, no high-order transport method is needed.

2.3.3.1 Data

The integral error over the 1cm $\times$ 1cm domain is calculated for the scalar flux and the integral error over the upper right corner for the exiting current,

$$\begin{aligned} \Delta\phi_{dom} &= \int_0^1 \int_0^1 [\phi_{exact}(x,y) - \phi(x,y)]\, dxdy, \\ \Delta J_{ur} &= \int_{0.5}^{1.0} [J_{x,exact}(1,y) - J_x(1,y)]\, dy + \int_{0.5}^{1.0} [J_{y,exact}(x,1) - J_y(x,1)]\, dx \end{aligned}$$

for the GGK, AK, and JM discretizations in Table 2.7 and Table 2.8. High-precision numerical integration is used to calculate the integrals of the exact solutions,

$$\begin{aligned} \int_0^1 \int_0^1 \phi_{exact}(x,y)\, dxdy &= 4.405008591, \\ \int_{0.5}^{1.0} J_{x,exact}(1,y)\, dy + \int_{0.5}^{1.0} J_{y,exact}(x,1)\, dx &= 0.9903481227. \end{aligned}$$

We also compare cell-average scalar flux solutions in the following norms,

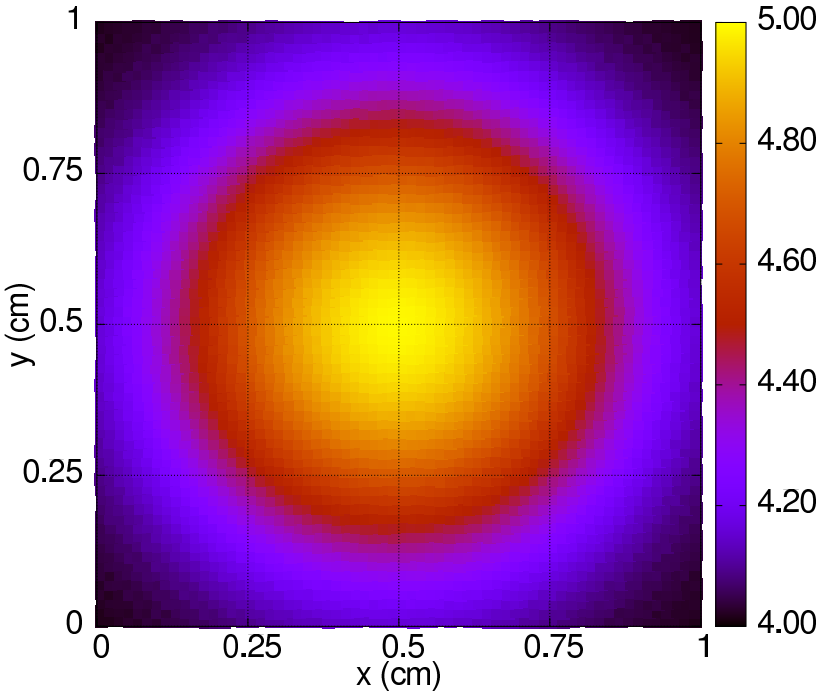
$$
\begin{aligned}
\|\phi^c_{exact} - \phi^c\|_{\infty} &= \max_c \left|\phi^c_{exact} - \phi^c\right|, \\
\|\phi^c_{exact} - \phi^c\|_{\ell_1} &= \sum_c V^c \left|\phi^c_{exact} - \phi^c\right|, \\
\|\phi^c_{exact} - \phi^c\|_{\ell_2} &= \sqrt{\sum_c V^c \left(\phi^c_{exact} - \phi^c\right)^2}, \\
\|\phi^c_{exact} - \phi^c\|_{rel.\ \ell_1} &= \frac{1}{N_C} \sum_c \left|1 - \frac{\phi^c}{\phi^c_{exact}}\right|, \\
\|\phi^c_{exact} - \phi^c\|_{rel.\ \infty} &= \max_c \left|1 - \frac{\phi^c}{\phi^c_{exact}}\right|,
\end{aligned}
$$

for the GGK, AK, and JM discretizations in tables Table 2.9, Table 2.10, and Table 2.11, respectively.

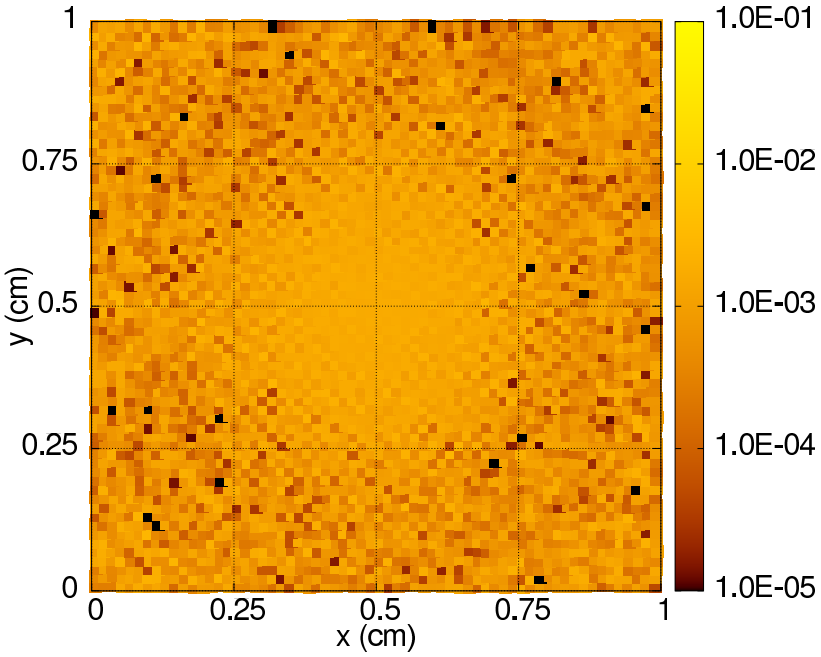
Table 2.7: Analytic Hump Test: Integral error in the scalar flux, $\Delta\phi_{dom}$.

	$\Delta\phi_{dom}$					
	GGK		AK		JM	
$1/h$	Ortho.	Rand. 20%	Ortho.	Rand. 20%	Ortho.	Rand. 20%
2	-1.8E-1	-1.8E-1	-1.5E-1	-1.5E-1	-1.8E-1	-1.8E-1
4	-5.0E-2	-5.0E-2	-2.0E-2	-2.2E-2	-5.0E-2	-5.2E-2
8	-1.3E-2	-1.2E-2	1.5E-2	1.2E-2	-1.3E-2	-1.4E-2
16	-3.3E-3	-1.2E-3	2.4E-2	2.8E-2	-3.3E-3	-3.2E-3
32	-8.3E-4	1.4E-3	2.6E-2	3.0E-2	-8.3E-4	-8.1E-4
64	-2.1E-4	2.1E-3	2.7E-2	3.1E-2	-2.1E-4	-2.0E-4
Order	2.0	*	*	*	2.0	2.0

$*$ Order less than zero

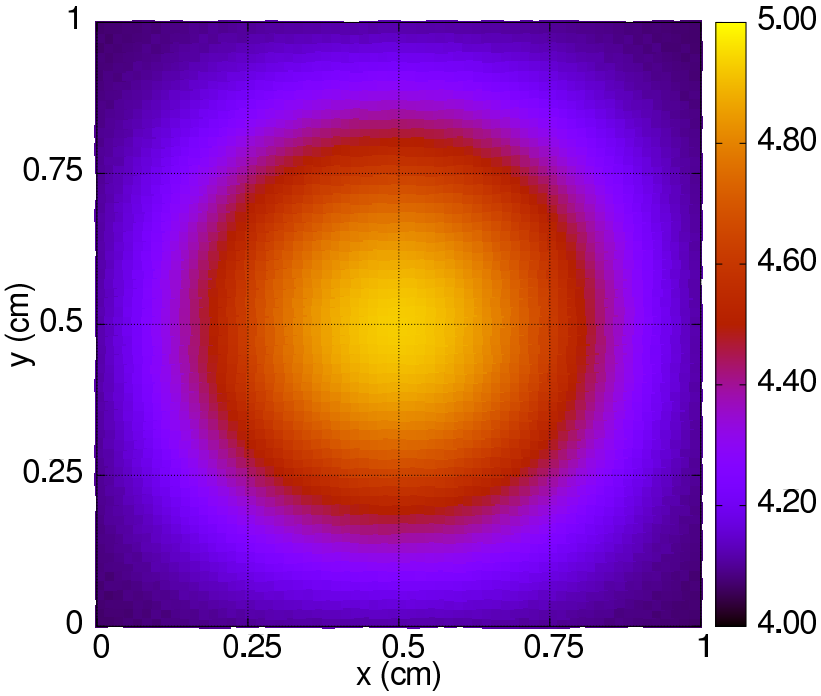


(a) scalar flux

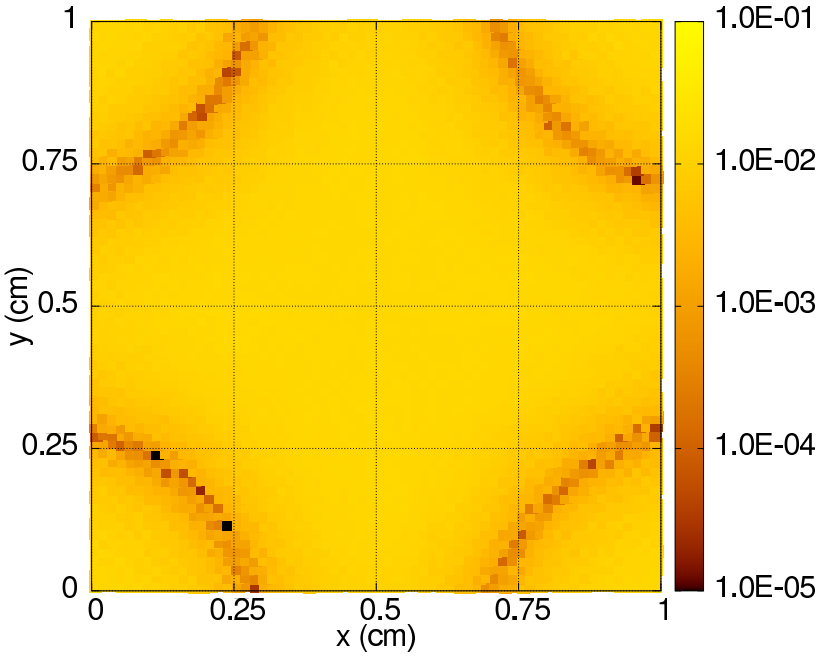


(b) relative error

Figure 2.17: Analytic Hump Test: Scalar flux on the 64×64 randomized mesh (a) and relative errors (b) for the GGK discretization.

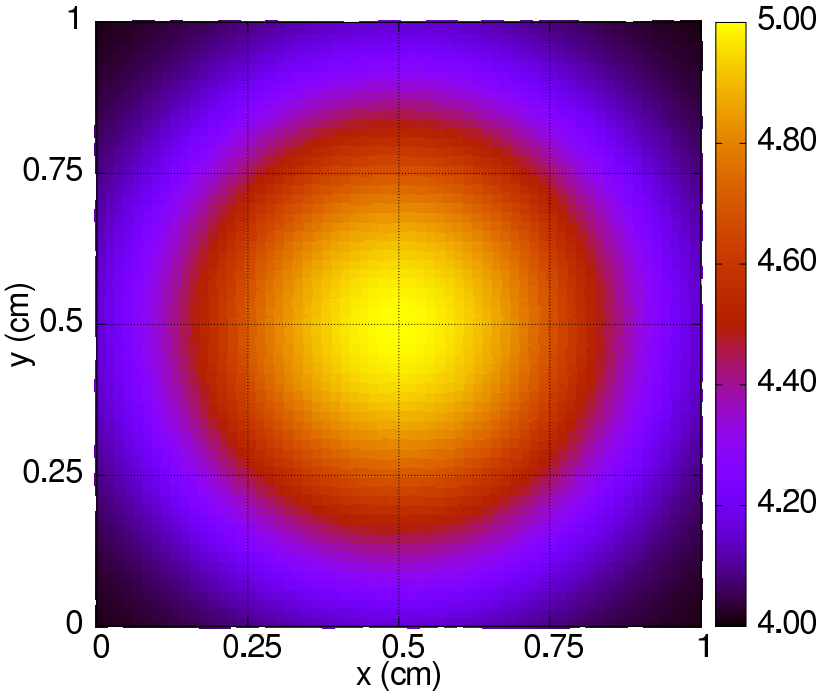


(a) scalar flux

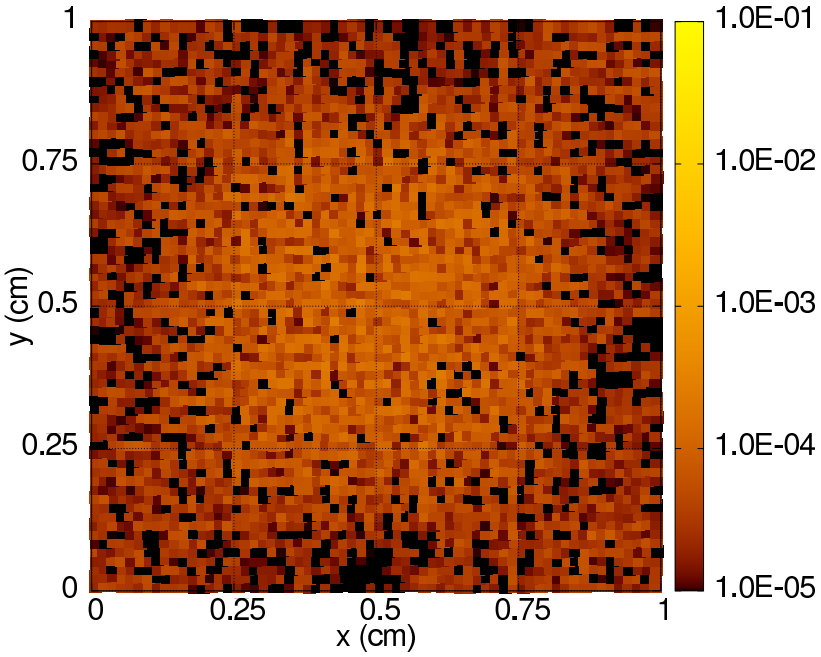


(b) relative error

Figure 2.18: Analytic Hump Test: Scalar flux on the 64×64 randomized mesh (a) and relative errors (b) for the AK discretization.



(a) scalar flux



(b) relative error

Figure 2.19: Analytic Hump Test: Scalar flux on the 64×64 randomized mesh (a) and relative errors (b) for the JM discretization.

Table 2.8: Analytic Hump Test: Integral error in the exiting current, ΔJ_{ur}.

	ΔJ_{ur}					
	GGK		AK		JM	
$1/h$	Ortho.	Rand. 20%	Ortho.	Rand. 20%	Ortho.	Rand. 20%
2	2.3E-2	2.3E-2	1.9E-2	1.9E-2	2.3E-2	2.3E-2
4	6.3E-3	1.1E-2	2.5E-3	1.4E-2	6.3E-3	1.1E-2
8	1.6E-3	1.0E-3	-1.9E-3	-5.6E-4	1.6E-3	1.9E-3
16	4.1E-4	-3.6E-4	-3.0E-3	-4.4E-3	4.1E-4	4.6E-4
32	1.0E-4	-1.3E-4	-3.3E-3	-3.5E-3	1.0E-4	8.9E-5
64	2.6E-5	-2.0E-4	-3.4E-3	-3.5E-3	2.6E-5	1.8E-5
Order	2.0	*	*	*	2.0	2.3

$*$ Order less than zero

Table 2.9: Analytic Hump Test: Error norms of the GGK cell-average scalar flux, ϕ^c.

Mesh	$1/h$	∞	ℓ_1	ℓ_2	rel. 1	rel. ∞
Ortho.	2	1.84E-1	1.84E-1	1.84E-1	4.167%	4.167%
	4	1.22E-1	5.00E-2	6.54E-2	1.090%	2.552%
	8	3.45E-2	1.32E-2	1.69E-2	0.286%	0.698%
	16	8.86E-3	3.32E-3	4.25E-3	0.072%	0.178%
	32	2.23E-3	8.32E-4	1.06E-3	0.018%	0.045%
	64	5.58E-4	2.08E-4	2.66E-4	0.005%	0.011%
Rand. 20%	2	1.84E-1	1.84E-1	1.84E-1	4.167%	4.167%
	4	1.75E-1	5.15E-2	7.48E-2	1.075%	3.661%
	8	5.45E-2	1.92E-2	2.37E-2	0.426%	1.182%
	16	3.29E-2	9.27E-3	1.13E-2	0.214%	0.789%
	32	2.04E-2	5.39E-3	6.58E-3	0.125%	0.502%
	64	1.28E-2	3.92E-3	4.75E-3	0.088%	0.293%

Table 2.10: Analytic Hump Test: Error norms of the AK cell-average scalar flux, ϕ^c.

Mesh	$1/h$	∞	ℓ_1	ℓ_2	rel. 1	rel. ∞
Ortho.	2	1.53E-1	1.53E-1	1.53E-1	3.464%	3.464%
	4	5.26E-2	2.21E-2	3.09E-2	0.489%	1.096%
	8	4.48E-2	2.68E-2	2.91E-2	0.608%	1.107%
	16	5.31E-2	3.50E-2	3.86E-2	0.783%	1.198%
	32	5.84E-2	3.72E-2	4.11E-2	0.830%	1.233%
	64	5.97E-2	3.77E-2	4.18E-2	0.841%	1.257%
Rand. 20%	2	1.53E-1	1.53E-1	1.53E-1	3.464%	3.464%
	4	1.18E-1	3.61E-2	4.95E-2	0.783%	2.467%
	8	7.86E-2	2.60E-2	3.08E-2	0.620%	1.946%
	16	8.38E-2	3.98E-2	4.51E-2	0.899%	1.760%
	32	7.45E-2	4.14E-2	4.62E-2	0.925%	1.590%
	64	7.27E-2	4.18E-2	4.67E-2	0.930%	1.524%

Table 2.11: Analytic Hump Test: Error norms of the JM cell-average scalar flux, ϕ^c.

Mesh	$1/h$	∞	ℓ_1	ℓ_2	rel. 1	rel. ∞
Ortho.	2	1.84E-1	1.84E-1	1.84E-1	4.167%	4.167%
	4	1.22E-1	5.00E-2	6.54E-2	1.090%	2.552%
	8	3.45E-2	1.32E-2	1.69E-2	0.286%	0.698%
	16	8.86E-3	3.32E-3	4.25E-3	0.072%	0.178%
	32	2.23E-3	8.32E-4	1.06E-3	0.018%	0.045%
	64	5.58E-4	2.08E-4	2.66E-4	0.005%	0.011%
Rand. 20%	2	1.84E-1	1.84E-1	1.84E-1	4.167%	4.167%
	4	1.63E-1	5.24E-2	7.34E-2	1.064%	3.397%
	8	6.88E-2	1.47E-2	2.12E-2	0.293%	1.436%
	16	1.76E-2	3.40E-3	4.63E-3	0.069%	0.359%
	32	4.23E-3	8.79E-4	1.22E-3	0.017%	0.088%
	64	1.29E-3	2.20E-4	3.02E-4	0.004%	0.026%

2.3.3.2 Analysis

This test is different from previous tests because the analytic solution is known and we can directly calculate the numerical convergence order via Eq. (1.21) instead of using the Aitken process. This test shows that the JM discretization is far superior to the GGK and AK discretizations on randomized meshes, having errors an order of magnitude lower than the GGK discretization and two orders of magnitude lower than the AK discretization, by both the integral errors in Table 2.7 and Table 2.8 and the variety of norms presented for the JM discretization in Table 2.11 compared to GGK in Table 2.9 and AK in Table 2.10.

2.3.4 Analytic Peak Test

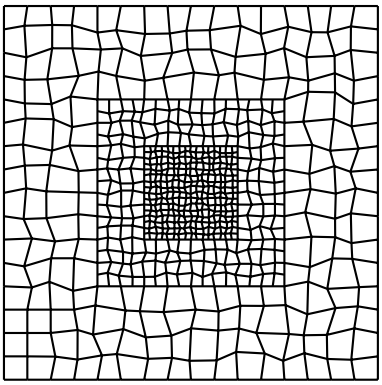


Figure 2.20: Analytic Peak Test: Randomized hanging-node mesh with $h = \frac{1}{64}$cm.

This test is similar to the last "Analytic Hump Test", the domain is 1cm $\times$ 1cm with $\sigma_t = 1\text{cm}^{-1}$, scattering $\sigma_s = 0.5\text{cm}^{-1}$, and vacuum BC, $\phi_{IN} = J_{IN} = 0$. However, the analytic solution has a much sharper peak (see Fig. 2.21),

$$\phi_{exact}(x, y) = 5 - \tanh\left[100\left(x - \frac{1}{2}\right)^2 + 100\left(y - \frac{1}{2}\right)^2\right], \tag{2.30}$$

due to the "100" factor inside tanh. We consider logically rectangular meshes as in the previous test as well as hanging-node meshes which have 3 levels of refinement about the central region as in Fig. 2.20. We refine the square region 0.25cm $\leq x, y \leq$ 0.75cm, and then refine again a smaller square region 0.375cm $\leq x, y \leq$ 0.625cm. Starting with an $N \times N$ mesh, this type of refinement leads to a 5.5 times increase in number of cells. Because this is an analytic test, no high-order transport method is needed.

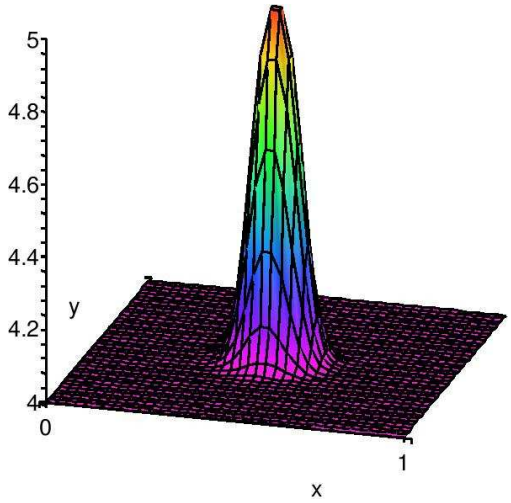


Figure 2.21: Analytic Peak Test: scalar flux solution $\phi_{exact}(x, y)$.

2.3.4.1 Data

Table 2.12, Table 2.13, and Table 2.14, show ℓ_2 error norms of the scalar flux on logically rectangular meshes for the GGK, AK, and JM LOQD discretizations, respectively. The ℓ_2 error norm of the scalar flux on hanging-node meshes in Table 2.15, Table 2.16, and Table 2.17, for the GGK, AK, and JM LOQD discretizations, respectively. The characteristic cell-width h is reported for that of the smallest cells in the mesh, i.e. in the center region for hanging-node meshes. The

scalar flux and relative error on 64×64 randomized meshes are shown in Fig. 2.22, Fig. 2.23, and Fig. 2.24, for the GGK, AK, and JM discretizations, respectively.

Table 2.12: Analytic Peak Test: GGK discretization ℓ_2 error norm on logically rectangular meshes.

		Ortho.		Rand. 20%	
N_C	$1/h$	ℓ_2 err.	*Order*	ℓ_2 err.	*Order*
4	2	2.53E-1	-	2.53E-1	-
16	4	6.18E-2	2.0	6.82E-2	1.9
64	8	2.24E-2	1.5	3.71E-2	0.9
256	16	1.36E-2	0.7	1.80E-2	1.0
1024	32	3.12E-3	2.1	6.92E-3	1.4
4096	64	7.80E-4	2.0	4.30E-3	0.7

Table 2.13: Analytic Peak Test: AK discretization ℓ_2 error norm on logically rectangular meshes.

		Ortho.		Rand. 20%	
N_C	$1/h$	ℓ_2 err.	*Order*	ℓ_2 err.	*Order*
4	2	2.34E-1	-	2.34E-1	-
16	4	5.90E-2	2.0	6.49E-2	1.9
64	8	2.26E-2	1.5	3.42E-2	0.9
256	16	1.30E-2	0.7	1.94E-2	0.8
1024	32	2.68E-3	2.1	7.34E-3	1.4
4096	64	7.29E-4	2.0	4.88E-3	0.6

Table 2.14: Analytic Peak Test: JM discretization ℓ_2 error norm on logically rectangular meshes.

		Ortho.		Rand. 20%	
N_C	$1/h$	ℓ_2 err.	*Order*	ℓ_2 err.	*Order*
4	2	2.53E-1	-	2.53E-1	-
16	4	6.18E-2	2.3	6.60E-2	1.9
64	8	2.24E-2	2.0	3.13E-2	1.1
256	16	1.36E-2	1.6	1.55E-2	1.0
1024	32	3.12E-3	2.0	2.86E-3	2.4
4096	64	7.80E-4	2.0	8.35E-4	1.8

Table 2.15: Analytic Peak Test: GGK discretization ℓ_2 error norm on hanging-node meshes.

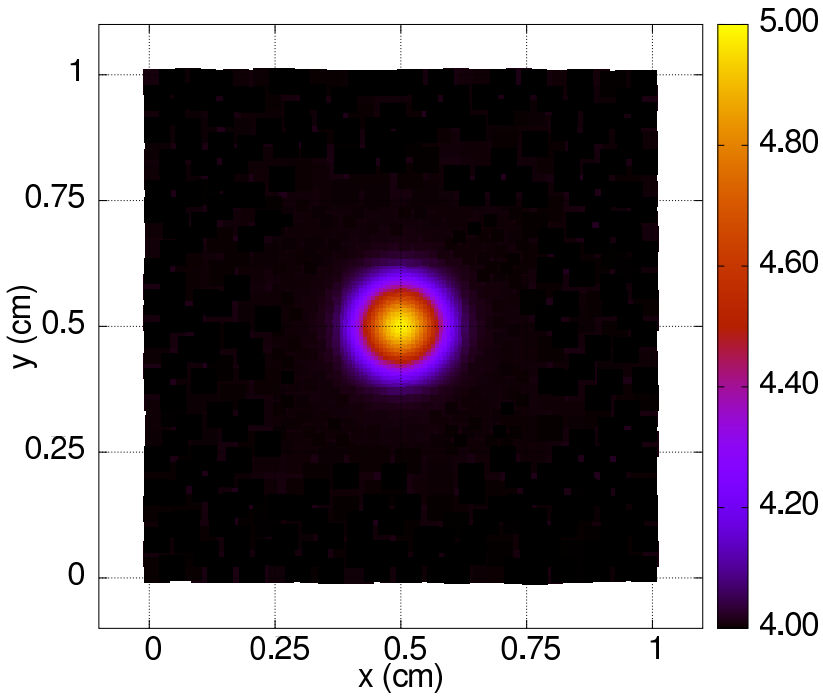
		Ortho.		Rand. 20%	
N_C	$1/h$	ℓ_2 err.	ratio	ℓ_2 err.	ratio
88	16	6.62E-2	-	6.52E-2	-
352	32	1.69E-2	2.0	2.29E-2	1.5
1408	64	4.28E-3	2.0	1.26E-2	0.9
5632	128	1.07E-3	2.0	6.58E-3	0.9

Table 2.16: Analytic Peak Test: AK discretization ℓ_2 error norm on hanging-node meshes.

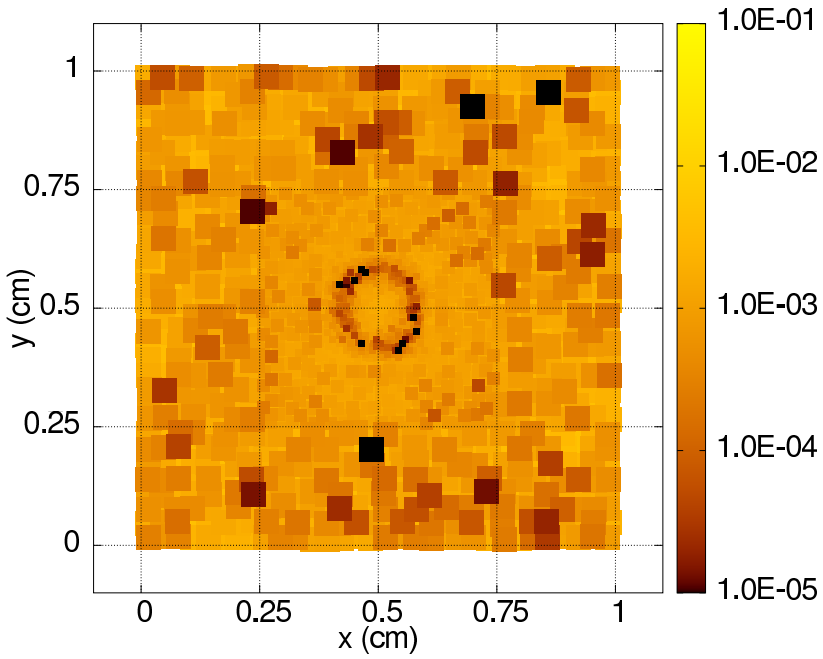
		Ortho.		Rand. 20%	
N_C	$1/h$	ℓ_2 err.	ratio	ℓ_2 err.	ratio
88	16	6.21E-2	-	6.21E-2	-
352	32	1.56E-2	2.0	2.12E-2	1.6
1408	64	3.83E-3	2.0	1.08E-2	1.0
5632	128	1.07E-3	1.8	5.92E-3	0.9

Table 2.17: Analytic Peak Test: JM discretization ℓ_2 error norm on hanging-node meshes.

		Ortho.		Rand. 20%	
N_C	$1/h$	ℓ_2 err.	ratio	ℓ_2 err.	ratio
88	16	6.62E-2	-	6.50E-2	-
352	32	1.69E-2	2.0	1.82E-2	1.9
1408	64	4.28E-3	2.0	4.73E-3	2.0
5632	128	1.07E-3	2.0	1.22E-3	2.0

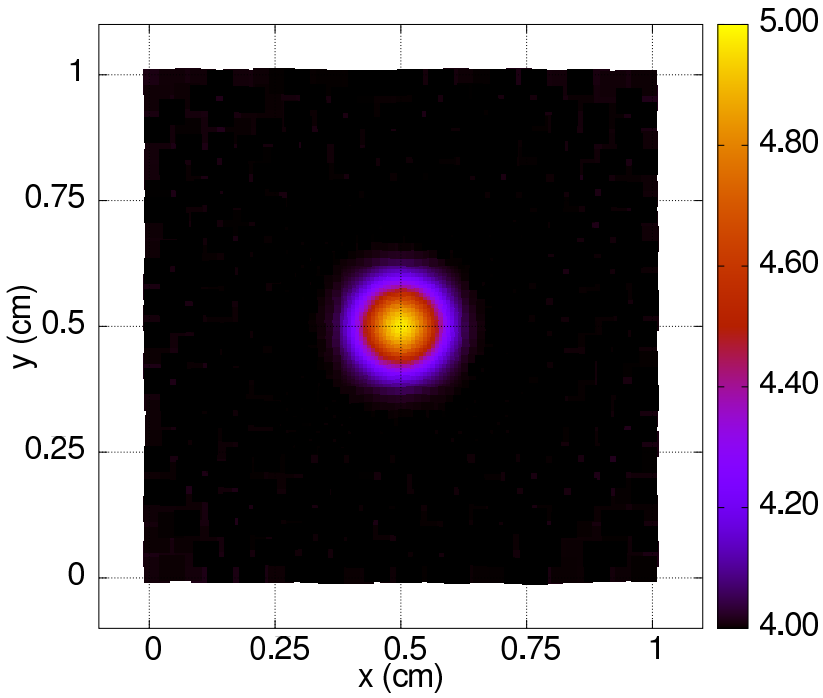


(a) scalar flux

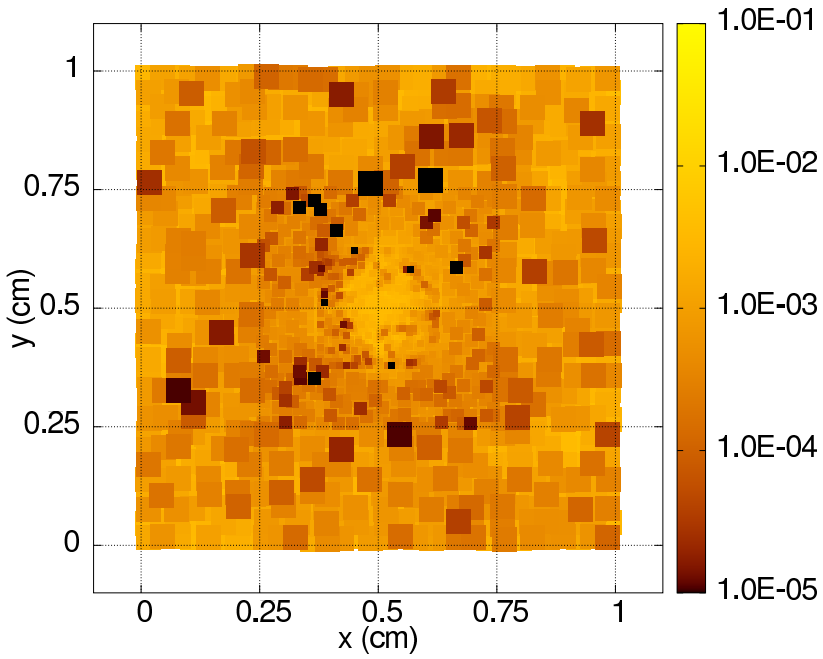


(b) relative error

Figure 2.22: Analytic Peak Test: Scalar flux on the 64×64 randomized mesh (a) and relative errors (b) for the GGK discretization.

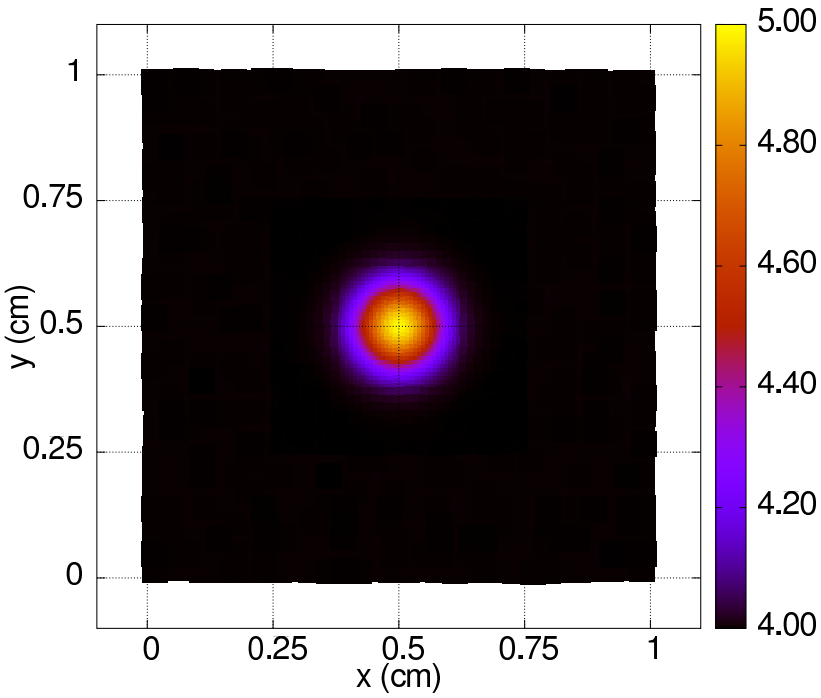


(a) scalar flux

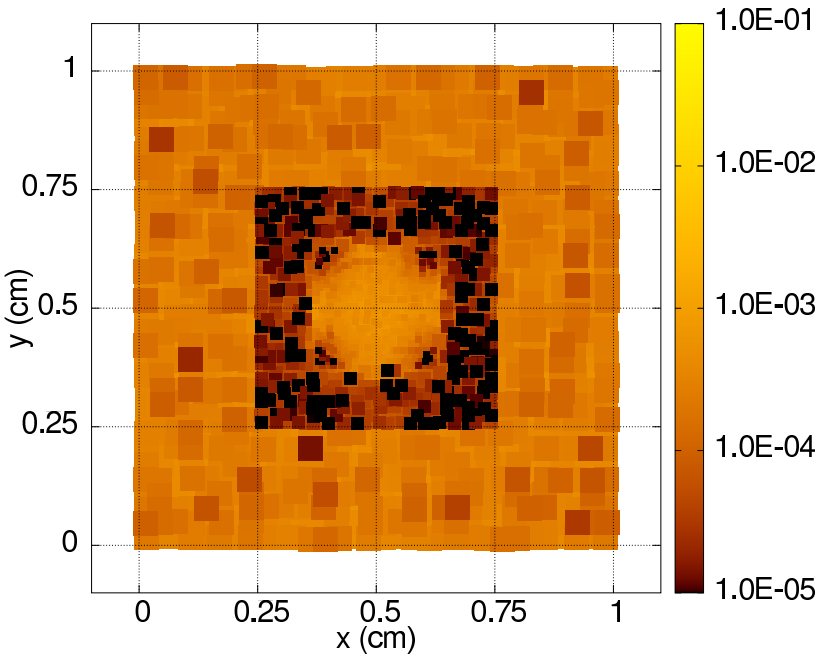


(b) relative error

Figure 2.23: Analytic Peak Test: Scalar flux on the 64×64 randomized mesh (a) and relative errors (b) for the AK discretization.



(a) scalar flux



(b) relative error

Figure 2.24: Analytic Peak Test: Scalar flux on the 64×64 randomized mesh (a) and relative errors (b) for the JM discretization.

2.3.4.2 Analysis

Our new JM discretization performs excellently on this analytic test, exhibiting second-order convergence in the ℓ_2 error norm on logically-rectangular and hanging-node meshes with random perturbations of vertices. On orthogonal meshes, the error in the ℓ_2 norm is similar for all discretizations tested, see Table 2.12-Table 2.17. On orthogonal meshes, all discretizations (GGK, AK, and JM) also show second order. However, on randomized meshes, the JM discretization shows second-order while AK and GGK show first order or less. Comparing error on the finest meshes, the JM discretization has a factor of 5 less error than GGK or AK. The relative errors in Fig. 2.24(b) show that our hanging-node meshes lead to much less error in the middle region surrounding the peak than either the inside or outside regions. Compared to GGK and AK solutions in Fig. 2.22(b) and Fig. 2.23(b), our JM discretization is clearly superior.

Although we do not adaptively refine meshes, it is easy to guess that adaption around the peak will lead to better approximation of the solution features, however there is no hanging-node mesh result which is clearly more efficient (less error per cell) than a logically-rectangular mesh. That is the coefficient associated with the $O(h^2)$ convergence is clearly smaller for the uniformly refined logically-rectangular meshes, at least in this global error quantity.

2.4 Summary

We have derived a new discretization of the LOQD equations in Eq. (2.20) which uses a corresponding reciprocal basis as in [89] to represent the tensor divergence in our general FV framework. We have also determined interface conditions for hanging-node interfaces, Eq. (2.8a), and for standard interfaces Eq. (2.7) that allow one to solve the LOQD equations on a large variety

of quadrilateral meshes, without introducing extra unknowns into the system (e.g. polygons) due to special interfaces. We conclude with the following statements.

1. On orthogonal meshes, the geometric factors reduce our new LOQD discretization to the GGK FV scheme [24, 54].

2. On randomized meshes in 2D Cartesian geometry, the new LOQD discretization has demonstrated superior accuracy compared to other tested FV discretizations [24, 54].

3. Hanging-node interface conditions may be imposed to solve problems on hanging-node meshes [105].

4. The discretization has a straightforward extension to polygons and polyhedra.

5. In the case of diffusion where the QD tensor is diagonal and equal to 1/3, our discretization reduces to Jim Morel's diffusion discretization [89].

Chapter 3

QD FACTOR CALCULATION, SCATTERING SOURCE REPRESENTATION, AND OTHER TRANSPORT-RELATED CONSIDERATIONS

In this chapter we discuss the manner in which to calculate the angular flux, ψ, that is compatible with our new LOQD FV discretization. Although the transport equation discretizations are not new, various modifications were necessary, and in an effort to be complete, we discuss them here. Completely new material in this section is 1) how to calculate the QD factors, $\mathrm{E}[\psi]$ and $C[\psi]$

from vertex-based data on unstructured meshes and 2) a linear representation of the scattering source in a cell.

3.1 A Short Characteristics Transport Discretization with Vertex Unknowns

On orthogonal meshes, a second-order (in space) method may be constructed by parabolic interpolation within the cell [24], which was used successfully with QD. We seek an appropriate extension of the orthogonal mesh variant to unstructured meshes of quadrilaterals in order to calculate face-average and cell-average QD factors, $E_{\alpha\beta}[\psi]$ and $C[\psi]$, for use with our FV LOQD discretization.

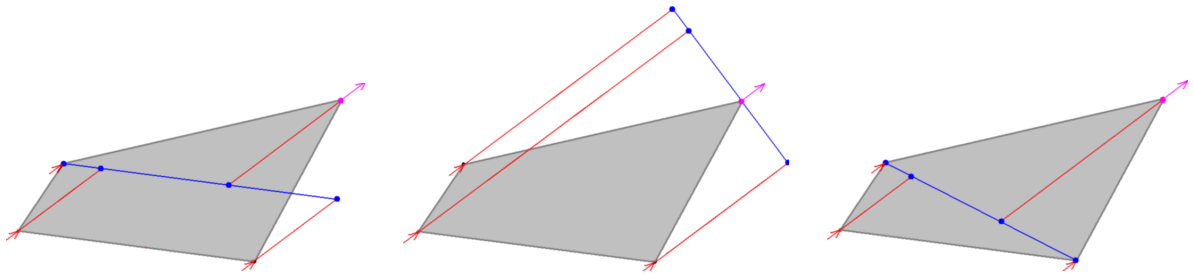
Short characteristics uses the form of the transport equation in Eq. (3.1), shown in 2D below for convenience,

$$\psi(s) = \psi_0 e^{-\sigma_t \frac{s}{\sin\theta}} + \frac{1}{\sin\theta}\int_0^s q(s')e^{-\sigma_t \frac{s-s'}{\sin\theta}}ds', \tag{3.1}$$

where $q(s)$ is the total source term which includes scattering and external sources. The characteristics are along lines given by $x = \Omega_x s/\sin\theta + x_0$ and $y = \Omega_y s/\sin\theta + y_0$ such that s now represents the projection of the distance along the characteristic to the xy-plane,

Parabolic interpolation requires three values on the same line, called the *interpolation line*, in order to determine the exiting angular flux, see Algorithm 3. We considered three options for parabolic interpolation within an arbitrary quadrilateral cell (see Fig. 3.1) before deciding on the diagonal line interpolation.

- Face-Parallel: parallel to the face intersected when travelling in $-\vec{\Omega}$ direction Fig. 3.1(a), (a



(a) Face-Parallel (b) Perpendicular (c) Diagonal

Figure 3.1: Parabolic Interpolation: interpolation lines (blue), with characteristics used (red), outgoing angular flux (magenta arrow), and incoming angular fluxes (red arrows).

natural extension of a short characteristic scheme with linear interpolation [49],

- Perpendicular: perpendicular to the characteristic ray, Fig. 3.1(b), and
- Diagonal: oriented on the diagonal of a quadrilateral cell, Fig. 3.1(c).

3.1.1 Diagonal Interpolation Line

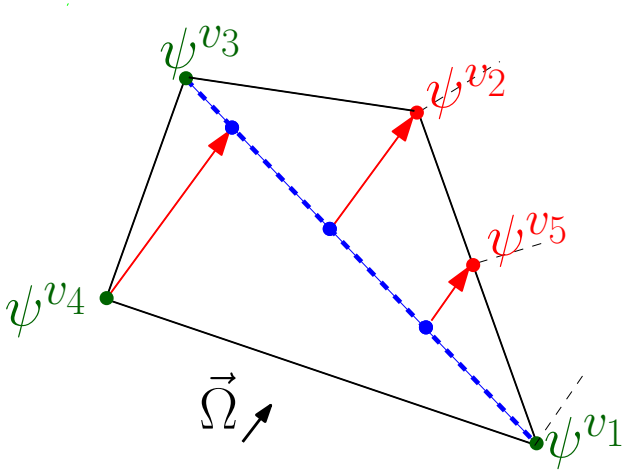


Figure 3.2: Short-characteristics high-order transport discretization on vertices with diagonal interpolation (SCV). For direction $\vec{\Omega}$, solve outgoing ψ (red) from 3 incoming ψ (green), using cell diagonal (blue).

We use the diagonal interpolation line because it results in all characteristics being com-

pletely contained within the cell. Obviously, if the cell cross sections, scattering source, and/or external source are different form its neighbors, additional approximation is introduced by the face-parallel and perpendicular lines that is not introduced with the diagonal, which on coarse meshes could lead to considerable approximation error.[1] From a computational point of view, the diagonal interpolation line requires only 2 evaluations of Eq. (3.1) while others require 2 or 3 evaluations. Evaluations of Eq. (3.1) should be minimized because they include fairly costly exponential evaluations. The computational cell is shown in Fig. 3.2.

3.1.2 Monotonization

As discussed in the Introduction, the angular fluxes used to calculate QD factors must be monotonic so that derivative terms which involve factors are bounded. Our interpolation is via a parabola, $Z^c(\ell) = a\ell^2 + b\ell + c$, calculated from three point values, which may have an extremum. To remedy, we use a min/max monotonization, where first we calculate the minimum and maximum values allowed based on the values used to construct the interpolant,

$$Y^c_{min} = \min(\psi_*^{v_{IN1}}, \psi_*^{v_{IN2}}, \psi_*^{v_{IN3}}), \tag{3.2}$$

$$Y^c_{max} = \max(\psi_*^{v_{IN1}}, \psi_*^{v_{IN2}}, \psi_*^{v_{IN3}}), \tag{3.3}$$

where the three $\psi_*^{v_{IN}}$ values are point angular flux values on the interpolation line, determined from angular fluxes at "incoming" vertices, $\psi^{v_{IN}}$, and Eq. (3.1). The monotonic, parabolic interpolant

[1] One could use the neighbors' properties for parts of characteristics that lie outside the cell (see Perpendicular in Fig. 3.1), but these are extra intersection calculations and as we have mentioned before, we desire a cell-local transport scheme, requiring only properties of the current cell in the transport sweep to compute the outgoing angular fluxes from the incoming.

Algorithm 3: Our Vertex-based Algorithm.

for *each direction* $\vec{\Omega}_m$ **do**

  for *each cell* c **do**

    for *each incoming vertex* v_{IN} **do**

      From the incoming vertex v_{IN}, determine the distance $s_*^{v_{IN}}$ along the characteristic to the interpolation line at position $\ell_*^{v_{IN}}$.

      Determine the interpolation line angular flux, $\psi_*^{v_{IN}} = \psi(s_*^{v_{IN}})$, via Eq. (3.1) with $\psi_0 = \psi^{v_{IN}}$.

    end

    Use $(\ell_*^{v_{IN}}, \psi_*^{v_{IN}})$ values along the interpolation line to determine a monotonic interpolating function $Y^c(\ell)$, where ℓ are local coordinates on the line:

    if *number of incoming vertices* > 2 **then**

      Construct parabolic interpolant from 3 $\psi_*^{v_{IN}}$ values.

    else

      Construct linear interpolant from 2 $\psi_*^{v_{IN}}$ values.

    end

    for *each outgoing vertex* v_{OUT} **do**

      Determine the interpolation line point $\ell_{**}^{v_{OUT}}$ and the distance $s_{**}^{v_{OUT}}$ from $\ell_{**}^{v_{OUT}}$ to vertex v_{OUT} on the characteristic.

      Determine the outgoing angular flux $\psi^{v_{OUT}} = \psi(s_{**}^{v_{OUT}})$ via Eq. (3.1) with $\psi_0 = Y^c(\ell_{**}^{v_{OUT}})$.

    end

  end

end

is evaluated as

$$Y^c(\ell) = \begin{cases} Y^c_{min} & \text{if } Z^c(\ell) < Y^c_{min}, \\ Y^c_{max} & \text{if } Z^c(\ell) > Y^c_{max}, \\ Z^c(\ell) & \text{otherwise.} \end{cases} \tag{3.4}$$

Note that the monotonization procedure causes the transport method to become nonlinear.

3.1.3 Determining QD Factors from Vertex-based Characteristics

In order to use vertex-based characteristics transport methods with our FV LOQD discretization, one must determine face-average and cell-average QD Factors, $E_{\alpha\beta}[\psi]$ and $C[\psi]$, from angular fluxes at vertices $\{\psi^v | v = 1, ..., N_v\}$. In order to do this, first we calculate the QD factors at vertices,

$$E^v_{\alpha\beta} = \frac{\sum_m w_m \Omega_{\alpha,m} \Omega_{\beta,m} \psi^v_m}{\sum_m w_m \psi^v_m}, \tag{3.5}$$

$$C^{v_b} = \frac{\sum_{m \in OUT} w_m (\vec{n}^{v_b} \cdot \vec{\Omega}_m) \psi^{v_b}_m}{\sum_{m \in OUT} w_m \psi^{v_b}_m}. \tag{3.6}$$

where ψ^v_m is the angular flux at vertex v in direction $\vec{\Omega}_m$ and v_b denotes a boundary vertex. The summation over $m \in OUT$ is the quadrature equivalent of integration over outgoing directions defined by $\vec{n}^{v_b} \cdot \vec{\Omega}_m > 0$, where $\vec{n}^{v_b}$ is the outward normal at a boundary vertex. The outward normal is not well-defined for a corner vertex on the boundary, but our FV LOQD discretization needs only face-average boundary C^{f_b} factors where the outward normal is defined by the face, $\vec{n}^{f_b}$, not the vertex. As a result we define two values for each C^{v_b} on the corner boundary vertices, associated with the two (possibly) different outward normals for faces.

3.1.3.1 Face-average QD Factor Calculation from Vertex QD Factors

In order to determine the face-average factors, we employ a simple trapezoid rule on vertex values. On simple two-vertex faces, $v_1, v_2 \in f$, this is simply

$$E^f_{\alpha\beta} = \frac{1}{2}(E^{v_1}_{\alpha\beta} + E^{v_2}_{\alpha\beta}). \tag{3.7}$$

On faces with more than two vertices, like on hanging-node meshes, this is simply

$$E^f_{\alpha\beta} = \frac{1}{A^f}\sum_{v\in f} A^v E^v_{\alpha\beta}, \tag{3.8}$$

$$C^f = \frac{1}{A^f}\sum_{v\in f} A^v C^v, \tag{3.9}$$

where the vertex weighting A^v is determined geometrically as per the trapezoid rule. For the boundary vertices, $v_{b1}, b_{b2} \in f_b$, the C^{f_b} boundary factor is calculated as

$$C^{f_b} = \frac{1}{2}(C^{v_{b1}} + C^{v_{b2}}), \tag{3.10}$$

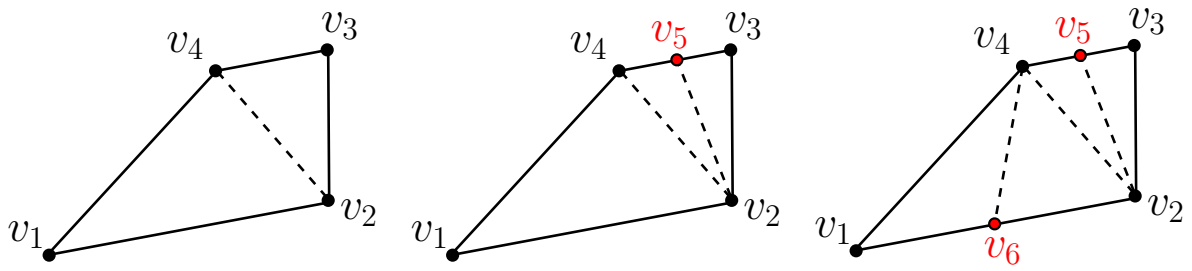
where at corner boundary vertices, the appropriate C^{v_b} associated with normal $\vec{n}^{f_b}$ is used. All face-average factor calculation strategies above are valid for both orthogonal and arbitrary meshes.

3.1.3.2 Cell-average QD Factor Calculation from Vertex QD Factors

In order to determine the cell-average factors, we must employ a different strategy on both orthogonal and arbitrary meshes. On rectangular meshes, this simple approximation is appropriate,

$$E^c_{\alpha\beta} = \frac{1}{4}(E^{BR}_{\alpha\beta} + E^{TR}_{\alpha\beta} + E^{TL}_{\alpha\beta} + E^{BL}_{\alpha\beta}), \tag{3.11}$$

where BR, TR, TL, and BL denote the bottom right, top right, top left, and bottom left vertices of the rectangle, respectively. On skewed quadrilateral cells and cells with hanging-nodes, we perform



(a) no hanging nodes (b) 1 hanging node (c) 1 hanging node

Figure 3.3: Delaunay triangulations of hanging node cells.

a Delaunay triangulation [67] of the vertices, construct a linear function on each triangle, and integrate the resulting linear functions over the cell,

$$E^c_{\alpha\beta} = \frac{1}{V^c} \sum_{t=1}^{N_{tri}} \sum_{n=1}^{3} V^{t,n} E^{v_{t,n}}_{\alpha\beta}, \tag{3.12}$$

where N_{tri} is the number of triangles (for a cell with N vertices there are $N-2$ triangles) and $v_{t,n}$ is the vertex index of the n-th node of the t-th triangle. The vertex weighting $V^{t,n}$ is dependent on the triangulation. Sample triangulations are shown in Fig. 3.3. If desired, the vertex weights may be precalculated and stored for each cell. For orthogonal cells, Eq. (3.12) reduces to the equal-weighted case of Eq. (3.11).

3.1.4 Difficulties with Vertex-based Characteristics

The short characteristics method described by Algorithm 3 that uses a diagonal interpolation line and min/max monotonization is referred to as SCV. On orthogonal meshes, SCV has $O(h^2)$ truncation error because in each cell the parabolic interpolant has $O(h^3)$ truncation error (see appendix), but characteristics travel through $O(1/h)$ cells in the worst case, so globally the truncation error is $O(h^2)$.

Preliminary results confirmed $O(h^2)$ for the SCV scheme on arbitrary meshes as well. However, for some problems, we did not observe consistent convergence of $O(h^2)$ which led us to the fact that cells exist in the mesh which, for a particular direction have only one incoming face and three outgoing faces. This means one must resort to a linear interpolant for these cell/directions. On the maximally randomized meshes of 30%, approximately 18% of all interpolations are linear. Thus the method is not $O(h^2)$, although it appears so in many tests.

A possible fix is then to add vertices to the centers of faces such that each face has 3 vertex angular flux values associated with it, and thus a parabolic interpolator may always be constructed even if there is only one incoming face.

However, with the same number of degrees of freedom, we can use a *conservative* short characteristics method with subcell balances (SCSB) which has both vertex and face-average angular flux unknowns. Because the SCSB scheme has face-average angular flux primary unknowns,[2] SCSB will generate better face-average and cell-average QD factors for the LOQD equations.

One additonal problem with any short characteristics method that has vertex-based unknowns only at corners of cells, is that acute-angled cells may be excluded from the transport sweep. For example, consider a high aspect-ratio triangle, where the small face is upwind and the smallest acute angle, θ_{min}, is downwind. If this wedge-shaped triangle has θ_{min} smaller than the angle between directions in the quadrature set, this cell will be completely excluded from the transport sweep [51]. With a face-based short characteristic method (or a vertex-based method that has vertices on faces), it is impossible to exclude cells.

[2]Cell-average angular fluxes are also defined from the characteristic equations, not from the balance equation, but are not primary unknowns.

3.2 A Short Characteristics Method with Subcell Balances

In an effort to ensure parabolic interpolation in all cells, we have developed a short characteristics method which utilizes subcell balances (SCSB). We do not present a full description of our subcell balance method here. For an overview, see Alg. 4 and Fig. 3.4. For the full derivation see Appendix A. It is similar in nature to other conservative (multiple-balance) short characteristic S_N transport methods: the Bakirova-Karpov-Mukhina (BKM) method [34,104], split-cell [61], slice balance [100].

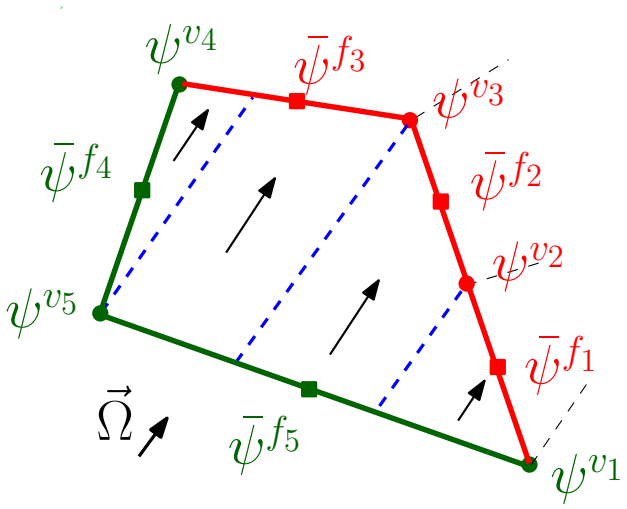


Figure 3.4: Conservative short-characteristics high-order transport discretization with subcell balances (SCSB). For direction $\vec{\Omega}$, solve outgoing ψ (red) from 3 incoming ψ (green).

3.2.1 Monotonization

As discussed in the Introduction, the angular fluxes used to calculate QD factors must be monotonic so that derivative terms which involve factors are bounded. Here, because we have vertex and face-average angular flux unknowns, we can use a parabolic interpolant with monotonization by the BKM method [34] which results in a nonlinear, monotonic, C_1 continuous, third-order $O(h^3)$

Algorithm 4: Our Subcell Balance Algorithm.

for *each direction* $\vec{\Omega}_m$ **do**

  for *each cell* c **do**

    for *each incoming face* f_{IN} **do**

      Use 2 known incoming vertex and 1 face-average angular fluxes $\psi^{v_{IN}}$ and $\psi^{f_{IN}}$ to construct a parabolic interpolation of the angular flux on the face, $Y^{f_{IN}}(\ell)$.

    end

    With all incoming face distributions $Y^{f_{IN}}(\ell)$ specified and total source within a cell known $q(\vec{r})$, characteristics may be used to determine the angular flux at any position within the cell, $Y^c(\vec{r})$.

    for *each outgoing vertex* v_{OUT} **do**

      Calculate outgoing vertex value $\psi^{v_{OUT}}$ by evaluating $Y^c(\vec{r})$ at each outgoing vertex.

    end

    for *each outgoing face* f_{OUT} **do**

      Calculate outgoing face-average $\psi^{f_{OUT}}$ by integrating $Y^c(\vec{r})$ over the outgoing face.

    end

    Calculate the cell-average ψ^c by integrating $Y^c(r)$ over the cell.

    ****Note that evaluating the relevant integrals over faces and cells is most easily accomplished by dividing the cell into angularly-dependent subcells (slices).*

  end

end

accurate representation of the angular flux on a face, $Y^{f_{IN}}(\ell)$, and is thus $O(h^2)$ accurate globally. Basically, if there is an extremum of the parabola $Z^c(\ell) = a\ell^2 + b\ell + c$, which is constructed by satisfying 2 vertex values and 1 average value on the face, a piecewise (but smooth) interpolator is comprised of a parabolic part and a constant part. Additional details are provided in Appendix B.

3.2.2 Determining QD Factors from Subcell Balance Characteristics

With SCSB, it is trivial to determine the QD factors at cells and faces from cell-average and face-average angular fluxes,

$$E^c_{\alpha\beta} = \frac{\sum_m w_m \Omega_{\alpha,m} \Omega_{\beta,m} \psi^c_m}{\sum_m w_m \psi^c_m}, \tag{3.13}$$

$$E^f_{\alpha\beta} = \frac{\sum_m w_m \Omega_{\alpha,m} \Omega_{\beta,m} \psi^f_m}{\sum_m w_m \psi^f_m}, \tag{3.14}$$

$$C^{f_b} = \frac{\sum_{m \in OUT} w_m (\vec{n}^{f_b} \cdot \vec{\Omega}_m) \psi^{f_b}_m}{\sum_{m \in OUT} w_m \psi^{f_b}_m}. \tag{3.15}$$

3.3 Error Discussion

Here we investigate the propagation of errors in ψ to errors in E which may impact the LOQD solution. Consider representing a component of the QD tensor as

$$E_{\alpha\beta} = \frac{K_{\alpha\beta}}{\phi},$$

where the second-moment is $K_{\alpha\beta} = \int_{4\pi} \Omega_\alpha \Omega_\beta \ \psi \ d\Omega$. Now consider representing our approximate angular flux, $\psi = \psi^* + \Delta\psi$, where ψ^* is the exact angular flux. We would like to determine the resulting error in the QD factor, $E = E^* + \Delta E$, where the subscripts have been dropped to avoid

clutter. We begin with the following error representation,

$$E^* + \Delta E = \frac{K^* + \Delta K}{\phi^* + \Delta\phi},$$

and divide top and bottom of the RHS by ϕ^*, expressing the numerator factor $\Delta K/\phi^*$ as $(\Delta K/K)(K/\phi^*)$ and using the definition $E^* = K^*/\phi^*$, we get

$$\frac{\Delta E}{E^*} = \frac{(\Delta K/K^*) - (\Delta\phi/\phi^*)}{1 + (\Delta\phi/\phi^*)}, \tag{3.16}$$

in terms of relative errors. In the case of small relative error in ϕ, $\Delta\phi/\phi^* < 1$, we can write a series expansion of the error

$$\frac{\Delta E}{E^*} = \left(\frac{\Delta K}{K^*} - \frac{\Delta\phi}{\phi^*}\right) \sum_{n=0}^{\infty} \left(-\frac{\Delta\phi}{\phi^*}\right)^n. \tag{3.17}$$

The formulas Eq. (3.17) and Eq. (3.16) illuminate one desirable aspect of the fractional factors: the relative error in the factor depends on the *difference* between the relative errors in the second and zeroeth moments. Thus if $\frac{\Delta K}{K^*}$ and $\frac{\Delta\phi}{\phi^*}$ are of the same sign, the QD factor E will be more accurate than either the zeroeth or second moment.

3.4 Scattering Source Representation

A representation of the scattering source must be specified that is compatible with the low-order problem unknown's centerings. In our case, we have face-average and cell-average scalar fluxes, ϕ^f and ϕ^c, from which we must construct a function of the scalar flux $\tilde{\phi}^c(\vec{r})$ in the cell to use in the scattering source $\frac{\sigma_s}{4\pi}\tilde{\phi}$ for the transport discretization, where the scattering cross-section σ_s is assumed constant in the cell. We introduce two approximations for $\tilde{\phi}$: the simple flat approximation,

$$\tilde{\phi}^c(\vec{r}) = \phi^c, \tag{3.18}$$

and the linear approximation,

$$\tilde{\phi}^c(\vec{r}) = \phi^c + \frac{1}{V^c} \sum_{f \in c} A^f \phi^f \vec{n}^f \cdot (\vec{r} - \vec{r}^c), \tag{3.19}$$

where $\vec{r}^c$ is the center of gravity of a cell. Eq. (3.19) is a convenient linear representation because it involves all faces and by definition, integration over any polygonal cell yields the cell-average ϕ^c. With both our SCV and SCSB discretizations, because they are monotonic, positivity of the angular flux is guaranteed if the scattering source is positive everywhere within a cell.

3.5 Difficulties with Characteristics through Arbitrary Cells

Compared to orthogonal meshes, a characteristics transport solver for randomized meshes (like SCV or SCSB) must handle a much larger range of optical thicknesses, $\sigma_t \Delta s$, where Δs is the mean subcell width in the direction of particle travel $\vec{\Omega}_m$. A representative distribution of subcell widths is shown in Fig. 3.5. The x-axis is the subcell center coordinates in terms of a coordinate system centered on the cell with the y-axis aligned with the direction of travel. The y-axis is the mean subcell width in the direction of travel (proportional to optical thickness), taken as the length of the line that passes center of subcell center and connects the entering and exiting edges of the subcell. On orthogonal meshes of rectangular cells, for a particular direction $\vec{\Omega}_m$, there are only 3 mean subcell widths. On randomized, single-level quadrilateral meshes there is a distribution about each of these 3 *and subcell widths which do not exist on orthogonal meshes.*

Also, the characteristic transport discretization should be valid for all σ_t, including voids where $\sigma_t \to 0$. The source term for the characteristic equation Eq. (3.1) includes integrals of the form, $\int_0^{\Delta s} q(s) \exp(-\sigma_t s)\ ds$. Using a naive integration and flat source $q(s) = q_0$, one obtains $(q_0/\sigma_t)(1 - \exp[-\sigma_t s])$. As $\sigma_t \to 0$, the naive result explodes, so one must use an expansion,

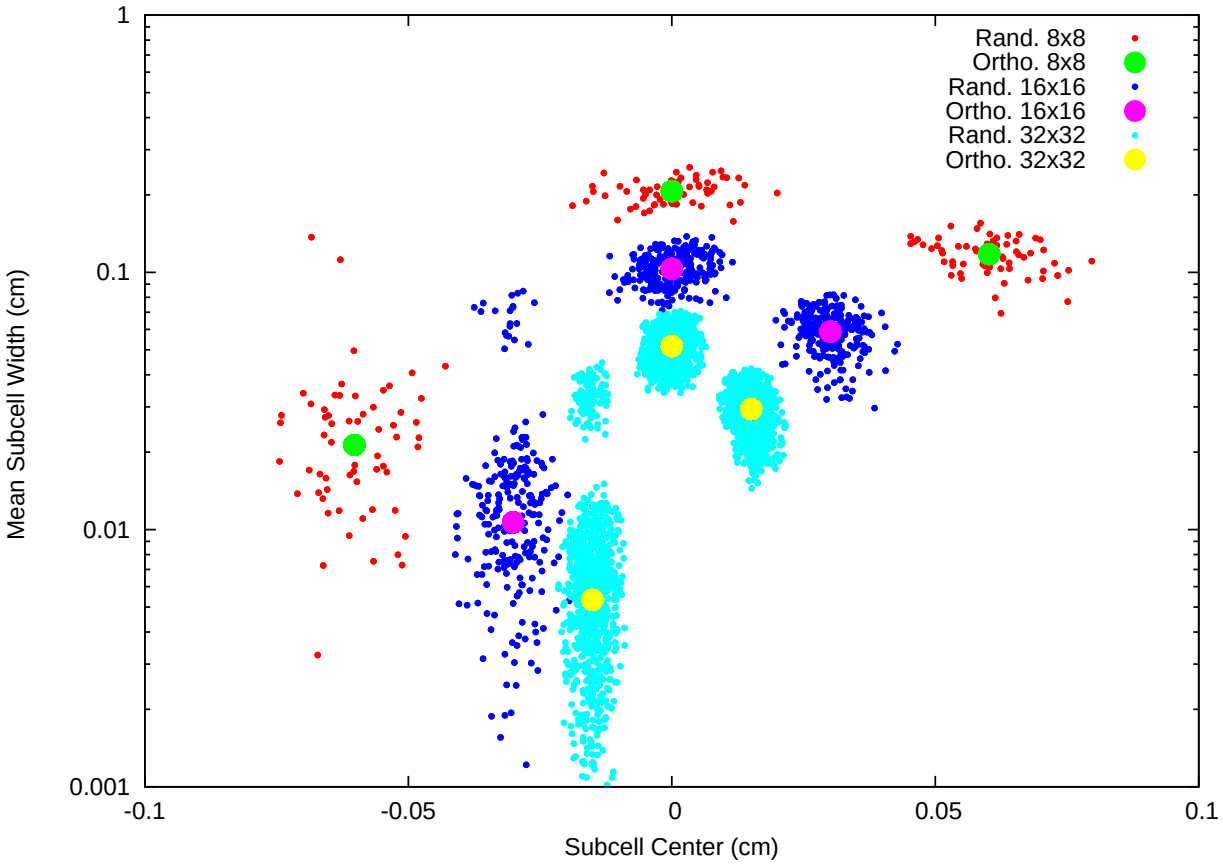


Figure 3.5: Representative distribution of subcell widths for randomized meshes.

$\exp[-\sigma_t \Delta s] \approx 1 - \sigma_t s + \sigma_t s^2/2 + ...$, and integrate the result. This handling of voids with $\sigma_t \to 0$ is straightforward for SCV.

However, for SCSB, the required averaging of Eq. (3.1) over subcells leads to some geometric terms that approach 1/0 for specific orientations of subcells. The most robust way to handle these integrations is by casting the solutions in terms of polynomial exponential moments [63]. We use a similar idea as [63] but have found advantage in choosing a different representation which we describe in Appendix A.1.

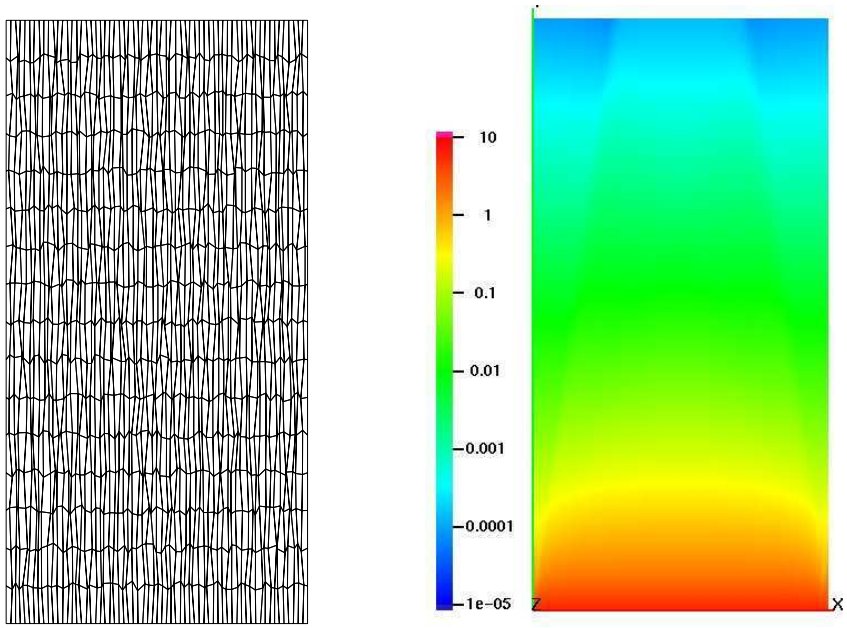
Another difficulty with characteristics on arbitrary meshes is grazing angles, where $\vec{n}^f \cdot$

$\vec{\Omega}_m \approx 0$, that is the outward normal of a face is near-perpendicular to an ordinate direction. There are no grazing angles on orthogonal meshes because standard quadrature sets do not have directions that lie on the xyz axes. Grazing angles lead to difficulties constructing 1) the transport sweep cell ordering and 2) calculating intersections and distances for the characteristics solver. For example, our robust handling of grazing angles with SCSB leads to no 1/0 cases for characteristics transport through a square cell along direction $(1, 0, 0)$.

3.6 Transport Test Problems

In this section, we perform various test problems on the SCV and SCSB transport discretizations proposed above, without consideration of the LOQD equations.

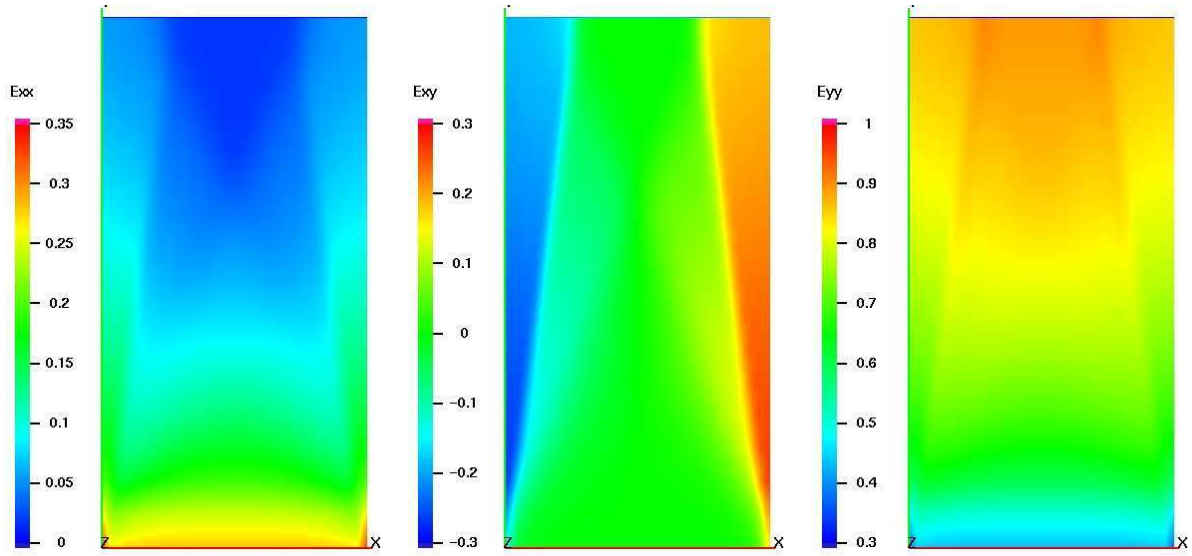
3.6.1 Pure Attenuation Test



(a) 64×16 randomized mesh (b) scalar flux ϕ

Figure 3.6: Pure Attenuation Test: Example mesh and scalar flux solution ϕ.

In this test we analyze pure attenuation (no scattering) in a homogeneous medium. The domain is 4cm $\times$ 8cm, the total cross section $\sigma_t = 1\text{cm}^{-1}$, scattering ratio $c = 0$, and there is a constant, incident angular flux on the bottom of the domain of $\psi_{IN} = 1^{\text{n}}/_{\text{cm}^2\cdot\text{s}}$. We choose to examime orthogonal and 20% randomized meshes with high aspect ratios in the test: the number of divisions in the x-direction is fixed, $N_x = 64$. The number of divisions in the y-direction, N_y



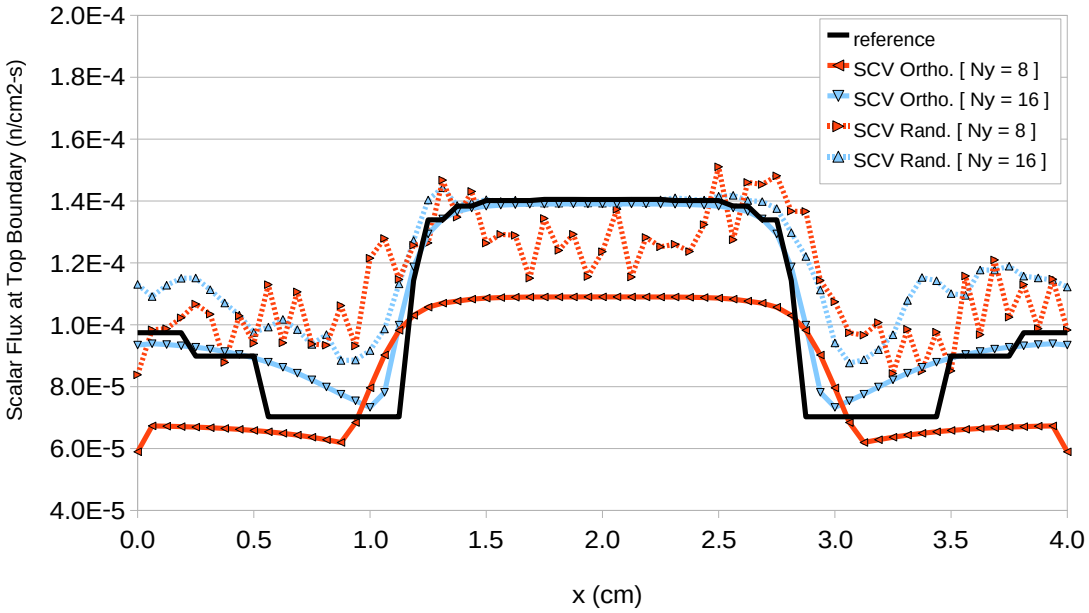
(a) E_{xx}. (b) E_{xy} (c) E_{yy}

Figure 3.7: Pure Attenuation Test: QD Factors.

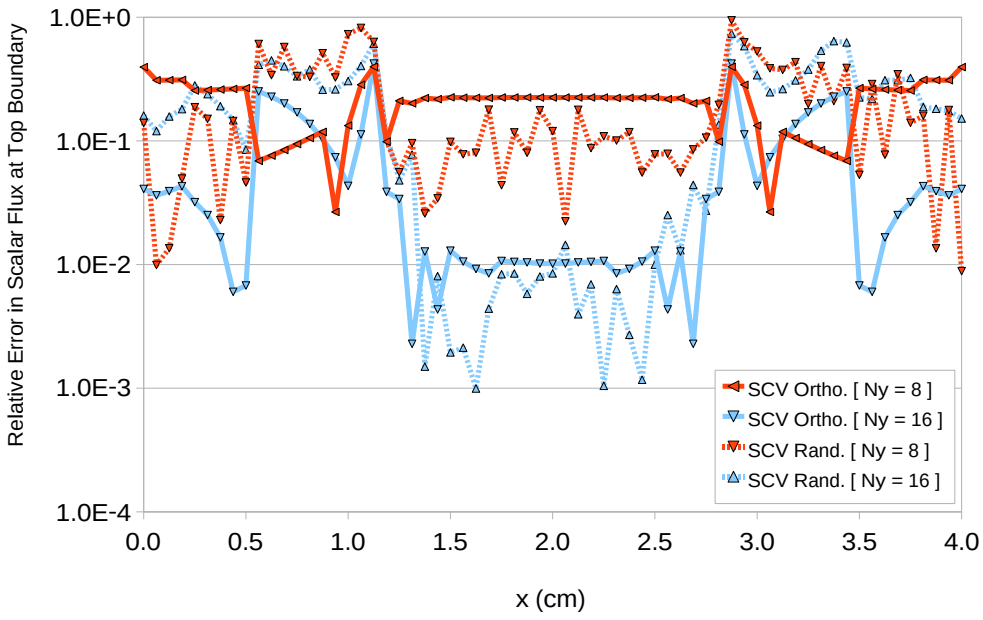
is varied. (On orthogonal meshes, $N_y = 2N_x$ leads to square cells.) The S_{16} level symmetric quadrature set is used [30]. A sample mesh with $N_y = 16$ and solution is shown in Fig. 3.6.

3.6.1.1 Data

We look at the scalar flux at the top (exiting side) of the domain, ϕ^T, for SCV, SCSB, bilinear discontinuous (BLD) discretization, and extended step characteristics (ESC). The relative error in ϕ^T compared to a reference long characteristics solution ϕ^T_{ref}. The relative error is given as $|\phi^T - \phi^T_{ref}|/\phi^T_{ref}$. Without scattering, the vertex-based long characteristics reference solution has error only due to the quadrature set. We also show our SCSB discretization without monotonization (SCSBnm).

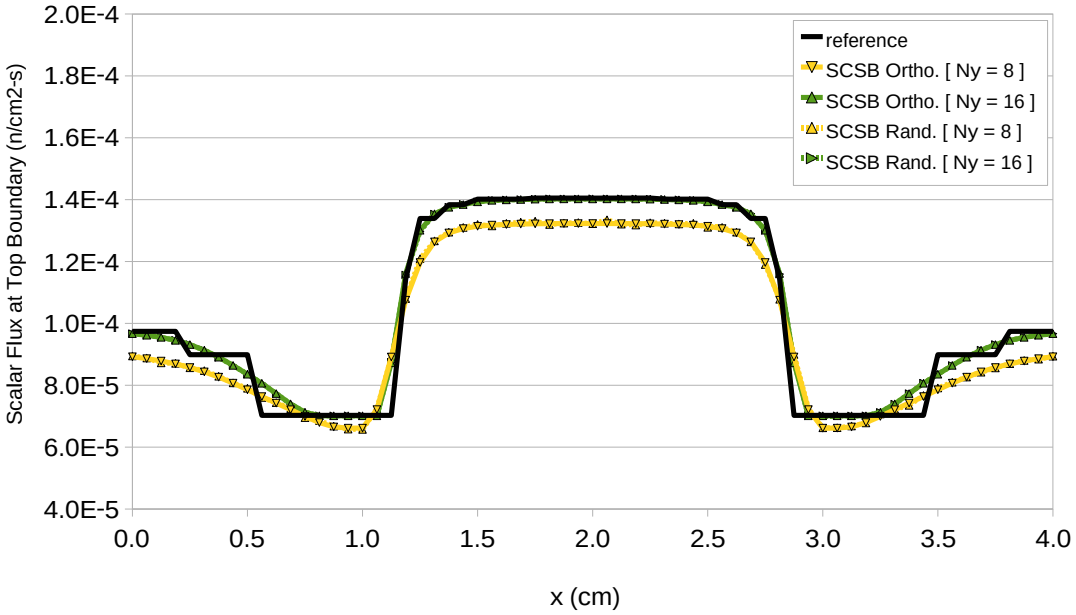


(a) SCV ϕ^T

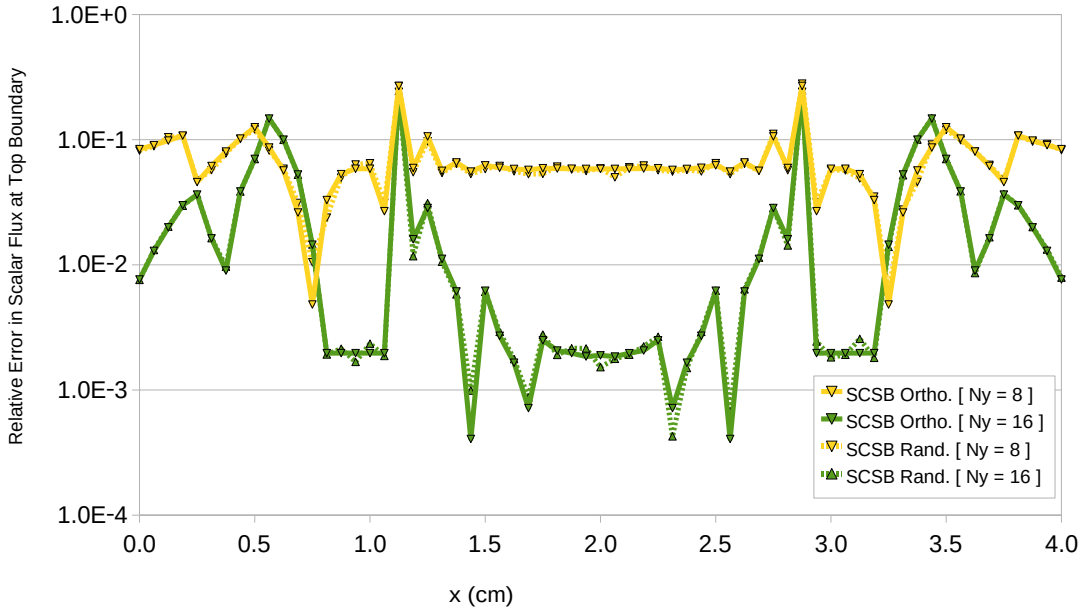


(b) SCV relative error

Figure 3.8: Pure Attenuation Test: SCV results.

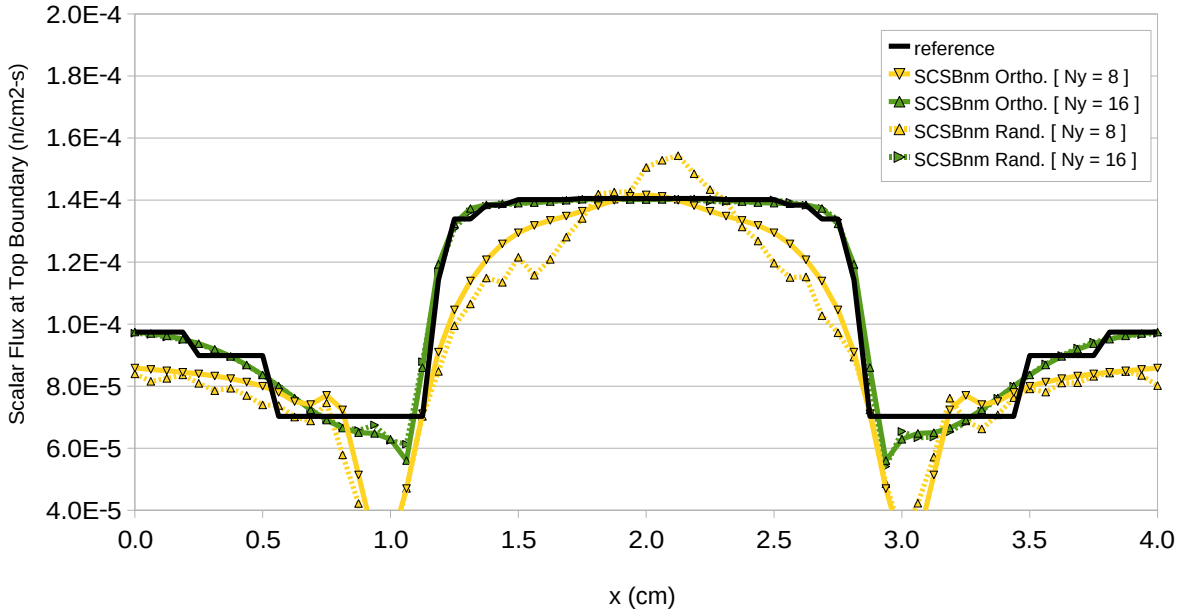


(a) SCSB ϕ^T

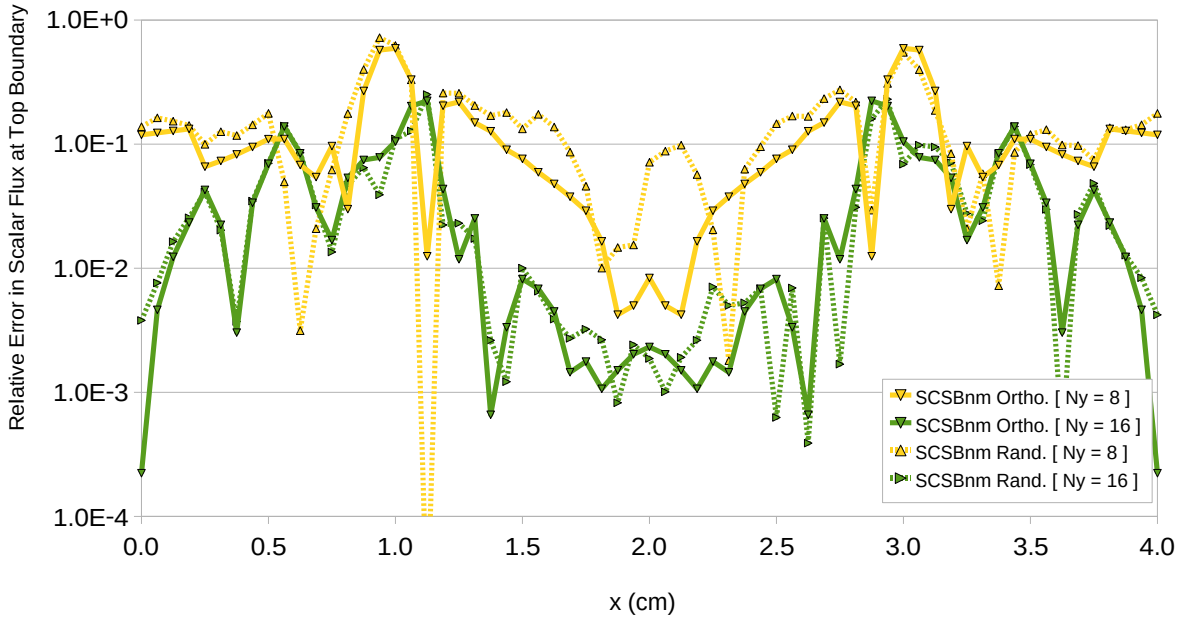


(b) SCSB relative error

Figure 3.9: Pure Attenuation Test: SCSB results.

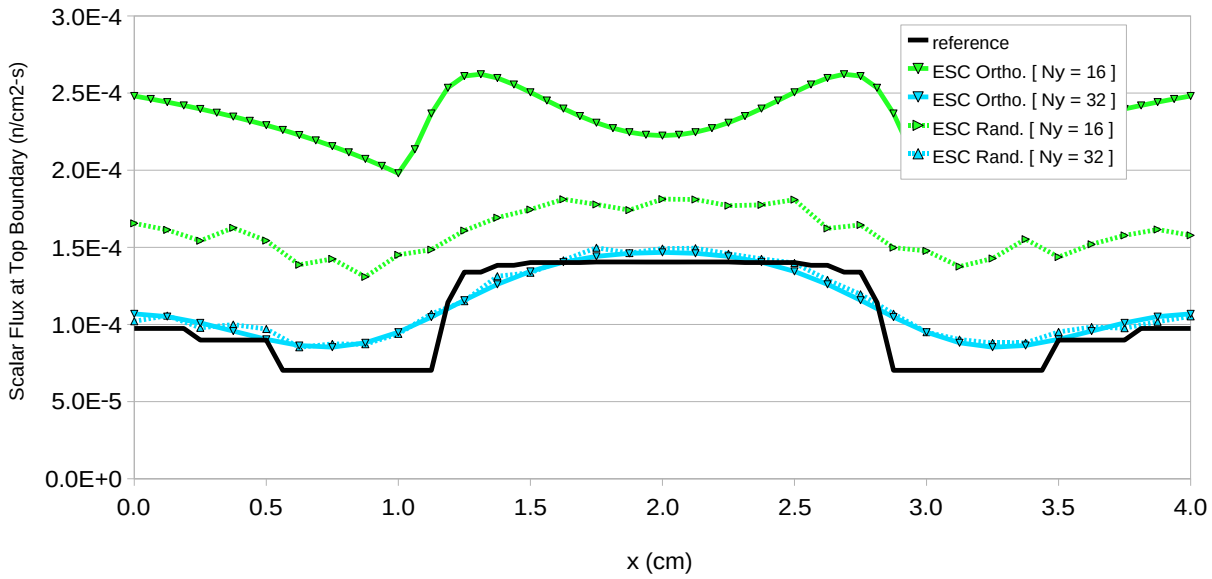


(a) SCSBnm ϕ^T

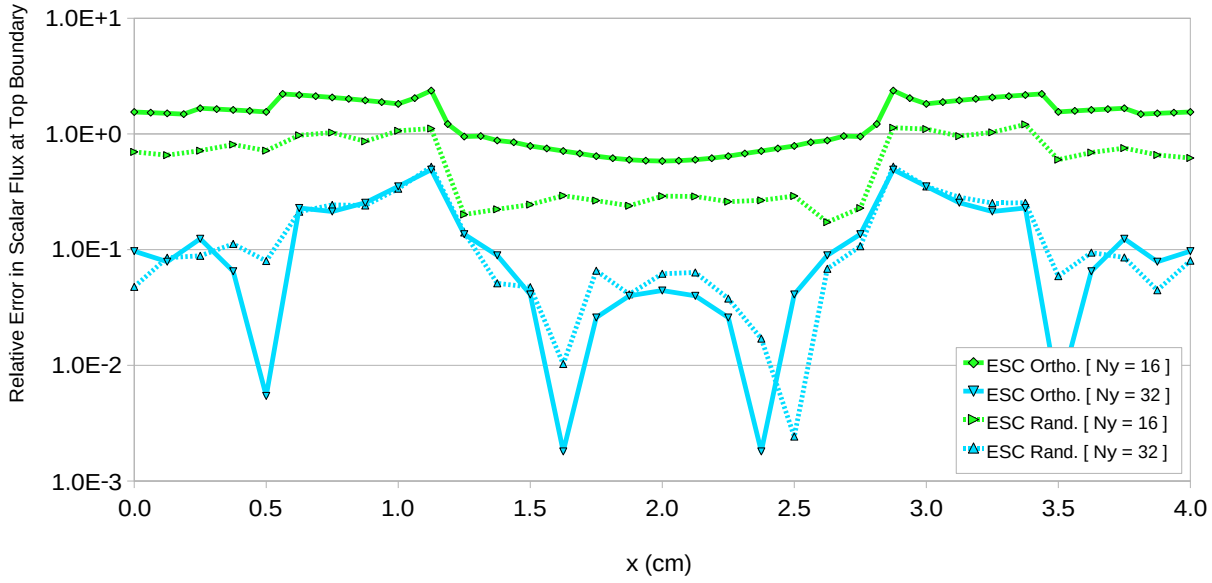


(b) SCSBnm relative error

Figure 3.10: Pure Attenuation Test: SCSB without monontization (SCSBnm) results.

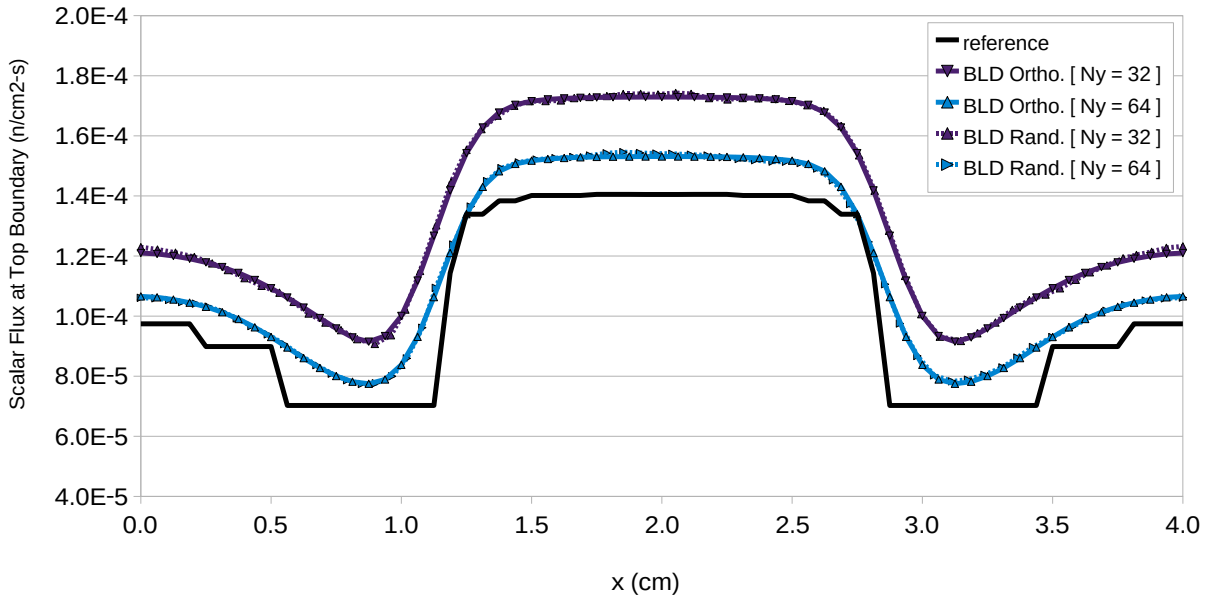


(a) ESC ϕ^T

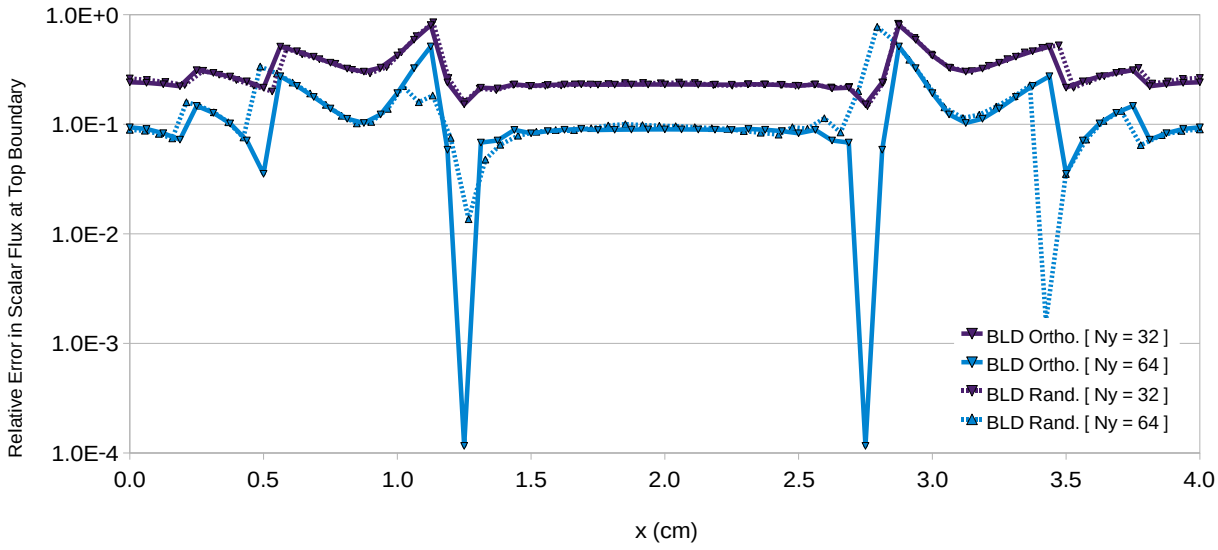


(b) ESC relative error

Figure 3.11: Pure Attenuation Test: ESC results.



(a) BLD ϕ^T



(b) BLD relative error

Figure 3.12: Pure Attenuation Test: bi-linear discontinous (BLD) results.

3.6.1.2 Analysis

Ray effects from S_N angular treatment lead to an "exact solution" at the top boundary that resembles a classic impulse problem, thus we may also visually compare how different discretizations smear (or disperse) the impulse. Because of dispersion and the coarseness of the mesh, all methods tested show a smooth solution near the edges, where the reference solution show "steps" due to ray effects and the discontinuity in the angular flux that is present for Dirichlet boundary conditions that do not agree at corners. In this test, the discontinuity occurs at the bottom left and right corner vertices, where the incoming angular flux changes from $\psi_{IN} = 1$ at the bottom to $\psi_{IN} = 0$ on the sides.

In Fig. 3.8, the SCV solution, despite monotonization, shows extreme oscillations on the randomized mesh with $N_y = 8$, where the aspect ratio is $h_y/h_x = 16$. On the $N_y = 16$ mesh where the aspect ratio is 8, the oscillations disappear. No oscillations appear for orthogonal meshes. The ability of SCV to have such wild oscillations, even on admittedly coarse and poorly conditioned meshes may preclude it from use for general transport problems on arbitrary meshes. With SCV relative errors on the $N_y = 8$ mesh are approximately 10% and on the $N_y = 16$ mesh are approximately 1%. To illustrate that even simple tests may lead to complicated QD factors, see Fig. 3.7.

On the other hand, SCSB in Fig. 3.9 shows virtually no difference between orthogonal and randomized meshes and none of the oscillations of SCV. The error of SCSB is approximately a factor of 2 lower than SCV. Without monotonization, SCSBnm shows in Fig. 3.10 drastically different results from SCSB for both orthogonal and randomized $N_y = 8$ meshes. Specifically, with SCSBnm, there is significant dispersion of the impulse and the shape of the impulse is not

smooth on randomized meshes. On the $N_y = 16$ meshes, the dispersion phenomena are less dramatic but still present compared to the SCSB results, where they are completely absent and smoothness of the impulse is mostly recovered. The relative errors for SCSBnm are approximately the same as SCV, a factor of 2 higher than in SCSB, where the monotonization is used. It is apparent from these results that monotonization is very important on randomized meshes, much more than on orthogonal meshes. On orthogonal meshes, the regularity of points allows simpler local representations (e.g. interpolation functions) to be adequate approximations of the exponential. On randomized meshes, a proper representation of an exponential must be produced for many more orientations of faces. Monotonizations enforce at least one essential property of the exponential on these local representations, monotonicity. The BKM monotonization, in particular, produces interpolators which are much closer to exponentials than parabolas.

In Fig. 3.11, ESC is shown with better aspect ratios of 8 and 4 for the $N_y = 16$ and $N_y = 32$ meshes, respectively, so that the solution is on a similar scale to SCV and SCSB. The solution on the orthogonal $N_y = 16$ mesh has very high dispersion because of the ESC's flat flux approximation. Also, because cells are "long" in the direction of radiation propagation and there is significant attenuation in a cell, vertical faces transfer too much radiation to outgoing faces. In this type of attenuation test, vertical faces are a "worst case scenario" for ESC and on randomized meshes there are fewer vertical faces, thus ESC does quite a bit better on coarser meshes ($N_y \leq 16$) with randomization than without. ESC has large relative errors of about 100% and 50% for the orthogonal and randomized $N_y = 16$ meshes, respectively, and about 20% relative error for $N_y = 32$ meshes.

In Fig. 3.12, BLD is shown with the same aspect ratios as ESC, again so that the solu-

tion is on a similar scale to SCV and SCSB. BLD shows little difference between orthogonal and randomized meshes and converges very uniformly to the reference solution, although errors are considerably larger than SCSB with 30% and 10% approximate relative error on the $N_y = 16$ and $N_y = 32$ meshes, respectively. It may appear from these results that BLD is a good candidate for QD on arbitrary meshes, exhibiting seemingly monotonic behavior and small differences between orthogonal and randomized meshes. However, BLD is not monotonic and can produce negative angular fluxes for problems with strong material discontinuities.

3.6.2 Analytic Transport Test

Here, we perform an analytic test from [108]. The domain is a 1cm $\times$ 1cm square with homogeneous material properties, $\sigma_t = 1.0\text{cm}^{-1}$, with scattering $\sigma_s = 0.5\text{cm}^{-1}$, and the the exact angular flux is given as

$$\psi_{exact}(x, y, \Omega_x, \Omega_y) = (1 + \Omega_x^2 + \Omega_y^2)(x^2 - x^4)(y^2 - y^4). \tag{3.20}$$

The boundary conditions are vacuum, $\psi_{IN} = 0$. (Note this angular flux is zero on the domain boundary for all directions $\vec{\Omega}$.) The exact scalar flux is given as

$$\phi_{exact}(x, y) = \frac{20\pi}{3} x^2 y^2 (1 - x^2)(1 - y^2), \tag{3.21}$$

shown in Fig. 3.14(a). We use the S_{12} level symmetric quadrature set. By substituting Eq. (3.20) into the transport equation of Eq. (1.3), we arrive at anisotropic external source,

$$q_{ext}(x, y, \Omega_x, \Omega_y) = \alpha \left(\Omega_x \frac{\partial \gamma}{\partial x} + \Omega_y \frac{\partial \gamma}{\partial y} \right) + \left(\sigma_t \alpha - \frac{5\sigma_s}{3} \right) \gamma, \tag{3.22}$$

where $\alpha(\Omega_x, \Omega_y) = 1 + \Omega_x^2 + \Omega_y^2$ and $\gamma(x, y) = (x^2 - x^4)(y^2 - y^4)$. Note that the source may be negative.

Table 3.1: Analytic Transport Test: ℓ_2 error norm.

	Ortho.		Rand. 10%		Rand. 20%		Rand. 30%	
$1/h$	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*
1	2.47E-2	-	2.47E-2	-	2.47E-2	-	2.47E-2	-
2	2.67E-2	0.92	2.59E-2	0.96	2.36E-2	1.05	2.26E-2	1.09
4	4.49E-3	5.95	4.48E-3	5.77	5.17E-3	4.56	6.55E-3	3.46
8	7.47E-4	6.02	7.29E-4	6.15	8.20E-4	6.30	1.12E-3	5.87
16	1.13E-4	6.59	1.03E-4	7.07	1.23E-4	6.65	1.51E-4	7.37
32	1.75E-5	6.47	1.60E-5	6.43	2.26E-5	5.46	3.67E-5	4.13
64	2.88E-6	6.07	2.63E-6	6.09	3.53E-6	6.39	5.63E-6	6.51

3.6.2.1 Data

We present results for the SCSB discretization with linear source representation. Because LOQD scalar fluxes are not available, the linear scattering source is constructed from high order scalar fluxes and the transport equation is iteratively solved with SI, which converges in a reasonable number of iterations because the scattering ratio is only $c = 0.5$. Table 3.1 shows the ℓ_2 error norm for the scalar flux on orthogonal and 10%, 20%, and 30% randomized meshes. The distribution of error is shown in Fig. 3.14(b) and Fig. 3.14(c) for a 64×64 orthogonal and randomized mesh, respectively.

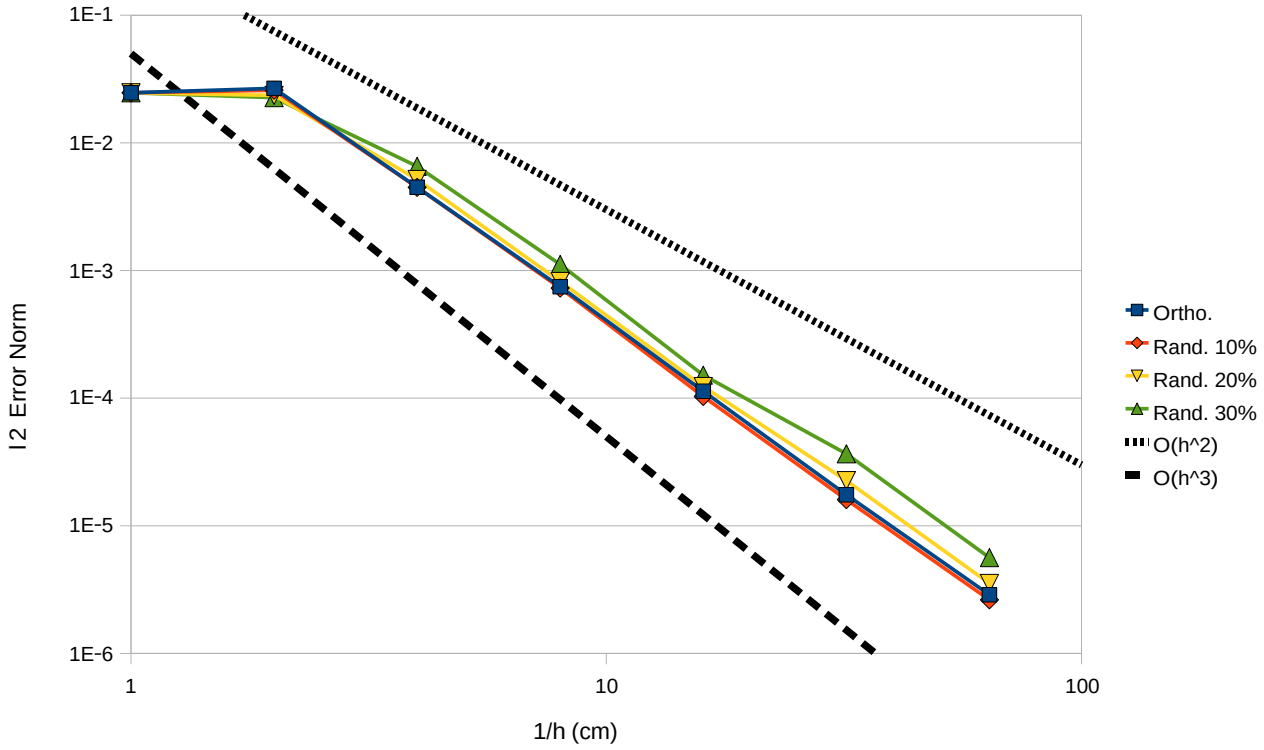
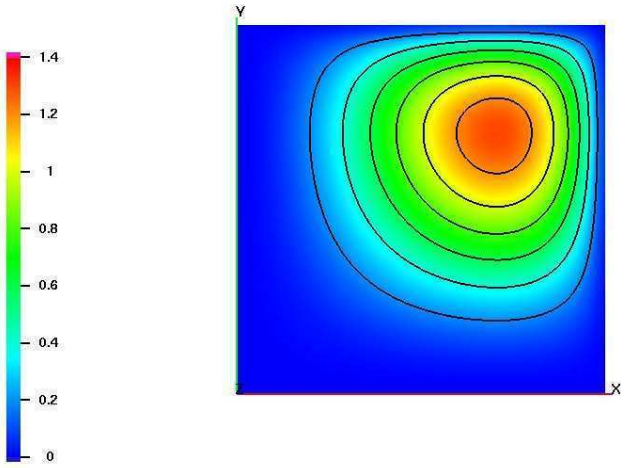
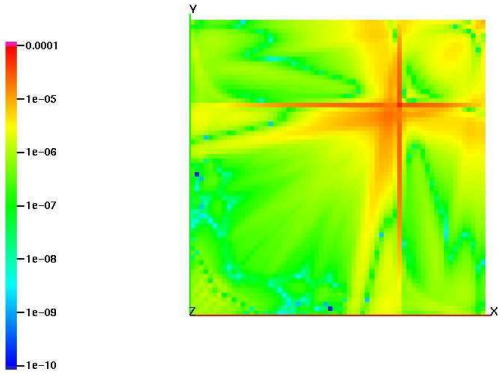


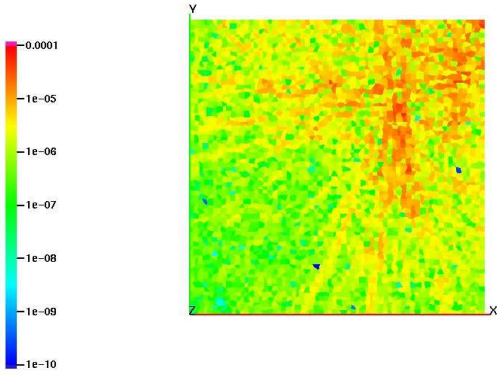
Figure 3.13: Transport Analytic Test: ℓ_2 error norm.



(a) The analytic solution $\phi_{exact}(x, y)$.



(b) orthogonal



(c) randomized

Figure 3.14: Transport Analytic Test: (a) analytic solution and absolute error in cell-averages $|\phi^c - \phi^c_{exact}|$ for the high-order analytic test on a 64×64 orthogonal (b) and randomized (c) mesh.

3.6.2.2 Analysis

This test exhibits numerical convergence between $O(h^2)$ and $O(h^3)$ with convergence ratios between 5 and 6 ($Ratio = 8$ indicates third order), tabulated in Table 3.1 and plotted in Fig. 3.14. The angular flux convergence in the ℓ_2 norm (not shown) exhibits closer to $O(h^2)$ convergence, with convergence ratios of approximately 4.5, so there appears to be a consistent cancellation of errors which leads to scalar fluxes that are more accurate than angular fluxes. This is not that unusual: the scalar flux involves weighted summation of angular fluxes over all directions. If the angular fluxes have error terms with different signs, the error in the scalar flux will be smaller (in a relative sense.) However, this only affects the coefficient of convergence not, the order. It is also possible that asymptotic convergence is not yet achieved for this test and the numerical convergence ratios in are tending toward 4. Even though on the finest mesh, $\sigma_t h \approx 0.016$, cells are still optically thick in some directions, $\sigma_t h / \sin\theta \approx 0.07$ for the smallest $\sin\theta$ in the S_{12} quadrature set.

3.7 Summary

We have presented two characteristics discretizations for transport problems on unstructured meshes, short characteristics on vertices (SCV) and (conservative) short characteristics with subcell balances (SCSB).

Short characteristics on vertices is not new, but we have developed a way to calculate face-average and cell-average QD factors from the vertex values that is acceptable for unstructured meshes. SCV was used as the high-order QD discretization for the tests in Sec. 2.3.2 where it was shown to produce adequate factors for the LOQD discretization on arbitrary quadrilateral meshes, including meshes with hanging nodes. Results in this chapter show that SCV can have poor

behavior on optically thick, skewed cells with high aspect ratios. Also, SCV was found to be only $O(h)$ on randomized meshes—for orthogonal meshes it is $O(h^2)$. Thus the LOQD discretization on arbitrary meshes that uses SCV is limited to first-order because the calculated QD factors will be first-order. However, because of cancellation of errors (see Sec. 3.3), QD factors $\mathrm{E}[\psi]$ may have *absolute error* that is considerably less than the solutions themselves. This means that the LOQD solution can numerically exhibit the second-order on finite meshes, even though factors are only first-order. Another way to think about this is by considering $\mathrm{E}[\psi]$ as data for the LOQD problem. This was exactly the case in Sec. 2.3.2.

We have improved upon SCV with a short characteristics with subcell balances (SCSB). It has the advantage that it is $O(h^2)$ for randomized meshes, permits direct calculations of face-average and cell-average QD factors, and behaves well even on optically thick, skewed cells with high aspect ratios. Additionally with SCSB, the variation between problems on orthogonal and randomized meshes is much smaller than SCV.

We have also developed a linear scattering source representation that may be used with SCSB (or SCV), for which numerical results will be presented in Chapter 5.

Chapter 4

QUASIDIFFUSION SOLVERS

Our FV discretization of the LOQD equations results in a non-symmetric linear system with matrix order $N_{LOQD} = 9N_C$, where N_C is the number of cells in the mesh. In matrix form we have

$$Ax = b, \tag{4.1}$$

where A is the coefficient matrix, b is the RHS, and x is the solution vector of unknowns from the general finite volume framework described in Chapter 2. Those unknowns are cell-average scalar fluxes ϕ^c, face-average scalar fluxes ϕ^f, and face-average normal currents J^f. This chapter focuses on efficient solution methods for the LOQD matrix equations using iterative methods. We also present timing results and briefly discuss implementation details for the transport discretizations described in Chapter 3.

4.1 Two Well-structured LOQD Systems

We have determined two unknown/equation orderings that lead to a well-structured LOQD system, as defined in [91], in the sense that all diagonal entries are non-zero.[1] First we describe a cell-based structure which is organized into blocks and outlying "bands". Then we describe an essential structure, which implicitly includes interface conditions and leads to a reduced number of face-based unknowns.

4.1.1 Cell-based Structure

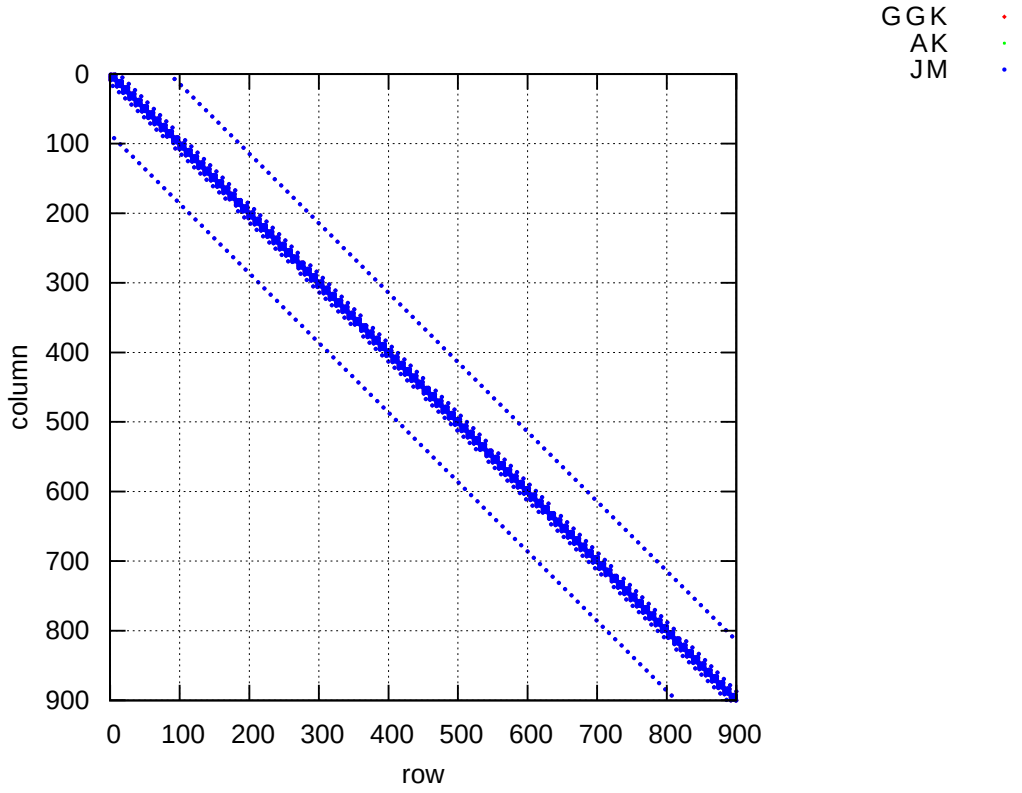


Figure 4.1: Global cell-based structure of the LOQD sparse matrix with GGK, AK, and JM discretizations on a 10×10 randomized mesh.

[1] More specifically, diagonal entries are non-zero for $\sigma_t > 0$. For voids where $\sigma_t = 0$, the continuous LOQD equations of Eq. (1.4) are singular. In practice, the resultant poor conditioning of the near-singular system allows a minimum $\sigma_t \approx 10^{-8}$.

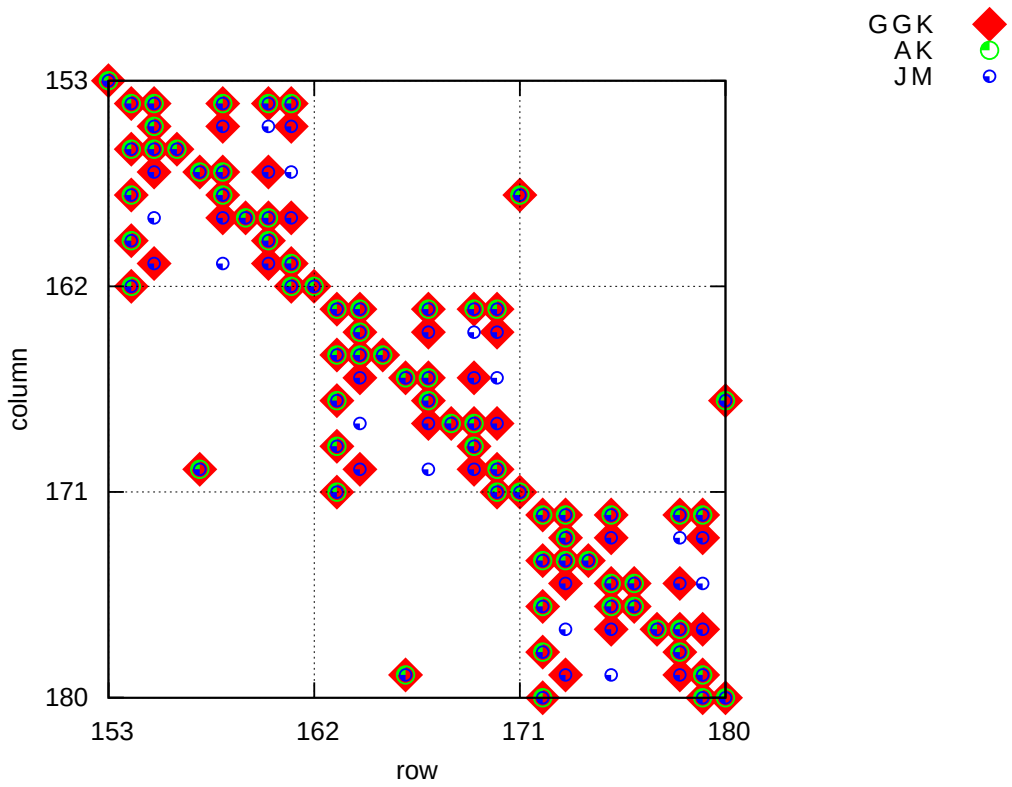


Figure 4.2: Local cell-based structure of the LOQD sparse matrix with GGK, AK, and JM discretizations on a 10×10 randomized mesh.

Our general FV framework for quadrilateral cells has 1 balance equation and 4 first-moment equations per cell. Interface conditions plus boundary conditions provide another 4 equations per cell, leading to a system of order $N_{LOQD} = 9N_C$. We order the solution vector by cell-local unknowns and arrive at a system which has N_C blocks (each 9×9) with 4 "bands" due to interface and boundary conditions, as in Fig. 4.1. For logically-rectangular meshes, the interface and boundary conditions will actually lie on bands for certain cell-orderings. For arbitrary meshes, this cannot be assured and the interface and boundary condition outliers will have no global pattern. The local cell-based structure is shown in Fig. 4.2 where the supports of three FV discretizations on non-orthogonal meshes may be compared. On non-orthogonal meshes, our JM discretization

has support of all faces in the cell and thus has the largest number of non-zero values per block with 33 for interior cells. The GGK discretization has a slightly smaller support with 29 and the AK discretization has 21. Cells which have a face on the boundary have one more non-zero value per face in the 9×9 block.

The key to the cell-based structure is in ordering face-based unknowns to coincide with interface/boundary and first moment equations. In discussing this, we use the the term "associate equation X with an unknown Y", by which we mean "place equation X so that the coefficient corresponding to unknown Y is on the diagonal."

1. For each cell, c, associate the balance equation with ϕ^c.

2. For each boundary face f_b in cell c,

   - associate the boundary condition with J^{f_b}. This way, for reflective boundary conditions which have $J^{f_b} = 0$, there is still a nonzero diagonal.
   - Then associate the first moment equation with ϕ^{f_b}.

3. For each interior face f in cell c,

   - associate an interface condition with either ϕ^f or J^f, depending on if it is a scalar flux or current interface condition, respectively.
   - Then, associate the first moment equation *with the other unknown*, taking advantage of the fact that the first moment equation for face f is guaranteed to have nonzero coefficients associated with ϕ^f and J^f (as long as $\sigma_t > 0$).

As an example, the cell-based structure is shown in Fig. 4.3 and Fig. 4.4, for a 1×1 and 2×1 mesh, respectively.

$$\begin{pmatrix} \text{Bal. Eqn. for cell } c \\ 1^{st} \text{ Mom. Eqn. for face } f_B \\ \text{BC for face } f_B \\ 1^{st} \text{ Mom. Eqn. for face } f_R \\ \text{BC for face } f_R \\ 1^{st} \text{ Mom. Eqn. for face } f_T \\ \text{BC for face } f_T \\ 1^{st} \text{ Mom. Eqn. for face } f_L \\ \text{BC for face } f_L \end{pmatrix} \begin{pmatrix} \phi^c \\ \phi^{f_B} \\ J^{f_B} \\ \phi^{f_R} \\ J^{f_R} \\ \phi^{f_T} \\ J^{f_T} \\ \phi^{f_L} \\ J^{f_L} \end{pmatrix}$$

Figure 4.3: Equations and unknown ordering for a 1×1 cell-based system with a single cell.

$$\begin{pmatrix} \text{Bal. Eqn. for cell } c_1 \\ 1^{st} \text{ Mom. Eqn. for face } f_{B1} \\ \text{BC for face } f_{B1} \\ \text{Scalar Flux IC for face } f_{R1} \\ 1^{st} \text{ Mom. Eqn. for face } f_{R1} \\ 1^{st} \text{ Mom. Eqn. for face } f_{T1} \\ \text{BC for face } f_{T1} \\ 1^{st} \text{ Mom. Eqn. for face } f_{L1} \\ \text{BC for face } f_{L1} \end{pmatrix} \begin{pmatrix} \phi^{c_1} \\ \phi^{f_{B1}} \\ J^{f_{B1}} \\ \phi^{f_{R1}} \\ J^{f_{R1}} \\ \phi^{f_{T1}} \\ J^{f_{T1}} \\ \phi^{f_{L1}} \\ J^{f_{L1}} \end{pmatrix} \qquad \begin{pmatrix} \text{Bal. Eqn. for cell } c_2 \\ 1^{st} \text{ Mom. Eqn. for face } f_{B2} \\ \text{BC for face } f_{B2} \\ 1^{st} \text{ Mom. Eqn. for face } f_{R2} \\ \text{BC for face } f_{R2} \\ 1^{st} \text{ Mom. Eqn. for face } f_{T2} \\ \text{BC for face } f_{T2} \\ 1^{st} \text{ Mom. Eqn. for face } f_{L2} \\ \text{Current IC for face } f_{L2} \end{pmatrix} \begin{pmatrix} \phi^{c_2} \\ \phi^{f_{B2}} \\ J^{f_{B2}} \\ \phi^{f_{R2}} \\ J^{f_{R2}} \\ \phi^{f_{T2}} \\ J^{f_{T2}} \\ \phi^{f_{L2}} \\ J^{f_{L2}} \end{pmatrix}$$

LEFT CELL RIGHT CELL

Figure 4.4: Equations and unknown ordering for a 2×1 cell-based system with two cells.

The cell-based structure must be used with hanging-node meshes unless we are willing to consider polygonal cells and increase the number of face-based unknowns accordingly.

4.1.2 Essential Structure

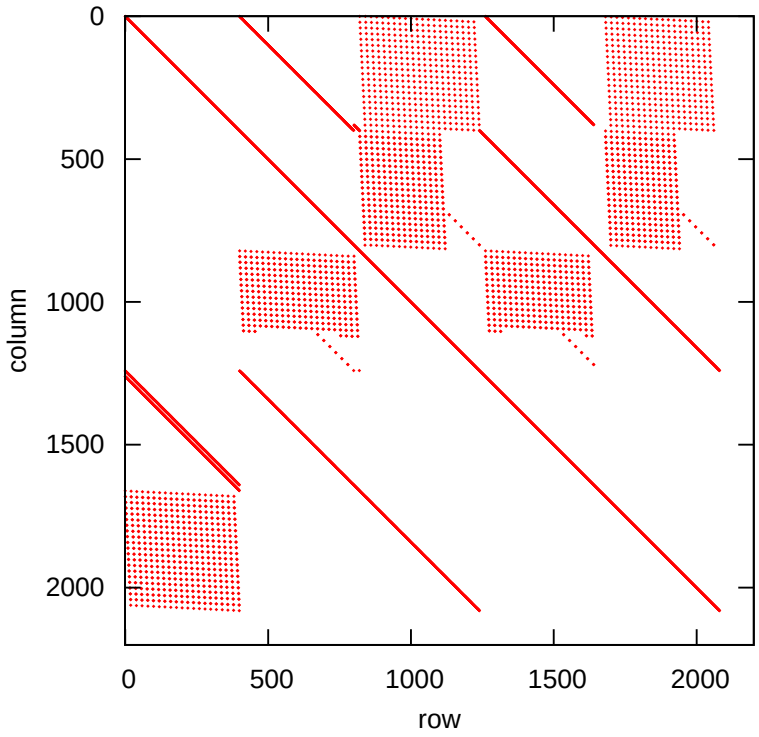


Figure 4.5: Global essential structure of the LOQD sparse matrix .

The essential structure is for meshes which have only *standard interfaces* (no hanging nodes), which by definition have 2 faces per interface. On these meshes, it is possible to satisfy the standard interface conditions implicitly by considering the 2 faces as a single face referenced by the same global index, cutting the number of interior face unknowns by half. Assuming an $N \times N$ logically rectangular mesh, the essential structure has order $N_{LOQD} = 5N^2 + 4N$, compared to the cell-based structure order, $N_{LOQD} = 9N^2$. Thus, in the limit of large meshes, the essential system

has 5/9 the number of unknowns!

For the essential structure, we order the solution vector by type of unknown,

$$x = (\quad \phi^{c_1}, \phi^{c_2}, \phi^{c_3}, ..., \phi^{c_{N_C}} \quad , \tag{4.2}$$

$$\phi^{f_1}, \phi^{f_2}, \phi^{f_3}, ..., \phi^{f_{N_F}} \quad , \tag{4.3}$$

$$J^{f_1}, J^{f_2}, J^{f_3}, ..., J^{f_{N_F}} \quad). \tag{4.4}$$

Balance equations are associated with the first N_C rows, boundary conditions are associated with the next N_{BF} rows, and first moment equations are associated with the final $2N_F - N_{BF}$. This ordering results in a sparse matrix with structure shown in Fig. 4.5. Although the essential structure has a reduced unknown count, it has a complicated structure and very large bandwidth because two adjacent cells share a face with a single global index.

4.1.3 Condition Numbers

To analyze the conditioning of the LOQD system, let us define the condition

$$\kappa(A) = \|A^{-1}\|_1 \|A\|_1,$$

where we have used the 1-norm of a matrix. The condition number is useful in estimating the precision of the result of solution of the system of equations. Generally, with a condition number of $\kappa(A)$, one can only expect the iterative solution to have precision $\epsilon \ \ \kappa(A)$ [85], where ϵ is the computer's working precision. Thus, with double precision (approximately 16 digits) a condition number of $\kappa = 10^{10}$ would lead to a maximum of 6 digits of accuracy. On logically rectangular $N \times N$ meshes, we observe the essential structure to have condition numbers that grow as $O(h^{-2})$. This is not unusual for 2D systems discretized by finite differences [85], which our difference formulas

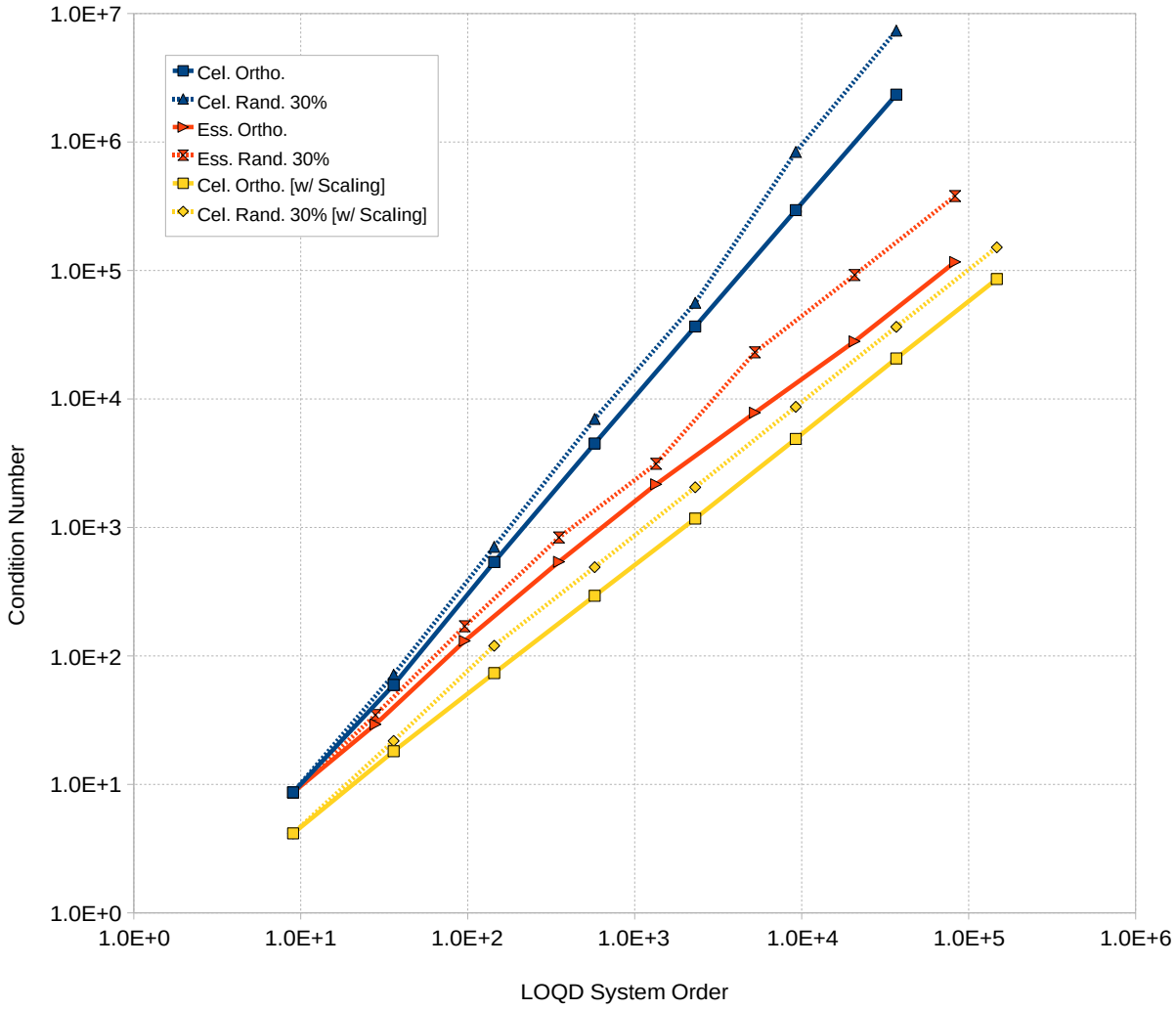


Figure 4.6: Condition numbers for the essential (Ess.) LOQD system and cell-based (Cel.) LOQD system.

of Eq. (2.18) used in the first moment equations in Eq. (2.20b) resemble. On the same tests, we observe the cell-based structure has condition numbers that grow as $O(h^{-3})$. The $O(h^{-3})$ growth for the cell-based structure is due to the "bands" which always have the value ± 1 and lie $O(N)$ columns from the diagonal blocks. Note, on these logically rectangular systems, $h = 1/N$. See Fig. 4.6 for representative condition numbers. The $O(h^{-3})$ growth for the cell-based structure is

reduced to $O(h^{-2})$ by column-scaling, described in Sec. 4.2.1. The condition numbers approach 10^7 for the cell-based structure on a 64×64 mesh, the last data point shown in Fig. 4.6.

4.2 Scaling, Preconditioning, and Solving the LOQD System

Here we investigate the efficient solution of the LOQD systems described in the previous section. First, we will investigate scaling the relevant equations and unknowns to lower the condition number. Then we will investigate Krylov solvers for the iterative solution. Finally, we will investigate preconditioning of the equations via incomplete LU factorization-based algebraic preconditioners (ILU).

4.2.1 Scaling

Scaling is a fairly cheap operation and should be employed solely because cross-section parameters σ_t and σ_a may vary by orders of magnitude in different regions of practical problems. We investigate both column and row scaling of the linear system $Ax = b$. Row-scaling calculates diagonal matrix,

$$R^{-1} = \frac{1}{||\vec{A}_i||_p}, \tag{4.5}$$

based on p-norms of each row $\vec{A}_i$ and column-scaling calculates diagonal matrix

$$C^{-1} = \frac{1}{||\vec{A}_j||_p}, \tag{4.6}$$

based on p-norms of each column $\vec{A}_j$. The resulting scaled system is

$$\tilde{A}\tilde{x} = \tilde{b}, \tag{4.7}$$

with

$$\tilde{A} = R^{-1}AC^{-1}, \tag{4.8}$$

$$\tilde{x} = Cx, \tag{4.9}$$

$$\tilde{b} = R^{-1}x. \tag{4.10}$$

After iterations converge, the original unknowns, x, are recovered via Eq. (4.9).

In our implementation, we use the ∞-norm, $p = \infty$ and first do column-scaling (which is equivalent to unknown scaling) and then apply row-scaling to the column-scaled matrix. We use ∞-norms for scaling because it leads to maximum entries of 1 which in turn leads to predictable tolerances for the incomplete LU (ILU) factorization preconditioners.

As shown in Fig. 4.6, column-scaling reduces the condition number growth for the cell-based structure from $O(h^{-3})$ to $O(h^{-2})$. Additional results for scaling will be deferred until preconditioners are introduced.

4.2.2 Iterative Solvers

We use iterative methods to solve our LOQD system, as opposed to direct methods. This is because of the well-known fact that, quoting from [85], "[direct methods] scale poorly with problem size in terms of operation counts and memory requirements, especially on problems arising from the discretization of PDEs ... [which] lead to linear systems comprising hundreds of millions or even billions of equations in as many unknowns. For such problems, iterative methods are the only option available." We have used iterative methods (with proper preconditioning) to solve the sparse LOQD system orders of magnitude faster than sparse direct methods.

We compare the following Krylov methods for nonsymmetric systems: generalized minimum residuals (GMRES), transpose-free quasi-minimum residuals (TFQMR), and bi-conjugate gradients with stabilization (BiCGstab). Fig. 4.7 shows a representative test where BiCGstab, TFQMR, and GMRES are compared by number of matrix vector multiplies to achieve a 10^{-7} residual. Fig. 4.8 shows BiCGstab solves the LOQD system of equations fastest, followed closely by TFQMR, and GMRES with approximately double the runtime. Because the LOQD systems cannot be solved iteratively without preconditioning, the representative test has row and column scaling and an ILUD preconditioner with drop tolerance $\epsilon_{drop} = 0.01$, which will be introduced in the next section. Most tests we performed indicated, as in Fig. 4.8, that BiCGstab was the fastest solver.

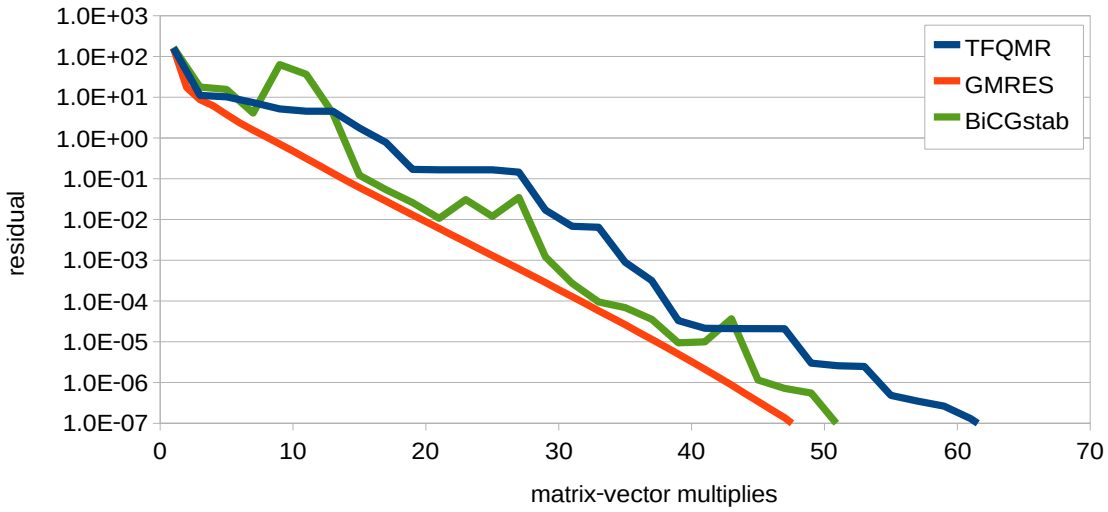


Figure 4.7: Krylov solver residual vs. matrix-vector multiplies.

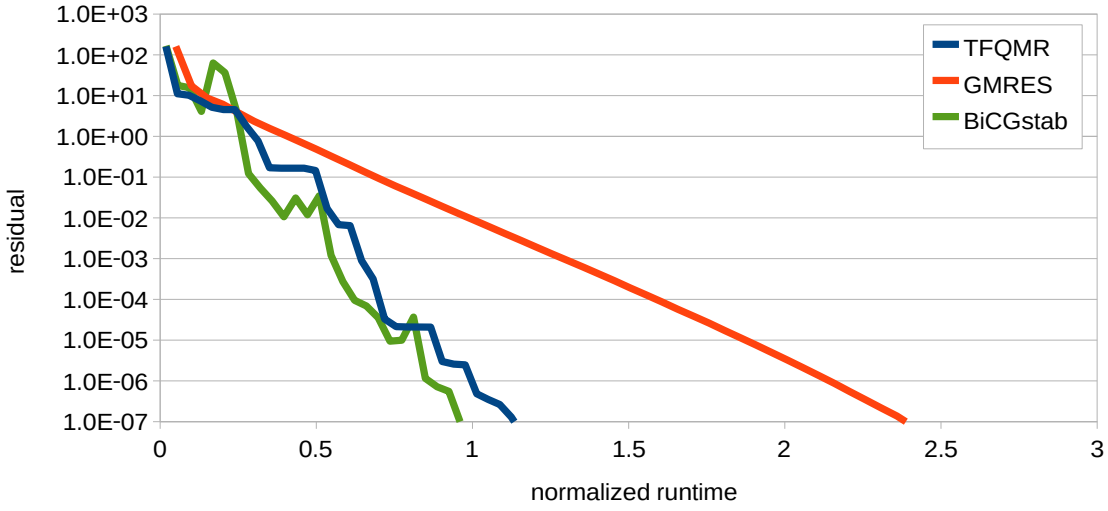


Figure 4.8: Krylov solver residual vs. normalized runtime (normalized by BiCGstab solution time).

4.2.3 Preconditioning

We consider left-preconditioned system,

$$M^{-1}Ax = M^{-1}b,$$

where M^{-1} is the preconditioning matrix. The general rule for preconditioners is that the preconditioned system should be easy to solve and cheap to apply. There are basically two types of preconditioners: algebraic and physics-based. Physics-based preconditioners can be useful for PDEs, where detailed analysis (of perhaps the continuous equations) can reveal an asymptotic equation or "nearby" PDE that is much easier to solve than the original one [85]. The diffusion synthetic acceleration (DSA) discussed in Sec. 1.3.1 can be regarded as such a physics-based preconditioner for the transport equation. Other physics-based preconditioners are multigrid (e.g. see [71]) and preconditioners based on lower order discretizations [85], like transport synthetic acceleration (TSA), also discussed in Sec. 1.3.1. A physics-based diffusion preconditioner may seem like a good idea for the LOQD system, but solving the diffusion problem with E = 1/3I is only about 10% cheaper than LOQD!

The alternative algebraic preconditioners are based simply on the system matrix A, and have the potential to be effective on a wider class of problems, although perhaps less effective than physics-based preconditioners in certain problems. Various algebraic preconditioners are available, for example, those based on incomplete factorizations, approximate inverses, and algebraic multigrid [85]. We have used the code SPARSKIT [74] to investigate incomplete LU factorizations, as described in [91].

4.2.3.1 ILU Preconditioners

We analyze two ILU preconditioners, ILU with dual truncation strategy (ILUT) [91] and ILU with diagonal compensation (ILUD) [91]. The size of the ILU preconditioner M is controlled via user-specified parameters designed to limit *fill-in*.[2] The main disadvantage of large fill-in is increased storage requirements and increased cost per preconditioner matrix multiply.

ILUT has two parameters to controll fill-in: the maximum level of fill, P_{fill}, and drop tolerance, ϵ_{drop}. With ILUT, small entries are dropped according to

$$|M_{ij}| < \epsilon_{drop}||\vec{M}_i||_1, \tag{4.11}$$

where $||\vec{M}_i||_1$ is the 1-norm of the row-vector for row i, $\vec{M}_i$.

ILUD has a single parameter: the drop tolerance, ϵ_{drop}. Instead of truncating small entries according to Eq. (4.11), ILUD adds them to the diagonal.[3]

The advantage of ILUT over ILUD is that the maximum memory used for the preconditioner[4] can be explicitly controlled by P_{fill}, where the definition of P_{fill} here is the *number of additional entries allowed per row*, in M beyond the nonzero entries already in A. With P_{fill} fixed and $\epsilon_{drop} = 0$, ILUT retains the largest additional P_{fill} entries on each row. ILUD cannot guarantee a preconditioner of a certain size.

[2] In general, the LU factorization of a general matrix A will be dense (i.e. have fill-in), with a sparsity structure unlike A. ILU preconditioners have various parameters that limit fill-in.

[3] The general ILUD algorithm adds αM_{ij} to the diagonal, where M_{ij} is a truncated entry. We consider only $\alpha = 1$ which makes this ILUD similar to modified incomplete Cholesky (MILU) preconditioners [91]

[4] Note, with ILU preconditioners the preconditioner is M, not M^{-1}, and the preconditioning stage performs a back solve of $My = Ax$ [85].

4.2.3.2 Results for Fixed ILU Preconditioners

In our first stage of investigating ILU preconditioners, we discovered the essential structure of the LOQD system leads to very large fill-in. An effective ILU preconditioner could not be constructed for the essential structure that did not grow unacceptably large with mesh refinement. This is probably not suprising considering the bandwidth of the essential structure in Fig. 4.5. The essential structure may benefit from other preconditioners, but we do not consider it further here and proceed with only the cell-based structure of Fig. 4.1.

To test preconditioners, we perform a parametrization study with drop tolerance, $\epsilon_{drop} \in \{1, 0.5, 0.1, 0.05, 0.01, 0.001, 0\}$; maximum fill-in, $P_{fill} \in \{1, 5, 10, 20, 40, 80\}$ (for ILUT only); norm used for row-scaling, $p_{row} \in \{-, 1, \infty\}$, where $-$ indicates it is not performed; and norm used for column-scaling, $p_{col} \in \{1, \infty\}$. This leads to 252 ILUT cases and 42 ILUD cases.

To compare ILU preconditioners, we evaluate the ratio of preconditioner construction time to system solve time, t_{prec}/t_{solv}; the condition numbers of the scaled system, $\kappa(\tilde{A})$; the condition numbers of the preconditioned system, $\kappa(M^{-1}\tilde{A})$; the convergence rate [5], R; the ratio of number of nonzeros in preconditioner to original system, $f_{prec} = nnz(M)/nnz(A)$; and the total runtime in ms, t_{tot}.

We then filter out poor combinations based on R, f_{prec}, and $\kappa(\tilde{A})$, specifically $R < 0.10$, $f_{prec} > 2$, $\kappa(M^{-1}\tilde{A}) > 10^5$. We show the fastest remaining options, sorted by total runtime t_{tot}. Data for a simple test with $\sigma_t = 1$, $\sigma_s = 0.9$ is shown in Table 4.1 and Table 4.2 for ILUD and ILUT

[5] The convergence rate R can be used as a measure of efficiency of the iteration process,

$$R = \log\left(\frac{||Ax_0 - b||}{||Ax_n - b||}\right)\frac{1}{N_{matvec} - N_{init}},$$

where x_0 and x_n are the initial and converged solutions, respectively, N_{matvec} is the number of matrix-vector multiplies, and N_{init} is the number of initializations [74].

Table 4.1: ILUD preconditioning/scaling results for Test 1: $\sigma_t = 1$, $\sigma_s = 0.9$.

ϵ_{drop}	p_{row}	p_{col}	$\frac{t_{prec}}{t_{solv}}$	$\kappa(\tilde{A})$	$\kappa(M^{-1}\tilde{A})$	R	f_{prec}	t_{tot}(ms)
0.05	∞	∞	0.13	2.9E+03	8.3E+03	0.22	1.04	34
0.01	∞	∞	0.16	2.9E+03	2.3E+03	0.23	1.32	36
0.01	∞	1	0.16	3.8E+03	7.9E+03	0.23	1.34	36
0.01	–	∞	0.16	7.5E+03	2.3E+03	0.24	1.37	36
0.05	–	∞	0.12	7.5E+03	8.4E+03	0.23	1.08	37
0.05	–	1	0.12	7.5E+03	8.4E+03	0.23	1.08	37
0.05	–	∞	0.11	3.6E+03	2.6E+04	0.21	1.10	39
0.05	–	1	0.11	3.6E+03	2.6E+04	0.21	1.10	39

Table 4.2: ILUT preconditioning/scaling results for Test 1: $\sigma_t = 1$, $\sigma_s = 0.9$.

P_{fill}	ϵ_{drop}	p_{row}	p_{col}	$\frac{t_{prec}}{t_{solv}}$	$\kappa(\tilde{A})$	$\kappa(M^{-1}\tilde{A})$	R	f_{prec}	t_{tot}
40	0.05	–	∞	0.19	7.5E+03	7.3E+03	0.28	1.13	32
80	0.05	∞	∞	0.18	2.9E+03	8.5E+03	0.27	1.08	33
40	0.05	∞	1	0.21	2.9E+03	8.5E+03	0.27	1.08	34
40	0.05	–	∞	0.21	7.5E+03	7.3E+03	0.28	1.13	35
40	0.05	–	1	0.21	3.6E+03	5.3E+03	0.31	1.13	35
80	0.05	–	∞	0.21	3.6E+03	5.3E+03	0.31	1.13	35

preconditioning, respectively. Data for a simple test with $\sigma_t = 10$, $\sigma_s = 0$ is shown in Table 4.3 and Table 4.4 for ILUD and ILUT preconditioning, respectively. Both tests are on 30% randomized 16×16 meshes. ILUD shows less variability with respect to norms choices for scaling and shows slightly better effectiveness with the ∞-norm. ILUT can be faster and use less memory, although not substantially so.

Table 4.3: ILUD preconditioning/scaling results for Test 2: $\sigma_t = 10$, $\sigma_s = 0$.

ϵ_{drop}	p_{row}	p_{col}	$\frac{t_{prec}}{t_{solv}}$	$\kappa(\tilde{A})$	$\kappa(M^{-1}\tilde{A})$	R	f_{prec}	t_{tot}(ms)
0.01	–	∞	0.57	1.6E+02	3.1E+00	1.16	0.95	11
0.05	1	∞	0.37	2.2E+01	9.7E+00	0.86	0.85	11
0.05	1	1	0.37	1.9E+01	1.3E+01	0.88	0.84	11
0.05	∞	1	0.37	3.1E+01	6.9E+00	0.86	0.84	11
0.01	–	1	0.57	1.6E+02	3.1E+00	1.16	0.95	11

Table 4.4: ILUT preconditioning/scaling results for Test 2: $\sigma_t = 10$, $\sigma_s = 0$.

P_{fill}	ϵ_{drop}	p_{row}	p_{col}	$\frac{t_{prec}}{t_{solv}}$	$\kappa(\tilde{A})$	$\kappa(M^{-1}\tilde{A})$	R	f_{prec}	t_{tot}
40	0.01	1	1	0.80	6.6E+01	1.4E+00	0.94	0.51	9
10	0.10	∞	∞	0.57	6.6E+01	7.1E+00	0.99	0.71	11
10	0.05	∞	1	0.57	3.1E+01	4.2E+00	1.25	0.85	11
5	0.10	∞	∞	0.37	6.6E+01	8.9E+00	0.86	0.70	11
20	0.50	∞	∞	0.22	3.1E+01	1.7E+01	3.08	0.42	11
40	0.50	1	1	0.22	6.6E+01	3.6E+01	0.42	0.28	11
20	0.50	1	1	0.22	6.6E+01	3.6E+01	0.42	0.86	11

4.2.3.3 Adaptive Recalculation Strategy for ILU Preconditioners

Whether ILUT or ILUD is better (and the optimal choice of parameters) is clearly problem-dependent. In practice, any ILU preconditioner can fail, so we use a simple process to adaptively recompute the preconditioner if the Krylov iterations diverge or converge slowly [85]. The adaptive strategy is simple. First, we perform N_{test} test iterations. If the convergence rate, R, shows slow convergence, $R < R_{min}$, the ILUD (ILUT) preconditioner is recomputed with decreased ϵ_{drop} (and increased P_{fill}). In practice, we use $N_{test} = 10$ test iterations, minimum convergence rate $R_{min} = 0.1$, and decrease ϵ_{drop} by a factor of 2 and increase P_{fill} by a factor of 5.

ILUT's advantage over ILUD is that it can limit the size of the preconditioner, thus we would prefer to use it, but we have found it sensitive to P_{fill}. If a proper P_{fill} can be specified, usually from experience with a particular problem, ILUT is faster than ILUD and uses less memory. If a proper P_{fill} cannot be specified, the ILUT preconditioner will be recalculated many times until a sufficient P_{fill} is reached. The two-parameter refinement with ILUT also leads to complications because we always refine ϵ_{drop} and P_{fill} simultaneously, so we can get unnecessarily low drop tolerances.

For this reason, we find ILUD a better option. With a scaled system, $\epsilon_{drop} = 0.01$

is sufficient to solve many problems. If not, the adaptive strategy usually finds 0.005 or 0.001 acceptable, which typically leads to modest memory requirements of $f_{prec} \approx 1.5$.

4.3 QD Solver Runtimes

The LOQD and transport solver times[6] are of course problem-dependent, but mostly because it is unknown a priori how many transport iterations will be required for any given problem. In terms of a single transport iteration, the amount of time to solve the high-order and LOQD equations depends mostly on problem size, i.e. the number of unknowns. For most problems, we see runtimes per transport iteration versus mesh refinement similar to Fig. 4.9, where the High-order QD solver is by transport sweep with SCSB discretization and the LOQD solver is BiCGstab with ILUD preconditioning of the JM discretization with cell-based structure. The iterative LOQD solution shows at least one and usually two orders of magnitude faster run times than sparse Gaussian elimination (LOQD Direct in the figure). We will remark on the features in Fig. 4.9 in the next section.

4.3.1 Transport Solver

As seen in Fig. 4.9, SCSB scales linearly with the number of cells N_C for most refinements, as is expected with the transport sweep. With SCSB, there is a jump after $N_C \approx 1000$ in Fig. 4.9. This jump in computational cost happens for cells where $\sigma_t h \approx 1$ because of the on-the-fly computation of the polynomial exponential moments. When the optical thickness is near unity, hundreds of expansion terms may be required and the cost per cell increases. The cost per cell is relatively

[6]The computer for all timing tests has a 2.33 GHz Intel Xeon CPU with 4MB cache and 8 GB RAM running Red Hat Linux. All code was written in Fortran and compiled with the Intel Fortran Compiler 9.1 with standard (O2) optimizations.

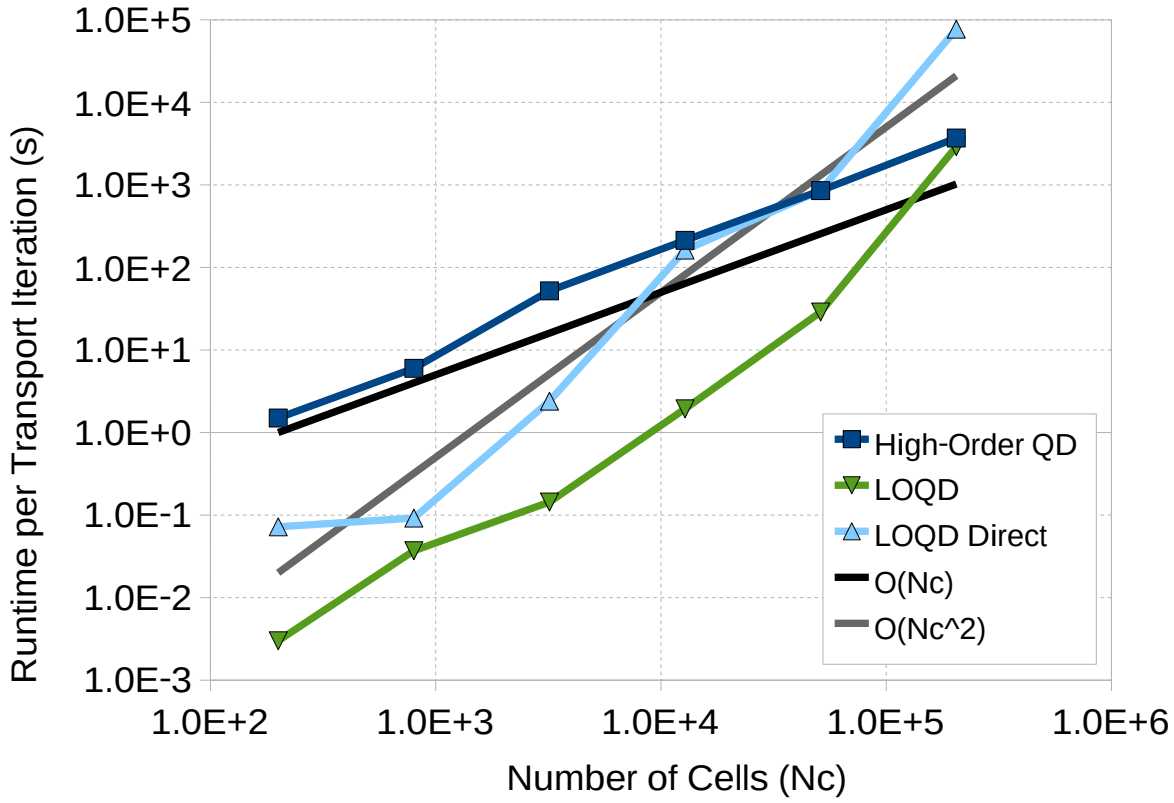


Figure 4.9: Representative solver runtimes.

constant for $\sigma_t h > 5$ and $\sigma_t h < 0.1$ and typically less than 10 terms are needed in the expansions.

A conservative estimate of the current SCSB implementation's unit cost is 20ms per cell, per direction—SCV is only 50μs per cell-direction. The breakdown in time spent per cell with SCSB is about: 50% for on-the-fly decomposition of cells into subcells, 45% for on-the-fly polynomial exponential moment calculation, and 5% for everything else.

Simply tabulating data for the polynomial exponential moments (which are well-behaved functions of two parameters) could decrease the evaluation cost by up to a factor of 5, thus a simply-optimized SCSB could have a unit cost of 13ms per cell-direction.

Preliminary estimates also show that the cell decompositions could be reduced by a factor of 2 if an explicit mesh structure is used[7] or if explicit representations of each cell are stored. The cell decompositions could also be cached as part of a setup phase, but this is a very large extra memory cost, quadrilaterals typically have 3 subcells in the decomposition for each direction for a total of $3N_CN_M$ subcells—memory requirements equal to the angular flux which would approximately double the total memory cost of simulation. We will discuss memory requirements in more detail shortly.

4.3.2 LOQD Solver

The BiCGstab Krylov iterative solver with ILU preconditioning has been shown an effective strategy to solve the LOQD system with JM discretization and cell-based system structure. In particular, the ILUD preconditioner with $\epsilon_{drop} = 0.01$ and adaptive recalculation strategy described in Sec. 4.2.3.3 leads to a robust algorithm we have used without failure for all tests presented in this work. Comparing our iterative solution to sparse Gaussian elimination with partial pivoting[8], we see faster runtimes by orders of magnitude in most tests, as seen in Fig. 4.9.

The LOQD solver shows quadratic scaling with number of cells up to about 8000 cells, as seen in Fig. 4.9, but for finer meshes, large memory management operations (e.g. allocating space for A and M) leads to a jump in cost, because we explicitly store A and M. Depending on the

[7]The arbitrary mesh implementation is *implicit* which means that only the locations of vertices are stored—the faces are references to vertices and cells are references to faces. The implicit mesh makes mesh construction and mesh alteration easier as all operations are performed on vertices, with recalculations of affected cells. Properties of cells and faces that are needed often (e.g. centers) may be cached for a runtime/memory tradeoff. However, in order to decompose cells as required by our SCSB discretization, we form the explicit representation of a cell: a list of counter-clockwise ordered (x, y) values.

[8]We use the highly-regarded, if not somewhat old, code NSPIV [17]. More current and advanced direct codes, such as SUPERLU [101], gain speedup over NSPIV algorithms by grouping columns which have identical nonzero structure into "super-nodes". Matrix operations are then performed on supernodes for substantial savings. It is unknown whether our cell-based structure can take advantage of supernodes or not.

problem, we have seen this behavior return to quadratic scaling or continue with super-quadratic scaling until memory is exhausted.

4.3.3 Storage Requirements

The limiting storage requirement is for the angular flux, proportional to the number of cells *times number of directions*, $O(N_C N_M)$. For a $N \times N$ mesh and SCSB, the constant is 4, or $4N_C N_M$ angular flux unknowns. Considering even a crude quadrature set like S_8 level symmetric, which has 40 directions in 2D, the storage cost of transport is about $160N_C$. We also use the S_{12} and S_{16} level symmetric quadrature sets, which have 84 and 144 directions, respectively.

However, with QD, the angular fluxes ψ^c, ψ^f, ψ^v do not need to be explicitly stored—they can be accumulated directly into LOQD factors E^c, E^f, E^v and C^f at the end of each direction's transport sweep. This could result in huge savings, from $160N_C$ to $8N_C$, for factor storage plus one direction's storage. For research purposes, it is useful to be able to analyze the angular fluxes, so we store them in our implementation. But in a production code, this should not be the case.

The LOQD system has order $9N_C$, with an average of $37/9 \approx 4$ nonzeros per row, for a storage cost of $36N_C$ for the A matrix, with $9N_C$ for x and b. Preconditioning with ILUD can be accomplished with no more than twice the number of nonzeros in M as A for a total for the LOQD system of $(36 + 9 + 9 + 72)N_C = 126N_C$. With a matrix free implementation, the storage cost for A can be eliminated, for a $90N_C$ storage cost for the LOQD system.

Eliminating the first-moment LOQD equations by substituting J^f into the balance equation is not a good strategy for reducing the storage requirements because it only leads to marginally fewer nonzero entries in A, so the storage cost of A is similar, but storage cost of ILU preconditioner

M may be greatly increased due to the loss in block structure. The decreased storage cost for x and b only reduces the total from $126N_C$ to $118N_C$. There is also additional temporary storage associated with the Krylov method that is reduced with the balance equation only system.

4.4 Summary

In this chapter we presented a robust and fast iterative solution strategy for the LOQD system of Chapter 2 with JM discretization consisting of a cell-based structuring of unknowns, row and column scaling of the resultant system, and ILUD preconditioning with adaptive recalculation strategy. Conditioning and timing results show the effectiveness of the approach, although it does result in large memory requirements. This is a standard problem with transport simulations and we have recommended some options to reduce this cost. We have had success with ILUD preconditioning for these general systems but investigations should continue into other preconditioners, such as algebraic multigrid or approximate inverse preconditioning. For example, an approximate inverse could be constructed by inverting each of the 9×9 blocks in the cell-based structuring of the LOQD system.

Chapter 5

NUMERICAL RESULTS

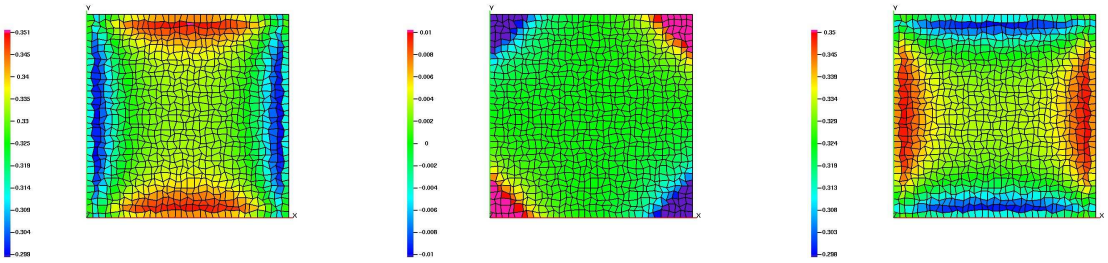
In this chapter, we present numerical tests of our new LOQD discretization described in Chapter 2, combined with the transport methods described in Chapter 3.

5.1 Uniform External Source Test

The domain is a 8cm $\times$ 8cm square with vacuum boundary conditions, $\psi_{IN} = 0$ with homogeneous material properties and source, $\sigma_t = 1\text{cm}^{-1}$, scattering $\sigma_s = 0\text{cm}^{-1}$, and $q_{ext} = 1\text{cm}^{-1}$. We examine the behavior of QD with SCSB and SCV transport discretizations and the JM LOQD discretization. Because there is no scattering, this test converges in one iteration. The S_8 level symmetric quadrature set is used.

5.1.1 Data

For this problem, SCSB and ESC on a 1 cell mesh reproduce the exact domain-average scalar flux, $\phi^D_{exact} = 0.87985528755226$. A numerical convergence analysis on the domain-average



(a) E_{xx}. (b) E_{xy} (c) E_{yy}

Figure 5.1: Uniform External Source Test: QD factors.

scalar flux ϕ^D is performed on sequences of $N \times N$ on orthogonal and randomized logically rectangular meshes. Randomized meshes have 30% perturbations of vertices. Sample factors are shown in Fig. 5.1 globally, and near the bottom left corner in Fig. 5.4. The reduction in error with mesh refinement is shown in Fig. 5.2 and Fig. 5.3 for orthogonal and randomized meshes, respectively. The numerical convergence orders corresponding to this data are tabulated in Table 5.1 and Table 5.2. The LOQD solution calculated with factors generated from SCSB and SCV is denoted LOQD[SCSB] and LOQD[SCV], respectively. The solutions from SCSB and ESC are offered for comparison. (We do not show domain-average scalar fluxes for SCV because it has only vertex-based unknowns, relying on the LOQD solution for cell-average scalar fluxes.)

Table 5.1: Uniform External Source Test: Convergence orders for ϕ^D on orthogonal meshes.

h(cm)	LOQD[SCV]	ESC	LOQD[SCSB]	SCSB
1	1.45	0.16	1.42	0.80
1/2	1.94	0.64	1.83	1.61
1/4	2.15	0.61	1.99	2.45
1/8	2.15	0.98	2.02	2.73
1/16	2.13	0.98	2.01	2.87

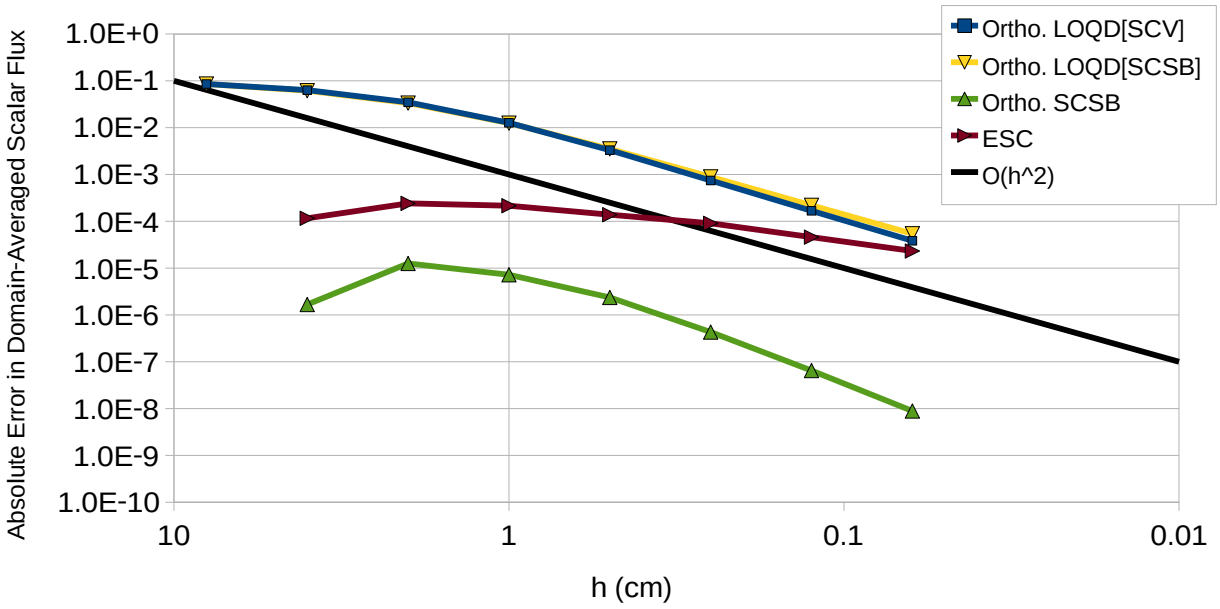


Figure 5.2: Uniform External Source Test: Absolute error in ϕ^D on orthogonal meshes.

Table 5.2: Uniform External Source Test: Convergence orders for ϕ^D on 30% randomized meshes.

h(cm)	LOQD[SCV]	ESC	LOQD[SCSB]	SCSB
1	1.27	0.16	1.23	0.51
1/2	1.80	0.64	1.69	1.86
1/4	2.18	0.61	1.85	2.13
1/8	2.70	0.98	1.93	2.61
1/16	2.59	0.98	1.59	3.08

5.1.2 Analysis

The expected $O(h)$ convergence is observed for ESC, which is implemented as SCSB with flat incoming angular flux (instead of parabolic interpolation with monotonization via the BKM method.) The SCSB transport method exhibits $O(h^3)$ convergence because in this test, near field effects are much more important than far-field ones and the reduction by one order due to

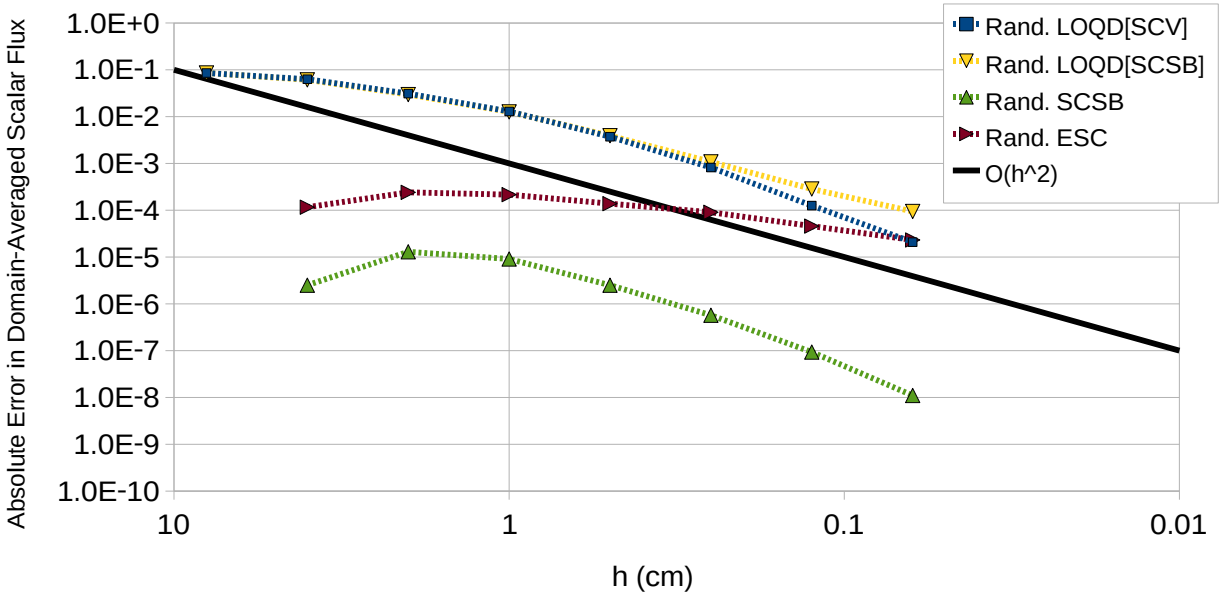
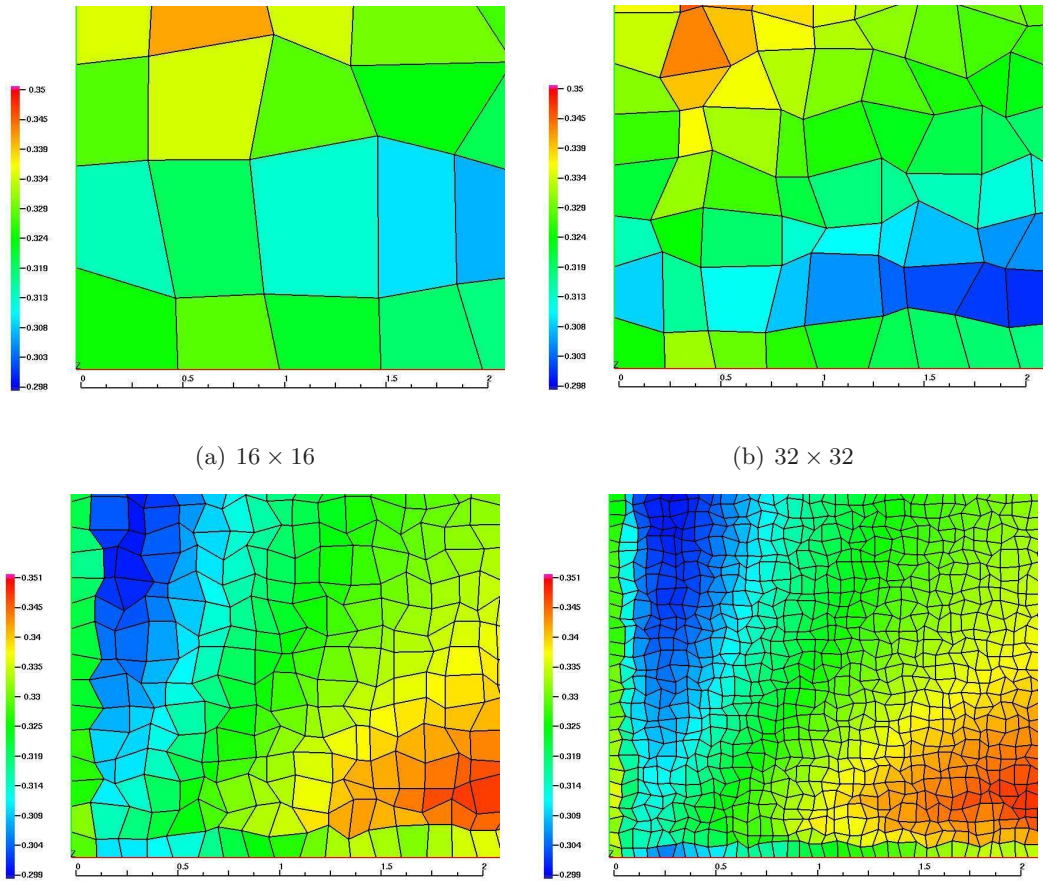


Figure 5.3: Uniform External Source Test: Absolute error in ϕ^D on 30% randomized meshes.

accumulated discretization error from $O(1/h)$ upwind cells is not yet felt. The SCSB discretization is very accurate for this type of problem exhibiting 10^{-8} absolute error on the most refined mesh. The ESC error is about 10^{-5}. The LOQD[SCSB] and LOQD[SCV] errors are also 10^{-5}, although it is clear that any refinement will lead to less error than ESC.

On orthogonal meshes the LOQD discretizations exhibit $O(h^2)$ convergence. On randomized meshes, LOQD[SCSB] shows convergence orders of 1.6 on the finest mesh, LOQD[SCV] shows 2.6. The degraded convergence with LOQD[SCSB] may be due to appearing boundary layers in LOQD factors, E_{xx}, E_{yy}, and E_{xy}. Representative boundary layer behavior in E_{xx} near the bottom left corner of the domain is shown in Fig. 5.4. With highly anisotropic angular fluxes (as exist at vacuum boundaries and strong material interfaces), the QD factors exhibit large spatial gradients. The LOQD discretization must resolve tensor divergence terms $\vec{\nabla} \cdot (\phi \mathrm{E}) = \mathrm{E}\vec{\nabla}\phi + \phi\vec{\nabla} \cdot \mathrm{E}$. The

first $\mathrm{E}\vec{\nabla}\phi$ term should be easy to resolve in this test but it is easy to imagine that approximating the second term $\phi\vec{\nabla}\cdot\mathrm{E}$ on unstructured meshes is much more difficult. In support of this theory, if a layer of cells with thickness 1/64cm is placed along the boundary and not perturbed, even though the interior of the mesh is randomized, LOQD[SCSB] shows convergence order of 1.89 and LOQD[SCV] shows 2.29, much closer to the expected values of 2.



(a) 16×16 (b) 32×32

(c) 64×64 (d) 128×128

Figure 5.4: Uniform Source Test: E_{xx} boundary behavior near corners with SCSB.

5.2 Diffusion Limit Test

The domain is a 1cm $\times$ 1cm square with vacuum boundary conditions, $\psi_{IN} = 0$. Diffusion limit conditions are tested with data $\sigma_t = 1/\epsilon$, $\sigma_a = \epsilon$, $\sigma_s = \sigma_t - \sigma_a$, $q_{ext} = \epsilon$, where the smallness parameter $\epsilon \to 0$. The scattering ratio is thus $c = 1 - \epsilon^2$. A 20×20 mesh is tested for $\epsilon = 0.1$ to 0.00001. Note that the case of smallness parameter $\epsilon = 0.1$ does not lead to a diffusion problem, but we consider it to see better how the transport solution approaches diffusion. We converge the scalar flux to a relative tolerance of 10^{-10}.

5.2.1 Data

The domain-average scalar fluxes ϕ^D are shown vs. smallness parameter ϵ and scattering ratio c in Table 5.3 for the reference diffusion, Table 5.4 for the high-order QD (with SCSB discretization), and Table 5.5 for the low-order QD (with JM discretization). The reference diffusion solution is provided by the GGK/JM LOQD discretization on orthogonal meshes under the diffusive conditions $E_{\alpha\beta} = 1/3\delta_{\alpha\beta}$ and $C = 1/2$. The scalar flux along a line from the bottom left corner of the domain to the top right is shown in Fig. 5.5. The number of transport iterations required is shown in Table 5.6.

Table 5.3: Diffusion Limit Test: ϕ^D of the reference diffusion discretization.

		Mesh Randomization			
ϵ	c	0%	10%	20%	30%
0.1	0.99	1.355464E-01	1.355627E-01	1.356090E-01	1.356847E-01
0.01	0.9999	9.755510E-02	9.757647E-02	9.763759E-02	9.773739E-02
0.001	0.999999	9.347169E-02	9.349383E-02	9.355717E-02	9.366046E-02
0.0001	0.99999999	9.305997E-02	9.308219E-02	9.314576E-02	9.324941E-02
0.00001	0.9999999999	9.301875E-02	9.304098E-02	9.310458E-02	9.320827E-02

Table 5.4: Diffusion Limit Test: ϕ^D of the high-order QD (SCSB) discretization.

		Mesh Randomization			
ϵ	c	0%	10%	20%	30%
0.1	0.99	1.399734E-01	1.399809E-01	1.400047E-01	1.400452E-01
0.01	0.9999	9.776839E-02	9.778728E-02	9.783870E-02	9.791591E-02
0.001	0.999999	9.347379E-02	9.349087E-02	9.353300E-02	9.359563E-02
0.0001	0.99999999	9.305917E-02	9.307628E-02	9.311777E-02	9.317911E-02
0.00001	0.9999999999	9.301866E-02	9.303579E-02	9.307732E-02	9.313872E-02

Table 5.5: Diffusion Limit Test: ϕ^D of the LOQD (JM) discretization.

		Mesh Randomization			
ϵ	c	0%	10%	20%	30%
0.1	0.99	1.400002E-01	1.400083E-01	1.400341E-01	1.400781E-01
0.01	0.9999	9.777771E-02	9.779652E-02	9.784757E-02	9.792422E-02
0.001	0.999999	9.347535E-02	9.349241E-02	9.353449E-02	9.359704E-02
0.0001	0.99999999	9.305934E-02	9.307644E-02	9.311792E-02	9.317925E-02
0.00001	0.9999999999	9.301868E-02	9.303581E-02	9.307734E-02	9.313873E-02

Table 5.6: Diffusion Limit Test: Number of QD iterations.

		Mesh Randomization			
ϵ	c	0%	10%	20%	30%
0.1	0.99	10	10	11	10
0.01	0.9999	8	8	8	8
0.001	0.999999	5	6	8	10
0.0001	0.99999999	4	6	8	10
0.00001	0.9999999999	3	6	8	10

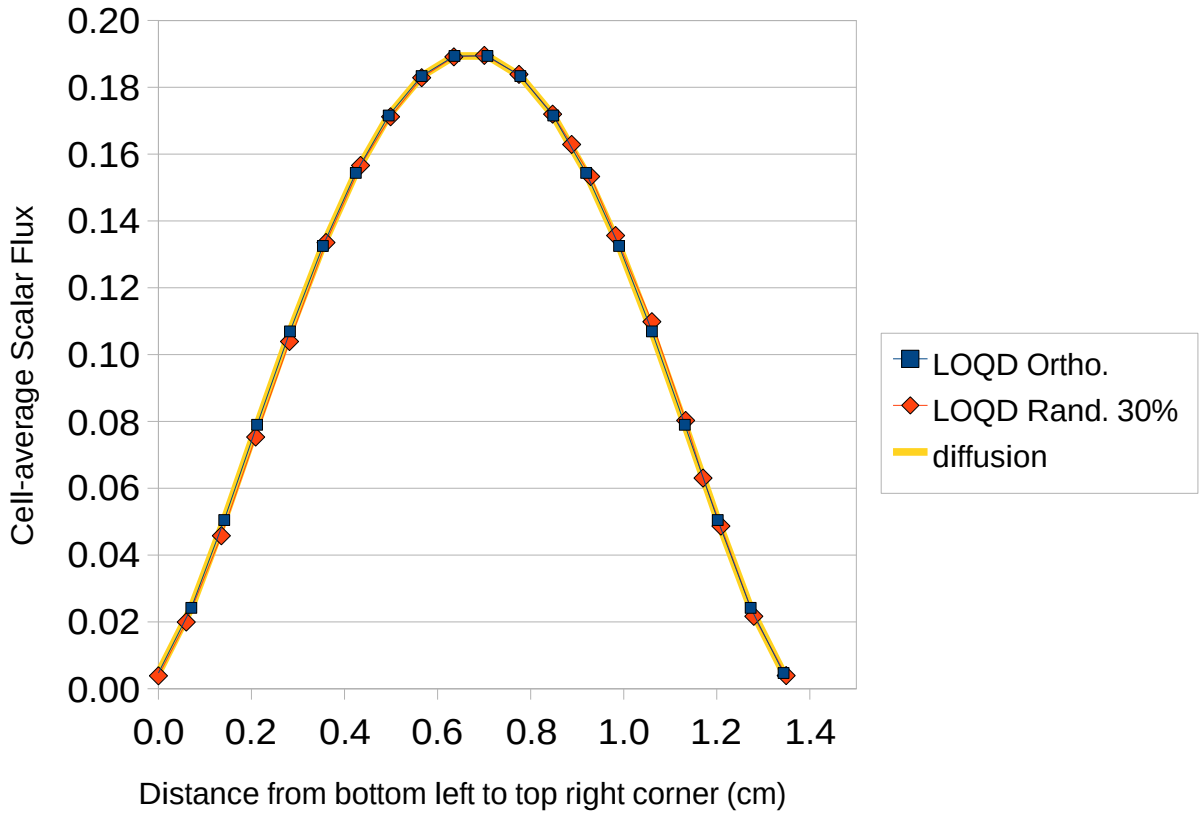


Figure 5.5: Diffusion Limit Test: Comparison of LOQD and Diffusion solution for $\epsilon = 0.0001$.

5.2.2 Analysis

For orthogonal 20×20 meshes, as $\epsilon \to 0$, QD shows a decrease in difference from the diffusion solution. For randomized meshes, however, QD shows 0.01% relative difference from the orthogonal. However, even for the reference diffusion solution, as $\epsilon \to 0$, we see a similar difference of approximately 0.01%. To investigate this effect, we performed 20 calculations in order to estimate the standard deviation of the mean domain-average scalar fluxes and relative standard deviations, $\bar{\phi}^D$ and $\sigma^D/\bar{\phi}^D$, for a 20×20 mesh and 40×40 mesh. The results, shown in Table 5.7 for SCSB

show small standard deviations and a decrease in standard deviation with mesh refinement for all c.

Table 5.7: Diffusion Limit Test: Randomization error for the high-order QD (SCSB) discretization on 30% randomized meshes.

		$\phi^D \pm \sigma^D / \phi^D$	
ϵ	c	20×20 mesh	40×40 mesh
0.1	0.99	1.3935E-01 ± 3.8E-04	1.3521E-01 ± 5.7E-05
0.01	0.9999	9.7486E-02 ± 4.0E-04	9.6912E-02 ± 9.8E-05
0.001	0.999999	9.3519E-02 ± 5.6E-04	9.2856E-02 ± 1.2E-04
0.0001	0.99999999	9.3120E-02 ± 6.5E-04	9.2446E-02 ± 1.2E-04
0.00001	0.9999999999	9.3105E-02 ± 6.5E-04	9.2407E-02 ± 1.2E-04

5.3 Analytic Transport Test, Revisited

We revisit the test in Sec. 3.6.2, but now we add the LOQD problem. The anisotropic external source from manufactured solutions, Eq. (3.22), requires that the first moment LOQD equations, Eq. (2.2b) have a source term on the RHS,

$$q_{ext,1}^{f} = \vec{q}_{ext,1}^{f} \cdot \vec{n}^{f},$$

where $\vec{n}^{f}$ is the outward normal of face f and $\vec{q}_{ext,1}^{f}$ is the first angular moment of the external source averaged over face f,

$$q_{ext,1}^{f} = \frac{1}{A^{f}} \int_{f} dA \; \vec{q}_{ext,1}(x,y),$$

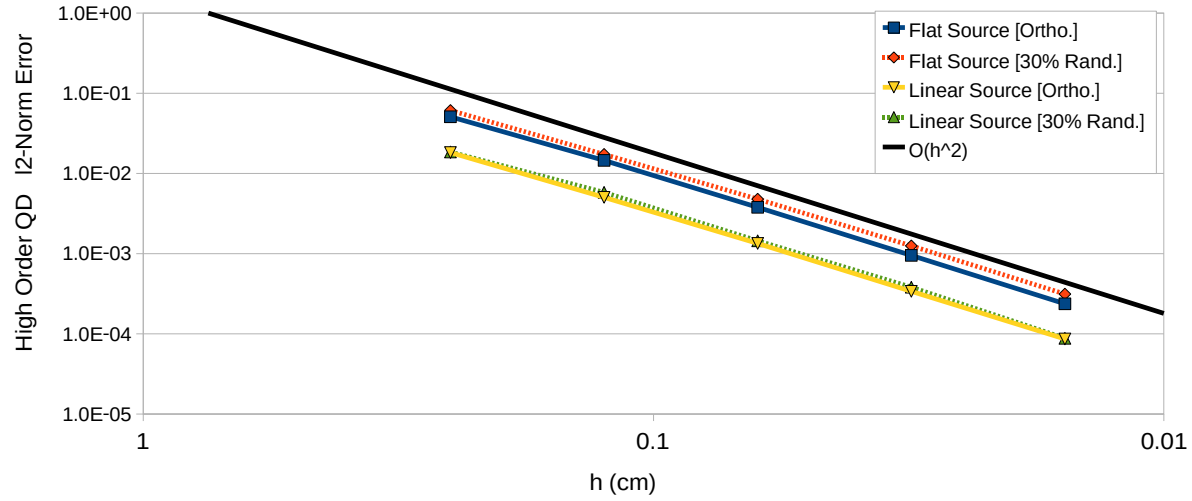
where $\vec{q}_{ext,1}$ is the first angular moment of the source, defined as

$$\vec{q}_{ext,1}(x,y) = \int_{4\pi} d\Omega \; \vec{\Omega} \; q_{ext}(x,y,\Omega_x,\Omega_y).$$

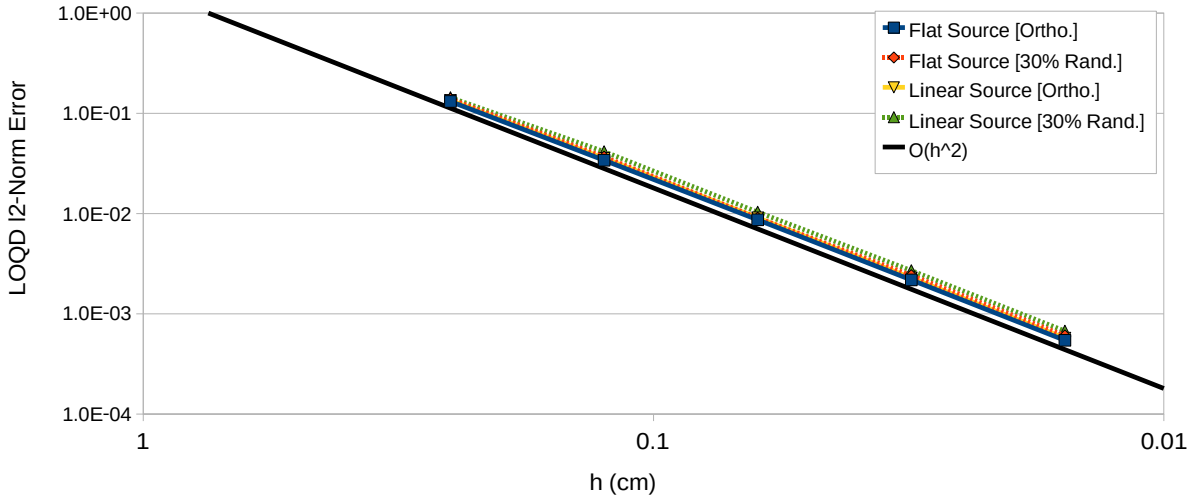
We use sequences of $N \times N$ logically rectangular meshes and the S_{12} level symmetric quadrature set.

5.3.1 Data

High-precision numerical integration is used to approximate the cell-average and face-average external source terms. The QD factors are not calculated from Eq. (3.20) but from the SCSB method. The LOQD equations have the JM FV discretization. Fig. 5.6(a) and Fig. 5.6(b) show the ℓ_2 error norms of the high-order QD and LOQD cell-average scalar fluxes, respectively. We show orthogonal and 30% randomized meshes with both flat and linear scattering source representation from Eq. (3.18) and Eq. (3.19) respectively. Data is tabulated in Table 5.8, Table 5.9 Table 5.10, and Table 5.11. .



(a) high-order QD



(b) LOQD

Figure 5.6: Analytic Transport Test, Revisited: ℓ_2 norm error for (a) the high-order QD and (b) LOQD scalar flux.

Table 5.8: Analytic Transport Test, Revisited: ℓ_2 error norm for high-order QD scalar flux with flat scattering source representations.

	Ortho.		Rand. 10%		Rand. 20%		Rand. 30%	
$1/h$	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*
8	1.63E-02	3.37	1.66E-02	3.42	1.73E-02	3.44	1.89E-02	3.41
16	4.25E-03	3.83	4.31E-03	3.85	4.57E-03	3.78	5.20E-03	3.63
32	1.07E-03	3.98	1.10E-03	3.93	1.18E-03	3.87	1.35E-03	3.84
64	2.66E-04	4.02	2.72E-04	4.03	2.92E-04	4.04	3.40E-04	3.98

Table 5.9: Analytic Transport Test, Revisited: ℓ_2 error norm for high-order QD scalar flux with linear scattering source representation.

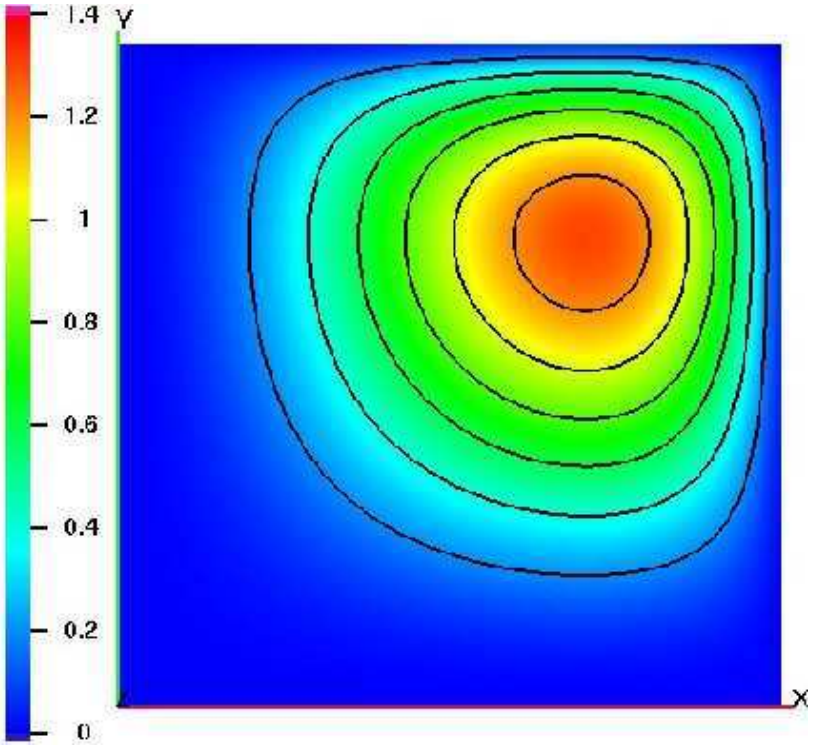
	Ortho.		Rand. 10%		Rand. 20%		Rand. 30%	
$1/h$	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*
8	5.04E-03	3.62	5.19E-03	3.55	5.46E-03	3.39	5.81E-03	3.19
16	1.34E-03	3.76	1.35E-03	3.85	1.38E-03	3.95	1.45E-03	4.02
32	3.39E-04	3.95	3.42E-04	3.94	3.54E-04	3.90	3.82E-04	3.79
64	8.63E-05	3.93	8.65E-05	3.96	8.76E-05	4.04	8.80E-05	4.34

Table 5.10: Analytic Transport Test, Revisited: ℓ_2 error norm for LOQD scalar flux with flat scattering source representation.

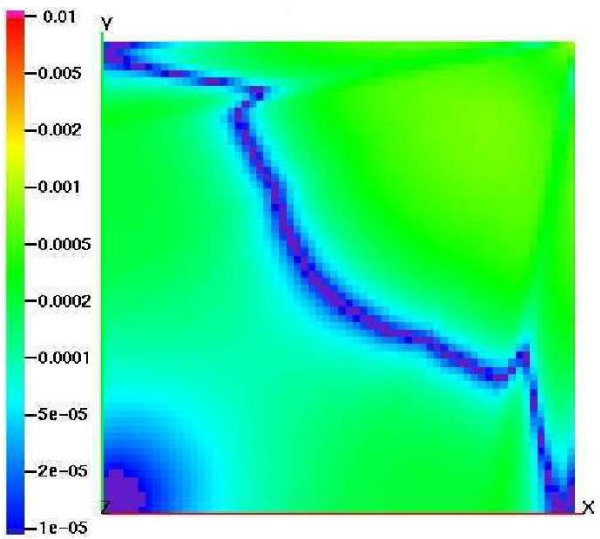
	Ortho.		Rand. 10%		Rand. 20%		Rand. 30%	
$1/h$	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*
8	3.33E-02	3.87	3.41E-02	3.70	3.44E-02	3.80	3.58E-02	3.85
16	8.50E-03	3.92	8.60E-03	3.97	8.60E-03	4.00	8.83E-03	4.05
32	2.13E-03	3.98	2.17E-03	3.96	2.24E-03	3.84	2.35E-03	3.75
64	5.34E-04	4.00	5.48E-04	3.96	5.65E-04	3.96	5.90E-04	3.99

Table 5.11: Analytic Transport Test, Revisited: ℓ_2 error norm for LOQD scalar flux with linear scattering source representation.

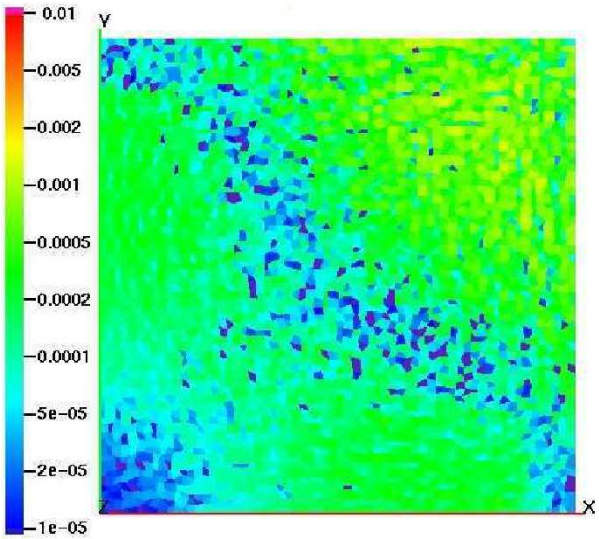
	Ortho.		Rand. 10%		Rand. 20%		Rand. 30%	
$1/h$	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*	ℓ_2 err.	*Ratio*
8	3.56E-02	3.78	3.66E-02	3.63	3.86E-02	3.52	4.15E-02	3.46
16	9.08E-03	3.92	9.21E-03	3.98	9.65E-03	4.01	1.04E-02	4.01
32	2.28E-03	3.98	2.33E-03	3.96	2.46E-03	3.91	2.70E-03	3.83
64	5.71E-04	3.99	5.84E-04	3.99	6.20E-04	3.98	6.73E-04	4.01



(a) The analytic solution $\phi_{exact}(x, y)$.

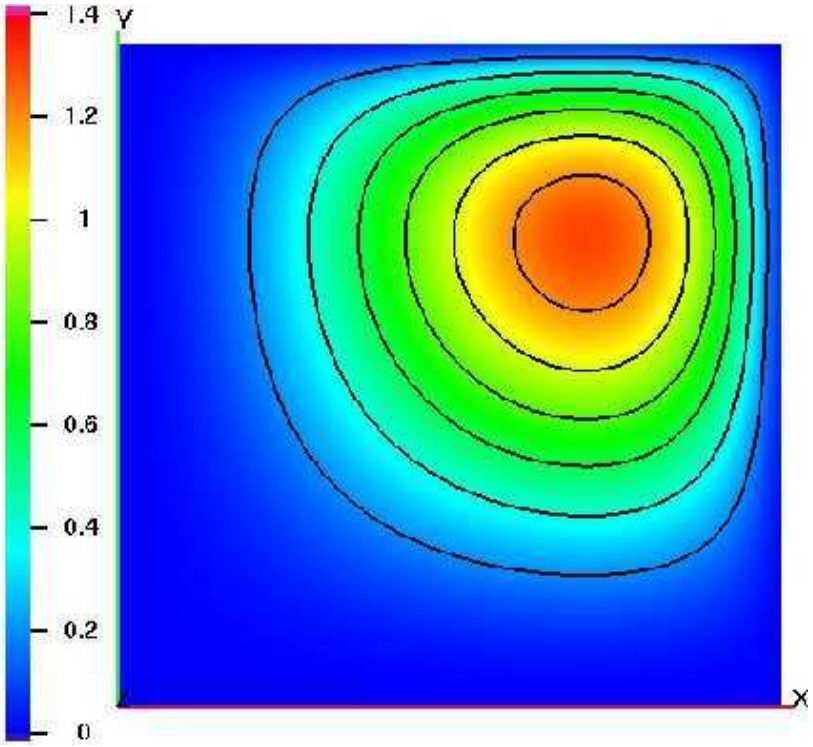


(b) orthogonal

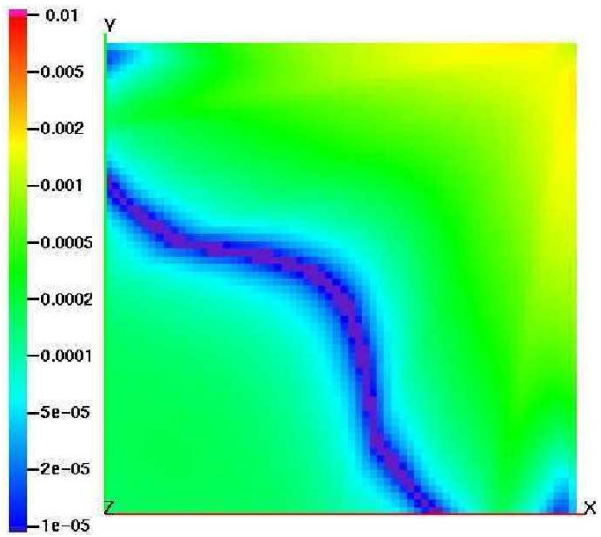


(c) randomized

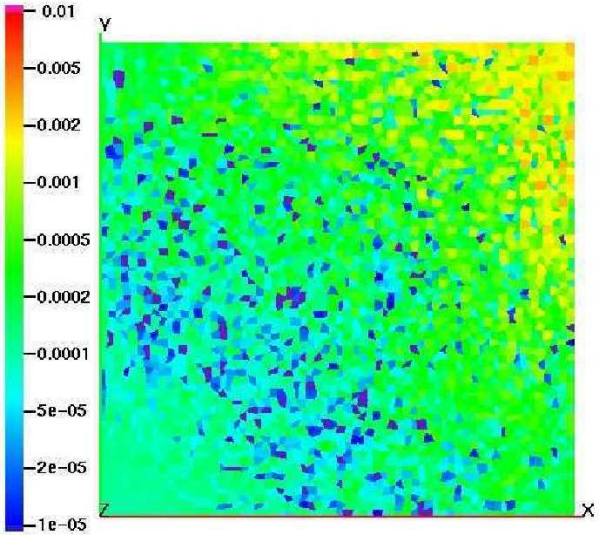
Figure 5.7: Transport Analytic Test, Revisited: (a) analytic solution and absolute error in cell-averages $|\phi^c - \phi^c_{exact}|$ for the high-order transport solution (via SCSB) on a 64×64 orthogonal (b) and randomized (c) mesh.



(a) The analytic solution $\phi_{exact}(x, y)$.



(b) orthogonal



(c) randomized

Figure 5.8: Transport Analytic Test, Revisited: (a) analytic solution and absolute error in cell-averages $|\phi^c - \phi^c_{exact}|$ for the LOQD solution (via JM) on a 64×64 orthogonal (b) and randomized (c) mesh.

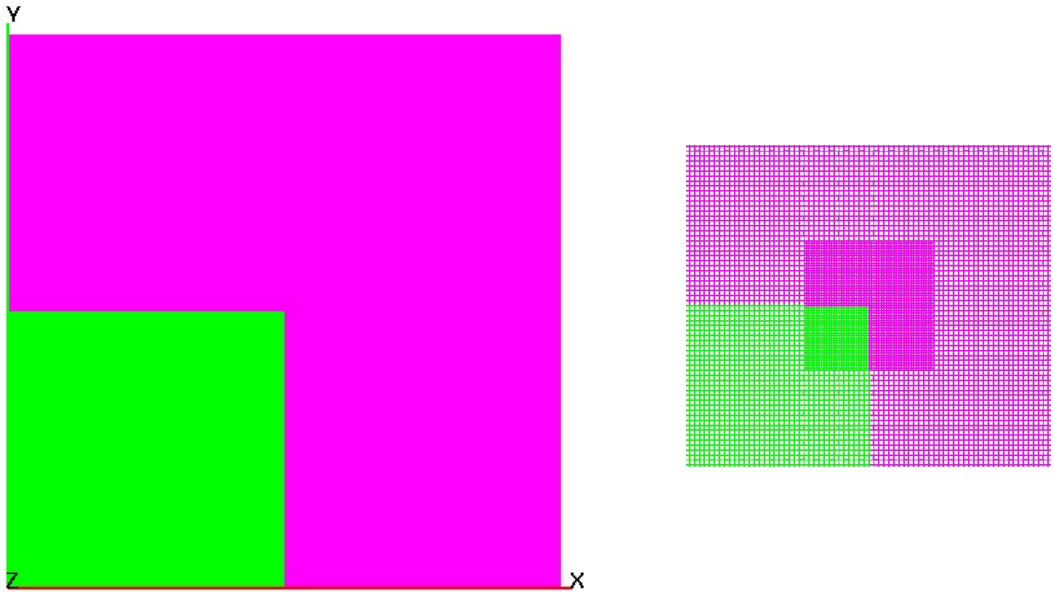
5.3.2 Analysis

The ℓ_2 error norms of the high-order QD and LOQD scalar fluxes exhibit second-order convergence ($Ratio \approx 4$) in Fig. 5.6(a) and Fig. 5.6(b), respectively and in the corresponding tables. With flat scattering source, the high-order QD ℓ_2 error is slightly lower than the LOQD ℓ_2 error. With the linear scattering source, the high-order QD ℓ_2 error is a factor of 5 times lower than with a flat scattering source. The LOQD error shows negligible difference between the two scattering source representations. Also, with the linear scattering source, the differences between results on orthogonal and randomized meshes is greatly reduced.

The anisotropic manufactured sources of this analytic test also lead to a nonzero RHS in the LOQD first-moment equations, which may be negative in many cells. The fact that this posed no problem for the JM discretization leads us to believe the discretization can handle multigroup anisotropic scattering problems as well.

5.4 Discontinuous Media/Source Problem

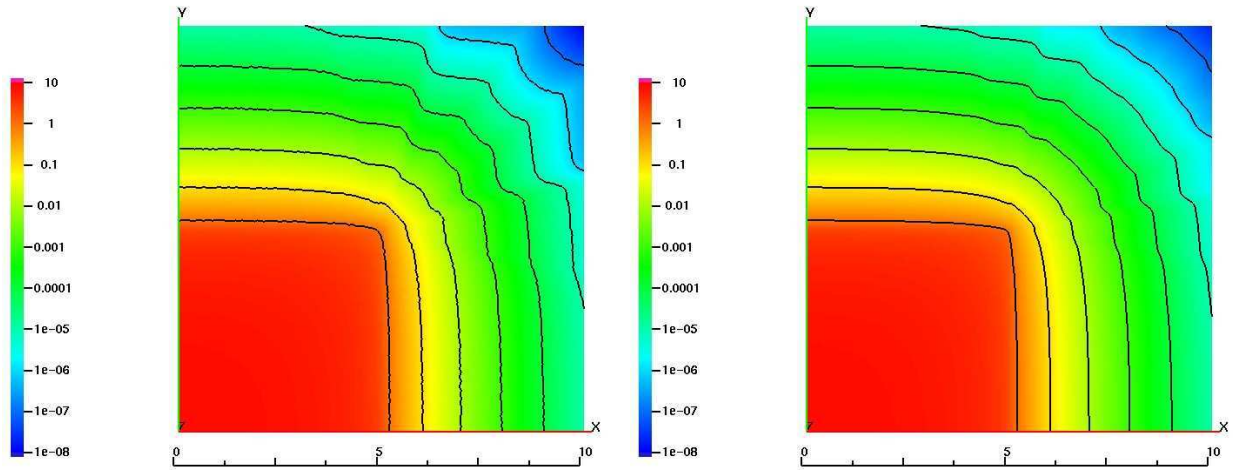
This test is on a 10cm × 10cm domain, with reflective BC on the left and bottom of the domain and vacuum BC on the top and right. There are two types of materials as shown in Fig. 5.9: a source material (green) with $\sigma_t = 1\text{cm}^{-1}$ and scattering $\sigma_s = 0.9\text{cm}^{-1}$, and a sink material (purple) with $\sigma_t = 2\text{cm}^{-1}$ and $\sigma_s = 0.1\text{cm}^{-1}$. The source material is located in $0\text{cm} \leq x, y \leq 5\text{cm}$, and the sink material is elsewhere. Sequences of $N \times N$ meshes are used (1-level), as well as 2-



(a) materials (b) refinement region

Figure 5.9: Discontinuous Media/Source Problem: Materials and refinement regions.

level meshes with refinement in the square $4\text{cm} \leq x, y \leq 6\text{cm}$. Contours of fine mesh solutions are shown in Fig. 5.10. We use 30% randomized meshes and for 2-level meshes, randomization of vertex locations are performed first, then the square region is refined. The S_{16} quadrature set is used. The scalar flux is converged to $\epsilon_\phi = 10^{-7}$.



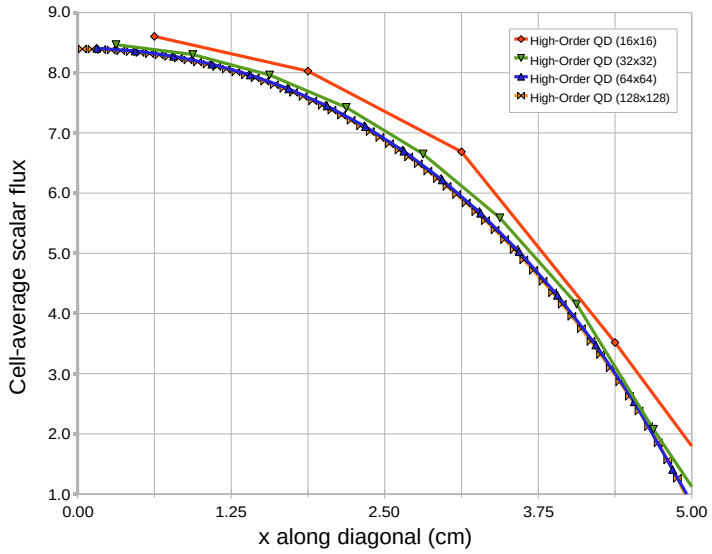
(a) 1-level

(b) 2-level

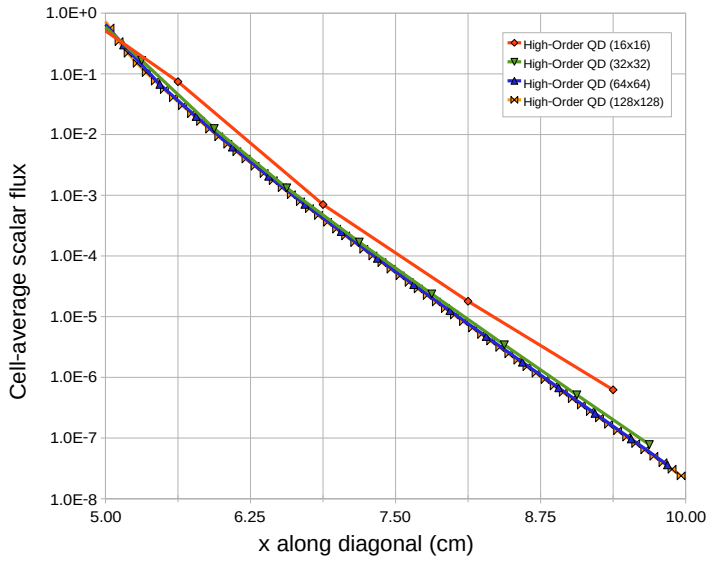
Figure 5.10: Discontinuous Media/Source Problem: Fine mesh solutions.

5.4.1 Data

We present data for 1-level and 2-level meshes with linear source approximation with our JM LOQD discretization and SCSB discretization of the high-order problem. We use a diagonal line from $(x, y) = (0\text{cm}, 0\text{cm})$ to $(10\text{cm}, 10\text{cm})$ to look at the spatial variation of cell-average scalar fluxes. In Fig. 5.11, cell-average scalar fluxes along this diagonal line are presented for various refinements and both the high-order and low-order representations of the scalar flux. Vertices of cells which lie on this diagonal are not perturbed by randomization so that a line connecting their centers may be constructed. These QD results are compared to BLD (with DSA acceleration) in Fig. 5.13 and Fig. 5.14. Spatial convergence results are shown in Fig. 5.12, for the source and sink region-average scalar fluxes, ϕ^{src} and ϕ^{sink}, respectively. The region-average values, differences between successive meshes, and ratios are shown for 1-level and 2-level meshes in Table 5.12, Table 5.13, Table 5.14, and Table 5.15.



(a) source region



(b) sink region

Figure 5.11: Discontinuous Media/Source Test: Cell-average scalar fluxes along diagonal line for the (a) source region and (b) sink region on 30% randomized meshes.

Table 5.12: Discontinuous Media/Source Problem: Source region-average scalar fluxes on 30% randomized 1-level meshes.

		high-order QD			LOQD		
N	h(cm)	ϕ_h^{src}	$\phi_{h/2}^{src} - \phi_h^{src}$	*Ratio*	ϕ_h^{src}	$\phi_{h/2}^{src} - \phi_h^{src}$	*Ratio*
16	0.625	6.017	-4.02E-1	1.64	6.110	-5.47E-1	1.66
32	0.313	5.856	-1.62E-1	2.49	5.877	-2.32E-1	2.36
64	0.156	5.811	-4.50E-2	3.59	5.817	-6.07E-2	3.83
128	0.078	5.800	-1.04E-2	4.35	5.803	-1.41E-2	4.29

Table 5.13: Discontinuous Media/Source Problem: Source region-average scalar fluxes on 30% randomized 2-level meshes.

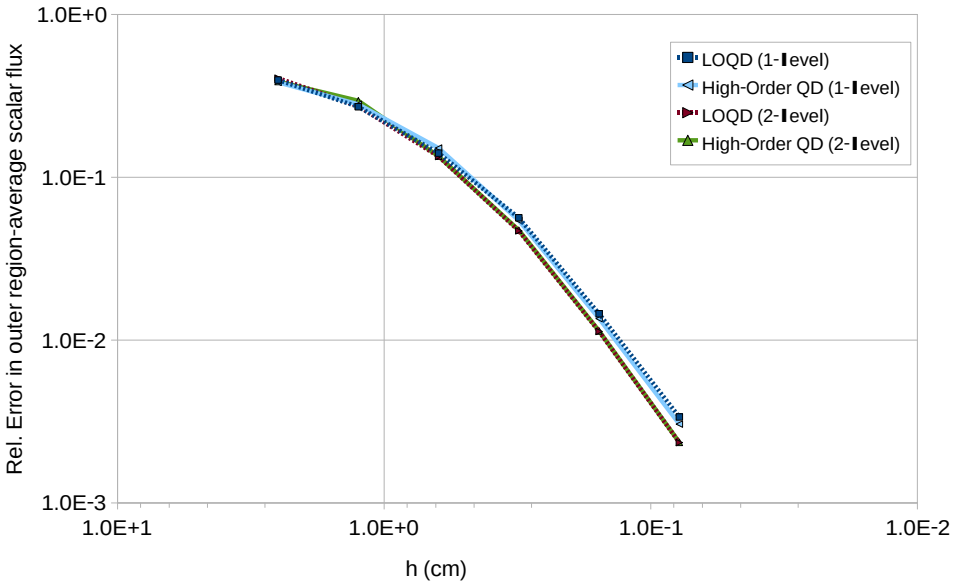
		high-order QD			LOQD		
N	h(cm)	ϕ_h^{src}	$\phi_{h/2}^{src} - \phi_h^{src}$	*Ratio*	ϕ_h^{src}	$\phi_{h/2}^{src} - \phi_h^{src}$	*Ratio*
16	0.625	5.990	-3.91E-1	1.76	6.057	-5.29E-1	1.76
32	0.313	5.847	-1.43E-1	2.73	5.864	-1.94E-1	2.73
64	0.156	5.813	-3.43E-2	4.18	5.816	-4.72E-2	4.10
128	0.078	5.806	-7.00E-3	4.89	5.806	-9.84E-3	4.80

Table 5.14: Discontinuous Media/Source Problem: Sink region-average scalar fluxes on 30% randomized 1-level meshes.

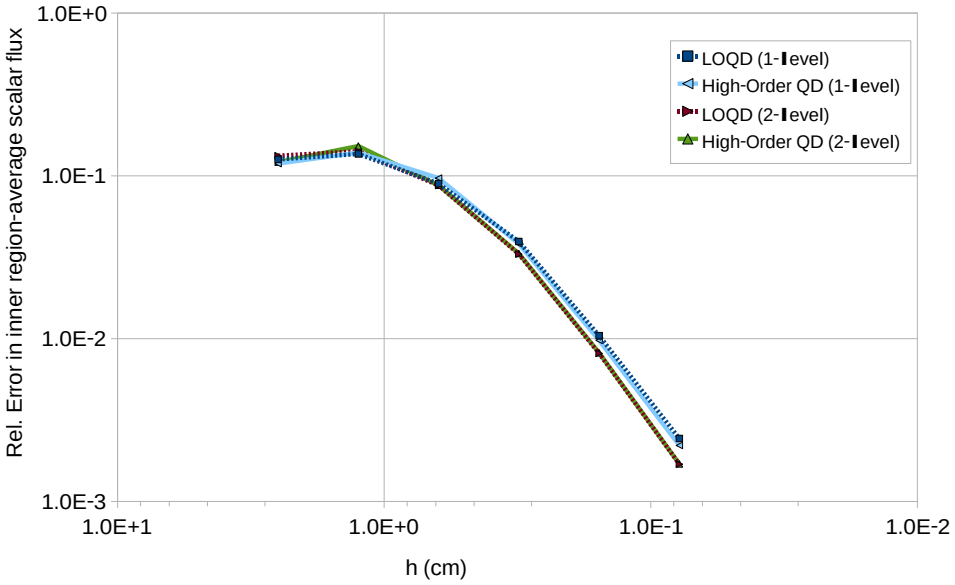
		high-order QD			LOQD		
N	h(cm)	ϕ_h^{sink}	$\phi_{h/2}^{sink} - \phi_h^{sink}$	*Ratio*	ϕ_h^{sink}	$\phi_{h/2}^{sink} - \phi_h^{sink}$	*Ratio*
16	0.625	8.242E-02	-1.58E-2	1.36	6.825E-02	9.60E-3	1.65
32	0.313	7.621E-02	-6.21E-3	2.55	7.233E-02	4.08E-3	2.36
64	0.156	7.443E-02	-1.78E-3	3.49	7.339E-02	1.07E-3	3.83
128	0.078	7.399E-02	-4.39E-4	4.05	7.364E-02	2.48E-4	4.29

Table 5.15: Discontinuous Media/Source Problem: Sink region-average scalar fluxes on 30% randomized 2-level meshes.

		high-order QD			LOQD		
N	h(cm)	ϕ_h^{sink}	$\phi_{h/2}^{sink} - \phi_h^{sink}$	*Ratio*	ϕ_h^{sink}	$\phi_{h/2}^{sink} - \phi_h^{sink}$	*Ratio*
16	0.625	8.036E-02	-1.36E-2	1.80	6.917E-02	9.28E-3	1.76
32	0.313	7.534E-02	-5.03E-3	2.71	7.257E-02	3.40E-3	2.73
64	0.156	7.400E-02	-1.33E-3	3.77	7.340E-02	8.29E-4	4.10
128	0.078	7.370E-02	-3.01E-4	4.44	7.357E-02	1.73E-4	4.80



(a) source region



(b) sink region

Figure 5.12: Discontinuous Media/Source Problem: Relative errors with refinement in (a) source region and (b) sink region on 30% randomized meshes.

5.4.2 Analysis

First, this problem exhibits eight orders of magnitude difference between the solution at $(0\text{cm}, 0\text{cm})$ and $(10\text{cm}, 10\text{cm})$, and leads to strong ray effects as seen in Fig. 5.10.

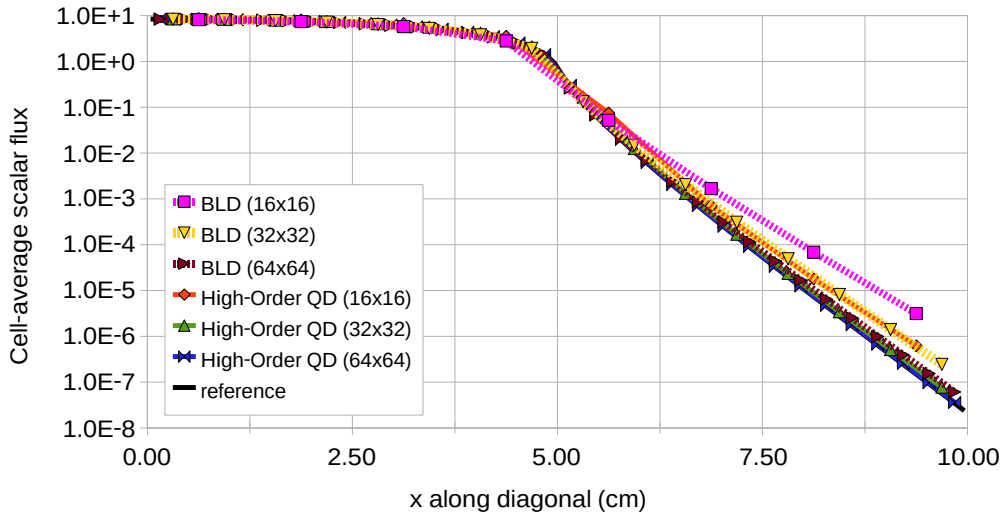
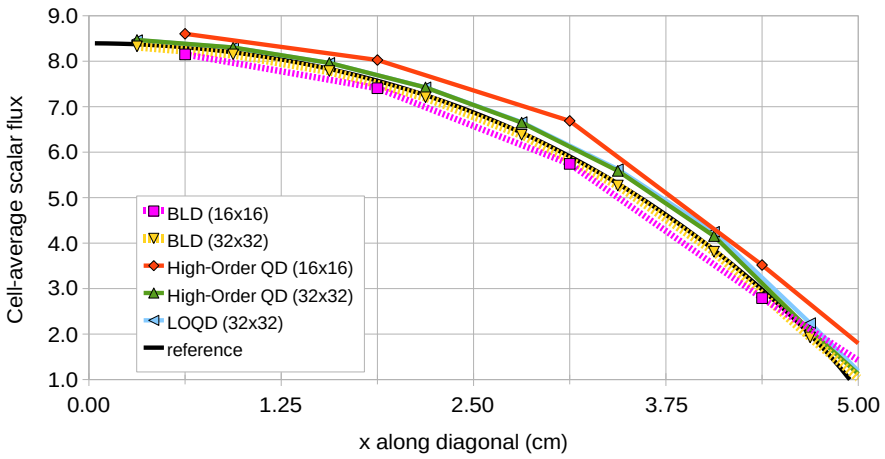


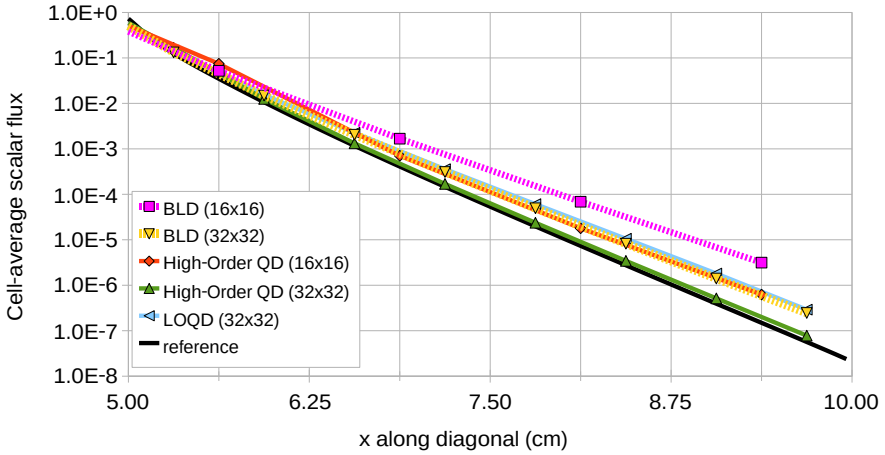
Figure 5.13: Discontinuous Media/Source Problem: Comparison to BLD over entire domain.

Fig. 5.14 shows that in the source region, high-order QD, LOQD, and BLD behave similarly. In the strongly attenuating sink region, high-order QD with h performs as well as BLD with $h/2$. In the sink region, LOQD behaves very similarly to BLD. The number of transport iterations for both BLD with DSA and QD were between 6 and 8. All convergence results show a convergence ratio that increases to 4 (the ratio for second order) and beyond. However, at the finest $h \approx 0.08\text{cm}$, the Aitken process uses values from $h \approx 0.16\text{cm}$ and $h \approx 0.32\text{cm}$ so we are not witnessing asymptotic convergence yet, and still have contributions from $O(h^3)$ terms.

The linear scattering source approximation used here leads to approximately 10% lower errors in the region-average quantities than the flat approximation (not shown) for coarse meshes



(a) source



(b) sink

Figure 5.14: Discontinuous Media/Source Problem: Comparison to BLD.

with $h > 0.625$cm. The advantage disappears for smaller h.

5.5 Duct Transport Problem

This problem simulates transport of radiation through a duct, as shown in Fig. 5.15. There is unit incident radiation, $\psi_{IN} = 1^{\text{n}}/_{\text{cm}^2\cdot\text{s}}$, on the entire left side. All other boundaries are vacuum, $\psi_{IN} = 0$. The duct material (green in Fig. 5.15) has $\sigma_t = 1\text{cm}^{-1}$ and $\sigma_s = 0.5\text{cm}^{-1}$. The wall material (purple in Fig. 5.15) has $\sigma_t = 10\text{cm}^{-1}$ and $\sigma_s = 1\text{cm}^{-1}$. The domain is a 16cm $\times$ 8cm rectangle. We examine results on $2N \times N$ meshes, both single-level and two-level, orthogonal and randomized by 20% perturbations of vertices. A representative two-level mesh is shown in Fig. 5.15 with refinement in the duct region, extending 0.5cm across the material interfaces. The S_{12} level symmetric quadrature set is used. The scalar flux is converged to a tolerance of $\epsilon_\phi = 10^{-6}$.

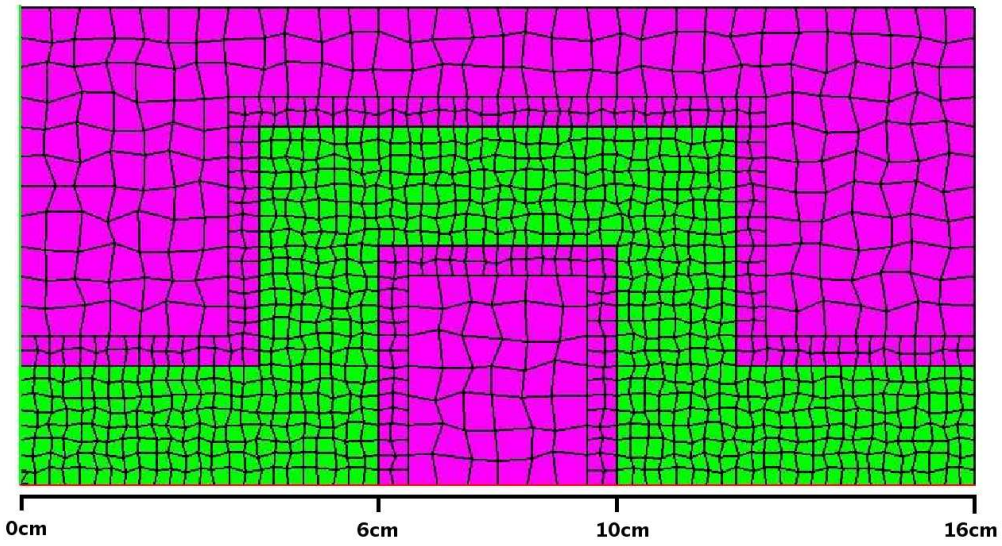
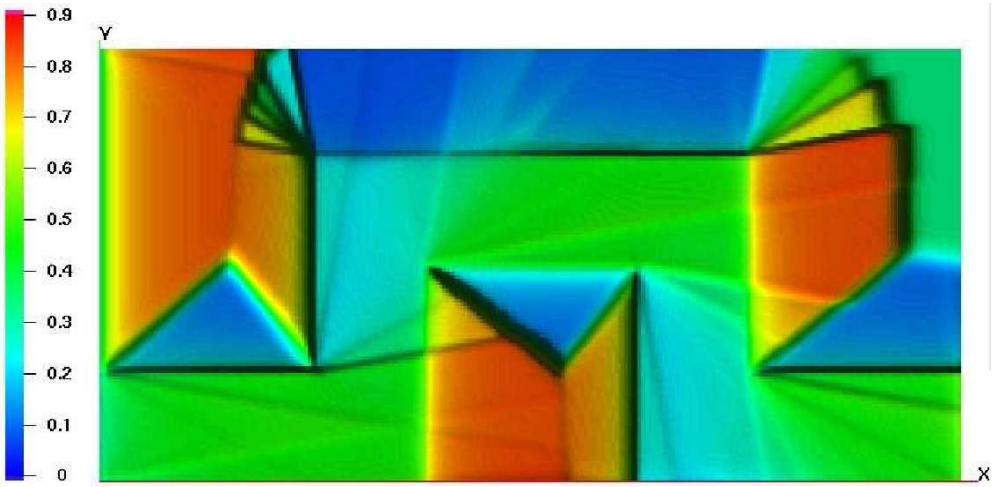
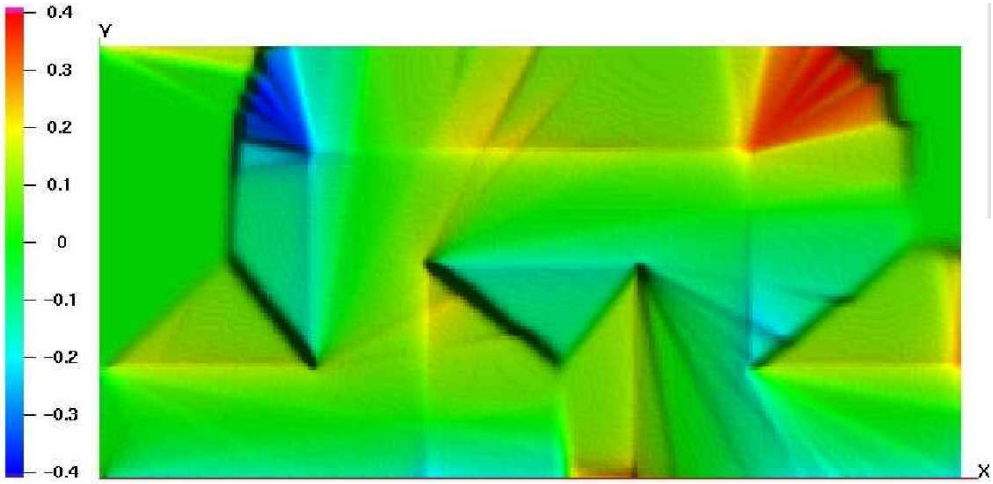


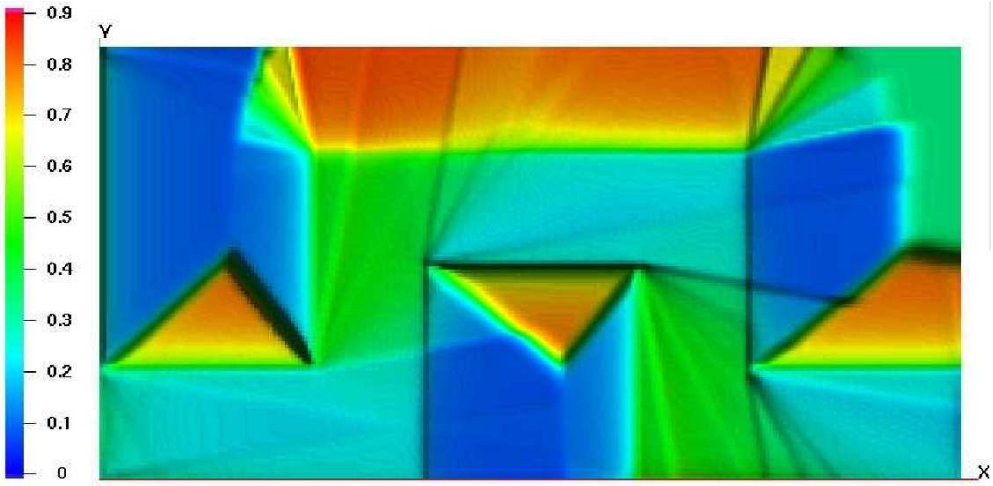
Figure 5.15: Duct Transport Problem: Material distributions on two-level mesh (32×16 with duct refinement).



(a) E_{xx}



(b) E_{xy}



(c) E_{yy}

Figure 5.16: Duct Transport Problem: QD Factors $E_{\alpha\beta}$.

5.5.1 Data

QD factors are shown in Fig. 5.16 for a 256×128 mesh. The scalar flux contours in Fig. 5.17 for SCSB with source iterations (SI), QD (with SCSB high-order discretization and JM LOQD discretization), and BLD on orthogonal 128×64 meshes. We have used the flat scattering source for all QD and SCSB results. Results for SCSB and QD on coarse randomized meshes and BLD on a coarse orthogonal mesh are shown in Fig. 5.18. We examine the region-average scalar flux in the block of wall from $6\text{cm} \leq x \leq 10\text{cm}$ and $0\text{cm} \leq y \leq 4\text{cm}$, ϕ^{block}, and the exiting flow rate of particles $F(\text{n/s})$ on the right side of the wall,

$$F_{out} = \int_{0\text{cm}}^{8\text{cm}} J_x(x = 16\text{cm}, y)\, dy\,.$$

The ϕ^{block} results with our QD method are tabulated in Table 5.16. and Table 5.17 for single-level and two-level meshes, respectively. The F_{out} results with our QD method are tabulated in Table 5.19. and Table 5.20 for single-level and two-level meshes, respectively. In tables, the column "rel. diff." is the relative difference between results on orthogonal and randomized meshes. Additionally, we present ϕ^{block} for BLD and SCSB transport discretizations in Table 5.18 on single-level meshes and F_{out} for SCSB on single-level meshes in Table 5.21.

Table 5.16: Duct Transport Problem: QD block region-average scalar flux for single-level meshes.

		High-Order ϕ^{block}			LOQD ϕ^{block}		
h(cm)	N_C	Ortho.	Rand.	rel. diff.	Ortho	Rand.	rel. diff.
1/2	512	2.7856E-04	2.8872E-04	3.583%	1.2536E-04	1.2428E-04	0.860%
1/4	2048	2.0587E-04	2.0284E-04	1.486%	1.4260E-04	1.4523E-04	1.824%
1/8	8192	1.6244E-04	1.6434E-04	1.162%	1.4420E-04	1.4424E-04	0.026%
1/16	32768	1.4637E-04	1.4798E-04	1.093%	1.4205E-04	1.4208E-04	0.021%

Table 5.17: Duct Transport Problem: QD block region-average scalar flux for two-level meshes.

		High-Order ϕ^{block}			LOQD ϕ^{block}		
h^*(cm)	N_C	Ortho.	Rand.	rel. diff.	Ortho	Rand.	rel. diff.
1/4	1298	2.0587E-04	2.0913E-04	1.571%	1.4260E-04	1.4531E-04	1.881%
1/8	5192	1.6244E-04	1.6428E-04	1.124%	1.4420E-04	1.4553E-04	0.918%
1/16	20768	1.4637E-04	1.4756E-04	0.810%	1.4205E-04	1.4319E-04	0.797%

$*$ the smaller mean-cell width on the two-level mesh

Table 5.18: Duct Transport Problem: SCSB and BLD block region-average scalar flux.

		SCSB			BLD
h(cm)	N_C	Ortho.	Rand.	rel. diff.	Ortho
1/2	512	1.5396E-04	1.5470E-04	0.485%	1.57463E-4
1/4	2048	1.4426E-04	1.4432E-04	0.046%	1.45705E-4
1/8	8192	1.4188E-04	1.4195E-04	0.048%	1.42417E-4
1/16	32768	1.4138E-04	1.4139E-04	0.009%	1.41520E-4

Table 5.19: Duct Transport Problem: QD exiting flow rate for single-level meshes.

		High-Order F_{out}			LOQD F_{out}		
h(cm)	N_C	Ortho.	Rand.	rel. diff.	Ortho	Rand.	rel. diff.
1/2	512	2.8396E-08	2.8303E-08	0.330%	3.0750E-08	3.0863E-08	0.367%
1/4	2048	7.3446E-09	7.3971E-09	0.712%	7.7731E-09	7.8688E-09	1.223%
1/8	8192	2.2051E-09	2.2711E-09	2.948%	2.2580E-09	2.3329E-09	3.264%
1/16	32768	1.2510E-09	1.2796E-09	2.264%	1.2588E-09	1.2856E-09	2.107%

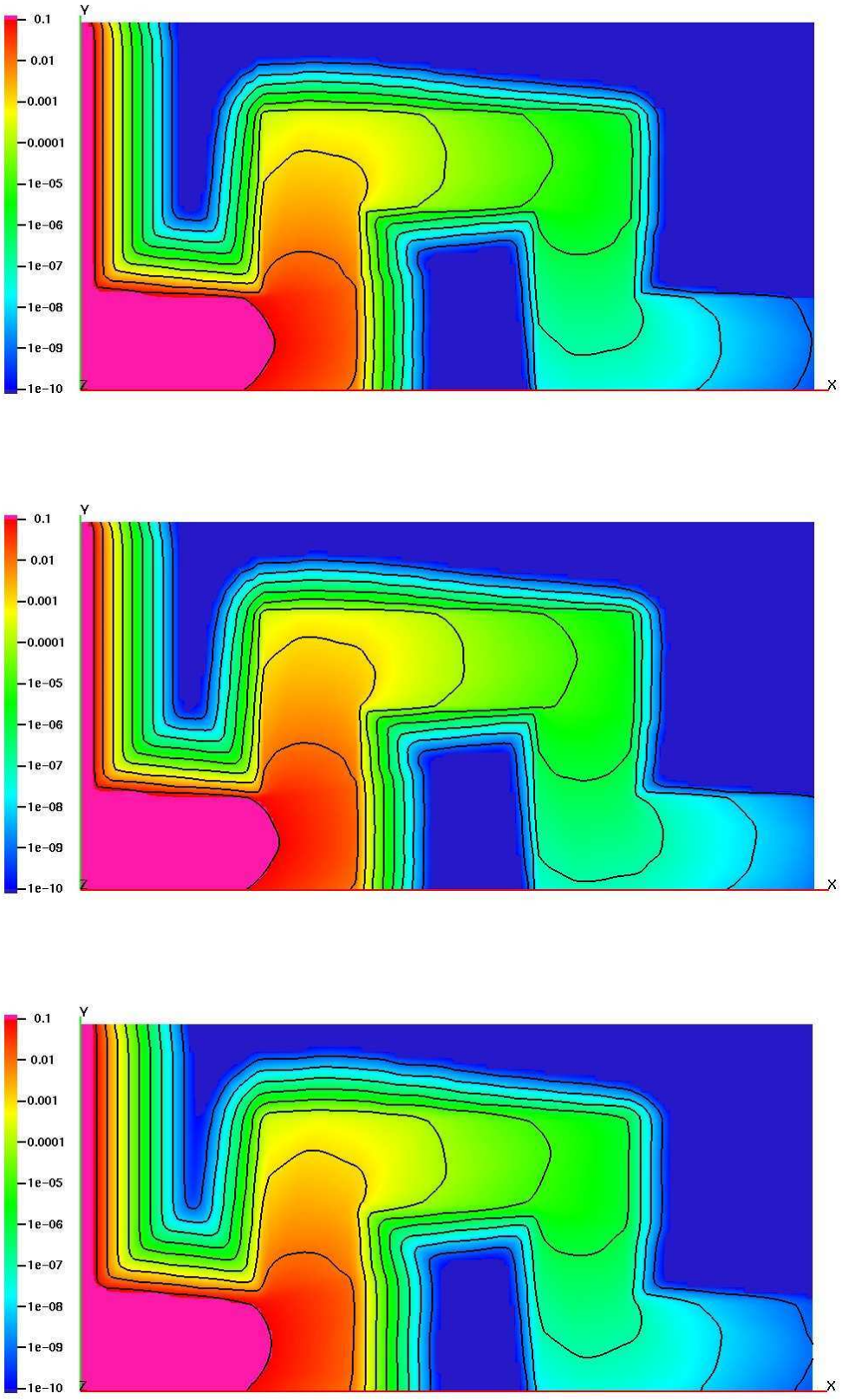
Table 5.20: Duct Transport Problem: QD exiting flow rate for two-level meshes.

		High-Order F_{out}			LOQD F_{out}		
h^*(cm)	N_C	Ortho.	Rand.	rel. diff.	Ortho	Rand.	rel. diff.
1/4	1298	7.3448E-09	7.8631E-09	6.816%	7.7734E-09	8.3276E-09	6.884%
1/8	5192	2.2051E-09	2.3103E-09	4.657%	2.2580E-09	2.3740E-09	5.009%
1/16	20768	1.2510E-09	1.2862E-09	2.775%	1.2588E-09	1.2978E-09	3.046%

$*$ the smaller mean-cell width on the two-level mesh

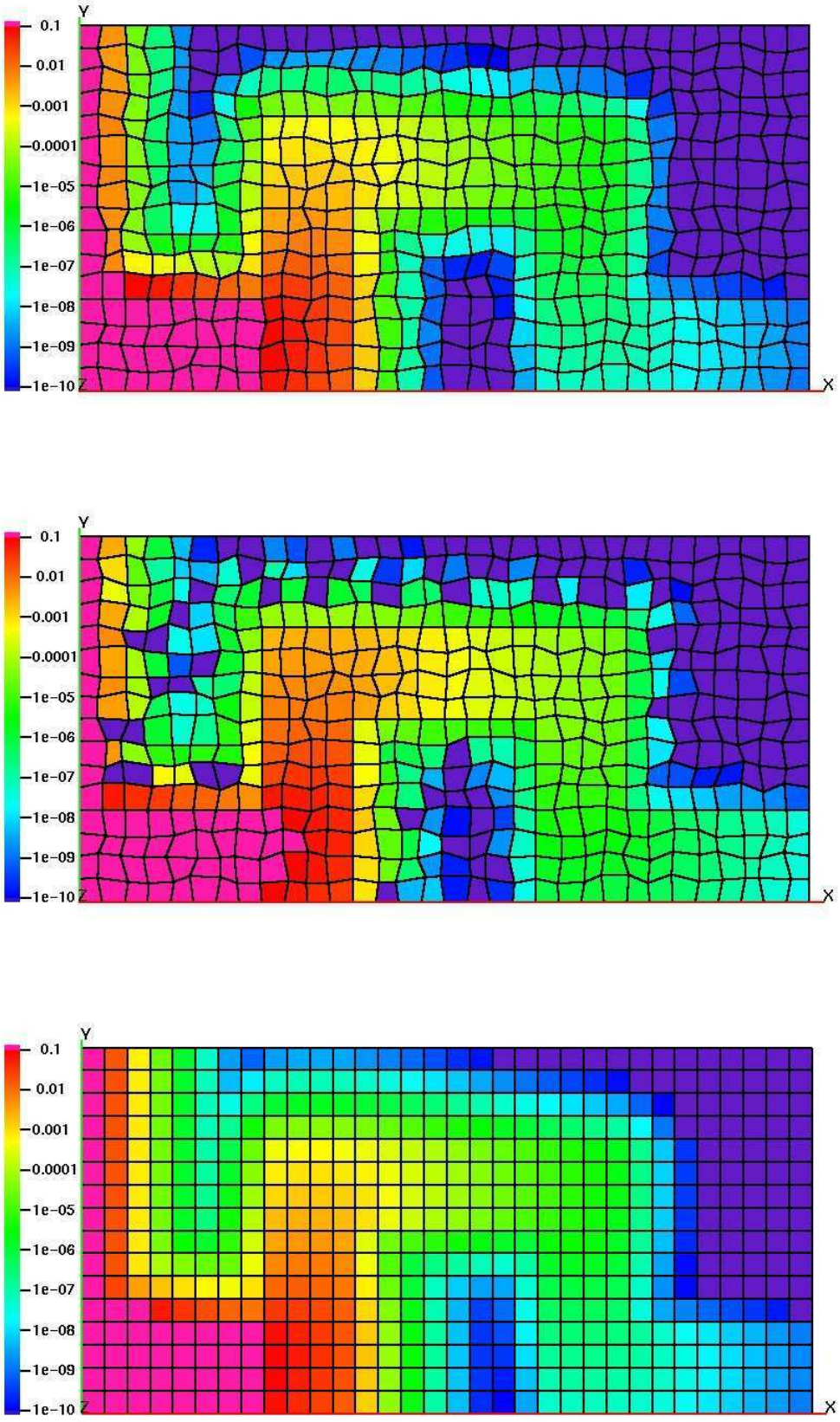
Table 5.21: Duct Transport Problem: SCSB exiting flow rate.

		SCSB		
h(cm)	N_C	Ortho.	Rand.	rel. diff.
1/2	512	1.2923E-09	1.2832E-09	0.711%
1/4	2048	1.0056E-09	1.0070E-09	0.146%
1/8	8192	9.7549E-10	9.7588E-10	0.040%
1/16	32768	9.7898E-10	9.7938E-10	0.041%



(c) BLD

Figure 5.17: Duct Transport Problem: Scalar flux contours on 128×64 meshes.



(c) BLD

Figure 5.18: Duct Transport Problem: Scalar flux on (coarse) 32×16 meshes.

5.5.2 Analysis

The QD factors $E_{\alpha\beta}$ in Fig. 5.16 show strong variations, including ray effects, in the duct system. All methods show similar contours up to the first bend in Fig. 5.17. Beyond the first bend, QD shows larger scalar fluxes than SCSB with SI or BLD. SCSB and QD show nearly identical attenuation in the wall region above the duct entrance, for a sufficiently fine mesh. With a coarse mesh, as in Fig. 5.18(c), the combination of strong incident flux plus large absorption cross sections, $\sigma_a = \sigma_t - \sigma_s = 9\text{cm}^{-1}$ leads to small, negative cell-average scalar fluxes with our JM discretization. In this problem, the breakdown occurs on randomized meshes only in cells with incident flux on one side and $\sigma_a h > 2$, which leads to approximately an order of magnitude change in the scalar flux across a cell. However, only the cells in the wall material have this breakdown and this does not seem to have a dramatic effect on ϕ^{block} or F_{out}. A potential remedy is discussed in the summary of this chapter.

From the perspective of acceleration, in the duct, the scattering ratio $c = 0.9$, leads to 27 transport iterations with SCSB using source iterations (SI), 7 iterations with QD, and 7 iterations with BLD (accelerated with DSA) with scalar flux tolerance of $\epsilon_\phi = 10^{-6}$. With our LOQD solver, the time spent in the low-order solver is a fraction of the time spent in the high-order transport sweep (an average of 1/8 the time for this problem), so the cost of QD is effectively $7/27 \approx 26\%$ the cost of SI. We have used the flat scattering source representation here, because for large gradients, the linear scattering source can be negative in parts of a cell, and this should be avoided. However, as it turns out, the linear scattering source representation may be used for this test, but there is not much advantage in doing so for F_{out} and ϕ^{block}—they show negligible difference for linear and flat scattering representations.

The region-average scalar fluxes ϕ^{block} in Table 5.16, show small relative differences between randomized and orthogonal meshes: about 1% for the high-order QD representation and about 0.02% for the LOQD representation. On two-level meshes, both high-order QD and LOQD representations differ on orthogonal and randomized meshes by about 0.8% on the finest meshes. Two-level meshes in Table 5.17 show significant efficiency gains, with nearly identical ϕ^{block} to a single level mesh with 50% more cells. Compared to the SCSB transport discretization in Table 5.17, where ϕ^{block} has almost 4 digits converged, our QD method is less accurate, with 2 digits converged for the high-order and 3 digits converged for the LOQD method. This is one example where the LOQD problem yields a more accurate solution than the high-order one. In this quantity, ϕ^{block}, the LOQD result is similar to the orthogonal BLD results shown in Table 5.18.

The exiting flow rates F_{out}, are on the order of 10^{-9}n/s, while the incident particle flow rate is $F_{in} = \int_0^8 \int_{2\pi} \psi_{IN} d\Omega \;\; dy = 16\pi$n/s, for about 11 orders of magnitude attenuation of the incident radiation through the duct system. On the last stage of refinement, F_{out} calculated by QD method changes by a factor of 2. With SCSB, it is converged to 2 digits. Thus, if the quantity of interest is the exiting flow rate, it may be useful to consider a framework for pure acceleration.

5.6 Summary

In this chapter, we have performed numerous tests of our QD method, which is composed of high-order QD discretization via SCSB and LOQD discretization via the JM FV method of Chapter 2. These tests have been chosen to show the behavior and performance of the method under a wide range of conditions. A short review of the tests follows.

1. Uniform External Source Test

We showed second-order convergence of our QD method on randomized meshes in a simple problem with no scattering. Complicated boundary behavior of E exists in this simple problem. The LOQD results with E calculated via SCV and SCSB show similar accuracy, less than the first-order characteristic method, ESC, but overtaking with mesh refinement. The SCSB method shows remarkable accuracy for this simple problem.

2. Diffusion Limit Test

We showed that our QD method does indeed limit to diffusion as the smallness parameter $\epsilon \to 0$. The number of transport iterations for the scattering does increase slightly with mesh randomization.

3. Analytic Transport Test, Revisited

In this analytic test, we showed second-order convergence for the high-order and LOQD solution in the ℓ_2 error norm. Because of the anisotropic external source in this test, the RHS of LOQD's first moment equations is nonzero and on some faces negative, which did not seem to cause any trouble for the discretization. This leads us to believe the discretization will behave well in multigroup problems with anisotropic scattering, where there are similar (possibly negative) terms. The high-order solution is more accurate than the LOQD solution in this test and the linear approximation of the scattering source further improves the accuracy of the high-order solution by a factor of 5.

4. Discontinuous Media/Source Problem

This problem with reflective boundary conditions showed near second-order numerical convergence for a problem with relatively strong source and sink materials. A patch of adaptive meshing on a corner improved results on coarse meshes.

5. Duct Transport Problem

This difficult problem transporting radiation through a duct showed great advantage in using two-level meshes, with refined cells in the duct. The QD discretization approximated a region-average scalar flux very well but the exiting flow rate of particles was overapproximated by factors of approximately $20, 8, 2, 1.2$ for $h = 1/2, 1/4, 1/8, 1/16$cm meshes, respectively, whereas SCSB with source iterations on the $h = 1/16$cm mesh was converged to 2 digits.

One potential problem with the LOQD discretization was discovered in the duct transport problem, where the combination of strong attenuation through the wall and coarse meshes lead to the scalar flux varying an order of magnitude across a single cell. On skewed cells, the JM discretization breaks down and the cell-average flux becomes negative. On orthogonal cells, however, where the flow direction is aligned with outward normals of faces, there is no breakdown. A possible remedy on skewed cells is switching to a different discretization in the cell, perhaps one with reduced support and designed to handle steep gradients, like the AK discretization [54], similar to lumping with DFEM methods [80].

Chapter 6

CONCLUSIONS

We have developed a quasidiffusion (QD) method for solving radiation transport problems on unstructured quadrilateral meshes in 2D Cartesian geometry which utilizes a new low-order quasidiffusion (LOQD) discretization via the finite volume (FV) method. In analytic and numeric tests, our JM discretization, which is based on the cell-centered diffusion discretization in [89], shows improved accuracy compared to LOQD discretizations from the literature [24] and [54]. Additionally, we have proposed interface conditions for cells with hanging nodes that still allow cells to be treated as quadrilaterals, thus not introducing additional unknowns into the system. We have found that efficient iterative solution of the LOQD equations can be achieved with Krylov methods and preconditioning via incomplete LU factorization with diagonal compensation.

We have exploited the fact that QD allows independent discretizations of the high-order and low-order problems and have used short characterstics with subcell balances (SCSB) for the high-order transport discretization. The SCSB transport discretization exhibits robust behavior in important limits for characteristic transport discretizations, e.g. grazing angles and voids, due to

the use of polynomial exponential moments and is second-order. Although SCSB is more expensive than the vertex-based, short characteristics (SCV) method we also developed for use with our LOQD discretization, an SCV method could not be extended to arbitrary meshes with retention of the second-order.

We have also developed a linear representation of the isotropic component of the scattering source based on face-average and cell-average scalar fluxes that is effective in some problems. For some problems, there is not much gain over the flat representation of the scattering source.

In numerical tests, the resultant QD method (LOQD plus transport discretization) with linear scattering source representation shows advantage over bilinear discontinuous (BLD) transport discretizations for coarse meshes.

Future Work

The first major avenue for future work is extension of the methods presented here to arbitrary meshes in 3D Cartesian geometry. The LOQD discretization we have developed may be easily extended to 3D polyhedra, like the diffusion discretization [89] from which it was adapted. The SCSB discretization lends itself to 3D extension in all manners except the procedure for parabolic interpolation with monotonization, for which an extension of the parabolic interpolant on 1D linear faces to a paraboloid interpolant on 2D planar faces is not immediately apparent.

Future work should also theoretically address the effect of the new hanging-node interface conditions. For example, one can compare the discretization of 1) a small 3-cell hanging node system with the hanging-node interface conditions with 2) the same system with the large cell treated as a 5-sided polygon and standard interface conditions. This analysis may lead to better

weights for the weak condition and strong conditions. If it does not, on could also use the high-order problem to calculate weights for hanging-node interfaces that force the low-order discretization to redistribute scalar flux and currents similarly to the distribution in the high-order problem.

The other major avenue for future work is in preconditioning the LOQD system. The cell-local discretization involving unknowns on cells and faces, plus interface conditions lends itself to a block-like structure of the LOQD matrix system, we call the cell-based structure, that has been solved effectively with BiCGstab and ILUD preconditioning. However, other preconditioners should be investigated: For example, approximate inverse and algebraic multigrid preconditioning should be tested.

BIBLIOGRAPHY

[1] A. Eddington, "The Internal Constitution of the Stars," p. 322. Dover, New York (1926).

[2] E.S. Kuznetsov, "On the Establishment of Equilibrium of Radiant Energy in an Absorbing and Scattering Atmosphere," lzv. AN SSSR, Sev. Geol. and Geophysics., 813-842 (1940), (in Russian).

[3] S.K. Godunov, "A Finite difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics," *Mat. Sb.*, 47:357-393 (1959).

[4] V.Ya. Gol'din, "A Quasi-Diffusion Method for Solving the Kinetic Equation," *USSR Comp. Math. and Math. Phys.* **4**, 136 (1964).

[5] K.M. Case and P.E. Zweifel, **Linear Transport Theory**, Addison-Wesley, Reading, MA (1967).

[6] V. Ya. Gol'din, N. N. Kalitkin , & T. V. Shishova, *USSR Comp. Math. and Math. Phys.* **5**, 229 (1968).

[7] A.M. Winslow, "Extensions of Asymptotic Neutron Diffusion Theory," *Nucl. Sci. Eng.*, **32**, 101 (1968).

[8] T.A. Germogenova, "Convergence of Some Approximate Methods of Solving the Transport Equation," *Soviet Mathematics - Doklady* **9**, 855 (1968).

[9] K. D. Lathrop, "Spatial Differencing of the Transport Equation: Positivity vs. Accuracy," *J. of Comp. Phys.*, 4, No. 4, pp. 475-498 (1969).

[10] G. C. Pomraning, "An Extension of the Eddington Approximation," *J. Quant. Spectrosc. Radiat. Transfer*, Vol. 9, pp. 407-422, Pergamon Press (1969).

[11] V.Ya. Gol'din, G.V. Danilova and B.N. Chetverushkin, "Approximate Method of Solving the Time-Dependent Kinetic Equation," in *Computational Methods in Transport Theory* (Atomizdat, Moscow, 1969) pp. 50-57 (in Russian).

[12] E.W. Larsen and J.B. Keller, "Asymptotic Solution of Neutron Transport Problems for Small Mean Free Paths," *J. Math. Phys.*, **15**, 75 (1974).

[13] T.R. Hill, Technical Report No. LA-5990-MS, Los Alamos Scientific Laboratory, June 1975 (unpublished).

[14] G.J. Habetler and B.J. Matkowsky, "Uniform Asymptotic Expansions in Transport Theory with Small Mean Free Paths, and the Diffusion Approximation," *J. Math. Phys.*, **16**, 846 (1975).

[15] D.S. Kershaw, "Flux Limiting Nature's Own Way," Lawrence Livermore National Laboratory, Livermore, CA, UCRL-78378 (1976).

[16] G. N. Minerbo, "Maximum entropy Eddington factor," *J. Quant. Spectrosc. Radiat. Transfer*, **20**, 541 (1978).

[17] A.H. Sherman, "Algorithm 533: NSPIV, a Fortran subroutine for sparse Gaussian elimination with partial pivoting [F4]," *ACM Trans. Math. Softw.*, **4**, 4, pp. 391-398 (1978).

[18] N. N. Aksenov and V. Ya. Gol'din, "Computation of the Two-Dimensional Stationary Equa-

tion of Neutron Transfer by the Quasi-Diffusion Method," *USSR Comp. Math. and Math. Phys.*, **19**, No. 5, 263-266 (1979).

[19] R.E. Alcouffe, E.W. Larsen, W.F. Miller and B.R. Wienke, "Computational Efficiency of Numerical Methods for the Multigroup, Discrete-Ordinates Neutron Transport Equations: The Slab Geometry Case," *Nucl. Sci. Eng.* **71**, 111 (1979).

[20] M.D. Brough and C.T. Chudley, "Characteristic Ray Solutions of the Transport Equation," *Adv. Nucl. Sci. Technol.*, 12, 1 (1980).

[21] R.E. Alcouffe, E.W. Larsen, W.F. Miller and B.R. Wienke, "A Review of Characteristic Methods Used to Solve the Linear Transport Equation," Proc. ANS/ENS Joint Topical Meeting, *Advances in Mathematical Methods for the Solution of Nuclear Engineering Problems*, April 27-29, 1981, Munich, **1**, 3 (1981).

[22] E. W. Larsen, "Unconditionally stable diffusion-synthetic acceleration methods for the slab geometry discrete ordinates equations. Part I: Theory", *Nucl. Sci. Eng.* **82**, 47 (1982).

[23] D. R. McCoy and E. W. Larsen, "Unconditionally stable diffusion-synthetic acceleration methods for the slab geometry discrete ordinates equations. Part II: Numerical results", *Nucl. Sci. Eng.* **82**, 64 (1982).

[24] V.Y. Gol'din, D. A. Gol'dina and A.V. Kolpakov, "The Solution of Two-Dimensional Stationary Quasidiffusion Problem," *Preprint of the Keldysh Institute for Applied Mathematics*, the USSR Academy of Sciences, No. 49 (1982) (in Russian).

[25] V.V. Gorskii and S.T. Surzhikov, "Use of the Semimoment Method to Solve the Shock Layer Radiative Heat-Transfer Problem," *J. of Eng. Physics*, **42**, No. 1, 108 (1982).

[26] V.Ya. Gol'din, A.V. Kolpakov, and A.V. Misyurev, "Solution of the nonstationary transport equation without explicit determination of the front," *Preprint of Keldysh Institute of Applied Mathematics, USSR Academy of Sciences*, **68**, (1983) (in Russian).

[27] C. D. Levermore, "Relating Eddington factors to flux limiters," *J. Quant. Spectroscop. Radiat. Transfer*, 31, 149-160 (1984).

[28] D. Mihalas and B. Mihalas, **Foundations of Radiation Hydrodynamics**, Oxford University Press, Oxford, 374-378 (1984).

[29] Edward W. Larsen, "Diffusion-Synthetic Acceleration Methods for Discrete-Ordinates Problems," *Transport Theory Statist. Phys.*, **13(1&2)**, 107-126 (1984).

[30] E.E.Lewis and W.F.Miller, Jr. "Computational Methods of Neutron Transport," Wiley- Interscience, New York. (1984). [Reprinted (1993). American Nuclear Society, LaGrange Park, Illinois.]

[31] G.I. Marchuk and V.I. Lebedev, **Numerical Methods in the Theory of Neutron Transport**, Harwood Academic Publishes, London (1986).

[32] A. Harten, B. Engquist, S. Osher, and S.R. Chakravarthy, "Uniformly high order accurate essentially non-oscillatory schemes," *J. of Comp. Phys.*, **71**, 231-303 (1987).

[33] E.W. Larsen, "Projected Discrete Ordinates Methods for Numerical Transport Problems," *Nucl. Sci. Eng.*, **92**, 197 (1986).

[34] M. I. Bakirova, V.Ya. Karpov, and M.I. Mukhina, **Differential Equations**, Consultants Bureau, New York, English translation (1987), Russian original (1986).

[35] M. J. Berger & P. Colella, "Local Adaptive Mesh Refinement for Shock Hydrodynamics," *J. of Comp. Phys.*, **88**, 64 (1989).

[36] J.E. Morel, "A hybrid collocation-Galerkin-S_n method for solving the Boltzmann transport equation," *Nucl. Sci. Eng.*, **101**, 1, pp. 72-87 (1989).

[37] P.S. Heckbert, "Fundamentals of Texture Mapping and Image Warping," Master's Thesis, University of California at Berkely (1989).

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

[39] G.R. Cefus and E.W. Larsen, "Stability Analysis of the Quasidiffusion and Second Moment Methods for Iteratively Solving Discrete-Ordinates Problems," *Transport Theory Statist. Phys.*, **18**, 493 (1989-90).

[40] M.M. Kulas and G. Moses, "Projected Discrete Ordinates Methods for S_N Problems in X-Y Geometry," Proc. ANS Topical Meeting, *Adv. in Math., Computation, and Reactor Phys.*, April 29 - May 2, 1991, Pittsburgh, Vol. 2, Sec. 11.2, pp. 2-1 through 2-12 (1991).

[41] T.Wareing and E.W. Larsen and M.L. Adams, "Diffusion Accelerated Discontinuous Finite Element Schemes for the SN Equations in Slab and X,Y Geometries," Proc. ANS Topical Meeting, Advances in Mathematics, Computations, and Reactor Physics, April 29 - May 2, 1991, Pittsburgh, 3, Sec. 11.1, p. 2-1 (1991).

[42] M.L. Adams, and W.R. Martin, "Diffusion Synthetic Acceleration of Discontinuous Finite Element Transport Iterations," *Nucl. Sci. Eng.*, **111**, 145-167 (1992).

[43] R. J. LeVeque, **Numerical Methods for Conservation Laws**, Birkhauser (1992).

[44] E. W. Larsen, "The Asymptotic Diffusion Limit of Discretized Transport Problems," *Nucl. Sci. Eng.*, **112**, 336 (1992).

[45] Y. Y. Azmy, "Arbitrarily High Order Characteristic Methods for Solving the Neutron Transport Equation," *Ann. Nucl. Energy*, **19**, 10-12, 593 (1992).

[46] M. D. DeHart, A Discrete Ordinates Approximation to the Neutron Transport Equation Applied to Generalized Geometries, Ph.D. Dissertation, Texas A&M University (1992).

[47] D.Y. Anistratov and V.Ya. Gol'din, "Nonlinear Methods for Solving Particle Transport Problems," *Transport Theory Statist. Phys.* **22**, 125 (1993).

[48] B.T. Adams and J.E. Morel, "A Two-Grid Acceleration Scheme for the Multigroup SN Equations with Neutron Upscattering," *Nucl. Sci. Eng.* **115**, 253 (1993).

[49] M.M. Miften and E.W. Larsen, "The Quasi-Diffusion Method for Solving Transport Problems in Planar and Spherical Geometries," *Transport Theory Statist. Phys.*, **22**, 165 (1993).

[50] M. Miften & E. W. Larsen, "A Symmetrized Quasidiffusion Method for Solving Transport Problems in Multidimensional Geometries," *Proceedings of Joint International Conference on Mathematical Methods and Supercomputing in Nuclear Applications, April 19-23, 1993, Karlsruhe, Germany.*

[51] M.M. Miften, "A Symmetrized Quasidiffusion Method for Solving Transport Problems in Multidimensional Geometries," PhD. Dissertation, The University of Michigan (1993).

[52] A.V. Voronkov and E.P. Sychugova "CDSN-method for solving the transport equation," *Transport Theory Statist. Phys.*, 22:2, 221-245 (1993).

[53] M.L. Adams, "New Nonlinear Methods for Linear Transport Calculations," in *Proceedings of Joint International Conference on Mathematical Methods and Supercomputing in Nuclear Applications, April 19-23, 1993, Karlsruhe, Germany* pp. 683-694.

[54] E.N.Aristova & A.V.Kolpakov, "A Combined Finite Difference Scheme for an Elliptic Operator in an Oblique-Angled Cell," *Mathematical Modeling and Computational Experiment*, **1**, 187 (1993).

[55] J.E. Morel and J.E. Dendy and T.A.Wareing, "Diffusion Accelerated Solution of the 2-D SN Equations with Bilinear-Discontinuous Differencing," *Nucl. Sci. Eng.* **115**, 304 (1993).

[56] M. Miften and E.W.Larsen, "The SQD Method for SN Problems on Equilateral Triangular Meshes," Trans. Am. Nucl. Soc. 70, 156 (1994).

[57] M. D. DeHart, R. E. Pevey, and T. A. Parish, "An Extended Step Characteristic Method for Solving the Transport Equation in General Geometries," *Nucl. Sci. Eng.*, **118**, 79 (1994).

[58] W.F. Walters, T.A. Wareing, and D.R. Marr, "The Nonlinear Characteristic Scheme for X-Y Geometry Transport Problems," Proceedings of *M&C 1995, International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses,* American Nuclear Society Topical Meeting of M&C Division, Portland, Oregon, April, 1995.

[59] S. F. Ashby, P. N. Brown, M. R. Dorr, and A. C. Hindmarsh, "A linear algebraic analysis of diffusion synthetic acceleration for the Boltzmann transport equation", *SIAM J. Numer. Anal.* **32**, 179 (1995).

[60] R.E. Grove, "A Characteristic-based, Multiple Balance Approach for Solving the S_N Equations on Arbitrary Polygonal Meshes, PhD. Dissertation, The University of Michigan (1996).

[61] D.J. Miller and K.A. Matthews, and C.R. Brennan, "Split-Cell Discrete Ordinates Transport on an Unstructured Grid of Triangular Cells," *Transport Theory Statist. Phys.*, **25(7)**, 833-867 (1996).

[62] G.L. Ramone, M.L. Adams, and P.E. Nowak, "A Transport Synthetic Acceleration Method for Transport Iterations," *Nucl. Sci. Eng.*, **125**, 257. (1997)

[63] K.A. Mathews, R.L. Miller, and C.R. Brennan, "Exponential Characteristic Nonlinear Radiation Transport Method for Unstructured Grids of Triangular Cells," *Nucl. Sci. Eng.*, **126**, 264 (1997).

[64] M.L. Adams, "Subcell Balance Methods for Radiative Transfer on Arbitrary Grids," *Transport Theory Statist. Phys.*, **26**, 4&5, 385 (1997).

[65] E.N. Aristova and V.Ya. Gol'din, "Computation of the anisostropy scattering of solar radiation in atmosphere (monoenergetic case)," *J. Quant. Spectrosc. Radiat. Transfer*, 67, 139-157 (2000).

[66] M.L. Adams, T.A. Wareing, and W.F. Walters, "Characteristic Methods in Thick Diffusive Problems," *Nucl. Sci. Eng.*, **130**, 18 (1998).

[67] Joseph O'Rourke, *Computational Geometry in C*, Cambridge University Press, Cambridge (1998).

[68] E. N. Aristova, V. Ya. Gol'din, & A. V. Kolpakov, "Multidimensional Calculations of Radiation Transport By Nonlinear Quasi-Diffusional Method," *Proceeding of M&C 1999 Topical Meeting*, Madrid, Spain, 667-676 (1999).

[69] K.A. Mathews, R.L. Miller, and C.R. Brennan, "Split-Cell Linear Characteristic Transport Method for Unstructured Tetrahedral Meshes," *Nucl. Sci. Eng.*, **136**, 178 (2000).

[70] Y. Azmy, "Acceleration of Multi-dimensional Discrete Ordinates Methods Via Adjacent-Cell Preconditioners," *Nucl. Sci. Eng.*, **136**, 202-226 (2000).

[71] V. A. Mousseau, D. A. Knoll, and W. J. Rider, "Physics-based preconditioning and the NewtonKrylov method for non-equilibrium radiation diffusion", *J. of Comp. Phys.* **160**, 743 (2000).

[72] M.R. Zika and M.L. Adams, "Transport Synthetic Acceleration Method for Long-Characteristics Assembly-Level Transport Problems," *Nucl. Sci. Eng.*, **134**, pp. 135-158 (2000).

[73] T.A. Wareing, J.E. Morel, J.M. McGhee, "Coupled Electron-Photon Transport Methods on 3-D Unstructured Grids," Trans Am. Nucl. Soc., Washington D.C., Vol 83, 2000.

[74] Y. Saad, computer code library SPARSKIT, University of Minnesota, Minneapolis, MN, USA (2000).

[75] Todd A. Wareing, John M. McGhee, Jim E. Morel, and Shawn D. Pautz "Discontinuous

Finite Element SN Methods on Three-Dimensional Unstructured Grids," *Nucl. Sci. Eng.*, **138**, 256-268 (2001).

[76] Charles R. Brennan, Rodney L. Miller, and Kirk A. Mathews, "Split-Cell Exponential Characteristic Transport Method for Unstructured Tetrahedral Meshes," *Nucl. Sci. Eng.*, **138**, 26-44 (2001).

[77] Y. Y. Azmy and D. A. Barnett, "Arbitrarily High Order Transport Method of the Characteristic Type for Tetrahedral Grids," *Proceedings of M&C 2001 Topical Meeting*, Salt Lake City, Utah, (2001).

[78] D.Y. Anistratov & E. W. Larsen, "Nonlinear and Linear α-Weighted Methods for Particle Transport Problems," *J. of Comp. Phys.*, **173**, 664-684 (2001).

[79] I.K. Abu-Shumays, "Angular Quadratures for Improved Transport Computations," *Transport Theory Statist. Phys.*, **30**, 169 (2001).

[80] M.L. Adams, "Discontinuous Finite Element Transport Solutions in Thick Diffusive Problems," *Nucl. Sci. Eng.*, **137**, 298-333 (2001).

[81] C.J. Park and N.Z. Cho, "A Linear Multiple Balance Method with High Order Accuracy for Discrete Ordinates Neutron Transport Equations," *Ann. Nucl. Energy*, **28**, 1499 (2001).

[82] B. Su, "Variable Eddington factors and flux limiters in radiative transfer," *Nucl. Sci. Eng.*, **137**, pp.281-297 (2001).

[83] B.W. Patton, J.P. Holloway, "Application of Preconditioned GMRES to the Numerical Solution of the Neutron Transport Equation," *Annals of Nuclear Energy*, **29**, 109 (2002).

[84] S.D. Pautz, "An algorithm for Parallel S_N Sweeps on Unstructured Meshes," *Nucl. Sci. Eng.*, **140**, pp.111-136, 2002.

[85] M.Benzi, "Preconditioning Techniques for Large Linear Systems: A Survey," *J. of Comp. Phys.* **182**, 418477 (2002).

[86] Rampp, M. and Janka, H.-T., Radiation hydrodynamics with neutrinos: Variable Eddington factor method for core-collapse supernova simulations, *Astronomy and Astrophysics*, **396**, 361-392 (2002).

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

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

[89] J. E. Morel, "A 3-D Cell-Centered Diffusion Discretization for Arbitrary Polyhedral Meshes," LANL Research Note, CCS-4:02-40(U) (2002).

[90] Raytcho D. Lazarov and Panayot S. Vassilevski, "Numerical Methods for Convection Diffusion Problems on General Grids," *Approximation Theory*, Darba, Sofia, pp. 258-283 (2002).

[91] Y. Saad, **Iterative Methods for Sparse Linear Systems**, Second Edition, SIAM (2003).

[92] J.S. Warsa and T.A. Wareing and J.E. Morel, "Solution of the discontinuous P_1 equations in two-dimensional Cartesian geometry with two-level preconditioning," *SIAM J. Sci. Comput.*, Vol. 24, No. 6, pp. 20932124 (2003).

[93] James S. Warsa, Todd A. Wareing, Jim E. Morel, "On the Degraded Effectiveness of Diffusion Synthetic Acceleration For Multidimensional S_N Calculations in the Presence of Material Discontinuities," Proceedings of *M&C 2003, International Conference on Mathematics and Computations,* American Nuclear Society Topical Meeting of M&C Division, Gatlinburg, Tennessee, April, 2003, 17 pp.

[94] Hiromi G. Stone and Marvin L. Adams, "A Piecewise Linear Finite Element Basis with Application to Particle Transport," Proceedings of *M&C 2003, International Conference on Mathematics and Computations,* American Nuclear Society Topical Meeting of M&C Division, Gatlinburg, Tennessee, April, 2003, 14 pp.

[95] H. Hiruta, Advanced Computational Methodology for Full-Core Neutronics Calculations, Ph.D. dissertation, North Carolina State University, Raleigh, NC, USA (2004).

[96] J.S. Warsa, T.A. Wareing, and J. E. Morel, "Krylov Iterative Methods and the Degraded Effectiveness of Diffusion Synthetic Acceleration for Multidimensional S_N Calculations in Problems with Material Discontinuities," *Nucl. Sci. Eng.*, **147**, 218-248 (2004).

[97] K. Lipnikov, J. Morel, & M. Shashkov, Mimetic finite difference methods for diffusion equations on non-orthogonal non-conformal meshes, *J. of Comp. Phys.*, **199**, 589-597 (2004).

[98] John I. Castor, **Radiation Hydrodynamics**, Cambridge University Press, Cambridge, 2004.

[99] L. Roberts & D.Y. Anistratov, "Nonlinear Weighted Flux Methods for Particle Transport Problems," Proceedings of *M&C 2005, International Conference on Mathematics and Computations, Supercomputing, Reactor Physics and Nuclear Biological Applications,* American Nuclear Society Topical Meeting of M&C Division, Avignon, France, September, 2005, 10 pp.

[100] R.E. Grove, "The Slice Balance Approach (SBA): A Characteristic-based, Multiple Balance S_N Approach on Unstructured Polyhedral Meshes," Proceedings of *M&C 2005, International Conference on Mathematics and Computations, Supercomputing, Reactor Physics and Nuclear Biological Applications,* American Nuclear Society Topical Meeting of M&C Division, Avignon, France, September, 2005, 12 pp.

[101] X.S. Li, "An overview of SuperLU: Algorithms, implementation, and user interface," *ACM Trans. Math. Softw.*, **31**, 3, pp.302–325 (2005).

[102] J.E. Morel, "Basic Krylov Methods with Application to Transport," Proceedings of *M&C 2005, International Conference on Mathematics and Computations, Supercomputing, Reactor Physics and Nuclear Biological Applications*, Avignon, France, September, 2005, American Nuclear Society Topical Meeting of M&C Division (2005).

[103] H. Hiruta and D.Y. Anistratov, "Homogenization Method for the Two-Dimensional Low-Order Quasi-Diffusion Equations for Reactor Core Calculations," *Nucl. Sci. Eng.*, **154**, pp. 328352 (2006).

[104] E.N. Aristova, D.F. Baydin, V.Ya. Gol'din, "Two Variants of Economical Method for Solving the Transport Equation in R-Z Geometry on the Basis of the Transition to Vladimirov's Variables," *Mathematical Modelling*, **v12**, No. 7, 43-52 (2006) (in Russian).

[105] W. A. Wieselquist and D. Y. Anistratov, "The Quasidiffusion Method for 2D Transport Problems on AMR Grids," *Trans. Am. Nucl. Soc.*, **96**, 472-474 (2007).

[106] W. A. Wieselquist and D. Y. Anistratov, "The Quasidiffusion Method for Transport Problems

in 2D Cartesian Geometry on Grids Composed Of Arbitrary Quadrilaterals," *Trans. Am. Nucl. Soc.*, **97**, 475-478 (2007).

[107] E.W. Larsen and A.B. Wollaber, "A Quantitative Theory of Angular Truncation Errors in Three-Dimensional S_N Calculations," *Nucl. Sci. Eng.*, **160**, 267283 (2008).

[108] James S. Warsa, "A Continuous Finite Element-Based Discontinuous Finite Element Method for S_N Transport," *Nucl. Sci. Eng.*, **160**, 385-400 (2008).

[109] R. Baker, Los Alamos National Laboratory, personal communication, 2008.

[110] W.A. Wieseqluist and D.Y. Anistratov, "A Quasidiffusion Method for Unstructured Quadrilateral Meshes in 2D XY Geometry," Proceedings of *M&C 2009, International Conference on Mathematics and Computations, and Reactor Physics,* American Nuclear Society Topical Meeting of M&C Division, Saratoga Springs, New York, May 3-7, 2009, (to appear).

[111] SCALE (Standardized Computer Analyses for Licensing Evaluation, http://www.ornl.gov/sci/scale, (1 Jan. 2009).

APPENDICES

Appendix A

Derivation of a Subcell Balance Characteristic Method2

A.1 Polynomial Exponential Moments

The basic goal of using polynomial exponential moments, $\mathcal{PM}$, is to ease construction of subcell characteristic transport schemes by abstracting the calculation of difficult integrals. All $\mathcal{PM}$ are smooth functions of their arguments and bounded by $[0, 1]$. For example, they allow voids ($\sigma = 0$) to be treated implicitly. Even more importantly, as the grid is refined, one will encounter many subcells with small dimensions in the direction of particle travel and if the case is not handled appropriately, one may see decreased accuracy as roundoff errors overwhelm the solution, or even lack of convergence in the extreme case.

A.1.1 1D Polynomial Exponential Moments

To derive the $\mathcal{PM}$, let us consider the 1D polynomial exponential moment function, defined as

$$\mathcal{PM}_n(x) \stackrel{\text{def}}{=} \int_0^1 dt\, t^n\, e^{-xt}. \tag{A.1}$$

The parameter x is an optical thickness of sorts, usually something like $x = \frac{\sigma \Delta s}{\sin\theta}$. Note that these polynomial exponential moment functions closely resemble Miller and Mathews exponential moments functions [69],

$$\mathcal{M}_n(x) = \int_0^1 dt\, (1-t)^n\, e^{-xt}.$$

Much of this work is based on their descriptions, however we find $\mathcal{PM}$ more basic—one can represent $\mathcal{M}_n(x)$ as a sum of $\mathcal{PM}_n(x)$ but the converse is more difficult. Note that the polynomial exponential moments are also closely related to the gamma function, Γ. The $\mathcal{PM}$ integral has a standard recursive solution from integration by parts [60],

Standard Forward Recursion, stable for $x \geq n$

$$\mathcal{PM}_n(x) = \begin{cases} \frac{1-e^{-x}}{x} & n = 0, \\ \frac{n\mathcal{PM}_{n-1}(x) - e^{-x}}{x} & n > 0. \end{cases} \tag{A.2}$$

However, such a forward recursion is only stable for $x \geq n$, otherwise one will see the leading n term grow without bound. As long as one is interested in lower-order moments, $n < 5$ or so, one can probably tolerate the introduction of error using the forward recursion. However, when doing multi-dimensional polynomial moments, or considering curved faces of subcells, one could

easily need moments of $n \geq 5$ for a second- or third- order method. The solution, as shown in [69] is to use backward recursion starting with the highest moment needed N,

Standard Backward Recursion, stable for $x \leq n+1$

$$\mathcal{PM}_n(x) = \begin{cases} \frac{\Gamma(1+n\,,0\,,x)}{x^{n+1}} & n = N, \\ \frac{x\mathcal{PM}_{n+1}(x)+e^{-x}}{n+1} & n < N, \end{cases} \tag{A.3}$$

where $\Gamma(a, z_0, z_1)$ is the generalized incomplete gamma function. Note that the stability regions of forward and backward recursion overlap. Because forward recursion is cheaper we prefer to use it in the region of overlap. Finally, in the case of small x, neither solution is appropriate as they have x in denominators. Thus we must write a series expansion for $x < 1$,

$$e^{-xt} = \sum_{k=0}^{\infty} \frac{(-xt)^k}{k!},$$

substitution into the integral expression yields,

Near Zero Expansion, stable for $x < 1$

$$\mathcal{PM}_n(x) = \sum_{k=0}^{\infty} \frac{1}{k+n+1} \frac{(-x)^k}{k!}. \tag{A.4}$$

This is an alternating sign series and may be truncated safely—the sum of all truncated terms is bounded by the absolute value of the last term used. Thus based on n and x, we choose

the evaluation method for the polynomial exponential moment: 1) Standard Forward Recursion if $x \geq n$ and $x \geq x_{min}$ 2) Standard Backward Recursion $x_{min} \leq x \leq n$ or 3) Near Zero Expansion if $x < x_{min}$. For stability, $x_{min} < 1$, however it is advantageous to choose x_{min} a bit smaller so that we require fewer terms to converge our series. For example, the choice of $x_{min} = 1/2$ leads to a worst case scenario of $\frac{2^{-k}}{(k+1)!}$—an error magnitude of 10^{-16} requires 14 terms.

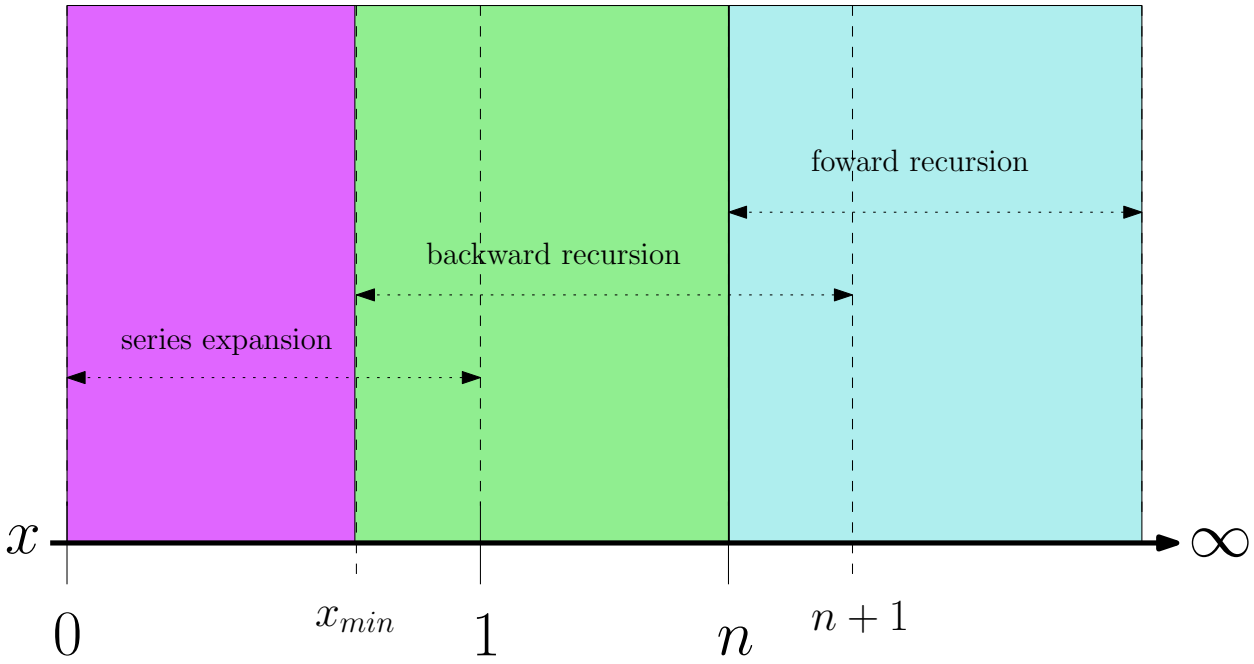


Figure A.1: Evaluation methods for the exponential moment functions, $\mathcal{PM}_n(x)$. Colored blocks indicate the method used. Arrows indicate the full stability regions.

A.1.2 2D Polynomial Exponential Moments

Multi-dimensional polynomial moments build on lower dimensional ones. In 2D, we have

$$\mathcal{PM}_{mn}(x, B(t)) \stackrel{\text{def}}{=} \int_0^1 dt\, t^m \int_0^{B(t)} ds\, s^n\, e^{-xs}. \tag{A.5}$$

The upper limit of the inner s integral is allowed to depend on t— we consider only linear functions but it is fairly straightforward to extend to any polynomial of t. The upper limit is actually required to be

$$B(t) = B_1 + B_t t, \quad B_1, B_t \geq 0, \quad B(t) \leq 1 \; \forall t.$$

To arrive at expressions for $\mathcal{PM}_{mn}(x, B)$, the basic concept is as follows: 1) derive a recursive relationship for the innermost integral for non-small x, 2) evaluate the needed powers of $B(t)$ and then 3) apply definitions of $\mathcal{PM}_m$ to the remaining terms. Finally, a near-zero expansion case must be derived in the case that x is small.

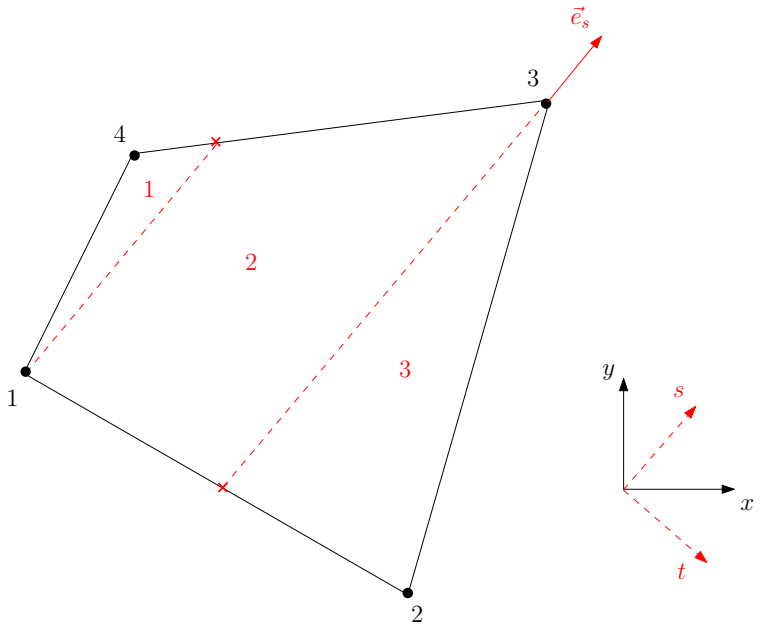


Figure A.2: A quadrilateral divided into 3 subcells.

A.2 Subcell Balances

The characteristic equation for an angular dependent subcell ("slice") in 2D XY geometry is

$$\psi(s,t) = \psi(s_{min}(t),t)e^{-\sigma_t[s-s_{min}(t)]/\sin\theta} + \frac{1}{\sin\theta}\int_{s_{min}(t)}^{s} ds'\; q(s',t)\; e^{-\sigma_t[s-s']/\sin\theta}\;, \qquad \text{(A.6)}$$

$$\text{for } t \in [t_{min}, t_{max}]\;,$$

$$\text{for } s \in [s_{min}(t), s_{max}(t)]\;.$$

given in terms of the (s,t) coordinates. The s-axis is positive in the direction $\vec{\Omega} = (\Omega_x, \Omega_y, \Omega_z)^t$, projected onto the xy-plane. The t-axis is a clockwise rotation of the s-axis.

$$\vec{e}_s = (\Omega_x, \Omega_y)^t \;\sin\theta$$

$$\vec{e}_t = (\Omega_y, -\Omega_x)^t \;\sin\theta$$

In each slice, t ranges from t_{min} to t_{max} but the incoming and outgoing faces are not necessarily aligned with the t-axis—that is $s_{min} = s_{min}(t)$ and $s_{max} = s_{max}(t)$. We know incoming values given at $s_{min}(t)$,

$$\psi_{in}(t) \stackrel{\text{def}}{=} \psi(s_{min}(t), t).$$

A.2.1 Transformation of Integral Bounds

Let the equation for a subcell consist of two components, $\psi(s,t) = \psi^E(s,t) + \psi^S(s,t)$, corresponding to the exponential term and the source term [52], respectively, where

$$\psi^E(s,t) = \psi(s_{min}(t),t)e^{-\sigma_t[s-s_{min}(t)]/\sin\theta}, \tag{A.7}$$

$$\psi^S(s,t) = \frac{1}{\sin\theta}\int_{s_{min}(t)}^{s} ds'\; q(s',t)\; e^{-\sigma_t[s-s']/\sin\theta}\;. \tag{A.8}$$

We require application of the following integrations to our E- and S-components:

$$\bar{\psi}_{out} = \frac{1}{\Delta t}\int_{t_1}^{t_2} dt\; \psi(s_{max}(t),t), \tag{A.9}$$

$$\bar{\bar{\psi}} = \frac{1}{\Delta\bar{s}\Delta t}\int_{t_1}^{t_2} dt\; \int_{s_{min}(t)}^{s_{max}(t)} ds\; \psi(s,t), \tag{A.10}$$

where $\Delta\bar{s} = (\Delta s_1 + \Delta s_2)/2$. For the contributions to the outgoing, face-average angular flux, we get integrals of the form

$$\bar{\psi}^E_{out} = \frac{1}{\Delta t}\int_{t_1}^{t_2} dt\; \psi_{in}(t)e^{-\frac{\sigma_t}{\sin\theta}[s_{max}(t)-s_{min}(t)]}, \tag{A.11}$$

$$\bar{\psi}^S_{out} = \frac{1}{\Delta t}\int_{t_1}^{t_2} dt\; \frac{1}{\sin\theta}\int_{s_{min}(t)}^{s_{max}(t)} ds\; q(s,t)\; e^{-\frac{\sigma_t}{\sin\theta}[s_{max}(t)-s]}\;. \tag{A.12}$$

For the contributions to the interior, cell-average angular flux, we get integrals of the form

$$\bar{\bar{\psi}}^E = \frac{1}{\Delta\bar{s}\Delta t}\int_{t_1}^{t_2} dt\; \int_{s_{min}(t)}^{s_{max}(t)} ds\; \psi_{in}(t)e^{-\frac{\sigma_t}{\sin\theta}[s-s_{min}(t)]}, \tag{A.13}$$

$$\bar{\bar{\psi}}^S = \frac{1}{\Delta\bar{s}\Delta t}\int_{t_1}^{t_2} dt\; \int_{s_{min}(t)}^{s_{max}(t)} ds\; \frac{1}{\sin\theta}\int_{s_{min}(t)}^{s} ds'\; q(s',t)\; e^{-\frac{\sigma_t}{\sin\theta}[s-s']}\;. \tag{A.14}$$

In order to use the polynomial exponential moment ($\mathcal{PM}$) functions, which are defined for *all* $x \geq 0$ of $\exp[-x]$, we transform our integrals according to

$$\int_{t_1}^{t_2} \bullet \, dt \rightarrow \int_0^1 \bullet \, d\tilde{t}$$

and

$$\int_{s_{min}(t)}^{s_{max}(t)} \bullet \, ds \rightarrow \int_0^{B(\tilde{t})} \bullet \, d\tilde{s}.$$

Also, for convergence of $\mathcal{PM}$ for small arguments of x, it is required that both B_1 and B_t be positive, $0 \leq B_1 \leq 1$ and $0 \leq B_t \leq 1$. This leads to two cases, based on the length of the left and right sides of the subcell: one in which $\Delta s_2 \geq \Delta s_1$, and one for the converse. We present only the former below.

A.2.2 Subcell Case 1, $\Delta s_2 \geq \Delta s_1$

First we concern ourselves with 1D terms that depend only on t, then we consider 2D terms.

A.2.2.1 1D Terms

We apply the following transformation: $\tilde{t} = (t - t_1)/\Delta t$, which leads to the outgoing E-component contribution, $\bar{\psi}^E_{out}$ of Eq. (A.11),

$$\bar{\psi}^E_{out} \;\; = \;\; \int_0^1 \, d\tilde{t} \; \psi_{in}(t) \;\; e^{-x \, B(\tilde{t})},$$

where $x = \sigma_t \Delta s_2 / \sin\theta$ and $\psi_{in}(t) = at^2 + bt + c$. The function $B(\tilde{t})$ is given by

$$B(\tilde{t}) = \frac{\Delta s(\tilde{t})}{\Delta s_2} = B_1 + B_t \tilde{t} \tag{A.15}$$

$$B_1 = \frac{\Delta s_1}{\Delta s_2}, \tag{A.16}$$

$$B_t = \frac{\Delta s_2 - \Delta s_1}{\Delta s_2}, \tag{A.17}$$

where we have used the distance between incoming and outgoing edges

$$\Delta s(\tilde{t}) = \Delta s_1 + (\Delta s_2 - \Delta s_1)\tilde{t},$$

given now in terms of the integration parameter $\tilde{t} \in [0, 1]$. Therefore, at the left side, $\tilde{t} = 0$ and the distance is Δs_1. At the right side, $\tilde{t} = 1$ and the distance is Δs_2. The distance $\Delta s(\tilde{t})$ varies linearly because the edges are linear. Because $\Delta s_2 \geq \Delta s_1$, $0 \leq B_t \leq 1$ and $0 \leq B_1 \leq 1$—this means these transformations will lead to stable expressions for polynomial moments of the exponential for all x.

Finally, substituting $t = t_1 + \Delta t\ \tilde{t}$ into our polynomial, $\psi_{in}(t_1 + \Delta t\ \tilde{t}) = a[t_1 + \Delta t\ \tilde{t}]^2 + b[t_1 + \Delta t\ \tilde{t}] + c$, we fully specify the E-component of the outgoing angular flux,

$$\bar{\psi}^E_{out} = e^{-xB_1}\Big(\begin{array}{ll} [at_1^2 + b\,t_1 + c] & \mathcal{PM}_0(xB_t) + \\ [2a\Delta t\,t_1 + b\Delta t] & \mathcal{PM}_1(xB_t) + \\ [a\Delta t^2] & \mathcal{PM}_2(xB_t) \end{array} \Big). \tag{A.18}$$

Now, for the S-component of the outgoing angular flux, $\bar{\psi}^S_{out}$ of Eq. (A.12), we choose the same transformation of t and the following for s: $\tilde{s} = (s_{max}(\tilde{t}) - s)/\Delta s_2$, where

$$s_{max}(\tilde{t}) = s_{max,1} + \Delta s_{max}\tilde{t},$$

and $\Delta s_{max} = s_{max,2} - s_{max,1}$. This leads to

$$\bar{\psi}^S_{out} \;=\; \frac{\Delta s_2}{\Delta \bar{s}}\, C \int_0^1 d\tilde{t} \int_0^{B(\tilde{t})} d\tilde{s}\; q(s,t)\; e^{-x\;\tilde{s}}\;,$$

where $C = \frac{1}{\sin\theta}$ and $q(s,t) = q_1 + q_s s + q_t t$. Now substituting $t = t_1 + \Delta t\;\tilde{t}$ and $s = s_{max}(\tilde{t}) - \Delta s_2 \tilde{s}$ into our polynomial, $q(s_{max}(\tilde{t}) - \Delta s_2 \tilde{s}\;,\; t_1 + \Delta t\;\tilde{t}) = q_1 + q_s[s_{max}(\tilde{t}) - \Delta s_2 \tilde{s}] + q_t[t_1 + \Delta t\;\tilde{t}]$, we determine the S-component of the outgoing average angular flux as

$$\bar{\psi}^S_{out} = \frac{2}{1+\frac{\Delta s_1}{\Delta s_2}}\Big(\begin{array}{lll} [q_1 + q_s s_{max,1} + q_t t_1] & \mathcal{PM}_{00}(x, B(\tilde{t}); x_0, C) + \\ [q_s \Delta s_{max} + q_t \Delta t] & \mathcal{PM}_{10}(x, B(\tilde{t}); x_0, C) - \\ [q_s \Delta s_2] & \mathcal{PM}_{01}(x, B(\tilde{t}); x_0, C) \end{array} \Big), \tag{A.19}$$

where $C = \frac{1}{\sin\theta}$ and $x_0 = x/C = \sigma_t \Delta s_2$ are additional parameters that must be passed to the polynomial exponential moment evaluation routines to properly handle $\sin\theta \to 0$.

A.2.2.2 2D Terms

Now we turn to the more challenging case of contributions of E- and S-components to the cell-average angular flux. We apply the following transformation: $\tilde{t} = (t - t_1)/\Delta t$ and $\tilde{s} = (s - s_{min}(\tilde{t}))/\Delta s_2$, which leads to the interior, cell-average E-component contribution, $\bar{\bar{\psi}}^E$ of Eq. (A.13),

$$\bar{\bar{\psi}}^E \;=\; \frac{\Delta s_2}{\Delta \bar{s}} \int_0^1 d\tilde{t}\; \psi_{in}(t) \int_0^{B(\tilde{t})} d\tilde{s}\; e^{-x\;\tilde{s}}.$$

Substituting $\psi_{in}(t)$, we arrive at the following expression for the contribution of the E-component to the cell-average angular flux,

$$\bar{\bar{\psi}}^E = \frac{2}{1+\frac{\Delta s_1}{\Delta s_2}}\Big(\quad [at_1^2 + bt_1 + c] \quad \mathcal{PM}_{00}(x, B(\tilde{t}); x_0, C) + \\ [2a\Delta t\, t_1 + b\Delta t] \quad \mathcal{PM}_{10}(x, B(\tilde{t}); x_0, C) + \\ [a\Delta t^2] \qquad\quad \mathcal{PM}_{20}(x, B(\tilde{t}); x_0, C) \quad \Big). \tag{A.20}$$

Now, using our above transformations in Eq. (A.14), $\tilde{t} = (t - t_1)/\Delta t$ and $\tilde{s} = (s - s_{min}(\tilde{t}))/\Delta s_2$, along with $\tilde{s}' = \tilde{s} + (s_{min}(\tilde{t}) - s')/\Delta s_2$, we get the following expression for the interior, cell-average S-component contribution, $\bar{\bar{\psi}}^S$ of Eq. (A.14),

$$\bar{\bar{\psi}}^S \;=\; \frac{\Delta s_2^2}{\Delta \bar{s}} C \int_0^1 d\tilde{t} \int_0^{B(\tilde{t})} d\tilde{s} \int_0^{\tilde{s}} d\tilde{s}'\; q(s', t)\; e^{-x\,\tilde{s}'}, \tag{A.21}$$

where the source has been written untransformed as $q(s', t)$, for brevity—the expressions for s' and t are $s' = \tilde{s}\Delta s_2 + s_{min}(\tilde{t}) - \tilde{s}'\Delta s_2$ and $t = t_1 + \tilde{t}\,\Delta t$. Solving the above Eq. (A.21) in terms of general exponential moments requires three dimensional moments functions, $\mathcal{PM}_{\ell mn}$, whereas all other moments have required two- and one- dimensional ones, $\mathcal{PM}_{mn}$ and $\mathcal{PM}_n$. Thus we choose to solve the innermost integral over $\tilde{s}'$, with appropriate series expansion for small x and express the result in terms of $\mathcal{PM}_{mn}$ and $\mathcal{PM}_n$. It is also convenient now to introduce the polynomial moments $\mathcal{P}_{mn}$,

$$\begin{aligned} \mathcal{P}_{mn} &= \int_0^1 d\tilde{t}\,\tilde{t}^m \int_0^{B(\tilde{t})} d\tilde{s}\tilde{s}^n \\ &= \frac{1}{n+1} \sum_{k=0}^{n+1} \frac{(n+1)!}{(n+1-k)!\,k!} \frac{B_t^k B_1^{n+1-k}}{k+m+1}. \end{aligned} \tag{A.22}$$

Finally, we consider the constant, linear in s, and linear in t parts of the integrand of Eq. (A.21), q, separately. Thus we will present the $\bar{\bar{\psi}}^S$ in parts due to each of the components of the linear source,

$$\bar{\bar{\psi}}^S = q_1 \bar{\bar{\psi}}_1^S + q_s \bar{\bar{\psi}}_s^S + q_t \bar{\bar{\psi}}_t^S. \tag{A.23}$$

A.2.2.3 2D Terms with x not too Small

For the case that x not small, we have the following expression for the constant q_1 part,

$$\bar{\bar{\psi}}_1^S = \frac{q_1}{\sigma_t}\Big(1 - f_2 \mathcal{PM}_{00}(x, B(\tilde{t})) \quad \Big), \tag{A.24}$$

where f_2 is a factor that comes up often, defined as the ratio $\Delta s_2 / \Delta \bar{s}$, or

$$f_2 = \frac{\Delta s_2}{\Delta \bar{s}} = \frac{2}{1 + \frac{\Delta s_1}{\Delta s_2}}.$$

Note that in Eq. (A.24) the exponential moment $\mathcal{PM}_{00}(x, B(\tilde{t}))$, the x_0 and C terms are not needed—this is because the $\frac{1}{\sin\theta}$ term cancels after the evaluation of the inner $\tilde{s}'$ integral, so $C = 1$ and $x_0 = x$. This equation is shown to fulfill a simple asymptotic: in a homogeneous, infinite media, as $x \to \infty$, $\mathcal{PM}_{00}(x, B(\tilde{t})) = 0$, which leads to $\psi(s,t) = \frac{q_1}{\sigma_t}$. The q_t part is also easy to evaluate,

$$\begin{aligned} \bar{\bar{\psi}}_t^S = \tfrac{q_t}{\sigma_t} f_2 \Big(& \quad [B_1 t_1 + B_1 \Delta t/2 + B_t t_1/2 + \Delta t\, B_t/3] - \\ & [t_1] \quad \mathcal{PM}_{00}(x, B(\tilde{t})) - \\ & [\Delta t] \quad \mathcal{PM}_{10}(x, B(\tilde{t})) \quad \Big). \end{aligned} \tag{A.25}$$

The algebra for the q_s part is very tedious, as a result, we have

$$\bar{\bar{\psi}}_s^S = \tfrac{q_s}{\sigma_t} f_2 \Big(\begin{array}{ll} [s_{min,1}] & \big[\mathcal{P}_{00} - \mathcal{PM}_{00}(x, B(\tilde{t}))\big] + \\ [s_{min,t}] & \big[\mathcal{P}_{10} - \mathcal{PM}_{10}(x, B(\tilde{t}))\big] + \\ [\Delta s_2] & \mathcal{P}_{01} - \\ [\tfrac{\sin\theta}{\sigma_t}] & \big[\mathcal{P}_{00} - \mathcal{PM}_{00}(x, B(\tilde{t}))\big] \end{array} \Big). \tag{A.26}$$

Note that, surprisingly, this integral expression does not contain $\mathcal{PM}_{01}$, which comes from integrations over s. In fact it does, but there are two terms which exactly cancel.

A.2.2.4 2D Terms with Small x

For the small x case, substituting the expansion

$$\exp(-x\,\tilde{s}') = 1 - x\tilde{s}' + \frac{(x\tilde{s}')^2}{2} - ... = \sum_{m=0}^{\infty} \frac{(-x\tilde{s}')^m}{m!}$$

into Eq. (A.21) leads to a different class of expressions that never experience division by x. Because of the form of the inner integral in Eq. (A.21), general moments of $\tilde{t}^i$ and $\tilde{s}^j$ require application of the multinomial theorem, which is unwieldy. Thus we choose to simply evaluate our expressions for the 3 cases we need. The close-to-zero approximations are denoted with a triple-asterisk.

$$\bar{\bar{\psi}}_1^{S***} = q_1 \, \frac{\Delta s_2}{\sin\theta} f_2 \, \sum_{m=0}^{\infty} (-x)^m \, C_1(m+2) \tag{A.27}$$

where

$$C_1(N) = \sum_{\ell=0}^{N} \frac{(-B_t)^\ell}{(N-\ell)!\,\ell!} \, \left[\frac{1}{\ell+1}\right].$$

Note, the expression above only depends on B_t because we used the fact that $B_1 = 1 - B_t$ before expanding a binomial expression. The $\bar{\bar{\psi}}_t^{S***}$ term is easy enough, it introduces a $(t_1 + \Delta t\tilde{t})$ term into the integral, which leads to the expression

$$\bar{\bar{\psi}}_t^{S***} = q_t \, \frac{\Delta s_2}{\sin\theta} f_2 \, \sum_{m=0}^{\infty} (-x)^m \, C_t(m+2) \tag{A.28}$$

where

$$C_t(N) = \sum_{\ell=0}^{N} \frac{(-B_t)^\ell}{(N-\ell)!\,\ell!} \, \left[\frac{t_1 + t_1\ell + t_2}{(\ell+1)(\ell+2)} \right].$$

The $\bar{\bar{\psi}}_s^{S***}$ term involves a lot of algebra, it introduces a $(t_1 + \Delta t\tilde{t})$ term into the integral, which leads to the expression

$$\bar{\bar{\psi}}_s^{S***} = q_s \, \frac{\Delta s_2}{\sin\theta} f_2 \, \sum_{m=0}^{\infty} (-x)^m \, C_s(m+2) \tag{A.29}$$

where

$$C_s(N) = \sum_{\ell=0}^{N} (-B_t)^\ell \frac{1}{(N-\ell)!\,\ell!} \, \left[\frac{s_{min,1} + \ell s_{min,1} + s_{min,2}}{(\ell+1)(\ell+2)} \right] + \sum_{\ell=0}^{N+1} (-B_t)^\ell \frac{1}{(N+1-\ell)!\,\ell!} \, \left[\frac{\Delta s_2}{\ell+1} \right].$$

In Fig. A.3 we see the convergence of a particular small $x = \frac{\sigma \Delta s}{\sin\theta}$ expansion as a function of σ. It shows the series converges as predicted and shows motivation for choosing the maximum x for which the series will be used, x_{min}, somewhat lower than the theoretically allowed $x_{min} = 0.99999999999...$, to allow for a more rapidly convergent series.

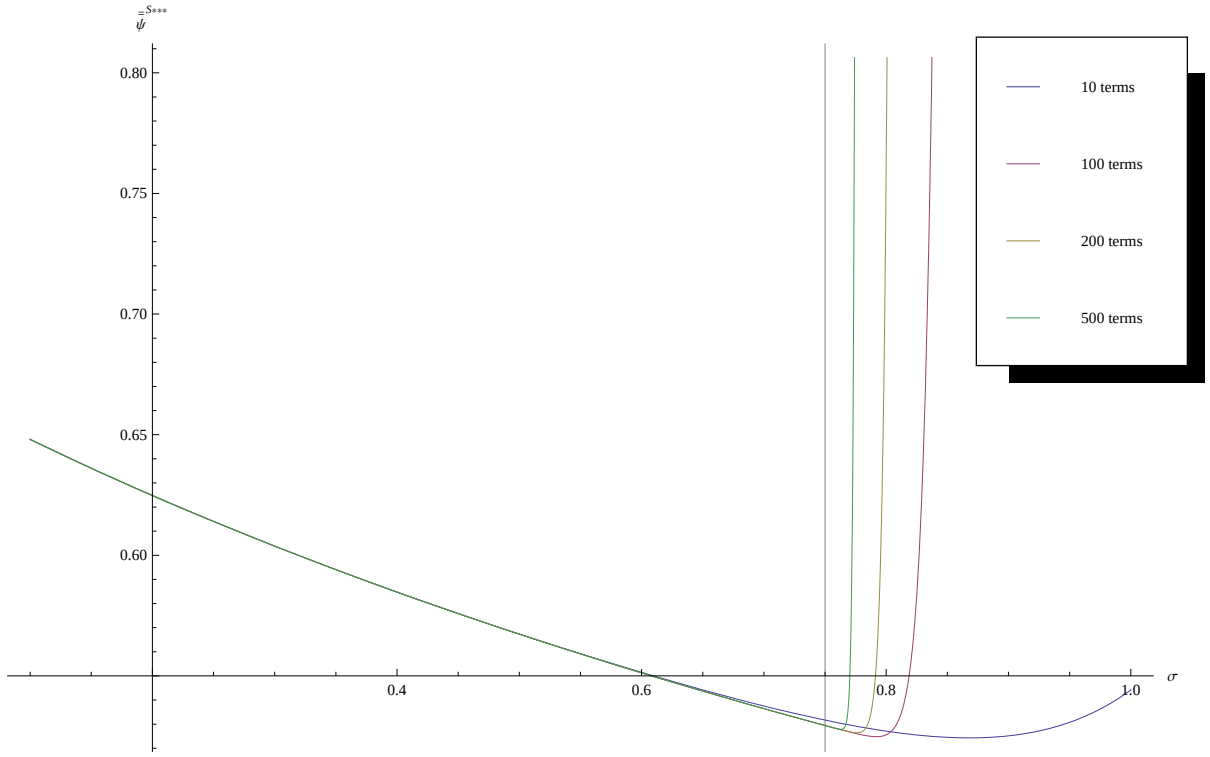


Figure A.3: Using the small x expansion, values of $\bar{\bar{\psi}}^{S***}$ are shown as a function of total cross section, σ, for a varying number of terms in the expansion. For the case shown, $\sin\theta$ and Δs were such that requiring $x = \frac{\Delta s \sigma}{\sin\theta} < 1$, leads to $\sigma < \frac{3}{4}$. With ten terms, one can see the expansion is fairly accurate around $\sigma = \frac{3}{4}$ (a vertical gray line) and may appear convergent beyond that. However, as more terms are included, one observes the *divergence* for $\sigma > \frac{3}{4}$.

Appendix B

Parabolic Interpolant with Monotonization

In the case of no extremum, the incoming flux is approximated as a quadratic on the face $\psi(x) = ax^2 + bx + c$, $x \in [0, h]$. Error of such interpolating polynomials can be given in Lagrangian form as

$$R(x) = \psi(x) - \psi^{ex}(x) = \left.\frac{d^{n+1}\psi^{ex}}{dx^{n+1}}\right|_{\xi} \prod_{i=0}^{n} \frac{(x - x_i)}{(n+1)!},$$

where x_i are the interpolation nodes and $\xi \in [0, h]$. The Lagrangian form is useful because it contains no higher-order derivatives—the disadvantage of using it is the value ξ is not known, but it is very useful for showing the order of local truncation error. In our case, the parabola is constructed from

- left and right point values,

$$\psi(0) = \psi_1,$$

$$\psi(h) = \psi_2,$$

- and average value,

$$\frac{1}{h}\int_0^h \psi(x)dx = \bar{\psi},$$

known from previous cells in the sweep or from boundary conditions. The parabolic coefficients are thus given by

$$a = \frac{3\psi_1 + 3\psi_2 - 6\bar{\psi}}{h^2}, \tag{B.1}$$

$$b = \frac{6\bar{\psi} - 4\psi_1 - 2\psi_2}{h}, \tag{B.2}$$

$$c = \psi_1. \tag{B.3}$$

If ψ_1, ψ_2, and $\bar{\psi}$ are known from previous cells in the sweep, there is error associated with those terms—otherwise, if given by boundary conditions, the terms may be considered exact. In order that the Lagrangian error apply, we must assume the 3 interpolation nodes, $x_0 = 0$, $x_1 = x(\bar{\psi})$, $x_2 = h$. For linear functions, $x_1 = \frac{h}{2}$. For quadratic functions, we choose to represent $x_1 = \eta\frac{h}{2}$ with $\eta \in [0, 2]$ by mean-value theorem and known monotonicity of the exact solution.

B.1 Local Truncation Error in Point Values

Now we consider the error in the incoming quadratic $\psi(x)$ *on the subcell*, denoted with subscript s. The incoming edge of the subcell goes from $x = a_s$ to $x = b_s$ with length $h_s = b_s - a_s$. It is also convenient to use the normalized coordinates denoted with a tilde, $\tilde{x} = x/h$. Using $\tilde{x} \in [0, 1]$ results in local error function for an incoming edge of a subcell,

$$R_s(\tilde{x}) = -\left(\frac{1}{2}Z(\eta - 2\tilde{x})(\tilde{x} - 1)\tilde{x}\right)h^3, \quad \tilde{x} \in [\tilde{a}_s, \tilde{b}_s].$$

The term Z is the derivative term from the Lagrange form of the error for a second-order polynomial (parabolic) interpolant,

$$Z = \left. \frac{d^3\psi^{ex}}{dx^3} \right|_{\xi}, \quad \xi \in [0, h].$$

Thus the point values obtained from interpolant $\psi(x)$ is $O(h^3)$ if Z is bounded on the interval. It is also exact at the endpoints, $\tilde{x} = 0$, $\tilde{x} = 1$, as well as where the function attains the average value, $\tilde{x} = \eta/2$.

B.2 Local Truncation Error in Average Values

By integration, the interpolation error in the average $\bar{R}_s = \frac{1}{h_s} \int_{a_s}^{b_s} \psi(x) - \psi^{ex}(x) dx$ is given by

$$\bar{R}_s = \frac{Z r_s^3}{12} (\eta - 1) h^3, \quad \eta \in [0, 2].$$

The term r is the ratio of the *subcell incoming edge length* to the *face edge length*, $r_s = h_s/h$. Thus the average value obtained from the interpolant is $O(h^3)$ if Z is bounded on the interval. Naturally, the average value is exact if $\psi^{ex}(x)$ is a quadratic (or lesser polynomial), due to $Z = 0$. The average value is also exact if $\eta = 1$, however, that only occurs for linear $\psi^{ex}(x)$, which will result in $Z = 0$ anyway.

B.3 Monotonization

The monotonization is based on a nonlinear selection procedure based known endpoint values ψ_1 and ψ_2, as well as average, $\bar{\psi}$. First, define the maximum and minimum values allowed, $\psi_{max} = \max(\psi_1, \psi_2)$ and $\psi_{min} = \min(\psi_1, \psi_2)$. If $\bar{\psi} > \psi_{max}$ or $\bar{\psi} < \psi_{min}$ then the solution cannot

be monotonic and we use the following representations,

$$\psi(x) = \begin{cases} \psi_{min} + 6\frac{\bar{\psi}-\psi_{min}}{h} + 6\frac{\psi_{min}-\bar{\psi}}{h^2}, & \text{if } \bar{\psi} < \psi_{min}, \\ \psi_{max} + 6\frac{\bar{\psi}-\psi_{max}}{h} + 6\frac{\psi_{max}-\bar{\psi}}{h^2}, & \text{if } \bar{\psi} > \psi_{max}. \end{cases} \tag{B.4}$$

Otherwise, we have a monotonic representation with $\psi_1 \leq \bar{\psi} \leq \psi_2$. First define the parameters,

$$\begin{aligned} x_L^* &= \frac{3\psi_2 - 3\bar{\psi}}{y_2 - y_1}h, \\ x_R^* &= \frac{2\psi_1 + \psi_2 - 3\bar{\psi}}{\psi_2 - \psi_1}h. \end{aligned} \tag{B.5}$$

If x_L^* is contained in the interval, $0 \leq x_L^* \leq h$, then we use the the piecewise representation that is constant on the right,

$$\psi(x) = \begin{cases} (\frac{\psi_1-\psi_2}{{x_L^*}^2})x^2 + (2\frac{\psi_2-\psi_1}{x_L^*})x + \psi_1 & , \quad 0 \leq x \leq x_L^*, \\ \psi_2 & , \quad x_L^* < x \leq h. \end{cases} \tag{B.6}$$

If x_R^* is contained in the interval, $0 \leq x_R^* \leq h$, then we use the piecewise representation that is constant on the left,

$$\psi(x) = \begin{cases} \psi_1 & , \quad 0 \leq x \leq x_R^*, \\ \left[\frac{\psi_2-\psi_1}{(h-x_R^*)^2}\right]x^2 + \left[2x_R^*\frac{\psi_1-\psi_2}{(h-x_R^*)^2}\right]x + \left[\psi_1 + {x_R^*}^2\frac{\psi_2-\psi_1}{(h-x_R^*)^2}\right] & , \quad x_R^* < x \leq h. \end{cases} \tag{B.7}$$

Otherwise, we do not have an extremum on the interval and we can use the simple parabolic $\psi(x) = ax^2 + bx + c$, with coefficients given in Eq. (B.1).