Discontinuous Galerkin Methods for the Linear Boltzmann Transport Equation

Thomas Radley

A thesis submitted to the University of Nottingham for the degree of Doctor of Philosophy

December 2022

Abstract

Radiation transport is an area of applied physics that is concerned with the propagation and distribution of radiative particle species such as photons and electrons within a material medium. Deterministic models of radiation transport are used in a wide range of problems including radiotherapy treatment planning, nuclear reactor design and astrophysics. The central object in many such models is the (linear) Boltzmann transport equation, a high-dimensional partial integro-differential equation describing the absorption, scattering and emission of radiation.

In this thesis, we present high-order discontinuous Galerkin finite element discretisations of the time-independent linear Boltzmann transport equation in the spatial, angular and energetic domains. Efficient implementations of the angular and energetic components of the scheme are derived, and the resulting method is shown to converge with optimal convergence rates through a number of numerical examples.

The assembly of the spatial scheme on general polytopic meshes is discussed in more detail, and an assembly algorithm based on employing quadrature-free integration is introduced. The quadrature-free assembly algorithm is benchmarked against a standard quadrature-based approach, and an analysis of the algorithm applied to a more general class of discontinuous Galerkin discretisations is performed.

In view of developing efficient linear solvers for the system of equations resulting from our discontinuous Galerkin discretisation, we exploit the variational structure of the scheme to prove convergence results and derive *a posteriori* solver error estimates for a family of iterative solvers. These *a posteriori* solver error estimators can be used alongside standard implementations of the generalised minimal residual method to guarantee that the linear solver error between the exact and approximate finite element solutions (measured in a problem-specific norm) is below a user-specified tolerance. We discuss a family of transport-based preconditioners, and our linear solver convergence results are benchmarked through a family of numerical examples.

Acknowledgements

I would first like to express my gratitude to my supervisors Matthew Hubbard and Paul Houston for their guidance, insight and kindness throughout the course of my PhD, and for encouraging and inspiring me to pursue further research. I extend my thanks to the rest of the Scientific Computation group at the School of Mathematical Sciences for their openness to discussing new research ideas.

I would like to thank Oliver Sutton and Richard Widdowson for their mentorship and companionship throughout my studies, and for always lifting my spirits when I needed it most. I also thank Fred Currell and Balder Villagomez-Bernabe for many inspiring conversations about physics and for showing me what interdisciplinary research can look like.

I would like to thank all the friends I have made during my time at the University of Nottingham for making the past few years happy and memorable.

Finally, I would like to thank my family for their endless love and support, and in particular to my parents, who have helped me through difficult times.

Contents

Chapter 1

Introduction

In the UK, around 1000 people are diagnosed with cancer every day, and of these people, around half will undergo radiotherapy as part of their treatment plan [26, 75]. Such treatments involve the delivery of photon and/or electron beams from linear particle accelerators to target areas of the patient's body. In the personalised treatment plan, it is vital that the radiative dose delivered to the tumour, as well as to healthy surrounding tissue and vital organs, is quantified accurately and precisely - previous studies suggest that a 5% error in dose estimation can change the tumour control probability by as much as 20% [76]. Additionally, such dose estimates must be quickly obtained by the clinician before the procedure takes place.

Typically, Monte Carlo (MC) approaches are employed to model the transport of individual radiative particles through the patient. The distance each particle travels between interactions with the medium, as well as the deflection angle (and associated energy loss) after each interaction, are randomly sampled using given cross-sectional data. Particle interactions with the medium may result in the liberation of an additional radiative particle whose trajectory must also be tracked. The total radiative dose delivered to the patient is then calculated as the sum of doses delivered by each particle. The use of Monte Carlo methods in hospitals is widespread and are considered the "gold standard" against which other methods for estimating dose delivery are validated [2, 10].

More recently, deterministic methods are being proposed as an alternative to Monte Carlo approaches for the simulation of radiation transport. The central object common to each of these methods is the linear Boltzmann transport equation (LBTE), a 7-dimensional partial integro-differential equation (PIDE) whose solution gives the angular fluence of radiative particles in the patient as a function of their position, direction of travel, energy and the time since the simulation was started. Once the angular fluence is known, the radiative dose delivered to the patient can be computed. Owing to the high dimensionality of the LBTE, many numerical methods have been proposed to approximate the angular fluence and the dose delivered to the patient.

One of the most popular deterministic methods for approximating the solution of spatial transport problems is the discontinuous Galerkin finite element method (DGFEM) of Reed and Hill [81]. Finite element methods seek a piecewise-polynomial approximation of the solution with respect to an underlying computational mesh whose elements typically have a simple geometry. Discontinuous Galerkin (DG) methods, on the other hand, are capable of employing meshes consisting of more general element geometries, which makes such methods suitable for problems with non-smooth data or are posed on complicated domains. DGFEMs have also been applied to the discretisation of the angular domain for the time-independent and mono-energetic LBTE [48].

By contrast, the discretisation of the energetic domain has most commonly been treated using a single standard approach known as the multigroup method [70]. After subdividing the energetic domain into a number of so-called energy groups, the multigroup methodology results in a collection of mono-energetic problems, one for each energy group. The solutions of these mono-energetic problems are then multiplied by an energy group-specific energetic function to yield an approximation of the exact angular fluence. In order for the energetic component of the solution to be well-resolved, one must either use a large number of narrow energy groups in the discretisation of the energetic domain, or perform an intermediate infinite-medium calculation to approximate the energetic dependence within each group.

The use of arbitrary (polytopic) elements in the mesh presents a challenge for the assembly of the system matrix, whose entries involve integrals over the elements and faces in the mesh. For simple element geometries, these entries are typically evaluated using numerical quadrature schemes defined on reference elements. However, general polytopic elements and faces require the construction of bespoke quadrature schemes before the system matrix can be assembled. These quadrature schemes may either use many more quadrature points and weights than is needed to achieve the desired order of accuracy, or perform an optimisation algorithm for each element and face in order to minimise the number of quadrature points used [73].

Recently, there has been interest in "quadrature-free" assembly methods [6] which, for problems with piecewise-polynomial data, perform decompositions of the element and face integrands into a linear combination of simple functions which may be integrated quickly and exactly. However, the implementation of such methods must be tailored to the problem at hand; in particular, we are not aware of applications of quadrature-free assembly methods to spatial transport problems. Moreover, while quadrature-free methods have been observed to improve assmebly times compared to standard quadrature-based techniques on polytopic meshes, no quantitative analysis has been performed that attempts to explain the performance of these algorithms on the meshes employed.

The application of DGFEMs to the numerical approximation of the (time-independent) linear Boltzmann transport equation yields a large and sparse system of

linear equations for the approximate angular flux. For such systems, direct solution methods are not feasible due to demanding memory requirements and so iterative methods are frequently preferred. These methods, which include stationary iterative methods and Krylov subspace methods, generate a sequence of approximations that converge to the true solution of the linear system. They may also contain configurable options to terminate the generation of this sequence prematurely; for example, once the error between successive iterations is deemed to be sufficiently small. A key object employed by iterative methods is a preconditioner, which can be thought of as an operator on vectors that closely approximates the action of the inverse of the original system matrix. A good choice of preconditioner is often problem-dependent and may greatly improve the convergence of an iterative method.

The analysis of iterative methods applied to the discretised linear Boltzmann transport equation has largely been conducted via Fourier analysis [1, 103]. However, the framework of finite element methods allows for an alternative approach to the study of iterative methods applied to discretised partial differential equations [63]. Firstly, one may derive computable and guaranteed *a posteriori* error estimates for the linear solver error through standard techniques in the analysis of finite element methods. Secondly, one can exploit the variational setting to define linear solvers whose convergence rates are independent of the discretisation parameters used in the prescription of the finite element space (e.g. mesh spacing or polynomial degree). Though these ideas have not to our knowledge been applied to problems in radiation transport, they are an ideal candidate for such analyses - a poorly-prescribed linear solver for the linear Boltzmann transport equation may either rapidly converge or stagnate depending on the mesh size parameter [103].

1.1 Thesis outline and contributions

The thesis is structured as follows. In Chapter 2, we first provide an extensive overview of the linear Boltzmann transport equation, as well as its numerical solution and its application to radiotherapy treatment planning. Specifically, we will discuss the various numerical methods that have been employed in the literature for the discretisation of the LBTE, with a particular focus on mesh-based deterministic methods. We shall give a brief review of the assembly of the linear system of equations arising from discontinuous Galerkin discretisations of general partial differential equations on non-standard polytopic meshes. We conclude the chapter with a discussion on the numerical solution of systems of equations arising from discretisations of the LBTE.

The main contribution of Chapter 3 is the prescription of a discontinuous Galerkin discretisation of the time-independent, poly-energetic linear Boltzmann transport equation. This method will employ DGFEM discretisations in the spatial, angular and en-

ergetic domains. We present stability and convergence results for the resulting scheme. We will then present an efficient implementation of the scheme and demonstrate the previous convergence results through poly- and mono-energetic examples. This chapter is based on work carried out in [53]; the original contributions to that work present in this thesis will be discussed in the introduction to Chapter 3.

In Chapter 4, we will give an overview of quadrature-free integration with specific applications to the exact integration of families of monomial functions on general polytopes. The fast integration of these functions will be compared against standard quadrature-based methods. Using these ideas, we propose a quadrature-free assembly method for the first-order, constant-coefficient linear transport equation. The main contribution of this chapter is a general analysis of quadrature-based and quadrature-free-based assembly methods. This analysis will illustrate the key factors that may contribute to the previously-observed accelerated assembly of the DGFEM equations using quadrature-free techniques. We conclude by presenting some examples that show that the resulting assembly method can outperform quadrature-based approaches.

In Chapter 5, we propose some iterative linear solvers, expressed using the variational framework of discontinuous Galerkin methods, and prove a number of original convergence properties and *a posteriori* solver error estimates. These solvers include an extension of the classical source iteration to the poly-energetic setting and an over-relaxed modification of source iteration in the mono-energetic setting. After a discussion of the spectral properties of the stationary-iteration operators, we turn our attention to the application of Krylov subspace-based iterative solvers. In particular, we demonstrate an approach to insert a previously-derived *a posteriori* solver error estimate into many implementations of the generalised minimal residual (GMRES) method. We present a number of transport-sweep-based preconditioners and conclude the chapter with an extensive look at a family of benchmark problems.

Chapter 2

Background

2.1 Formulation

Radiation transport is an area of applied physics that studies the transmission of energy (either in the form of particles or waves) through a material medium. More specifically, the discipline concerns itself with the distribution of particles (such as photons, neutrons and electrons) in an irradiated medium as a function of their position and energy, and their movement through the given medium. Comprehensive studies of radiation transport applied to nuclear reactor theory are given in [15, 90]; an overview of solution methods for the resulting *linear Boltzmann transport equations* (LBTEs) with application in radiotherapy treatment is given in [11]. While we do not provide a formal derivation of the LBTE, we give a brief overview of the key ideas used in its derivation.

To motivate the mathematical formulation of such phenomena, we first introduce the notion of an *angular flux* or *fluence* (corresponding to a given particle species), denoted by $u(\mathbf{x}, \boldsymbol{\mu}, E, t)$. This function describes the expected distribution of particles as a function of time t, position $\mathbf{x}$ within some physical d-dimensional domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, energy $E > 0$, and the angle/direction on the unit sphere in which particles travel $\boldsymbol{\mu} \in \mathbb{S}$, where $\mathbb{S} = S^{d-1}$ denotes the surface of the unit sphere in $\mathbb{R}^d$.

Now consider a single particle of the species under consideration, and its interaction with the matter it moves through. For simplicity, we assume that the only events that alter the trajectory of the particle are those when it interacts with matter, and *not* with other particles of the same species. There are a number of things that may happen:

- the particle may travel through the medium unimpeded;
- the particle may be deflected by matter in a different direction and with a different energy;
- the particle may be absorbed by matter or leave the system through the boundary of the domain;

- the particle may enter the system, either through the boundary of the domain or via some source located in the interior of the domain.

Many potential particle interactions are only likely/possible for different energy bands [37, 85]. The range of particle interactions becomes even broader when one considers the simulation of multiple particle species in the same medium; furthermore, the presence of nanoparticles also allows for the possibility of changing the energy bands in which different processes are dominant [37]. To mention just a few scattering processes in radiative transfer problems, we refer to [11] for descriptions of Compton, Møller and Mott scattering processes, and to [37] for descriptions of the photo-electric effect, Compton scattering, pair production and the effect of gold nanoparticles with application to medical physics. More in-depth discussions of Compton and Møller scattering are deferred until Chapter 2.1.1.

We will first state the equation satisfied by the angular flux and then highlight the key points in its derivation. To this end, we introduce the basic form of the *linear Boltzmann transport equation* (LBTE) [15, 70, 90]:

$$\begin{aligned}\frac{1}{v}\frac{\partial u}{\partial t}(\mathbf{x},\boldsymbol{\mu},E,t)+\boldsymbol{\mu}\cdot\nabla_{\mathbf{x}}u(\mathbf{x},\boldsymbol{\mu},E,t)+(\alpha(\mathbf{x},\boldsymbol{\mu},E,t)+\beta(\mathbf{x},\boldsymbol{\mu},E,t))u(\mathbf{x},\boldsymbol{\mu},E,t)\\ =\int_{\mathbb{Y}}\int_{\mathbb{S}}\theta(\mathbf{x},\boldsymbol{\mu}'\to\boldsymbol{\mu},E'\to E,t)u(\mathbf{x},\boldsymbol{\mu}',E',t)\ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}E'+f(\mathbf{x},\boldsymbol{\mu},E,t).\end{aligned} \tag{2.1}$$

Equation (2.1) is a seven-dimensional partial integro-differential equation for the angular flux $u(\mathbf{x},\boldsymbol{\mu},E,t)$. The notation is explained below:

- $\mathbf{x}$ - position in $\Omega\subset\mathbb{R}^d$;
- $\boldsymbol{\mu}$, $\boldsymbol{\mu}'$ - angles/directions in $\mathbb{S}$;
- E, E' - energies in $\mathbb{Y}=[0,\infty)$;
- t - time in $\mathbb{R}^+$;
- $u(\mathbf{x},\boldsymbol{\mu},E,t)$ - angular flux at position $\mathbf{x}$ and time t for particles with energy E travelling in direction $\boldsymbol{\mu}$;
- $v(E)$ - energy-dependent particle speed;
- $\theta(\mathbf{x},\boldsymbol{\mu}'\to\boldsymbol{\mu},E'\to E,t)$ - differential scattering cross-section of particles at position $\mathbf{x}$ and time t initially travelling in a direction $\boldsymbol{\mu}'$ with energy E' and, after interacting with the medium, scattering to a direction $\boldsymbol{\mu}$ with energy E (typically the dependence on $\boldsymbol{\mu}'$ and $\boldsymbol{\mu}$ is in the combination $\boldsymbol{\mu}'\cdot\boldsymbol{\mu}$, which is recognisable as the cosine of the angle between $\boldsymbol{\mu}'$ and $\boldsymbol{\mu}$);
- $\alpha(\mathbf{x},\boldsymbol{\mu},E,t)$ - macroscopic absorption cross-section of the medium describing the rate of removal of particles of energy E travelling in direction $\boldsymbol{\mu}$ at position $\mathbf{x}$ and time t from the system as a result of absorption processes;

- $\beta(\mathbf{x}, \boldsymbol{\mu}, E, t)$ - macroscopic scattering cross-section of the medium describing the rate of removal of particles of energy E travelling in direction $\boldsymbol{\mu}$ at position $\mathbf{x}$ and time t from the system as a result of scattering processes - this may be expressed in terms of the differential scattering cross-section as

$$\beta(\mathbf{x}, \boldsymbol{\mu}, E, t) = \int_{\mathbb{Y}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \to \boldsymbol{\mu}', E \to E', t) \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E'; \tag{2.2}$$

- $f(\mathbf{x}, \boldsymbol{\mu}, E, t)$ - external source term.

A derivation of (2.1) is given in [70]; an analogous derivation of the same equation with energetic dependencies dropped is given in [90]. The basic idea is to instead consider the time-evolution of a quantity $N(\mathbf{x}, \boldsymbol{\mu}, E, t)$, called the *particle density distribution*, where $N(\mathbf{x}, \boldsymbol{\mu}, E, t)\Delta\mathbf{x}\Delta\boldsymbol{\mu}\Delta E$ denotes the expected number of particles in a volume element $\Delta\mathbf{x}$ about $\mathbf{x}$ with energies between E and $E + \Delta E$ travelling in the cone of directions $\Delta\boldsymbol{\mu}$ about $\boldsymbol{\mu}$. It is typical to write $\Delta\mathbf{x} = \Delta\ell\Delta A$, where $\Delta\ell$ denotes the length of the volume element $\Delta\mathbf{x}$ in the direction of $\boldsymbol{\mu}$ and ΔA is the corresponding area of the volume element $\Delta\mathbf{x}$ perpendicular to $\boldsymbol{\mu}$. The particle density distribution is related to the particle fluence by

$$u(\mathbf{x}, \boldsymbol{\mu}, E, t) = v(E)N(\mathbf{x}, \boldsymbol{\mu}, E, t). \tag{2.3}$$

The change in the number of particles in the space-angle-element $\Delta\mathbf{x}\Delta\boldsymbol{\mu}\Delta E$ between times t and $t + \Delta t$ is given by

$$\begin{aligned} &\big[N(\mathbf{x}, \boldsymbol{\mu}, E, t + \Delta t) - N(\mathbf{x}, \boldsymbol{\mu}, E, t + \mathrm{d}t)\big]\Delta\mathbf{x}\Delta\boldsymbol{\mu}\Delta E \\ &\quad = \text{increases due to external source terms} \\ &\qquad - \text{decreases due to streaming out of } \Delta\mathbf{x} \\ &\qquad - \text{decreases due to collisions out of } \Delta\boldsymbol{\mu}\Delta E \\ &\qquad + \text{increases due to scattering into } \Delta\boldsymbol{\mu}\Delta E. \end{aligned} \tag{2.4}$$

The prescription of these contributions in terms of $N(\mathbf{x}, \boldsymbol{\mu}, E, t)$, $\Delta\ell$, ΔA, $\Delta\boldsymbol{\mu}$, ΔE and Δt is beyond the scope of this discussion; see [71] for more details. A partial integro-differential equation for $N(\mathbf{x}, \boldsymbol{\mu}, E, t)$ is obtained by completing the balance equation (2.4), dividing by $\Delta\ell\Delta A\Delta\boldsymbol{\mu}\Delta E\Delta t$ and taking appropriate limits as Δt and $\Delta\ell$ tend to zero. Equation (2.1) is obtained by rewriting the particle density function in terms of the angular flux via (2.3).

Equation (2.1) is supplemented with an initial condition $u(\mathbf{x}, \boldsymbol{\mu}, E, 0) = u_0(\mathbf{x}, \boldsymbol{\mu}, E)$ and a boundary condition whose specification can be difficult. In order to achieve the latter, we form the following boundary sets:

$$\begin{aligned} \Gamma_{in} &= \{(\mathbf{x}, \boldsymbol{\mu}, E, t) \in \Omega \times \mathbb{S} \times \mathbb{Y} \times [0, \infty) : \boldsymbol{\mu} \cdot \mathbf{n}(\mathbf{x}) < 0\}, \\ \Gamma_{out} &= \{(\mathbf{x}, \boldsymbol{\mu}, E, t) \in \Omega \times \mathbb{S} \times \mathbb{Y} \times [0, \infty) : \boldsymbol{\mu} \cdot \mathbf{n}(\mathbf{x}) \geq 0\}, \end{aligned}$$

where $\mathbf{n}(\mathbf{x})$ denotes the outward unit normal to Ω on the boundary $\partial\Omega$. The two main classes of boundary conditions for (2.1) are specified on Γ_{in} and read as follows [90]:

- $u(\mathbf{x}, \boldsymbol{\mu}, E, t) = g(\mathbf{x}, \boldsymbol{\mu}, E, t)$ for some function g. This is the typical Dirichlet boundary condition - the special case $g = 0$ is often referred to as a *vacuum boundary condition* and the corresponding spatial part of Γ_{in} (for fixed $\boldsymbol{\mu} \in \mathbb{S}$) is referred to as a *free surface*.

- $u(\mathbf{x}, \boldsymbol{\mu}, E, t) = \int_{\mathbb{Y}} \int_{\mathbb{S}} \kappa(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}, E' \to E, t) u(\mathbf{x}, \boldsymbol{\mu}', E', t) \, \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E'$ for some function κ. This describes a general class of reflecting/scattering boundary conditions when radiative particles interact with the spatial boundary. The function $\kappa(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}, E' \to E, t)$ is an albedo function describing the relative likelihood that a particle travelling in direction $\boldsymbol{\mu}'$ is deflected along $\boldsymbol{\mu}$ upon interacting with the boundary[1]. Whenever particles are reflected back according to the general reflection rule, the boundary condition is referred to as a *specular boundary condition*; whenever particles are deflected back in an isotropic distribution, the boundary condition is referred to as a *white boundary condition* [68].

For many practical applications, the angular flux is often not a quantity of interest, but rather another quantity called the *scalar flux* [90], defined by

$$\phi(\mathbf{x}, E, t) = \int_{\mathbb{S}} u(\mathbf{x}, \boldsymbol{\mu}, E, t) \ \mathrm{d}\boldsymbol{\mu}. \tag{2.5}$$

In this sense, the angular flux quantifies the distribution of particles at a given spatial position $\mathbf{x}$ with energy E at time t moving in the direction $\boldsymbol{\mu}$, whereas the scalar flux quantifies the distribution of particles at a given spatial position $\mathbf{x}$ with energy E at time t moving in *any* direction.

Equation (2.1) provides a basic template for the types of systems arising in more complex mathematical models of radiative transfer. For example, many applications only consider the steady-state solution, which satisfies the *time-independent, poly-energetic linear Boltzmann transport equation*:

$$\begin{aligned} &\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}, E) + (\alpha(\mathbf{x}, \boldsymbol{\mu}, E) + \beta(\mathbf{x}, \boldsymbol{\mu}, E)) u(\mathbf{x}, \boldsymbol{\mu}, E) \\ &\qquad = \int_{\mathbb{Y}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}, E' \to E) u(\mathbf{x}, \boldsymbol{\mu}', E') \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E' + f(\mathbf{x}, \boldsymbol{\mu}, E). \end{aligned} \tag{2.6}$$

Equation (2.1) is often replaced with (2.6) whenever the particle speed $v(E)$ is expected to be very large. The discretisation of Equation (2.6) shall be the main focus of this work.

2.1.1 Physical processes in radiation transport

As mentioned earlier, photons and electrons may undergo a variety of scattering and absorption interactions as they are transported through a medium. A couple of these

[1] We adopt the convention that $\kappa = 0$ whenever $\boldsymbol{\mu}' \cdot \mathbf{n} < 0$.

interactions are explained below; a more complete overview of other types of interactions is given in [85].

Compton scattering of photons Compton scattering describes a process whereby an incident (high-energy) photon γ (with energy E) is absorbed by an electron e in an atom (typically assumed as initially being at rest), resulting in the electron recoiling and a new photon γ' with energy E' (typically with $E' < E$) emitted at an angle φ from the initial photon's incoming trajectory. The difference in energy between the incident and emitted photon is transferred to the recoiling electron [64]. Figure 2.1 depicts this interaction.

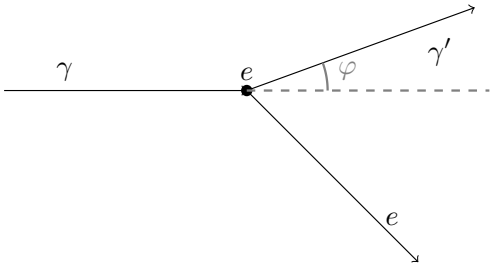

Figure 2.1: Schematic diagram of Compton scattering.

The differential scattering cross-section $\theta(\mathbf{x}, \boldsymbol{\mu} \to \boldsymbol{\mu}', E \to E')$ is defined by

$$\theta(\mathbf{x}, \boldsymbol{\mu} \to \boldsymbol{\mu}', E \to E') = \rho(\mathbf{x})\theta_{KN}(\boldsymbol{\mu} \cdot \boldsymbol{\mu}', E, E'),$$

where $\rho(\mathbf{x})$ denotes the (local) electron density and $\theta_{KN}(\cos\varphi, E, E')$ denotes the *Klein-Nishina differential scattering cross-section per electron* [36] defined by

$$\theta_{KN}(\cos\varphi, E, E') = \frac{r_e^2}{2}\left(\frac{E'}{E}\right)^2\left(\frac{E'}{E} + \frac{E}{E'} - \sin^2\varphi\right). \tag{2.7}$$

Here, $r_e \approx 2.818\times10^{-15}$m denotes the classical electron radius and energies are measured in units of kiloelectron volts (keV).

The fraction of energy retained by the recoiling photon can be determined from kinematic considerations and is given by

$$\frac{E'}{E} = \frac{1}{1 + \frac{E}{m_e c^2}(1 - \cos\varphi)} =: P(\cos\varphi, E), \tag{2.8}$$

where $m_e c^2 \approx 511$keV denotes the electron rest energy. Such kinematic constraints can also be used to determine the trajectory of the recoiling electron, which is typically not perpendicular to that of the recoiling photon. This constraint can be incorporated into the definition of the Klein-Nishina differential scattering cross-section by introducing a Dirac delta distribution:

$$\begin{aligned}\theta_{KN}(\cos\varphi, E, E') = \frac{r_e^2}{2}P(\cos\varphi, E)^2\left(P(\cos\varphi, E) + P(\cos\varphi, E)^{-1} - \sin^2\varphi\right)\cdot \\ \delta\left(E' - EP(\cos\varphi, E)\right) \\ = \theta_{KN}(\cos\varphi, E)\delta\left(E' - EP(\cos\varphi, E)\right).\end{aligned}$$

It is typical to express energies in units of electron rest energy by introducing the rescaled energetic variables $\alpha_\gamma = \frac{E}{m_e c^2}$ (resp. $\alpha'_\gamma = \frac{E'}{m_e c^2}$) - here, we use the subscript γ to distinguish the photon energies α_γ and α'_γ from the macroscopic absorption cross-section α. In this notation, we may write

$$\theta_{KN}(\cos\varphi, \alpha_\gamma) = \frac{r_e^2}{2}\left(\frac{1}{1+\alpha_\gamma(1-\cos\varphi)}\right)^2\left(1+\cos^2\varphi + \frac{\alpha_\gamma^2(1-\cos\varphi)^2}{1+\alpha_\gamma(1-\cos\varphi)}\right).$$

The expression for $\theta_{KN}(\cos\varphi, \alpha_\gamma)$ as given above may also be used as a mono-energetic scattering kernel for fixed $\alpha_\gamma > 0$. The first few coefficients in the Fourier-Legendre series of this scattering kernel have been studied in [82].

By substituting (2.7) into (2.2), the following expression for the macroscopic Compton scattering cross-section $\beta_C(\mathbf{x}, \alpha_\gamma)$ (with energy measured in units of electron rest energies) can be obtained [50]:

$$\beta_C(\mathbf{x}, \alpha_\gamma) = 2\pi r_e^2 \rho(\mathbf{x})\left[\frac{1+\alpha_\gamma}{\alpha_\gamma^2}\left(\frac{2(1+\alpha_\gamma)}{1+2\alpha_\gamma} - \frac{\log(1+2\alpha_\gamma)}{\alpha_\gamma}\right) + \frac{\log(1+2\alpha_\gamma)}{2\alpha_\gamma} - \frac{1+3\alpha_\gamma}{(1+2\alpha_\gamma)^2}\right].$$

The description of the Compton scattering kinematics by the Klein-Nishina differential scattering cross-section is only reliable for interactions in which the momentum transfer is much larger than the average momentum of the electron [85]. Outside of the range of photon energies in which Compton scattering is the dominant scattering process, additional corrections to the differential scattering cross-section are made at both low energies (due to electron binding effects) and high energies (due to, among other things, the double Compton effect) [55]. For a given material, macroscopic cross-sections for many types of scattering interactions are often given as tabulated data [18, 91], which may be used to construct fitted functions for use in Monte Carlo simulations [36].

Møller scattering of electrons Møller scattering describes a process whereby an incident electron e of energy E collides with another electron e', resulting in two free recoiling electrons. For simplicity of presentation, we shall assume that e' is initially at rest. The recoiling electrons are indistinguishable and it is conventional to denote by e_p (resp. e_s) the primary (resp. secondary) electron recoiling with higher (resp. lower) energy [72]. That is, the primary electron is the one that recoils the fastest. The maximum allowable energy transfer from the incident electron to the secondary electron is therefore given by $\frac{E}{2}$.

The differential scattering cross-section $\theta(\mathbf{x}, \boldsymbol{\mu} \to \boldsymbol{\mu}', E \to E')$ (for both primary and secondary electrons) is defined by

$$\theta(\mathbf{x}, \boldsymbol{\mu} \to \boldsymbol{\mu}', E \to E') = \rho(\mathbf{x})\theta_M(\boldsymbol{\mu}\cdot\boldsymbol{\mu}', E, E'),$$

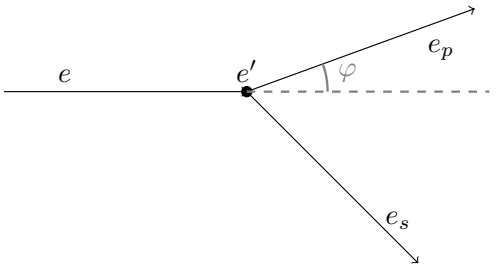

Figure 2.2: Schematic diagram of Møller scattering.

where $\rho(\mathbf{x})$ denotes the (local) electron density and $\theta_M(\cos\varphi, E, E')$ denotes the *Møller differential scattering cross-section per electron* [19] defined by

$$\theta_M(\cos\varphi, E, E') = \frac{2\pi r_e^2 m_e c^2}{\beta_e^2 (E')^2}\left(1 + \frac{(E')^2}{(E-E')^2} + \frac{\alpha_e^2}{(\alpha_e+1)^2}\left(\frac{E'}{E}\right)^2 - \frac{2\alpha_e+1}{(\alpha_e+1)^2}\frac{E'}{E-E'}\right), \tag{2.9}$$

where r_e^2 and $m_e c^2$ are as before, $\alpha_e = \frac{E}{m_e c^2}$ denotes the kinetic energy of the incident electron expressed in units of electron rest energies and $\beta_e = \sqrt{\frac{\alpha(\alpha+2)}{(\alpha+1)^2}}$ denotes the speed of the incident electron divided by the speed of light. We have used the subscript e to differentiate the quantities α_e and β_e from α (the macroscopic absorption cross-section) and β (the macroscopic scattering cross-section).

Denoting by $\alpha'_e = \frac{E'}{m_e c^2}$ the kinetic energy of (either of) the recoiling electrons, it can be shown via kinematic considerations that the incident electron energy α_e, recoiling electron energy α'_e and scattering angle φ are related in the following way:

$$\cos\varphi = \sqrt{\frac{\alpha'_e}{\alpha_e}\cdot\frac{\alpha_e+2}{\alpha'_e+2}}. \tag{2.10}$$

The trajectories of the recoiling electrons are typically not perpendicular to each other. This constraint can be incorporated into the definition of the Møller differential scattering cross-section by introducing a Dirac delta distribution in a similar fashion as was done in the Compton scattering case above.

Owing to the singularities of the Møller differential scattering cross-section at $E' \approx 0$ and $E' \approx E$ for a given incoming electron kinetic energy E, the usual definition of the Møller macroscopic scattering cross-section $\beta_M(\mathbf{x}, E) = \beta(\mathbf{x}, E)$ with $\beta(\mathbf{x}, E)$ as in (2.2) is no longer finite. The kinematic constraint (2.10) also implies that primary (resp. secondary) electrons with recoiling kinetic energy $E' \approx E$ (resp. $E' \approx 0$) are deflected through an angle $\varphi \approx 0$ (resp. $\varphi \approx \frac{\pi}{2}$). Møller scattering is therefore considered to be a *highly peaked* scattering process.

In the context of estimating dose deposition by electron ionisation, Hensel et al. [50] offer an alternative approach by considering primary and secondary electrons separately. They remark that low-energy secondary electrons deposit their energy locally as they have a very short range in most media and thus focus on the transport of high-energy

primary electrons. A Fokker-Planck approximation for the forward-peaked scattering process is derived via the methodology in [79] to replace the linear Boltzmann transport equation (2.6) with the asymptotic equation

$$\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}, E) - T(\mathbf{x}, E) \Delta_{\mathbb{S}} u(\mathbf{x}, \boldsymbol{\mu}, E) - \frac{\partial}{\partial E} \left[S(\mathbf{x}, E) u(\mathbf{x}, \boldsymbol{\mu}, E) \right] = f(\mathbf{x}, \boldsymbol{\mu}, E),$$

where $\Delta_{\mathbb{S}}$ denotes the Laplace operator on $\mathbb{S}$ and the coefficients $T(\mathbf{x}, E)$ and $S(\mathbf{x}, E)$ are defined for a general differential scattering cross-section $\theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E \to E')$ by

$$\begin{aligned} T(\mathbf{x}, E) &= \int_{\mathbb{Y}} \int_{\mathbb{S}} (1 - \boldsymbol{\mu}' \cdot \boldsymbol{\mu}) \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E' \to E) \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E', \\ S(\mathbf{x}, E) &= \int_{\mathbb{Y}} \int_{\mathbb{S}} (E - E') \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E \to E') \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E'. \end{aligned}$$

Coupled photon-electron models In radiation oncology applications, it is often the case that multiple species of radiative particles must be tracked. For instance, if a patient is to undergo radiotherapy using beams consisting (primarily) of photons, then it is necessary to keep track of both the photon fluence (denoted by u_γ) as well as the electron fluence (denoted by u_e), since photons may undergo Compton scattering events. Such scattering events liberate electrons from the tissue medium, which may continue to travel through the patient and cause further ionisation. In this model, we may describe the distribution of the photon and electron fluences using a coupled system of LBTEs [50, 99]:

$$\begin{aligned} &\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u_\gamma(\mathbf{x}, \boldsymbol{\mu}, E) + (\alpha_\gamma(\mathbf{x}, E) + \beta_\gamma(\mathbf{x}, E)) u_\gamma(\mathbf{x}, \boldsymbol{\mu}, E) \\ &\qquad = \int_{\mathbb{Y}} \int_{\mathbb{S}} \theta_{\gamma\gamma}(\mathbf{x}, \boldsymbol{\mu}' \cdot \boldsymbol{\mu}, E' \to E) u_\gamma(\mathbf{x}, \boldsymbol{\mu}', E') \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E + f_\gamma(\mathbf{x}, \boldsymbol{\mu}, E), \end{aligned} \tag{2.11}$$

$$\begin{aligned} &\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u_e(\mathbf{x}, \boldsymbol{\mu}, E) + (\alpha_e(\mathbf{x}, E) + \beta_e(\mathbf{x}, E)) u_e(\mathbf{x}, \boldsymbol{\mu}, E) \\ &\qquad = \int_{\mathbb{Y}} \int_{\mathbb{S}} \theta_{\gamma e}(\mathbf{x}, \boldsymbol{\mu}' \cdot \boldsymbol{\mu}, E' \to E) u_\gamma(\mathbf{x}, \boldsymbol{\mu}', E') \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E \\ &\qquad\quad + \int_{\mathbb{Y}} \int_{\mathbb{S}} \theta_{ee}(\mathbf{x}, \boldsymbol{\mu}' \cdot \boldsymbol{\mu}, E' \to E) u_e(\mathbf{x}, \boldsymbol{\mu}', E') \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E \\ &\qquad\quad + f_e(\mathbf{x}, \boldsymbol{\mu}, E). \end{aligned} \tag{2.12}$$

Here, the terms α_γ, β_γ and $\theta_{\gamma\gamma}$ (resp. α_e, β_e and θ_{ee}) denote the typical macroscopic absorption, macroscopic scattering and differential scattering cross-sections associated with the physics of photons (resp. electrons). The additional scattering term in (2.12) including the differential cross-section $\theta_{\gamma e}$ represents a source of electrons produced by, e.g., Compton scattering of photons against electrons in the medium.

In the example given above, the photon-electron coupling is one-way. That is, one may numerically solve (2.11) for the photon fluence u_γ, which is used to generate the source term in (2.12) for the electron fluence.

2.2 Deterministic methods for radiation transport

Given the structure of Equation (2.6), the spatial, angular and energetic components of the solution are typically discretised using a number of numerical techniques. We shall give a brief overview of the methods used to discretise each component.

2.2.1 Energetic discretisation

Historically, the standard approach to discretising the energetic component of the time-independent LBTE is the *multigroup approximation* [11, 70]. The energy domain $\mathbb{Y}$ is first restricted to a (physically-relevant) finite interval (E_{min}, E_{max}). This range is further divided into a fixed number of energy groups $N \geq 1$ with the g^{th} energy group, $1 \leq g \leq N$, given by the range (E_g, E_{g-1}), where

$$E_{max} = E_0 \geq E_1 \geq \cdots \geq E_{N-1} \geq E_N = E_{min}.$$

Following the convention set out in [11, 70], the energy groups $\{(E_g, E_{g-1})\}_{g=1}^N$ and the energy group cut-offs $\{E_g\}_{g=0}^N$ are listed in *descending* order. For each energy group g, the function $u_g(\mathbf{x}, \boldsymbol{\mu})$ is defined by

$$u_g(\mathbf{x}, \boldsymbol{\mu}) = \int_{E_g}^{E_{g-1}} u(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}E \tag{2.13}$$

and we further suppose that there exists a function $w : (E_{min}, E_{max}) \to \mathbb{R}$ satisfying, for all $1 \leq g \leq N$, the following separability and normalisation conditions:

$$u(\mathbf{x}, \boldsymbol{\mu}, E) \approx w(E) u_g(\mathbf{x}, \boldsymbol{\mu}) \quad \text{for } E_g < E < E_{g-1}, \tag{2.14}$$

$$\int_{E_g}^{E_{g-1}} w(E) \ \mathrm{d}E = 1. \tag{2.15}$$

Note that the normalisation condition (2.15) is necessary to ensure that (2.13) holds whenver $u(\mathbf{x}, \boldsymbol{\mu}, E)$ is separable (that is, whenever (2.14) holds with equality). Moreover, let us define group-dependent data terms for $1 \leq g, g' \leq N$ as follows:

$$\begin{aligned}
\alpha_g(\mathbf{x}, \boldsymbol{\mu}) &= \int_{E_g}^{E_{g-1}} w(E)\alpha(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}E, \\
\beta_g(\mathbf{x}, \boldsymbol{\mu}) &= \int_{E_g}^{E_{g-1}} w(E)\beta(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}E, \\
\theta_{g' \to g}(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}) &= \int_{E_g}^{E_{g-1}} \int_{E_{g'}}^{E_{g'-1}} w(E')\theta(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}, E' \to E) \ \mathrm{d}E' \, \mathrm{d}E, \\
f_g(\mathbf{x}, \boldsymbol{\mu}) &= \int_{E_g}^{E_{g-1}} w(E) f(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}E.
\end{aligned}$$

It can be shown that integrating (2.6) over a single energy group g and replacing $u(\mathbf{x}, \boldsymbol{\mu}, E)$ with $\sum_{g'=1}^N w(E) u_{g'}(\mathbf{x}, \boldsymbol{\mu})$ yields the following coupled system of mono-energetic problems for the mono-energetic group approximations $\{u_g\}_{g=1}^N$:

$$\begin{aligned}
&\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u_g(\mathbf{x}, \boldsymbol{\mu}) + \left(\alpha_g(\mathbf{x}, \boldsymbol{\mu}) + \beta_g(\mathbf{x}, \boldsymbol{\mu})\right) u_g(\mathbf{x}, \boldsymbol{\mu}) \\
&\qquad = \sum_{g'=1}^{N} \int_{\mathbb{S}} \theta_{g' \to g}(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}) u_{g'}(\mathbf{x}, \boldsymbol{\mu}') \ \mathrm{d}\boldsymbol{\mu}' + f_g(\mathbf{x}, \boldsymbol{\mu}).
\end{aligned} \tag{2.16}$$

As remarked in [70], the evaluation of the group cross-sectional data requires knowledge of the weight function $w(E)$, as well as the original cross-sectional data $\alpha(\mathbf{x}, \boldsymbol{\mu}, E)$ and $\beta(\mathbf{x}, \boldsymbol{\mu}, E)$. The selection of $w(E)$ can be made in a number of ways. If very fine energy grids are to be employed, analytical or semi-analytical approximations to $w(E)$ may be sufficient - in this case, the simplest prescription of $w(E)$ may be given by the piecewise-constant function

$$w(E) = \frac{1}{E_{g-1} - E_g}$$

for $1 \leq g \leq N$. On coarser energy grids, prescriptions of $w(E)$ can be made by considering a model problem based on the time-independent poly-energetic LBTE (2.6) in which the solution u has no dependence on space or angle.

In the special case $N = 1$, (2.16) reduces to the *time-independent, mono-energetic linear Boltzmann transport equation*:

$$\begin{aligned} \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}) + (\alpha(\mathbf{x}, \boldsymbol{\mu}) + \beta(\mathbf{x}, \boldsymbol{\mu}))\, u(\mathbf{x}, \boldsymbol{\mu}) \\ = \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}) u(\mathbf{x}, \boldsymbol{\mu}')\ \mathrm{d}\boldsymbol{\mu}' + f(\mathbf{x}, \boldsymbol{\mu}), \end{aligned} \tag{2.17}$$

where we have suppressed the subscript notation for brevity.

When the weight function $w(E)$ is selected to be a piecewise-constant function, the resulting multigroup scheme (2.16) can be thought of as the lowest-order discontinuous Galerkin (DG) semidiscretisation in energy of the poly-energetic LBTE. In Chapter 3, we build upon this observation to derive a high-order energetic DG scheme whose structure is analogous to that of (2.16).

2.2.2 Angular discretisation

Once the energy domain has been appropriately discretised, an angular semidiscretisation of the resulting (system of) mono-energetic problems (2.17) may be performed. There are two commonly-used approaches to this semidiscretisation, which will be outlined below.

Discrete ordinates methods The first approach, called the *discrete ordinates method* [7, 57], attempts to replace integrals over the angular domain with an approximate quadrature/collocation scheme of the form

$$\int_{\mathbb{S}} f(\mathbf{x}, \boldsymbol{\mu})\ \mathrm{d}\boldsymbol{\mu} \approx \sum_{q=1}^{N} w_q f(\mathbf{x}, \boldsymbol{\mu}_q).$$

We henceforth denote by $Q = \{(w_q, \boldsymbol{\mu}_q)\}_{q=1}^{N} \subset \mathbb{R}_{>0} \times \mathbb{S}$ the set of quadrature weights and points used in the discrete ordinates semidiscretisation. Additional assumptions on the weights and points in Q ensure numerical stability of the quadrature scheme on $\mathbb{S}$; similar assumptions are standard in the wider numerical quadrature literature. Beyond this, there are a number of key ideas used in prescribing "good" quadrature schemes Q; we will review these ideas later.

On substitution of the above quadrature rule into the time-independent, mono-energetic LBTE (2.17), one reduces the problem from one over all angular directions in $\mathbb{S}$ to one over a finite subset of directions $\{\boldsymbol{\mu}_q\}_{q=1}^N$. This yields a system of hyperbolic PDEs coupled via the scattering term, with each equation corresponding to a single ordinate direction. Specifically, writing $u_N^{(q)}(\mathbf{x}) \approx u(\mathbf{x}, \boldsymbol{\mu}_q)$ to denote the approximate angular flux corresponding to the fixed ordinate direction $\boldsymbol{\mu}_q$, the discrete ordinates semidiscretisation reads as follows: find $\{u_N^{(q)}\}_{q=1}^N$ such that

$$\boldsymbol{\mu}_q \cdot \nabla u_N^{(q)}(\mathbf{x}) + (\alpha(\mathbf{x}) + \beta(\mathbf{x}))u_N^{(q)}(\mathbf{x}) = \sum_{q'=1}^{N} w_{q'}\theta(\mathbf{x}, \boldsymbol{\mu}_{q'} \to \boldsymbol{\mu}_q)u_N^{(q')} + f(\mathbf{x}, \boldsymbol{\mu}_q) \quad (2.18)$$

for each $1 \leq q \leq N$. Further spatial discretisations of the system (2.18) can then be selected from any of the methods outlined in 2.2.3. Boundary conditions for the semidiscrete angular fluxes may be prescribed straightforwardly on replacing the continuum variable $\boldsymbol{\mu}$ by each of the ordinate directions $\boldsymbol{\mu}_q$. Owing to the nature of the coupling of the transport problems in (2.18), iterative methods are typically employed to solve the coupled transport problems; see Chapter 2.5 for some commonly-employed approaches.

Many suitable choices of quadrature schemes Q have been proposed for solving (2.18), many of which share a number of recommended design choices [66]. The following list summarises some desirable properties:

- The quadrature scheme Q should be able to integrate spherical harmonic functions $Y_{l,m}(\boldsymbol{\mu})$ (see below) of up to some degree L; i.e. for $0 \leq l \leq L$ and $-l \leq m \leq l$. These functions can be considered as polynomials in $\mathbb{R}^d$ restricted to $\mathbb{S}$ in the case $d = 3$.

- The choice of the ordinate directions $\boldsymbol{\mu}_q$ must be directionally unbiased, so that the set of quadrature points remains unchanged under selected rotations[2]. For example, many angular quadrature schemes are chosen to be invariant under rotations by $\pi/2$ in any direction. Furthermore, if any two quadrature points $\boldsymbol{\mu}_q$ and $\boldsymbol{\mu}_{q'}$ can be reached from each other only by rotations by $\pi/2$, the corresponding weights w_q and $w_{q'}$ must be equal - such a symmetry is referred to as "octahedral symmetry".

- The choice of the ordinate directions $\boldsymbol{\mu}_q$ should respect the so-called "principle of optical reciprocity". The statement pertains to the scattering kernel $\theta(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu})$; we shall briefly omit the spatial argument for simplicity of presentation. The principle states that, for any quadrature points $\boldsymbol{\mu}_q$ and $\boldsymbol{\mu}_{q'}$, we have

$$\theta(\boldsymbol{\mu}_q \to \boldsymbol{\mu}_{q'}) = \theta(\boldsymbol{\mu}_{q'} \to \boldsymbol{\mu}_q).$$

[2]The stronger condition - that the set of quadrature points should remain unchanged under *any* rotation, cannot be satisfied for any finite set of ordinate directions.

The principle states that any particle in the (mono-energetic) system will undergo the same collision events whether it moves along its trajectory "forwards" or "backwards". For this reason, many quadrature schemes are constructed so that, for any quadrature point $\boldsymbol{\mu}_q$ in Q, its antipode $-\boldsymbol{\mu}_q$ is also a quadrature point in Q. This also ensures that odd spherical harmonics are integrated exactly [12].

- The quadrature scheme Q should have nodes and weights that satisfy

$$\sum_{q=1}^{N} w_q \boldsymbol{\mu}_q = \mathbf{0}.$$

  This requirement ensures that the number of particles in the system remains conserved.

There is a vast literature on quadrature schemes on the surface of the unit sphere $\mathbb{S} \subset \mathbb{R}^3$. A number of key families of schemes are summarised below.

- **Level-symmetric quadrature:** Denoted by S_N, level-symmetric quadratures are a widely-used family of quadrature schemes used in discrete ordinate methods for solving the LBTE. Here, N is understood to be an even positive integer. The idea is to insert $\frac{N}{4}\left(\frac{N}{2}+1\right)$ quadrature points in each octant of the unit sphere $\mathbb{S} \subset \mathbb{R}^3$ [90]. Within each octant, the quadrature points are chosen to lie on one of the latitudes of the sphere under different orientations. Once the points are selected, the weights are chosen to integrate as many spherical harmonic functions as possible, starting from those of the lowest degree.

- T_N **quadrature:** A subtly different scheme to the S_N method above, the T_N family of quadrature schemes [97] first triangulate the surface of the sphere. Each "triangle" then corresponds to a quadrature point and weight, with the quadrature point taken to be the centroid of the "triangle" and the quadrature weight its area. In this sense, this can be considered a zeroth-order discontinuous Galerkin discretisation in angle.

- **Lebedev quadrature:** Like the level-symmetric quadrature schemes, Lebedev quadrature schemes also rely on octahedral symmetry, though the surface of the unit sphere is not first subdivided into octants and the quadrature points and weights are computed simultaneously. As before, the quadrature schemes are designed to integrate all spherical harmonics up to some given degree exactly. However, the resulting nonlinear system for the points/weights is drastically reduced in size by invoking a theorem by Sobolev [12]. This theorem states that, for any quadrature scheme on the sphere that is invariant with respect to some group G, the scheme can exactly integrate all spherical harmonics of maximal degree if and only if it can exactly integrate those functions that are invariant under G.

- **Spherical t-designs:** These quadrature schemes are designed so that each quadrature point has the same weight, and are thus considered Chebyshev quadratures [12]. As before, the points are chosen to integrate all spherical harmonics up to some maximal degree t; however, these schemes may not be symmetric or even unique.

- **Lagrange discrete ordinates (LDO):** A relatively new approach is to shift away from the quadrature-based philosophy of the classical S_N methods towards an approach based on Lagrange interpolation of functions defined on the unit sphere [3]. The resulting numerical method requires little modification of pre-existing S_N discrete ordinate codes; furthermore, the angular flux can easily be evaluated at points other than those found in the original quadrature set.

If the physical system under investigation is advection-dominated (i.e. there is very little scattering of radiation) and contains strong sources of localised emission, spurious oscillations may manifest in the quantity

$$\phi_N(\mathbf{x}) = \sum_{q=1}^{N} \omega_q u_N^{(q)}(\mathbf{x}),$$

which we recognise as an approximation of the scalar flux $\phi(\mathbf{x})$ defined in (2.5) using the discrete ordinates quadrature scheme in angle. An example of these artifacts, called *ray effects*, can be seen in Figure 2.3. Since information about the angular flux is only advected along those ordinate directions chosen in the discrete ordinates scheme, ray effects arise when there is little scattering of this information away from those directions; this can be observed in Figure 2.3 as the "wedge-like" regions in which the scalar flux is under-approximated. The magnitude of these fluctuations in the approximate scalar flux can be reduced, and thus ray effects can be mitigated, by taking more ordinate directions in the discrete ordinates scheme.

Characteristic methods A similar method to the discrete ordinates method is the *method of long characteristics* proposed by Askew [8]. The formulation of the method starts with the semidiscretisation (2.18) as well as a discretisation of the spatial geometry into a mesh consisting of (polytopic) cells. Rather than applying spatial discretisation techniques such as finite element methods to this system of PDEs (see Chapter 2.2.3), the method of long characteristics first rewrites (2.18) as

$$\sum_{k=1}^{d} \mu_{q,k} \frac{\partial u_N^{(q)}(\mathbf{x})}{\partial x_k} = \sum_{q'=1}^{N} w_{q'} \theta(\mathbf{x}, \boldsymbol{\mu}_{q'} \to \boldsymbol{\mu}_q) u_N^{(q')}(\mathbf{x}) + f(\mathbf{x}, \boldsymbol{\mu}_q) - (\alpha(\mathbf{x}) + \beta(\mathbf{x})) u_N^{(q)}(\mathbf{x}),$$

where we have expressed the spatial variable as $\mathbf{x} = (x_k)_{k=1}^d$, where d denotes the spatial dimension of the problem.

We now rewrite each of the N linear transport equations as a Cauchy problem; that is, we write $x_k = x_k(t; s)$ for $1 \leq k \leq d$ and $u_N^{(q)} = u_N^{(q)}(t; s)$ for $1 \leq q \leq N$,

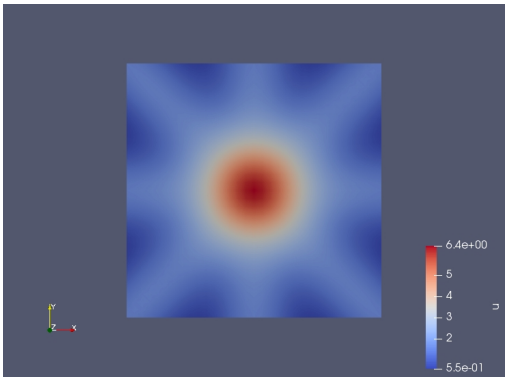

Figure 2.3: Plot of an approximate scalar flux arising from a discrete ordinates method using $N = 8$ equally-weighted ordinate directions applied to a mono-energetic LBTE with a point source at the origin. The exact scalar flux is radially-symmetric about the origin.

where t parametrises distance along characteristic curves (from the inflow boundary) and s parametrises the initial data on the inflow boundary given by $x_{k,0} = x_k(0; s)$ and $u_{N,D}(\mathbf{x}_0, \boldsymbol{\mu}_q) = u_N^{(q)}(0; s)$. For each $1 \leq q \leq N$, the characteristic equations then become

$$\begin{aligned} \frac{\mathrm{d}x_k}{\mathrm{d}t} &= \mu_k^{(q)} \quad \text{for } 1 \leq k \leq d, \\ \frac{\mathrm{d}u_N^{(q)}}{\mathrm{d}t} &= \sum_{q'=1}^{N} w_k \theta(\mathbf{x}, \boldsymbol{\mu}_{q'} \to \boldsymbol{\mu}_q) u_N^{(q')}(\mathbf{x}) + f(\mathbf{x}, \boldsymbol{\mu}_q) - (\alpha(\mathbf{x}) + \beta(\mathbf{x})) u_N^{(q)}(\mathbf{x}). \end{aligned}$$

These ODEs form the basis of the method of long characteristics, which seeks to approximately solve the LBTE along each characteristic curve for different initial data (parametrised by s) by integrating the equations above over intervals (t_m, t_{m+1}); this allows one to infer the angular flux at position t_{m+1} from the flux at position t_m along the characteristic curve.

In the simplest case, the method of long characteristics assigns to each element κ in a spatial mesh $\mathcal{T}_\Omega$ of the spatial domain Ω a constant polynomial approximation of the angular flux $u(\mathbf{x}, \boldsymbol{\mu})$. In order to achieve this, one requires that, for each ordinate direction $\boldsymbol{\mu}_q$, there are sufficiently-many characteristic curves (corresponding to different initial data) that each element κ in the underlying mesh has a characteristic passing through it; the value of $u_N^{(q)}(\mathbf{x})|_\kappa$ is then given as the average of the inflow and outflow angular flux values on the characteristics passing through κ.

The requirement in the method of long characteristics that each κ has at least one characteristic curve passing through it for every ordinate direction means that such methods struggle in regions where the mesh is fine [74]. One idea to overcome this is the method of short characteristics proposed by Takeuchi [94], whereby a set of characteristic curves are selected for each element in the mesh.

Spherical harmonics method The second class of methods for the angular discretisation of the time-independent mono-energetic LBTE are based on truncated expansions of the angular flux in terms of an orthogonal basis of the unit sphere in d dimensions. For simplicity, we shall limit ourself to the three-dimensional case $\mathbb{S} \subset \mathbb{R}^3$, for which any $\boldsymbol{\mu} \in \mathbb{S}$ may be written in terms of parameters $(\psi, \varphi) \in (0, \pi) \times [0, 2\pi)$ as

$$\boldsymbol{\mu} = (\sin\psi\cos\varphi, \sin\psi\sin\varphi, \cos\psi).$$

With a slight abuse of notation, we shall henceforth write the angular flux as $u(\mathbf{x}, \boldsymbol{\mu}) = u(\mathbf{x}, \psi, \varphi)$. We then express u as a *spherical harmonic decomposition* [90]

$$u(\mathbf{x}, \boldsymbol{\mu}) = u(\mathbf{x}, \psi, \varphi) = \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} \sum_{m=-l}^{l} \phi_{l,m}(\mathbf{x}) Y_{l,m}(\psi, \varphi).$$

The spherical harmonic functions $Y_{l,m}(\psi, \varphi)$ are defined (with abuse of notation) by[3]

$$Y_{l,m}(\psi, \varphi) = Y_{l,m}(\boldsymbol{\mu}) = \sqrt{\frac{(l-m)!}{(l+m)!}} P_l^m(\cos\psi) e^{im\varphi},$$

where each function $P_l^m(\cdot)$ is an *associated Legendre function*:

$$P_l^m(x) = \begin{cases} (1-x^2)^{\frac{m}{2}} \frac{\mathrm{d}^m P_l(x)}{\mathrm{d}x^m} & m \geq 0, \\ (-1)^{|m|} P_l^{|m|}(x) & m < 0, \end{cases}$$

and $P_l(\cdot)$ is the l^{th} *Legendre polynomial*, defined recursively by

$$\begin{aligned} P_0(x) &= 1, \\ P_1(x) &= x, \\ (2l+1)xP_l(x) &= (l+1)P_{l+1}(x) + lP_{l-1}(x) \quad \text{for } l \geq 2. \end{aligned}$$

The spherical harmonic method for solving the linear Boltzmann transport equation simply truncates the series expansion at, say, $l = N$ for some non-negative integer N, yielding the approximation:

$$u(\mathbf{x}, \boldsymbol{\mu}) \approx \sum_{l'=0}^{N} \frac{2l'+1}{4\pi} \sum_{m'=-l'}^{l'} \phi_{l',m'}(\mathbf{x}) Y_{l',m'}(\boldsymbol{\mu}).$$

In addition, the following useful properties of the spherical harmonic functions are exploited:

- As is the case for the Legendre polynomials, the spherical harmonic functions also satisfy a recursion relation:

$$(2l+1)xP_l^m(x) = (l-m+1)P_{l+1}^m(x) + (l+m)P_{l-1}^m(x);$$

- The complex conjugate of a spherical harmonic function is another spherical harmonic function:

$$Y_{l,m}(\boldsymbol{\mu}) = Y_{l,-m}^*(\boldsymbol{\mu});$$

[3]The normalisation used here is the Schmidt semi-normalised variant.

- The spherical harmonic functions are orthogonal with respect to the $L^2(\mathbb{S})$-inner product:
$$\int_{\mathbb{S}} Y^*_{l',m'}(\boldsymbol{\mu}) Y_{l,m}(\boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu} = \frac{4\pi}{2l+1}\delta_{l,l'}\delta_{m,m'},$$
where δ denotes the Kronecker delta function;

- The *addition theorem* allows one to rewrite the l^{th} Legendre polynomial in a convenient fashion:
$$P_l(\boldsymbol{\mu}\cdot\boldsymbol{\mu}') = \sum_{m=-l}^{l} Y_{l,m}(\boldsymbol{\mu}) Y^*_{l,m}(\boldsymbol{\mu}').$$

To this end, expanding the source term and scattering kernel as

$$f(\mathbf{x},\boldsymbol{\mu}) \approx \sum_{l'=0}^{N} \frac{2l'+1}{4\pi} \sum_{m'=-l'}^{l'} f_{l',m'}(\mathbf{x}) Y_{l',m'}(\boldsymbol{\mu}),$$
$$\theta(\mathbf{x},\boldsymbol{\mu}\cdot\boldsymbol{\mu}') \approx \sum_{l'=0}^{N} \frac{2l'+1}{4\pi} \theta_{l'}(\mathbf{x}) P_{l'}(\boldsymbol{\mu}\cdot\boldsymbol{\mu}'),$$

it can be shown, by substituting the truncated series expansion for $u(\mathbf{x},\boldsymbol{\mu})$ and the expansions above into the LBTE, multiplying by $Y^*_{l,m}(\boldsymbol{\mu})$ and integrating over $\mathbb{S}$ with respect to $\boldsymbol{\mu}$, that the set of spherical harmonic moments

$$\{\phi_{l,m}(\mathbf{x}) : 0 \le l \le N, -l \le m \le l\}$$

satisfy the following system of first-order PDEs for $0 \le l \le N$ and $-l \le m \le l$:

$$\begin{aligned}
\frac{1}{2l+1}\Bigg[&\frac{1}{2}\sqrt{(l+m+2)(l+m+1)}\left(-\frac{\partial}{\partial x} - i\frac{\partial}{\partial y}\right)\phi_{l+1,m+1}(\mathbf{x}) \\
&+ \frac{1}{2}\sqrt{(l-m+2)(l-m+1)}\left(\frac{\partial}{\partial x} - i\frac{\partial}{\partial y}\right)\phi_{l+1,m-1}(\mathbf{x}) \\
&+ \frac{1}{2}\sqrt{(l-m-1)(l-m)}\left(\frac{\partial}{\partial x} + i\frac{\partial}{\partial y}\right)\phi_{l-1,m+1}(\mathbf{x}) \\
&+ \frac{1}{2}\sqrt{(l+m-1)(l+m)}\left(-\frac{\partial}{\partial x} + i\frac{\partial}{\partial y}\right)\phi_{l-1,m-1}(\mathbf{x}) \\
&+ \sqrt{(l+m+1)(l-m+1)}\frac{\partial\phi_{l+1,m}(\mathbf{x})}{\partial z} \\
&+ \sqrt{(l+m)(l-m)}\frac{\partial\phi_{l-1,m}(\mathbf{x})}{\partial z}\Bigg] + (\alpha(\mathbf{x}) + \beta(\mathbf{x}))\phi_{l,m}(\mathbf{x}) \\
&= \theta_l(\mathbf{x})\phi_{l,m}(\mathbf{x}) + f_{l,m}(\mathbf{x}).
\end{aligned} \tag{2.19}$$

From here, the moments $\phi_{l,m}(\mathbf{x})$ (which are now potentially complex-valued) can be discretised in space using, for example, a finite element method. In the system of equations above, we set $\phi_{l',m'}(\mathbf{x}) = 0$ whenever $(l',m') \notin \{(l,m) : 0 \le l \le N, -l \le m \le l\}$. The resulting method is called the *spherical harmonic method*, or P_N method.

The system of $(N+1)^2$ equations in the form above is very cumbersome to solve numerically, and often N is chosen to be fairly small, say $N = 1$ or $N = 3$, in numerical computations. However, a number of simplifications can be made:

- A typical assumption that is often made is that the original LBTE is posed in a *slab geometry*; i.e. where the solution $u(\mathbf{x}, \boldsymbol{\mu})$ and the functions $\alpha(\mathbf{x})$, $\beta(\mathbf{x})$, $\theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}')$ and $f(\mathbf{x}, \boldsymbol{\mu})$ depend only on $\boldsymbol{\mu}$ and the z-component of $\mathbf{x}$. Here, the dependence of u on the polar angle φ is unimportant, and thus we need only consider the case $m = 0$. This simplification yields a system of $N+1$ ordinary differential equations:

$$\frac{l+1}{2l+1}\frac{\mathrm{d}\phi_{l+1}(z)}{\mathrm{d}z} + \frac{l}{2l+1}\frac{\mathrm{d}\phi_{l-1}(z)}{\mathrm{d}z} + (\alpha(z) + \beta(z))\phi_l(z) = \theta_l(z)\phi_l(z) + f_l(z),$$

  where we have suppressed the dependence on m for simplicity.

- Rather than discretising each equation in (2.19) separately, the system of first-order PDEs (2.19) may be reduced to a smaller system of second- or higher-order PDEs before a spatial discretisation is performed. For example, the P_1 equations can be rearranged (under the assumption that $f_{1,m}(\mathbf{x}) = 0$ for $-1 \leq m \leq 1$) to yield the following equation for the scalar flux $\phi(\mathbf{x}) = \phi_{0,0}(\mathbf{x})$:

$$-\nabla \cdot \left(\frac{1}{3\left(\alpha(\mathbf{x}) + \beta(\mathbf{x}) - \theta_1(\mathbf{x})\right)} \nabla\phi(\mathbf{x}) \right) + \alpha(\mathbf{x})\phi(\mathbf{x}) = f_{0,0}(\mathbf{x}) \tag{2.20}$$

The system of first-order PDEs (2.19) must additionally be supplemented by boundary conditions; we will primarily focus on the imposition of standard Dirichlet boundary conditions of the form

$$u(\mathbf{x}, \boldsymbol{\mu}) = g(\mathbf{x}, \boldsymbol{\mu}) \text{ on } \Gamma_{in},$$

where Γ_{in} is defined by

$$\Gamma_{in} = \{(\mathbf{x}, \boldsymbol{\mu}) \in \partial\Omega \times \mathbb{S} : \boldsymbol{\mu} \cdot \mathbf{n}(\mathbf{x}) < 0\}$$

and $\mathbf{n}(\mathbf{x})$ denotes the outward unit normal to Ω on $\partial\Omega$.

The first class of boundary conditions we consider are *Marshak boundary conditions*. Here, the (truncated) spherical harmonic decomposition of $u(\mathbf{x}, \boldsymbol{\mu})$ is substituted into the Dirichlet boundary condition. The resulting equation is then multiplied by a certain subset of the spherical harmonic functions $Y^*_{l,m}(\boldsymbol{\mu})$ (usually the *odd* functions) and integrated over the hemispherical domain $\mathbb{S}^{in}(\mathbf{x}) \subset \mathbb{S}$, which is defined for any $\mathbf{x} \in \partial\Omega$ by

$$\mathbb{S}^{in}(\mathbf{x}) = \{\boldsymbol{\mu} \in \mathbb{S} : \boldsymbol{\mu} \cdot \mathbf{n}(\mathbf{x}) < 0\}.$$

This yields the following boundary condition for $\mathbf{x} \in \partial\Omega$:

$$\begin{aligned}\phi_{l,m}(\mathbf{x}) &= \int_{\mathbb{S}^{in}(\mathbf{x})} \left(\sum_{l'=0}^{N} \frac{2l'+1}{4\pi} \sum_{m'=-l'}^{l'} \phi_{l',m'}(\mathbf{x}) Y_{l',m'}(\boldsymbol{\mu}) \right) Y^*_{l,m}(\boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu} \\ &= \int_{\mathbb{S}^{in}(\mathbf{x})} g(\mathbf{x}, \boldsymbol{\mu}) Y^*_{l,m}(\boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu}.\end{aligned}$$

When employing the P_N method, additional care must be taken to ensure that the correct number of boundary conditions are supplemented [25]. Specifically, one needs

to prescribe $\frac{N(N+1)}{2}$ boundary conditions, which are typically chosen to correspond to the spherical harmonics $Y_{l,m}$ sharing the same parity as N (i.e. for l odd whenever N is odd, and vice versa).

An alternative approach to imposing the Dirichlet boundary condition above, valid in a one-dimensional slab geometry, is to force the (truncated) spherical harmonic decomposition of u to match the Dirichlet condition at the left-hand (resp. right-hand) boundary for certain angles $\{\boldsymbol{\mu}_i\}_{i=1}^{\frac{N+1}{2}}$ given by the positive (resp. negative) roots of the $(N+1)^{th}$ Legendre polynomial. This is called a *Mark boundary condition* [90].

Finite element methods More recently, finite element discretisations of the angular domain have been studied. Starting from the angular domain $\mathbb{S}$, or a sufficiently-accurate polytopic approximation of $\mathbb{S}$, a computational mesh $\mathcal{T}_{\mathbb{S}}$ is constructed on the surface consisting of $(d-1)$-dimensional polytopic (angular) elements $\kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}$. One may then introduce (continuous or discontinuous) piecewise-polynomial finite element spaces $\mathcal{V}_{\mathbb{S}}$ consisting of functions $v \in \mathcal{V}_{\mathbb{S}}$ such that $v|_{\kappa_{\mathbb{S}}}$ is a polynomial function for all $\kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}$.

Upon selection of an angular basis $\{v_i(\boldsymbol{\mu})\}_{i=1}^{N} \subset \mathcal{V}_{\mathbb{S}}$, $N = \dim \mathcal{V}_{\mathbb{S}}$, the spatial semidiscretisation of (2.17) reads as follows: for each $\mathbf{x} \in \Omega$, find $u(\mathbf{x}, \cdot) \in \mathcal{V}_{\mathbb{S}}$ such that

$$\begin{aligned}&\int_{\mathbb{S}} \boldsymbol{\mu} \cdot \nabla u(\mathbf{x}, \boldsymbol{\mu}) v(\boldsymbol{\mu}) + (\alpha(\mathbf{x}) + \beta(\mathbf{x})) u(\mathbf{x}, \boldsymbol{\mu}) v(\boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu} \\ &\quad = \int_{\mathbb{S}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}) u(\mathbf{x}, \boldsymbol{\mu}') v(\boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}\boldsymbol{\mu} + \int_{\mathbb{S}} f(\mathbf{x}, \boldsymbol{\mu}) v(\boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu} \end{aligned} \tag{2.21}$$

for all $v \in \mathcal{V}_{\mathbb{S}}$. Once boundary conditions for the mono-energetic LBTE are prescribed, an additional spatial discretisation may be applied to (2.21) to yield a full discretisation of (2.17) in both space and angle.

It is apparent that the computational difficulty of solving the resulting finite-dimensional problem (on selection of an appropriate spatial discretisation of (2.21)) drastically increases as the number of degrees of freedom in the angular space $\mathcal{V}_{\mathbb{S}}$ increases. It is perhaps for this reason that this approach has yet to be fully explored; works that have utilised this approach typically consider "quasi-3D" problems where the solution is assumed to be independent of one of the three spatial dimensions [17, 48, 67, 74]. These works typically employ a finite element discretisation in space, see Chapter 2.2.3 for more details.

The authors of these works develop and employ a preconditioning strategy based on the computation of the inverse of the discretised transport operator; see Chapter 2.5 for more details. This turns out to be nontrivial since, unlike the standard transport equation with constant advection field, the presence of the angular component means spatial elements (associated with the same angular element) become coupled and there is no natural ordering of spatial elements in which a sweeping preconditioner can be applied. Instead, they exploit Tarjan's strongly-connected components algorithm [96] to identify blocks of spatial elements for which a sweeping procedure can be applied.

A comparison between discrete ordinates methods and angular finite element methods can be made about their respective orders of approximation. Many discrete ordinates methods employ angular quadrature schemes designed to exactly integrate sets of high-order spherical harmonic functions; the resulting (semi)discrete solution is then viewed as a *global high-order* approximation (in angle) of the analytical solution. In contrast, the (semi)discrete solution arising from angular finite element methods may be considered as a *local low-order* approximation (in angle), since the angular mesh $\mathcal{T}_{\mathbb{S}}$ may not be rotationally-symmetric and the approximation is formed from comparatively low-degree polynomial approximations on each mesh element. However, finite element methods can handle global or local mesh refinement in a natural way, allowing for the adaptive generation of angular meshes designed to resolve the angular component of the solution around localised parts of the angular domain. For example, problems containing source terms with a strong directional bias (say, in the direction of $\boldsymbol{\mu}^*$) may require specially-designed angular meshes to resolve the solution around $\boldsymbol{\mu}^*$. On the other hand, the adaptive generation of quadrature schemes for use in discrete ordinates methods is typically more difficult.

The idea of discretising the angular domain in a finite element fashion is relatively new, and the resulting numerical method has received little attention in terms of *a priori* and *a posteriori* error analyses from a theoretical perspective. However, *a priori* analysis of a mixed finite element approach [40] has been conducted; in that work, this was achieved by a parity splitting in the angular domain.

In Chapter 3, we will introduce discontinuous Galerkin discretisations of the angular domain, as well as an efficient implementation of the resulting scheme as a discrete ordinates method of the form (2.18).

2.2.3 Spatial discretisation

Once the time-independent mono-energetic (or poly-energetic) LBTE has been semidiscretised in angle (and energy), the resulting field variables to solve for are functions of space only. The type of spatial problem left to solve depends heavily on the angular semidiscretisation employed. For example:

- if a discrete ordinates method is employed for the angular discretisation of the mono-energetic LBTE, the resulting problem is the system of first-order linear hyperbolic PDEs given in (2.18) for the angular flux u;
- if instead a P_1 spherical harmonics method is employed for the angular discretisation, the resulting problem is the second-order elliptic PDE given in (2.20) for the scalar flux ϕ.

We will primarily focus on the discretisation of the first-order linear hyperbolic PDE

defined by

$$\begin{aligned} \boldsymbol{\mu} \cdot \nabla u(\mathbf{x}) + a(\mathbf{x}) u(\mathbf{x}) &= f(\mathbf{x}) \quad \text{in } \Omega, \\ u(\mathbf{x}) &= g(\mathbf{x}) \quad \text{on } \partial_- \Omega, \end{aligned} \tag{2.22}$$

where the inflow boundary $\partial_- \Omega \subset \partial\Omega$ is defined by

$$\partial_- \Omega = \{\mathbf{x} \in \partial\Omega : \boldsymbol{\mu} \cdot \mathbf{n}(\mathbf{x}) < 0\},$$

where $\mathbf{n}$ denotes the outward unit normal to Ω on $\partial\Omega$.

Discontinuous Galerkin finite element methods (DGFEMs) The discontinuous Galerkin finite element method (DGFEM) developed by Reed and Hill [81] is one of the most popular numerical methods for transport problems of the form (2.22). Let $\mathcal{T}_\Omega$ denote a shape-regular subdivision of the spatial domain Ω into disjoint open elements $\kappa \in \mathcal{T}_\Omega$ such that $\bar{\Omega} = \cup_{\kappa \in \mathcal{T}_\Omega} \bar{\kappa}$. For each $\kappa \in \mathcal{T}_\Omega$ we assign a non-negative polynomial degree $p_\kappa \geq 0$ and write the shorthand vector $\mathbf{p} = (p_\kappa : \kappa \in \mathcal{T}_\Omega)$. We define the following function space:

$$\mathcal{V}_\Omega = \left\{ v \in L^2(\Omega) : v|_\kappa \in \mathbb{H}^{p_\kappa}(\kappa) \text{ for all } \kappa \in \mathcal{T}_\Omega \right\}.$$

Here, $\mathbb{H}^p(\kappa)$ denotes a space of polynomial functions defined on κ. The two most popular choices of this space are $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$, the space of polynomial functions on κ of maximal total degree p, and $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$, the space of polynomial functions on κ of maximal degree p in each of the d independent variables (in Cartesian coordinates). Finally, for a function v defined on the boundary $\partial\kappa$ of a spatial element $\kappa \in \mathcal{T}_\Omega$, we denote by v_κ^+ (resp. v_κ^-) the interior (resp. exterior) trace of v on $\partial\kappa$. It will always be understood on which element κ the notation applies; henceforth, the subscript κ shall be suppressed.

To this end, the full discontinuous Galerkin discretisation of (2.22) reads as follows: find $u_h \in \mathcal{V}_\Omega$ such that

$$A(u_h, v_h) = \ell(v_h) \tag{2.23}$$

for all $v_h \in \mathcal{V}_\Omega$, where

$$\begin{aligned} A(w_h, v_h) &= \sum_{\kappa \in \mathcal{T}_\Omega} \Bigg(- \boldsymbol{\mu} w_h \cdot \nabla v_h + a w_h v_h \ \mathrm{d}\mathbf{x} \\ &\qquad + \int_{\partial\kappa} H(w_h^+, w_h^-, \mathbf{n}_\kappa) v^+ \ \mathrm{d}s \Bigg), \\ \ell(v_h) &= \sum_{\kappa \in \mathcal{T}_\Omega} \int_\kappa f v_h \ \mathrm{d}\mathbf{x}. \end{aligned}$$

Here, $H(\cdot, \cdot, \cdot)$ denotes a numerical flux function. Denoting by Γ the set of faces of elements in $\mathcal{T}_\Omega$, the numerical flux function is typically chosen to have the following properties:

- **Consistency:** for the problem above, $H(\cdot,\cdot,\cdot)$ is *consistent* if we have $H(u^+,u^-,\mathbf{n}) = (\boldsymbol{\mu}\cdot\mathbf{n})u$ whenever u is a smooth function satisfying the inflow boundary condition;

- **Conservation:** $H(\cdot,\cdot,\cdot)$ is *conservative* if it is single-valued on Γ.

A common choice for the numerical flux function above for transport problems is the *upwind flux* given by

$$H(w^+,w^-,\mathbf{n}_\kappa)|_{\partial\kappa} = \begin{cases} \boldsymbol{\mu}\cdot\mathbf{n}_\kappa\ w^+(\mathbf{x},\boldsymbol{\mu},E) & \mathbf{x}\in\partial_+\kappa, \\ \boldsymbol{\mu}\cdot\mathbf{n}_\kappa\ w^-(\mathbf{x},\boldsymbol{\mu},E) & \mathbf{x}\in\partial_-\kappa\setminus\partial_-\Omega, \\ \boldsymbol{\mu}\cdot\mathbf{n}_\kappa\ g(\mathbf{x},\boldsymbol{\mu},E) & \mathbf{x}\in\partial_-\kappa\cap\partial_-\Omega, \end{cases}$$

Assuming that both the domain Ω and the mesh $\mathcal{T}_\Omega$ are polytopic, but otherwise arbitrary, finite element methods can handle complicated domain geometries in a straightforward manner. By tailoring the finite element spaces on each element, one can also generate approximate solutions with high-order accuracy. It is known [27, 54] that the convergence of finite element methods behaves like $O(h^s)$, where h denotes the spatial mesh size parameter and s denotes a function of the polynomial degree of approximation p and the smoothness of the exact solution. By exploiting hp-adaptivity, DGFEMs can achieve exponential convergence rates with respect to the number of degrees of freedom N employed in the method. It is known that, for elliptic problems, the DG-norm of the discretisation error scales like $O(e^{-bN^{\frac{1}{3}}})$ for two-dimensional problems [104] and $O(e^{-bN^{\frac{1}{5}}})$ for three-dimensional problems [87] for some $b>0$.

Adaptive finite element methods can also be applied to the problem of functional error estimation. Provided that one can prescribe an appropriate dual problem and output/goal functional $J(\cdot)$, such methods exploit computable *a posteriori* error estimators which provide information about the local accuracy of the computed primal and dual solutions [49, 52]. Examples of output functionals relevant to the field of radiation transport include the k_{eff}-eigenvalue arising from neutron transport criticality problems [48] and the deposited dose in photon radiotherapy [50].

Streamline upwind Petrov-Galerkin methods (SUPG) A typical weak formulation of a (linear) partial differential equation reads as follows: find $u\in\mathcal{V}$ such that

$$A(u,v) = \ell(v) \tag{2.24}$$

for all $v\in\mathcal{V}$, where $\mathcal{V}$ is typically a Hilbert space, $A:\mathcal{V}\times\mathcal{V}\to\mathbb{R}$ is a given bilinear form and $\ell:\mathcal{V}\to\mathbb{R}$ is a given linear functional. However, a typical finite element formulation reads as follows: find $u_h\in\mathcal{V}_\Omega$ such that

$$A_h(u_h,v_h) = \ell_h(v_h) \tag{2.25}$$

for all $v_h \in \mathcal{V}_\Omega$, where $A_h : \mathcal{V}_\Omega \times \mathcal{V}_\Omega \to \mathbb{R}$ and $\ell_h : \mathcal{V}_\Omega \to \mathbb{R}$ are possibly different from A and ℓ respectively. The finite element method is said to be *conforming* if the finite element space satisfies $\mathcal{V}_\Omega \subset \mathcal{V}$ and the discrete bilinear form and linear functional satisfy $A_h(w_h, v_h) = A(w_h, v_h)$ and $\ell_h(v_h) = \ell(v_h)$ for all $w_h, v_h \in \mathcal{V}_\Omega$. By selecting $v = v_h \in \mathcal{V}_\Omega$ in (2.24), we immediately have that

$$A(u, v_h) = \ell(v_h) \tag{2.26}$$

for all $v_h \in \mathcal{V}_\Omega$, and so we obtain the following Galerkin orthogonality result:

$$A(u - u_h, v_h) = 0 \tag{2.27}$$

for all $v_h \in \mathcal{V}_\Omega$. Conversely, the finite element method is said to be *non-conforming* if the finite element space satisfies $\mathcal{V}_\Omega \not\subset \mathcal{V}$; an example of such a method is the discontinuous Galerkin finite element method outlined earlier. For these methods, the condition (2.27) must be checked separately.

An example of a conforming finite element method for first-order transport equations is the streamline upwind Petrov-Galerkin method [42]. Starting from the transport equation with zero inflow boundary condition:

$$\begin{aligned} \boldsymbol{\mu} \cdot \nabla u + a u &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \Gamma_{in}, \end{aligned}$$

one multiplies by a test function of the form $v + \delta \boldsymbol{\mu} \cdot \nabla v$ for $v \in \mathcal{V}$, where the small mesh-dependent parameter $\delta > 0$ is typically chosen to satisfy $\delta|_\kappa = O(h_\kappa)$ for each $\kappa \in \mathcal{T}_\Omega$. Integrating the result over Ω yields the following variational problem: find $u \in \mathcal{V}$ such that

$$\int_\Omega \left(\boldsymbol{\mu} \cdot \nabla u + a u\right) \left(v + \delta \boldsymbol{\mu} \cdot \nabla v\right) \, \mathrm{d}\mathbf{x} = \int_\Omega f \left(v + \delta \boldsymbol{\mu} \cdot \nabla v\right) \, \mathrm{d}\mathbf{x}.$$

for all $v \in \mathcal{V}$. The streamline upwind Petrov-Galerkin method then follows by replacing $\mathcal{V}$ with a finite element space $\mathcal{V}_\Omega$ of *continuous* piecewise-polynomial functions with respect to a spatial mesh $\mathcal{T}_\Omega$. The streamline upwind Petrov-Galerkin method reads as follows: find $u_h \in \mathcal{V}_\Omega$ such that

$$\int_\Omega \left(\boldsymbol{\mu} \cdot \nabla u_h + a u_h\right) \left(v_h + \delta \boldsymbol{\mu} \cdot \nabla v_h\right) \, \mathrm{d}\mathbf{x} = \int_\Omega f \left(v_h + \delta \boldsymbol{\mu} \cdot \nabla v_h\right) \, \mathrm{d}\mathbf{x}.$$

for all $v \in \mathcal{V}$.

Streamline upwind Petrov-Galerkin methods have been used to numerically approximate the solutions of mono-energetic linear Boltzmann transport equations arising in astrophysics applications [43, 58].

Virtual element methods (VEMs) Virtual element methods (VEMs) can be thought of as an extension of finite element methods which are applicable to general

polytopic meshes [14], as well as an evolution of mimetic finite difference methods towards a finite element-like presentation [13]. The main point of departure of these methods from classical finite element methods is the choice of test and trial functions. Specifically, virtual element spaces are defined locally with respect to elements and contain, in addition to polynomial functions of maximal degree p, non-polynomial functions. These additional "virtual" functions need not be computed, but information about them can be inferred from particular carefully-chosen degrees of freedom - these are typically selected point-evaluations on the faces and vertices of elements as well as internal moments. In the elliptic case, by employing certain projection operators that are computable using only the specified degrees of freedom, elemental matrix contributions can be decomposed into a fully-computable term and a stabilising term that can be approximated without degrading the accuracy of the method.

The original formulation of the virtual element method was employed for the conforming discretisation of the Poisson problem, but non-conforming variants exist [30]. The virtual element method has also been studied in the context of discontinuous Galerkin methods [23] and applied to non-symmetric problems [16, 20].

2.3 Stochastic methods for radiation transport

While deterministic methods for the simulation of radiation transport through matter have attracted more attention in recent years [11, 99], it is important to note that radiation transport has almost always been modelled as a stochastic process by the wider medical physics community. There is a vast body of literature regarding Monte Carlo methods for radiotherapy treatment planning [9, 21, 56, 60, 61, 62, 71, 80, 101] covering topics including dosimetry, Monte Carlo code design and applications to realistic "phantoms" or domains.

In this setting, the macroscopic total cross-section $\sigma(\mathbf{x}) = \alpha(\mathbf{x}) + \beta(\mathbf{x})$ is interpreted as a scaled probability (per unit atom/electron density and per unit path length) that a particle will have an interaction while travelling a spatial distance, and all other macroscopic cross-sections (e.g. scattering and absorption) are interpreted as scaled probabilities that any given interaction is of that type [15, 90].

Monte Carlo codes for particle transport problems (e.g. for neutrons, photons and electrons) typically generate a large number of particle histories by individually following each particle through every interaction it experiences. The trajectory along which a given particle travels is essentially generated randomly (using macroscopic and differential cross-sections in the sampling process), and the history of each particle is assumed to be independent and identically distributed. The following is a very simplified example of how such a particle history for a mono-energetic simulation is generated.

Assume that a particle enters the system at a single point $\mathbf{x}_0 \in \Omega$ in space and

in an initial direction $\boldsymbol{\mu}_0 \in \mathbb{S}$. The particle will then travel a distance s along this direction until it undergoes its first interaction; this is dependent on the total cross-section $\sigma(s) = \sigma(\mathbf{x}_0 + s\boldsymbol{\mu}_0)$ along the trajectory, and is drawn from the probability distribution defined by the following probability density function[4]

$$f(s) = \begin{cases} \sigma(s)e^{-\int_0^s \sigma(s')\ \mathrm{d}s'} & s > 0, \\ 0 & s \leq 0. \end{cases}$$

Having chosen a random value of s, say s_0, set $\mathbf{x}_1 = \mathbf{x}_0 + s_0\boldsymbol{\mu}_0$. We now consider what type of interaction the particle undergoes at this point, namely one of the following processes:

- **Absorption/leakage:** the particle is removed from the system and its history recorded.
- **Scattering/fission:** a random scattering angle is generated from an associated probability distribution (based on the differential scattering cross-section $\theta(\boldsymbol{\mu}, \boldsymbol{\mu}')$) and the previous process of drawing a new traversal distance is repeated. In addition, any energy lost in the interaction is subtracted from the particle's energy and any new particles created/liberated during the interaction are allocated their own histories.

The process outlined above only requires one to generate sequences of random numbers, meaning that many particle histories can be generated very quickly. If the independent and identically-distributed random variables $X_1, X_2, \ldots, X_n$ represent the histories of n simulated particles, one might want to study, say, the (independent and identically-distributed) random variables $Y_i = h(X_i)$, where h is some real-valued function of interest (and which is defined for all values that each X_i can take). For example, h might denote the dose (per radiative particle) of radiation delivered to a particular region in the body. In particular, one may wish to take the estimated average:

$$Z_n = \frac{1}{n}\sum_{i=1}^{n} Y_i.$$

We may then take the expectation and variance of Z_n, under the assumption that the random variables X_i (and thus Y_i) are independent:

$$\mathbb{E}[Z_n] = \mathbb{E}[h(X_1)],$$
$$Var[Z_n] = \frac{1}{n}Var[h(X_1)],$$

[4]Notice that, in the case of a homogeneous medium, this is precisely the probability density function of an exponential random variable with rate parameter σ and mean σ^{-1}; in particular, the more optically-thick the medium is (i.e. the larger the rate parameter is), the shorter the distance between successive interactions. This probability distribution exhibits "memorylessness" - the probability that a particle will travel a distance greater than $s+t$ from where it started, given that it has already travelled a distance s, is equal to the probability that the same particle would have travelled a distance greater than t from its initial position.

k	$p_k = P\left(\mu - \frac{k\sigma}{\sqrt{n}} \leq Y_n \leq \mu + \frac{k\sigma}{\sqrt{n}}\right)$
1	0.6827
2	0.9545
3	0.9973

Table 2.1: Probabilities that a given normally-distributed random variable Y_n with mean μ and variance σ^2 takes a value within k standard deviations of the mean, for different values of $k > 0$.

where we have used the fact that the random variables Y_i are independent and identically-distributed. The calculations above tell us that the expected value of the mean of $h(X_i)$ over n sample histories is equal to the expected value of $h(X_i)$, and that the variance of the mean of $h(X_i)$ behaves like $O(n^{-1})$. Thus, as we take more and more particle histories, we can be more and more confident in the computed mean of h.

The observations above can be formalised by way of the *central limit theorem*. Suppose that X_1, X_2, ..., X_n are independent and identically-distributed random variables with mean μ and variance σ^2, and consider the sample mean

$$Y_n = \frac{1}{n}\sum_{i=1}^{N} X_i.$$

Then Y_n converges in distribution to a normal random variable with mean μ and variance $\frac{\sigma^2}{n}$; in notation, this reads

$$Y_n \rightarrow^d N\left(\mu, \frac{\sigma^2}{n}\right) \text{ as } n \rightarrow \infty.$$

Thus, as $n \rightarrow \infty$, we can think of Y_n as "approximately" normally-distributed, and use well-known properties of the normal distribution about the likelihood that a randomly-sampled average Y_n will be "sufficiently close" to the true mean μ. Since Y_n is random, deterministic notions of convergence (for example, *a priori* error bounds of the form $||u - u_h||_V \leq Ch^s$ in the case of finite element methods) make no sense in this context. Instead, the term "sufficiently close" will be taken to mean "to within k standard deviations of the true mean". The standard deviation of a distribution is taken as the square-root of its variance; thus, the standard deviation is a measure of spread. In the case of the normal distribution above, the standard deviation is given by $\frac{\sigma}{\sqrt{n}}$.

To this end, we consider the probability that Y_n takes a value in some confidence interval $I_k = \left(\mu - \frac{k\sigma}{\sqrt{n}}, \mu + \frac{k\sigma}{\sqrt{n}}\right)$, with k chosen such that the probability that Y_n fails to lie in this interval is small. Table 2.1 gives the probability that a randomly-sampled average lies in I_k for different values of k.

Fixing some value of k, and thus a confidence of how close Y_n is to μ, it is easy to see that the range of I_k decays as $O\left(\frac{1}{\sqrt{n}}\right)$ as $n \rightarrow \infty$. In clinical practice, it is often the case that a particular quantity of interest is sought to some level of confidence - this is

the motivation for selecting a sufficiently-large value of k such that p_k is "close enough" to 1. The arguments outlined above show that

$$|Y_n - \mu| \leq Cn^{-\frac{1}{2}} \text{ with probability } p_k \text{ as } n \to \infty$$

for some $C > 0$ dependent on k and σ. The stipulation "with probability p_k" makes this differ from similar convergence bounds one might derive for deterministic methods. The statement above means that, in order to halve the length of the confidence interval, one must take four times as many samples.

As mentioned earlier, the constant C appearing in the bound above is dependent on the variance of the random variables X_i. Indeed, a smaller variance results in a smaller value of C; it is therefore of interest to reduce the variance of the random variables in the hope of obtaining a sharper bound, whilst also maintaining the mean of the random variables. This can be done by way of variance reduction techniques [61]. In the case of neutron transport codes, this is achieved by changing the sampling process outlined above; namely, by altering the probability distributions from which random collision distances/interaction types are drawn, and weighting the terms appearing in the sample average according to their "importance" (in some sense) to a quantity of interest. This is known as "importance sampling" [90] and is typically achieved by introducing the *importance* or *adjoint* function, which is the solution to an associated adjoint PDE.

The computational costs associated with both stochastic and deterministic methods for the numerical solution of the LBTE have been studied [22]. In that work, an attempt to find relations of the form $\mathcal{E} = O(\mathcal{W}^{-q})$ between the error in a computation $\mathcal{E}$ and the work $\mathcal{W}$ (defined by the number of arithmetic operations); the parameter q then quantifies how fast the error decays as more computational effort is expended[5]. Comparisons of different values of q between Monte Carlo methods and grid-based deterministic methods suggested that high-order deterministic methods may yield smaller error estimates than Monte Carlo methods for the same amount of work; however, this did not take into consideration the effect of using different computer architectures. Meanwhile, an empirical comparison was made in [45] which found that both the CPU time and dose calculation of a finite element implementation of the LBTE compared favourably against those of Monte Carlo simulations.

[5]For Monte Carlo simulations, $\mathcal{E}$ instead denotes the standard deviation of the error, centred at zero, and $\mathcal{W}$ denotes the standard deviation of the work.

2.4 Polytopic methods for finite element discretisations

A typical finite element method is performed by first rewriting the underlying PDE in a weak/variational form reading as follows: find $u \in \mathcal{X}$ such that

$$A(u, v) = \ell(v)$$

for all $v \in \mathcal{Y}$. Here, $\mathcal{X}$ and $\mathcal{Y}$ are (usually Banach or Hilbert) spaces of functions (which may be the same) inferred from the original PDE problem. A finite element formulation follows upon taking finite-dimensional subspaces of $\mathcal{X}$ and $\mathcal{Y}$ - typically these subspaces will contain piecewise-polynomial functions defined on an underlying triangulation $\mathcal{T}_\Omega$ of the computational domain Ω, which we shall assume is polytopic and exactly covered by the elements $\kappa \in \mathcal{T}_\Omega$. For simplicity, we suppose that $\mathcal{Y} = \mathcal{X} \subset H^k(\Omega)$, where $H^k(\Omega)$ denotes the Sobolev space of square-integrable functions on Ω whose weak derivatives up to order k are also square-integrable. In this setting, a typical finite element discretisation will replace the infinite-dimensional space $\mathcal{X}$ with a finite-dimensional subspace $\mathcal{V}_\Omega$. The finite element formulation then reads as follows: find $u_h \in \mathcal{V}_\Omega$ such that

$$A_h(u_h, v_h) = \ell_h(v_h)$$

for all $v_h \in \mathcal{V}_\Omega$. Here, $A_h(\cdot, \cdot)$ and $\ell_h(\cdot)$ denote (possibly inexact) replacements of $A(\cdot, \cdot)$ and $\ell(\cdot)$ respectively in the original weak formulation of the PDE and may depend on discretisation parameters.

The finite element space $\mathcal{V}_\Omega$ is frequently chosen to contain piecewise-polynomial functions of a given degree p on each element $\kappa \in \mathcal{T}_\Omega$ (which may be continuous or discontinuous across element boundaries, depending on the method chosen). Such a test space[6] may, for example, be defined as follows:

$$\mathcal{V}_\Omega = \{v \in L^2(\mathcal{T}_\Omega) : v|_\kappa \in \mathbb{P}^p(\kappa) \text{ for all } \kappa \in \mathcal{T}_\Omega\}.$$

One can select a basis for $\mathcal{V}_\Omega$ since it is finite-dimensional; consequently, under the assumption that $A_h : \mathcal{V}_\Omega \times \mathcal{V}_\Omega \to \mathbb{R}$ is a bilinear form and $\ell_h : \mathcal{V}_\Omega \to \mathbb{R}$ is a linear functional, a matrix system results for the coefficients of the solution u_h expanded in the finite element basis:

$$\mathbf{A}\mathbf{u} = \boldsymbol{\ell}.$$

If $\mathcal{V}_\Omega = \text{span}\{\phi_i\}_{i=1}^N$, $N = \dim \mathcal{V}_\Omega$, with each ϕ_i a basis function, then the matrix $\mathbf{A}$ has coefficients $A_{ij} = A_h(\phi_j, \phi_i)$ and the vector $\boldsymbol{\ell}$ has coefficients $\ell_i = \ell_h(\phi_i)$.

It is therefore apparent that, in order to construct the matrix system, one must be able to integrate polynomial functions over the elements of the mesh $\mathcal{T}_\Omega$. Typically,

[6] Note that this test space is suitable for discontinuous Galerkin methods, *not* for continuous Galerkin methods.

$\mathcal{T}_\Omega$ consists of simplicial elements (triangles in 2D, tetrahedra in 3D) or tensor-product elements (rectangles in 2D, cuboids in 3D). For these shapes, high-order quadrature schemes exist that integrate any polynomial function of maximal degree p exactly.

Recently, attention has turned towards the employment of general polytopic meshes in the finite element framework. Such meshes can be formed via agglomeration of a fine mesh [5, 28] or via Voronoi tessellations restricted to the domain of interest [95]. This has been motivated in part by the desire to capture fine geometrical features within the computational domain whilst reducing the number of elements in the domain partition (and therefore number of degrees of freedom in the resulting matrix system) [28].

A key computational task is therefore to integrate finite element basis functions (i.e. polynomial functions) on an arbitrary polytopic domain; we shall assume that the coefficients arising in the PDE problem are piecewise-constant with respect to $\mathcal{T}_\Omega$. One solution is to further triangulate each polytopic element in $\mathcal{T}_\Omega$ into standard element shapes (simplices or tensor-product elements) and employ a high-order quadrature on each subelement [93]. An example of such a quadrature scheme is illustrated in Figure 2.4. This can be expensive in the context of the assembly of the matrix system arising from a finite element discretisation, particularly if element geometries are complex. In two spatial dimensions, the time taken to generate a triangulation of a simple polygon grows linearly with the number of faces of the polygon [31]. Furthermore, if the polytopic mesh $\mathcal{T}_\Omega$ arises from an agglomeration of elements of a fine (simplicial or tensor-product) mesh $\mathcal{T}_\Omega^{fine}$ (and each $\kappa \in \mathcal{T}_\Omega$ inherits its quadrature scheme from the underlying fine-mesh elements comprising it), then the assembly of the system of equations on $\mathcal{T}_\Omega$ is no faster than on $\mathcal{T}_\Omega^{fine}$ [6], though the resulting system will be smaller.

Another approach is to use moment-based quadrature rules [73]. In these quadrature schemes, one first establishes an upper bound for the number of quadrature points required to integrate (restrictions of) polynomial functions over a given polygon $\mathcal{P}$ up to some given degree - this can be achieved by generating the quadrature scheme described above. This serves as an initial guess for an iterative algorithm designed to generate a quadrature scheme over $\mathcal{P}$ which is exact for polynomials of maximal degree p with as few quadrature points/weights as possible. At each iteration, the quadrature point/weight with least "importance" is discarded from the scheme and the remaining points/weights are adjusted to satisfy a moment-fitting criterion - this takes the form of an algebraic system of equations to solve numerically. The algorithm returns a near-optimal quadrature scheme on $\mathcal{P}$ when the moment-fitting criterion can no longer be satisfied by the set of remaining quadrature points/weights. In the context of a finite element assembly implementation, this can become very expensive since this procedure must be carried out for each unique (non-affine) element shape in $\mathcal{T}_\Omega$. Furthermore, there is no guarantee that quadrature points will lie in $\mathcal{P}$, nor that their weights are nonnegative - this potentially compromises the numerical stability of the quadrature

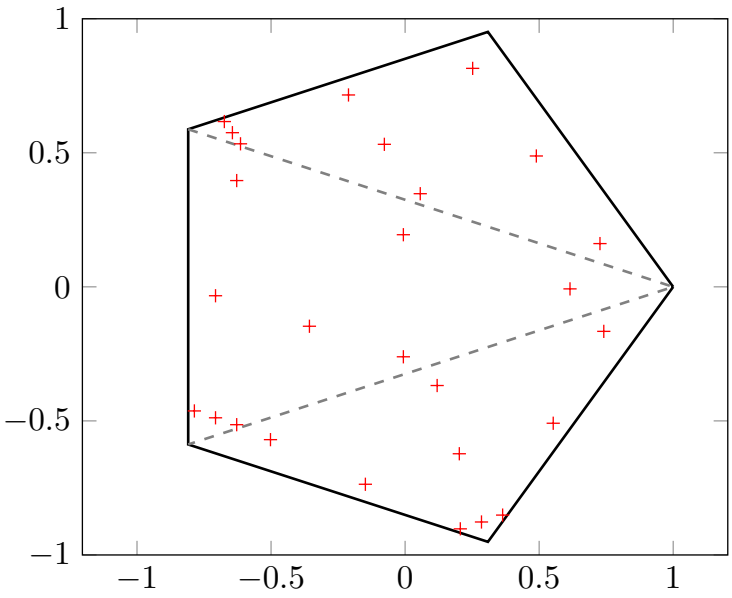

Figure 2.4: Example of a quadrature scheme on a polygon generated via subtriangulation. Here, the domain is taken to be a regular pentagon and the quadrature scheme exactly integrates quartic polynomial functions defined on the domain.

scheme.

In each case, the computational cost of using each quadrature strategy becomes prohibitively large, both as the mesh size decreases and as the geometry of elements in $\mathcal{T}_\Omega$ becomes more complex. One key idea that has been exploited is to restrict attention to *homogeneous* functions [69]. A function $f : \mathbb{R}^d \setminus \{\mathbf{0}\} \to \mathbb{R}$ is said to be *(positively) homogeneous of degree* $k \in \mathbb{R}$ if, for all $\alpha > 0$, we have that $f(\alpha\mathbf{x}) = \alpha^k f(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$; furthermore, if $k > 0$, then this results extends to all functions $f : \mathbb{R}^d \to \mathbb{R}$. It is easily seen that any polynomial functions of degree p can be decomposed into a sum of at most $p + 1$ homogeneous functions.

Within the context of integration of homogeneous functions over polytopic domains, *Euler's homogeneous function theorem* proves fundamental. This theorem states that, for any continuously differentiable function $f : \mathbb{R}^d \setminus \{\mathbf{0}\} \to \mathbb{R}$, f is positively homogeneous of degree k if and only if $kf(\mathbf{x}) = \mathbf{x} \cdot \nabla f(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$.

For any polytopic domain $\mathcal{P} \in \mathbb{R}^d$, we denote by $\{\partial\mathcal{P}_i\}_{i=1}^m$ the set of $(d-1)$-dimensional planar boundary faces on which $\mathbf{x} \cdot \mathbf{n}_i = a_i \in \mathbb{R}$ for all $\mathbf{x} \in \partial\mathcal{P}_i$, where $\mathbf{n}_i$ denotes the outward unit normal to $\mathcal{P}$ on $\partial\mathcal{P}_i$. It is straightforward to show that any positively homogeneous function $f : \mathcal{P} \to \mathbb{R}$ of degree k satisfies

$$\int_{\mathcal{P}} f(\mathbf{x}) \ \mathrm{d}\mathbf{x} = \frac{1}{d+k} \sum_{i=1}^{m} a_i \int_{\partial\mathcal{P}_i} f(\mathbf{x}) \ \mathrm{d}s.$$

That is, the integral of a homogeneous function over an d-dimensional polytope can be rewritten as a linear combination of integrals of the *same* homogeneous function over its $(d-1)$-dimensional (polytopic) boundary faces. With a slight modification, one can

continue to reduce each boundary integral to an expression involving a weighted sum of integrals over its corresponding $(d-2)$-dimensional (polytopic) edges [6]. This forms the basis of a recursive algorithm that computes the volume integral of a homogeneous function f over $\mathcal{P}$ based on the weighted sum of evaluations of f at its vertices; this algorithm is given in [6]. A non-recursive implementation of this algorithm for three-dimensional polytopic elements is given in [34], where point evaluations of all monomials are taken at each vertex, followed by the computation of all line integrals along every one-dimensional edge, followed by the computation of all surface integrals over every face, and finalised with the computation of all volume integrals over the polyhedron.

It was found that this numerical integration technique is substantially faster than the subtessellation approach within the context of finite element methods [6], and that the computational speedup was much more significant in the evaluation of volume integrals than face integrals for an interior penalty DGFEM discretisation of Poisson's equation.

This idea has also been applied in more general settings. For example, [69] derives expressions when the integrand is weighted with elementary non-homogeneous functions; in that case, closed-form expressions of volume integrals in terms of face integrals exist only for certain integrands. Another extension of this idea is given in [33], where the boundary faces of $\mathcal{P}$ are no longer required to be planar/polytopic; instead, each face $\partial\mathcal{P}_i$ need only be prescribed in the form $h_i(\mathbf{x}) = a_i \in \mathbb{R}$, where each h_i is a homogeneous function. The idea presented there is applied to the case where each face admits a representation in terms of polar coordinates, and numerical results are obtained by first reducing the volume integrals to face integrals and performing a quadrature on each face integral.

In Chapter 4, we introduce a quadrature-free approach to the assembly of linear systems arising from discontinuous Galerkin discretisations of the linear, first-order and constant-coefficient transport equation. We also present a floating-point operation analysis of a general assembly method using quadrature-free integration and compare the results against standard quadrature-based methods.

2.5 Linear solvers for discretised radiation transport problems

The discretisation of the time-independent, poly-energetic linear Boltzmann transport equation (2.6) (and the mono-energetic LBTE (2.17)) using any of the methods outlined earlier typically yields a large and sparse linear system of equations. This is in part due to the high dimensionality on which the LBTE is posed. Even in the mono-energetic setting, the angular flux $u(\mathbf{x}, \boldsymbol{\mu})$ is a function of $2d-1$ independent variables (d spatial and $d-1$ angular variables) for $d = 2, 3$. For the remainder of this discussion, we will focus on schemes for the mono-energetic LBTE employing discrete ordinate discretisa-

tions in angle and discontinuous Galerkin discretisations in space. In Chapter 5, we will study the convergence of iterative methods applied to variational problems, with a particular focus on discretisations of the mono-energetic LBTE using DGFEMs in both the spatial and angular domains.

The mono-energetic LBTE may be written abstractly in an operator form:

$$\mathcal{T}u = \mathcal{S}u + f,$$

where the operators $\mathcal{T}$ and $\mathcal{S}$ act on functions $v : \Omega \times \mathbb{S} \to \mathbb{R}$ by:

$$\begin{aligned}\mathcal{T}v &= \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} v(\mathbf{x}, \boldsymbol{\mu}) + (\alpha(\mathbf{x}, \boldsymbol{\mu}) + \beta(\mathbf{x}, \boldsymbol{\mu}))\, v(\mathbf{x}, \boldsymbol{\mu}),\\ \mathcal{S}v &= \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}) v(\mathbf{x}, \boldsymbol{\mu}')\ \mathrm{d}\boldsymbol{\mu}'.\end{aligned}$$

When a discrete ordinates scheme employing an ordinate set $\{\boldsymbol{\mu}_q\}_{q=1}^N$ is used for the angular discretisation and a discontinuous Galerkin scheme is used for the spatial discretisation, one obtains a linear system of the form

$$\mathbf{T}\mathbf{u} = \mathbf{S}\mathbf{u} + \mathbf{f}.$$

The matrices $\mathbf{T}$ and $\mathbf{S}$ admit a block structure. The matrix $\mathbf{T} = \mathrm{diag}_{q=1}^N(\mathbf{T}_q)$ can be written as a block-diagonal matrix whose on-diagonal blocks $\{\mathbf{T}_q\}_{q=1}^N$ are precisely the matrices arising from DGFEM discretisations (2.22) of the transport problem (2.22) associated with the ordinate directions $\{\boldsymbol{\mu}_q\}_{q=1}^N$. The matrix $\mathbf{S}$, on the other hand, can be partitioned as

$$\mathbf{S} = \begin{pmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} & \dots & \mathbf{S}_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_{N1} & \mathbf{S}_{N2} & \dots & \mathbf{S}_{NN} \end{pmatrix},$$

where $\mathbf{S}_{ij}$ denotes the (spatial) mass matrix associated with the weight function $w_i\theta(\mathbf{x}, \boldsymbol{\mu}_j \to \boldsymbol{\mu}_i)$.

Owing to the size of the matrix $\mathbf{T} - \mathbf{S}$, direct solvers such as Gaussian elimination become prohibitively expensive, both in terms of the number of floating-point operations required to compute $(\mathbf{T} - \mathbf{S})^{-1}$ and in terms of storage of the inverse. Therefore, a number of iterative techniques have been employed to solve these equations - an overview is given in [1].

2.5.1 Source iteration

One of the most common methods for solving the discrete LBTE system is *source iteration* (SI). The method exploits the splitting of the (continuum) operator $\mathcal{T} - \mathcal{S}$ into a streaming operator $\mathcal{T}$ and a scattering operator $\mathcal{S}$. As remarked earlier, discrete ordinates discretisations of the operator $\mathcal{T}$ generally yield block-diagonal system matrices with each block corresponding to a single spatial transport problem. On the other hand,

such discretisations of the operator $\mathcal{S}$ yield comparatively more dense system matrices. For this reason, the action of $(\mathcal{T}-\mathcal{S})^{-1}$ is generally more difficult to compute than that of $\mathcal{T}^{-1}$, the latter of which can be performed using a direct solver. The source iteration method computes a sequence of approximations $\{u_n\}_{n=0}^{\infty}$ generated from an initial guess u_0 (typically chosen to be zero) via the iteration

$$\mathcal{T}u_{n+1} = \mathcal{S}u_n + f \quad \text{for all } n \geq 0.$$

This solution method can also be applied to the discrete equations: starting from an initial guess $\mathbf{u}_0$, we may construct the sequence of approximations $\{\mathbf{u}_n\}_{n=0}^{\infty}$ via the iteration

$$\mathbf{T}\mathbf{u}_{n+1} = \mathbf{S}\mathbf{u}_n + \mathbf{f} \quad \text{for all } n \geq 0.$$

We remark that the action of $\mathbf{T}^{-1}$ on a vector is generally computationally easier than the action of $(\mathbf{T}-\mathbf{S})^{-1}$ on a vector, owing to the block-diagonal structure of $\mathbf{T}$ and the fact that the action of the inverse of each $\mathbf{T}_q$ on a vector can be performed in a sweeping fashion. When a DGFEM discretisation is employed in space, the action of $\mathbf{T}_q^{-1}$ can be performed on an element-by-element basis by first ordering the elements κ of the spatial mesh $\mathcal{T}_\Omega$ according to the wind direction using Tarjan's strongly-connected components algorithm [74]; see Figure 2.5.

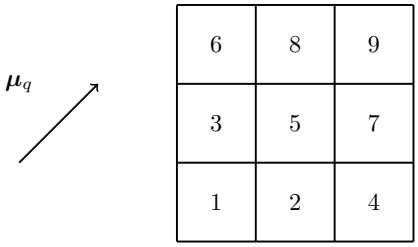

Figure 2.5: Plot of 3-by-3 square mesh $\mathcal{T}_\Omega$ and given wind direction $\boldsymbol{\mu}_q$. Tarjan's strongly-connected components algorithm is used to topologically sort the elements of $\mathcal{T}_\Omega$ with respect to the wind direction. The action of $\mathbf{T}_q^{-1}$ can be evaluated on an element-wise basis by looping through this ordering. In this way, the transport problem on element κ_Ω is only assembled and solved once the solutions on all elements adjacent to its inflow boundary $\partial_-\kappa(\boldsymbol{\mu}_q)$ are known.

It can be shown that source iteration applied to the continuum problem is convergent provided that the (global) *scattering ratio*, denoted by c and defined by

$$c = \operatorname*{ess\,sup}_{\mathbf{x}\in\Omega} \frac{\beta(\mathbf{x})}{\alpha(\mathbf{x})+\beta(\mathbf{x})},$$

satisfies $c < 1$ [1]. That is, we have $u_n \to u$ as $n \to \infty$, where u denotes the analytical solution of the LBTE. Moreover, the same condition on c ensures that source iteration applied to the discrete problem is also convergent; that is, we have $\mathbf{u}_n \to \mathbf{0}$ as $n \to \infty$.

In the continuum and infinite-medium setting, the spectral radius $\rho(\mathcal{T}^{-1}\mathcal{S})$ of the iteration operator $\mathcal{T}^{-1}\mathcal{S}$ characterises the rate of convergence of source iteration and can be shown to satisfy $\rho(\mathcal{T}^{-1}\mathcal{S}) = c$. That is, source iteration is rapidly convergent for $c \approx 0$ but stagnates for $c \approx 1$. In the discrete and finite-medium setting, the spectral radius $\rho(\mathbf{T}^{-1}\mathbf{S})$ of the iteration operator $\mathbf{T}^{-1}\mathbf{S}$ satisfies $\rho(\mathbf{T}^{-1}\mathbf{S}) \leq c$. This is because radiative particles can only leave the system via absorption in the infinite-medium case, whereas particles can additionally leave the system via exiting the domain in the finite-medium case [1].

2.5.2 Diffusion-synthetic acceleration

One of the most popular techniques to remedy the slow convergence of source iteration is *diffusion-synthetic acceleration* (DSA) [103]. The key idea is that source iteration effectively suppresses the error modes with strong spatial and angular dependence, and thus that stagnation of source iteration is associated with the slow decay of error modes with weak spatial and angular dependence; this can be demonstrated through Fourier analysis [1, 103].

At the n^{th} step of the DSA method, a single source iteration is employed to generate an intermediate angular flux $u_{n+1/2}$:

$$\mathcal{T}u_{n+1/2} = \mathcal{S}u_n + f. \tag{2.28}$$

Writing the exact angular flux as u, we would like to write an equation for the difference $\delta_{n+1/2} \approx u - u_{n+1/2}$ so that the updated flux

$$u_{n+1} = u_{n+1/2} + \delta_{n+1/2}$$

is a much better approximation to u than $u_{n+1/2}$ is.

Through some algebraic manipulation, one can find an equation for the difference $u - u_{n+1/2}$:

$$(\mathcal{T} - \mathcal{S})(u - u_{n+1/2}) = \mathcal{S}(u_{n+1/2} - u_n).$$

The spherical harmonics method can now be applied to obtain an approximation $\delta_{n+1/2}$ of the angular flux difference $u - u_{n+1/2}$. In the simplest case of isotropic scattering, one may obtain the diffusion equation (2.20) with $\phi(\mathbf{x}) = \delta_{n+1/2}(\mathbf{x})$ and source term

$$\begin{aligned} f_{0,0}(\mathbf{x}) &= \theta_0(\mathbf{x}) \int_{\mathbb{S}} \left(u_{n+1/2}(\mathbf{x}, \boldsymbol{\mu}) - u_n(\mathbf{x}, \boldsymbol{\mu})\right) \, \mathrm{d}\boldsymbol{\mu} \\ &= \theta_0(\mathbf{x}) \left(\phi_{0,0}^{(n+1/2)}(\mathbf{x}) - \phi_{0,0}^{(n)}(\mathbf{x})\right), \end{aligned}$$

where we have defined by $\phi_{0,0}^{(n)}$ the scalar flux associated with the angular flux u_n and defined as in (2.5) with $\phi = \phi_{0,0}$ and $u = u_n$. Explicitly, the DSA correction equation in the isotropic scattering case is given by

$$-\nabla \cdot \left(\frac{1}{3(\alpha(\mathbf{x}) + \beta(\mathbf{x}))} \nabla \delta_{n+1/2} \right) + \alpha(\mathbf{x}) \delta_{n+1/2}(\mathbf{x}) = \theta_0(\mathbf{x}) \left(\phi_{0,0}^{(n+1/2)}(\mathbf{x}) - \phi_{0,0}^{(n)}(\mathbf{x}) \right).$$

Fourier analysis of the DSA method on an infinite medium shows that this new process yields an iteration operator $\mathcal{G}$ with a spectral radius $\rho(\mathcal{G}) \approx 0.2247c$ [1]. This means that, even in scattering-dominated regimes ($c \approx 1$), DSA is rapidly convergent. However, a practical challenge arises when one seeks to discretise both the LBTE and the DSA correction equation - if the discretisation of the diffusion equation is not "consistent" with that of the transport equation, then the rapid convergence of DSA may be lost [1, 4, 103]. By "consistent" we mean the following:

- the DSA discretisation is *stable*; that is, the discrete iteration operator $\mathbf{G}$ satisfies $\rho(\mathbf{G}) < 1$ for all cell aspect ratios;
- the DSA discretisation is *effective*; that is, $\rho(\mathbf{G}) \leq \rho_0 < 1$ for all cell aspect ratios (with ρ_0 an analytically-derived spectral radius when a particular angular quadrature is used for the initial source-iteration step).

In other words, a "consistent" discretisation of the DSA correction equation cannot result in stagnation of the DSA iterations (in the case where scattering is isotropic).

The DSA step in the fully-consistent method of Warsa, Wareing and Morel [103] can be thought of as a discontinuous Petrov-Galerkin discretisation of the diffusion method outlined earlier. This can become prohibitively expensive to solve, particularly in three dimensions, so a number of so-called "partially consistent" methods have been proposed. These are stable methods that are not necessarily effective, but whose spetral radii are still bounded away from one [102, 103]. DSA discretisations that respect optical thickness of the medium (which can be loosely described by the magnitude of the macroscopic total cross-section) and the cell aspect ratio of the mesh (which can be thought of as the product of the mesh size parameter and the macroscopic total cross-section) have also been studied [89].

While DSA is a rapidly-convergent iterative method for the numerical solution of the LBTE with isotropic scattering, its performance can deteriorate when highly-peaked scattering kernels are present. One idea around this is to perform an "angular multigrid" method, whereby source iteration steps are repeatedly performed with ordinate sets of decreasing size and culminate in a final DSA correction [1].

2.5.3 GMRES

Recently, the generalised minimal residual (GMRES) method of Saad and Schulz [83, 84] has been proposed as an alternative to DSA for the solution of the discretised time-

independent, mono-energetic LBTE [77]. It is a member of the family of Krylov subspace methods for large and sparse linear systems of the form $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{C}^{N\times N}$ and $\mathbf{b} \in \mathbb{C}^N$. GMRES is often used as an iterative solver as it generates a sequence of approximate solutions $\{\mathbf{x}_n\}_{n\geq 0} \subset \mathbb{C}^N$ for a given initial guess $\mathbf{x}_0$, but in exact arithmetic it is a direct solver since it returns the exact solution after N iterations.

Introducing the Krylov subspaces $\mathcal{K}_n$ for a given initial guess $\mathbf{x}_0$ defined by

$$\mathcal{K}_n = \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots, \mathbf{A}^{n-1}\mathbf{r}_0\},$$

where $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$ denotes the initial residual vector, the n^{th} approximate solution vector $\mathbf{x}_n$ solves the following residual-minimisation problem:

$$\mathbf{x}_n = \underset{\mathbf{y}\in\mathbf{x}_0+\mathcal{K}_n}{\arg\min} ||\mathbf{b} - \mathbf{A}\mathbf{y}||_2.$$

Preconditioned versions of GMRES have been studied in the context of radiation transport [77], where the system matrix $\mathbf{A} = \mathbf{T} - \mathbf{S}$. These authors focussed on two preconditioning strategies: one based on the streaming/transport part of $\mathbf{A}$, and one based on incomplete LU (ILU) factorisations of $\mathbf{A}$. It was found that the convergence rates of both strategies are comparable to DSA in some test cases. However, the CPU time taken by the ILU-preconditioned GMRES method scaled nonlinearly with respect to the number of energy groups and angular quadrature order, whereas the transport-preconditioned GMRES method scaled linearly with respect to these quantities.

GMRES in Hilbert spaces It is worth noting that Krylov subspace methods can be extended to functional settings; i.e. tailored to functional equations of the form $\mathcal{A}x = b$, where $x \in X$, $\mathcal{A} \in \mathcal{L}(X, X^*)$ (assumed invertible) and $b \in X^*$ [44, 47]. Here, X is a Hilbert space and $\mathcal{L}(X, X^*)$ denotes the space of linear operators from X into its dual space X^*. Furthermore, one introduces a duality pairing $\langle\cdot,\cdot\rangle_{X,X^*}$ between X and X^* and an inner product $(\cdot,\cdot)_M$ on X defined for all $u, v \in X$ by $(u, v)_M = \langle u, Mv\rangle$, where $M \in \mathcal{L}(X, X^*)$ denotes the Riesz isomorphism from $(X, ||\cdot||_M)$ to $(X^*, ||\cdot||_{M^{-1}})$.

In this setting, residual vectors are replaced with residual *functionals* $r_n = b - \mathcal{A}x_n \in X^*$. One subtle departure from the linear-algebraic formulation of GMRES is the introduction of a right-preconditioning operator $\mathcal{P} \in \mathcal{L}(X, X^*)$, which is necessary for the definition of the Krylov subspace $\mathcal{K}_n$:

$$\mathcal{K}_n = \text{span}\left\{r_0, \mathcal{A}\mathcal{P}^{-1}r_0, \ldots, (\mathcal{A}\mathcal{P}^{-1})^{n-1}r_0\right\}.$$

The elements of the sequence of approximations $\{x_n\}_{n\geq 0} \subset X$ are defined as the solutions to the residual-minimisation problems

$$x_n = \underset{y\in x_0+\mathcal{K}_n}{\arg\min} ||b - \mathcal{A}\mathcal{P}^{-1}y||_{M^{-1}}.$$

The extension of GMRES in this fashion is in part motivated by the free choice of the scalar product $(\cdot,\cdot)_M$ in X, which fixes the choice of the norm $||\cdot||_{M^{-1}}$ in which

GMRES minimises $||b-\mathcal{A}x_n||_{M^{-1}}$. In many GMRES implementations for linear systems of the form $\mathbf{Ax} = \mathbf{b}$ (for which $X = \mathbb{C}^N$), the choice of the scalar product in X is fixed to the standard Euclidean product of vectors in $\mathbb{C}^N$, although generalisations to other scalar products in X have been studied [32]. In Chapter 5, we discuss how standard GMRES implementations can be used to minimise linear solver residual errors measured in norms other than the standard Euclidean norm, and in particular how residual-based *a posteriori* solver error estimates arising from finite element discretisations of PDEs can be used to prematurely terminate GMRES solvers.

Chapter 3

Discontinuous Galerkin Discretisation of the Time-Independent Linear Boltzmann Transport Equation

In Chapter 2, we introduced the linear Boltzmann transport equation (LBTE), a seven-dimensional partial integro-differential used to model the fluence of a radiative particle species, as well as a number of simplifications and special cases commonly studied in radiation transport literature. We also reviewed a number of techniques employed in the discretisation of the time-independent LBTE in each of the spatial, angular and energetic domains. In particular, we noted that the multigroup discretisation of the energetic domain may require a large number of energy groups in order to resolve the energetic component of the solution. Moreover, multigroup discrete ordinates methods are not well-suited to local angular and energetic mesh refinement.

In this chapter, we will seek to discretise the time-independent, poly-energetic LBTE using high-order discontinuous Galerkin (DG) methods in space, angle and energy. We start by carefully prescribing meshes and finite element function spaces for each of the spatial, angular and energetic domains. A full discontinuous Galerkin finite element method (DGFEM) for the poly-energetic LBTE is presented, which may be used to derive DGFEMs for the mono-energetic LBTE as well as linear first-order transport equations with constant wind direction.

The resulting DGFEM scheme applied to the poly-energetic LBTE is shown to be sta-

ble under reasonable assumptions on the cross-sectional data. The scheme is also proven to be convergent, and optimal-order convergence in an associated problem-dependent norm is demonstrated through poly- and mono-energetic numerical examples.

Finally, we discuss how the scheme may be implemented in an efficient fashion. By carefully selecting the angular and energetic basis functions, we show that the scheme can be written as a classical multigroup discrete ordinates method without compromising high-order accuracy in the angular and energetic domains.

This chapter is based on the work carried out in [53]. In particular, the author is responsible for both the collection of numerical results and the technical discussions concerning the (poly-energetic) scattering operator and its effect on the coercivity result for the space-angle-energy DGFEM scheme.

3.1 Model Problems

Let $\Omega \subset \mathbb{R}^d$ denote an open and bounded polyhedral spatial domain, $d = 2, 3$, and let $\partial\Omega$ denote the union of its $(d-1)$-dimensional open faces. Let $\mathbb{S} = S^{d-1} = \{\boldsymbol{\mu} \in \mathbb{R}^d : ||\boldsymbol{\mu}||_2 = 1\}$ denote the angular domain, where $||\cdot||_2$ denotes the Euclidean norm on $\mathbb{R}^d$, and let $\mathbb{Y} = \{E \in \mathbb{R} : E > 0\}$ denote the energetic domain. Let $\mathcal{D} = \Omega \times \mathbb{S} \times \mathbb{Y}$ denote the space-angle-energy domain, and define the inflow boundary $\Gamma_{in} = \Gamma_{in}(\partial\Omega \times \mathbb{S})$ of the space-angle domain $\partial\Omega \times \mathbb{S}$ by

$$\Gamma_{in} = \{(\mathbf{x}, \boldsymbol{\mu}) \in \partial\Omega \times \mathbb{S} : \mathbf{n}(\mathbf{x}) \cdot \boldsymbol{\mu} < 0\}, \tag{3.1}$$

where $\mathbf{n}(\mathbf{x})$ denotes the outward unit normal vector to Ω on $\partial\Omega$. For a given (constant) wind direction $\boldsymbol{\mu} \in \mathbb{R}^d$, let $\partial_+\Omega(\boldsymbol{\mu})$ and $\partial_-\Omega(\boldsymbol{\mu}) = \Gamma_{in}(\Omega; \boldsymbol{\mu})$ be defined by

$$\partial_+\Omega(\boldsymbol{\mu}) = \{\mathbf{x} \in \partial\Omega : \mathbf{n}(\mathbf{x}) \cdot \boldsymbol{\mu} \geq 0\}, \tag{3.2}$$

$$\partial_-\Omega(\boldsymbol{\mu}) = \{\mathbf{x} \in \partial\Omega : \mathbf{n}(\mathbf{x}) \cdot \boldsymbol{\mu} < 0\}. \tag{3.3}$$

Here, $\partial_+\Omega(\boldsymbol{\mu}) \subset \partial\Omega$ (resp. $\partial_-\Omega(\boldsymbol{\mu}) \subset \partial\Omega$) denotes the outflow boundary (resp. inflow boundary) of Ω corresponding to the wind direction $\boldsymbol{\mu}$. When discussing inflow boundaries (either of the full spatial domain Ω or on open polytopic subsets of Ω), the dependence on $\boldsymbol{\mu}$ will always be made explicit. Thus, for any $(\mathbf{x}, \boldsymbol{\mu}) \in \Gamma_{in}$, we have that $\mathbf{x} \in \partial_-\Omega(\boldsymbol{\mu})$.

The time-independent *linear Boltzmann transport equation* (LBTE) for a function $u : \mathcal{D} \to \mathbb{R}$ reads:

$$\begin{aligned} \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}, E) + (\alpha(\mathbf{x}, \boldsymbol{\mu}, E) + \beta(\mathbf{x}, \boldsymbol{\mu}, E))\, u(\mathbf{x}, \boldsymbol{\mu}, E) \\ = S[u](\mathbf{x}, \boldsymbol{\mu}, E) + f(\mathbf{x}, \boldsymbol{\mu}, E) \quad \text{in } \mathcal{D}, \\ u(\mathbf{x}, \boldsymbol{\mu}, E) = g(\mathbf{x}, \boldsymbol{\mu}, E) \quad \text{on } \Gamma_{in} \times \mathbb{Y}. \end{aligned} \tag{3.4}$$

Here, $\nabla_{\mathbf{x}}$ denotes the gradient operator acting on only the spatial components of functions defined on $\mathcal{D}$, and $\alpha \in L^\infty(\mathcal{D})$, $f \in L^2(\mathcal{D})$ and $g \in L^2_{|\boldsymbol{\mu}\cdot\mathbf{n}|}(\Gamma_{in} \times \mathbb{Y})$ are given data terms, where $L^2_{|\boldsymbol{\mu}\cdot\mathbf{n}|}(\Gamma_{in} \times \mathbb{Y})$ denotes the weighted Lebesgue space of all measurable functions f on $\Gamma_{in} \times \mathbb{Y}$ for which

$$||f||_{L^2_{|\boldsymbol{\mu}\cdot\mathbf{n}|}(\Gamma_{in}\times\mathbb{Y})} = \left(\int_{\mathbb{Y}}\int_{\mathbb{S}}\int_{\partial\Omega} |\boldsymbol{\mu}\cdot\mathbf{n}||f(\mathbf{x},\boldsymbol{\mu},E)|^2 \ \mathrm{d}s\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E\right)^{\frac{1}{2}} < \infty.$$

The term $S[u]$ denotes a scattering operator acting on u and is defined by

$$S[u](\mathbf{x},\boldsymbol{\mu},E) = \int_{\mathbb{S}}\int_{\mathbb{Y}} \theta(\mathbf{x},\boldsymbol{\mu}'\to\boldsymbol{\mu},E'\to E)u(\mathbf{x},\boldsymbol{\mu}',E') \ \mathrm{d}E'\,\mathrm{d}\boldsymbol{\mu}'. \tag{3.5}$$

The function $\theta(\mathbf{x},\boldsymbol{\mu}\to\boldsymbol{\mu}',E\to E')$ is a given scattering kernel. The data term $\beta : \mathcal{D} \to \mathbb{R}$ is related to θ by

$$\beta(\mathbf{x},\boldsymbol{\mu},E) = \int_{\mathbb{S}}\int_{\mathbb{Y}} \theta(\mathbf{x},\boldsymbol{\mu}\to\boldsymbol{\mu}',E\to E') \ \mathrm{d}E'\,\mathrm{d}\boldsymbol{\mu}'. \tag{3.6}$$

For the sake of notational simplicity, the dependence of the PDE/data terms on $\mathbf{x}$, $\boldsymbol{\mu}$ and E will be suppressed where such dependence is obvious.

The physical interpretation of (3.4) is given in Chapter 2.1. We remind the reader of the terminology used in the previous discussion of the LBTE: θ denotes the *differential scattering cross-section*, α denotes the *macroscopic absorption cross-section*, β denotes the *macroscopic scattering cross-section* and $\alpha+\beta$ denotes the *macroscopic total cross-section*.

The following (physically-reasonable) simplifying assumptions will be made about the data terms. The differential scattering cross-section (and the kinematics of the scattering event) will be assumed to depend on its angular arguments only through the cosine of the angle between them; that is, $\theta(\mathbf{x},\boldsymbol{\mu}\to\boldsymbol{\mu}',E\to E') = \theta(\mathbf{x},\boldsymbol{\mu}\cdot\boldsymbol{\mu}',E\to E')$. Consequently, this means that $\beta(\mathbf{x},\boldsymbol{\mu},E) = \beta(\mathbf{x},E)$. Moreover, we assume that particles can only lose energy during scattering events, so that $\theta(\mathbf{x},\boldsymbol{\mu}\cdot\boldsymbol{\mu}',E\to E') = 0$ for $E' > E$. We will also assume that the medium is angularly isotropic so that $\alpha(\mathbf{x},\boldsymbol{\mu},E) = \alpha(\mathbf{x},E)$, and that $\alpha(\mathbf{x},E) \geq 0$ for all $\mathbf{x} \in \Omega$ and $E > 0$.

For the forthcoming analysis in Chapter 3.3, we introduce two additional data-dependent terms $\gamma, \bar{\alpha} : \mathcal{D} \to \mathbb{R}$ defined by

$$\gamma(\mathbf{x},\boldsymbol{\mu},E) = \int_{\mathbb{S}}\int_{\mathbb{Y}} \theta(\mathbf{x},\boldsymbol{\mu}'\to\boldsymbol{\mu},E'\to E) \ \mathrm{d}E'\,\mathrm{d}\boldsymbol{\mu}', \tag{3.7}$$

$$\bar{\alpha}(\mathbf{x},\boldsymbol{\mu},E) = \alpha(\mathbf{x},\boldsymbol{\mu},E) + \frac{1}{2}\left(\beta(\mathbf{x},\boldsymbol{\mu},E) - \gamma(\mathbf{x},\boldsymbol{\mu},E)\right). \tag{3.8}$$

Under the previous assumptions on α and θ, it can be shown that γ and $\bar{\alpha}$ are independent of the angular variable $\boldsymbol{\mu}$. We remark that the definition of γ in (3.7) is in general *not* identical to that of β in (3.6) due to the reversal of the order of the energetic variables E and E', as well as the angular variables $\boldsymbol{\mu}$ and $\boldsymbol{\mu}'$, in the differential scattering cross-section θ.

In view of discretising (3.4) using a discontinuous Galerkin finite element method (DGFEM) approach, it is useful to introduce a number of sub-problems. To this end, we will further define a mono-energetic version of the LBTE for a function $u(\mathbf{x}, \boldsymbol{\mu})$ independent of energy:

$$\begin{aligned}
\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}) + (\alpha(\mathbf{x}) + \beta(\mathbf{x})) u(\mathbf{x}, \boldsymbol{\mu}) & \\
= S[u](\mathbf{x}, \boldsymbol{\mu}) + f(\mathbf{x}, \boldsymbol{\mu}) \quad & \text{in } \Omega \times \mathbb{S}, \\
u(\mathbf{x}, \boldsymbol{\mu}) = g(\mathbf{x}, \boldsymbol{\mu}) \quad & \text{on } \Gamma_{in}(\Omega \times \mathbb{S}), \\
S[u](\mathbf{x}, \boldsymbol{\mu}) = \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \cdot \boldsymbol{\mu}) u(\mathbf{x}, \boldsymbol{\mu}') \ \mathrm{d}\boldsymbol{\mu}', & \\
\beta(\mathbf{x}) = \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \cdot \boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu}', &
\end{aligned} \tag{3.9}$$

as well as a first-order hyperbolic transport equation for a function $u(\mathbf{x})$ independent of energy and angle:

$$\begin{aligned}
\boldsymbol{\mu} \cdot \nabla u(\mathbf{x}) + \alpha(\mathbf{x}) u(\mathbf{x}) = f(\mathbf{x}) \quad & \text{in } \Omega, \\
u(\mathbf{x}) = g(\mathbf{x}) \quad & \text{on } \Gamma_{in}(\Omega).
\end{aligned} \tag{3.10}$$

Here, the problem domains and inflow boundaries for the mono-energetic and transport problems are defined similarly to the case of the poly-energetic problem.

3.2 DGFEM Discretisation

We will now discretise the poly-energetic problem (3.4) using a discontinuous Galerkin finite element method (DGFEM) approach, and then present DGFEM discretisations of the mono-energetic problem (3.9) and the transport equation (3.10) as special cases. The spatial, angular and energetic domains will be discretised separately, and the finite element solutions sought by the discretisation of the poly-energetic problem will take the form of a linear combination of discontinuous piecewise-polynomial functions defined on each space-angle-energy mesh element. Before we present the DGFEM scheme for the poly-energetic problem, we first introduce the finite element meshes and spaces employed in each of the spatial, angular and energetic domains.

3.2.1 Spatial discretisation

Let $\mathcal{T}_\Omega$ denote a subdivision of Ω into disjoint open polytopic elements κ_Ω such that $\bar{\Omega} = \bigcup_{\kappa_\Omega \in \mathcal{T}_\Omega} \bar{\kappa}_\Omega$. To each spatial element $\kappa_\Omega \in \mathcal{T}_\Omega$ we denote its diameter by h_{κ_Ω} and assign a non-negative integer polynomial degree p_{κ_Ω}. We collect these polynomial degrees into a vector $\mathbf{p}_\Omega = (p_{\kappa_\Omega} : \kappa_\Omega \in \mathcal{T}_\Omega)$ and define a finite element space $\mathcal{V}_\Omega = \mathcal{V}_\Omega^{\mathbf{p}_\Omega}(\mathcal{T}_\Omega)$ of spatial discontinuous piecewise-polynomial functions by

$$\mathcal{V}_\Omega^{\mathbf{p}_\Omega}(\mathcal{T}_\Omega) = \left\{ v_\Omega \in L^2(\Omega) : v_\Omega|_{\kappa_\Omega} \in \mathbb{H}^{p_{\kappa_\Omega}}(\kappa_\Omega) \text{ for all } \kappa_\Omega \in \mathcal{T}_\Omega \right\}.$$

The space $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$ denotes the set of polynomial functions of maximal total degree p on κ; the space $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$ denotes the set of polynomial functions of maximal degree p in each independent variable on κ. We remark that the former choice of $\mathbb{H}^p(\kappa)$ is the polynomial space of least dimension that will ensure the validity of the forthcoming approximation properties of the scheme [35]. However, we will consider both choices of $\mathbb{H}^p(\kappa)$ in the context of quadrature-free assembly methods; see Chapter 4.

Let $\partial\kappa_\Omega$ denote the boundary of κ_Ω consisting of planar $(d-1)$-dimensional faces, and define a partition $\partial\kappa_\Omega = \partial_-\kappa_\Omega(\boldsymbol{\mu}) \cup \partial_+\kappa_\Omega(\boldsymbol{\mu})$ by

$$\partial_-\kappa_\Omega(\boldsymbol{\mu}) = \{\mathbf{x} \in \partial\kappa_\Omega : \mathbf{n}_{\kappa_\Omega}(\mathbf{x}) \cdot \boldsymbol{\mu} < 0\}, \tag{3.11}$$

$$\partial_+\kappa_\Omega(\boldsymbol{\mu}) = \{\mathbf{x} \in \partial\kappa_\Omega : \mathbf{n}_{\kappa_\Omega}(\mathbf{x}) \cdot \boldsymbol{\mu} \geq 0\}, \tag{3.12}$$

where $\mathbf{n}_{\kappa_\Omega}(\mathbf{x})$ denotes the outward unit normal to κ_Ω on $\partial\kappa_\Omega$.

Let $\mathcal{F}_\Omega$ denote the collection of element faces in $\mathcal{T}_\Omega$ and partition this set as $\mathcal{F}_\Omega = \mathcal{F}_\Omega^\partial \cup \mathcal{F}_\Omega^{int}$, where $\mathcal{F}_\Omega^\partial$ denotes the set of faces lying on the spatial boundary $\partial\Omega$ and $\mathcal{F}_\Omega^{int}$ denotes the set of interior faces. For a given wind direction $\boldsymbol{\mu} \in \mathbb{S}$, let $\mathcal{F}_\Omega^\partial = \mathcal{F}_\Omega^+(\boldsymbol{\mu}) \cup \mathcal{F}_\Omega^-(\boldsymbol{\mu})$, where

$$\mathcal{F}_\Omega^+(\boldsymbol{\mu}) = \{f \in \mathcal{F}_\Omega^\partial : f \subset \partial_+\Omega(\boldsymbol{\mu})\}, \tag{3.13}$$

$$\mathcal{F}_\Omega^-(\boldsymbol{\mu}) = \{f \in \mathcal{F}_\Omega^\partial : f \subset \partial_-\Omega(\boldsymbol{\mu})\}. \tag{3.14}$$

Finally, for a function v_{κ_Ω} defined on the boundary on an element $\partial\kappa_\Omega$, we denote by $v_{\kappa_\Omega}^+$ (resp. $v_{\kappa_\Omega}^-$) the interior (resp. exterior) trace of v on $\partial\kappa_\Omega$. Henceforth, it shall be clear on which element this notation applies, and so the subscript shall be dropped.

3.2.2 Angular discretisation

An obvious method for constructing a mesh on the (three-dimensional) unit sphere $\mathbb{S}$ is to map a mesh of the rectangle $(0, 2\pi) \times (0, \pi)$ onto $\mathbb{S}$ via the parametrisation

$$\boldsymbol{\mu} = (\sin\psi\cos\varphi, \sin\psi\sin\varphi, \cos\psi)\,,$$

where $(\varphi, \psi) \in (0, 2\pi) \times (0, \pi)$. However, such a mapping becomes singular at the poles of $\mathbb{S}$ (i.e. when $\psi = 0$ and $\psi = \pi$) and forces elements adjacent to the poles to have degenerate faces. We shall instead employ a cube-sphere mesh of the angular domain $\mathbb{S}$ on which we shall define a discrete function space of discontinuous piecewise-tensor-product polynomials. The construction of such a mesh requires that one already has a mesh of the surface of the d-dimensional unit cube and that the map from the unit cube to the unit sphere is smooth and invertible. In principle, the meshes employed on each face of the cube can be arbitrary; however, in view of constructing special polynomial basis functions that allow for a simplified assembly of the resulting DGFEM equations (see Chapter 3.4), we consider only mapped tensor-product meshes.

Let $\mathbb{F}^{d-1} \subset \mathbb{R}^d$ denote the surface of the unit cube defined by $\mathbb{F}^{d-1} = \{\boldsymbol{\mu} \in \mathbb{R}^d : ||\boldsymbol{\mu}||_\infty = 1\}$, where $||\boldsymbol{\mu}||_\infty = \max_{i=1}^d |\mu_i|$ denotes the vector ℓ^∞-norm on $\mathbb{R}^d$. For each $1 \leq f \leq 2d$, the face $\mathbb{F}_f \subset \mathbb{F}^{d-1}$ is a $(d-1)$-dimensional hypercube on which we will define a mesh $\mathcal{T}_{\mathbb{F}_f}$ consisting of disjoint open tensor-product elements $\kappa_{\mathbb{F}_f}$ such that $\mathbb{F}_f = \bigcup_{\kappa_{\mathbb{F}_f} \in \mathcal{T}_{\mathbb{F}_f}} \bar{\kappa}_{\mathbb{F}_f}$ and that there exists an affine mapping $\chi_{\kappa_{\mathbb{F}_f}} : \mathcal{K} \to \kappa_{\mathbb{F}_f}$ from the reference open hypercube $\mathcal{K} = (-1,1)^{d-1}$ to $\kappa_{\mathbb{F}_f}$ for every $\kappa_{\mathbb{F}_f} \in \mathcal{T}_{\mathbb{F}_f}$. We denote by $\mathcal{T}_{\mathbb{F}^{d-1}}$ the mesh obtained by taking the union of the meshes $\mathcal{T}_{\mathbb{F}_f}$ for $1 \leq f \leq 2d$; i.e.

$$\mathcal{T}_{\mathbb{F}^{d-1}} = \bigcup_{f=1}^{2d} \mathcal{T}_{\mathbb{F}_f}.$$

Let $T : \mathbb{F}^{d-1} \to \mathbb{S}$ be the smooth invertible mapping defined by $T(\boldsymbol{\xi}) = \boldsymbol{\xi}/|\boldsymbol{\xi}|_2$ for all $\boldsymbol{\xi} \in \mathbb{F}^{d-1}$, and denote by $T^{-1} : \mathbb{S} \to \mathbb{F}^{d-1}$ the corresponding inverse mapping $T^{-1}(\boldsymbol{\mu}) = \boldsymbol{\mu}/|\boldsymbol{\mu}|_\infty$ for all $\boldsymbol{\mu} \in \mathbb{S}$. Here, $|\cdot|_2$ (resp. $|\cdot|_\infty$) denotes the vector ℓ_2 (resp. vector ℓ_∞) norm on $\mathbb{R}^d$. We define the cube-sphere mesh $\mathcal{T}_\mathbb{S}$ to be the union of the images of the elements of $\mathcal{T}_{\mathbb{F}^{d-1}}$ under T:

$$\mathcal{T}_\mathbb{S} = \{T(\kappa_{\mathbb{F}^{d-1}}) : \kappa_{\mathbb{F}^{d-1}} \in \mathcal{T}_{\mathbb{F}^{d-1}}\}$$

The construction of the cube-sphere mesh is outlined in Figure 3.1.

To each angular element $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$ we denote its diameter by $h_{\kappa_\mathbb{S}}$ and assign a non-negative integer polynomial degree $q_{\kappa_\mathbb{S}}$. We collect these polynomial degrees into a vector $\mathbf{q}_\mathbb{S} = (q_{\kappa_\mathbb{S}} : \kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S})$ and define a finite element space $\mathcal{V}_\mathbb{S} = \mathcal{V}_\mathbb{S}^{\mathbf{q}_\mathbb{S}}(\mathcal{T}_\mathbb{S})$ of angular discontinuous piecewise-polynomial functions by

$$\mathcal{V}_\mathbb{S}^{\mathbf{q}_\mathbb{S}}(\mathcal{T}_\mathbb{S}) = \left\{v_\mathbb{S} \in L^2(\mathbb{S}) : v_\mathbb{S}|_{\kappa_\mathbb{S}} \circ T \circ \chi_{T^{-1}\kappa_{\mathcal{T}_\mathbb{S}}} \in \mathbb{H}^{q_{\kappa_\mathbb{S}}}(\mathcal{K}) \text{ for all } \kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}\right\}.$$

Here, we may select either $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$ or $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$. We shall henceforth only consider the latter choice as it simplifies the practical implementation outlined in Chapter 3.4. While the details are not relevant here, we will require the construction of a family of N_p-point quadrature scheme on $[-1,1]^{d-1}$ with $N_p = \dim \mathbb{H}^p(\kappa)$ and which exactly integrates any function in $\mathbb{H}^p(\kappa)$; such a construction is generally easier when $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$.

3.2.3 Energetic discretisation

We shall restrict the energy domain to a finite interval by selecting maximum and minimum energy cutoffs E_{max} and E_{min} respectively. These limits should be chosen such that the true solution of (3.4) is compactly supported in energy, and by abuse of notation, we shall refer to $\mathbb{Y}$ as the restricted energy domain (E_{min}, E_{max}). We subdivide the interval (E_{min}, E_{max}) into $N_\mathbb{Y} \geq 1$ *energy groups* $\kappa_g = (E_g, E_{g-1})$ such that

$$E_{max} = E_0 \geq E_1 \geq \cdots \geq E_{N_\mathbb{Y}-1} \geq E_{N_\mathbb{Y}} = E_{min}.$$

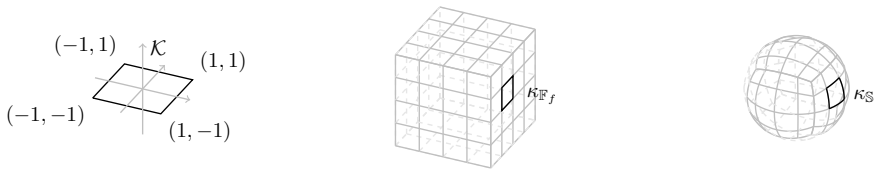

Figure 3.1: Left: reference element $\mathcal{K} = (0,1)^2$ embedded in $\mathbb{R}^d$. Middle: cube mesh $\mathbb{F}^{d-1}$ with an element $\kappa_{\mathbb{F}_f}$ highlighted. Right: cube-sphere mesh $\mathcal{T}_{\mathbb{S}}$ with an element $\kappa_{\mathbb{S}}$ highlighted. Left-to-middle: action of $\chi_{\kappa_{\mathbb{F}_f}}$ on $\mathcal{K}$. Middle-to-right: action of T on $\kappa_{\mathbb{F}_f}$.

We define the energetic mesh $\mathcal{T}_{\mathbb{Y}} = \{\kappa_g\}_{g=1}^{N_{\mathbb{Y}}}$. To each energetic element $\kappa_{\mathbb{Y}} \in \mathcal{T}_{\mathbb{Y}}$ we denote its diameter by $h_{\kappa_{\mathbb{Y}}}$ and assign a non-negative integer polynomial degree $r_{\kappa_{\mathbb{Y}}}$. We collect these polynomial degrees into a vector $\mathbf{r}_{\mathbb{Y}} = (r_{\kappa_{\mathbb{Y}}} : \kappa_{\mathbb{Y}} \in \mathcal{T}_{\mathbb{Y}})$ and define a finite element space $\mathcal{V}_{\mathbb{Y}} = \mathcal{V}_{\mathbb{Y}}^{\mathbf{r}_{\mathbb{Y}}}(\mathcal{T}_{\mathbb{Y}})$ of energetic discontinuous piecewise-polynomial functions by

$$\mathcal{V}_{\mathbb{Y}}^{\mathbf{r}_{\mathbb{Y}}}(\mathcal{T}_{\mathbb{Y}}) = \{v_{\mathbb{Y}} \in L^2(E_{min}, E_{max}) : v_{\mathbb{Y}}|_{\kappa_{\mathbb{Y}}} \in \mathbb{P}^{r_{\kappa_{\mathbb{Y}}}}(\kappa_{\mathbb{Y}}) \text{ for all } \kappa_{\mathbb{Y}} \in \mathcal{T}_{\mathbb{Y}}\}.$$

3.2.4 DGFEM Poly-Energetic Scheme

Seeking to derive a DGFEM scheme for the poly-energetic LBTE (3.4), we must first specify a computational mesh on the space-angle-energy domain $\mathcal{D}$ and a function space over the resulting mesh. We consider a tensorised space-angle-element mesh $\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}$ defined by

$$\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}} = \mathcal{T}_{\Omega} \times \mathcal{T}_{\mathbb{S}} \times \mathcal{T}_{\mathbb{Y}} = \{\kappa_{\Omega} \times \kappa_{\mathbb{S}} \times \kappa_{\mathbb{Y}} : \kappa_{\Omega} \in \mathcal{T}_{\Omega}, \kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}, \kappa_{\mathbb{Y}} \in \mathcal{T}_{\mathbb{Y}}\}. \tag{3.15}$$

Furthermore, we will define the following function spaces:

$$\mathcal{G} = \left\{v \in L^2(\mathcal{D}) : \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} v + (\alpha + \beta)v \in L^2(\mathcal{D})\right\}, \tag{3.16}$$

$$\mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}) = \left\{v \in L^2(\mathcal{D}) : (\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} v + (\alpha + \beta)v)\,|_{\kappa} \in L^2(\kappa) \text{ for all } \kappa \in \mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}\right\}. \tag{3.17}$$

The space-angle-energy mesh $\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}$ is equipped with a finite element space of discontinuous piecewise-polynomial space-angle-energy functions $\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ constructed via

$$\begin{aligned}\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} &= \mathcal{V}_{\Omega} \otimes \mathcal{V}_{\mathbb{S}} \otimes \mathcal{V}_{\mathbb{Y}} \\ &= \operatorname{span}\left\{v_{\Omega} v_{\mathbb{S}} v_{\mathbb{Y}} : v_{\Omega} \in \mathcal{V}_{\Omega}, v_{\mathbb{S}} \in \mathcal{V}_{\mathbb{S}}, v_{\mathbb{Y}} \in \mathcal{V}_{\mathbb{Y}}\right\}.\end{aligned}$$

Notice that $\mathcal{G} \subset \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}})$ and $\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \subset \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}})$ but $\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \not\subset \mathcal{G}$.

Multiplying the first equation in (3.4) by a test function $v \in \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}})$ and integrating over $\mathcal{D}$, we get

$$\int_{\mathbb{Y}}\int_{\mathbb{S}}\int_{\Omega} \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} uv + (\alpha + \beta)uv \;\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E = \int_{\mathbb{Y}}\int_{\mathbb{S}}\int_{\Omega} (S[u]v + fv) \;\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E,$$

where we have momentarily suppressed the dependence of all functions on $\mathbf{x}$, $\boldsymbol{\mu}$ and E for simplicity. Isolating the integral of the streaming term $\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} uv$ over the spatial domain for given $\boldsymbol{\mu}$ and E, we have

$$\begin{aligned}
\int_{\Omega} \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}, E) v(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}\mathbf{x} &= \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \int_{\kappa_\Omega} \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}, E) v(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}\mathbf{x} \\
&= \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Big(\int_{\partial \kappa_\Omega} u^+(\mathbf{x}, \boldsymbol{\mu}, E) v^+(\mathbf{x}, \boldsymbol{\mu}, E) \boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega} \ \mathrm{d}s \\
&\qquad - \int_{\kappa_\Omega} u(\mathbf{x}, \boldsymbol{\mu}, E) \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} v(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}\mathbf{x} \Big) \\
&= \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Big(- \int_{\kappa_\Omega} u(\mathbf{x}, \boldsymbol{\mu}, E) \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} v(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}\mathbf{x} \\
&\qquad + \int_{\partial \kappa_\Omega} H_{\boldsymbol{\mu}}(u^+, u^-, \mathbf{n}_{\kappa_\Omega}) v^+(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}s \Big).
\end{aligned}$$

Here we have replaced the term $u^+ \boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}$ with a numerical flux with a numerical flux $H_{\boldsymbol{\mu}}(u^+, u^-, \mathbf{n}_{\kappa_\Omega})$ satisfying the following assumptions:

- $H_{\boldsymbol{\mu}}(\cdot, \cdot, \mathbf{n}_{\kappa_\Omega})$ is consistent - we have that

  $$H_{\boldsymbol{\mu}}(w^+, w^-, \mathbf{n}_{\kappa_\Omega})|_{\partial \kappa_\Omega} = \boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega} w|_{\partial \kappa_\Omega}$$

  whenever w is a smooth function satisfying the inflow boundary conditions;

- $H_{\boldsymbol{\mu}}(\cdot, \cdot, \mathbf{n}_{\kappa_\Omega})$ is conservative - we have that

  $$H_{\boldsymbol{\mu}}(w^+, w^-, \mathbf{n}_{\kappa_\Omega}) = -H_{\boldsymbol{\mu}}(w^-, w^+, -\mathbf{n}_{\kappa_\Omega})$$

  and so $H_{\boldsymbol{\mu}}(\cdot, \cdot, \cdot)$ is single-valued on $\Gamma_\Omega(\mathcal{T}_\Omega) = \cup_{\kappa_\Omega \in \mathcal{T}_\Omega} \partial \kappa_\Omega$, the set of spatial element boundaries in $\mathcal{T}_\Omega$.

We shall select an upwind numerical flux for our scheme; see [98] for a number of commonly-employed numerical fluxes. The upwind flux is both consistent and conservative:

$$H_{\boldsymbol{\mu}}(w^+, w^-, \mathbf{n}_{\kappa_\Omega})|_{\kappa_\Omega} = \begin{cases} \boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega} \ w^+(\mathbf{x}, \boldsymbol{\mu}, E) & \mathbf{x} \in \partial_+ \kappa_\Omega(\boldsymbol{\mu}), \\ \boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega} \ w^-(\mathbf{x}, \boldsymbol{\mu}, E) & \mathbf{x} \in \partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial_- \Omega(\boldsymbol{\mu}), \\ \boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega} \ g(\mathbf{x}, \boldsymbol{\mu}, E) & \mathbf{x} \in \partial_- \kappa_\Omega(\boldsymbol{\mu}) \cap \partial_- \Omega(\boldsymbol{\mu}), \end{cases}$$

where $\partial_+ \kappa_\Omega(\boldsymbol{\mu})$ (resp. $\partial_- \kappa_\Omega(\boldsymbol{\mu})$) denotes the outflow (resp. inflow) boundary of κ_Ω and defined in (3.12) (resp. (3.11)). With this choice of numerical flux, we have the following expression for the boundary integral over $\partial \kappa_\Omega$:

$$\begin{aligned}
&\int_{\partial \kappa_\Omega} H_{\boldsymbol{\mu}}(u^+, u^-, \mathbf{n}_{\kappa_\Omega}) v^+(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}s \\
&\qquad = \int_{\partial_+ \kappa(\boldsymbol{\mu})} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| u^+(\mathbf{x}, \boldsymbol{\mu}, E) v^+(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}s \\
&\qquad\quad - \int_{\partial_- \kappa(\boldsymbol{\mu}) \setminus \partial \Omega} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| u^-(\mathbf{x}, \boldsymbol{\mu}, E) v^+(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}s \\
&\qquad\quad - \int_{\partial_- \kappa(\boldsymbol{\mu}) \cap \partial \Omega} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| g(\mathbf{x}, \boldsymbol{\mu}, E) v^+(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}s.
\end{aligned}$$

By summing over all $\kappa = \kappa_\Omega \times \kappa_\mathbb{S} \times \kappa_\mathbb{Y} \in \mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}$, the variational formulation reads as follows: find $u \in \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}})$ such that

$$T(u,v) = S(u,v) + \ell(v) \tag{3.18}$$

for all $v \in \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}})$, where the bilinear forms $T, S : \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}) \times \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}) \to \mathbb{R}$ and the linear functional $\ell : \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}) \to \mathbb{R}$ are defined for all $w, v \in \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}})$ by

$$\begin{aligned}
T(w,v) = \int_\mathbb{Y} \int_\mathbb{S} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Bigg(& \int_{\kappa_\Omega} (-w\boldsymbol{\mu} \cdot \nabla_\mathbf{x} v + (\alpha+\beta)wv) \ \mathrm{d}\mathbf{x} \\
& + \int_{\partial_+ \kappa_\Omega(\boldsymbol{\mu})} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| w^+ v^+ \ \mathrm{d}s \\
& - \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| w^- v^+ \ \mathrm{d}s \Bigg) \ \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E, \qquad (3.19) \\
S(w,v) = \int_\mathbb{Y} \int_\mathbb{S} \int_\Omega S[w] v \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E \qquad & (3.20) \\
= \int_\mathbb{Y} \int_\mathbb{S} \int_\Omega \int_\mathbb{Y} \int_\mathbb{S} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E' \to E) w(\mathbf{x}, \boldsymbol{\mu}', E') v(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E' \, \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E, & \\
\ell(v) = \int_\mathbb{Y} \int_\mathbb{S} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left(\int_{\kappa_\Omega} f v \ \mathrm{d}\mathbf{x} + \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \cap \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| g v^+ \ \mathrm{d}s \right) \ \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E. \qquad & (3.21)
\end{aligned}$$

Here, we use the condensed notation $\int_\Omega = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \int_{\kappa_\Omega}$, $\int_\mathbb{S} = \sum_{\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}} \int_{\kappa_\mathbb{S}}$ and $\int_\mathbb{Y} = \sum_{\kappa_\mathbb{Y} \in \mathcal{T}_\mathbb{Y}} \int_{\kappa_\mathbb{Y}}$ for simplicity of presentation.

By replacing $u, v \in \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}})$ with discrete functions $u_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$, the poly-energetic DGFEM scheme thus reads as follows: find $u_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ such that

$$T(u_h, v_h) = S(u_h, v_h) + \ell(v_h) \tag{3.22}$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$. We note that the scheme is consistent whenever the numerical flux $H_{\boldsymbol{\mu}}(\cdot,\cdot,\cdot)$ is consistent. If the analytical solution u to (3.4) satisfies $u \in \mathcal{G}$, then by taking $v = v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ in (3.18) we have

$$T(u, v_h) = S(u, v_h) + \ell(v_h)$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$.

Finally, we note that the data terms β and γ retain their original definitions in (3.6) and (3.7) respectively, so that the definition of $\bar{\alpha}$ in (3.8) remains unchanged:

$$\bar{\alpha}(\mathbf{x}, \boldsymbol{\mu}, E) = \alpha(\mathbf{x}, \boldsymbol{\mu}, E) + \frac{1}{2}(\beta(\mathbf{x}, \boldsymbol{\mu}, E) - \gamma(\mathbf{x}, \boldsymbol{\mu}, E)).$$

By our assumptions that the medium is angularly isotropic and that the differential scattering cross-section depends on $\boldsymbol{\mu}$ and $\boldsymbol{\mu}'$ only through the combination $\boldsymbol{\mu} \cdot \boldsymbol{\mu}'$, we may eliminate the dependence on $\boldsymbol{\mu}$ in α, β and γ, so that $\bar{\alpha}$ is a function of space and energy.

Example: Compton scattering

Consider the case where the differential scattering cross-section

$$\theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E \to E') = \rho(\mathbf{x})\theta_{KN}(\cos\varphi, E, E'),$$

where $\cos\varphi = \boldsymbol{\mu} \cdot \boldsymbol{\mu}'$, $\rho(\mathbf{x})$ denotes the (local) electron density and θ_{KN} denotes the Klein-Nishina differential scattering cross-section per electron [64], which we repeat from Chapter 2.1.1:

$$\theta_{KN}(\cos\varphi, E, E') = \frac{r_e^2}{2}\left(\frac{E'}{E}\right)^2\left(\frac{E}{E'} + \frac{E'}{E} - \sin^2\varphi\right), \tag{3.23}$$

where $r_e \approx 2.818 \times 10^{-15}$m denotes the classical electron radius. Henceforth, it shall be convenient assume that E and E' are specified in units of electron rest energy (i.e. multiples of $m_e c^2 \approx 511$keV).

We additionally have the following (equivalent) kinematic constraints on $\frac{E'}{E}$, the fraction of energy retained by a photon with initial energy E undergoing a Compton scattering event and recoiling with energy E':

$$\begin{aligned}\frac{E'}{E} &= \frac{1}{1 + E(1 - \cos\varphi)} =: P(\cos\varphi, E),\\ \frac{E'}{E} &= 1 - E'(1 - \cos\varphi) =: Q(\cos\varphi, E').\end{aligned}$$

These constraints may be implemented as Dirac delta functions multiplying $\theta_{KN}(\cos\varphi, E, E')$ and allow one of the angular or energetic integrals in (3.6) and (3.7) to be eliminated.

The associated macroscopic scattering cross-section $\beta(\mathbf{x}, E)$ associated with the differential scattering cross-section $\theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E \to E')$ is a classical result [64], and is given as follows:

$$\begin{aligned}\beta(\mathbf{x}, E) &= \int_{\mathbb{Y}}\int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E \to E')\ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}E'\\ &= 2\pi r_e^2 \rho(\mathbf{x})\left[\frac{1+E}{E^2}\left(\frac{2(1+E)}{1+2E} - \frac{\log(1+2E)}{E}\right)\right.\\ &\qquad\left. + \frac{\log(1+2E)}{2E}\frac{1+3E}{(1+2E)^2}\right].\end{aligned} \tag{3.24}$$

The evaluation of the associated coefficient $\gamma(\mathbf{x}, E)$ defined in (3.7) requires more care. We repeat this definition using slightly different notation:

$$\gamma(\mathbf{x}, E') = \int_{\mathbb{Y}}\int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E \to E')\ \mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E. \tag{3.25}$$

Since $0 \le E' \le E$, the fraction $\frac{E'}{E} = Q(\cos\varphi, E')$ must lie in the interval $[0, 1]$, and so we must have

$$0 \le 1 - \cos\varphi \le \min\left\{2, \frac{1}{E'}\right\}. \tag{3.26}$$

This is because we must have $0 \le 1 - \cos\varphi \le 2$ since $\cos\varphi \in [-1, 1]$ for all possible deflection angles φ, and additionally we must have $0 \le 1 - \cos\varphi \le \frac{1}{E'}$ by the constraint on $Q(E', E)$.

The physical understanding of (3.26) is as follows: for a photon leaving a Compton scattering event with energy $E' \leq \frac{1}{2}$, we can deduce that it may have scattered through any deflection angle φ. However, if the recoiling photon has energy $E' > \frac{1}{2}$, it can only have scattered through a small enough angle. Mathematically, the deflection angle φ must satisfy

$$0 \leq \varphi \leq \bar{\varphi}(E') := \begin{cases} 2\pi, & E' \leq \frac{1}{2}, \\ \arccos\left(1 - \frac{1}{E'}\right), & E' > \frac{1}{2}. \end{cases}$$

We are now ready to evaluate the integral in (3.25). By abuse of notation, we shall understand integrals over $\mathbb{S}$ as the integral over all allowable scattering angles as permitted by the scattering kinematics; i.e. those angle satisfying (3.26). We have

$$\begin{aligned} \gamma(\mathbf{x}, E') &= \frac{r_e^2 \rho(\mathbf{x})}{2} \int_{\mathbb{Y}} \int_{\mathbb{S}} Q(\cos\varphi, E')^2 \left(Q(\cos\varphi, E') + \frac{1}{Q(\cos\varphi, E')} - \sin^2\varphi \right) \cdot \\ &\qquad \delta\left(E - \frac{E'}{Q(\cos\varphi, E')} \right) \, \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E \\ &= \pi r_e^2 \rho(\mathbf{x}) \int_0^{\bar{\varphi}(E')} (1 - E'(1-\cos\varphi))^2 \cdot \\ &\qquad \left[\cos^2\varphi + \frac{1}{1 - E'(1-\cos\varphi)} - E'(1-\cos\varphi) \right] \sin\varphi \, \mathrm{d}\varphi. \end{aligned}$$

Making the substitution $y = 1 - \cos\varphi$, this integral can be evaluated as

$$\begin{aligned} \gamma(\mathbf{x}, E') &= \pi r_e^2 \rho(\mathbf{x}) \int_0^{\min\left\{2, \frac{1}{E'}\right\}} (1 - E'y)^2 \left[(1-y)^2 + \frac{1}{1 - E'y} - E'y \right] \, \mathrm{d}y \\ &= \pi r_e^2 \rho(\mathbf{x}) \begin{cases} \frac{8}{3} - \frac{16E'}{3} + \frac{32(E')^2}{5} - 4(E')^3, & E' \leq \frac{1}{2}, \\ \frac{3}{4E'} - \frac{1}{6(E')^2} + \frac{1}{30(E')^3}, & E' > \frac{1}{2}. \end{cases} \end{aligned} \tag{3.27}$$

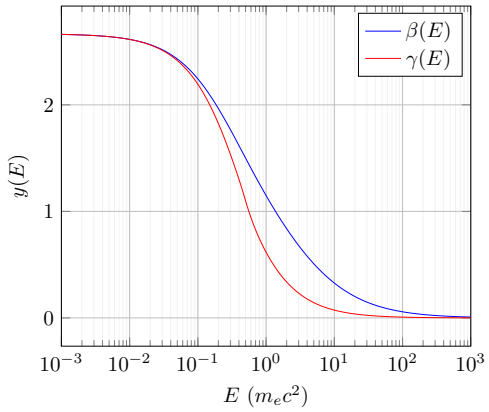

Figure 3.2: Plot of $\beta(E) = \beta(\mathbf{x}, E)/\pi r_e^2 \rho(\mathbf{x})$ and $\gamma(E) = \gamma(\mathbf{x}, E)/\pi r_e^2 \rho(\mathbf{x})$ as functions of energy over the energy range $(10^{-3}, 10^3)$ in units of electron rest energy.

Figure 3.2 shows the energetic dependence of $\beta(\mathbf{x}, E)$ and $\gamma(\mathbf{x}, E)$ (defined in (3.6) and (3.7) respectively) in the case that both functions are derived from the Klein-Nishina differential scattering cross-section (3.23). Both functions tend to zero in the high-energy limit, and to $\frac{8}{3}\pi r_e^2$ in the low-energy limit. Figure 3.3 shows that, over the

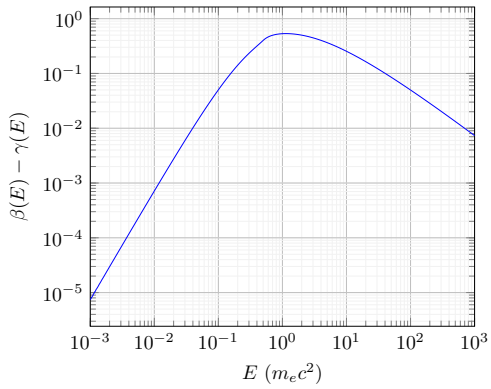

Figure 3.3: Plot of $\beta(E) - \gamma(E) = (\beta(\mathbf{x}, E) - \gamma(\mathbf{x}, E))/\pi r_e^2 \rho(\mathbf{x})$ as a function of energy over the energy range $(10^{-3}, 10^3)$ in units of electron rest energy.

range of energies presented, we have that $\beta(\mathbf{x}, E) - \gamma(\mathbf{x}, E) \geq 0$ - while not proven here, it is expected that this result also holds for any $0 < E < \infty$.

3.2.5 DGFEM Mono-Energetic Scheme

To derive a DGFEM scheme for the mono-energetic LBTE (3.9), it suffices to consider the scheme (3.22) in the case where the energetic component of the test and trial functions and the data terms are suppressed. This removes the need to derive the mono-energetic scheme from scratch. We consider a tensorised space-angle mesh $\mathcal{T}_{\Omega,\mathbb{S}}$ defined by

$$\mathcal{T}_{\Omega,\mathbb{S}} = \mathcal{T}_\Omega \times \mathcal{T}_\mathbb{S} = \{\kappa_\Omega \times \kappa_\mathbb{S} : \kappa_\Omega \in \mathcal{T}_\Omega, \kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}\}. \tag{3.28}$$

The space-angle mesh $\mathcal{T}_{\Omega,\mathbb{S}}$ is equipped with a finite element space of discontinuous piecewise-polynomial space-angle functions $\mathcal{V}_{\Omega,\mathbb{S}}$ constructed via

$$\mathcal{V}_{\Omega,\mathbb{S}} = \mathcal{V}_\Omega \otimes \mathcal{V}_\mathbb{S}.$$

Equivalently, we may define the space-angle finite element space as the subspace of $\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ consisting of space-angle-energy functions which are constant in the energetic argument:

$$\mathcal{V}_{\Omega,\mathbb{S}} = \{v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} : v_h(\cdot,\cdot,E) = v_h(\cdot,\cdot,E') \text{ for all } E, E' \in \mathbb{Y}\}.$$

Finally, we shall define energy-independent data terms[1] by

$$\begin{aligned}
\theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}', E' \to E) &= \frac{1}{|\mathbb{Y}|}\theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}'),\\
\alpha(\mathbf{x}, \boldsymbol{\mu}, E) &= \alpha(\mathbf{x}, \boldsymbol{\mu}),\\
f(\mathbf{x}, \boldsymbol{\mu}, E) &= f(\mathbf{x}, \boldsymbol{\mu}),\\
g(\mathbf{x}, \boldsymbol{\mu}, E) &= g(\mathbf{x}, \boldsymbol{\mu}).
\end{aligned}$$

[1] Note that the definition of θ only makes sense when the energy domain $\mathbb{Y}$ is a finite interval; however, we made this assumption upon discretising the energy domain.

Upon replacing the data terms in (3.20) and replacing the finite element space $\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ with $\mathcal{V}_{\Omega,\mathbb{S}}$, the mono-energetic DGFEM scheme thus reads as follows: find $u_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ such that

$$T(u_h, v_h) = S(u_h, v_h) + \ell(v_h) \tag{3.29}$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$, where

$$\begin{aligned} T(w_h, v_h) = \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Bigg(& \int_{\kappa_\Omega} (-w_h \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} v_h + (\alpha + \beta) w_h v_h) \ \mathrm{d}\mathbf{x} \\ & + \int_{\partial_+ \kappa_\Omega(\boldsymbol{\mu})} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| w_h^+ v_h^+ \ \mathrm{d}s \\ & - \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| w_h^- v_h^+ \ \mathrm{d}s \Bigg) \mathrm{d}\boldsymbol{\mu}, \end{aligned} \tag{3.30}$$

$$S(w_h, v_h) = \int_{\mathbb{S}} \int_{\Omega} S[w_h] v_h \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \tag{3.31}$$

$$= \int_{\mathbb{S}} \int_{\Omega} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') w_h(\mathbf{x}, \boldsymbol{\mu}') v_h(\mathbf{x}, \boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu},$$

$$\ell(v_h) = \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left(\int_{\kappa_\Omega} f v \ \mathrm{d}\mathbf{x} + \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \cap \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| g v^+ \ \mathrm{d}s \right) \mathrm{d}\boldsymbol{\mu}. \tag{3.32}$$

Contrary to the poly-energetic case, the data terms β and γ in the mono-energetic case are identical:

$$\begin{aligned} \beta(\mathbf{x}, \boldsymbol{\mu}) &= \int_{\mathbb{Y}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E \to E') \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E' \\ &= \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') \ \mathrm{d}\boldsymbol{\mu}', \\ \gamma(\mathbf{x}, \boldsymbol{\mu}) &= \int_{\mathbb{Y}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E' \to E) \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E' \\ &= \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') \ \mathrm{d}\boldsymbol{\mu}'. \end{aligned}$$

As a consequence, the definition of $\bar{\alpha}$ in (3.8) reduces to the macroscopic absorption cross-section

$$\bar{\alpha}(\mathbf{x}, \boldsymbol{\mu}) = \alpha(\mathbf{x}, \boldsymbol{\mu}).$$

By our assumptions that the medium is angularly isotropic and that the differential scattering cross-section depends on $\boldsymbol{\mu}$ and $\boldsymbol{\mu}'$ only through the combination $\boldsymbol{\mu} \cdot \boldsymbol{\mu}'$, we may eliminate the dependence on $\boldsymbol{\mu}$ in α, β and γ, so that $\bar{\alpha}$ is a function of space only.

As before, the scheme is consistent whenever the numerical flux $H_{\boldsymbol{\mu}}(\cdot, \cdot, \cdot)$ is consistent; that is, if the analytical solution u to (3.9) is sufficently smooth, we have

$$T(u, v_h) = S(u, v_h) + \ell(v_h)$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$.

3.2.6 DGFEM Transport Scheme

To derive a DGFEM scheme for the transport equation (3.10), it suffices to consider the scheme (3.29) in the case where the angular component of the test and trial functions

is suppressed. This removes the need to derive the transport scheme from scratch. We consider only the spatial mesh $\mathcal{T}_\Omega$ equipped with a finite element space of discontinuous piecewise-polynomial space functions $\mathcal{V}_\Omega$.

We shall set the differential scattering cross-section $\theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') = 0$ - this has the consequence that $\beta(\mathbf{x}, \boldsymbol{\mu}) = \gamma(\mathbf{x}, \boldsymbol{\mu}) = 0$. We shall define angle-independent data terms by

$$\begin{aligned} \alpha(\mathbf{x}, \boldsymbol{\mu}) &= \alpha(\mathbf{x}), \\ f(\mathbf{x}, \boldsymbol{\mu}) &= f(\mathbf{x}), \\ g(\mathbf{x}, \boldsymbol{\mu}) &= g(\mathbf{x}). \end{aligned}$$

Upon replacing the data terms in (3.29), replacing the finite element space $\mathcal{V}_{\Omega,\mathbb{S}}$ with $\mathcal{V}_\Omega$ and eliminating the outer integral over $\mathbb{S}$, the transport DGFEM scheme corresponding to a fixed wind direction $\boldsymbol{\mu} \in \mathbb{S}$ reads as follows: find $u_h \in \mathcal{V}_\Omega$ such that

$$T(u_h, v_h) = \ell(v_h) \tag{3.33}$$

for all $v_h \in \mathcal{V}_\Omega$, where

$$\begin{aligned} T(w_h, v_h) = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Bigg(& \int_{\kappa_\Omega} (-w_h \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} v_h + \alpha w_h v_h) \ \mathrm{d}\mathbf{x} \\ & + \int_{\partial_+ \kappa_\Omega(\boldsymbol{\mu})} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| w_h^+ v_h^+ \ \mathrm{d}s \\ & - \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| w_h^- v_h^+ \ \mathrm{d}s \Bigg), \end{aligned} \tag{3.34}$$

$$\ell(v_h) = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left(\int_{\kappa_\Omega} f v \ \mathrm{d}\mathbf{x} + \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \cap \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| g v^+ \ \mathrm{d}s \right). \tag{3.35}$$

The definition of $\bar{\alpha}$ in (3.8) reduces to the macroscopic absorption cross-section:

$$\bar{\alpha}(\mathbf{x}) = \alpha(\mathbf{x}).$$

As before, the scheme is consistent whenever the numerical flux $H_{\boldsymbol{\mu}}(\cdot, \cdot, \cdot)$ is consistent; that is, if the analytical solution u to (3.10) is sufficently smooth, we have

$$T(u, v_h) = S(u, v_h) + \ell(v_h)$$

for all $v_h \in \mathcal{V}_\Omega$.

3.3 Stability and Convergence Analysis

We shall now analyse the stability and convergence of the DGFEM scheme (3.22) applied to the poly-energetic LBTE (3.4) - the analysis of the DGFEM schemes (3.29) and (3.33) applied to the mono-energetic LBTE (3.9) and the first-order transport equation (3.10) respectively follow by removing the dependence on the energetic and angular variables.

We assume that there exists a constant $\alpha_0 > 0$ such that

$$\bar{\alpha}(\mathbf{x}, \boldsymbol{\mu}, E) \geq \alpha_0 \tag{3.36}$$

almost everywhere in $\mathcal{D}$, where we recall the definition of $\bar{\alpha}$ in (3.8). In the case of the first-order transport problem (3.33), this assumption is equivalent to the standard positivity assumption for Friedrichs' systems [38]. In the case of the mono-energetic problem (3.29), as well as the poly-energetic problem (3.22), the assumption (3.36) is understood as a generalisation to problems incorporating an additional integral source term of the form

$$\int_{\mathbb{Y}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}, E' \to E) u(\mathbf{x}, \boldsymbol{\mu}', E') - \theta(\mathbf{x}, \boldsymbol{\mu} \to \boldsymbol{\mu}', E \to E') u(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E'.$$

In order to make the forthcoming analysis rigorous for polytopic meshes $\mathcal{T}_\Omega$, we shall introduce an extra mesh size parameter $h^\perp_{\kappa_\Omega}$ for each $\kappa_\Omega \in \mathcal{T}_\Omega$ as in [28, 53]. Given $\kappa_\Omega \in \mathcal{T}_\Omega$, define $\mathcal{F}_\flat^{\kappa_\Omega}$ as the set of all possible d-dimensional simplices contained in κ_Ω and with at least one face $F \subset \partial\kappa_\Omega$ in common with κ_Ω. We define the extra mesh size parameter $h^\perp_{\kappa_\Omega}$ by

$$h^\perp_{\kappa_\Omega} = \min_{F \subset \partial\kappa_\Omega} \frac{d}{|F|} \sup\{|\kappa_\flat^F| : \kappa_\flat^F \in \mathcal{F}_\flat^{\kappa_\Omega} \text{ with } F \subset \partial\kappa_\flat^F\}. \tag{3.37}$$

For all $\kappa_\Omega \in \mathcal{T}_\Omega$, we have $h^\perp_{\kappa_\Omega} \leq h_{\kappa_\Omega}$.

For the stability analysis, we shall introduce the DGFEM-*energy norm* $|||\cdot|||_{DG} : \mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}) \to \mathbb{R}$ as in [53]:

$$\begin{aligned} |||v|||^2_{DG} = {} & ||\bar{\alpha}^{\frac{1}{2}} v||^2_{L^2(\mathcal{D})} \\ & + \frac{1}{2} \int_{\mathbb{Y}} \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left(|v^+ - v^-|^2_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} + |v^+|^2_{\partial\kappa_\Omega \cap \partial\Omega} \right) \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E, \end{aligned} \tag{3.38}$$

where $|\cdot|_\omega$ for $\omega \subset \partial\kappa_\Omega$ denotes the seminorm associated with the semi-inner product $(v, w)_\omega = \int_\omega |\boldsymbol{\mu} \cdot \mathbf{n}_{\kappa_\Omega}| vw \ \mathrm{d}s$ and $\bar{\alpha}$ denotes the function in (3.8). For the convergence analysis, we shall introduce the *streamline norm*

$$|||v|||^2_s = |||v|||^2_{DG} + \int_{\mathbb{Y}} \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \tau_{\kappa_\Omega} ||\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} v||^2_{L^2(\kappa_\Omega)} \ \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E, \tag{3.39}$$

where τ_{κ_Ω} is defined for each $\kappa_\Omega \in \mathcal{T}_\Omega$ by

$$\tau_{\kappa_\Omega} = \frac{h^\perp_{\kappa_\Omega}}{p^2_{\kappa_\Omega}}.$$

and $h^\perp_{\kappa_\Omega}$ is defined in (3.37).

Before we analyse the stability and convergence of the poly-energetic scheme, it is useful to introduce the following lemma regarding the scattering operator.

Lemma 3.3.1. *For all $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$, we have*

$$\int_{\mathbb{Y}} \int_{\mathbb{S}} \int_\Omega S[w_h] v_h \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E \leq ||\beta^{\frac{1}{2}} w_h||_{L^2(\mathcal{D})} ||\gamma^{\frac{1}{2}} v_h||_{L^2(\mathcal{D})}. \tag{3.40}$$

Proof. By employing the Cauchy-Schwarz inequality directly, we have

$$\begin{aligned}
&\int_{\mathbb{Y}}\int_{\mathbb{S}}\int_{\Omega} S[w_h]v_h \ \mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E \\
&\quad= \int_{\mathbb{Y}}\int_{\mathbb{S}}\int_{\Omega}\int_{\mathbb{Y}}\int_{\mathbb{S}} \theta(\mathbf{x},\boldsymbol{\mu}\cdot\boldsymbol{\mu}',E'\to E) w_h(\mathbf{x},\boldsymbol{\mu}',E') v_h(\mathbf{x},\boldsymbol{\mu},E) \ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}E'\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E \\
&\quad\leq \left(\int_{\mathbb{Y}}\int_{\mathbb{S}}\int_{\Omega}\int_{\mathbb{Y}}\int_{\mathbb{S}} \theta(\mathbf{x},\boldsymbol{\mu}\cdot\boldsymbol{\mu}',E'\to E) w_h(\mathbf{x},\boldsymbol{\mu}',E')^2 \ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}E'\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E\right)^{\frac{1}{2}} \\
&\qquad\cdot \left(\int_{\mathbb{Y}}\int_{\mathbb{S}}\int_{\Omega}\int_{\mathbb{Y}}\int_{\mathbb{S}} \theta(\mathbf{x},\boldsymbol{\mu}\cdot\boldsymbol{\mu}',E'\to E) v_h(\mathbf{x},\boldsymbol{\mu},E)^2 \ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}E'\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E\right)^{\frac{1}{2}} \\
&\quad= ||\beta^{\frac{1}{2}} w_h||_{L^2(\mathcal{D})} \ ||\gamma^{\frac{1}{2}} v_h||_{L^2(\mathcal{D})}.
\end{aligned}$$

□

We shall start by proving coercivity of the bilinear form $A : \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \times \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \to \mathbb{R}$ defined for all $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ by

$$A(w_h, v_h) := T(w_h, v_h) - S(w_h, v_h). \tag{3.41}$$

Theorem 3.3.2 (Coercivity)**.** *The bilinear form $A : \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \times \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \to \mathbb{R}$ used in the DGFEM scheme for problem (3.4) is coercive with respect to the DGFEM-energy norm. That is, we have*

$$A(v_h, v_h) \geq |||v_h|||^2_{DG}$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$.

Proof. We shall treat the terms $T(v_h, v_h)$ and $S(v_h, v_h)$ in (3.41) separately, starting with $T(v_h, v_h)$. Noting that

$$\begin{aligned}
\int_{\Omega} v_h\boldsymbol{\mu}\cdot\nabla_{\mathbf{x}} v_h \ \mathrm{d}\mathbf{x} &= \sum_{\kappa_\Omega\in\mathcal{T}_\Omega}\int_{\kappa_\Omega} v_h\boldsymbol{\mu}\cdot\nabla_{\mathbf{x}} v_h \ \mathrm{d}\mathbf{x} \\
&= \sum_{\kappa_\Omega\in\mathcal{T}_\Omega}\int_{\kappa_\Omega} \nabla_{\mathbf{x}}\cdot\left(\frac{1}{2}\boldsymbol{\mu} v_h^2\right) \ \mathrm{d}\mathbf{x} \\
&= \frac{1}{2}\sum_{\kappa_\Omega\in\mathcal{T}_\Omega}\int_{\partial\kappa_\Omega} (\boldsymbol{\mu}\cdot\mathbf{n})(v_h^+)^2 \ \mathrm{d}s, \\
&= \frac{1}{2}\sum_{\kappa_\Omega\in\mathcal{T}_\Omega}\left[(v_h^+, v_h^+)_{\partial_+\kappa_\Omega(\boldsymbol{\mu})} - (v_h^+, v_h^+)_{\partial_-\kappa_\Omega(\boldsymbol{\mu})}\right],
\end{aligned}$$

we can write

$$
\begin{aligned}
T(v_h, v_h) &= \int_{\mathbb{Y}} \int_{\mathbb{S}} \int_{\Omega} -v_h \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} v_h + (\alpha + \beta) v_h^2 \ \mathrm{d}\mathbf{x}\, \mathrm{d}\boldsymbol{\mu}\, \mathrm{d}E \\
&\quad + \int_{\mathbb{Y}} \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[(v_h^+, v_h^+)_{\partial_+ \kappa_\Omega(\boldsymbol{\mu})} - (v_h^-, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} \right] \ \mathrm{d}\boldsymbol{\mu}\, \mathrm{d}E \\
&= ||(\alpha + \beta)^{\frac{1}{2}} v_h||^2_{L^2(\mathcal{D})} \\
&\quad - \frac{1}{2} \int_{\mathbb{Y}} \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[(v_h^+, v_h^+)_{\partial_+ \kappa_\Omega(\boldsymbol{\mu})} - (v_h^+, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu})} \right] \ \mathrm{d}\boldsymbol{\mu}\, \mathrm{d}E \\
&\quad + \int_{\mathbb{Y}} \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[(v_h^+, v_h^+)_{\partial_+ \kappa_\Omega(\boldsymbol{\mu})} - (v_h^-, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} \right] \ \mathrm{d}\boldsymbol{\mu}\, \mathrm{d}E \\
&= ||(\alpha + \beta)^{\frac{1}{2}} v_h||^2_{L^2(\mathcal{D})} \\
&\quad + \frac{1}{2} \int_{\mathbb{Y}} \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[(v_h^+, v_h^+)_{\partial_+ \kappa_\Omega(\boldsymbol{\mu})} - 2(v_h^-, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} \right. \\
&\qquad\qquad \left. + (v_h^+, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu})} \right] \ \mathrm{d}\boldsymbol{\mu}\, \mathrm{d}E
\end{aligned}
$$

We now manipulate the remaining face integrals by conditioning on whether the face is adjacent to the spatial boundary:

$$
\begin{aligned}
&\sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[(v_h^+, v_h^+)_{\partial_+ \kappa_\Omega(\boldsymbol{\mu})} - 2(v_h^-, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} + (v_h^+, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu})} \right] \\
&\quad = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[(v_h^+, v_h^+)_{\partial_+ \kappa_\Omega(\boldsymbol{\mu}) \cap \partial\Omega} + (v_h^+, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \cap \partial\Omega} \right] \\
&\qquad + \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[(v_h^+, v_h^+)_{\partial_+ \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} - 2(v_h^-, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} + (v_h^+, v_h^+)_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} \right] \\
&\quad = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[(v_h^+, v_h^+)_{\partial \kappa_\Omega \cap \partial\Omega} + (v_h^+ - v_h^-, v_h^+ - v_h^-)_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} \right] \\
&\quad = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[|v^+ - v^-|^2_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} + |v^+|^2_{\partial \kappa_\Omega \cap \partial\Omega} \right].
\end{aligned}
$$

We therefore have

$$
T(v_h, v_h) = ||(\alpha + \beta)^{\frac{1}{2}} v_h||^2_{L^2(\mathcal{D})} + \frac{1}{2} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left[|v^+ - v^-|^2_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial\Omega} + |v^+|^2_{\partial \kappa_\Omega \cap \partial\Omega} \right].
$$

For the treatment of $S(v_h, v_h)$, we may use Lemma 3.3.1 together with the arithmetic-geometric mean inequality:

$$
\begin{aligned}
S(v_h, v_h) &= \int_{\mathbb{Y}} \int_{\mathbb{S}} \int_{\Omega} S[v_h] v_h \ \mathrm{d}\mathbf{x}\, \mathrm{d}\boldsymbol{\mu}\, \mathrm{d}E \\
&\leq ||\beta^{\frac{1}{2}} v_h||_{L^2(\mathcal{D})} \ ||\gamma^{\frac{1}{2}} v_h||_{L^2(\mathcal{D})} \\
&\leq \frac{1}{2} \left(||\beta^{\frac{1}{2}} v_h||^2_{L^2(\mathcal{D})} + ||\gamma^{\frac{1}{2}} v_h||^2_{L^2(\mathcal{D})} \right).
\end{aligned}
$$

Putting everything together, we have

$$\begin{aligned} A(v_h, v_h) &\geq ||(\alpha+\beta)^{\frac{1}{2}} v_h||^2_{L^2(\mathcal{D})} - \frac{1}{2}\left(||\beta^{\frac{1}{2}} v_h||^2_{L^2(\mathcal{D})} + ||\gamma^{\frac{1}{2}} v_h||^2_{L^2(\mathcal{D})}\right) \\ &\quad + \frac{1}{2}\int_{\mathbb{Y}}\int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left(|v_h^+ - v_h^-|^2_{\partial_- \kappa_\Omega(\boldsymbol{\mu})\setminus\partial\Omega} + |v_h^+|^2_{\partial\kappa_\Omega\cap\partial\Omega}\right) \mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E \\ &= |||v_h|||^2_{DG}. \end{aligned}$$

□

We now present a summary of the stability and convergence results for the poly-energetic DGFEM scheme as proven in [53]. We henceforth assume that the spatial mesh $\mathcal{T}_\Omega$ satisfies the following assumptions:

- $\mathcal{T}_\Omega$ is shape-regular; that is, there exists a positive constant C_{shape}, independent of the mesh parameters, such that

  $$\frac{h_{\kappa_\Omega}}{\rho_{\kappa_\Omega}} \leq C_{\text{shape}}$$

  for all $\kappa_\Omega \in \mathcal{T}_\Omega$, where ρ_{κ_Ω} denotes the diameter of the largest ball contained in κ_Ω.

- Every spatial element $\kappa_\Omega \in \mathcal{T}_\Omega$ has at most C_F $(d-1)$-dimensional boundary faces, where $C_F \leq \infty$ is independent of the mesh parameters.

- There exist constants $n_{\mathcal{T}_\Omega} \in \mathbb{N}$ and $\hat{c} > 0$, independent of $\mathcal{T}_\Omega$, such that every spatial element $\kappa_\Omega \in \mathcal{T}_\Omega$ admits a sub-triangulation into at most $n_{\mathcal{T}_\Omega} \leq \infty$ shape-regular simplices $\kappa_\Omega^{(i)}$, $1 \leq i \leq n_{\mathcal{T}_\Omega}$, such that $\bar{\kappa_\Omega} = \cup_{i=1}^{n_{\mathcal{T}_\Omega}} \bar{\kappa_\Omega}^{(i)}$ and

  $$|\kappa_{\mathcal{T}_\Omega}^{(i)}| \geq \hat{c}|\kappa_{\mathcal{T}_\Omega}|$$

  for all $1 \leq i \leq n_{\mathcal{T}_\Omega}$.

Under these assumptions on the spatial mesh, the following inf-sup result with respect to the streamline norm $|||\cdot|||_s$ in (3.39) is proven in [53].

Theorem 3.3.3 (Inf-sup stability)**.** *The poly-energetic DGFEM scheme (3.22) is inf-sup stable in the streamline norm; that is, there exists a constant $\Lambda > 0$, independent of discretisation parameters, such that*

$$\inf_{v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}\setminus\{0\}} \sup_{w_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}\setminus\{0\}} \frac{A(v_h, w_h)}{|||v_h|||_s |||w_h|||_s} \geq \Lambda.$$

Also under the previous assumptions on the spatial mesh, the following *a priori* convergence result with respect to the streamline norm $|||\cdot|||_s$ is proven in [53].

Theorem 3.3.4 (*A priori* convergence in the streamline norm)**.** *Let $u_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ denote the DGFEM solution to (3.22) approximating $u \in H^1(\mathcal{D})$, the solution to (3.4). Furthermore, for each $\kappa = \kappa_\Omega \times \kappa_\mathbb{S} \times \kappa_\mathbb{Y} \in \mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}$, assume that $u|_\kappa \in H^{l_\kappa}(\kappa) \cup H^1(\kappa_\Omega; H^{l_\kappa}(\kappa_\mathbb{S} \times \kappa_\mathbb{Y}))$ for $l_\kappa > 1$. Then the following error bound holds:*

$$
\begin{aligned}
|||u-u_h|||_s^2 \leq C \sum_{\kappa \in \mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}} \Bigg(& \frac{h_{\kappa_\Omega}^{2s_{\kappa_\Omega}}}{p_{\kappa_\Omega}^{2l_\kappa}} \Big(\mathcal{L}_\kappa(\alpha,\beta,\gamma) + \frac{p_{\kappa_\Omega}^2}{h_{\kappa_\Omega}^\perp} + \frac{h_{\kappa_\Omega}^\perp}{h_{\kappa_\Omega}^2} + \frac{h_{\kappa_\Omega}^2}{h_{\kappa_\Omega}^\perp} \Big) ||u||_{H^{l_\kappa}(\kappa)}^2 \\
& + \Big(\frac{h_{\kappa_\mathbb{S}}^{2s_{\kappa_\mathbb{S}}}}{q_{\kappa_\mathbb{S}}^{2l_\kappa}} + \frac{h_{\mathbb{Y}}^{2s_{\kappa_\mathbb{Y}}}}{r_{\kappa_\mathbb{Y}}^{2l_\kappa}} \Big) \Big(\Big(\mathcal{L}_\kappa(\alpha,\beta,\gamma) + \frac{1}{h_{\kappa_\Omega}^\perp}(1+p_{\kappa_\Omega}^2) \Big) ||u||_{H^{l_\kappa}(\kappa)}^2 \\
& + \Big(\frac{h_{\kappa_\Omega}^2}{h_{\kappa_\Omega}^\perp} + \frac{h_{\kappa_\Omega}^\perp}{p_{\kappa_\Omega}^2} \Big) ||u||_{H^1(\kappa_\Omega; H^{l_\kappa}(\kappa_\mathbb{S} \times \kappa_\mathbb{Y}))}^2 \Big) \Big),
\end{aligned}
$$

where $\mathcal{L}_\kappa(\alpha,\beta,\gamma) = ||\bar{\alpha}||_{L^\infty(\kappa)} + (||\alpha+\beta||_{L^\infty(\kappa)}^2 + ||\beta||_{L^\infty(\kappa)}||\gamma||_{L^\infty(\kappa)})/\alpha_0$, $s_{\kappa_\Omega} = \min\{p_{\kappa_\Omega}, l_\kappa\}$, $s_{\kappa_\mathbb{S}} = \min\{q_{\kappa_\mathbb{S}}, l_\kappa\}$, $s_{\kappa_\mathbb{Y}} = \{r_{\kappa_\mathbb{Y}}, l_\kappa\}$, and C is a positive constant independent of the discretisation parameters.

Remark. *The discretisation parameters referred to in the result of Theorem 3.3.4 are the spatial, angular and energetic mesh-size parameters h_{κ_Ω}, $h_{\kappa_\mathbb{S}}$ and $h_{\kappa_\mathbb{Y}}$, respectively, and the spatial, angular and energetic polynomial degrees p_{κ_Ω}, $q_{\kappa_\mathbb{S}}$ and $r_{\kappa_\mathbb{Y}}$, respectively, for each $\kappa_\Omega \in \mathcal{T}_\Omega$, $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$ and $\kappa_\mathbb{Y} \in \mathcal{T}_\mathbb{Y}$.*

Remark. *Denote $h = \max\{\mathrm{diam}(\kappa) : \kappa \in \mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}\}$ and consider the case of uniform polynomial orders; that is, $p_{\kappa_\Omega} = q_{\kappa_\mathbb{S}} = r_{\kappa_\mathbb{Y}} = p$ for all $\kappa_\Omega \in \mathcal{T}_\Omega$, $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$ and $\kappa_\mathbb{Y} \in \mathcal{T}_\mathbb{Y}$, and $s_\kappa = s = \min\{p+1, l\}$ for all $\kappa \in \mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}$, $l \geq 1$. Furthermore, assume that the diameter of the spatial faces of each $\kappa_\Omega \in \mathcal{T}_\Omega$ is comparable to the diameter of the element, so that $h_{\kappa_\Omega}^\perp \sim h_{\kappa_\Omega}$. The* a priori *error bound given in Theorem 3.3.4 simplifies to*

$$
|||u-u_h|||_s^2 \leq C \frac{h^{s-1/2}}{p^{l-1}} ||u||_{H^l(\mathcal{D})}^2,
$$

where C is a constant independent of the discretisation parameters. This bound is optimal with respect to the space-angle-energy mesh size parameter h, but suboptimal in the space-angle-energy polynomial degree of approximation p by half an order [27].

3.4 Discrete Ordinates Galerkin (DOG) Implementation

The poly-energetic DGFEM scheme (3.22), on first glance, appears to couple the spatial, angular and energetic degrees of freedom in both of the bilinear forms $T(\cdot,\cdot)$ and $S(\cdot,\cdot)$. By careful selection of the angular and energetic basis functions, we can approximately rewrite the scheme as a multigroup discrete ordinates scheme for which standard iterative procedures requiring the solution of (sequences of) spatial transport problems may be employed; see Chapter 5. The key observations are that:

- no derivatives of the test or trial functions in (3.22) nor jump terms are taken with respect to the angular or energetic variables;
- for any $p \in \mathbb{N}_0$, $k \in \mathbb{N} \setminus \{0\}$ and any k-dimensional tensor-product element κ, there exists a quadrature scheme consisting of $N = \dim \mathbb{Q}^p(\kappa)$ quadrature points and weights that can exactly integrate the product of any two elements of $\mathbb{Q}^p(\kappa)$ over κ;
- such a quadrature scheme can be used to define an orthogonal basis of $\mathbb{Q}^k(\kappa)$.

3.4.1 Implementation in energy

The poly-energetic DGFEM scheme (3.22) can be solved sequentially on a per-energy-group basis, starting with the highest energy group (corresponding to $g = 1$) and ending with the lowest energy group (corresponding to $g = N_{\mathbb{Y}}$). To see this, consider the problem of solving (3.4) for $u(\mathbf{x}, \boldsymbol{\mu}, \hat{E})$ at a fixed energy $\hat{E}$:

$$\begin{aligned} \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}, \hat{E}) + (\alpha(\mathbf{x}, \boldsymbol{\mu}, \hat{E}) + \beta(\mathbf{x}, \boldsymbol{\mu}, \hat{E}))u(\mathbf{x}, \boldsymbol{\mu}, \hat{E}) \\ = S[u](\mathbf{x}, \boldsymbol{\mu}, \hat{E}) + f(\mathbf{x}, \boldsymbol{\mu}, \hat{E}) \quad \text{in } \mathcal{D}, \\ u(\mathbf{x}, \boldsymbol{\mu}, \hat{E}) = g(\mathbf{x}, \boldsymbol{\mu}, \hat{E}) \quad \text{on } \partial\mathcal{D}. \end{aligned}$$

We recognise that the left-hand-side of the first equation specifies a mono-energetic problem for $u(\mathbf{x}, \boldsymbol{\mu}, \hat{E})$, and only the right-hand-side of the first equation involves a coupling over the whole energy domain. However, by the definition of the scattering operator $S[\cdot]$ in (3.5) and the assumption $\theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E' \to E) > 0$ only when $E' > E$, it can be shown that the energy cut-off function

$$u^+(\mathbf{x}, \boldsymbol{\mu}, E) = \begin{cases} u(\mathbf{x}, \boldsymbol{\mu}, E) & \text{for } E > \hat{E}, \\ 0 & \text{otherwise,} \end{cases}$$

satisfies $S[u](\mathbf{x}, \boldsymbol{\mu}, \hat{E}) = S[u^+](\mathbf{x}, \boldsymbol{\mu}, \hat{E})$. Therefore, assuming that u^+ is known beforehand, the scattering operator in the poly-energetic LBTE acts as a source term for a fixed-energy mono-energetic problem.

Extending this idea to the discretised setting, we denote by $\kappa_g = (E_g, E_{g-1})$ the g^{th} energy group, $1 \leq g \leq N_{\mathbb{Y}}$, and introduce the following family of cutoff energy functions:

$$u_g^+(\mathbf{x}, \boldsymbol{\mu}, E) = \begin{cases} u_h(\mathbf{x}, \boldsymbol{\mu}, E) & \text{for } E \geq E_{g-1}, \\ 0 & \text{otherwise.} \end{cases}$$

Notice that $u_g^+ = 0$ for the highest energy group $g = 1$.

On each energy group κ_g, the approximate fluence can be expanded in terms of an energetic basis $\{\varphi_g^j\}_{j=1}^{r_{\kappa_g}+1}$ of $\mathbb{P}^{r_{\kappa_g}}(\kappa_g)$ supported on κ_g:

$$u_h(\mathbf{x}, \boldsymbol{\mu}, E)|_{\kappa_g} = u_g(\mathbf{x}, \boldsymbol{\mu}, E) = \sum_{j=1}^{r_{\kappa_g}+1} u_g^j(\mathbf{x}, \boldsymbol{\mu}) \varphi_g^j(E).$$

Here, each $u_g^j \in \mathcal{V}_{\Omega,\mathbb{S}}$. Notice that u_g^+ and u_g may be rewritten respectively as

$$u_g^+(\mathbf{x},\boldsymbol{\mu},E) = \sum_{g'=1}^{g-1} u_{g'}(\mathbf{x},\boldsymbol{\mu},E) = \sum_{g'=1}^{g-1}\sum_{j=1}^{r_{\kappa_{g'}}+1} u_{g'}^j(\mathbf{x},\boldsymbol{\mu})\varphi_{g'}^j(E),$$

$$u_h(\mathbf{x},\boldsymbol{\mu},E) = \sum_{g'=1}^{N_{\mathbb{Y}}}\sum_{j=1}^{r_{\kappa_g'}+1} u_{g'}^j(\mathbf{x},\boldsymbol{\mu})\varphi_{g'}^j(E).$$

In particular, u_g^+ can be thought of as a truncation (or restriction) of u_h to the first $g-1$ energy groups.

By selecting test functions of the form $v_h = v_g\varphi_g^i \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ with $v_g \in \mathcal{V}_{\Omega,\mathbb{S}}$ and substituting the expression for u_h into (3.22), we arrive at the following problem: for each $1 \le g \le N_{\mathbb{Y}}$, find $\{u_g^j\}_{j=1}^{r_{\kappa_g}+1} \subset \mathcal{V}_{\Omega,\mathbb{S}}$ such that

$$\begin{aligned}\sum_{j=1}^{r_{\kappa_g}+1}\big(T(u_g^j\varphi_g^j, v_g\varphi_g^i) - S(u_g^j\varphi_g^j, v_g\varphi_g^i)\big) &= S(u_g^+, v_g\varphi_g^i) + \ell(v_g\varphi_g^i) \\ &= \sum_{g'=1}^{g-1}\sum_{j=1}^{r_{\kappa_{g'}}+1} S(u_{g'}^j\varphi_{g'}^j, v_g\varphi_g^i) + \ell(v_g\varphi_g^i)\end{aligned} \tag{3.42}$$

for all $v_g \in \mathcal{V}_{\Omega,\mathbb{S}}$ and $1 \le i \le r_{\kappa_g}+1$. We have thus rewritten the original poly-energetic DGFEM scheme (3.22) as a collection of $N_{\mathbb{Y}}$ sub-problems which may be solved sequentially.

We shall now simplify the structure of the poly-energetic DGFEM scheme above by prescribing the basis functions $\{\varphi_g^j\}_{j=1}^{r_{\kappa_g}+1}$. For a given energy group κ_g, let $\{E_g^q\}_{q=1}^{r_{\kappa_g}+1} \subset \kappa_g$ denote the $r_{\kappa_g}+1$ Gauss-Legendre quadrature points on κ_g with associated weights $\{\omega_g^q\}_{q=1}^{r_{\kappa_g}+1} \subset \mathbb{R}_{\ge 0}$. Let φ_g^i be defined for $E \in \kappa_g$ as the Lagrange interpolating polynomial

$$\varphi_g^i(E) = \prod_{\substack{q=1\\q\neq i}}^{r_{\kappa_g}+1} \frac{E - E_g^q}{E_g^i - E_g^q}$$

and $\varphi_g^i(E) = 0$ otherwise. Note that we have $\varphi_g^i(E_g^j) = \delta_{ij}$ for all $1 \le i,j \le r_{\kappa_g}+1$, where δ_{ij} denotes the Kronecker delta function defined by $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ otherwise. Moreover, since $\deg \varphi_g^i(E)\varphi_g^j(E) \le 2r_{\kappa_g}$ and the quadrature rule

$$\int_{\kappa_g} f(E)\ \mathrm{d}E \approx \sum_{q=1}^{r_{\kappa_g}+1} \omega_g^q f(E_g^q)$$

is exact whenever $f(E)$ is a polynomial of degree at most $2r_{\kappa_g}+1$, we have that

$$\int_{\kappa_g} \varphi_g^i(E)\varphi_g^j(E)\ \mathrm{d}E = \sum_{q=1}^{r_{\kappa_g}+1} \omega_g^q\varphi_g^i(E_g^q)\varphi_g^j(E_g^q) = \omega_g^i\delta_{ij}\delta_{iq}. \tag{3.43}$$

That is, the basis functions $\{\varphi_g^j\}_{j=1}^{r_{\kappa_g}+1}$ are orthogonal with respect to the $L^2(\kappa_g)$-inner product.

We will approximate the energetic integrals in the terms $T(u_g^j\varphi_g^j, v_g\varphi_g^i)$ and $\ell(v_g\varphi_g^i)$ with an energetic quadrature scheme. While any appropriately-high-order quadrature scheme may be used in principle, we select the energetic quadrature scheme to

be the same Gauss-Legendre quadrature scheme defining the energetic basis functions $\{\varphi_g^j\}_{j=1}^{r_{\kappa_g}+1}$. We have

$$\begin{aligned} T(u_g^j\varphi_g^j, v_g\varphi_g^i) &\approx \sum_{q=1}^{r_{\kappa_g}+1} \omega_g^q \varphi_g^j(E_g^q)\varphi_g^i(E_g^q)\tilde{T}_g^q(u_g^j, v_g) \\ &= \omega_g^i \tilde{T}_g^i(u_g^i, v_g)\delta_{ij}, \end{aligned} \tag{3.44}$$

$$\begin{aligned} \ell(v_g\varphi_g^i) &\approx \sum_{q=1}^{r_{\kappa_g}+1} \omega_g^q \varphi_g^i(E_g^q)\tilde{\ell}_g^q(v_g) \\ &= \omega_g^i \tilde{\ell}_g^i(v_g), \end{aligned} \tag{3.45}$$

where the bilinear forms $\tilde{T}_g^i : \mathcal{V}_{\Omega,\mathbb{S}} \times \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ and linear functionals $\tilde{\ell}_g^i : \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ are defined for all $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ by

$$\begin{aligned} \tilde{T}_g^i(w_h, v_h) = \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Bigg(\int_{\kappa_\Omega} \left(-w_h \boldsymbol{\mu}\cdot\nabla_{\mathbf{x}} v_h + (\alpha(\mathbf{x},\boldsymbol{\mu},E_g^i) + \beta(\mathbf{x},\boldsymbol{\mu},E_g^i))w_h v_h\right) \,\mathrm{d}\mathbf{x} \\ + \int_{\partial_+\kappa_\Omega(\boldsymbol{\mu})} |\boldsymbol{\mu}\cdot\mathbf{n}_\kappa| w_h^+ v_h^+ \,\mathrm{d}s - \int_{\partial_-\kappa_\Omega(\boldsymbol{\mu})\setminus\partial\Omega} |\boldsymbol{\mu}\cdot\mathbf{n}_{\kappa_\Omega}| w_h^- v_h^+ \,\mathrm{d}s \Bigg) \mathrm{d}\boldsymbol{\mu}, \\ \tilde{\ell}_g^i(v_h) = \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Bigg(\int_{\kappa_\Omega} f(\mathbf{x},\boldsymbol{\mu},E_g^i) v_h \,\mathrm{d}\mathbf{x} \\ + \int_{\partial_-\kappa_\Omega(\boldsymbol{\mu})\cap\partial\Omega} |\boldsymbol{\mu}\cdot\mathbf{n}_{\kappa_\Omega}| g(\mathbf{x},\boldsymbol{\mu},E_g^i) v_h^+ \,\mathrm{d}s \Bigg) \,\mathrm{d}\boldsymbol{\mu}. \end{aligned}$$

That is, $\tilde{T}_g^i(\cdot,\cdot)$ is precisely the mono-energetic bilinear form $T(\cdot,\cdot)$ in (3.30) with the coefficient data α and β evaluated at the energy E_g^i, and $\tilde{\ell}_g^i(\cdot)$ is precisely the mono-energetic linear functional $\ell(\cdot)$ in (3.32) with the forcing data f and g evaluated at the energy E_g^i.

We will treat the scattering bilinear forms $S(u_{g'}^j\varphi_{g'}^j, v_g\varphi_g^i)$ in a different manner. Instead, we proceed by writing the bilinear form in full:

$$\begin{aligned} S(u_{g'}^j\varphi_{g'}^j, v_g\varphi_g^i) = \int_{\mathbb{Y}}\int_{\mathbb{S}}\int_{\Omega}\int_{\mathbb{Y}}\int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}', E'\to E)\varphi_g^i(E)\varphi_{g'}^j(E')\cdot \\ u_{g'}^j(\mathbf{x},\boldsymbol{\mu}')v_g(\mathbf{x},\boldsymbol{\mu}) \,\mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}E'\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E. \end{aligned}$$

We define the following family of mono-energetic scattering kernels for each $1 \le i \le r_{\kappa_g}+1$, $1 \le j \le r_{\kappa_{g'}}+1$ and $1 \le g, g' \le N_{\mathbb{Y}}$:

$$\Theta_{g',g}^{j,i}(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}') = \int_{\mathbb{Y}}\int_{\mathbb{Y}} \theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}', E'\to E)\varphi_g^i(E)\varphi_{g'}^j(E') \,\mathrm{d}E'\,\mathrm{d}E.$$

Then we may write $S(u_{g'}^j\varphi_{g'}^j, v_g\varphi_g^i) = \tilde{S}_{g',g}^{j,i}(u_{g'}^j, v_g)$, where the bilinear form $\tilde{S}_{g',g}^{j,i} : \mathcal{V}_{\Omega,\mathbb{S}} \times \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ is defined for all $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ by

$$\tilde{S}_{g',g}^{j,i}(w_h, v_h) = \int_{\mathbb{S}}\int_{\Omega}\int_{\mathbb{S}} \Theta_{g',g}^{j,i}(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}')w_h(\mathbf{x},\boldsymbol{\mu}')v_h(\mathbf{x},\boldsymbol{\mu}) \,\mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu}. \tag{3.46}$$

Substituting the approximations of the terms $T(u_g^j\varphi_g^j, v_g\varphi_g^i)$, $S(u_{g'}^j\varphi_{g'}^j, v_g\varphi_g^i)$ and $\ell(v_g\varphi_g^i)$ (from (3.44), (3.46) and (3.45) respectively) into the rewritten poly-energetic

DGFEM scheme (3.42) yields the following problem: for each $1 \leq g \leq N_{\mathbb{Y}}$, find $\{u_g^j\}_{j=1}^{r_{\kappa_g}+1} \subset \mathcal{V}_{\Omega,\mathbb{S}}$ such that

$$\omega_g^i \tilde{T}_g^i(u_g^i, v_g) - \sum_{j=1}^{r_{\kappa_g}+1} \tilde{S}_{g,g}^{j,i}(u_g^j, v_g) = \sum_{g'=1}^{g-1} \sum_{j=1}^{r_{\kappa_{g'}}+1} \tilde{S}_{g',g}^{j,i}(u_{g'}^j, v_g) + \omega_g^i \tilde{\ell}_g^i(v_g) \tag{3.47}$$

for all $v_g \in \mathcal{V}_{\Omega,\mathbb{S}}$.

The result of this treatment of (3.22) is a method in which the approximate fluence $u_h(\mathbf{x}, \boldsymbol{\mu}, E)$ on each energy group κ_g requires the solution of $r_{\kappa_g} + 1$ mono-energetic DGFEM problems which are coupled only through a term representing the scattering of the group angular flux u_g between different energies in the current energy group; this is given by the sum on the left-hand-side of (3.47). The solution of these systems in high energy groups are subsequently used as incoming forcing terms for the problems in low energy groups.

A special case of (3.47) occurs when $r_{\kappa_g} = 0$ for all $\kappa_g \in \mathcal{T}_{\mathbb{Y}}$; i.e. when a piecewise-constant approximation of u in energy is sought. Here, the poly-energetic DGFEM scheme reduces to the classical multigroup approximation: for each $1 \leq g \leq N_{\mathbb{Y}}$, find $u_g \in \mathcal{V}_{\Omega,\mathbb{S}}$ such that

$$\tilde{T}_g(u_g, v_g) - \frac{1}{|\kappa_g|} \tilde{S}_{g,g}(u_g, v_g) = \frac{1}{|\kappa_g|} \sum_{g'=1}^{g-1} \tilde{S}_{g',g}(u_{g'}, v_g) + \tilde{\ell}_g(v_g)$$

for all $v_g \in \mathcal{V}_{\Omega,\mathbb{S}}$. In this case, the data/forcing terms in $\tilde{T}_g(\cdot,\cdot)$ and $\tilde{\ell}_g(\cdot)$ are evaluated at the midpoint of κ_g, and the scattering bilinear form $\tilde{S}_{g',g}(\cdot,\cdot)$ assumes the following form for all $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$:

$$\tilde{S}_{g',g}(w_h, v_h) = \int_{\mathbb{S}} \int_{\Omega} \int_{\mathbb{S}} \Theta_{g',g}(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') w_h(\mathbf{x}, \boldsymbol{\mu}') v_h(\mathbf{x}, \boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu},$$

where $\Theta_{g',g}(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}')$ is defined by

$$\Theta_{g',g}(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') = \int_{\kappa_g} \int_{\kappa_{g'}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E' \to E) \ \mathrm{d}E' \, \mathrm{d}E.$$

While the numerical solution of (3.47) is deferred to Chapter 5, we shall provide an example of a stationary iterative method for the solution of (3.47) within an energy group. For each $1 \leq i \leq r_{\kappa_g} + 1$, let $f_g^i : \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ denote the linear functional

$$f_g^i(v_g) = \sum_{g'=1}^{g-1} \sum_{j=1}^{r_{\kappa_{g'}}+1} \tilde{S}_{g',g}^{j,i}(u_{g'}^j, v_g) + \omega_g^i \tilde{\ell}_g^i(v_g)$$

for all $v_g \in \mathcal{V}_{\Omega,\mathbb{S}}$. Note that f_g^i requires prior knowledge of $u_{g'}^j$ for all higher-energy basis functions $1 \leq j \leq r_{\kappa_{g'}} + 1$ and all higher energy groups $1 \leq g' \leq g - 1$. For each $1 \leq i \leq r_{\kappa_g} + 1$, we introduce a sequence $\{u_g^{i,n}\}_{n \geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S}}$ of approximations to the true solution u_g^i of (3.47) for some initial guess $u_g^{i,0}$. For each $n \geq 1$, we define $u_g^{i,n}$ as the solution to the variational problem

$$\omega_g^i \tilde{T}_g^i(u_g^{i,n}, v_g) = \sum_{j=1}^{r_{\kappa_g}+1} \tilde{S}_{g,g}^{j,i}(u_g^{j,n-1}, v_g) + f_g^i(v_g) \tag{3.48}$$

for all $v_g \in \mathcal{V}_{\Omega,\mathbb{S}}$. Notice that the group scattering term appearing in the left-hand-side of (3.47) now corresponds to the right-hand sum in (3.48), where we have replaced u_g^j with $u_g^{j,n-1}$. We defer the question of whether such an iterative method converges to Chapter 5.

Upon selection of a basis $\{\phi_a\}_{a=1}^N \subseteq \mathcal{V}_{\Omega,\mathbb{S}}$, $N = \dim \mathcal{V}_{\Omega,\mathbb{S}}$, it can be seen that the iteration above can be expressed in the following block-matrix form:

$$\begin{pmatrix} \omega_g^1 \mathbf{T}_g^1 & & & \\ & \omega_g^2\ \mathbf{T}_g^2 & & \\ & & \ddots & \\ & & & \omega_g^{r_{\kappa_g}+1} \mathbf{T}_g^{r_{\kappa_g}+1} \end{pmatrix} \begin{pmatrix} \mathbf{u}_g^{1,n} \\ \mathbf{u}_g^{2,n} \\ \vdots \\ \mathbf{u}_g^{r_{\kappa_g}+1,n} \end{pmatrix}$$
$$= \begin{pmatrix} \mathbf{S}_{g,g}^{1,1} & \mathbf{S}_{g,g}^{2,1} & \dots & \mathbf{S}_{g,g}^{r_{\kappa_g}+1,1} \\ \mathbf{S}_{g,g}^{1,2} & \mathbf{S}_{g,g}^{2,2} & \dots & \mathbf{S}_{g,g}^{r_{\kappa_g}+1,2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_{g,g}^{1,r_{\kappa_g}+1} & \mathbf{S}_{g,g}^{2,r_{\kappa_g}+1} & \dots & \mathbf{S}_{g,g}^{r_{\kappa_g}+1,r_{\kappa_g}+1} \end{pmatrix} \begin{pmatrix} \mathbf{u}_g^{1,n-1} \\ \mathbf{u}_g^{2,n-1} \\ \vdots \\ \mathbf{u}_g^{r_{\kappa_g}+1,n-1} \end{pmatrix} + \begin{pmatrix} \mathbf{f}_g^1 \\ \mathbf{f}_g^2 \\ \vdots \\ \mathbf{f}_g^{r_{\kappa_g}+1} \end{pmatrix}.$$

Here, $\mathbf{T}_g^i$, $\mathbf{S}_{g,g}^{j,i} \in \mathbb{R}^{N\times N}$ and $\mathbf{f}_g^i \in \mathbb{R}^N$ are matrix/vector representations of $\tilde{T}_g^i(\cdot,\cdot)$, $\tilde{S}_{g,g}^{j,i}(\cdot,\cdot)$ and $f_g^i(\cdot)$ respectively, and $\mathbf{u}_g^{i,n}$ represents the coefficients in the expansion of $u_g^{i,n}$ in the basis $\{\phi_a\}_{a=1}^N \subseteq \mathcal{V}_{\Omega,\mathbb{S}}$; that is,

$$u_g^{i,n}(\mathbf{x},\boldsymbol{\mu}) = \sum_{b=1}^{N} (\mathbf{u}_g^{i,n})_b \phi_b(\mathbf{x},\boldsymbol{\mu}).$$

By setting $v_g = \phi_a$ for $1 \le a \le N$, the matrix/vector terms are defined entry-wise by

$$\begin{aligned} (\mathbf{T}_g^i)_{ab} &= \tilde{T}_g^i(\phi_b,\phi_a), \\ (\mathbf{S}_{g,g}^{j,i})_{ab} &= \tilde{S}_{g,g}^{j,i}(\phi_b,\phi_a), \\ (\mathbf{f}_g^i)_a &= f_g^i(\phi_a) \end{aligned}$$

for $1 \le a,b \le N$. Owing to the angular coupling in the definition of $\tilde{S}_{g,g}^{j,i}(\cdot,\cdot)$, the matrices $\mathbf{S}_{g,g}^{j,i}$ are generally denser that $\mathbf{T}_g^i$; we refer to Chapter 3.4.2 for more details about their structure.

3.4.2 Implementation in angle

We may apply the same methodology for approximately implementing the mono-energetic DGFEM scheme (3.29). On each angular element $\kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}$, the approximate mono-energetic fluence can be expanded in terms of an angular basis $\{\varphi_{\kappa_{\mathbb{S}}}^j\}_{j=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}}$ of $\mathbb{Q}^{q_{\kappa_{\mathbb{S}}}}(\kappa_{\mathbb{S}})$ supported on $\kappa_{\mathbb{S}}$:

$$u_h(\mathbf{x},\boldsymbol{\mu})|_{\kappa_{\mathbb{S}}} = u_{\kappa_{\mathbb{S}}}(\mathbf{x},\boldsymbol{\mu}) = \sum_{j=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}} u_{\kappa_{\mathbb{S}}}^j(\mathbf{x}) \varphi_{\kappa_{\mathbb{S}}}^j(\boldsymbol{\mu}).$$

Here, each $u^j_{\kappa_\mathbb{S}} \in \mathcal{V}_\Omega$. Notice that u_h may be rewritten as

$$u_h(\mathbf{x}, \boldsymbol{\mu}) = \sum_{\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}} u_{\kappa_\mathbb{S}}(\mathbf{x}, \boldsymbol{\mu}) = \sum_{\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}} \sum_{j=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} u^j_{\kappa_\mathbb{S}}(\mathbf{x}) \varphi^j_{\kappa_\mathbb{S}}(\boldsymbol{\mu}).$$

By selecting test functions of the form $v_h = v_{\kappa_\mathbb{S}} \varphi^i_{\kappa_\mathbb{S}}$ with $v_{\kappa_\mathbb{S}} \in \mathcal{V}_\Omega$ and substituting the expression for u_h into (3.29), we arrive at the following problem: for each $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$, find $\{u^j_{\kappa_\mathbb{S}}\}_{j=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} \subset \mathcal{V}_\Omega$ such that

$$\begin{aligned} \sum_{j=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} T(u^j_{\kappa_\mathbb{S}} \varphi^j_{\kappa_\mathbb{S}}, v_{\kappa_\mathbb{S}} \varphi^i_{\kappa_\mathbb{S}}) &= S(u_h, v_{\kappa_\mathbb{S}} \varphi^i_{\kappa_\mathbb{S}}) + \ell(v_{\kappa_\mathbb{S}} \varphi^i_{\kappa_\mathbb{S}}) \\ &= \sum_{\kappa'_\mathbb{S} \in \mathcal{T}_\mathbb{S}} \sum_{j=1}^{(q_{\kappa'_\mathbb{S}}+1)^{d-1}} S(u^j_{\kappa'_\mathbb{S}} \varphi^j_{\kappa'_\mathbb{S}}, v_{\kappa_\mathbb{S}} \varphi^i_{\kappa_\mathbb{S}}) + \ell(v_{\kappa_\mathbb{S}} \varphi^i_{\kappa_\mathbb{S}}). \end{aligned} \tag{3.49}$$

for all $v_{\kappa_\mathbb{S}} \in \mathcal{V}_\Omega$ and $1 \leq i \leq (q_{\kappa_\mathbb{S}} + 1)^{d-1}$.

As was done for the poly-energetic DGFEM scheme, we shall now simplify the structure of the mono-energetic DGFEM scheme above by prescribing the basis functions $\{\varphi^j_{\kappa_\mathbb{S}}\}_{j=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}}$. We refer to Chapter 3.2.2 for further details on the construction of a basis on $\mathbb{S}$. For a given angular element $\kappa_\mathbb{S}$, let $\{\hat{\boldsymbol{\mu}}^q\}_{q=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} \subset \mathcal{K}$ denote the $(q_{\kappa_\mathbb{S}}+1)^{d-1}$ tensor-product Gauss-Legendre quadrature points on the $(d-1)$-dimensional reference element $\mathcal{K} = (-1, 1)^{d-1}$ formed by placing $q_{\kappa_\mathbb{S}} + 1$ points in each coordinate direction. Let the associated quadrature weights be denoted by $\{\hat{\omega}^q_{\kappa_\mathbb{S}}\}_{q=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} \subset \mathbb{R}_{\geq 0}$.

Let $\hat{\varphi}^i_{\kappa_\mathbb{S}} \in \mathbb{Q}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}}(\mathcal{K})$ be defined for $\hat{\boldsymbol{\mu}} \in \mathcal{K}$ as the $(d-1)$-variable Lagrange interpolating polynomial satisfying $\hat{\varphi}^i_{\kappa_\mathbb{S}}(\hat{\boldsymbol{\mu}}^q_{\kappa_\mathbb{S}}) = \delta_{iq}$ and $\hat{\varphi}^i_{\kappa_\mathbb{S}}(\hat{\boldsymbol{\mu}}) = 0$ otherwise. We define the basis function $\varphi^i_{\kappa_\mathbb{S}}$ for $\boldsymbol{\mu} \in \kappa_\mathbb{S}$ by

$$\varphi^i_{\kappa_\mathbb{S}}(\boldsymbol{\mu}) = \hat{\varphi}^i_{\kappa_\mathbb{S}}(\chi^{-1}_{T^{-1}\kappa_\mathbb{S}} T^{-1} \boldsymbol{\mu})$$

and $\varphi^i_{\kappa_\mathbb{S}}(\boldsymbol{\mu}) = 0$ otherwise, where the map $\chi^{-1}_{T^{-1}\kappa_\mathbb{S}} \circ T^{-1} : \kappa_\mathbb{S} \to \mathcal{K}$. Finally, we set $\boldsymbol{\mu}^i_{\kappa_\mathbb{S}} = T\chi_{T^{-1}\kappa_\mathbb{S}} \hat{\boldsymbol{\mu}}^i_{\kappa_\mathbb{S}}$ and $\omega^i_{\kappa_\mathbb{S}} = \mathcal{J}(\hat{\boldsymbol{\mu}}^i_{\kappa_\mathbb{S}}) \hat{\omega}^i_{\kappa_\mathbb{S}}$, where $\mathcal{J}$ denotes the square root of the determinant of the first fundamental form of the mapping $T \circ \chi_{T^{-1}\kappa_\mathbb{S}} : \mathcal{K} \to \kappa_\mathbb{S}$. Note that we have $\varphi^i_{\kappa_\mathbb{S}}(\boldsymbol{\mu}^j_{\kappa_\mathbb{S}}) = \delta_{ij}$, but the basis functions $\{\varphi^j_{\kappa_\mathbb{S}}\}_{j=1}^{(q_{\kappa_\mathbb{S}}+1)^2}$ are generally *not* orthogonal with respect to the $L^2(\kappa_\mathbb{S})$-inner product; this is due to the inclusion of a non-polynomial Jacobian weighting term in the definition of the inner product which cannot be integrated exactly.

We will approximate the angular integrals in the terms $T(u^j_{\kappa_\mathbb{S}} \varphi^j_{\kappa_\mathbb{S}}, v_{\kappa_\mathbb{S}} \varphi^i_{\kappa_\mathbb{S}})$ and $\ell(v_{\kappa_\mathbb{S}} \varphi^i_{\kappa_\mathbb{S}})$ with an angular quadrature scheme. While any appropriately-high-order quadrature scheme may be used in principle, we select the angular quadrature scheme to be the same Gauss-Legendre quadrature scheme defining the angular basis functions

$\{\varphi^j_{\kappa_\mathbb{S}}\}_{j=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}}$. We have

$$T(u^j_{\kappa_\mathbb{S}}\varphi^j_{\kappa_\mathbb{S}}, v_{\kappa_\mathbb{S}}\varphi^i_{\kappa_\mathbb{S}}) \approx \sum_{q=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} \omega^q_{\kappa_\mathbb{S}}\varphi^j_{\kappa_\mathbb{S}}(\boldsymbol{\mu}^q_{\kappa_\mathbb{S}})\varphi^i_{\kappa_\mathbb{S}}(\boldsymbol{\mu}^q_{\kappa_\mathbb{S}})\tilde{T}^q_{\kappa_\mathbb{S}}(u^j_{\kappa_\mathbb{S}}, v_{\kappa_\mathbb{S}})$$

$$= \omega^i_{\kappa_\mathbb{S}}\tilde{T}^i_{\kappa_\mathbb{S}}(u^i_{\kappa_\mathbb{S}}, v_{\kappa_\mathbb{S}})\delta_{ij}, \tag{3.50}$$

$$\ell(v_{\kappa_\mathbb{S}}\varphi^i_{\kappa_\mathbb{S}}) \approx \sum_{q=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} \omega^q_{\kappa_\mathbb{S}}\varphi^i_{\kappa_\mathbb{S}}(\boldsymbol{\mu}^q_{\kappa_\mathbb{S}})\tilde{\ell}^q_{\kappa_\mathbb{S}}(v_{\kappa_\mathbb{S}})$$

$$= \omega^i_{\kappa_\mathbb{S}}\tilde{\ell}^i_{\kappa_\mathbb{S}}(v_{\kappa_\mathbb{S}}), \tag{3.51}$$

where the bilinear forms $\tilde{T}^i_{\kappa_\mathbb{S}} : \mathcal{V}_\Omega \times \mathcal{V}_\Omega \to \mathbb{R}$ and linear functionals $\tilde{\ell}^i_{\kappa_\mathbb{S}} : \mathcal{V}_\Omega \to \mathbb{R}$ are defined for all $w_h, v_h \in \mathcal{V}_\Omega$ by

$$\begin{aligned}\tilde{T}^i_{\kappa_\mathbb{S}}(w_h, v_h) = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Bigg(&\int_{\kappa_\Omega} \left(-w_h \boldsymbol{\mu}^i_{\kappa_\mathbb{S}} \cdot \nabla_{\mathbf{x}} v_h + (\alpha(\mathbf{x}, \boldsymbol{\mu}^i_{\kappa_\mathbb{S}}) + \beta(\mathbf{x}, \boldsymbol{\mu}^i_{\kappa_\mathbb{S}}))w_h v_h\right) \,\mathrm{d}\mathbf{x} \\ &+ \int_{\partial_+\kappa_\Omega(\boldsymbol{\mu}^i_{\kappa_\mathbb{S}})} |\boldsymbol{\mu}^i_{\kappa_\mathbb{S}} \cdot \mathbf{n}_{\kappa_\Omega}| w_h^+ v_h^+ \,\mathrm{d}s \\ &- \int_{\partial_-\kappa_\Omega(\boldsymbol{\mu}^i_{\kappa_\mathbb{S}})\setminus\partial\Omega} |\boldsymbol{\mu}^i_{\kappa_\mathbb{S}} \cdot \mathbf{n}_{\kappa_\Omega}| w_h^- v_h^+ \,\mathrm{d}s\Bigg),\end{aligned}$$

$$\tilde{\ell}^i_{\kappa_\mathbb{S}}(v_h) = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left(\int_{\kappa_\Omega} f(\mathbf{x}, \boldsymbol{\mu}^i_{\kappa_\mathbb{S}}) v_h \,\mathrm{d}\mathbf{x} + \int_{\partial_-\kappa_\Omega(\boldsymbol{\mu}^i_{\kappa_\mathbb{S}})\cap\partial\Omega} |\boldsymbol{\mu}^i_{\kappa_\mathbb{S}} \cdot \mathbf{n}_{\kappa_\Omega}| g(\mathbf{x}, \boldsymbol{\mu}^i_{\kappa_\mathbb{S}}) v_h^+ \,\mathrm{d}s \right).$$

That is, $\tilde{T}^i_{\kappa_\mathbb{S}}(\cdot,\cdot)$ is precisely the transport bilinear form $T(\cdot,\cdot)$ in (3.34) with the wind direction and coefficient data α and β evaluated at the direction $\boldsymbol{\mu}^i_{\kappa_\mathbb{S}}$, and $\tilde{\ell}^i_{\kappa_\mathbb{S}}(\cdot)$ is precisely the transport linear functional $\ell(\cdot)$ in (3.35) with the forcing data f and g evaluated at the direction $\boldsymbol{\mu}^i_{\kappa_\mathbb{S}}$.

As before, we will treat the scattering bilinear forms $S(u^j_{\kappa'_\mathbb{S}}\varphi^j_{\kappa'_\mathbb{S}}, v_{\kappa_\mathbb{S}}\varphi^i_{\kappa_\mathbb{S}})$ by first writing it in full:

$$S(u^j_{\kappa'_\mathbb{S}}\varphi^j_{\kappa'_\mathbb{S}}, v_{\kappa_\mathbb{S}}\varphi^i_{\kappa_\mathbb{S}}) = \int_\mathbb{S}\int_\Omega\int_\mathbb{S} \theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}')\varphi^i_{\kappa_\mathbb{S}}(\boldsymbol{\mu})\varphi^j_{\kappa'_\mathbb{S}}(\boldsymbol{\mu}') \cdot u^j_{\kappa'_\mathbb{S}}(\mathbf{x}) v_{\kappa_\mathbb{S}}(\mathbf{x}) \,\mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu}.$$

We define the following family of macroscopic cross-sections for each $1 \le i \le (q_{\kappa_\mathbb{S}}+1)^{d-1}$, $1 \le j \le (q_{\kappa'_\mathbb{S}}+1)^{d-1}$ and $\kappa_\mathbb{S}, \kappa'_\mathbb{S} \in \mathcal{T}_\mathbb{S}$:

$$\beta^{j,i}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(\mathbf{x}) = \int_\mathbb{S}\int_\mathbb{S} \theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}')\varphi^i_{\kappa_\mathbb{S}}(\boldsymbol{\mu})\varphi^j_{\kappa'_\mathbb{S}}(\boldsymbol{\mu}') \,\mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}\boldsymbol{\mu}.$$

Then we may write $S(u^j_{\kappa'_\mathbb{S}}\varphi^j_{\kappa'_\mathbb{S}}, v_{\kappa_\mathbb{S}}\varphi^i_{\kappa_\mathbb{S}}) = \tilde{S}^{j,i}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(u^j_{\kappa'_\mathbb{S}}, v_{\kappa_\mathbb{S}})$, where the bilinear form $\tilde{S}^{j,i}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}} : \mathcal{V}_\Omega \times \mathcal{V}_\Omega \to \mathbb{R}$ is defined for all $w_h, v_h \in \mathcal{V}_\Omega$ by

$$\tilde{S}^{j,i}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(w_h, v_h) = \int_\Omega \beta^{j,i}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(\mathbf{x}) w_h(\mathbf{x}) v_h(\mathbf{x}) \,\mathrm{d}\mathbf{x}. \tag{3.52}$$

Substituting the approximations of the terms $T(u^j_{\kappa_\mathbb{S}}\varphi^j_{\kappa_\mathbb{S}}, v_{\kappa_\mathbb{S}}\varphi^i_{\kappa_\mathbb{S}})$, $S(u^j_{\kappa'_\mathbb{S}}\varphi^j_{\kappa'_\mathbb{S}}, v_{\kappa_\mathbb{S}}\varphi^i_{\kappa_\mathbb{S}})$ and $\ell(v_{\kappa_\mathbb{S}}\varphi^i_{\kappa_\mathbb{S}})$ (from (3.50), (3.52) and (3.51) respectively) into the rewritten mono-energetic DGFEM scheme (3.49) yields the following problem: for each $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$, find $\{u^j_{\kappa_\mathbb{S}}\}_{j=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} \subset \mathcal{V}_\Omega$ such that

$$\omega^i_{\kappa_\mathbb{S}}\tilde{T}^i_{\kappa_\mathbb{S}}(u^i_{\kappa_\mathbb{S}}, v_{\kappa_\mathbb{S}}) = \sum_{\kappa'_\mathbb{S}\in\mathcal{T}_\mathbb{S}} \sum_{j=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} \tilde{S}^{j,i}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(u^j_{\kappa'_\mathbb{S}}, v_{\kappa_\mathbb{S}}) + \omega^i_{\kappa_\mathbb{S}}\tilde{\ell}^i_{\kappa_\mathbb{S}}(v_{\kappa_\mathbb{S}}) \tag{3.53}$$

for all $v_{\kappa_\mathbb{S}} \in \mathcal{V}_\Omega$.

The result of this treatment of (3.29) is a method in which the approximate mono-energetic fluence $u_h(\mathbf{x}, \boldsymbol{\mu})$ requires the solution of $\sum_{\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}} (q_{\kappa_\mathbb{S}} + 1)^{d-1}$ transport DGFEM problems which are coupled only through a scattering term. Unlike our presentation of (3.47), where we wrote the bilinear forms $\tilde{S}^{j,i}_{g,g}$ on the left-hand-side, we opt to keep the bilinear forms $\tilde{S}^{j,i}_{\kappa_\mathbb{S},\kappa_\mathbb{S}}$ (corresponding to the scattering of $u_{\kappa_\mathbb{S}}$ between different directions within $\kappa_\mathbb{S}$) on the right-hand-side of (3.53), since $u_{\kappa_\mathbb{S}}$ cannot be found sequentially for any ordering of the angular elements in $\mathcal{T}_\mathbb{S}$.

A special case of (3.53) occurs when $q_{\kappa_\mathbb{S}} = 0$ for all $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$; i.e. when a piecewise-constant approximation of u in angle is sought. Here, the poly-energetic DGFEM scheme reduces to the classical discrete ordinates scheme: for each $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$, find $u_{\kappa_\mathbb{S}}$ such that

$$\tilde{T}_{\kappa_\mathbb{S}}(u_{\kappa_\mathbb{S}}, v_{\kappa_\mathbb{S}}) = \frac{1}{|\kappa_\mathbb{S}|} \sum_{\kappa'_\mathbb{S} \in \mathcal{T}_\mathbb{S}} \tilde{S}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(u_{\kappa'_\mathbb{S}}, v_{\kappa_\mathbb{S}}) + \tilde{\ell}_{\kappa_\mathbb{S}}(v_{\kappa_\mathbb{S}})$$

for all $v_{\kappa_\mathbb{S}} \in \mathcal{V}_\Omega$. In this case, the data/forcing terms in $\tilde{T}_{\kappa_\mathbb{S}}(\cdot,\cdot)$ and $\tilde{\ell}_{\kappa_\mathbb{S}}(\cdot)$ are evaluated at the midpoint of $\kappa_\mathbb{S}$, and the scattering bilinear form $\tilde{S}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(\cdot,\cdot)$ assumes the following form for all $w_h, v_h \in \mathcal{V}_\Omega$:

$$\tilde{S}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(w_h, v_h) = \int_\Omega \beta_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(\mathbf{x}) w_h(\mathbf{x}) v_h(\mathbf{x}) \ \mathrm{d}\mathbf{x},$$

where $\beta_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(\mathbf{x})$ is defined by

$$\beta_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(\mathbf{x}) = \int_{\kappa_\mathbb{S}} \int_{\kappa'_\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}\boldsymbol{\mu}.$$

While the numerical solution of (3.53) is deferred to Chapter 5, we shall provide an example of a stationary iterative method for the solution of (3.53). For each $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$ and $1 \leq i \leq (q_{\kappa_\mathbb{S}} + 1)^{d-1}$, we introduce a sequence $\{u^{i,n}_{\kappa_\mathbb{S}}\}_{n \geq 0} \subset \mathcal{V}_\Omega$ of approximations to the true solution $u^i_{\kappa_\mathbb{S}}$ of (3.53) for some initial guess $u^{i,0}_{\kappa_\mathbb{S}}$. For each $n \geq 1$, we define $u^{i,n}_{\kappa_\mathbb{S}}$ as the solution to the variational probem

$$\omega^i_{\kappa_\mathbb{S}} \tilde{T}_{\kappa_\mathbb{S}}(u^{i,n}_{\kappa_\mathbb{S}}, v_{\kappa_\mathbb{S}}) = \sum_{\kappa'_\mathbb{S} \in \mathcal{T}_\mathbb{S}} \sum_{j=1}^{(q_{\kappa_\mathbb{S}}+1)^{d-1}} \tilde{S}^{j,i}_{\kappa'_\mathbb{S},\kappa_\mathbb{S}}(u^{j,n-1}_{\kappa'_\mathbb{S}}, v_{\kappa_\mathbb{S}}) + \omega^i_{\kappa_\mathbb{S}} \tilde{\ell}^i_{\kappa_\mathbb{S}}(v_{\kappa_\mathbb{S}})$$

for all $v_{\kappa_\mathbb{S}} \in \mathcal{V}_\Omega$. We defer the question of whether such an iterative method converges to Chapter 5.

Upon selection of a basis $\{\phi_a\}_{a=1}^N \subseteq \mathcal{V}_\Omega$, $N = \dim \mathcal{V}_\Omega$, and an ordering of the angular elements $\{\kappa_k\}_{k=1}^M$, $M = |\mathcal{T}_\mathbb{S}|$, it can be seen that the iteration above can be expressed in

the following block-matrix form:

$$\begin{pmatrix} \mathbf{T}_{\kappa_1} & & & \\ & \mathbf{T}_{\kappa_2} & & \\ & & \ddots & \\ & & & \mathbf{T}_{\kappa_M} \end{pmatrix} \begin{pmatrix} \mathbf{u}_{\kappa_1}^n \\ \mathbf{u}_{\kappa_2}^n \\ \vdots \\ \mathbf{u}_{\kappa_M}^n \end{pmatrix}$$

$$= \begin{pmatrix} \mathbf{S}_{\kappa_1,\kappa_1} & \mathbf{S}_{\kappa_2,\kappa_1} & \dots & \mathbf{S}_{\kappa_M,\kappa_1} \\ \mathbf{S}_{\kappa_1,\kappa_2} & \mathbf{S}_{\kappa_2,\kappa_2} & \dots & \mathbf{S}_{\kappa_M,\kappa_2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_{\kappa_1,\kappa_M} & \mathbf{S}_{\kappa_2,\kappa_M} & \dots & \mathbf{S}_{\kappa_M,\kappa_M} \end{pmatrix} \begin{pmatrix} \mathbf{u}_{\kappa_1}^{n-1} \\ \mathbf{u}_{\kappa_2}^{n-1} \\ \vdots \\ \mathbf{u}_{\kappa_M}^{n-1} \end{pmatrix} + \begin{pmatrix} \mathbf{f}_{\kappa_1} \\ \mathbf{f}_{\kappa_2} \\ \vdots \\ \mathbf{f}_{\kappa_M} \end{pmatrix}.$$

The matrices $\mathbf{T}_{\kappa_k}$ and $\mathbf{S}_{\kappa_m,\kappa_k}$ and the vectors $\mathbf{u}_{\kappa_k}^n$ and $\mathbf{f}_{\kappa_k}$ assume further block-matrix forms:

$$\mathbf{T}_{\kappa_k} = \begin{pmatrix} \omega_{\kappa_k}^1 \mathbf{T}_{\kappa_k}^1 & & & \\ & \omega_{\kappa_k}^2 \mathbf{T}_{\kappa_k}^2 & & \\ & & \ddots & \\ & & & \omega_{\kappa_k}^{(q_{\kappa_k}+1)^{d-1}} \mathbf{T}_{\kappa_k}^{(q_{\kappa_k}+1)^{d-1}} \end{pmatrix},$$

$$\mathbf{S}_{\kappa_m,\kappa_k} = \begin{pmatrix} \mathbf{S}_{\kappa_m,\kappa_k}^{1,1} & \mathbf{S}_{\kappa_m,\kappa_k}^{2,1} & \dots & \mathbf{S}_{\kappa_m,\kappa_k}^{(q_{\kappa_m}+1)^{d-1},1} \\ \mathbf{S}_{\kappa_m,\kappa_k}^{1,2} & \mathbf{S}_{\kappa_m,\kappa_k}^{2,2} & \dots & \mathbf{S}_{\kappa_m,\kappa_k}^{(q_{\kappa_m}+1)^{d-1},2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_{\kappa_m,\kappa_k}^{1,(q_{\kappa_k}+1)^{d-1}} & \mathbf{S}_{\kappa_m,\kappa_k}^{2,(q_{\kappa_k}+1)^{d-1}} & \dots & \mathbf{S}_{\kappa_m,\kappa_k}^{(q_{\kappa_m}+1)^{d-1},(q_{\kappa_k}+1)^{d-1}} \end{pmatrix},$$

$$\mathbf{u}_{\kappa_k}^n = \begin{pmatrix} \mathbf{u}_{\kappa_k}^{1,n} \\ \mathbf{u}_{\kappa_k}^{2,n} \\ \vdots \\ \mathbf{u}_{\kappa_k}^{(q_{\kappa_k}+1)^{d-1},n} \end{pmatrix}, \qquad \mathbf{f}_{\kappa_k}^n = \begin{pmatrix} \omega_{\kappa_k}^1 \mathbf{f}_{\kappa_k}^1 \\ \omega_{\kappa_k}^2 \mathbf{f}_{\kappa_k}^2 \\ \vdots \\ \omega_{\kappa_k}^{(q_{\kappa_k}+1)^{d-1}} \mathbf{f}_{\kappa_k}^{(q_{\kappa_k}+1)^{d-1}} \end{pmatrix}.$$

Here, $\mathbf{T}_{\kappa_k}^i$, $\mathbf{S}_{\kappa_m,\kappa_k}^{j,i} \in \mathbb{R}^{N\times N}$ and $\mathbf{f}_{\kappa_k}^i \in \mathbb{R}^N$ are matrix/vector representations of $\tilde{T}_{\kappa_k}^i(\cdot,\cdot)$, $\tilde{S}_{\kappa_m,\kappa_k}^{j,i}(\cdot,\cdot)$ and $\ell_{\kappa_k}^i(\cdot)$ respectively, and $\mathbf{u}_{\kappa_k}^{i,n}$ represents the coefficents in the expansion of $u_{\kappa_k}^{i,n}$ in the basis $\{\phi_a\}_{a=1}^N$; that is,

$$u_{\kappa_k}^{i,n} = \sum_{b=1}^{N} (\mathbf{u}_{\kappa_k}^{i,n})_b \phi_b.$$

By setting $v_{\kappa_{\mathbb{S}}} = \phi_a$ for $1 \le a \le N$, the matrix/vector terms are defined entry-wise by

$$(\mathbf{T}_{\kappa_k}^i)_{ab} = \tilde{T}_{\kappa_k}^i(\phi_b, \phi_a),$$
$$(\mathbf{S}_{\kappa_m,\kappa_k}^{j,i})_{ab} = \tilde{S}_{\kappa_m,\kappa_k}^{j,i}(\phi_b, \phi_a),$$
$$(\mathbf{f}_{\kappa_k}^i)_a = \tilde{\ell}_{\kappa_k}^i(\phi_a)$$

for $1 \le a, b \le N$. We remark that the matrix $\mathbf{T}_{\kappa_k}^i$ is actually the DGFEM matrix for a linear first-order transport equation with constant wind direction $\boldsymbol{\mu}_{\kappa_k}^i$ given by the i^{th} quadrature point on κ_k. Since $\mathbf{S}_{\kappa_m,\kappa_k}^{j,i}$ is a weighted spatial mass matrix, it is slightly

sparser that $\mathbf{T}^i_{\kappa_k}$. However, owing to its block structure, the matrix $\mathbf{T}_{\kappa_k}$ is much sparser than $\mathbf{S}_{\kappa_m,\kappa_k}$; this is due to the judicious choice of the angular basis functions.

3.4.3 Discussion

We have detailed how the poly-energetic DGFEM scheme (3.22) can be approximately replaced with a multigroup discrete ordinates scheme. The resulting method (3.47) (for poly-energetic problems) enjoys several benefits over a direct implementation of (3.22) as well as some drawbacks. Most notably, the poly-energetic scheme (3.47) can be easily incorporated into existing radiation transport codes. As was noted earlier, (3.47) is identical to the multigroup equations when an energetic basis of piecewise-constant functions is employed; for $p \geq 1$, one may interpret (3.47) as the multigroup equations with upscattering limited to each energetic element. Likewise, the mono-energetic scheme (3.53) can also be incorporated into existing radiation transport codes as it is identical to the discrete ordinates equations.

While the schemes (3.47) and (3.53) may be implemented as a multigroup discrete ordinates scheme, the resulting approximate solution u_h is defined over the whole space-angle-energy domain. Owing to the definition of the angular and energetic basis functions, u_h may be evaluated at angles and energies other than the quadrature points employed in (3.47) and (3.53).

The main drawback from this treatment of the scheme (3.22) is that we are no longer implementing the exact DGFEM equations. In addition to the discretisation error incurred in a finite element discretisation of the LBTE, we are also introducing an inconsistency error arising from the quadrature-based approximation of the angular and energetic integrals in (3.22). As such, the convergence results of Chapter 3.3 no longer hold exactly when the finite element approximation u_h is generated from the schemes (3.47) and (3.53).

However, one could argue that such an inconsistency error is practically unavoidable since most implementations of the original scheme (3.22) would use numerical quadrature to construct the resulting system of equations. Moreover, such a quadrature scheme is unlikely to exactly integrate the Jacobian of the mapping from the unit cube to the unit sphere (introduced in Chapter 3.2.2). From this perspective, we have simply selected angular and energetic basis functions that simplify the implementation of (3.22) when appropriate quadrature schemes are selected. Therefore, even though we have not proven a convergence result similar to that of Theorem 3.3.4 for the simplified scheme, one can still expect similar convergence rates if the additional "quadrature-inconsistency" term decays sufficiently fast as a function of the discretisation parameters.

3.5 Numerical Results

We now focus on some benchmark problems to assess the error incurred by the poly- and mono-energetic DGFEM schemes (3.22) and (3.29) respectively. In particular, the error $e_h = u - u_h$ between an exact solution u to these problems and their corresponding DGFEM approximation u_h will be computed for a range of different mesh sizes and polynomial degree of approximation. The error e_h will be measured both in the DGFEM-energy norm (3.38) and the L^2-norm, which is denoted by $||\cdot||_{L^2(\mathcal{D})}$ and defined for $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ by

$$||v_h||^2_{L^2(\mathcal{D})} = \int_{\mathbb{Y}} \int_{\mathbb{S}} \int_{\Omega} v_h^2 \ \mathrm{d}\mathbf{x}\, \mathrm{d}\boldsymbol{\mu}\, \mathrm{d}E$$

for poly-energetic problems, and for $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ by

$$||v_h||^2_{L^2(\mathcal{D})} = \int_{\mathbb{S}} \int_{\Omega} v_h^2 \ \mathrm{d}\mathbf{x}\, \mathrm{d}\boldsymbol{\mu}$$

for mono-energetic problems.

3.5.1 Poly-Energetic 2D

We first employ the space-angle-energy DGFEM scheme (3.22) to a model poly-energetic benchmark problem assuming the form of (3.4) in two spatial dimensions. We describe the discretisations and associated finite element spaces for the spatial, angular and energetic domains below:

- The spatial domain is taken to be the unit square $\Omega = (0,1)^2$, which we discretise using a family of (non-nested) polygonal meshes $\mathcal{T}_\Omega$ with $|\mathcal{T}_\Omega| \in \{16, 64, 256, 1024, 4096\}$ polygonal elements. On each spatial element $\kappa_\Omega \in \mathcal{T}_\Omega$ we define a basis of $\mathbb{P}^p(\kappa_\Omega)$, where $p \in \{0,1,2\}$ and is uniform across all elements in the spatial mesh. Figure 3.4 shows the polygonal meshes employed in the spatial domain.

- The angular domain is taken to be the unit circle $\mathbb{S} = S^1$, which we discretise using the aforementioned cube-sphere meshes $\mathcal{T}_\mathbb{S}$ with $|\mathcal{T}_\mathbb{S}| \in \{8, 16, 32, 64, 128\}$ curved one-dimensional elements. On each angular element $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$ we define a basis of $\mathbb{Q}^p(\kappa_\mathbb{S})$, where $p \in \{0,1,2\}$ and is uniform across all elements in the angular mesh. Figure 3.4 shows the cube-sphere meshes employed in the angular domain.

- The energetic domain is taken to be the energy interval $\mathbb{Y} = (500\text{keV}, 1000\text{keV})$, which we discretise with a one-dimensional uniform mesh $\mathcal{T}_\mathbb{Y}$ with $|\mathcal{T}_\mathbb{Y}| \in \{4, 8, 16, 32, 64\}$. On each energetic element $\kappa_\mathbb{Y} \in \mathcal{T}_\mathbb{Y}$ we define a basis of $\mathbb{P}^p(\kappa_\mathbb{Y})$, where $p \in \{0,1,2\}$ and is uniform across all elements in the energetic mesh.

The space-angle-energy domain $\mathcal{D}$ is discretised using the resulting space-angle-energy mesh $\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}$ with $|\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}| \in \{512, 8192, 131072, 2097152, 33554432\}$ elements. On each

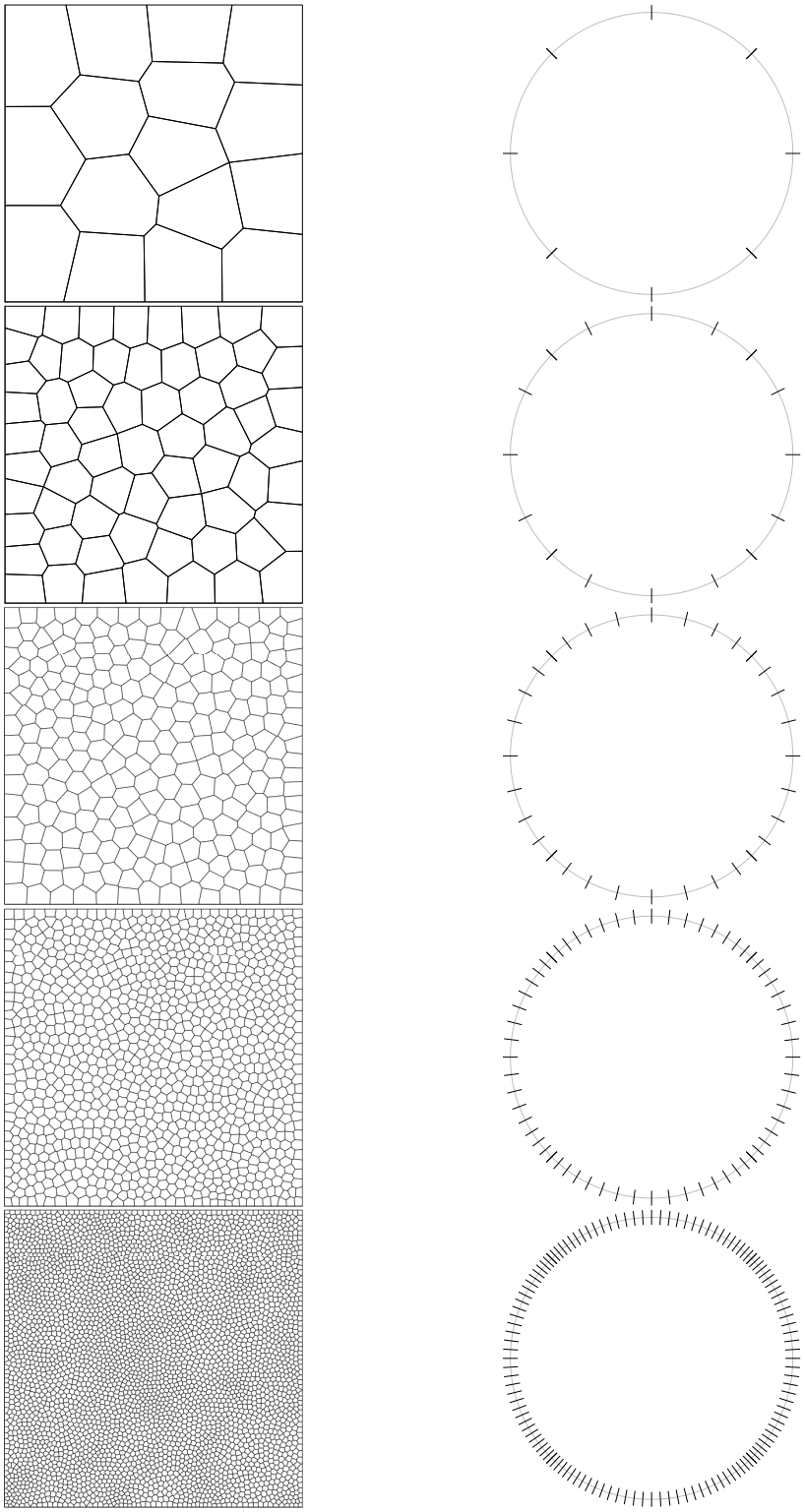

Figure 3.4: Plot of space-angle meshes employed in poly-energetic 2D problem. Each row corresponds to a space-angle mesh $\mathcal{T}_{\Omega,\mathbb{S}}$ with $|\mathcal{T}_{\Omega,\mathbb{S}}| \in \{128, 1024, 8192, 65536, 524288\}$ space-angle elements. Left column: polygonal spatial meshes of $(0,1)^2$. Right column: cube-sphere angular meshes of S^1.

space-angle-energy element $\kappa_\Omega \times \kappa_\mathbb{S} \times \kappa_\mathbb{Y} \in \mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}$ we define a tensor-product basis $\mathbb{P}^p(\kappa_\Omega) \times \mathbb{Q}^p(\kappa_\mathbb{S}) \times \mathbb{P}^p(\kappa_\mathbb{Y})$ from the constituent bases described above, where $p \in \{0, 1, 2\}$ and is uniform across all elements in the space-angle-energy mesh.

The data terms α and θ are chosen to mimic the Compton scattering of photons travelling through a slab of water. The macroscopic absorption cross-section is selected to be $\alpha = 0$ over $\mathcal{D}$, and the differential scattering cross-section θ is selected to be as in [50]:

$$\theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E' \to E) = \rho(\mathbf{x}) \theta_{KN}(\boldsymbol{\mu} \cdot \boldsymbol{\mu}', E' \to E) \delta\left(F(E', E, \boldsymbol{\mu} \cdot \boldsymbol{\mu}')\right),$$

where

- $\rho = \rho(\mathbf{x})$ denotes the electron density of water and is approximately equal to $\rho(\mathbf{x}) \approx 3.34281 \times 10^{29} \mathrm{m}^{-3}$. This is computed as in [36, 85] using atomic weight data in [55]. Denoting by n_H (resp. n_O) the number of atoms per unit volume of hydrogen (resp. oxygen) atoms, we have

$$n_H = \frac{\mathcal{N} \rho_{H2O} w_H}{A_H},$$
$$n_O = \frac{\mathcal{N} \rho_{H2O} w_O}{A_O},$$

  where $\mathcal{N} \approx 6.022045 \times 10^{23} \mathrm{mol}^{-1}$ denotes Avogadro's constant, $\rho_{H2O} \approx 997 \mathrm{kg/m}^3$ denotes the density of water, $A_H \approx 1.0079$ (resp. $A_O \approx 15.9994$) denotes the standard atomic weight[2] of hydrogen (resp. oxygen), and $w_H = \frac{2A_H}{2A_H + A_O}$ (resp. $w_H = \frac{A_O}{2A_H + A_O}$) denotes the proportion (by mass) of hydrogen (resp. oxygen) in water. Thus, the electron density of water can be computed as

$$\rho = Z_H n_H + Z_O n_O,$$

  where $Z_H = 1$ (resp. $Z_O = 8$) denotes the atomic number of hydrogen (resp. oxygen).

- $\theta_{KN}(\cos\varphi, E' \to E)$ denotes the Klein-Nishina differential scattering cross-section [64]

$$\theta_{KN}(\cos\varphi, E' \to E) = \frac{1}{2} r_e^2 \left(\frac{E}{E'}\right)^2 \left(\frac{E}{E'} + \frac{E'}{E} - \sin^2\varphi\right),$$

  where $r_e \approx 2.81794 \times 10^{-15} \mathrm{m}$ denotes the classical electron radius [50].

- δ denotes the Dirac delta distribution and $F(E', E, \cos\varphi) = 0$ enforces the following kinematic constraint between the incoming and recoiling photon energies and the deflection cosine:

$$E = \frac{E'}{1 + \frac{E'}{511\mathrm{keV}}(1 - \cos\varphi)},$$

[2] Strictly, atomic masses should be used in place of standard atomic weights, in which case $A_H = 1$ and $A_O = 16$.

where E and E' are both measured in units of keV. Hence, we define F by

$$F(E', E, \cos\varphi) = E - \frac{E'}{1 + \frac{E'}{511\text{keV}}(1 - \cos\varphi)}.$$

The data terms β and γ are defined using (3.6) and (3.7) respectively, where the angular integrals are taken over S^1. Finally, the forcing terms f and g are selected so that the analytical solution to (3.4) is given by

$$u(\mathbf{x}, \boldsymbol{\mu}, E) = \exp\left(-\left(\frac{E\boldsymbol{\mu}\cdot\mathbf{x}}{E_{max}}\right)^2\right)\phi\left(\frac{E}{E_{max}}\right),$$

where $E_{max} = 1000\text{keV}$ and $\phi(x) = \exp\left(-\frac{1}{1-x^2}\right)$ denotes a mollifier ensuring that u is compactly supported in energy.

It was shown in Chapter 3.2.4 that, in the three-dimensional setting, the quantities β and γ (defined in (3.6) and (3.7) respectively) satisfy $\beta - \gamma \geq 0$ over a restricted energy domain; moreover, the difference $\beta - \gamma$ is likely to be bounded away from zero over any finite interval not containing $E = 0$. This implies that the condition (3.36) is satisfied even when $\alpha = 0$.

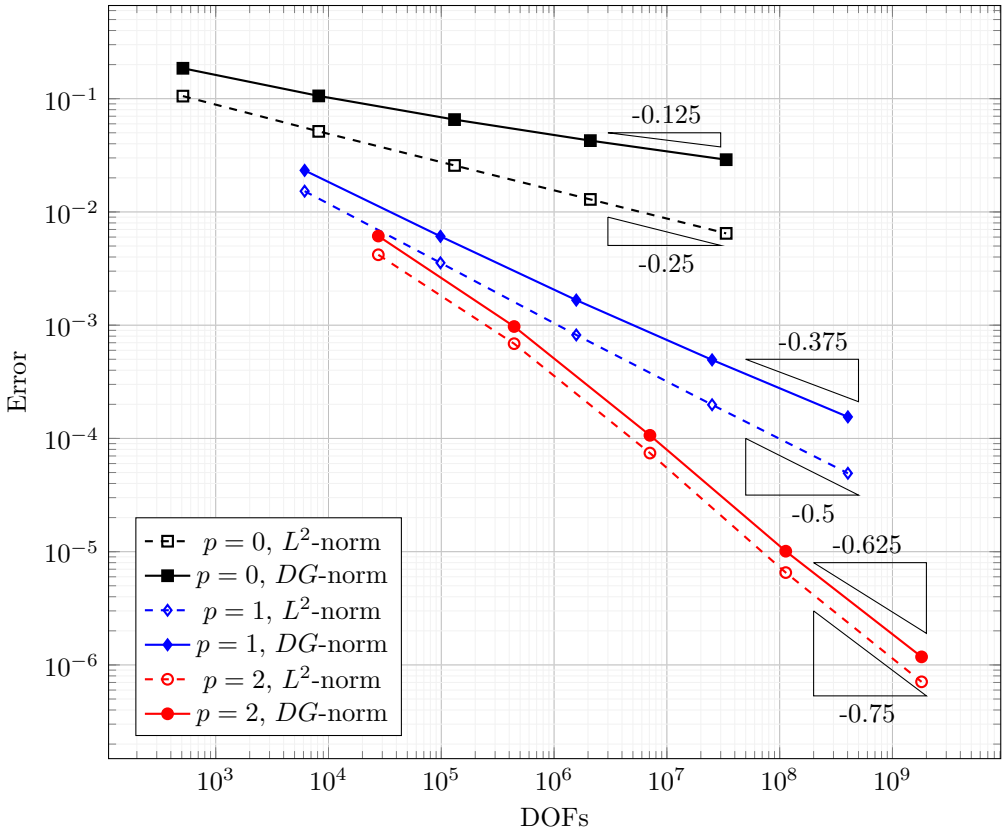

Figure 3.5: Convergence of full DGFEM discretisation of poly-energetic benchmark problem. Gradients represent optimal convergence rate of L^2- and DGFEM-energy-norm errors.

Figure 3.5 shows the convergence of the poly-energetic DGFEM scheme in the L^2- and DGFEM-energy-norms. For all polynomial degrees reported, the rate of convergence

of the fluence error measured in the L^2-norm is $O(N^{\frac{p+1}{k}})$, where N denotes the number of degrees of freedom and $k = 4$ denotes the total dimension of the problem domain $\mathcal{D}$ (i.e. the sum of two independent spatial dimensions, one angular dimension and one energetic dimension). Equivalently, the L^2-norm fluence error is $O\left(h^{p+1}\right)$, where h denotes the space-angle-energy mesh size parameter under uniform refinement. This convergence rate matches the optimal L^2-norm convergence rate expected when the DGFEM scheme is applied to the transport equation (3.10) [28, 78], though we note that this result is not guaranteed on general meshes.

For both $p = 0$ and $p = 1$, the DGFEM-energy-norm rate of convergence of the fluence error is $O(N^{\frac{p+1/2}{k}})$, or equivalently $O(h^{p+1/2})$. This convergence rate is sub-optimal by a factor of $h^{\frac{1}{2}}$, which matches the expected rate when the DGFEM scheme is applied to the transport equation (3.10) [28, 78]. This result also agrees with Theorem 3.3.4. The set of results for $p = 2$ suggest that the convergence behaviour of the scheme is pre-asymptotic. However, we stress that the finite element meshes employed in this benchmark are still coarse, despite the large number of degrees of freedom employed.

3.5.2 Mono-Energetic 3D

We now employ the space-angle DGFEM scheme to a model mono-energetic benchmark problem assuming the form of (3.9) in three spatial dimensions. We describe the discretisations and associated finite element spaces for the spatial and angular domains below:

- The spatial domain is taken to be the unit cube $\Omega = (0, 1)^3$, which we discretise using a family of regular cube meshes $\mathcal{T}_\Omega$ with $|\mathcal{T}_\Omega| \in \{8, 64, 512, 4096, 32768\}$ cubic elements. On each spatial element $\kappa_\Omega \in \mathcal{T}_\Omega$ we define a basis of $\mathbb{P}^p(\kappa_\Omega)$, where $p \in \{0, 1, 2\}$ and is uniform across all elements in the mesh.

- The angular domain is taken to be the unit sphere $\mathbb{S} = S^2$, which we discretise using the aforementioned cube-sphere meshes $\mathcal{T}_\mathbb{S}$ with $|\mathcal{T}_\mathbb{S}| \in \{6, 24, 96, 384, 1536\}$ curved quadrilateral elements. On each angular element $\kappa_\mathbb{S} \in \mathcal{T}_\mathbb{S}$ we define a basis of $\mathbb{Q}^p(\kappa_\mathbb{S})$, where $p = \{0, 1, 2\}$ and is uniform across all elements in the angular mesh.

The space-angle domain $\mathcal{D}$ is discretised using the resulting space-angle mesh $\mathcal{T}_{\Omega,\mathbb{S}}$ with $|\mathcal{T}_{\Omega,\mathbb{S}}| \in \{48, 1536, 49152, 1572864, 50331648\}$ elements. On each space-angle element $\kappa_\Omega \times \kappa_\mathbb{S} \in \mathcal{T}_{\Omega,\mathbb{S}}$ we define a tensor-product basis $\mathbb{P}^p(\kappa_\Omega) \times \mathbb{Q}^p(\kappa_\mathbb{S})$ from the constituent bases described above, where $p \in \{0, 1, 2\}$ and is uniform across all elements in the space-angle mesh.

The macroscopic absorption cross-section is selected to be $\alpha = 1$ over $\mathcal{D}$, and the

differential scattering cross-section θ is selected to be

$$\theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') = \frac{1}{|\mathbb{S}|} = \frac{1}{4\pi},$$

so that $\beta = 1$. Finally, the forcing terms f and g are selected so that the analytical solution is given by

$$u(\mathbf{x}, \boldsymbol{\mu}) = \cos(4\phi)(x \cos y + y \sin x),$$

where the angular variable is parameterised by $\boldsymbol{\mu} = (\sin\phi\cos\varphi, \sin\phi\sin\varphi, \cos\phi)$ for $0 \leq \phi \leq \pi$ and $0 \leq \varphi \leq 2\pi$.

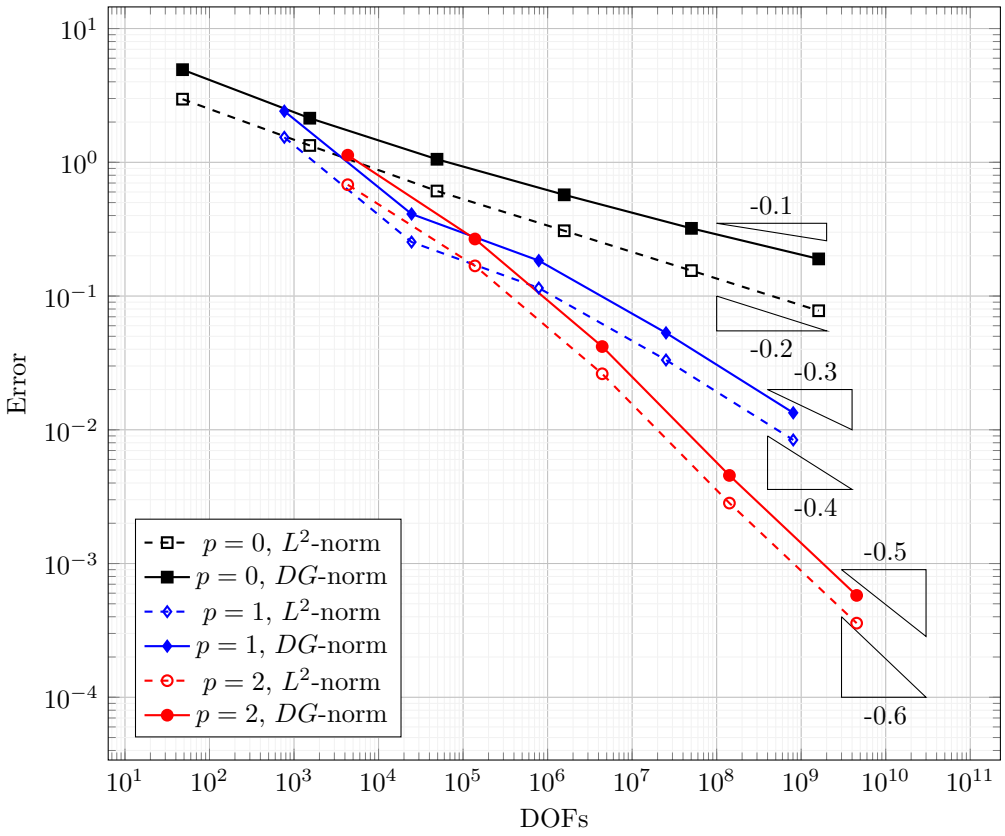

Figure 3.6: Convergence of full DGFEM discretisation of 3D mono-energetic benchmark problem. Gradients represent optimal convergence rate of L^2- and DGFEM-energy-norm errors. An additional space-angle mesh was employed for $p = 0$ with $|\mathcal{T}_\Omega| = 262144$, $|\mathcal{T}_\mathbb{S}| = 6144$ and $|\mathcal{T}_{\Omega,\mathbb{S}}| = 1610612736$.

Figure 3.6 shows the convergence of the mono-energetic DGFEM scheme in the L^2- and DGFEM-energy-norms. As in the previous benchmark, the rate of convergence of the fluence error measured in the L^2-norm is $O(N^{\frac{p+1}{k}})$, or equivalenty $O(h^{p+1})$. Again, N denotes the number of degrees of freedom and $k = 5$ denotes the total dimension of $\mathcal{D}$ (i.e. three independent spatial dimensions and two independent angular dimensions). The rate of convergence of the fluence error measured in the DGFEM-energy-norm is $O(N^{\frac{p+1/2}{k}})$, or equivalently $O(h^{p+\frac{1}{2}})$, only for the set of results for $p = 0$; we attribute the results for $p = 1$ and $p = 2$ to the numerical scheme being in the pre-asymptotic regime.

We conclude that the mono-energetic DGFEM scheme exhibits optimal convergence in the DGFEM-energy- and L^2-norms as in [28, 78], although the latter result is not guaranteed on general meshes. As before, we stress that the finite element meshes employed in this benchmark are still coarse, despite the large number of degrees of freedom employed.

Chapter 4

Quadrature-Free Implementation of the Discontinuous Galerkin Method for Transport Problems

In Chapter 3, we derived a discontinuous Galerkin finite element method (DGFEM) for the first-order linear transport equation. In practice, the DGFEM problem is converted to a linear algebra problem of the form

$$\mathbf{A}\mathbf{u} = \mathbf{f},$$

where $\mathbf{A} \in \mathbb{R}^{N \times N}$ denotes the DGFEM transport matrix, $\mathbf{f} \in \mathbb{R}^N$ denotes the forcing vector and $N = \dim \mathcal{V}_\Omega$ denotes the dimension of the (spatial) finite element space. These matrix/vector quantities are often assembled by applying numerical integration techniques to the corresponding bilinear form or linear functional. Such techniques typically employ quadrature schemes on elements and faces of the mesh $\mathcal{T}_\Omega$.

When $\mathcal{T}_\Omega$ consists of standard (simplicial or tensor-product) elements, one may construct mappings between each element $\kappa \in \mathcal{T}_\Omega$ and a *reference element* $\hat{\kappa}$. Such mappings can be exploited to prescribe quadrature schemes and basis functions defined on each element κ in terms of a single quadrature scheme and a set of basis functions defined on $\hat{\kappa}$.

Recently, there has been growing interest in employing meshes $\mathcal{T}_\Omega$ consisting of non-standard (polytopic) elements [27, 28]. The question arises as to how numerical integration over the elements of such meshes should be performed. Since elements of $\mathcal{T}_\Omega$ may

have different numbers of boundary faces, it is not clear how elements may be mapped to a single reference element. In view of performing numerical integration on polytopic elements, one possibility is to inherit the quadrature schemes from a subtesselation of the element into simplices. The number of quadrature points and weights in this scheme for the polytopic element can be reduced via optimisation algorithms [73].

This chapter will discuss an alternative approach to assembling the DGFEM matrix for transport problems via homogeneous function integration [6, 33, 69]. We will first review some important properties of homogeneous functions and derive expressions for their integrals over polytopes in terms of their integrals over boundary faces. We will compare the resulting quadrature-free algorithm for homogeneous function integration against a standard quadrature-based algorithm.

We will then discuss how homogeneous function integration can be used to assemble matrices arising from a DGFEM discretisation of the first-order linear transport equation. We will rewrite the weak formulation by decomposing the (polynomial) integrands into a linear combination of homogeneous functions. The resulting quadrature-free assembly algorithm will be benchmarked against a standard quadrature-based implementation. A floating-point operation analysis of both methods will be performed, highlighting the advantages and disadvantages of quadrature-free methods for the assembly of more general DGFEM matrices.

4.1 Overview of Quadrature-Free Integration

We first give an overview of the principle techniques we shall employ in the quadrature-free assembly procedure, cf. [6, 33, 69]. The central idea is to consider the class of *homogeneous functions*: a function $f : \mathbb{R}^d \setminus \{\mathbf{0}\} \to \mathbb{R}$ is said to be *(positively) homogeneous of degree k* if we have

$$f(\alpha \mathbf{x}) = \alpha^k f(\mathbf{x})$$

for all $\alpha > 0$ and $\mathbf{x} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$. If $k > 0$, then this extends to functions $f : \mathbb{R}^d \to \mathbb{R}$. The following theorem about homogeneous functions is useful for quadrature-free integration of polynomial functions:

Theorem 4.1.1 (Euler's homogeneous function theorem, [88])**.** *Let $f : \mathbb{R}^d \setminus \{\mathbf{0}\} \to \mathbb{R}$ be continuously differentiable. Then f is positively homogeneous of degree k if and only if*

$$\mathbf{x} \cdot \nabla f(\mathbf{x}) = k f(\mathbf{x})$$

for all $\mathbf{x} \in \mathbb{R} \setminus \{\mathbf{0}\}$.

Remark. *An example of a homogeneous function $f : \mathbb{R}^d \to \mathbb{R}$ is given by $f(\mathbf{x}) = \mathbf{x}^{\boldsymbol{\alpha}} = \prod_{i=1}^d x_i^{\alpha_i}$, where $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^d \in \mathbb{N}_0^d$ is a multi-index of length d. To see that f is*

homogeneous, note that

$$\begin{aligned}\mathbf{x}\cdot\nabla f(\mathbf{x}) &= \sum_{j=1}^{d} x_j \frac{\partial}{\partial x_j}\prod_{i=1}^{d} x_i^{\alpha_i} \\ &= \sum_{j=1}^{d} x_j\cdot\alpha_j x_j^{\alpha_j-1}\prod_{\substack{i=1\\ i\neq j}}^{d} x_i^{\alpha_i} \\ &= \sum_{j=1}^{d}\alpha_j\prod_{i=1}^{d} x_i^{\alpha_i} \\ &= \sum_{j=1}^{d}\alpha_j f(\mathbf{x}).\end{aligned}$$

Thus, $f(\mathbf{x})$ is a positively homogeneous function of degree $k = |\boldsymbol{\alpha}| = \sum_{i=1}^{d}\alpha_i$.

To see how Euler's homogeneous function theorem can be applied to the implementation of finite element methods, we remark that often one seeks a piecewise-polynomial approximation of the PDE variable(s). On selecting a basis of piecewise-polynomial functions, the entries of the discontinuous Galerkin finite element matrix frequently involve integrals of polynomial functions over the elements and faces of the mesh. Such integrands may be decomposed as a linear combination of monomial functions, which are also homogeneous.

Let $f:\mathbb{R}^d\to\mathbb{R}$ be a positively homogeneous function of degree $k\geq 0$ and $\mathcal{P}\subset\mathbb{R}^d$ a d-dimensional polytope with boundary $\partial\mathcal{P} = \bigcup_{i=1}^{m}\mathcal{F}_i$, where each $\mathcal{F}_i$ is a $(d-1)$-dimensional planar polytopic boundary face on which $\mathbf{x}\cdot\mathbf{n}_i = a_i$. Here, $a_i\in\mathbb{R}$ and $\mathbf{n}_i$ denotes the outward unit normal to $\mathcal{P}$ on $\mathcal{F}_i$. We seek to find an expression for the volume integral

$$\mathcal{I} = \int_{\mathcal{P}} f(\mathbf{x})\ \mathrm{d}\mathbf{x}.$$

Rather than manipulating the integral $\mathcal{I}$ directly, we will instead express the integral

$$\mathcal{J} = \int_{\mathcal{P}} \nabla\cdot(\mathbf{x}f(\mathbf{x}))\ \mathrm{d}\mathbf{x}$$

in two different ways. The first way is to invoke the divergence theorem:

$$\begin{aligned}\mathcal{J} &= \int_{\partial\mathcal{P}} f(\mathbf{x})\mathbf{x}\cdot\mathbf{n}\ \mathrm{d}s \\ &= \sum_{i=1}^{m}\int_{\mathcal{F}_i} f(\mathbf{x})\mathbf{x}\cdot\mathbf{n}_i\ \mathrm{d}s \\ &= \sum_{i=1}^{m} a_i\int_{\mathcal{F}_i} f(\mathbf{x})\ \mathrm{d}s.\end{aligned}$$

The second way is to expand the divergence term and invoke Euler's homogeneous function theorem:

$$\begin{aligned}\mathcal{J} &= \int_{\mathcal{P}} (\nabla\cdot\mathbf{x})f(\mathbf{x}) + \mathbf{x}\cdot\nabla f(\mathbf{x})\ \mathrm{d}\mathbf{x} \\ &= (d+k)\int_{\mathcal{P}} f(\mathbf{x})\ \mathrm{d}\mathbf{x}.\end{aligned}$$

On rearranging the two expressions for $\mathcal{J}$, we obtain the following result:

$$\int_{\mathcal{P}} f(\mathbf{x}) \ \mathrm{d}\mathbf{x} = \frac{1}{d+k} \sum_{i=1}^{m} a_i \int_{\mathcal{F}_i} f(\mathbf{x}) \ \mathrm{d}s. \tag{4.1}$$

It can be seen that, whenever f is a positively homogeneous function, we can always write its volume integral over a polytopic domain as a linear combination of surface integrals over its boundary faces. However, one can use the same procedure to rewrite these surface integrals in terms of line integrals over the boundary edges of the face [6]. For a given $(d-1)$-dimensional face $\mathcal{F}_i$, let $\partial\mathcal{F}_i = \{\mathcal{E}_{ij}\}_{j=1}^{n}$ denote the set of $(d-2)$-dimensional (planar) boundary edges of $\mathcal{F}_i$. For any homogeneous function f of degree $k \geq 0$, we have

$$\int_{\mathcal{F}_i} f(\mathbf{x}) \ \mathrm{d}s = \frac{1}{d+k-1} \left(\sum_{j=1}^{n} d_{ij} \int_{\mathcal{E}_{ij}} f(\mathbf{x}) \ \mathrm{d}\nu + \int_{\mathcal{F}_i} \mathbf{x}_{i,0} \cdot \nabla f(\mathbf{x}) \ \mathrm{d}s \right). \tag{4.2}$$

Here, $\mathbf{x}_{i,0}$ is an arbitrary point lying in the same hyperplane $\mathcal{H}_i$ as $\mathcal{F}_i$, d_{ij} is the Euclidean distance between $\mathbf{x}_{i,0}$ and $\mathcal{E}_{ij}$, and $\mathrm{d}\nu$ denotes the $(d-2)$-dimensional surface measure on the boundary edges of $\mathcal{F}_i$. Figure 4.1 highlights the key geometric quantities in the case of a two-dimensional polytope.

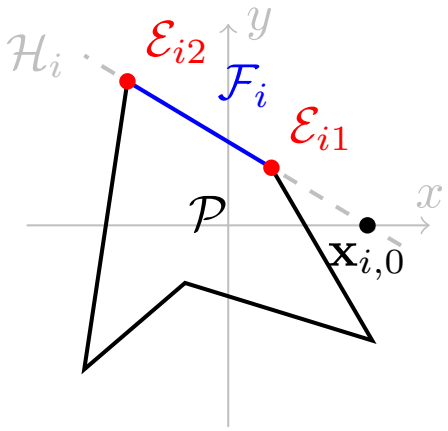

Figure 4.1: An example d-dimensional polytope $\mathcal{P}$ (black) for $d = 2$. For the highlighted $(d-1)$-dimensional boundary face $\mathcal{F}_i$ (blue), its $(d-2)$-dimensional boundary edges $\mathcal{E}_{ij}$ (red) are marked. Also shown are he hyperplane $\mathcal{H}_i$ (grey) containing $\mathcal{F}_i$ and a choice of $\mathbf{x}_{i,0} \in \mathcal{H}_i$.

Remark. *In the aforementioned case when $f(\mathbf{x}) = \mathbf{x}^{\boldsymbol{\alpha}}$, the relationships (4.1) and (4.2) respectively become*

$$\int_{\mathcal{P}} \mathbf{x}^{\boldsymbol{\alpha}} \ \mathrm{d}\mathbf{x} = \frac{1}{|\boldsymbol{\alpha}|+d} \sum_{i=1}^{m} a_i \int_{\mathcal{F}_i} \mathbf{x}^{\boldsymbol{\alpha}} \ \mathrm{d}s, \tag{4.3}$$

$$\int_{\mathcal{F}_i} \mathbf{x}^{\boldsymbol{\alpha}} \ \mathrm{d}s = \frac{1}{|\boldsymbol{\alpha}|+d-1} \left(\sum_{j=1}^{n} d_{ij} \int_{\mathcal{E}_{ij}} \mathbf{x}^{\boldsymbol{\alpha}} \ \mathrm{d}\nu + \sum_{k=1}^{d} \mathbf{x}_{i,0}^{(k)} \alpha_k \int_{\mathcal{F}_i} \mathbf{x}^{\boldsymbol{\alpha}-\mathbf{e}_k} \ \mathrm{d}s \right), \tag{4.4}$$

where $\mathbf{e}_k \in \mathbb{N}_0^d$ *is defined by* $(\mathbf{e}_k)_i = \delta_{ik}$*. In particular, the formula for* $\int_{\mathcal{F}_i} \mathbf{x}^{\boldsymbol{\alpha}}\ \mathrm{d}s$ *is recursive; this recursion ends when* $\mathbf{x}_{i,0}^{(k)}\alpha_k = 0$ *for all* $1 \leq k \leq d$*. By careful selection of the point* $\mathbf{x}_{i,0}$*, the authors of [6] describe an algorithm to evaluate this integral for a single multi-index* $\boldsymbol{\alpha}$ *using as few recursive function calls as possible.*

One may recursively apply this "dimension-reducing" procedure to single-point evaluations of the function f (and its gradient) at the vertices of the polytope $\mathcal{P}$ [6, 34]. For the example illustrated in the remark above, it can be shown that integrals of monomials of the form $\mathbf{x}^{\boldsymbol{\alpha}}$ over any face $\mathcal{F}_i \subset \partial\mathcal{P}$ can be reduced to sums of integrals of the same integrand over the boundary facets of $\mathcal{F}_i$ plus a sum of integrals of lower-degree monomials of the form $\mathbf{x}^{\boldsymbol{\alpha}-\mathbf{e}_k}$ over $\mathcal{F}_i$.

Equation (4.4) holds more generally when $\mathcal{F}_i$ denotes a general k-dimensional polytopic facet, $1 \leq k \leq d-1$, embedded in $\mathbb{R}^d$. For instance, (4.4) is true when $\mathcal{F}_i$ denotes a 1-dimensional line segment in $\mathbb{R}^3$ - here, $\{\mathcal{E}_{ij}\}_{j=1}^2$ denote the end-points of $\mathcal{F}_i$ and d_{ij} denotes the distance between $\mathcal{E}_{ij}$ and $\mathbf{x}_{i,0} \in \mathbb{R}^3$, where $\mathbf{x}_{i,0}$ is chosen along the line containing $\mathcal{F}_i$.

This idea can be exploited to compute families of integrals of monomials of the form

$$\mathcal{S}_{p,\mathbf{p}} = \left\{ \int_{\mathcal{P}} \mathbf{x}^{\boldsymbol{\alpha}}\ \mathrm{d}\mathbf{x} : 0 \leq |\boldsymbol{\alpha}| \leq p,\ \mathbf{0} \leq \boldsymbol{\alpha} \leq \mathbf{p} \right\} \tag{4.5}$$

where p denotes the maximum total polynomial degree of any moment in $\mathcal{S}_{p,\mathbf{p}}$, $\mathbf{p} = (p_i)_{i=1}^d \in \mathbb{N}_0^d$ denotes the maximum component-wise polynomial degree of any moment in $\mathcal{S}_{p,\mathbf{p}}$, $\mathbf{0} = (0, 0, \ldots, 0) \in \mathbb{N}_0^d$ and the notation $\boldsymbol{\alpha} \leq \mathbf{p}$ means $\alpha_i \leq p_i$ for $1 \leq i \leq d$. A recursive algorithm was developed in [6] to evaluate the elements of $\mathcal{S}_{p,\mathbf{p}}$ for given p and $\mathbf{p}$, and is reproduced in Algorithm 1.

We may also consider the more general case where the function f assumes the form $f(x) = \sum_{j=1}^n f_j(x)$ with each f_j a homogeneous function of degree k_j. For example, (4.1) may be written as

$$\begin{aligned}
\int_{\mathcal{P}} f(\mathbf{x})\ \mathrm{d}\mathbf{x} &= \sum_{j=1}^{n} \int_{\mathcal{P}} f_j(\mathbf{x})\ \mathrm{d}\mathbf{x} \\
&= \sum_{j=1}^{n} \frac{1}{d+k_j} \sum_{i=1}^{m} a_i \int_{\mathcal{F}_i} f_j(\mathbf{x})\ \mathrm{d}s \\
&= \sum_{i=1}^{m} a_i \sum_{j=1}^{n} \frac{1}{d+k_j} \int_{\mathcal{F}_i} f_j(\mathbf{x})\ \mathrm{d}s.
\end{aligned}$$

Note that we may still write the integral of f over $\mathcal{P}$ as a sum of contributions from each face $\mathcal{F}_i$.

Remark. *For example, let* $I \subset \mathbb{N}_0^d$ *be a finite subset of multi-indices, and define* f *by* $f(x) = \sum_{\boldsymbol{\alpha}\in I} c_{\boldsymbol{\alpha}} \mathbf{x}^{\boldsymbol{\alpha}}$ *where each* $c_{\boldsymbol{\alpha}} \in \mathbb{R}$*. We have*

$$\int_{\mathcal{P}} f(\mathbf{x})\ \mathrm{d}\mathbf{x} = \sum_{i=1}^{m} a_i \sum_{\boldsymbol{\alpha}\in I} \frac{c_{\boldsymbol{\alpha}}}{d+|\boldsymbol{\alpha}|} \int_{\mathcal{F}_i} \mathbf{x}^{\boldsymbol{\alpha}}\ \mathrm{d}s.$$

Algorithm 1 Integration of all monomial functions of maximal total degree p and component-wise degree $\mathbf{p} = (p_i)_{i=1}^d$ on a d-dimensional polytope $\mathcal{P}$.

Get polytope boundary $\partial\mathcal{P} = \{\mathcal{P}_i\}_{i=1}^m$, where $\mathcal{P}_i \subset \partial\mathcal{P}$

$F = \texttt{FaceIntegrals}(d-1, \mathcal{P}_1, \ldots, \mathcal{P}_m)$ ▷ *Get integrals* $\int_{\mathcal{F}_i} \mathbf{x}^{\boldsymbol{\alpha}} \, \mathrm{d}\mathbf{x}$

▷ *Compute* $\int_{\mathcal{P}} \mathbf{x}^{\boldsymbol{\alpha}} \, \mathrm{d}\mathbf{x}$ *using (4.3)*

for $k_1 = 0, \ldots, \min\{p, p_1\}$, $k_2 = 0, \ldots, \min\{p - k_1, p_2\}$, $\ldots$, $k_d = 0, \ldots, \min\{p - \sum_{n=1}^{d-1} k_n, p_d\}$ **do**

  $V(k_1, \ldots, k_d) = \frac{1}{d + \sum_{n=1}^d k_n} \sum_{i=1}^m a_i F(k_1, \ldots, k_d, i)$

end for

procedure $F = \texttt{FaceIntegrals}(N, \mathcal{E}_1, \ldots, \mathcal{E}_r)$

  $F(-1:p, \ldots, -1:p, 1:r) = 0$

  for $i = 1, \ldots, r$ **do**

    Choose $\mathbf{x}_{i,0}$ as the first vertex of $\mathcal{E}_i$

    Get face boundary $\partial\mathcal{E}_i = \{\mathcal{E}_{ij}\}_{j=1}^{m_i}$, where $\mathcal{E}_{ij} \subset \partial\mathcal{E}_i$

    Compute the distance d_{ij} between $\mathbf{x}_{0,i}$ and (the hyperplane containing) $\mathcal{E}_{ij}$

    if $N > 1$ **then**

      $E = \texttt{FaceIntegrals}(N-1, \mathcal{E}_{i1}, \ldots, \mathcal{E}_{im_i})$ ▷ *Get integrals* $\int_{\mathcal{E}_{ij}} \mathbf{x}^{\boldsymbol{\alpha}} \, \mathrm{d}\nu$

    else if $N = 1$ **then** ▷ *Here,* $\mathcal{E}_{ij} = v \in \mathbb{R}^d$ *is a point*

      $E(k_1, \ldots, k_d, j) = v_1^{k_1} \ldots v_d^{k_d}$ for each $0 \le k_n \le p_n$, $1 \le n \le d$ and $1 \le j \le m_i$

    end if

  ▷ *Compute* $\int_{\mathcal{E}_i} \mathbf{x}^{\boldsymbol{\alpha}} \, \mathrm{d}\mathbf{x}$ *using (4.4)*

  for $k_1 = 0, \ldots, \min\{p, p_1\}$, $k_2 = 0, \ldots, \min\{p - k_1, p_2\}$, $\ldots$, $k_d = 0, \ldots, \min\{p - \sum_{n=1}^{d-1} k_n, p_d\}$ **do**

$$F(k_1, \ldots, k_d, i) = \frac{1}{N + \sum_{n=1}^d k_n} \left(\sum_{j=1}^{m_i} d_{ij} E(k_1, \ldots, k_d, j) + \sum_{n=1}^d (\mathbf{x}_{i,0})_n k_n F(k_1, \ldots, k_{n-1}, k_n - 1, k_{n+1}, \ldots, k_d, i) \right)$$

  end for

end for

end procedure

4.1.1 Numerical example

To see how Algorithm 1 can outperform classical numerical quadrature techniques in the evaluation of the set $\mathcal{S}_{p,\mathbf{p}}$ defined in (4.5), we apply both methods to the problem of computing the family of integrals $\mathcal{S}_p$ given by

$$\mathcal{S}_p = \left\{ \int_{\mathcal{P}_n} \mathbf{x}^{\boldsymbol{\alpha}} \, \mathrm{d}\mathbf{x} : 0 \le |\boldsymbol{\alpha}| \le p \right\},$$

where $\mathcal{P}_n \subset \mathbb{R}^2$ denotes the n-gon with vertices $\left\{\mathbf{v}_k^{(n)} = \left(\cos\frac{2\pi k}{n}, \sin\frac{2\pi k}{n}\right)\right\}_{k=1}^{n}$ for $5 \leq n \leq 16$ and $p \in \{2, 4, 8, 16, 32\}$.

For the quadrature-based method, a quadrature scheme $Q_n = \{(\mathbf{x}_i, \omega_i)\}_{i=1}^{N} \subset \mathbb{R}^2 \times \mathbb{R}_{\geq 0}$ on $\mathcal{P}_n$ is defined by first forming a subtriangulation consisting of $n-2$ triangular elements by joining $\mathbf{v}_1^{(n)}$ to every other vertex in $\mathcal{P}_n$. On each element, a $(q+1)^2$-point quadrature scheme, $q = \lceil \frac{p+1}{2} \rceil$, is defined by constructing a reference quadrature scheme on the unit square $(-1, 1)^2$ exactly integrating all bivariate polynomials of degree $p+1$ and then mapping the reference quadrature scheme to the triangular element via a Duffy transformation [39]. The resulting quadrature scheme on $\mathcal{P}_n$ thus has $(n-2)(q+1)^2$ points. The set $\mathcal{S}_p$ is computed entry-by-entry by forming the weighted sum

$$\int_{\mathcal{P}_p} \mathbf{x}^{\boldsymbol{\alpha}} \ \mathrm{d}\mathbf{x} = \sum_{i=1}^{N} \omega_i (\mathbf{x}_i)_1^{\alpha_1} (\mathbf{x}_i)_2^{\alpha_2}$$

for each $0 \leq |\boldsymbol{\alpha}| \leq p$. We record the times taken by the quadrature-based algorithm to evaluate $\mathcal{S}_p$ in this manner - the time taken to generate the quadrature scheme on $\mathcal{P}_n$ is not included in this timing.

For the quadrature-free-based method, Algorithm 1 was specialised to the two-dimensional setting. Two arrays $\mathbf{V}, \mathbf{F} \in \mathbb{R}^{(p+1)\times(p+1)}$ are used to store the monomial integrals $(\mathbf{V})_{ij} = \int_{\mathcal{P}_n} x^i y^j \ \mathrm{d}\mathbf{x}$ and $(\mathbf{F})_{ij} = \int_{\partial\mathcal{P}_n^{(k)}} x^i y^j \ \mathrm{d}\mathbf{x}$, where $\partial\mathcal{P}_n^{(k)} \subset \partial\mathcal{P}_n$ and $1 \leq k \leq n$. Algorithm 1 is implemented in such a way that $\mathbf{F}$ can be re-initialised at each face, sparing the need to keep n copies of the matrix.

The CPU times taken for both methods to evaluate $\mathcal{S}_p$ are given in Figures 4.2 and 4.3, averaged over 100 calls to both integration procedures. The quadrature-based evaluation of $\mathcal{S}_p$ has computational complexity $O(np^4)$. To see this, note that $|\mathcal{S}_p| = \frac{1}{2}(p+1)(p+2) = O(p^2)$ and that, for each element $s_{ij} = \int_{\mathcal{P}_n} x^i y^j \ \mathrm{d}\mathbf{x} \in \mathcal{S}_p$, a quadrature scheme employing $(n-2)(q+1)^2 = O(np^2)$ points/weights is used to evaluate the integral. On the other hand, the quadrature-free-based evaluation of $\mathcal{S}_p$ has computational complexity $O(np^2)$. This is given in [6], but can be argued as follows. For each of the n faces of $\mathcal{P}_n$, Algorithm 1 loops over each of the $\frac{1}{2}(p+1)(p+2) = O(p^2)$ elements of $\mathcal{S}_p$ and performs an $O(1)$ floating-point operation, owing to the recurrence (4.2).

In the context of finite element assembly methods, we point out that the insertion time taken by the quadrature-free algorithm to update the entries of $\mathbf{V}$ is likely to be less than the inerstion time taken by the quadrature-based algorithm to perform the same action. While insertion time is not likely to be a dominating factor between the two algorithms, the quadrature-based computation of $\int_{\mathcal{P}_n} x^i y^j \ \mathrm{d}\mathbf{x}$ requires one insertion into $(\mathbf{V})_{ij}$ for each quadrature point (which scales with the size of $\mathcal{S}_p$), while the number of insertions into $(\mathbf{V})_{ij}$ performed by the quadrature-free-base computation of the same integral does not scale with the size of $\mathcal{S}_p$.

It is important to note that the performance of the quadrature-based algorithm (in

finite element applications) can be greatly improved on massively-parallel architectures [65]. It is expected that the quadrature-free integration method outlined earlier may not be competitive with such quadrature-based methods, in part due to the fact that the entries of $\mathbf{V}$ (and $\mathbf{F}$) must be assembled in a specific order.

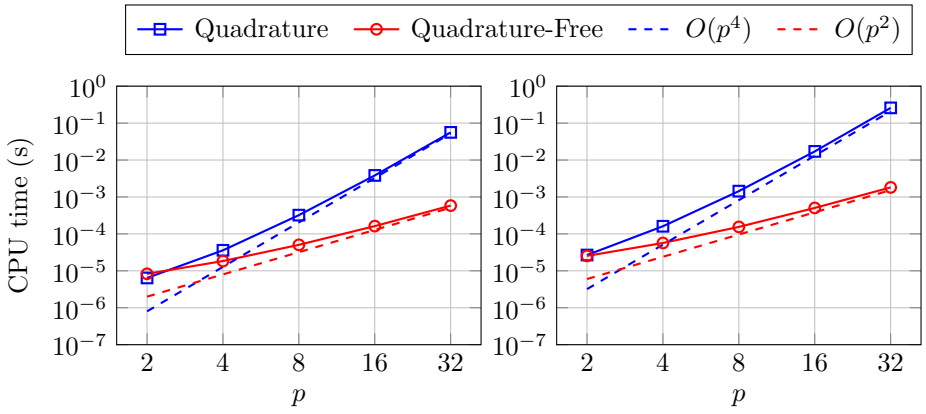

Figure 4.2: CPU times taken by the quadrature-based and quadrature-free-based methods to evaluate $\mathcal{S}_p$ for $p = 2, 4, 8, 16, 32$ on a regular n-gon. Left: $n = 5$. Right: $n = 16$.

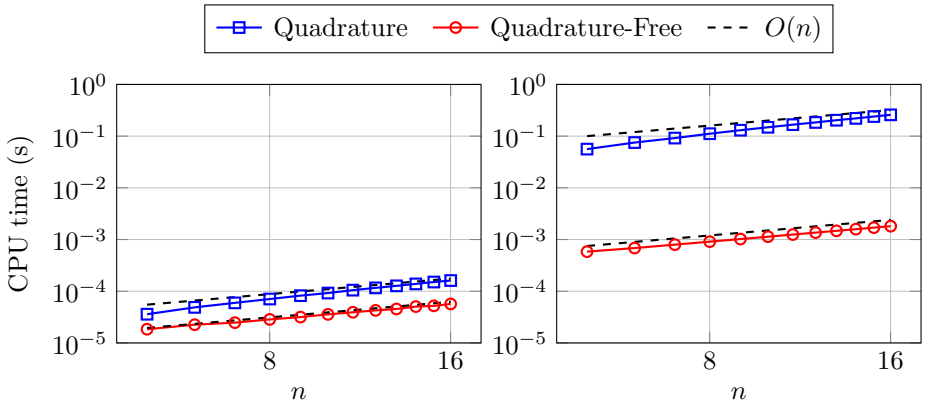

Figure 4.3: CPU times taken by the quadrature-based and quadrature-free-based methods to evaluate $\mathcal{S}_p$ for $5 \leq n \leq 16$ and fixed p. Left: $p = 4$. Right: $p = 32$.

4.2 Application to DG Methods

We now turn our attention to the application of the quadrature-free integration method for homogeneous functions to the discontinuous Galerking finite element method applied to the transport equation (2.22). The DGFEM scheme reads as in (2.23): find $u_h \in \mathcal{V}_\Omega$ such that

$$\sum_{\kappa \in \mathcal{V}_\Omega} \left(\int_\kappa -u_h \boldsymbol{\mu} \cdot \nabla v_h + b u_h v_h \ \mathrm{d}\mathbf{x} + \int_{\partial_+ \kappa} |\boldsymbol{\mu} \cdot \mathbf{n}| u_h^+ v_h^+ \ \mathrm{d}s - \int_{\partial_- \kappa \setminus \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}| u_h^- v_h^+ \ \mathrm{d}s \right)$$
$$= \sum_{\kappa \in \mathcal{V}_\Omega} \left(\int_\kappa f v_h \ \mathrm{d}\mathbf{x} + \int_{\partial_- \kappa \cap \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}| g v_h^+ \ \mathrm{d}s \right) \tag{4.6}$$

for all $v_h \in \mathcal{V}_\Omega$. Here, we suppress the dependence on $\boldsymbol{\mu} \in \mathbb{S}$ of the element inflow/outflow boundaries $\partial_\pm \kappa$ for notational simplicity, and denote the reaction coefficient by b rather than α to avoid notational clashes. Henceforth, we shall assume that the data term b is piecewise-constant with respect to the mesh $\mathcal{T}_\Omega$; that is, $b|_\kappa = b^{(\kappa)} \in \mathbb{R}_{\geq 0}$ for all $\kappa \in \mathcal{T}_\Omega$.

Recall that $\mathcal{V}_\Omega$ denotes a finite element space of piecewise-polynomial functions defined on the mesh $\mathcal{T}_\Omega$ such that, for any $v_h \in \mathcal{V}_\Omega$, we have $v_h|\kappa \in \mathbb{H}^{p_\kappa}(\kappa)$, where p_κ denotes the maximal polynomial degree of any function on $\kappa \in \mathcal{T}_\Omega$. Common choices for $\mathbb{H}^p(\kappa)$ are $\mathbb{P}^p(\kappa)$, the space of all polynomial functions of maximal total degree p on κ, and $\mathbb{Q}^p(\kappa)$, the space of all polynomial functions of maximal degree p in each variable on κ. We shall postpone the discussion of different choices of $\mathbb{H}^p(\kappa)$ until the analysis of the quadrature-free implementation, but we will enforce that $p_\kappa = p$ for all $\kappa \in \mathcal{T}_\Omega$ for simplicity.

In practice, the implementation of (4.6) is first performed by selecting a complete and linearly-independent finite element basis $\{\phi_i\}_{i=1}^N \subset \mathcal{V}_\Omega$ with $N = \dim \mathcal{V}_\Omega$ and constructing the system

$$\mathbf{T}\mathbf{u} = \mathbf{f}, \tag{4.7}$$

where the $(i,j)^{th}$ entry of $\mathbf{T}$ and the i^{th} entry of $\mathbf{f}$ are given respectively by

$$\begin{aligned}(\mathbf{T})_{ij} = \sum_{\kappa \in \mathcal{V}_\Omega} \Bigg(&\int_\kappa -\phi_j \boldsymbol{\mu} \cdot \nabla \phi_i + b^{(\kappa)} \phi_i \phi_j \ \mathrm{d}\mathbf{x} + \int_{\partial_+ \kappa} |\boldsymbol{\mu} \cdot \mathbf{n}| \phi_i^+ \phi_j^+ \ \mathrm{d}s \\ &- \int_{\partial_- \kappa \setminus \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}| \phi_i^+ \phi_j^- \ \mathrm{d}s \Bigg),\end{aligned} \tag{4.8}$$

$$(\mathbf{f})_i = \sum_{\kappa \in \mathcal{V}_\Omega} \left(\int_\kappa f \phi_i \ \mathrm{d}\mathbf{x} + \int_{\partial_- \kappa \cap \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}| g \phi_i^+ \ \mathrm{d}s \right). \tag{4.9}$$

Before introducing the quadrature-free-based method, we shall separate the volume and face contributions to the system matrix in (4.8). For each element $\kappa \in \mathcal{T}_\Omega$, we seek to compute the local elemental contribution $\mathbf{T}_v^\kappa$ with entries

$$\begin{aligned}(\mathbf{T}_v^\kappa)_{ij} &= \int_\kappa -\phi_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) \boldsymbol{\mu} \cdot \nabla \phi_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) + b^{(\kappa)} \phi_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) \ \mathrm{d}\mathbf{x} \\ &= \int_\kappa -\sum_{k=1}^d \mu_k \frac{\partial \phi_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x})}{\partial x_k} \phi_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) + b^{(\kappa)} \phi_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) \ \mathrm{d}\mathbf{x}.\end{aligned} \tag{4.10}$$

We will also seek to compute local face contributions, for which it shall be useful to rewrite the element boundary contributions in (4.8) as sums of integrals over the faces in the skeleton of the spatial mesh. To this end, we shall denote by $\mathcal{F}_\Omega$ the set of faces in the mesh $\mathcal{T}_\Omega$, and further partition $\mathcal{F}_\Omega$ as

$$\mathcal{F}_\Omega = \mathcal{F}_\Omega^- \cup \mathcal{F}_\Omega^+ \cup \mathcal{F}_\Omega^{int},$$

where $\mathcal{F}_\Omega^-$ denotes the set of inflow boundary faces on $\partial_-\Omega$, $\mathcal{F}_\Omega^+$ denotes the set of outflow boundary faces on $\partial_+\Omega$, and $\mathcal{F}_\Omega^{int}$ denotes the set of interior faces. Henceforth, we assume that each face $e \in \mathcal{F}_\Omega$ is planar.

Additionally, we shall redefine our trace notation. For each face $e \in \mathcal{F}_\Omega$, we assign a unit normal $\mathbf{n}_e$. The element for which $\mathbf{n}_e$ is an outward (resp. inward) unit normal is denoted by $\kappa^+ = \kappa_e^+$ (resp. $\kappa^- = \kappa_e^-$) and for any function $v \in \mathcal{V}_\Omega$ we denote by v^+ (resp. v^-) the trace of v on e from κ^+ (resp. κ^-).

For functions $\phi_1, \phi_2 \in \mathcal{V}_\Omega$, the sum over element boundary integrals may be rewritten as a sum over face integrals as follows:

$$\begin{aligned}
&\sum_{\kappa \in \mathcal{T}_\Omega} \left(\int_{\partial_+ \kappa} |\boldsymbol{\mu} \cdot \mathbf{n}| \phi_1^+ \phi_2^+ \ \mathrm{d}s - \int_{\partial_- \kappa \setminus \partial\Omega} |\boldsymbol{\mu} \cdot \mathbf{n}| \phi_1^+ \phi_2^- \ \mathrm{d}s \right) \\
&= \sum_{e \in \mathcal{F}_\Omega^+} |\boldsymbol{\mu} \cdot \mathbf{n}_e| \int_e \phi_1^+ \phi_2^+ \ \mathrm{d}s + \sum_{e \in \mathcal{F}_\Omega^{int}} |\boldsymbol{\mu} \cdot \mathbf{n}_e| \mathbb{I}_{\{\boldsymbol{\mu} \cdot \mathbf{n}_e \geq 0\}} \int_e \phi_1^+ \phi_2^+ - \phi_1^- \phi_2^+ \ \mathrm{d}s \\
&\quad + \sum_{e \in \mathcal{F}_\Omega^{int}} |\boldsymbol{\mu} \cdot \mathbf{n}_e| \mathbb{I}_{\{\boldsymbol{\mu} \cdot \mathbf{n}_e < 0\}} \int_e \phi_1^- \phi_2^- - \phi_1^+ \phi_2^- \ \mathrm{d}s.
\end{aligned}$$

Here, the trace notation used on the left-hand-side of the equality is that used in Chapter 2.2.3 and the trace notation used on the right-hand-side of the equality is described above. For a statement S, $\mathbb{I}_S$ denotes the indicator function returning 1 if S is true and 0 if S is false. By setting $\phi_1 = \phi_{\boldsymbol{\alpha}^{(i)}}$ and $\phi_2 = \phi_{\boldsymbol{\alpha}^{(j)}}$, we seek to compute the local face contributions $\mathbf{T}_f^e$ with entries

$$\begin{aligned}
(\mathbf{T}_f^e)_{ij} = &\sum_{e \in \mathcal{F}_\Omega^+} |\boldsymbol{\mu} \cdot \mathbf{n}_e| \int_e \phi^+_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi^+_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) \ \mathrm{d}s \\
&+ \sum_{e \in \mathcal{F}_\Omega^{int}} |\boldsymbol{\mu} \cdot \mathbf{n}_e| \mathbb{I}_{\{\boldsymbol{\mu} \cdot \mathbf{n}_e \geq 0\}} \int_e \phi^+_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi^+_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) - \phi^-_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi^+_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) \ \mathrm{d}s \\
&+ \sum_{e \in \mathcal{F}_\Omega^{int}} |\boldsymbol{\mu} \cdot \mathbf{n}_e| \mathbb{I}_{\{\boldsymbol{\mu} \cdot \mathbf{n}_e < 0\}} \int_e \phi^-_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi^-_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) - \phi^+_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi^-_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) \ \mathrm{d}s.
\end{aligned} \tag{4.11}$$

We note that the assembly of the matrix $\mathbf{T}_f^e$ is equivalent to the assembly of the on-diagonal and off-diagonal matrices $\mathbf{T}_f^{(e,\pm,\pm)}$ and $\mathbf{T}_f^{(e,\pm,\mp)}$ with entries

$$(\mathbf{T}_f^{(e,\pm,\pm)})_{ij} = \int_e \phi^\pm_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi^\pm_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) \ \mathrm{d}s, \tag{4.12}$$

$$(\mathbf{T}_f^{(e,\pm,\mp)})_{ij} = \int_e \phi^\pm_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi^\mp_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) \ \mathrm{d}s. \tag{4.13}$$

Depending on the sign of $\boldsymbol{\mu} \cdot \mathbf{n}_e$, only one of the matrices in (4.12) needs to be assembled. Furthermore, if e is an interior face, only one of the matrices in (4.13) needs to be assembled.

We shall focus primarily on the assembly of the matrix $\mathbf{T}$ due to our simplifying assumption that the wind direction $\boldsymbol{\mu}$ is constant on Ω and the coefficient b is piecewise-constant with respect to the spatial mesh $\mathcal{T}_\Omega$. While we will not cover the assembly of the vector $\mathbf{f}$ in (4.9), we emphasize that this term is typically assembled in a standard manner using quadrature schemes on the elements and faces in the mesh $\mathcal{T}_\Omega$, since the functions f and g are generally not homogeneous functions (or linear combinations thereof). However, if f and g can be expressed as linear combinations of the basis

functions $\{\phi_i\}_{i=1}^N$ one can, in principle, exploit the quadrature-free assembly approach outlined below.

4.2.1 Defining bases on polytopic elements

The basis functions $\{\phi_i\}_{i=1}^N$ employed in DGFEM discretisations are commonly selected to be compactly supported on each element $\kappa \in \mathcal{T}_\Omega$, but can otherwise be prescribed arbitrarily. When standard element shapes are employed, the basis functions are often defined on a reference element which is (affinely) mapped onto the physical elements. However, when $\mathcal{T}_\Omega$ consists of polytopic elements, mappings from reference to physical elements may be highly complicated (if the local-to-physical mapping is allowed to be non-affine) or may require a cumbersome number of reference elements (if the local-to-physical mapping is restricted to be affine). Following [6, 29], we opt to define basis functions on a simpler reference geometry which may be affinely mapped to a bounding box of the physical element.

For any d-dimensional polytope $\kappa \in \mathcal{T}_\Omega$, define the *bounding box* $B_\kappa \in \mathbb{R}^d$ by

$$B_\kappa = \prod_{k=1}^{d} [r_k^{(\kappa)}, s_k^{(\kappa)}] \tag{4.14}$$

with each $r_k^{(\kappa)}, s_k^{(\kappa)} \in \mathbb{R}$ and $r_k^{(\kappa)} < s_k^{(\kappa)}$, $1 \leq k \leq d$, chosen such that B_κ is the d-dimensional hypercube of smallest measure satisfying $B_\kappa \supseteq \kappa$. We also define a reference bounding box $\hat{B}$ by

$$\hat{B} = \prod_{k=1}^{d} [-1, 1]. \tag{4.15}$$

Remark. *By applying a rotation $R : \mathbb{R}^d \to \mathbb{R}^d$ to κ, it may be possible to find a bounding box, say $B_{R\kappa}$, for the rotated element $R\kappa$ of the form (4.14) with smaller Lebesgue measure than B_κ. A different (and tighter) bounding box for κ is then given by $R^{-1}B_{R\kappa}$, though we note that the edges of $R^{-1}B_{R\kappa}$ are not necessarily aligned with the Cartesian coordinate directions.*

Let $F_\kappa : \hat{B} \to B_\kappa$ denote the affine map between the two bounding boxes of the form $F_\kappa(\hat{\mathbf{x}}) = \mathbf{J}_\kappa \hat{\mathbf{x}} + \mathbf{t}_\kappa$ for $\hat{\mathbf{x}} \in \hat{B}$, where $\mathbf{J}_\kappa \in \mathbb{R}^{d \times d}$ and $\mathbf{t}_\kappa \in \mathbb{R}^d$ are respectively defined by

$$\mathbf{J}_\kappa = \begin{pmatrix} \frac{s_1^{(\kappa)} - r_1^{(\kappa)}}{2} & & & \\ & \frac{s_2^{(\kappa)} - r_2^{(\kappa)}}{2} & & \\ & & \ddots & \\ & & & \frac{s_d^{(\kappa)} - r_d^{(\kappa)}}{2} \end{pmatrix}, \tag{4.16}$$

$$\mathbf{t}_\kappa = \begin{pmatrix} \frac{r_1^{(\kappa)} + s_1^{(\kappa)}}{2} \\ \frac{r_2^{(\kappa)} + s_2^{(\kappa)}}{2} \\ \vdots \\ \frac{r_d^{(\kappa)} + s_d^{(\kappa)}}{2} \end{pmatrix}. \tag{4.17}$$

Note that $\mathbf{J}_\kappa$ is a diagonal matrix and denotes the Jacobian of the mapping F_κ, and that $\mathbf{t}_\kappa$ denotes the translation of the centroid of $\hat{B}$ to the centroid of B_κ. Moreover, the determinant of the Jacobian is constant and is given by

$$|\mathbf{J}_\kappa| = \prod_{k=1}^{d} (\mathbf{J}_\kappa)_{kk} = \prod_{k=1}^{d} \frac{s_k^{(\kappa)} - r_k^{(\kappa)}}{2}. \tag{4.18}$$

In order to introduce a basis of $\mathbb{H}^p(\kappa)$, we first define a set of multi-indices $\mathcal{I}_\kappa = \mathcal{I}_{\mathbb{H}^p(\kappa)}$ by

$$\mathcal{I}_\kappa = \left\{ \boldsymbol{\alpha} = (\alpha_k)_{k=1}^d \in \mathbb{N}_0^d : \mathbf{x}^{\boldsymbol{\alpha}} = \prod_{k=1}^{d} x_k^{\alpha_k} \in \mathbb{H}^p(\kappa) \right\}. \tag{4.19}$$

As in [6], we define reference basis functions $\hat{\phi}_{\boldsymbol{\alpha}} : \hat{B} \to \mathbb{R}$ for each $\boldsymbol{\alpha} \in \mathcal{I}_\kappa$ as a product of univariate Legendre polynomials of degrees specified by the entries of $\boldsymbol{\alpha}$:

$$\hat{\phi}_{\boldsymbol{\alpha}}(\mathbf{x}) = \prod_{k=1}^{d} L_{\alpha_k}(\hat{x}_k), \tag{4.20}$$

where $L_n(x)$ denotes the Legendre polynomial of degree $n \geq 0$ defined on $[-1, 1]$ by

$$L_n(x) = \frac{1}{2^n n!} \frac{\mathrm{d}^n}{\mathrm{d}x^n} (x^2 - 1)^n. \tag{4.21}$$

It can be seen that $\{\hat{\phi}_{\boldsymbol{\alpha}}\}_{\boldsymbol{\alpha} \in \mathcal{I}_\kappa}$ is a complete, linearly-independent and orthogonal set of basis functions of $\mathbb{H}^p(\kappa)$.

Finally, the basis $\{\phi_{\boldsymbol{\alpha}}^{(\kappa)}\}_{\boldsymbol{\alpha} \in \mathcal{I}_\kappa} \subset \mathbb{H}^p(\kappa)$ is constructed for each $\boldsymbol{\alpha} \in \mathcal{I}_\kappa$ and $\kappa \in \mathcal{T}_\Omega$ by

$$\phi_{\boldsymbol{\alpha}}^{(\kappa)}(\mathbf{x}) = \hat{\phi}_{\boldsymbol{\alpha}}(F_\kappa^{-1}\mathbf{x})|_{\hat{\kappa}} = \begin{cases} \hat{\phi}_{\boldsymbol{\alpha}}(F_\kappa^{-1}\mathbf{x}) & \text{for } \mathbf{x} \in \kappa, \\ 0 & \text{otherwise.} \end{cases} \tag{4.22}$$

In other words, the basis function $\phi_{\boldsymbol{\alpha}}^{(\kappa)}$ is the restriction of $\hat{\phi}_{\boldsymbol{\alpha}} \circ F_\kappa^{-1}$ to $\hat{\kappa} = F_\kappa^{-1}\kappa$, the image of κ under F_κ^{-1}. The basis of $\mathcal{V}_\Omega$ can therefore be written as the following set:

$$\left\{ \phi_{\boldsymbol{\alpha}}^{(\kappa)} : \boldsymbol{\alpha} \in \mathcal{I}_\kappa, \kappa \in \mathcal{T}_\Omega \right\}. \tag{4.23}$$

Henceforth, the basis function $\phi_i^{(\kappa)}$, $1 \leq i \leq \dim \mathbb{H}^p(\kappa)$ shall be uniquely associated to a multi-index $\boldsymbol{\alpha}^{(i)} \in \mathcal{I}_\kappa$, and we will write $\phi_i = \phi_{\boldsymbol{\alpha}^{(i)}}$ for clarity.

The basis functions (4.22) have a few desirable properties. Since $\{\phi_i\}_{i=1}^{\dim \mathbb{H}^p(\kappa)}$ are defined as products of univariate Legendre polynomials defined on bounding boxes, they are orthogonal with respect to the $L^2(B_\kappa)$-inner product and the corresponding mass matrix $\mathbf{M}_{B_\kappa} \in \mathbb{R}^{\dim \mathbb{H}^p(\kappa) \times \dim \mathbb{H}^p(\kappa)}$ has entries:

$$(\mathbf{M}_{B_\kappa})_{ij} = \int_{B_\kappa} \phi_i(\mathbf{x}) \phi_j(\mathbf{x}) \ \mathrm{d}\mathbf{x} \begin{cases} > 0 & \text{for } i = j, \\ = 0 & \text{for } i \neq j. \end{cases}$$

As a result, the corresponding mass matrix is well-conditioned. While $\{\phi_i\}_{i=1}^{\dim \mathbb{H}^p(\kappa)}$ are no longer orthogonal with respect to the $L^2(\kappa)$-inner product, the corresponding mass-matrix $\mathbf{M}_\kappa \in \mathbb{R}^{\dim \mathbb{H}^p(\kappa) \times \dim \mathbb{H}^p(\kappa)}$ is expected to be well-conditioned, provided that the bounding box B_κ is not too much larger than κ.

Moreover, noting that the inverse mapping $F_\kappa^{-1} : B_\kappa \to \hat{B}$ is given by $F_\kappa^{-1}(\mathbf{x}) = \mathbf{J}_\kappa^{-1}(\mathbf{x} - \mathbf{t}_\kappa)$, we have that

$$\begin{aligned}\hat{\phi}_{\boldsymbol{\alpha}}(F_\kappa^{-1}\mathbf{x}) &= \prod_{k=1}^{d} L_{\alpha_k}\left((\mathbf{J}_\kappa^{-1}(\mathbf{x}-\mathbf{t}_\kappa)_k\right) \\ &= \prod_{k=1}^{d} L_{\alpha_k}\left(\frac{2}{s_k^{(\kappa)} - r_k^{(\kappa)}}\left(x_k - \frac{r_k^{(\kappa)} + s_k^{(\kappa)}}{2}\right)\right),\end{aligned} \tag{4.24}$$

where we have written $\mathbf{x} = (x_k)_{k=1}^d \in \mathbb{R}^d$. Thus, the basis functions $\hat{\phi}_{\boldsymbol{\alpha}}$ are separable - this turns out to be a useful property in the upcoming quadrature-free development.

Remark. *If the bounding box of κ has been rotated as in the previous remark, the mapping $F_\kappa : \hat{B} \to R^{-1}B_{R\kappa}$ can still be written in the form $F_\kappa(\hat{\mathbf{x}}) = \mathbf{J}_\kappa\hat{\mathbf{x}} + \mathbf{t}_\kappa$; however, $\mathbf{J}_\kappa$ is no longer diagonal. Thus, the basis functions defined in (4.22) cannot be decomposed as in (4.24).*

For the remainder of this section, it will prove important to express the Legendre polynomials (4.21) and their first derivatives as a linear combination of monomials:

$$L_n(x) = \sum_{k=0}^{n} C_{n,k}x^k, \tag{4.25}$$

$$\frac{\mathrm{d}L_n(x)}{\mathrm{d}x} = \sum_{k=0}^{n-1} C'_{n,k}x^k = \sum_{k=0}^{n-1}(k+1)C_{n,k+1}x^k, \tag{4.26}$$

as well as their pairwise-products:

$$\begin{aligned}L_m(x)L_n(x) &= \sum_{k=0}^{m+n} C_{m,n,k}x^k \\ &= \sum_{k=0}^{m+n}\left(\sum_{\ell_1+\ell_2=k} C_{m,\ell_1}C_{n,\ell_2}\right)x^k,\end{aligned} \tag{4.27}$$

$$\begin{aligned}\frac{\mathrm{d}L_m(x)}{\mathrm{d}x}L_n(x) &= \sum_{k=0}^{m+n-1} C'_{m,n,k}x^k \\ &= \sum_{k=0}^{m+n-1}\left(\sum_{\ell_1+\ell_2=k} C'_{m,\ell_1}C_{n,\ell_2}\right)x^k.\end{aligned} \tag{4.28}$$

4.2.2 Rewriting the volume integrals

In order to apply the quadrature-free integration technique to the volume integrals appearing in (4.10), we shall change variables to integrate over $\hat{\kappa} = F_\kappa^{-1}\kappa \subseteq \hat{B}$. By virtue of the fact that $\mathbf{J}_\kappa$ is diagonal, we have

$$\frac{\partial\phi_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x})}{\partial x_k} = \frac{1}{(\mathbf{J}_\kappa)_{kk}}\frac{\partial\hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x})}{\partial\hat{x}_k}$$

for $1 \le k \le d$. This allows us to rewrite (4.10) as

$$\begin{aligned}(\mathbf{T}_v^\kappa)_{ij} &= \int_{\hat{\kappa}}\left[-\sum_{k=1}^{d}\frac{\mu_k}{(\mathbf{J}_\kappa)_{kk}}\frac{\partial\hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x})}{\partial x_k}\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) + b^{(\kappa)}\hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x})\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x})\right]|\mathbf{J}_\kappa|\ \mathrm{d}\hat{\mathbf{x}} \\ &= \int_{\hat{\kappa}}\left[-\sum_{k=1}^{d}\hat{\mu}_k^{(\kappa)}\frac{\partial\hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x})}{\partial x_k}\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) + b^{(\kappa)}\hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x})\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x})\right]|\mathbf{J}_\kappa|\ \mathrm{d}\hat{\mathbf{x}},\end{aligned}$$

where we have defined the scaled wind direction $\hat{\boldsymbol{\mu}}^{(\kappa)} = (\hat{\mu}_k^{(\kappa)})_{k=1}^d \in \mathbb{R}^d$ by $\hat{\boldsymbol{\mu}}^{(\kappa)} = \mathbf{J}_\kappa^{-1}\boldsymbol{\mu}$ for brevity. Using (4.27), the reaction term can be written as

$$\begin{aligned} b^{(\kappa)}\hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x})\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) &= b^{(\kappa)}\left(\prod_{k=1}^d L_{\alpha_k^{(i)}}(\hat{x}_k)\right)\left(\prod_{k=1}^d L_{\alpha_k^{(j)}}(\hat{x}_k)\right) \\ &= b^{(\kappa)}\prod_{k=1}^d L_{\alpha_k^{(i)}}(\hat{x}_k)L_{\alpha_k^{(j)}}(\hat{x}_k) \\ &= b^{(\kappa)}\prod_{k=1}^d \sum_{n_k=0}^{\alpha_k^{(i)}+\alpha_k^{(j)}} C_{\alpha_k^{(i)},\alpha_k^{(j)},n_k}\hat{x}_k^{n_k} \\ &= b^{(\kappa)}\sum_{\mathbf{0}\le\boldsymbol{\alpha}\le\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}} C_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}\hat{\mathbf{x}}^{\boldsymbol{\alpha}}, \end{aligned}$$

where the coefficients $C_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ are defined by

$$C_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} = \prod_{k=1}^d C_{\alpha_k^{(i)},\alpha_k^{(j)},\alpha_k}.$$

Similarly, using (4.27) and (4.28), the streaming term can be written as

$$\begin{aligned} \hat{\mu}_k^{(\kappa)}\frac{\partial\hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x})}{\partial x_k}\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) &= \hat{\mu}_k^{(\kappa)}\left(\frac{\partial}{\partial\hat{x}_k}\prod_{\ell=1}^d L_{\alpha_\ell^{(i)}}(\hat{x}_\ell)\right)\left(\prod_{\ell=1}^d L_{\alpha_\ell^{(j)}}(\hat{x}_\ell)\right) \\ &= \hat{\mu}_k^{(\kappa)}\left(\prod_{\substack{\ell=1\\\ell\neq k}}^d L_{\alpha_\ell^{(i)}}(\hat{x}_\ell)L_{\alpha_\ell^{(j)}}(\hat{x}_\ell)\right)\frac{\mathrm{d}L_{\alpha_k^{(i)}}(\hat{x}_k)}{\mathrm{d}\hat{x}_k}L_{\alpha_k^{(j)}}(\hat{x}_k) \\ &= \hat{\mu}_k^{(\kappa)}\left(\prod_{\substack{\ell=1\\\ell\neq k}}^d \sum_{n_\ell=0}^{\alpha_\ell^{(i)}+\alpha_\ell^{(j)}} C_{\alpha_\ell^{(i)},\alpha_\ell^{(j)},n_\ell}\hat{x}_\ell^{n_\ell}\right)\sum_{n=0}^{\alpha_k^{(i)}+\alpha_k^{(j)}-1} C'_{\alpha_k^{(i)},\alpha_k^{(j)},n}\hat{x}_k^n \\ &= \hat{\mu}_k^{(\kappa)}\sum_{\mathbf{0}\le\boldsymbol{\alpha}\le\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}} C^{(k)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}\hat{\mathbf{x}}^{\boldsymbol{\alpha}}, \end{aligned}$$

where the coefficients $C^{(k)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ are defined for each $1 \le k \le d$ by

$$C^{(k)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} = \left(\prod_{\substack{\ell=1\\\ell\neq k}}^d C_{\alpha_\ell^{(i)},\alpha_\ell^{(j)},\alpha_k}\right)C'_{\alpha_k^{(i)},\alpha_k^{(j)},\alpha_k}.$$

To this end, we are now ready to rewrite (4.10) in a form that is readily amenable for a quadrature-free implementation. For each element $\kappa \in \mathcal{T}_\Omega$, define the element-dependent coefficients $\mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ for each $\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)} \in \mathcal{I}_\kappa$ (or equivalently for each $1 \le i,j \le \dim\mathbb{H}^p(\kappa)$) and $\mathbf{0} \le \boldsymbol{\alpha} \le \boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}$ by

$$\mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} = -\sum_{k=1}^d \hat{\mu}_k^{(\kappa)}C^{(k)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} + b^{(\kappa)}C_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}. \tag{4.29}$$

Furthermore, denote the element boundary $\partial\hat{\kappa} = \{\partial\hat{\kappa}_m\}_{m=1}^n$, where n denotes the number of faces of $\hat{\kappa}$ and $\partial\hat{\kappa}_k$ denotes an external face of $\hat{\kappa}$ on which $\hat{\mathbf{x}}\cdot\hat{\mathbf{n}}_k = a_k$ with $a_k \in \mathbb{R}$

and $\hat{\mathbf{n}}_k$ the unit normal to $\hat{\kappa}$ on $\partial\hat{\kappa}_k$. Then, (4.10) may be written in the form

$$(\mathbf{T}_v^\kappa)_{ij} = \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}} \mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} |\mathbf{J}_\kappa| \int_{\hat{\kappa}} \hat{\mathbf{x}}^{\boldsymbol{\alpha}} \ \mathrm{d}\hat{\mathbf{x}} \tag{4.30}$$

$$= \sum_{k=1}^{n} a_k \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}} \frac{\mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} |\mathbf{J}_\kappa|}{d+|\boldsymbol{\alpha}|} \int_{\partial\hat{\kappa}_k} \hat{\mathbf{x}}^{\boldsymbol{\alpha}} \ \mathrm{d}\hat{s} \tag{4.31}$$

$$= \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}} \mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} \sum_{k=1}^{n} m_{\boldsymbol{\alpha}}^{(v,\kappa,\partial\kappa_k)},$$

where we have defined the quantity $m_{\boldsymbol{\alpha}}^{(v,\kappa,\partial\kappa_k)}$ for each face $\partial\kappa_k \in \partial\kappa$ by

$$m_{\boldsymbol{\alpha}}^{(v,\kappa,\partial\kappa_k)} = \frac{a_k|\mathbf{J}_\kappa|}{d+|\boldsymbol{\alpha}|} \int_{\partial\hat{\kappa}_k} \hat{\mathbf{x}}^{\boldsymbol{\alpha}} \ \mathrm{d}\hat{s}. \tag{4.32}$$

Equations (4.30) and (4.31) represent two different ways in which $(\mathbf{T}_v^\kappa)_{ij}$ may be assembled. For example, an implementation of (4.30) may compute the coefficients $\mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ and moments $\int_{\hat{\kappa}} \hat{\mathbf{x}}^{\boldsymbol{\alpha}} \ \mathrm{d}\hat{\mathbf{x}}$ once per element, whereas an implementation of (4.31) may recompute $\mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ and $\int_{\partial\hat{\kappa}_k} \hat{\mathbf{x}}^{\boldsymbol{\alpha}} \ \mathrm{d}\hat{s}$ once for each face.

In a practical implementation, the coefficient $\mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ need not be computed and stored for each element $\kappa \in \mathcal{T}_\Omega$. Instead, the coefficients $C_{m,n,k}$ and $C'_{m,n,k}$ (in (4.27) and (4.28) respectively) are independent of κ, and so may be pre-computed before assembly. The element-dependent coefficients $\mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ may then be computed on-the-fly as necessary.

Algorithm 2 shows pseudocode for a quadrature-free implementation of the element volume terms $\mathbf{T}_v^\kappa$ in (4.10) for a first-order transport problem. Algorithm 2 mirrors a similar algorithm outlined in [6], though in that case the quadrature-free volume assembly was tailored to a second-order elliptic PDE problem. As remarked in the earlier discussion of the quadrature-free monomial integration procedure outlined in Algorithm 1, the computational complexity of the operation on line 4 is linear with respect to the number of faces of κ. However, the computational complexity of the main assembly procedure in Algorithm 2 (lines 7 to 21) is independent of the geometry of κ. This contrasts with standard quadrature-based assembly methods in which lines 11 to 19 are replaced by a loop over volume quadrature points (which may grow as a function of the number of faces of κ if the geometry of κ is complex). A more rigorous comparison of quadrature-based and quadrature-free-based assembly methods is deferred to Chapter 4.3.

4.2.3 Rewriting the face integrals

We now apply the quadrature-free integration technique to the face integrals appearing in (4.12) and (4.13). We shall treat each case separately.

On-diagonal face matrix In this case, the integrand in (4.12) is either supported on the element κ^+ if $e \in \mathcal{F}_\Omega^+ \cup \mathcal{F}_\Omega^-$, or is supported on exactly one of the neighbouring

Algorithm 2 Quadrature-free assembly of $\mathbf{T}_v^\kappa$ using the assembly procedure (4.30) in the case of a two-dimensional element κ.

▷ *Setup*

Compute the coefficients $C_{m,n,k}$ and $C'_{m,n,k}$ as in (4.27) and (4.28) (if not already available)

Generate index set $\mathcal{J}_\kappa = \{\boldsymbol{\alpha}^{(i)} + \boldsymbol{\alpha}^{(j)} : \boldsymbol{\alpha}^{(i)}, \boldsymbol{\alpha}^{(j)} \in \mathcal{I}_\kappa\}$

$\hat{\kappa} \leftarrow F_\kappa^{-1}\kappa$

▷ *Compute integrals* $|\mathbf{J}_\kappa| \int_{\hat{\kappa}} \hat{\mathbf{x}}^{\boldsymbol{\alpha}} \ \mathrm{d}\hat{\mathbf{x}}$ *in (4.30), where* $\int_{\hat{\kappa}} \hat{\mathbf{x}}^{\boldsymbol{\alpha}} \ \mathrm{d}\hat{\mathbf{x}}$ *is evaluated using Algorithm 1*

$m_{(\alpha_1,\alpha_2)} \leftarrow |\mathbf{J}_\kappa| \int_{\hat{\kappa}} \hat{x}^{\alpha_1}\hat{y}^{\alpha_2} \ \mathrm{d}\hat{\mathbf{x}}$ for $(\alpha_1, \alpha_2) \in \mathcal{J}_\kappa$

$(\mathbf{T}_v^\kappa)_{ij} \leftarrow 0$ for $1 \le i, j \le \mathbb{H}^p(\kappa) =: N_\kappa$

$\hat{\mu}_k^{(\kappa)} \leftarrow \mu_k/(\mathbf{J}_\kappa)_{kk}$ for $1 \le k \le d$ ▷ Here, $d = 2$

▷ *Loop over test and trial functions on* κ*; note that the following operations are all independent of the geometric complexity of* κ

for $i = 1, \ldots, N_\kappa$ **do**

  Get $\boldsymbol{\alpha}^{(i)} = (\alpha_1^{(i)}, \alpha_2^{(i)})$

  for $j = 1, \ldots, N_\kappa$ **do**

    Get $\boldsymbol{\alpha}^{(j)} = (\alpha_1^{(j)}, \alpha_2^{(j)})$

    ▷ *Loop over all monomials in integrand of* $T(\phi_j, \phi_i)$ *- one "for" loop for each independent variable*

    for $\alpha_1 = 0, \ldots, \alpha_1^{(i)} + \alpha_1^{(j)}$ **do**

      $c_1 \leftarrow C_{\alpha_1^{(i)},\alpha_1^{(j)},\alpha_1}$

      $c_1' \leftarrow C'_{\alpha_1^{(i)},\alpha_1^{(j)},\alpha_1}$

      for $\alpha_2 = 0, \ldots, \alpha_2^{(i)} + \alpha_2^{(j)}$ **do**

        $c_2 \leftarrow C_{\alpha_2^{(i)},\alpha_2^{(j)},\alpha_2}$

        $c_2' \leftarrow C'_{\alpha_2^{(i)},\alpha_2^{(j)},\alpha_2}$

        ▷ *Increment* $(\mathbf{T}_v^\kappa)_{ij}$ *with single term from sum in (4.30); here,* $\mathcal{C}^{(\kappa)}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} = -\hat{\mu}_1^{(\kappa)} c_1' c_2 - \hat{\mu}_2^{(\kappa)} c_1 c_2' + b^{(\kappa)} c_1 c_2$ *(as in (4.29)) with* $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$

        $(\mathbf{T}_v^\kappa)_{ij} \leftarrow (\mathbf{T}_v^\kappa)_{ij} + m_{(\alpha_1,\alpha_2)} \left[-\hat{\mu}_1^{(\kappa)} c_1' c_2 - \hat{\mu}_2^{(\kappa)} c_1 c_2' + b^{(\kappa)} c_1 c_2 \right]$

      end for

    end for

  end for

end for

elements $\kappa^\pm$ if $e \in \mathcal{F}_\Omega^{int}$. We shall apply the map $F_{\kappa^\pm}^{-1}$ to the face and obtain a face integral over the mapped face $\hat{e}_{\kappa^\pm} = F_{\kappa^\pm}^{-1} e \subset \hat{B}$:

$$\int_e \phi^\pm_{\boldsymbol{\alpha}^{(i)}}(\mathbf{x}) \phi^\pm_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) \ \mathrm{d}s = \int_{\hat{e}_{\kappa^\pm}} \hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\hat{\mathbf{x}}) \hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(\hat{\mathbf{x}}) \ ||\mathbf{J}_{\kappa^\pm}^{-\top} \hat{\mathbf{n}}_{\hat{e}^\pm}|| \ |\mathbf{J}_{\kappa^\pm}| \ \mathrm{d}\hat{s}.$$

Remarking that $||\mathbf{J}_{\kappa^\pm}^{-1} \hat{\mathbf{n}}_{\hat{e}^\pm}|| \ |\mathbf{J}_{\kappa^\pm}|$ is constant on $\hat{e}_{\kappa^\pm}$, and reusing our previous work

concerning the treatment of the volume terms, we rewrite the on-diagonal matrix (4.12) as

$$(\mathbf{T}_f^{(e,\pm,\pm)})_{ij} = \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}} C_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} ||\mathbf{J}_{\kappa^\pm}^{-\top}\hat{\mathbf{n}}_{\hat{e}^\pm}||\ |\mathbf{J}_{\kappa^\pm}| \int_{\hat{e}_{\kappa^\pm}} \hat{\mathbf{x}}^{\boldsymbol{\alpha}}\ \mathrm{d}\hat{s} \tag{4.33}$$
$$= \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}} C_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} m_{\boldsymbol{\alpha}}^{(f,\kappa^\pm,e_{\kappa^\pm})},$$

where we have defined the quantity $m_{\boldsymbol{\alpha}}^{(f,\kappa,e)}$ for each face $e \in \partial\kappa$ by

$$m_{\boldsymbol{\alpha}}^{(f,\kappa,e)} = ||\mathbf{J}_{\kappa^\pm}^{-\top}\hat{\mathbf{n}}_{\hat{e}^\pm}||\ |\mathbf{J}_{\kappa^\pm}| \int_{\hat{e}_{\kappa^\pm}} \hat{\mathbf{x}}^{\boldsymbol{\alpha}}\ \mathrm{d}\hat{s}. \tag{4.34}$$

Off-diagonal face matrix In this case, the test and trial functions in the integrand of (4.13) are supported on different neighbouring elements to $e \in \mathcal{F}_\Omega^{int}$. Applying the map $F_{\kappa^\pm}^{-1}$ to the face yields the following face integral:

$$\int_e \phi_{\boldsymbol{\alpha}^{(i)}}^{\pm}(\mathbf{x})\phi_{\boldsymbol{\alpha}^{(j)}}^{\mp}(\mathbf{x})\ \mathrm{d}s = \int_{\hat{e}_{\kappa^\pm}} \phi_{\boldsymbol{\alpha}^{(i)}}^{\pm}(F_{\kappa^\pm}\hat{\mathbf{x}})\phi_{\boldsymbol{\alpha}^{(j)}}^{\mp}(F_{\kappa^\pm}\hat{\mathbf{x}})\ ||\mathbf{J}_{\kappa^\pm}^{-\top}\hat{\mathbf{n}}_{\hat{e}^\pm}||\ |\mathbf{J}_{\kappa^\pm}|\ \mathrm{d}\hat{s}$$
$$= \int_{\hat{e}_{\kappa^\pm}} \hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\hat{\mathbf{x}})\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(F_{\kappa^\mp}^{-1}F_{\kappa^\pm}\hat{\mathbf{x}})\ ||\mathbf{J}_{\kappa^\pm}^{-\top}\hat{\mathbf{n}}_{\hat{e}^\pm}||\ |\mathbf{J}_{\kappa^\pm}|\ \mathrm{d}\hat{s}.$$

The composite map $F_{\kappa^\mp}^{-1}F_{\kappa^\pm} : \hat{B} \to \mathbb{R}^d$ describes the "cross-face" coordinate transformation mapping $F_{\kappa^\pm}^{-1}e$ to $F_{\kappa^\mp}^{-1}e$ and must be explicitly computed in order to exploit the quadrature-free integration algorithm. This map is affine and its action on $\hat{\mathbf{x}} \in \hat{B}$ can be written in the form $F_{\kappa^\mp}^{-1}F_{\kappa^\pm}\hat{\mathbf{x}} = \hat{\mathbf{J}}_{\kappa^\pm}\hat{\mathbf{x}} + \hat{\mathbf{t}}_{\kappa^\pm}$, where

$$\hat{\mathbf{J}}_{\kappa^\pm} = \mathbf{J}_{\kappa^\mp}^{-1}\mathbf{J}_{\kappa^\pm}, \tag{4.35}$$
$$\hat{\mathbf{t}}_{\kappa^\pm} = \mathbf{J}_{\kappa^\mp}^{-1}(\mathbf{t}_{\kappa^\pm} - \mathbf{t}_{\kappa^\mp}). \tag{4.36}$$

We note that the Jacobian of the map $F_{\kappa^\mp}^{-1}F_{\kappa^\pm}$ is diagonal. The product of the basis functions in the integrand above can then be written as

$$\hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\hat{\mathbf{x}})\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(F_{\kappa^\mp}^{-1}F_{\kappa^\pm}\hat{\mathbf{x}}) = \hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\hat{\mathbf{x}})\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(\hat{\mathbf{J}}_{\kappa^\pm}\hat{\mathbf{x}} + \hat{\mathbf{t}}_{\kappa^\pm})$$
$$= \prod_{k=1}^{d} L_{\alpha_k^{(i)}}(\hat{x}_k)L_{\alpha_k^{(j)}}\left((\hat{\mathbf{J}}_{\kappa^\pm})_{kk}\hat{x}_k + (\hat{\mathbf{t}}_{\kappa^\pm})_k\right).$$

We arrive at a product of Legendre polynomials in which one of the terms has a scaled and shifted argument. In order to obtain a representation in terms of monomials, we use (4.25) and the binomial theorem. Denoting by $p_{\kappa^\pm}$ the maximum value of $|\boldsymbol{\alpha}|$

for $\boldsymbol{\alpha} \in \mathcal{I}_{\kappa^\pm}$ we have that

$$\begin{aligned}
L_m(x)L_n(Jx+t) &= \left(\sum_{r=0}^{m} C_{m,r}x^r\right)\left(\sum_{s=0}^{n} C_{n,s}(Jx+t)^s\right) \\
&= \left(\sum_{r=0}^{m} C_{m,r}x^r\right)\left(\sum_{s=0}^{n} C_{n,s}\sum_{a=0}^{s}\binom{s}{a}(Jx)^a t^{s-a}\right) \\
&= \sum_{r=0}^{m}\sum_{s=0}^{n}\sum_{a=0}^{s}\binom{s}{a} C_{m,r}C_{n,s}J^a t^{s-a}x^{r+a} \\
&= \sum_{r=0}^{m}\sum_{a=0}^{n}\sum_{s=a}^{n}\binom{s}{a} C_{m,r}C_{n,s}J^a t^{s-a}x^{r+a} \\
&= \sum_{r=0}^{m}\sum_{a=0}^{n}\left(C_{m,r}J^a\sum_{s=a}^{n}\binom{s}{a}C_{n,s}t^{s-a}\right)x^{r+a} \\
&= \sum_{q=0}^{m+n}\left(\sum_{r+a=q} C_{m,r}J^a\sum_{s=a}^{n}\binom{s}{a}C_{n,s}t^{s-a}\right)x^q
\end{aligned}$$

for any $0 \leq m, n \leq \max\{p_{\kappa^+}, p_{\kappa^-}\}$. Therefore, by defining the coefficients

$$\tilde{C}^{(\kappa^\pm,k)}_{m,n,q} = \sum_{r+a=q} C_{m,r}(\tilde{\mathbf{J}}_{\kappa^\pm})^a_{kk}\sum_{s=a}^{n}\binom{s}{a}C_{n,s}(\tilde{\mathbf{t}}_{\kappa^\pm})^{s-a}_k \tag{4.37}$$

for $1 \leq k \leq d$, $0 \leq q \leq m+n$ and $0 \leq m, n \leq \max\{p_{\kappa^+}, p_{\kappa^-}\}$, we can write

$$\begin{aligned}
\hat{\phi}_{\boldsymbol{\alpha}^{(i)}}(\hat{\mathbf{x}})\hat{\phi}_{\boldsymbol{\alpha}^{(j)}}(F^{-1}_{\kappa^\mp}F_{\kappa^\pm}\hat{\mathbf{x}}) &= \prod_{k=1}^{d}\sum_{\alpha_k=0}^{\alpha_k^{(i)}+\alpha_k^{(j)}} \tilde{C}^{(\kappa^\pm,k)}_{\alpha_k^{(i)},\alpha_k^{(j)},\alpha_k}\hat{x}_k^{\alpha_k} \\
&= \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}}\left(\prod_{k=1}^{d}\tilde{C}^{(\kappa^\pm,k)}_{\alpha_k^{(i)},\alpha_k^{(j)},\alpha_k}\right)\hat{\mathbf{x}}^{\boldsymbol{\alpha}} \\
&= \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}}\tilde{C}^{\kappa^\pm}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}\hat{\mathbf{x}}^{\boldsymbol{\alpha}},
\end{aligned}$$

where the coefficient $\tilde{C}^{\kappa^\pm}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ is defined for each $\boldsymbol{\alpha}^{(i)} \in \mathcal{I}_{\kappa^\pm}$, $\boldsymbol{\alpha}^{(j)} \in \mathcal{I}_{\kappa^\mp}$ and $\mathbf{0} \leq \boldsymbol{\alpha} \leq \boldsymbol{\alpha}^{(i)} + \boldsymbol{\alpha}^{(j)}$ by by

$$\tilde{C}^{\kappa^\pm}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} = \prod_{k=1}^{d}\tilde{C}^{(\kappa^\pm,k)}_{\alpha_k^{(i)},\alpha_k^{(j)},\alpha_k}. \tag{4.38}$$

Finally, the off-diagonal matrix (4.13) may be written as

$$\begin{aligned}
(\mathbf{T}_f^{(e,\pm,\mp)})_{ij} &= \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}}\tilde{C}^{\kappa^\pm}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}\,||\mathbf{J}^{-\top}_{\kappa^\pm}\hat{\mathbf{n}}_{\hat{e}^\pm}||\,|\mathbf{J}_{\kappa^\pm}|\int_{\hat{e}_{\kappa^\pm}}\hat{\mathbf{x}}^{\boldsymbol{\alpha}}\,\mathrm{d}\hat{s} \\
&= \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\boldsymbol{\alpha}^{(i)}+\boldsymbol{\alpha}^{(j)}}\tilde{C}^{\kappa^\pm}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}\,m^{(f,\kappa^\pm,e_{\kappa^\pm})}_{\boldsymbol{\alpha}},
\end{aligned} \tag{4.39}$$

where we recall the definition of $m^{(f,\kappa,e)}_{\boldsymbol{\alpha}}$ from (4.34).

Owing to the cross-face mapping $F^{-1}_{\kappa^\mp}F_{\kappa^\pm}$, the coefficients $\tilde{C}^{\kappa^\pm}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ (or equivalently $\tilde{C}^{(\kappa^\pm,k)}_{m,n,q}$ for $1 \leq k \leq d$) must be computed for each face $e \in \mathcal{F}_\Omega$. These may be computed via Algorithm 3 - these coefficients are also computed in [6].

Algorithm 3 Computation of cross-face coefficients $\tilde{C}_{m,n,q}^{(\kappa^\pm,k)}$ in (4.37) for $0 \le m \le p_1$, $0 \le n \le p_2$, $0 \le q \le p_1 + p_2$ and $1 \le k \le d$ for any $d \ge 1$.

▷ *Setup*

Compute the coefficients $C_{a,b}$ for $0 \le a \le \max\{p_1, p_2\}$ and $0 \le b \le a$ as in (4.25) (if not already available)

Compute the binomial coefficients $\binom{s}{a}$ for $0 \le s \le p_2$ and $0 \le a \le s$ (if not already available)

$\tilde{C}_{n,a}^{(\kappa^\pm,k)} \leftarrow 0$ for $0 \le n \le p_2$, $0 \le a \le n$ and $1 \le k \le d$

$\tilde{C}_{m,n,q}^{(\kappa^\pm,k)} \leftarrow 0$ for $0 \le m \le p_1$, $0 \le n \le p_2$, $0 \le q \le p_1 + p_2$ and $1 \le k \le d$

▷ *Compute intermediate cross-face coefficients* $(\tilde{\mathbf{J}}_{\kappa^\pm})_{kk}^a \sum_{s=a}^n \binom{s}{a} C_{n,s} (\tilde{\mathbf{t}}_{\kappa^\pm})_k^{s-a}$ *appearing in (4.37)*

for $n = 0, \dots, p_2$ **do**
  for $a = 0, \dots, n$ **do**
    for $s = a, \dots, n$ **do**
      $\tilde{C}_{n,a}^{(\kappa^\pm,k)} \leftarrow \tilde{C}_{n,a}^{(\kappa^\pm,k)} + \binom{s}{a} C_{n,s} (\tilde{\mathbf{t}}_{\kappa^\pm})_k^{s-a}$ for $1 \le k \le d$
    end for
    $\tilde{C}_{n,a}^{(\kappa^\pm,k)} \leftarrow \left(\tilde{\mathbf{J}}_{\kappa^\pm}\right)_{kk}^a \tilde{C}_{n,a}^{(\kappa^\pm,k)}$ for $1 \le k \le d$
  end for
end for

▷ *Compute cross-face coefficients (4.37)*

for $m = 0, \dots, p_1$ **do**
  for $n = 0, \dots, p_2$ **do**
    for $q = 0, \dots, p_1 + p_2$ **do**
      for $a = \max\{0, q - m\}, \dots, \min\{q, n\}$ **do**
        $\tilde{C}_{m,n,q}^{(\kappa^\pm,k)} \leftarrow \tilde{C}_{m,n,q}^{(\kappa^\pm,k)} + C_{m,q-a} \tilde{C}_{n,a}^{(\kappa^\pm,k)}$ for $1 \le k \le d$
      end for
    end for
  end for
end for

4.2.4 Simultaneous computation of volume and face moments

For a given element $\kappa \in \mathcal{T}_\Omega$ and boundary face $e \in \partial\kappa$, the volume and face moments $m_{\boldsymbol{\alpha}}^{(v,\kappa,e)}$ and $m_{\boldsymbol{\alpha}}^{(f,\kappa,e)}$ defined in (4.32) and (4.34) may be computed simultaneously via a slight modification of Algorithm 1. An implementation for a two-dimensional element κ is given in Algorithm 4. Here, $p \in \mathbb{N}_0$ and $\mathbf{p} = (p_k)_{k=1}^d \in \mathbb{N}_0^d$ denote, respectively, the maximum total degree and maximum component-wise degree of any multi-index in the set $\mathcal{I}_\kappa = \mathcal{S}_{p,\mathbf{p}}$ defined in (4.5). Since we seek to evaluate integrals like $\int_\kappa \mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\beta}} \, \mathrm{d}\mathbf{x}$ with

$\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathcal{I}_\kappa$, we will define an intermediate set of multi-indices $\mathcal{J}_\kappa$ defined by

$$\mathcal{J}_\kappa = \{\boldsymbol{\alpha} + \boldsymbol{\beta} : \boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathcal{I}_\kappa\}, \tag{4.40}$$

For any $\boldsymbol{\alpha} \in \mathcal{J}_\kappa$, we have $0 \leq |\boldsymbol{\alpha}| \leq 2p$ and $0 \leq \boldsymbol{\alpha} \leq 2\mathbf{p}$.

Remark. *The choice of the index set $\mathcal{I}_\kappa = \mathcal{S}_{p,\mathbf{p}}$ must relate to the local polynomial space $\mathbb{H}^p(\kappa)$ via the relation $\mathbb{H}^p(\kappa) = \operatorname{span}\{\mathbf{x}^{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathcal{I}_\kappa\}$. If $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$, then one may take $\mathcal{I}_\kappa = \mathcal{S}_{p,p\mathbf{1}}$ where $\mathbf{1} = (1, 1, \cdots, 1) \in \mathbb{N}_0^d$. If $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$, then one may take $\mathcal{I}_\kappa = \mathcal{S}_{dp,p\mathbf{1}}$.*

Algorithm 4 Quadrature-free assembly of volume moments $m_{\boldsymbol{\alpha}}^{(v,\kappa,e)}$ in (4.32) and face moments $m_{\boldsymbol{\alpha}}^{(f,\kappa,e)}$ in (4.34) with $\boldsymbol{\alpha} \in \mathcal{J}_\kappa$ corresponding to a given one-dimensional face $e \in \partial\kappa$ of a given two-dimensional element $\kappa \in \mathcal{T}_\Omega$.

▷ *Setup*

Get $\mathbf{J}_\kappa$ and $\mathbf{t}_\kappa$

Get endpoints $\mathbf{x}_1 = (x_1, y_1)$ and $\mathbf{x}_2 = (x_2, y_2)$ of e

$(\hat{x}_i, \hat{y}_i) = \hat{\mathbf{x}}_i \leftarrow \mathbf{J}_\kappa^{-1}(\mathbf{x}_i - \mathbf{t}_\kappa)$ for $i = 1, 2$ ▷ $\hat{e} = F_\kappa^{-1} e$

$d \leftarrow ||\hat{\mathbf{x}}_2 - \hat{\mathbf{x}}_1||$

Construct outward unit normal $\hat{\mathbf{n}}$ to $\hat{\kappa} = F_\kappa^{-1}(\kappa)$ on $\hat{e}$

$a \leftarrow \hat{\mathbf{x}}_1 \cdot \hat{\mathbf{n}}$

▷ *Assembly*

$V(-1 : \min\{2p, 2p_1\}, -1 : \min\{2p, 2p_2\}) \leftarrow 0$ ▷ Array containing $m_{\boldsymbol{\alpha}}^{(v,\kappa,e)}$

$F(-1 : \min\{2p, 2p_1\}, -1 : \min\{2p, 2p_2\}) \leftarrow 0$ ▷ Array containing $m_{\boldsymbol{\alpha}}^{(f,\kappa,e)}$

for $q_1 = 0, \ldots, \min\{2p, 2p_1\}$ **do**

  for $q_2 = 0, \ldots, \min\{2p - q_1, 2p_2\}$ **do**

    $q \leftarrow q_1 + q_2$

▷ *Compute $F(q_1, q_2) = \int_{\hat{e}} \hat{x}^{q_1} \hat{y}^{q_2} \, \mathrm{d}\hat{s}$ as in (4.2) and the single contribution of $\hat{e}$ to $V(q_1, q_2)$ as in (4.1)*

    $F(q_1, q_2) \leftarrow \frac{1}{1+q} \left[d\hat{x}_2^{q_1} \hat{y}_2^{q_2} + q_1 \hat{x}_1 F(q_1 - 1, q_2) + q_2 \hat{y}_1 F(q_1, q_2 - 1) \right]$

    $V(q_1, q_2) \leftarrow \frac{a}{2+q} F(q_1, q_2)$

  end for

end for

▷ *Finalise*

$V \leftarrow |\mathbf{J}_\kappa| \, V$ ▷ Applied to all array elements

$F \leftarrow ||\mathbf{J}_\kappa^{-\top} \hat{\mathbf{n}}|| \, |\mathbf{J}_\kappa| \, F$ ▷ Applied to all array elements

---

For the most general implementations, there are a number of noticeable differences between Algorithms 1 and 4. The first difference is that Algorithm 4 initially maps the element κ, as well as the face $e \in \partial\kappa$, on to the reference bounding box.

Secondly, the sum over faces in Algorithm 1 is replaced with a single contribution from the face e in Algorithm 4. Thus, the array V in Algorithm 4 does not correspond

to the integrals of (shifted and scaled) monomials over κ, but rather a contribution to such integrals. In fact, the array V in Algorithm 4 corresponds to the integrals of (shifted and scaled) monomials over the hyperpyramid formed by joining the vertices of e to the centroid of κ. Therefore, by calling Algorithm 4 for each $e \in \partial\kappa$ and summing up the arrays V, the resulting array corresponds to the integrals of (shifted and scaled) monomials over κ.

Finally, Algorithm 4 additionally returns an array F corresponding to the integrals of (shifted and scaled) monomials over the face e. In Algorithm 1, this was originally temporary storage required to store contributions to the full volume integral. This face integral array may be used for the quadrature-free assembly of the face terms outlined in Chapter 4.2.3.

4.3 Implementation and Analysis

In the previous section, we saw how the volume and face integrals arising from the DGFEM discretisation of the first-order, constant-coefficient transport equation can be rewritten as linear combinations of monomials integrated along the faces of the computational mesh. Our motivation for doing this was to decrease the total time taken to assemble the linear system of equations in the case where the underlying mesh consists of arbitrary polytopic elements. This removes the necessity of constructing specialised quadrature schemes on each polytopic element. We will now show that, under specific conditions, this does indeed decrease the assembly time of the system.

To this end, we will present pseudocode detailing the face-based implementation of a general discontinuous Galerkin finite element method with a system matrix defined by

$$(\mathbf{A})_{ij} = \sum_{\kappa \in \mathcal{T}_\Omega} \int_\kappa F_v(\phi_j, \phi_i) \ \mathrm{d}\mathbf{x} \tag{4.41}$$

$$+ \sum_{e \in \mathcal{F}_\Omega} \int_e \left[F_f^{++}(\phi_j^+, \phi_i^+) + F_f^{+-}(\phi_j^+, \phi_i^-) + F_f^{-+}(\phi_j^-, \phi_i^+) + F_f^{--}(\phi_j^-, \phi_i^-) \right] \ \mathrm{d}s, \tag{4.42}$$

where F_v, $F_f^{\pm\pm}$ and $F_f^{\pm\mp}$ denote integrands in the DGFEM discretisation of a PDE for a single field variable which admit monomial expansions of the form

$$F_v(\phi_j, \phi_i) = \sum_{\boldsymbol{\alpha} \geq \mathbf{0}} c_{\boldsymbol{\alpha}} \left[F_v(\phi_j, \phi_i) \right] \mathbf{x}^{\boldsymbol{\alpha}},$$

$$F_f^{\pm\pm}(\phi_j^\pm, \phi_i^\pm) = \sum_{\boldsymbol{\alpha} \geq \mathbf{0}} c_{\boldsymbol{\alpha}} \left[F_f^{\pm\pm}(\phi_j^\pm, \phi_i^\pm) \right] \mathbf{x}^{\boldsymbol{\alpha}},$$

$$F_f^{\pm\mp}(\phi_j^\pm, \phi_i^\mp) = \sum_{\boldsymbol{\alpha} \geq \mathbf{0}} c_{\boldsymbol{\alpha}} \left[F_f^{\pm\mp}(\phi_j^\pm, \phi_i^\mp) \right] \mathbf{x}^{\boldsymbol{\alpha}}.$$

Here, the function $c_{\boldsymbol{\alpha}}[\cdot]$ accepts multivariate polynomial functions as arguments and returns the coefficient of the monomial $\mathbf{x}^{\boldsymbol{\alpha}}$ in the corresponding monomial expansion.

For a boundary edge $e \in \mathcal{F}_\Omega^\partial = \mathcal{F}_\Omega^+ \cup \mathcal{F}_\Omega^-$ with $\mathcal{F}_\Omega^+ = \mathcal{F}_\Omega^+(\boldsymbol{\mu})$ and $\mathcal{F}_\Omega^- = \mathcal{F}_\Omega^-(\boldsymbol{\mu})$ as in (3.13) and (3.14) respectively, we remark that only one of the face integrands is required for assembly; however, we shall henceforth assume that exactly two of the integrals involving $F_f^{\pm\pm}$ and $F_f^{\pm\mp}$ will be evaluated as this simplifies the forthcoming analysis.

In light of assembling the full system matrix $\mathbf{A}$ in (4.41), we make the following assumptions on the integrands:

- The number of floating-point operations required to increment a real number by $\omega F_v(\phi_j(\mathbf{x}), \phi_i(\mathbf{x}))$ (i.e. to evaluate $\omega F_v(\phi_j(\mathbf{x}), \phi_i(\mathbf{x}))$ and add the result to another real number) is constant and denoted by $\mathtt{F}_v^q$.

- The number of floating-point operations required to increment a real number by $\omega F_f^{\pm\pm}(\phi_j^\pm(\mathbf{x}), \phi_i^\pm(\mathbf{x}))$ or $\omega F_f^{\pm\mp}(\phi_j^\pm(\mathbf{x}), \phi_i^\mp(\mathbf{x}))$ is constant and denoted by $\mathtt{F}_f^q$.

- The number of floating-point operations required to increment a real number by $m_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}}[F_v(\phi_j, \phi_i)]$ is constant and denoted by $\mathtt{F}_v^{qf}$, where $m_{\boldsymbol{\alpha}} = m_{\boldsymbol{\alpha}}^{(v,\kappa,e)}$ for a given $\kappa \in \mathcal{T}_\Omega$ and $e \in \partial\kappa$ is given by (4.32) and $c_{\boldsymbol{\alpha}}[F_v(\phi_j, \phi_i)]$ denotes the coefficient of $\mathbf{x}^{\boldsymbol{\alpha}}$ in the monomial expansion of $F_v(\phi_j, \phi_i)$.

- The number of floating-point operations required to increment a real number by $m_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}}[F_f^{\pm\pm}(\phi_j^\pm, \phi_i^\pm)]$ or $m_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}}[F_f^{\pm\mp}(\phi_j^\pm, \phi_i^\mp)]$ is constant and denoted by $\mathtt{F}_f^{qf}$, where $m_{\boldsymbol{\alpha}} = m_{\boldsymbol{\alpha}}^{(f,\kappa^\pm,e)}$ for a given $\kappa^\pm \in \mathcal{T}_\Omega$ and $e \in \partial\kappa^\pm$ is given by (4.34) and $c_{\boldsymbol{\alpha}}[F_f^{\pm\pm}(\phi_j^\pm, \phi_i^\pm)]$ (resp. $c_{\boldsymbol{\alpha}}[F_f^{\pm\mp}(\phi_j^\pm, \phi_i^\mp)]$) denotes the coefficient of $\mathbf{x}^{\boldsymbol{\alpha}}$ in the monomial expansion of $F_f^{\pm\pm}(\phi_j^\pm, \phi_i^\pm)$ (resp. $F_f^{\pm\mp}(\phi_j^\pm, \phi_i^\mp)$).

In addition to these assumptions on the weak form, we shall also make the following assumptions on the mesh and polynomial spaces:

- On each element $\kappa \in \mathcal{T}_\Omega$, we will place a polynomial basis $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$ (so that there are $\binom{p+d}{d}$ degrees of freedom on each element) or $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$ (so that there are $(p+1)^d$ degrees of freedom on each element).

- The number of faces each element κ has is denoted by f_κ, and is uniformly bounded; that is, there exists $f_{min}, f_{max} \in \mathbb{N}$ such that $f_{min} \leq f_\kappa \leq f_{\max}$ for all $\kappa \in \mathcal{T}_\Omega$.

- The mesh $\mathcal{T}_\Omega$ is conforming in the following sense. For each face $e \in \mathcal{F}_\Omega^\partial$, there exists exactly one element κ for which e is a face of κ. For each $e \in \mathcal{F}_\Omega^{int}$, there exists exactly two elements κ_1 and κ_2 for which e is a face of both elements. Note that, by this definition, it may be the case that two neighbouring elements may be separated by multiple faces and that adjacent faces may be coplanar.

- For the purposes of quadrature-based assembly, we will assume that:

– Each element $\kappa \in \mathcal{T}_\Omega$ can be decomposed into n_κ standard d-dimensional element types on which a quadrature scheme employing $r_v = r_v(p)$ quadrature points and weights exactly integrating polynomial functions of maximal degree $2p$ can be used. The quantity n_κ is also uniformly bounded; that is, there exists $n_{v,min}, n_{v,max} \in \mathbb{N}$ such that $n_{v,min} \leq n_\kappa \leq n_{v,max}$ for all $\kappa \in \mathcal{T}_\Omega$. Thus, a full quadrature scheme on κ will utilise $n_\kappa r_v$ quadrature points and weights.

– Each face $e \in \{\partial\kappa_i\}_{i=1}^{f_\kappa}$ for a given element $\kappa \in \mathcal{T}_\Omega$ can be decomposed into n_e standard $(d-1)$-dimensional element types on which a quadrature employing $r_f = r_f(p)$ quadrature points and weights exactly integrating polynomial functions of maximal degree $2p$ can be used. The quantity n_e is also uniformly bounded; that is, there exists $n_{f,min}, n_{f,max} \in \mathbb{N}$ such that $n_{f,min} \leq n_e \leq n_{f,max}$ for all $e \in \mathcal{F}_\Omega$. Thus, a full quadrature scheme on e will utilise $n_e r_f$ quadrature points and weights.

It will also prove useful to introduce some definitions. The *average number of faces per element* is defined by

$$\bar{f} = \frac{1}{|\mathcal{T}_\Omega|} \sum_{\kappa \in \mathcal{T}_\Omega} f_\kappa, \tag{4.43}$$

the *average number of volume subdomains per element* is defined by

$$\bar{n} = \frac{1}{|\mathcal{T}_\Omega|} \sum_{\kappa \in \mathcal{T}_\Omega} n_\kappa, \tag{4.44}$$

and the *average number of face subdomains per element* is defined by

$$\bar{m} = \frac{1}{|\mathcal{T}_\Omega|} \sum_{\kappa \in \mathcal{T}_\Omega} \sum_{i=1}^{f_\kappa} n_{\partial\kappa_i}, \tag{4.45}$$

where $\{\partial\kappa_i\}_{i=1}^{f_\kappa} \subset \mathcal{F}_\Omega$ denotes the set of faces of κ. Notice that $\bar{m} \geq \bar{f}$ with equality when $n_e = 1$ for all $e \in \mathcal{F}_\Omega$.

Proposition 4.3.1. *The number of elements in $\mathcal{T}_\Omega$ is related to the number of internal and boundary faces in $\mathcal{F}_\Omega$ in the following way:*

$$|\mathcal{T}_\Omega| = \frac{|\mathcal{F}_\Omega^\partial| + 2|\mathcal{F}_\Omega^{int}|}{\bar{f}}.$$

Proof. For each $\kappa \in \mathcal{T}_\Omega$, we have that

$$f_\kappa = |\partial\kappa \cap \mathcal{F}_\Omega^\partial| + |\partial\kappa \cap \mathcal{F}_\Omega^{int}|,$$

where $\partial\kappa = \{\partial\kappa_i\}_{i=1}^{f_\kappa}$ is understood as the set of faces of κ. Summing over each $\kappa \in \mathcal{T}_\Omega$, we have that

$$\sum_{\kappa \in \mathcal{T}_\Omega} f_\kappa = \sum_{\kappa \in \mathcal{T}_\Omega} |\partial\kappa \cap \mathcal{F}_\Omega^\partial| + \sum_{\kappa \in \mathcal{T}_\Omega} |\partial\kappa \cap \mathcal{F}_\Omega^{int}|.$$

Notice that each face in $\mathcal{F}_\Omega^\partial$ appears exactly once in the first sum on the right-hand side, and that each face in $\mathcal{F}_\Omega^{int}$ appears exactly twice in the second sum on the right-hand-side. Using the definition of $\bar{f}$, we can write the equation above as

$$\bar{f}|\mathcal{T}_\Omega| = |\mathcal{F}_\Omega^\partial| + 2|\mathcal{F}_\Omega^{int}|,$$

and the result follows on rearrangement. □

Corollary 4.3.1.1. *The average number of face subdomains per element can be written as*

$$\bar{m} = \frac{1}{|\mathcal{T}_\Omega|}\left(\sum_{e\in\mathcal{F}_\Omega^\partial} n_e + 2\sum_{e\in\mathcal{F}_\Omega^{int}} n_e\right).$$

4.3.1 Analysis of general quadrature-based assembly

Algorithm 5 Quadrature-based assembly of the matrix $\mathbf{A}$ in (4.41).

▷ *Load volume integrals*
for $\kappa \in \mathcal{T}_\Omega$ **do**
  $\mathbf{A}_\kappa \leftarrow$ `AssembleElementMatrix`(κ)
  Insert $\mathbf{A}_\kappa$ into $\mathbf{A}$
end for
▷ *Load face integrals*
for $e \in \mathcal{F}_\Omega^\partial$ **do**
  $\mathbf{A}_e^{++} \leftarrow$ `AssembleFaceMatrix`(e, κ^+, κ^+)
  Insert $\mathbf{A}_e^{++}$ into $\mathbf{A}$
end for
for $e \in \mathcal{F}_\Omega^{int}$ **do**
  for $s_1 \in \{+,-\}$ **do**
    for $s_2 \in \{+,-\}$ **do**
      $\mathbf{A}_e^{s_1 s_2} \leftarrow$ `AssembleFaceMatrix`$(e, \kappa^{s_1}, \kappa^{s_2})$
      Insert $\mathbf{A}_e^{s_1 s_2}$ into $\mathbf{A}$
    end for
  end for
end for

Algorithm 5 describes the standard quadrature-based assembly procedure required to load the matrix $\mathbf{A}$ in (4.41). The procedure loops over the elements in the mesh and assembles local elemental contributions $\mathbf{A}_\kappa$ which are then added into the correct rows and columns of the global matrix. Similar loops over the internal and boundary faces in the skeleton of the mesh are performed. In the former case, four matrix contributions $\mathbf{A}_e^{\pm\pm}$ and $\mathbf{A}_e^{\pm\mp}$ are computed and added to the global matrix. In the latter case, a single matrix contribution $\mathbf{A}_e^{++}$ is computed and added to the global matrix. For the purpose

Algorithm 5 (continued)

▷ *Compute volume contribution*

procedure $\mathbf{B} = \texttt{AssembleElementMatrix}(\kappa)$

  $(\mathbf{B})_{ij} \leftarrow 0$ for $1 \leq i, j \leq \dim \mathbb{H}^p(\kappa)$

▷ *Loop over entries of* $\mathbf{B}$

  for $i = 1, \ldots, \dim \mathbb{H}^p(\kappa)$ **do**

    for $j = 1, \ldots, \dim \mathbb{H}^p(\kappa)$ **do**

▷ *Loop over subdomains of* κ *and associated quadrature points/weights*

      for $k = 1, \ldots, n_\kappa$ **do**

        for $q = 1, \ldots, r_v$ **do**

          $(\mathbf{B})_{ij} \leftarrow (\mathbf{B})_{ij} + \omega_q^{(k)} F_v(\phi_j(\mathbf{x}_q^{(k)}), \phi_i(\mathbf{x}_q^{(k)}))$

        end for

      end for

    end for

  end for

end procedure

▷ *Compute face contribution*

procedure $\mathbf{B} = \texttt{AssembleFaceMatrix}(e, \kappa^{s_1}, \kappa^{s_2})$

  $(\mathbf{B})_{ij} \leftarrow 0$ for $1 \leq i \leq \dim \mathbb{H}^p(\kappa^{s_1})$ and $1 \leq j \leq \dim \mathbb{H}^p(\kappa^{s_2})$

▷ *Loop over entries of* $\mathbf{B}$

  for $i = 1, \ldots, \dim \mathbb{H}^p(\kappa^{s_1})$ **do**

    for $j = 1, \ldots, \dim \mathbb{H}^p(\kappa^{s_2})$ **do**

▷ *Loop over subdomains of* e *and associated quadrature points/weights*

      for $k = 1, \ldots, n_e$ **do**

        for $q = 1, \ldots, r_f$ **do**

          $(\mathbf{B})_{ij} \leftarrow (\mathbf{B})_{ij} + \omega_q^{(k)} F_f^{s_2 s_1}(\phi_j^{s_2}(\mathbf{x}_q^{(k)}), \phi_i^{s_1}(\mathbf{x}_q^{(k)}))$

        end for

      end for

    end for

  end for

end procedure

of simplifying the forthcoming analysis, we will assume that an additional matrix $\mathbf{A}_e^{\pm\mp}$ is assembled.

In order to assess the computational expense of a standard quadrature-based assembly procedure, we will perform a basic count of floating-point operations (FLOPs). We denote by $\texttt{FLOP}_v^q = \texttt{FLOP}_v^q(p)$ (resp. $\texttt{FLOP}_f^q = \texttt{FLOP}_f^q(p)$) the number of floating-point operations required to assemble all volume (resp. face) terms via quadrature. The number

of FLOPs required to assemble the volume terms is given by

$$\begin{aligned}\texttt{FLOP}_v^q(p) &= \sum_{\kappa\in\mathcal{T}_\Omega}\sum_{i=1}^{\dim\mathbb{H}^p(\kappa)}\sum_{j=1}^{\dim\mathbb{H}^p(\kappa)}\sum_{k=1}^{n_\kappa}\sum_{q=1}^{r_v}\mathtt{F}_v^q \\ &= r_v\mathtt{F}_v^q\left(\dim\mathbb{H}^p(\kappa)\right)^2\sum_{\kappa\in\mathcal{T}_\Omega}n_\kappa \\ &= \bar{n}r_v\mathtt{F}_v^q\left(\dim\mathbb{H}^p(\kappa)\right)^2|\mathcal{T}_\Omega|.\end{aligned}$$

For the face terms, we assume that a uniform polynomial degree of approximation p is employed on each $\kappa\in\mathcal{T}_\Omega$, so that $\dim\mathbb{H}^p(\kappa_1)=\dim\mathbb{H}^p(\kappa_2)$ for all $\kappa_1,\kappa_2\in\mathcal{T}_\Omega$. This simplifies the analysis of the number of FLOPs required to assemble the volume terms, which is given by

$$\begin{aligned}\texttt{FLOP}_f^q(p) &= 2\sum_{e\in\mathcal{F}_\Omega^\partial}\sum_{i=1}^{\dim\mathbb{H}^p(\kappa^+)}\sum_{j=1}^{\dim\mathbb{H}^p(\kappa^+)}\sum_{k=1}^{n_e}\sum_{q=1}^{r_f}\mathtt{F}_f^q \\ &\quad+4\sum_{e\in\mathcal{F}_\Omega^{int}}\sum_{i=1}^{\dim\mathbb{H}^p(\kappa^\pm)}\sum_{j=1}^{\dim\mathbb{H}^p(\kappa^\pm)}\sum_{k=1}^{n_e}\sum_{q=1}^{r_f}\mathtt{F}_f^q \\ &= 2r_f\mathtt{F}_f^q\left(\dim\mathbb{H}^p(\kappa^\pm)\right)^2\left(\sum_{e\in\mathcal{F}_\Omega^\partial}n_e+2\sum_{e\in\mathcal{F}_\Omega^{int}}n_e\right) \\ &= 2\bar{m}r_f\mathtt{F}_f^q\left(\dim\mathbb{H}^p(\kappa^\pm)\right)^2|\mathcal{T}_\Omega|.\end{aligned}$$

The total number of FLOPs required to compute all local element and face matrices in Algorithm 5 is denoted by $\texttt{FLOP}^q=\texttt{FLOP}^q(p)$ and given by

$$\begin{aligned}\texttt{FLOP}^q(p) &= \texttt{FLOP}_v^q(p)+\texttt{FLOP}_f^q(p) \\ &= \left(\dim\mathbb{H}^p(\kappa)\right)^2\left(\bar{n}r_v\mathtt{F}_v^q+2\bar{m}r_f\mathtt{F}_f^q\right)|\mathcal{T}_\Omega|. \qquad (4.46)\end{aligned}$$

It is useful to consider how (4.46) behaves as a function of the polynomial degree p. Since $\dim\mathbb{H}^p(\kappa)=O(p^d)$, $r_v=O(p^d)$ and $r_f=O(p^{d-1})$, we have that $\texttt{FLOP}^q=O(p^{3d})$ at leading order, and we remark that the assembly of the volume terms is the most expensive procedure in Algorithm 5.

4.3.2 Analysis of general quadrature-free-based assembly

Algorithm 6 describes a quadrature-free assembly procedure that requires only a loop over the faces in the skeleton of the mesh. For simplicity, we present pseudocode for a general model problem posed in two dimensions; however, our algorithmic analysis will be valid for a general problem posed in $d\geq 1$ dimensions. For each $e\in\mathcal{F}_\Omega$, two sets $M^{(v,\kappa^\pm,e)}$ and $M^{(f,\kappa^\pm,e)}$ of integrals for each of the elements $\kappa^\pm$ adjacent to e are defined by

$$M^{(v,\kappa^\pm,e)}=\left\{m_{\boldsymbol{\alpha}}^{(v,\kappa^\pm,e)}=\frac{a^{(\kappa^\pm,e)}|\mathbf{J}_{\kappa^\pm}|}{d+|\boldsymbol{\alpha}|}\int_{\hat{e}^\pm}\hat{\mathbf{x}}^{\boldsymbol{\alpha}}\ \mathrm{d}\hat{s}:\boldsymbol{\alpha}\in\mathcal{J}_\kappa\right\}, \qquad (4.47)$$

$$M^{(f,\kappa^\pm,e)}=\left\{m_{\boldsymbol{\alpha}}^{(f,\kappa^\pm,e)}=||\mathbf{J}_{\kappa^\pm}^{-\top}\hat{\mathbf{n}}_{\hat{e}^\pm}||\ |\mathbf{J}_{\kappa^\pm}|\int_{\hat{e}^\pm}\hat{\mathbf{x}}^{\boldsymbol{\alpha}}\ \mathrm{d}\hat{s}:\boldsymbol{\alpha}\in\mathcal{J}_\kappa\right\}, \qquad (4.48)$$

Algorithm 6 Quadrature-free-based assembly of the matrix $\mathbf{A}$ in (4.41). The general structure of the matrix assembly algorithm is valid for all spatial dimensions d, but the functions `AssembleOnDiagonal` and `AssembleOffDiagonal` must be specifically written for each given d - in this example, these functions are written under the assumption $d = 2$.

▷ *Load boundary face integrals*

for $e \in \mathcal{F}_\Omega^\partial$ **do**
  Compute $M^{(v,\kappa^+,e)}$ and $M^{(f,\kappa^+,e)}$
  $\mathbf{A}_e^{++} \leftarrow$ `AssembleOnDiagonal`$(e, \kappa^+, M^{(v,\kappa^+,e)}, M^{(f,\kappa^+,e)})$
  Insert $\mathbf{A}_e^{++}$ into $\mathbf{A}$
end for

▷ *Load interior face integrals*

for $e \in \mathcal{F}_\Omega^{int}$ **do**
  Compute $M^{(v,\kappa^+,e)}$ and $M^{(f,\kappa^+,e)}$
  Compute "cross-face" coefficients (e.g. Algorithm 3)
  $\mathbf{A}_e^{++} \leftarrow$ `AssembleOnDiagonal`$(e, \kappa^+, M^{(v,\kappa^+,e)}, M^{(f,\kappa^+,e)})$
  $\mathbf{A}_e^{--} \leftarrow$ `AssembleOnDiagonal`$(e, \kappa^-, M^{(v,\kappa^-,e)}, M^{(f,\kappa^-,e)})$
  $\mathbf{A}_e^{+-} \leftarrow$ `AssembleOffDiagonal`$(e, \kappa^+, \kappa^-, M^{(v,\kappa^+,e)}, M^{(f,\kappa^+,e)})$
  $\mathbf{A}_e^{-+} \leftarrow$ `AssembleOffDiagonal`$(e, \kappa^-, \kappa^+, M^{(v,\kappa^-,e)}, M^{(f,\kappa^-,e)})$
  Insert $\mathbf{A}_e^{s_1 s_2}$ into $\mathbf{A}$ for $s_1, s_2 \in \{+, -\}$
end for

where $a^{(\kappa^\pm,e)} = \hat{\mathbf{x}} \cdot \hat{\mathbf{n}}^\pm$ for any $\hat{\mathbf{x}} \in e$ and $\hat{\mathbf{n}}^\pm$ denotes the outward unit normal to $\kappa^\pm$ on e. Here, $\mathcal{J}_\kappa$ describes the monomial set defined in (4.40) and the integrals $m_{\boldsymbol{\alpha}}^{(v,\kappa^\pm,e)}$ and $m_{\boldsymbol{\alpha}}^{(f,\kappa^\pm,e)}$ (defined in (4.32) and (4.34) respectively) can be computed using Algorithm 4. Therefore, the sets $M^{(v,\kappa^\pm,e)}$ and $M^{(f,\kappa^\pm,e)}$ are the arrays V and F produced by Algorithm 4.

As in the quadrature-based assembly, four matrix contributions $\mathbf{A}_e^{\pm\pm}$ and $\mathbf{A}_e^{\pm\mp}$ are computed and added to the global matrix for interior faces $e \in \mathcal{F}_\Omega^{int}$, and only one matrix contribution $\mathbf{A}_e^{++}$ is computed and added to the global matrix for boundary faces $e \in \mathcal{F}_\Omega^\partial$. However, an additional volume contribution is incorporated into the on-diagonal matrices $\mathbf{A}_e^{\pm\pm}$. For the purposes of simplifying the forthcoming analysis, we will assume that an additional matrix $\mathbf{A}_e^{\pm\mp}$ is assembled.

As before, we will count the number of FLOPs in the face-based quadrature-free assembly procedure. We denote by $\texttt{FLOP}_{on}^{qf} = \texttt{FLOP}_{on}^{qf}(p)$ (resp. $\texttt{FLOP}_{off}^{qf} = \texttt{FLOP}_{off}^{qf}(p)$) the number of floating-point operations required to assemble the on-diagonal (resp. off-diagonal) terms via Algorithm 6. Furthermore, we will again assume that $\dim \mathbb{H}^p(\kappa_1) = \dim \mathbb{H}^p(\kappa_2)$ for all $\kappa_1, \kappa_2 \in \mathcal{T}_\Omega$. The number of FLOPs required to assemble the on-

Algorithm 6 (continued)

▷ *Compute on-diagonal face contribution*

procedure $\mathbf{B} = \texttt{AssembleOnDiagonal}(e, \kappa^s, M^{(v,\kappa^s,e)}, M^{(f,\kappa^s,e)})$

  $(\mathbf{B})_{ij} \leftarrow 0$ for $1 \leq i, j \leq \dim \mathbb{H}^p(\kappa^s)$

▷ *Loop over entries of* $\mathbf{B}$ *and retrieve corresponding multi-indices*

  for $i = 1, \ldots, \dim \mathbb{H}^p(\kappa^s)$ **do**

    Get $\boldsymbol{\alpha}^{(i)} = (\alpha_1^{(i)}, \alpha_2^{(i)})$

    for $j = 1, \ldots, \dim \mathbb{H}^p(\kappa^s)$ **do**

      Get $\boldsymbol{\alpha}^{(j)} = (\alpha_1^{(j)}, \alpha_2^{(j)})$

▷ *Loop over all monomials in integrands of volume and face integrals - one "for" loop for each independent variable*

        for $\alpha_1 = 0, \ldots, \alpha_1^{(i)} + \alpha_1^{(j)}$ **do**

          for $\alpha_2 = 0, \ldots, \alpha_2^{(i)} + \alpha_2^{(j)}$ **do**

▷ *Assemble partial volume integral and full face integral from constituent monomials*

$$(\mathbf{B})_{ij} \leftarrow (\mathbf{B})_{ij} + m_{\boldsymbol{\alpha}}^{(v,\kappa^s,e)} c_{\boldsymbol{\alpha}}[F_v(\phi_j^s, \phi_i^s)] + m_{\boldsymbol{\alpha}}^{(f,\kappa^s,e)} c_{\boldsymbol{\alpha}}[F_f^{ss}(\phi_j^s, \phi_i^s)]$$

          end for

        end for

      end for

  end for

end procedure

diagonal terms is given by

$$\begin{aligned}
\texttt{FLOP}_{on}^{qf}(p) = &\sum_{e \in \mathcal{F}_\Omega^\partial} \sum_{i=1}^{\dim \mathbb{H}^p(\kappa^+)} \sum_{j=1}^{\dim \mathbb{H}^p(\kappa^+)} \sum_{\mathbf{0} \leq \boldsymbol{\alpha} \leq \boldsymbol{\alpha}^{(i)} + \boldsymbol{\alpha}^{(j)}} \left(\texttt{F}_v^{qf} + \texttt{F}_f^{qf} \right) \\
&+ 2 \sum_{e \in \mathcal{F}_\Omega^{int}} \sum_{i=1}^{\dim \mathbb{H}^p(\kappa^\pm)} \sum_{j=1}^{\dim \mathbb{H}^p(\kappa^\pm)} \sum_{\mathbf{0} \leq \boldsymbol{\alpha} \leq \boldsymbol{\alpha}^{(i)} + \boldsymbol{\alpha}^{(j)}} \left(\texttt{F}_v^{qf} + \texttt{F}_f^{qf} \right) \\
= &\left(\texttt{F}_v^{qf} + \texttt{F}_f^{qf} \right) \left(|\mathcal{F}_\Omega^\partial| + 2|\mathcal{F}_\Omega^{int}| \right) \sum_{i=1}^{\dim \mathbb{H}^p(\kappa^+)} \sum_{j=1}^{\dim \mathbb{H}^p(\kappa^+)} \sum_{\mathbf{0} \leq \boldsymbol{\alpha} \leq \boldsymbol{\alpha}^{(i)} + \boldsymbol{\alpha}^{(j)}} 1 \\
= &\bar{f} \left(\texttt{F}_v^{qf} + \texttt{F}_f^{qf} \right) Q_d(p) |\mathcal{T}_\Omega|,
\end{aligned}$$

where we have used Proposition 4.3.1 and defined the dimensionally-dependent function $Q_d(p)$ by

$$Q_d(p) = \sum_{i=1}^{\dim \mathbb{H}^p(\kappa)} \sum_{j=1}^{\dim \mathbb{H}^p(\kappa)} \sum_{\mathbf{0} \leq \boldsymbol{\alpha} \leq \boldsymbol{\alpha}^{(i)} + \boldsymbol{\alpha}^{(j)}} 1 \tag{4.49}$$

for any $\kappa \in \mathcal{T}_\Omega$.

Algorithm 6 (continued)

▷ *Compute off-diagonal face contribution*

procedure $\mathbf{B} = \texttt{AssembleOffDiagonal}(e, \kappa^{s_1}, \kappa^{s_2}, M^{(v,\kappa^{s_1},e)}, M^{(f,\kappa^{s_1},e)})$

  $(\mathbf{B})_{ij} \leftarrow 0$ for $1 \le i \le \mathbb{H}^p(\kappa^{s_1})$ and $1 \le j \le \mathbb{H}^p(\kappa^{s_2})$

  ▷ *Loop over entries of* $\mathbf{B}$ *and retrieve corresponding multi-indices*

  for $i = 1, \dots, \dim \mathbb{H}^p(\kappa^{s_1})$ **do** ▷ Test functions supported on κ^{s_1}

    Get $\boldsymbol{\alpha}^{(i)} = (\alpha_1^{(i)}, \alpha_2^{(i)})$

    for $j = 1, \dots, \dim \mathbb{H}^p(\kappa^{s_2})$ **do** ▷ Trial functions supported on κ^{s_2}

      Get $\boldsymbol{\alpha}^{(j)} = (\alpha_1^{(j)}, \alpha_2^{(j)})$

      ▷ *Loop over all monomials in integrands of volume and face integrals - one "for" loop for each independent variable*

      for $\alpha_1 = 0, \dots, \alpha_1^{(i)} + \alpha_1^{(j)}$ **do**

        for $\alpha_2 = 0, \dots, \alpha_2^{(i)} + \alpha_2^{(j)}$ **do**

          ▷ *Assemble full face integral from constituent monomials using cross-face mapping coefficents from* κ^{s_2} *to* κ^{s_1}

          $(\mathbf{B})_{ij} \leftarrow (\mathbf{B})_{ij} + m_{\boldsymbol{\alpha}}^{(f,\kappa^{s_1},e)} c_{\boldsymbol{\alpha}} [F_f^{s_2 s_1}(\phi_j^{s_2}, \phi_i^{s_1})]$

        end for

      end for

    end for

  end for

end procedure

Similarly, the number of FLOPs required to assemble the off-diagonal terms is given by

$$\begin{aligned}
\texttt{FLOP}_{off}^{qf}(p) = &\sum_{e \in \mathcal{F}_\Omega^\partial} \sum_{i=1}^{\dim \mathbb{H}^p(\kappa^+)} \sum_{j=1}^{\dim \mathbb{H}^p(\kappa^+)} \sum_{\mathbf{0} \le \boldsymbol{\alpha} \le \boldsymbol{\alpha}^{(i)} + \boldsymbol{\alpha}^{(j)}} \mathtt{F}_f^{qf} \\
&+ 2 \sum_{e \in \mathcal{F}_\Omega^{int}} \sum_{i=1}^{\dim \mathbb{H}^p(\kappa^\pm)} \sum_{j=1}^{\dim \mathbb{H}^p(\kappa^\pm)} \sum_{\mathbf{0} \le \boldsymbol{\alpha} \le \boldsymbol{\alpha}^{(i)} + \boldsymbol{\alpha}^{(j)}} \mathtt{F}_f^{qf} \\
=&\mathtt{F}_f^{qf} \left(|\mathcal{F}_\Omega^\partial| + 2|\mathcal{F}_\Omega^{int}| \right) \sum_{i=1}^{\dim \mathbb{H}^p(\kappa^+)} \sum_{j=1}^{\dim \mathbb{H}^p(\kappa^+)} \sum_{\mathbf{0} \le \boldsymbol{\alpha} \le \boldsymbol{\alpha}^{(i)} + \boldsymbol{\alpha}^{(j)}} 1 \\
=&\bar{f} \mathtt{F}_f^{qf} Q_d(p) |\mathcal{T}_\Omega|.
\end{aligned}$$

The total number of FLOPs required to compute all on- and off-diagonal matrices in Algorithm 6 is denoted by $\texttt{FLOP}^{qf} = \texttt{FLOP}^{qf}(p)$ and given by

$$\begin{aligned}
\texttt{FLOP}^{qf}(p) &= \texttt{FLOP}_{on}^{qf}(p) + \texttt{FLOP}_{off}^{qf}(p) \\
&= \bar{f} \left(\mathtt{F}_v^{qf} + 2\mathtt{F}_f^{qf} \right) Q_d(p) |\mathcal{T}_\Omega|. \qquad (4.50)
\end{aligned}$$

As before, we consider how (4.50) behaves as a function of the polynomial degree p. All of the dependence of $\texttt{FLOP}^{qf}$ on p is contained in the function $Q_d(p)$, which can

be shown to be $O(p^{3d})$ when $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$ or $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$. The following lemma shows that $Q_d(p)$ cannot grow faster than $O(p^{3d})$ in these two cases.

Lemma 4.3.2. *The function $Q_d(p)$ in (4.49) satisfies $Q_d(p) = (p+1)^{3d}$ when $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$ and*

$$Q_d(p) \leq \frac{(p+d)^{3d}}{2d!^2} \left(\frac{2}{d}\right)^d$$

when $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$.

Proof. We first remark that both polynomial spaces admit linearly-independent bases of monomial functions:

$$\mathbb{Q}^p(\kappa) = \operatorname{span}\left\{\mathbf{x}^{\boldsymbol{\alpha}} : \mathbf{0} \leq \boldsymbol{\alpha} \leq \mathbf{p}\right\},$$
$$\mathbb{P}^p(\kappa) = \operatorname{span}\left\{\mathbf{x}^{\boldsymbol{\alpha}} : 0 \leq |\boldsymbol{\alpha}| \leq p\right\},$$

and that

$$|\{\boldsymbol{\alpha} : \mathbf{0} \leq \boldsymbol{\alpha} \leq \mathbf{p}\}| = (p+1)^d,$$
$$|\{\boldsymbol{\alpha} : 0 \leq |\boldsymbol{\alpha}| \leq p\}| = \binom{p+d}{d}.$$

Here, $\mathbf{p} = (p)_{k=1}^d \in \mathbb{N}_0^d$ denotes the multi-index of length d whose entries are all equal to p.

When $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$, we have

$$\begin{aligned}
Q_d(p) &= \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\mathbf{p}} \sum_{\mathbf{0}\leq\boldsymbol{\beta}\leq\mathbf{p}} \sum_{\mathbf{0}\leq\boldsymbol{\gamma}\leq\boldsymbol{\alpha}+\boldsymbol{\beta}} 1 \\
&= \sum_{\mathbf{0}\leq\boldsymbol{\alpha}\leq\mathbf{p}} \sum_{\mathbf{0}\leq\boldsymbol{\beta}\leq\mathbf{p}} \prod_{k=1}^{d} (1+\alpha_k+\beta_k) \\
&= \prod_{k=1}^{d} \left(\sum_{\alpha_k=0}^{p} \sum_{\beta_k=0}^{p} (1+\alpha_k+\beta_k) \right) \\
&= \left(\sum_{\alpha_k=0}^{p} \frac{1}{2}(p+1)\left(p+2\left(1+\alpha_k\right)\right) \right)^d \\
&= (p+1)^{3d}.
\end{aligned}$$

When $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$, we have

$$\begin{aligned}
Q_d(p) &= \sum_{0\leq|\boldsymbol{\alpha}|\leq p} \sum_{0\leq|\boldsymbol{\beta}|\leq p} \sum_{\mathbf{0}\leq\boldsymbol{\gamma}\leq\boldsymbol{\alpha}+\boldsymbol{\beta}} 1 \\
&= \sum_{a=0}^{p} \sum_{b=0}^{p} \sum_{|\boldsymbol{\alpha}|=a} \sum_{|\boldsymbol{\beta}|=b} \prod_{k=1}^{d} (1+\alpha_k+\beta_k).
\end{aligned}$$

Noting that

$$\begin{aligned}\left|\left\{\boldsymbol{\alpha}\in\mathbb{N}_0^d : |\boldsymbol{\alpha}| = a\right\}\right| &= \left|\left\{\boldsymbol{\alpha}\in\mathbb{N}_0^d : |\boldsymbol{\alpha}| \le a\right\}\right| - \left|\left\{\boldsymbol{\alpha}\in\mathbb{N}_0^d : |\boldsymbol{\alpha}| \le a-1\right\}\right| \\ &= \binom{a+d}{d} - \binom{a+d-1}{d} \\ &= \binom{a+d-1}{d-1}\end{aligned}$$

and invoking the arithmetic-geometric mean inequality for the innermost product:

$$\prod_{k=1}^{d}(1+\alpha_k+\beta_k) \le \left(\frac{1}{d}\sum_{k=1}^{d}(1+\alpha_k+\beta_k)\right)^d = \left(1+\frac{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|}{d}\right)^d,$$

we have

$$\begin{aligned}Q_d(p) &\le \sum_{a=0}^{p}\sum_{b=0}^{p}\sum_{|\boldsymbol{\alpha}|=a}\sum_{|\boldsymbol{\beta}|=b}\left(1+\frac{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|}{d}\right)^d \\ &= \sum_{a=0}^{p}\sum_{b=0}^{p}\binom{a+d-1}{d-1}\binom{b+d-1}{d-1}\left(1+\frac{a+b}{d}\right)^d \\ &\le \frac{1}{d^d(d-1)!^2}\sum_{a=0}^{p}\sum_{b=0}^{p}(a+d-1)^{d-1}(b+d-1)^{d-1}(a+b+d)^d.\end{aligned}$$

We shall further bound this sum from above by a double integral. We will first consider the case $d \ge 2$ and then show that the same inequality also holds for $d = 1$. For $d \ge 2$, we have

$$\begin{aligned}&\sum_{a=0}^{p}\sum_{b=0}^{p}(a+d-1)^{d-1}(b+d-1)^{d-1}(a+b+d)^d \\ &\quad\le \int_0^{p+1}\int_0^{p+1}(a+d-1)^{d-1}(b+d-1)^{d-1}(a+b+d)^d \ \mathrm{d}b\,\mathrm{d}a \\ &\quad= \int_{d-1}^{p+d}\int_{d-1}^{p+d}\alpha^{d-1}\beta^{d-1}(\alpha+\beta+2-d)^d \ \mathrm{d}\beta\,\mathrm{d}\alpha \\ &\quad\le \int_{d-1}^{p+d}\int_{d-1}^{p+d}\alpha^{d-1}\beta^{d-1}(\alpha+\beta)^d \ \mathrm{d}\beta\,\mathrm{d}\alpha \\ &\quad= \sum_{k=0}^{d}\binom{d}{k}\left(\int_{d-1}^{p+d}\alpha^{d+k-1} \ \mathrm{d}\alpha\right)\left(\int_{d-1}^{p+d}\beta^{2d-k-1} \ \mathrm{d}\beta\right) \\ &\quad\le \sum_{k=0}^{d}\binom{d}{k}\frac{(p+d)^{d+k}}{d+k}\cdot\frac{(p+d)^{2d-k}}{2d-k} \\ &\quad\le \frac{2^d(p+d)^{3d}}{2d^2}.\end{aligned}$$

Therefore, for $d \ge 2$, we have

$$Q_d(p) \le \frac{1}{d^d(d-1)!^2}\cdot\frac{2^d(p+d)^{3d}}{2d^2} = \frac{(p+d)^{3d}}{2d!^2}\left(\frac{2}{d}\right)^d.$$

Finally, we can explicitly compute $Q_1(p) = (p+1)^3$, and so the result of the lemma extends to all $d \ge 1$. □

Remark. *For the choice $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$, it has been validated for the cases $d = 1, 2, 3$ that we have*

$$Q_d(p) = \binom{p+d}{d}^2 q_d(p), \tag{4.51}$$

where the function $q_d(p)$ is a polynomial of degree d in p given by

$$\begin{aligned} q_1(p) &= p + 1, \\ q_2(p) &= \frac{1}{18}(7p^2 + 21p + 18), \\ q_3(p) &= \frac{1}{120}(p + 2)(11p^2 + 44p + 60). \end{aligned}$$

It is conjectured that (4.51) also holds for $d \geq 4$ for some sequence of polynomials $\{q_d(p)\}_{d \geq 4}$ with $\deg q_d(p) = d$.

4.4 Comparison of Assembly Procedures

We are now ready to compare the face-based and quadrature-free assembly procedure outlined by Algorithm 6 with the standard quadrature-based assembly procedure outlined by Algorithm 5. This is most straightforwardly demonstrated by comparing the number of floating-point operations required to assemble the system matrix. To this end, we define the following ratio:

$$\mathtt{r}(p) = \frac{\mathtt{FLOP}^{qf}(p)}{\mathtt{FLOP}^{q}(p)} = \frac{\bar{f}(\mathtt{F}_v^{qf} + 2\mathtt{F}_f^{qf})Q_d(p)}{(\dim \mathbb{H}^p(\kappa))^2(\bar{n}r_v(p)\mathtt{F}_v^q + 2\bar{m}r_f(p)\mathtt{F}_f^q)}. \tag{4.52}$$

The function $\mathtt{r}(p)$ denotes the fraction of floating-point operations required to execute the quadrature-free-based assembly procedure compared to the quadrature-based assembly procedure. It is assumed that both assembly procedures can be run on machines with identical floating-point operations per second (FLOPS), so that $\mathtt{r}(p)$ can be used as a measure of computational speed-up. However, the validity of using the function $\mathtt{r}(p)$ as an analytical tool is still potentially imprecise. For example, since we have only considered FLOP-counting arguments, $\mathtt{r}(p)$ will not incorporate effects due to array accesses, which may significantly skew the expected performance of both assembly procedures.

Due to low-level effects, such as instantiation of data structures and additional routines required for imposition of boundary conditions, (4.52) is only a reliable measure of performance improvement in the large-p limit. Denoting by $\mathtt{r}_\infty$ the limit of $\mathtt{r}(p)$ as $p \to \infty$, we have that

$$\mathtt{r}_\infty = \lim_{p \to \infty} \mathtt{r}(p) = \frac{\bar{f}}{\bar{n}} \cdot \frac{\mathtt{F}_v^{qf} + 2\mathtt{F}_f^{qf}}{\mathtt{F}_v^q} \cdot \lim_{p \to \infty} \frac{Q_d(p)}{r_v(p)(\dim \mathbb{H}^p(\kappa))^2}. \tag{4.53}$$

Since $\mathtt{r}_\infty$ denotes the ratio of floating-point operations performed between the quadrature-free and quadrature-based algorithms, values of $\mathtt{r}_\infty$ less than one indicate that the quadrature-free algorithm is generally faster than the quadrature-based

algorithm, and vice versa. Each of the isolated ratios plays an important role in the comparison of both algorithms:

- The ratio $\frac{\bar{f}}{\bar{n}}$ is dependent only on the underlying computational mesh and how one treats volume- and face-integrals on it. Under the assumptions outlined earlier, it is independent of the implementations of both assembly procedures. The existence of such a mesh-dependent quantity motivates the idea of seeking different assembly procedures tailored for given families of meshes.
- The ratio $\frac{\mathtt{F}_v^{qf}+2\mathtt{F}_f^{qf}}{\mathtt{F}_v^{q}}$ is dependent only on the implementations of both the quadrature-based and quadrature-free-based assembly procedures, and not on the underlying mesh. This quantity is problem-dependent, and so is also dimensionally-dependent. From a programming perspective, this ratio tells us to what extent code optimisation within the innermost `for` loops can boost the speed of quadrature-free-based assembly compared to quadrature-based assembly.
- The ratio $\frac{Q_d(p)}{r_v(p)(\dim \mathbb{H}^p(\kappa))^2}$, as well as the value of the limit $\lim_{p\to\infty} \frac{Q_d(p)}{r_v(p)(\dim \mathbb{H}^p(\kappa))^2}$, is dependent on the spatial dimension of the problem, as well as the polynomial spaces employed on each mesh element $\kappa \in \mathcal{T}_\Omega$. It is independent of the geometry of the mesh and the implementation of both assembly procedures. Notice that we have eliminated the quadrature-based face integral contribution in the denominator under the assumption that $r_f(p) \ll r_v(p)$ in the limit $p \to \infty$.

We will investigate each of these ratios in more depth to better understand the feasibility of the quadrature-free assembly method against standard quadrature-based asembly methods.

4.4.1 Mesh-dependent ratio

As remarked earlier, the ratio $\frac{\bar{f}}{\bar{n}}$ is a function of the mesh $\mathcal{T}_\Omega$ and possibly the face sets $\mathcal{F}_\Omega^\partial$ and $\mathcal{F}_\Omega^{int}$. Table 4.1 records the values of this ratio for a wide class of two-dimensional and three-dimensional mesh types. An explanation for each mesh type is given below.

Standard meshes The first class of meshes under study are the meshes that are deemed standard in most practical applications. These meshes consist of elements whose geometries are either all *simplicial* (i.e. triangular in two dimensions, or tetrahedral in three dimensions) or all *tensor-product* (i.e. rectangles in two dimensions, or cuboids in three dimensions). For these meshes, elements do not need to be subdivided in order to perform quadrature-based volume integrals, since exact quadrature rules exist for these element types. Furthermore, each element in the mesh has a constant number of faces, say f. Therefore, the ratio $\frac{\bar{f}}{\bar{n}}$ for these meshes is given simply by f.

Element type	$\bar{f}$	$\bar{n}$	$\frac{\bar{f}}{\bar{n}}$
Simplex (2D)	3	1	3
Tensor-product (2D)	4	1	4
Simplex (3D)	4	1	4
Tensor-product (3D)	6	1	6
Polygonal (centroid)	$\bar{f}$	$\bar{f}$	1
Polygonal (ear-clipping)	$\bar{f}$	$\bar{f}-2$	$\frac{\bar{f}}{\bar{f}-2}$

Table 4.1: Comparison of the mesh-dependent quantity $\frac{\bar{f}}{\bar{n}}$ for different two- and three-dimensional mesh types. For the (two-dimensional) polygonal mesh types, two methods for splitting the volume integral into simplicial elements are considered.

Polygonal meshes The second class of meshes (in two dimensions only) are those meshes whose elements are general simple polygons with no underlying fine structure. These elements have no "holes", but may otherwise be convex or non-convex. For the sake of presentation, we shall assume that the polygonal meshes we encounter are convex; this is true, for example, in the case that the mesh is the restriction of a Voronoi tesselation to the given domain geometry, as is true for meshes generated by `PolyMesher` [95]. For an element $\kappa \in \mathcal{T}_\Omega$ of such a mesh with f_κ faces, there are a couple of ways to triangulate κ for the purposes of quadrature-based volume integration. The simplest is to join each vertex of κ to its centroid, yielding a triangulation consisting of $n_\kappa = f_\kappa$ triangles; a more sophisticated approach is to employ a so-called "ear-clipping" method which yields a triangulation consisting of $n_\kappa = f_\kappa - 2$ triangles (and can also be applied to non-convex elements). A linear-time algorithm for finding such a triangulation is given in [31].

Agglomerated meshes The final class of meshes under consideration are the family of *agglomerated* meshes. Denoted by $\mathcal{T}_\Omega = \mathcal{T}_\Omega^{agg}$, these meshes are formed from the agglomeration, or "joining together", of elements of a fine-mesh $\mathcal{T}_\Omega^{fine}$ consisting of standard (simplicial or tensor-product) element types to form polytopic elements. Such elements are often referred to as *agglomerated elements*, and are treated as unions of fine-mesh elements. Alternatively, $\mathcal{T}_\Omega^{agg}$ can be thought of as a partition of $\mathcal{T}_\Omega^{fine}$ into agglomerated elements $\{\tau\}_{\tau \in \mathcal{T}_\Omega^{agg}}$ such that each fine-mesh element $\kappa \in \mathcal{T}_\Omega^{fine}$ is contained in exactly one agglomerated element $\tau \in \mathcal{T}_\Omega^{agg}$. Such meshes can be generated by graph partitioning packages such as METIS [59], whereby the fine mesh is expressed as a graph whose vertices are mesh elements and whose edges are mesh faces between neighbouring elements. From Proposition 4.3.1, we can express $\bar{f}$ in terms of the number of elements and faces of $\mathcal{T}_\Omega^{agg}$ as

$$\bar{f} = \frac{|\mathcal{F}_\Omega^{\partial,agg}| + 2|\mathcal{F}_\Omega^{int,agg}|}{|\mathcal{T}_\Omega^{agg}|}.$$

The quantity $\bar{n}$ denotes the average number of subpartitions per *agglomerated* element on which an exact quadrature rule can be performed. Since the underlying fine-mesh elements constituting an agglomerated element suffice as a suitable subpartition, $\bar{n}$ can be reformulated in this context as

$$\bar{n} = \frac{|\mathcal{T}_\Omega^{fine}|}{|\mathcal{T}_\Omega^{agg}|}.$$

Thus, the ratio $\frac{\bar{f}}{\bar{n}}$ can be expressed in terms of agglomerated- and fine-mesh quantities as

$$\frac{\bar{f}}{\bar{n}} = \frac{|\mathcal{F}_\Omega^{\partial,agg}| + 2|\mathcal{F}_\Omega^{int,agg}|}{|\mathcal{T}_\Omega^{fine}|}. \tag{4.54}$$

Note that the face sets $\mathcal{F}_\Omega^{\partial,agg}$ and $\mathcal{F}_\Omega^{int,agg}$ contain *fine-mesh* faces that are present in the agglomerated mesh.

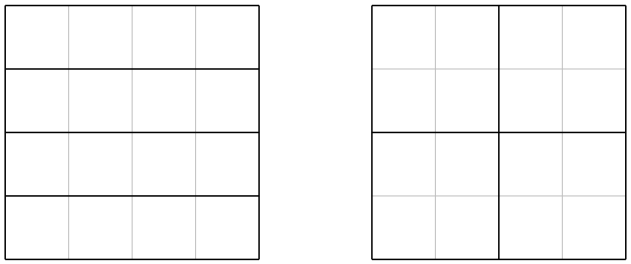

Figure 4.4: Two different agglomerated meshes with $|\mathcal{T}_\Omega^{agg}| = 4$ agglomerated elements formed from a fine mesh with $|\mathcal{T}_\Omega^{fine}| = 16$ square elements. Left: $\frac{\bar{f}}{\bar{n}} = \frac{5}{2}$. Right: $\frac{\bar{f}}{\bar{n}} = 2$.

Figure 4.4 shows how $\frac{\bar{f}}{\bar{n}}$ may depend on the agglomeration of the fine mesh. A two-dimensional tensor-product mesh $\mathcal{T}_\Omega^{fine}$ with $|\mathcal{T}_\Omega^{fine}| = 16$ square elements is agglomerated to two different agglomerated meshes $\mathcal{T}_\Omega^{agg}$ with $|\mathcal{T}_\Omega^{agg}| = 4$ agglomerated meshes. For the first mesh consisting of four 4×1 agglomerated elements, we have $\frac{\bar{f}}{\bar{n}} = \frac{5}{2}$, while for the second mesh consisting of four 2×2 agglomerated elements, we have $\frac{\bar{f}}{\bar{n}} = 2$. While both agglomerated meshes consist of the same number of elements, the quadrature-free-based implementation will assemble the DGFEM matrix faster on the second mesh since the corresponding value of $\frac{\bar{f}}{\bar{n}}$ is smaller. This is because Algorithm 6 will loop over fewer (internal) faces in the agglomerated mesh.

In view of minimising $\frac{\bar{f}}{\bar{n}}$, and thus improving the performance of quadrature-free-based assembly procedures relative to quadrature-based ones, the result above suggests that the agglomerated meshes for which quadrature-free-based methods work particularly well are those meshes for which $|\mathcal{F}_\Omega^{int,agg}|$ is small (the other mesh-based quantities above are fixed on selection of $\mathcal{T}_\Omega^{fine}$). The problem of determining an optimal agglomerated mesh $\mathcal{T}_\Omega^{agg}$ with $|\mathcal{T}_\Omega^{agg}| = k$ from a given fine-mesh $\mathcal{T}_\Omega^{fine}$ is therefore analogous to the problem of k-way graph partitioning. Here, the elements of $\mathcal{T}_\Omega^{fine}$ are interpreted as vertices of a graph and the faces between fine-mesh elements are interpreted as edges

between vertices in the graph. Such partitioning problems have wide-ranging applications and a number of methods for partitioning large graphs have been developed; [24] gives an overview of graph partitioning problems.

4.4.2 Implementation-dependent ratio

The problem- and implementation-specific ratio $\frac{\mathtt{F}_v^{qf}+2\mathtt{F}_f^{qf}}{\mathtt{F}_v^q}$ denotes the ratio of floating-point operations between the quadrature-free-based and quadrature-based assembly procedures. More specifically, $\mathtt{F}_v^q$, $\mathtt{F}_v^{qf}$ and $\mathtt{F}_f^{qf}$ denote, respectively, the number of floating-point operations required to increment a real number by $\omega F_v(\phi_j(\mathbf{x}), \phi_i(\mathbf{x}))$, $m_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}}[F_v(\phi_j, \phi_i)]$ and $m_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}}[F_f^{\pm\pm}(\phi_j^{\pm}, \phi_i^{\pm})]$ (or $m_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}}[F_f^{\pm\mp}(\phi_j^{\pm}, \phi_i^{\mp})]$). We will consider each term in the case where both methods are used to assemble the system matrix (4.8) in the DGFEM discretisation of the first-order, constant-coefficent transport equation.

- $\mathtt{F}_v^q$ denotes the number of floating-point operations required to increment a real number by the integrand of the volume integral in (4.10); that is, to increment the system matrix entry $(\mathbf{T})_{ij}$ by $\omega_q \phi_j(\mathbf{x}_q) \left(-\boldsymbol{\mu} \cdot \nabla\phi_i(\mathbf{x}_q) + b^{(\kappa)}\phi_i(\mathbf{x}_q)\right)$, where $(\omega_q, \mathbf{x}_q) \subset \mathbb{R}_{>0} \times \mathbb{R}^d$ denotes a quadrature point. Assuming that $\phi_j(\mathbf{x}_q)$, $\phi_i(\mathbf{x}_q)$ and $\nabla\phi_i(\mathbf{x}_q)$ are all precomputed, this operation can be computed using

  $$\mathtt{F}_v^q = 2(d+2)$$

  floating-point operations.

- $\mathtt{F}_v^{qf}$ denotes the number of floating-point operations required to increment a real number by the product of a monomial integral $m_{\boldsymbol{\alpha}}^{(v,\kappa,e)}$ (which are precomputed using Algorithm 4) and the corresponding coefficient in the monomial expansion of the volume integrand in (4.10). An example of this operation in two spatial dimensions is given in line 17 of Algorithm 2, but can be straightforwardly generalised to any number of dimensions. This operation can be computed using

  $$\mathtt{F}_v^{qf} = (d+1)^2 + 1$$

  floating-point operations.

- $\mathtt{F}_f^{qf}$ denotes the number of floating-point operations required to increment a real number by the product of a monomial integral $m_{\boldsymbol{\alpha}}^{(f,\kappa^{\pm},e^{\pm})}$ (which are precomputed using Algorithm 4) and the corresponding coefficient in the monomial expansion of the face integrand in (4.11). Since only half of the terms $F_f^{\pm\pm}(\phi_j^{\pm}, \phi_i^{\pm})$ and $F_f^{\pm\mp}(\phi_j^{\pm}, \phi_i^{\mp})$ are non-zero, we will instead take $\mathtt{F}_f^{qf}$ to be half of the number of floating-point operations required to assemble any non-zero term. Owing to our previous quadrature-free analysis of the face integrals, the increment to $(\mathbf{T})_{ij}$ takes the form of a product of $|\boldsymbol{\mu} \cdot \mathbf{n}|$ with a monomial integral $m_{\boldsymbol{\alpha}}^{(f,\kappa^{\pm},e^{\pm})}$ and either:

– a face-independent coefficient $C_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} = \prod_{k=1}^{d} C_{\alpha_k^{(i)},\alpha_k^{(j)},\alpha_k}$, if an on-diagonal block is being assembled; or

– a face-dependent coefficient $\tilde{C}^{\kappa^{\pm}}_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}} = \prod_{k=1}^{d} \tilde{C}^{(\kappa^{\pm},k)}_{\alpha_k^{(i)},\alpha_k^{(j)},\alpha_k}$ (e.g. from Algorithm 3), if an off-diagonal block is being assembled.

This operation can be computed using $d+2$ floating-point operations, and so we take

$$\mathtt{F}_f^{qf} = \frac{1}{2}(d+2).$$

Thus, we have the following expression for the $\frac{\mathtt{F}_v^{qf}+2\mathtt{F}_f^{qf}}{\mathtt{F}_v^{q}}$ as a function of the spatial dimension only:

$$\frac{\mathtt{F}_v^{qf}+2\mathtt{F}_f^{qf}}{\mathtt{F}_v^{q}} = \frac{d^2+3d+4}{2(d+2)} = \begin{cases} \frac{7}{4} & d=2, \\ \frac{11}{5} & d=3. \end{cases}$$

Note that the number of floating-point operations required to evaluate the inner-most loop in the quadrature-based assembly approach scales linearly with the spatial dimension, whereas the number of floating-point operations required to evaluate the inner-most loop in the quadrature-free-based assembly approach scales quadratically with the spatial dimension. Thus, we expect that the inner-most evaluations in the quadrature-free-based approach will be slower than those of the quadrature-based approach for DGFEMs applied to high-dimensional first-order constant-coefficient transport problems.

We stress that the ratio $\frac{\mathtt{F}_v^{qf}+2\mathtt{F}_f^{qf}}{\mathtt{F}_v^{q}}$ characterises the relative expense of the computations performed within the inner-most loops in the quadrature-based and quadrature-free-based methods, and the above analysis for their application to the transport equation does not assume that either method is fully optimised. For example, the loops over quadrature points, test and trial functions in the quadrature-based approach can be ordered in such a way that some floating-point operations required to increment an entry of the system matrix $\mathbf{T}$ can be moved outside the inner-most loop. Similar tricks can be employed in the quadrature-free-based approach as well - for example, the face-independent coefficients $C_{\boldsymbol{\alpha}^{(i)},\boldsymbol{\alpha}^{(j)},\boldsymbol{\alpha}}$ used in the on-diagonal face assembly procedure can be pre-computed offline and re-used for each face. Therefore, the value of this ratio may vary significantly depending on the efficiency of the assembly methods implemented.

4.4.3 Function space-dependent ratio

The ratio $\lim_{p\to\infty} \frac{Q_d(p)}{r_v(p)(\mathbb{H}^p(\kappa))^2}$ is dependent on the polynomial spaces $\mathbb{H}^p(\kappa)$ employed for each $\kappa \in \mathcal{T}_\Omega$ and the number of quadrature points $r_v(p)$ selected for each subdivision of κ. For convenience, we shall define the shorthand

$$R(p) = \frac{Q_d(p)}{r_v(p)(\mathbb{H}^p(\kappa))^2} \tag{4.55}$$

which we remark is an approximation of $\mathtt{r}(p)$ (given in (4.52)) obtained by discarding the lower-order contributions to the numerator and denominator (with respect to p).

We offer the following interpretation of $R(p)$. For a given edge e with neighbouring elements κ_1 and κ_2, let n^{qf} denote the number of times that the operation within the inner-most loops of Algorithm 6 (i.e. on lines 23 or 37) is executed. For a given element κ, let n^q denote the number of times that the operation within the inner-most loops of Algorithm 5 (i.e. on line 23) is executed. We then have that $R(p) = \frac{n^{qf}}{n^q}$.

Values of $R(p) < 1$ indicate that the quadrature-free assembly algorithm (Algorithm 6) requires fewer inner-most function evaluations (per face) than the quadrature-based assembly algorithm (Algorithm 5).

Henceforth, we will consider the following cases for $r_v(p)$:

- $r_v(p) = (p+1)^d$ - in this case, the quadrature scheme on each element subdivision is constructed by mapping a tensor-product quadrature scheme on the reference tensor-product element $(-1,1)^d$;
- $r_v(p) = \begin{pmatrix} p+d \\ d \end{pmatrix}$ - this is the theoretical lower bound for the size of a quadrature scheme that can exactly integrate polynomials of maximal degree $2p$ on an arbitrary d-dimensional polytope (see [92]), although such a minimal quadrature scheme is difficult to construct in practice [73].

We will consider two common choices of $\mathbb{H}^p(\kappa)$:

- $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$, the space of all polynomials of maximal total degree p on κ with $\dim \mathbb{H}^p(\kappa) = \begin{pmatrix} p+d \\ d \end{pmatrix}$;
- $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$, the space of all polynomials of maximal degree p in each independent variable on κ with $\dim \mathbb{H}^p(\kappa) = (p+1)^d$.

Table 4.2 shows the value of $\lim_{p\to\infty} R(p)$ under different choices of $\mathbb{H}^p(\kappa)$ and $r_v(p)$. For the choice $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$, the exact value of the desired limit for all $d \geq 1$ is reported; for the choice $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$, Lemma 4.3.2 is used to generate an upper bound on the desired limit that is valid for all $d \geq 1$.

	$\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$	$\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$
$r_v(p) = (p+1)^d$	1	$\leq \frac{1}{2}\left(\frac{2}{d}\right)^d$
$r_v(p) = \begin{pmatrix} p+d \\ d \end{pmatrix}$	$d!$	$\leq \frac{d!}{2}\left(\frac{2}{d}\right)^d$

Table 4.2: Values of $\lim_{p\to\infty} R(p)$ for different choices of $\mathbb{H}^p(\kappa)$ and $r_v(p)$. Note that the entries in the second column are all less than or equal to 1 for $d \geq 1$.

In each case, the ratio $\lim_{p\to\infty} R(p)$ exist and is finite, and represents the ratio of inner-most integrand calls made between the quadrature-free-based and quadrature-based assembly methods. This ratio is independent of both the mesh used to discretise

the spatial domain and the PDE problem under consideration. The smallest value of this ratio appears to be attained when both:

- the dimension of the space of polynomial basis functions $\mathbb{H}^p(\kappa)$ is minimised for a given degree p - note that, in order to ensure suitable approximation results [35], we must have $\mathbb{P}^p(\kappa) \subset \mathbb{H}^p(\kappa)$ and so $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$ represents the minimal choice for $\mathbb{H}^p(\kappa)$.

- the number of quadrature points $r_v(p)$ required to integrate polynomial functions of maximal degree $2p$ on the smallest standard subelements is maximised for a given degree p - note that any arbitrarily-large quadrature scheme may be selected, but for practical purposes the largest quadrature scheme one would employ on standard subelements contains no more than $r_v(p) = (p+1)^d$ quadrature points/weights.

Remark. *We can calculate the entries in the second column of Table 4.2 (i.e. the case $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$) exactly for $d = 1, 2, 3$ from the results of a previous remark about expressions for $Q_d(p)$ for these values of d. In this case, we have*

$$\lim_{p\to\infty} \frac{Q_d(p)}{r_v(p)(\dim \mathbb{H}^p(\kappa))^2} = \begin{cases} 1 & \text{if } d = 1, \\ \frac{7}{18} & \text{if } d = 2, \\ \frac{11}{120} & \text{if } d = 3, \end{cases}$$

when $r_v(p) = (p+1)^d$ and

$$\lim_{p\to\infty} \frac{Q_d(p)}{r_v(p)(\dim \mathbb{H}^p(\kappa))^2} = \begin{cases} 1 & \text{if } d = 1, \\ \frac{7}{9} & \text{if } d = 2, \\ \frac{11}{20} & \text{if } d = 3, \end{cases}$$

when $r_v(p) = \begin{pmatrix} p+d \\ d \end{pmatrix}$.

From Table 4.2, we note that $\lim_{p\to\infty} R(p) \leq 1$ independently of $r_v(p)$ whenever the finite element space for each $\kappa \in \mathcal{T}_\Omega$ is chosen as $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$, and conversely we have $\lim_{p\to\infty} R(p) \geq 1$ independently of $r_v(p)$ whenever the finite element space for each $\kappa \in \mathcal{T}_\Omega$ is chosen as $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$. This suggests that the quadrature-free assembly algorithm is more likely to yield faster assembly times (relative to the quadrature-based assembly algorithm) when the finite element space on each element is chosen as $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$. Moreover, Table 4.2 suggests that the speed-up in the assembly procedure using quadrature-free methods may improve for problems posed in higher dimensions.

We may also consider the behaviour of $R(p)$ for small values of p - this is displayed in Figure 4.5 for $d = 1, 2, 3$. Recall that the limit $p \to \infty$ was taken in (4.53) to compare

the most computationally-demanding processes in the quadrature-based and quadrature-free-based assembly methods. For small values of p, the quadrature-free integration of monomial functions, the computation of "cross-face" coefficients, the evaluation of test and trial functions at volume quadrature points, and the quadrature-based assembly of face terms arising in the DGFEM formulation all become significant in terms of computational complexity (relative to the assembly of volume terms). Indeed, these processes are not taken into account by the expression for $\mathtt{r}(p)$ in (4.52). Figure 4.5 shows that there is also variability in the ratio $R(p)$ of the leading-order costs of both methods for $d \geq 2$. Moreover, depending on the choices of d, $\mathbb{H}^p(\kappa)$ and $r_v(p)$, the ratio $R(p)$ may not even be close to its asymptotic limit for moderate values of p.

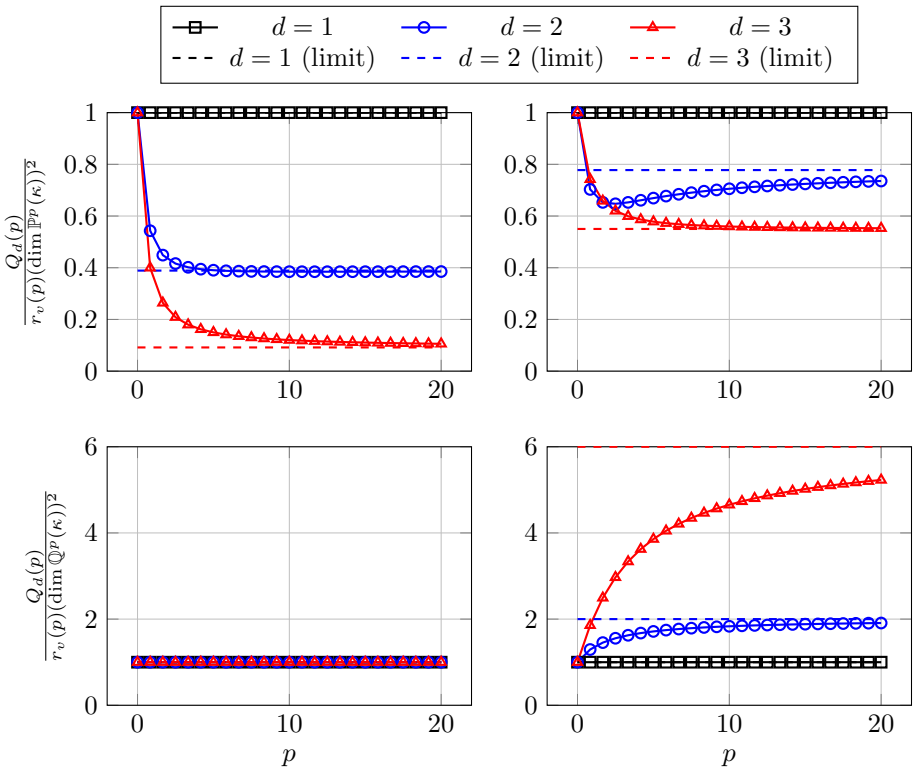

Figure 4.5: Plots of the rational function $R(p)$ and its limit as $p \to \infty$ for $d = 1, 2, 3$. Top row: $\mathbb{H}^p(\kappa) = \mathbb{P}^p(\kappa)$. Bottom row: $\mathbb{H}^p(\kappa) = \mathbb{Q}^p(\kappa)$. Left column: $r_v(p) = (p+1)^d$. Right column: $r_v(p) = \binom{p+d}{d}$.

Even when the optimal choice $r_v(p) = \binom{p+d}{d}$ is chosen for the size of the quadrature scheme employed on each subelement, the quadrature-free-based implementation can save on inner-most function executions compared to the quadrature-based implementation, provided that bases of $\mathbb{P}^p(\kappa)$ are employed on each spatial element.

4.5 Numerical Results

We shall now perform two sets of experiments showing the performance of the quadrature-based and quadrature-free-based methods for the assembly of the transport DGFEM matrix (4.8). For a given mesh $\mathcal{T}_\Omega$, we shall compute the mesh-dependent ratio $\frac{\bar{f}}{\bar{n}}$ and record the times $\mathtt{t}_q$ and $\mathtt{t}_{qf}$ taken by the quadrature-based and quadrature-free-based assembly methods respectively. The ratio $\frac{\mathtt{t}_{qf}}{\mathtt{t}_q}$ can then be interpreted as a measure of how much time can be saved by employing the quadrature-free-based method compared to the quadrature-based method. When the quadrature-free-based method is faster than the quadrature-based method, we have $\frac{\mathtt{t}_{qf}}{\mathtt{t}_q} < 1$. The ratio $\frac{\mathtt{t}_{qf}}{\mathtt{t}_q}$ is intended as a surrogate for the quantity $\mathtt{r}(p)$ defined in (4.52); for large enough p, $\frac{\mathtt{t}_{qf}}{\mathtt{t}_q}$ may also be used as a surrogate for the quantity $\mathtt{r}_\infty$ defined in (4.53).

When agglomerated meshes are used, we shall first construct a fine mesh $\mathcal{T}_\Omega^{fine}$ from which we form an agglomerated mesh $\mathcal{T}_\Omega^{agg}$. Interpreting the elements of $\mathcal{T}_\Omega^{fine}$ as vertices of a graph and the faces of $\mathcal{T}_\Omega^{fine}$ as edges of a graph, the graph-partitioning software METIS [59] is used to generate $\mathcal{T}_\Omega^{agg}$ with a user-specified number of agglomerated elements (or connected components of the underlying graph). In what follows, we always seek to select $|\mathcal{T}_\Omega^{agg}|$ to be an integer power of 2; however, METIS occasionally failed to generate such partitions.

4.5.1 Test 1 - Comparison on different mesh types

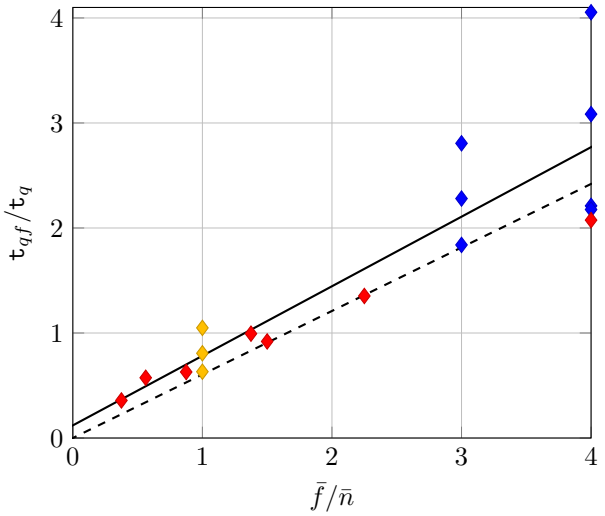

Figure 4.6: Time taken for quadrature-free-based assembly of the transport DGFEM matrix (4.8) as a fraction of the time taken for quadrature-based assembly, plotted against $\frac{\bar{f}}{\bar{n}}$. Blue: family of standard (simplicial/tensor-product) meshes. Yellow: family of polygonal meshes. Red: family of agglomerated mesh. Solid line: best fit line through data with gradient ≈ 0.6624 and intercept ≈ 0.1203. Dashed line: $\mathtt{r}_\infty$ as a function of $\frac{\bar{f}}{\bar{n}}$.

Figure 4.6 shows the dependency of the ratio of quadrature-free-based and quadrature-based CPU times taken to assemble the transport DGFEM matrix (4.8) against the ratio $\frac{\bar{f}}{\bar{n}}$ corresponding to a number of standard, polygonal and agglomerated meshes of varying number of elements. In every case, the spatial domain is given by $\Omega = (0,1)^2$, the wind direction is given by $\boldsymbol{\mu} = (1,1)$ and the reaction coefficient is given by $b = 1$. The families of meshes on Ω are defined as follows:

- For the standard mesh types (simplicial/tensor-product), the mesh $\mathcal{T}_\Omega$ denotes a regular mesh consisting of $|\mathcal{T}_\Omega| \in \{124, 512, 2048\}$ triangular elements or $|\mathcal{T}_\Omega| \in \{64, 256, 1024, 4096\}$ square elements.

- For the polygonal mesh types, the mesh $\mathcal{T}_\Omega$ denotes a non-nested polygonal mesh consisting of $|\mathcal{T}_\Omega| \in \{64, 256, 1024\}$ polygonal elements generated from PolyMesher [95]. For the purposes of volume quadrature, each polygonal element is sub-triangulated by joining each vertex to the barycentre of the element.

- For the agglomerated mesh types, the mesh $\mathcal{T}_\Omega = \mathcal{T}_\Omega^{agg}$ denotes a mesh formed from agglomerating a fine mesh $\mathcal{T}_\Omega^{fine}$ consisting of $|\mathcal{T}_\Omega^{fine}| = 8192$ triangular elements or $|\mathcal{T}_\Omega^{fine}| = 4096$ square elements. The graph-partitioning package METIS [59] is used to form the agglomerated mesh. The corresponding agglomerated meshes consist of $|\mathcal{T}_\Omega^{agg}| \in \{128, 512, 2048\}$ elements if $\mathcal{T}_\Omega^{fine}$ is a triangular mesh, or $|\mathcal{T}_\Omega^{agg}| \in \{64, 256, 1018, 4096\}$ elements if $\mathcal{T}_\Omega^{fine}$ is a square mesh.

The polynomial space employed on each $\kappa \in \mathcal{T}_\Omega^{agg}$ is chosen to be $\mathbb{P}^{16}(\kappa)$.

Figure 4.6 displays a strong linear correlation between the ratios $\frac{\mathtt{t}_{qf}}{\mathtt{t}_q}$, suggesting that the relative performance of the quadrature-based and quadrature-free-based assembly methods is dependent on the underlying meshes. On standard meshes, corresponding to $\frac{\bar{f}}{\bar{n}} = 3$ for triangular meshes and $\frac{\bar{f}}{\bar{n}} = 4$ for square meshes, the quadrature-free-based assembly method does not outperform the quadrature-based assembly method, since no further subdivision of elements into simplices was required by the latter method in order to perform numerical quadrature. However, on polygonal and agglomerated meshes, the quadrature-fee-based assembly method can offer savings in total assembly time compared to the quadrature-based assembly method. Recalling (4.54), small values of $\frac{\bar{f}}{\bar{n}}$ correspond to small numbers of internal faces in the agglomerated mesh (and thus small numbers of agglomerated elements in the mesh).

It should be noted that the coarsest standard and polygonal meshes (corresponding to $\frac{\bar{f}}{\bar{n}} \in \{1, 3, 4\}$) have the largest corresponding value of $\frac{\mathtt{t}_{qf}}{\mathtt{t}_q}$. Therefore, for sufficiently fine meshes of these classes, the ratio $\frac{\mathtt{t}_{qf}}{\mathtt{t}_q}$ reasonably approximates the ratio $\mathtt{r}_\infty$ defined in (4.53). This trend extends to agglomerated meshes, though $\frac{\mathtt{t}_{qf}}{\mathtt{t}_q}$ tends to deviate further from the predicted value of $\mathtt{r}_\infty$ for this class of meshes.

4.5.2 Test 2 - Comparison on different agglomeration sizes

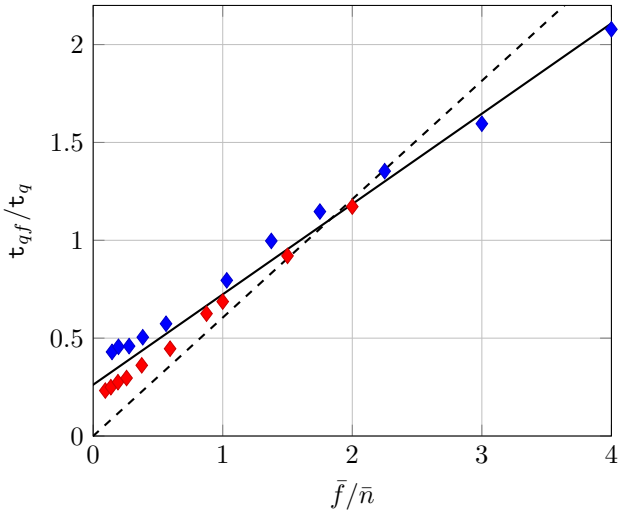

Figure 4.7: Time taken for quadrature-free-based assembly of the transport DGFEM matrix (4.8) as a fraction of the time taken for quadrature-based assembly, plotted against $\frac{\bar{f}}{\bar{n}}$. Blue: family of agglomerated meshes based on 64×64 square mesh. Red: family of agglomerated mesh based on 64×64 triangular mesh. Solid line: best fit line through data with gradient ≈ 0.4616 and intercept ≈ 0.2620. Dashed line: $\mathtt{r}_\infty$ as a function of $\frac{\bar{f}}{\bar{n}}$.

Figure 4.7 shows the dependency of the ratio of quadrature-free-based and quadrature-based CPU times taken to assemble the transport DGFEM matrix (4.8) against the ratio $\frac{\bar{f}}{\bar{n}}$ corresponding to two families of agglomerated meshes. In every case, the spatial domain $\Omega = (0,1)^2$ is first subdivided into a fine-mesh $\mathcal{T}_\Omega^{fine}$ consisting of either $|\mathcal{T}_\Omega^{fine}| = 8192$ triangular elements or $|\mathcal{T}_\Omega^{fine}| = 4096$ square elements. From these fine meshes, a number of agglomerated meshes $\mathcal{T}_\Omega^{agg}$ are generated using METIS, a graph-partitioning software - the number of elements in the generated meshes are:

- $|\mathcal{T}_\Omega^{agg}| \in \{8, 16, 32, 64, 128, 256, 512, 1024, 2048, 3950\}$ (when the underlying fine-mesh consists of triangular elements); and
- $|\mathcal{T}_\Omega^{agg}| \in \{4, 8, 16, 32, 64, 128, 256, 512, 1018, 1922, 4096\}$ (when the underlying fine-mesh consists of square elements).

The polynomial space employed on each $\kappa \in \mathcal{T}_\Omega^{agg}$ is chosen to be $\mathbb{P}^{16}(\kappa)$.

Figure 4.7 displays a strong linear correlation between the ratios $\frac{\mathtt{t}_{qf}}{\mathtt{q}_t}$ and $\frac{\bar{f}}{\bar{n}}$, suggesting that the relative performance of the quadrature-based and quadrature-free-based assembly methods is dependent on the underlying meshes. Moreover, Figure 4.7 confirms that the quadrature-free-based assembly outperforms the quadrature-based assembly on meshes with smaller values of $\frac{\bar{f}}{\bar{n}}$. We remark that, in this test, smaller values of $\frac{\bar{f}}{\bar{n}}$ correspond to coarser agglomerated meshes and larger values of $\frac{\bar{f}}{\bar{n}}$ correspond to finer

agglomerated meshes.

However, we do not observe that $\frac{\mathtt{t}_{qf}}{\mathtt{q}_t}$ is proportional to $\frac{\bar{f}}{\bar{n}}$, as is suggested by (4.53). We speculate that, on the coarsest agglomerated meshes (for which $\frac{\bar{f}}{\bar{n}}$ is small), the computational cost of assembling the individual block matrices in a quadrature-free fashion is outweighed by another process not accounted for in the analysis.

4.6 Summary

In this chapter, we have developed and analysed techniques for the assembly of linear systems arising from the application of DGFEMs to linear first-order partial differential equations with constant coefficients, with particular focus on employing meshes consisting of arbitrary polytopic elements. We have opted to assemble the DGFEM matrix using a quadrature-free approach as opposed to standard quadrature-based methods, which may become expensive when the elements of the underlying mesh have complicated geometries. The key idea is to decompose the integrand into a sum of homogeneous functions for which fast integration techniques can be developed. We started by reviewing homogeneous function integration with a particular focus on integrating families of monomial functions. This approach rewrites integrals over a polytopic domain as a sum of integrals over the planar boundary facets of the domain. Through a practical example, we showed that the proposed quadrature-free method was able to integrate sets of monomial integrals more rapidly than standard quadrature-based methods based on domain subtessellation.

Next, we focussed on applying the quadrature-free integration method to the problem of assembling DGFEM matrices for transport problems. By defining the polynomial basis functions on each polytopic element with respect to the element's bounding box, we were able to explicitly decompose pairwise-products of basis functions as linear combinations of monomial functions which may be integrated in a quadrature-free fashion. We proposed a quadrature-free assembly method that requires a single loop over each of the faces in the polytopic mesh and compared the time taken to assemble a DGFEM matrix with this approach against standard quadrature-based methods.

The face-based and quadrature-free method introduced in this chapter can be generalised to the assembly of DGFEM matrices from the discretisation of other problems. In an attempt to understand why quadrature-free-based assembly algorithms have been observed to outperform quadrature-based algorithms in terms of CPU time taken, an analysis of the number of floating-point operations required to assemble a general DGFEM matrix via both methods was performed. It was observed that there are three different factors that can affect the performance of quadrature-free methods (relative to quadrature-based methods): the geometry of the mesh, the polynomial spaces and quadrature methods employed, and the structure of the weak formulation of the PDE.

Our analysis suggests that quadrature-free-based methods generally outperform quadrature-based methods on non-standard meshes; i.e. meshes whose elements require further subdivision into simplicies and tensor-product subelements on which standard quadrature schemes can be employed. Conversely, quadrature-based methods should always be employed on standard meshes. Furthermore, when quadrature-free-based methods are to be employed, one should always employ polynomial bases on each element whose dimension is "minimal" with respect to the desired degree of approximation; i.e. one should always employ the function space $\mathbb{H}^{p_\kappa}(\kappa) = \mathbb{P}^{p_\kappa}(\kappa)$ for each element $\kappa \in \mathcal{T}_\Omega$.

The structure of the weak formulation of the PDE is clearly problem-dependent and we have not been able to demonstrate its effect on the assembly times of the quadrature-based and quadrature-free-based algorithms for a range of PDE problems, although we have provided an analysis for first-order transport problems. It is also expected that the extent to which this factor influences the time taken for quadrature-free-based methods to assemble DGFEM matrices is highly dependent on the implementation of the assembly algorithm.

Chapter 5

Iterative Solvers for the Linear Boltzmann Transport Equation

In Chapter 3, we derived a discontinuous Galerkin finite element method for the time-independent linear Boltzmann transport equation and gave a convergence result on the DGFEM-energy norm error $|||u - u_h|||_{DG}$ between the analytical solution u and the DGFEM approximation u_h. However, the problem for u_h is equivalent to the solution of a large and sparse linear system of equations for which direct solution methods are impractical. One must therefore turn to iterative methods which typically generate a sequence of approximate solutions $\{u_h^{(n)}\}_{n\geq 0}$ converging to u_h as $n \to \infty$.

Motivated by the optimisation of computational resources to quickly obtain good approximations of u, we may not always need to solve the discrete problem for u_h to a very high accuracy. For example, consider the following inequality between the analytical solution u, the exact DGFEM approximation u_h and the inexact DGFEM approximation $u_h^{(n)}$ generated by terminating an iterative solver after n iterations:

$$|||u - u_h^{(n)}|||_{DG} \leq \underbrace{|||u - u_h|||_{DG}}_{\text{discretisation error}} + \underbrace{|||u_h - u_h^{(n)}|||_{DG}}_{\text{solver error}}.$$

We can see that (in view of minimising the left-hand side) there is little point in minimising the solver error past the order of magnitude of the discretisation error - the sequence $\{u_h^{(n)}\}_{n\geq 0}$ is converging to a poor approximation of u. Thus, by comparing (estimates of) the solver and discretisation errors, we can determine whether we should take another step of the iterative solver or refine the computational mesh. If one can provide computable *a posteriori* error estimators for the discretisation and solver errors, this choice can be made automatically within an adaptive procedure.

This chapter will discuss the numerical solution of the linear system of equations

arising from discretisations of the linear Boltzmann transport equation, with particular emphasis on the mono-energetic form of the partial integro-differential equation. Rather than appealing to Fourier-analytical techniques to prove convergence results, we shall exploit the variational setting of our discrete methods, cf. [63]. We will first verify the convergence of a family of stationary iterative methods based on source iteration for the discretised mono-energetic problem, and that the convergence rate is characterised by the problem data. We will later prove that the classical source iteration method applied to the discretised poly-energetic problem is also convergent. In both cases, computable *a posteriori* error estimates will be presented.

Motivated by the development of more sophisticated Krylov subspace-based solvers, we investigate the spectral properties of the resulting discrete iteration operators in order to develop and test classes of preconditioners. A key challenge is that such preconditioners must be computationally inexpensive to implement and adaptable to a wide range of scattering models and optical thicknesses. In this work, we shall define the *optical thickness* or *cell aspect ratio* of a medium with respect to a given spatial mesh as the product

$$\varepsilon = (\alpha + \beta)h, \tag{5.1}$$

where $\alpha+\beta$ denotes the macroscopic total cross-section and h denotes the spatial mesh-size parameter - a motivation for this definition is given in Chapter 5.3.3. The effectiveness of these preconditioners is assessed qualitatively through model problems.

5.1 Introduction

While this chapter primarily concerns itself with iterative methods applied to the time-independent and mono-energetic linear Boltzmann transport equation, we shall initially consider the poly-energetic problem (3.4) from Chapter 3.1 for the fluence $u : \Omega\times\mathbb{S}\times\mathbb{Y} \to \mathbb{R}$ of a species of radiative particles travelling through a medium satisfying

$$\begin{aligned} \boldsymbol{\mu}\cdot\nabla_{\mathbf{x}} u(\mathbf{x},\boldsymbol{\mu},E) + (\alpha(\mathbf{x},\boldsymbol{\mu},E)+\beta(\mathbf{x},\boldsymbol{\mu},E))\,u(\mathbf{x},\boldsymbol{\mu},E) \\ = S[u](\mathbf{x},\boldsymbol{\mu},E) + f(\mathbf{x},\boldsymbol{\mu},E) \quad \text{in } \mathcal{D}, \\ u(\mathbf{x},\boldsymbol{\mu},E) = g(\mathbf{x},\boldsymbol{\mu},E) \quad \text{on } \Gamma_{in}, \end{aligned}$$

where the scattering operator $S[u]$ is defined by

$$S[u](\mathbf{x},\boldsymbol{\mu},E) = \int_{\mathbb{Y}}\int_{\mathbb{S}} \theta(\mathbf{x},\boldsymbol{\mu}'\to\boldsymbol{\mu},E'\to E)u(\mathbf{x},\boldsymbol{\mu}',E')\ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}E'.$$

The coefficient $\alpha(\mathbf{x},\boldsymbol{\mu},E) \geq 0$ denotes the macroscopic absorption cross-section and the coefficient $\beta(\mathbf{x},\boldsymbol{\mu},E)$ denotes the macroscopic absorption cross-section defined as in (2.2); that is, we have

$$\beta(\mathbf{x},\boldsymbol{\mu},E) = \int_{\mathbb{Y}}\int_{\mathbb{S}} \theta(\mathbf{x},\boldsymbol{\mu}\to\boldsymbol{\mu}',E\to E')\ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}E.$$

We henceforth assume that the medium is angularly isotropic. This allows the macroscopic cross-section to be written as $\alpha(\mathbf{x}, \boldsymbol{\mu}, E) = \alpha(\mathbf{x}, E)$ and the differential scattering cross-section to assume the form $\theta(\mathbf{x}, \boldsymbol{\mu} \to \boldsymbol{\mu}', E \to E') = \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E \to E')$; that is, the differential scattering cross-section depends on the cosine of the angle between the incoming and outgoing directions. This has the further consequence that $\beta(\mathbf{x}, \boldsymbol{\mu}, E) = \beta(\mathbf{x}, E)$.

Abstractly, we may introduce a "transport-plus-absorption" or "streaming" operator $\mathcal{L}$ and a "scattering" operator $\mathcal{S}$ whose actions on a space-angle-energy function $v(\mathbf{x}, \boldsymbol{\mu}, E)$ are given by

$$\begin{aligned}
\mathcal{L}v(\mathbf{x}, \boldsymbol{\mu}, E) &= (\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} + \alpha(\mathbf{x}, E) + \beta(\mathbf{x}, E))\, v(\mathbf{x}, \boldsymbol{\mu}, E), \\
\mathcal{S}v(\mathbf{x}, \boldsymbol{\mu}, E) &= \int_{\mathbb{Y}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}, E' \to E) v(\mathbf{x}, \boldsymbol{\mu}', E')\ \mathrm{d}\boldsymbol{\mu}'\, \mathrm{d}E',
\end{aligned}$$

and rewrite the LBTE in the following operator form:

$$\mathcal{L}u = \mathcal{S}u + f. \tag{5.2}$$

As remarked in Adams and Larsen [1] in the mono-energetic setting, the operator $\mathcal{L} - \mathcal{S}$ is difficult to invert in practice, since the scattering operator introduces a coupling over all pairs of angles and energies; this is also true for any discretisation of (5.2). The streaming operator $\mathcal{L}$, on the other hand, is generally much easier to invert; for instance, if a discrete ordinates method in angle and a multigroup method in energy are employed, the action of $\mathcal{L}$ can be computed on a "per-energy group" basis, where in each energy group a (large number of) first-order linear partial differential equations (PDEs) are solved, each with a constant wind direction.

In light of this observation, the classical *source iteration method* has formed the basis of a wide variety of numerical methods selected to solve the discrete equations. Starting from any initial guess of the fluence $u^{(0)}$ (a typical choice is $u^{(0)} = 0$), a sequence of fluence iterates $\{u^{(n)}\}_{n \geq 0}$ is constructed iteratively via the relation

$$\mathcal{L}u^{(n+1)} = \mathcal{S}u^{(n)} + f \qquad \text{for } n \geq 0. \tag{5.3}$$

In the limit $n \to \infty$, it is hoped that $u^{(n)} \to u$, where u denotes the analytical solution to (5.2). Denoting by $\alpha(\mathbf{x})$ and $\beta(\mathbf{x})$ the *mono-energetic* macroscopic absorption and scattering cross-sections, the iteration defined by (5.3) is guaranteed to converge for mono-energetic problems [1] provided that

$$\operatorname*{ess\,sup}_{\mathbf{x} \in \Omega} \frac{\beta(\mathbf{x})}{\alpha(\mathbf{x}) + \beta(\mathbf{x})} < 1.$$

An algebraically-equivalent and arguably easier-to-implement method can be ob-

tained by introducing an auxiliary sequence $\{r^{(n)}\}_{n\geq 0}$:

$$\begin{aligned} u^{(0)} &= r^{(0)} = f, \\ \mathcal{L}r^{(n+1)} &= \mathcal{S}r^{(n)} \qquad \text{for } n \geq 0, \\ u^{(n+1)} &= u^{(n)} + r^{(n+1)} \qquad \text{for } n \geq 0. \end{aligned}$$

From a mathematical perspective, source iteration is nothing more than a preconditioned Richardson iteration of the form

$$u^{(n+1)} = u^{(n)} + \omega\mathcal{P}^{-1}\left(f - (\mathcal{L} - \mathcal{S})u^{(n)}\right) \qquad \text{for } n \geq 0, \tag{5.4}$$

with the specific choice of preconditioner $\mathcal{P} = \mathcal{L}$ and relaxation parameter $\omega = 1$.

5.1.1 Discretisation

In Chapter 3.2, we presented a discontinuous Galerkin discretisation of the poly- and mono-energetic forms of the LBTE, and in Chapter 3.4 we introduced the so-called *discrete ordinates Galerkin* (DOG) method for the efficient implementation of the DGFEM scheme. While both methods lead to a system of equations of the form

$$\mathbf{Tu} = \mathbf{Su} + \mathbf{f}$$

where $\mathbf{u}$ is a vector of unknowns, the discrete ordinates Galerkin method resulted in a transport matrix $\mathbf{T}$ with a (sparser) block-diagonal structure. Specifically, each on-diagonal block matrix in $\mathbf{T}$ corresponded to a DGFEM matrix associated with a first-order linear hyperbolic PDE for a single ordinate direction. We then hinted that this equation may be solved approximately using a stationary iterative method of the form

$$\mathbf{Tu}^{(n+1)} = \mathbf{Su}^{(n)} + \mathbf{f}.$$

While the convergence of iterative methods is the focus of this chapter, we will *not* primarily focus on such methods applied to the DOG scheme. We instead opt to perform our analysis on the original DGFEM schemes in Chapter 3.2, although we stress that the following results can be extended to the DOG implementation; this will be discussed in Chapter 5.5.2.

For simplicity of presentation, we shall restate the poly-energetic DGFEM problem: find $u_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ such that

$$T(u_h, v_h) = S(u_h, v_h) + \ell(v_h) \tag{5.5}$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$, where

$$T(w_h, v_h) = \int_{\mathbb{Y}} \int_{\mathbb{S}} \Bigg(\sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Bigg(\int_{\kappa_\Omega} -w_h \boldsymbol{\mu} \cdot \nabla v_h + (\alpha + \beta) w_h v_h \ \mathrm{d}\mathbf{x} + \int_{\partial_+ \kappa_\Omega(\boldsymbol{\mu})} |\boldsymbol{\mu} \cdot \mathbf{n}| w_h^+ v_h^+ \ \mathrm{d}s - \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \setminus \partial_- \Omega(\boldsymbol{\mu})} |\boldsymbol{\mu} \cdot \mathbf{n}| w_h^- v_h^+ \ \mathrm{d}s \Bigg) \Bigg) \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E, \tag{5.6}$$

$$S(w_h, v_h) = \int_{\mathbb{Y}} \int_{\mathbb{S}} \int_{\Omega} \int_{\mathbb{Y}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}', E' \to E) \cdot w_h(\mathbf{x}, \boldsymbol{\mu}', E') v_h(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}E' \, \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E, \tag{5.7}$$

$$\ell(v_h) = \int_{\mathbb{Y}} \int_{\mathbb{S}} \int_{\Omega} f(\mathbf{x}, \boldsymbol{\mu}, E) v_h(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E + \int_{\mathbb{Y}} \int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}) \cap \partial_- \Omega(\boldsymbol{\mu})} g(\mathbf{x}, \boldsymbol{\mu}, E) v_h^+(\mathbf{x}, \boldsymbol{\mu}, E) \ \mathrm{d}s \, \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}E. \tag{5.8}$$

As was noted in Chapter 3.2.5, we may obtain discretisations of the mono-energetic LBTE by discarding the dependency of the solution and problem data on the energetic variables E and E' in the poly-energetic problem above.

Since we are unable to solve the full DGFEM problem (5.43) directly, we may employ a source iteration method to find an approximate solution. The full source iteration scheme thus reads as follows: given $u_h^{(0)} \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$, find $\{u_h^{(n)}\}_{n \geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ such that

$$T(u_h^{(n+1)}, v_h) = S(u_h^{(n)}, v_h) + \ell(v_h) \tag{5.9}$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$. On selection of an appropriate basis $\{\phi_i\}_{i=1}^N \subset \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$, $N = \dim \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$, we may define the matrices $\mathbf{T}, \mathbf{S} \in \mathbb{R}^{N \times N}$ and the vector $\mathbf{f} \in \mathbb{R}^N$ by

$$(\mathbf{T})_{ij} = T(\phi_j, \phi_i), \tag{5.10}$$

$$(\mathbf{S})_{ij} = S(\phi_j, \phi_i), \tag{5.11}$$

$$(\mathbf{f})_i = \ell(\phi_i), \tag{5.12}$$

and rewrite the source iteration method in the following linear algebraic form:

$$\mathbf{T}\mathbf{u}^{(n+1)} = \mathbf{S}\mathbf{u}^{(n)} + \mathbf{f}. \tag{5.13}$$

Here, $\mathbf{u}^{(n)}, \mathbf{u}^{(n+1)} \in \mathbb{R}^N$ denote the solution vectors for the coefficients of $u_h^{(n)}$ and $u_h^{(n+1)}$, respectively, for the basis $\{\phi_i\}_{i=1}^N$, i.e.,

$$u_h^{(n)} = \sum_{i=1}^N (\mathbf{u}_h^{(n)})_i \phi_i,$$

$$u_h^{(n+1)} = \sum_{i=1}^N (\mathbf{u}_h^{(n+1)})_i \phi_i.$$

The study of the convergence of (5.9) is deferred to Chapter 5.5.1.

We also point out the following useful connection between the functional and linear-algebraic forms of source iteration, which will be used throughout our analysis. Suppose

that $u_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ and $\mathbf{u}, \mathbf{v} \in \mathbb{R}^N$ are related by

$$u_h = \sum_{i=1}^{N} (\mathbf{u})_i \phi_i,$$
$$v_h = \sum_{i=1}^{N} (\mathbf{v})_i \phi_i;$$

then we have the following equivalences:

$$\mathbf{v}^* \mathbf{T} \mathbf{u} = T(u_h, v_h),$$
$$\mathbf{v}^* \mathbf{S} \mathbf{u} = S(u_h, v_h),$$
$$\mathbf{v}^* \mathbf{f} = \ell(v_h).$$

We briefly remark that the quantities $\mathbf{T}$, $\mathbf{S}$ and $\mathbf{f}$ are always understood to be real, and the vectors $\mathbf{u}$ and $\mathbf{v}$ are also assumed real whenever they correspond to functions in $\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ expanded in the basis described above - in this case, $\mathbf{v}^*$ refers to the transpose of $\mathbf{v}$. However, we will permit $\mathbf{u}$ and $\mathbf{v}$ to be complex when discussing (potentially complex) eigenvalues of the discrete iteration operators, although we will always decompose them into their real and imaginary parts in practice.

5.2 Error Analysis for Mono-Energetic Modified Source Iteration

We shall analyse a family of stationary iterative methods generalising the classical source iteration method for the solution of mono-energetic problems of the form

$$\begin{aligned} \boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}) + (\alpha(\mathbf{x}) + \beta(\mathbf{x}))\, u(\mathbf{x}, \boldsymbol{\mu}) & \\ = S[u](\mathbf{x}, \boldsymbol{\mu}) + f(\mathbf{x}, \boldsymbol{\mu}) \quad & \text{in } \mathcal{D}, \\ u(\mathbf{x}, \boldsymbol{\mu}) = g(\mathbf{x}, \boldsymbol{\mu}) \quad & \text{on } \partial\mathcal{D}, \end{aligned}$$

where the scattering operator $S[u]$ is defined by

$$S[u](\mathbf{x}, \boldsymbol{\mu}) = \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \cdot \boldsymbol{\mu}) u(\mathbf{x}, \boldsymbol{\mu}') \ \mathrm{d}\boldsymbol{\mu}'.$$

The mono-energetic LBTE can be written in the form (5.2), instead with the energetic dependence removed from u, the data α, β and θ, and the scattering operator. By our assumptions outlined in Chapter 3.2.5 we assume that the data α and β are functions of space only.

Introducing a parameter $\omega \in [0, 1]$ and subtracting $\omega\beta(\mathbf{x})u(\mathbf{x}, \boldsymbol{\mu})$ from both sides of (5.2), we arrive at the following (mathematically equivalent) operator equation:

$$\mathcal{L}_\omega u = \mathcal{S}_\omega u + f,$$

where the actions of the modified streaming and scattering operators $\mathcal{L}_\omega$ and $\mathcal{S}_\omega$, respectively, on a space-angle function $v(\mathbf{x}, \boldsymbol{\mu})$ are given by

$$\mathcal{L}_\omega v(\mathbf{x}, \boldsymbol{\mu}) = (\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} + \alpha(\mathbf{x}) + (1-\omega)\beta(\mathbf{x}))\, v(\mathbf{x}, \boldsymbol{\mu}),$$
$$\mathcal{S}_\omega v(\mathbf{x}, \boldsymbol{\mu}) = \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \cdot \boldsymbol{\mu}) v(\mathbf{x}, \boldsymbol{\mu}')\ \mathrm{d}\boldsymbol{\mu}' - \omega\beta(\mathbf{x}) v(\mathbf{x}, \boldsymbol{\mu}),$$

For an initial guess $u^{(0)}$, we construct a sequence of approximations to u according to the iteration

$$\mathcal{L}_\omega u^{(n+1)} = \mathcal{S}_\omega u^{(n)} + f \qquad \text{for } n \geq 0. \tag{5.14}$$

Note that the choice $\omega = 0$ reduces (5.14) to the classical source iteration (5.3).

In many important contexts, the differential scattering cross-section θ is a symmetric positive-semidefinite kernel satisfying Mercer's condition [86]: for all $v \in L^2(\mathbb{S})$ and $\mathbf{x} \in \Omega$, we have that

$$\int_{\mathbb{S}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') v(\boldsymbol{\mu}) v(\boldsymbol{\mu}')\ d\boldsymbol{\mu}' d\boldsymbol{\mu} \geq 0. \tag{5.15}$$

This turns out to be a desirable property when discussing the eigenvalues of the iteration matrix, as well as the forthcoming analysis. We shall assume that θ satisfies (5.15). By the definition of $\beta(\mathbf{x})$ in the mono-energetic setting (given in Chapter 3.2.5), (5.15) implies that $\beta(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in \Omega$.

5.2.1 Discretisation

We can recast the iteration (5.14) for the discretised mono-energetic LBTE in either a variational or linear algebraic form - for completeness, we present both forms. Introducing the bilinear form $M : \mathcal{V}_{\Omega,\mathbb{S}} \times \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ and weighted mass matrix $\mathbf{M} \in \mathbb{R}^{N \times N}$ defined for all $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ by

$$M(w_h, v_h) = \int_\Omega \int_{\mathbb{S}} \beta(\mathbf{x}) w_h(\mathbf{x}, \boldsymbol{\mu}) v_h(\mathbf{x}, \boldsymbol{\mu})\ d\boldsymbol{\mu} d\mathbf{x},$$
$$(\mathbf{M})_{ij} = M(\phi_j, \phi_i), \tag{5.16}$$

the modified source iteration method can be rewritten in the following functional and algebraic forms:

$$T(u_h^{(n+1)}, v_h) - \omega M(u_h^{(n+1)}, v_h) = S(u_h^{(n)}, v_h) - \omega M(u_h^{(n)}, v_h) + \ell(v_h) \qquad \text{for all } v_h \in \mathcal{V}_{\Omega,\mathbb{S}}, \tag{5.17}$$

$$(\mathbf{T} - \omega\mathbf{M})\, \mathbf{u}^{(n+1)} = (\mathbf{S} - \omega\mathbf{M})\, \mathbf{u}^{(n)} + \mathbf{f}. \tag{5.18}$$

We note that, in practice, (5.18) is no more difficult to implement than the standard source iteration method (5.13) - the matrix $\mathbf{T} - \omega\mathbf{M}$ (resp. $\mathbf{S} - \omega\mathbf{M}$) shares the same sparsity pattern as $\mathbf{T}$ (resp. $\mathbf{S}$) and the action of $(\mathbf{T} - \omega\mathbf{M})^{-1}$ (resp. $\mathbf{S} - \omega\mathbf{M}$) on a vector can be computed in an almost identical fashion as the action of $\mathbf{T}^{-1}$ (resp. $\mathbf{S}$) on a vector.

5.2.2 Analysis

We seek to show that the iteration defined by (5.17) (or equivalently (5.18)) is a contraction mapping on $\mathcal{V}_{\Omega,\mathbb{S}}$ and that the exact solution u_h is a fixed point of this contraction. Rather than proving convergence in the DG-energy norm (3.38) defined in Chapter 3.3, it is useful to define a family of norms $|||\cdot|||_{DG(\omega)} : \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ for $\omega \in [0,1]$ and every $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$:

$$\begin{aligned}|||v_h|||^2_{DG(\omega)} = \int_{\mathbb{S}}\int_{\Omega} (\alpha(\mathbf{x}) + (1-\omega)\beta(\mathbf{x}))|v_h(\mathbf{x},\boldsymbol{\mu})|^2 \ \mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu} \\ + \frac{1}{2}\int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \big(||v_h^+(\cdot,\boldsymbol{\mu}) - v_h^-(\cdot,\boldsymbol{\mu})||^2_{\partial_- \kappa_\Omega(\boldsymbol{\mu})\setminus\partial\Omega} \\ + ||v_h^+(\cdot,\boldsymbol{\mu})||^2_{\partial_- \kappa_\Omega(\boldsymbol{\mu})\cap\partial\Omega}\big) \ \mathrm{d}\boldsymbol{\mu}.\end{aligned}$$

Note that the family of $|||\cdot|||_{DG(\omega)}$-norms for $\omega \in [0,1]$ represents a generalisation of the DG-energy norm $|||\cdot|||_{DG}$; in particular, we have that $|||\cdot|||_{DG} = |||\cdot|||_{DG(1)}$.

The non-negativity of α and β, together with the restriction that α is bounded away from zero (in the mono-energetic case - see Chapter 3.2.5), suffice to prove that $|||\cdot|||_{DG(\omega)}$ is indeed a norm on $\mathcal{V}_{\Omega,\mathbb{S}}$. In fact, it is the natural norm in which to prove coercivity of the bilinear form $T - \omega M$; moreover, we have the following identity for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$:

$$|||v_h|||^2_{DG(\omega)} = T(v_h, v_h) - \omega M(v_h, v_h).$$

When analysing the modified source iteration with a particular choice of ω, we shall always use the $|||\cdot|||_{DG(\omega)}$-norm in our analysis. However, the following lemma tells us that these norms are actually equivalent for any choice of $\omega \in [0,1]$, provided that the *global scattering ratio*

$$c = \operatorname*{ess\,sup}_{\mathbf{x}\in\Omega} \left(\frac{\beta(\mathbf{x})}{\alpha(\mathbf{x}) + \beta(\mathbf{x})} \right) \tag{5.19}$$

satisfies $c < 1$.

Lemma 5.2.1 ($|||\cdot|||_{DG(\omega)}$-norm equivalence)**.** *Suppose $c < 1$. For $0 \le \omega_1 \le \omega_2 \le 1$ and any $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$, we have*

$$|||v_h|||_{DG(\omega_2)} \le |||v_h|||_{DG(\omega_1)} \le \sqrt{\frac{1-\omega_1 c}{1-\omega_2 c}}\ |||v_h|||_{DG(\omega_2)}.$$

Proof. For the first inequality, we remark that

$$\alpha + (1-\omega_2)\beta \le \alpha + (1-\omega_1)\beta$$

for $\omega_1 \le \omega_2$ and any $\alpha, \beta \ge 0$, and so we immediately have

$$|||v_h|||^2_{DG(\omega_2)} \le |||v_h|||^2_{DG(\omega_1)}.$$

For the second inequality, we define the *local scattering ratio* $\tilde{c}(\mathbf{x})$ by

$$\tilde{c}(\mathbf{x}) = \frac{\beta(\mathbf{x})}{\alpha(\mathbf{x}) + \beta(\mathbf{x})} \tag{5.20}$$

and consider the following:

$$\begin{aligned}
|||v_h|||^2_{DG(\omega_1)} &= \int_{\mathbb{S}}\int_{\Omega}(\alpha+(1-\omega_1)\beta)|v_h|^2 \ \mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu} + \underbrace{R(v_h)}_{\text{non-negative face contributions}} \\
&= \int_{\mathbb{S}}\int_{\Omega}\frac{\alpha+(1-\omega_1)\beta}{\alpha+(1-\omega_2)\beta}(\alpha+(1-\omega_2)\beta)|v_h|^2 \ \mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu} + R(v_h) \\
&= \int_{\mathbb{S}}\int_{\Omega}\frac{1-\omega_1\tilde{c}}{1-\omega_2\tilde{c}}(\alpha+(1-\omega_2)\beta)|v_h|^2 \ \mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu} + R(v_h) \\
&\leq \frac{1-\omega_1 c}{1-\omega_2 c}\int_{\mathbb{S}}\int_{\Omega}(\alpha+(1-\omega_2)\beta)|v_h|^2 \ \mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu} + R(v_h) \\
&\leq \frac{1-\omega_1 c}{1-\omega_2 c}\left(\int_{\mathbb{S}}\int_{\Omega}(\alpha+(1-\omega_2)\beta)|v_h|^2 \ \mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{\mu} + R(v_h)\right) \\
&= \frac{1-\omega_1 c}{1-\omega_2 c}\ |||v_h|||^2_{DG(\omega_2)}.
\end{aligned}$$

□

We shall now turn our attention to the bilinear forms appearing on the right-hand side of (5.17). The bilinear forms S and M are related through the differential scattering cross-section θ, as the following lemma shows.

Lemma 5.2.2. *The bilinear forms $S, M : \mathcal{V}_{\Omega,\mathbb{S}} \times \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ are symmetric and positive-semidefinite, and the following relationship holds for all $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$:*

$$|S(w_h, v_h)| \leq M(w_h, w_h)^{\frac{1}{2}} M(v_h, v_h)^{\frac{1}{2}}.$$

Proof. For any $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$, we have

$$\begin{aligned}
S(w_h, v_h) &= \int_{\Omega}\int_{\mathbb{S}}\int_{\mathbb{S}}\theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}')w_h(\mathbf{x}, \boldsymbol{\mu}')v_h(\mathbf{x}, \boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}\mathbf{x} \\
&= \int_{\Omega}\int_{\mathbb{S}}\int_{\mathbb{S}}\theta(\mathbf{x}, \boldsymbol{\mu}'\cdot\boldsymbol{\mu})v_h(\mathbf{x}, \boldsymbol{\mu})w_h(\mathbf{x}, \boldsymbol{\mu}') \ \mathrm{d}\boldsymbol{\mu}\,\mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}\mathbf{x} \\
&= S(v_h, w_h),
\end{aligned}$$

and a similar result can be shown for M. Positive-semidefiniteness of S follows from recalling that $\theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}')$ satisfies Mercer's condition for any $\mathbf{x} \in \Omega$. Positive-semidefiniteness of M follows from the non-negativity of β.

The first relationship between S and M follows from the Cauchy-Schwarz inequality and the connection between θ and β:

$$\begin{aligned}
S(w_h, v_h) &= \int_{\Omega}\int_{\mathbb{S}}\int_{\mathbb{S}}\theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}')w_h(\mathbf{x}, \boldsymbol{\mu}')v_h(\mathbf{x}, \boldsymbol{\mu}) \ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}\mathbf{x} \\
&\leq \left(\int_{\Omega}\int_{\mathbb{S}}\int_{\mathbb{S}}\theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}')w_h(\mathbf{x}, \boldsymbol{\mu}')^2 \ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}\mathbf{x}\right)^{\frac{1}{2}} \cdot \\
&\qquad \left(\int_{\Omega}\int_{\mathbb{S}}\int_{\mathbb{S}}\theta(\mathbf{x}, \boldsymbol{\mu}\cdot\boldsymbol{\mu}')v_h(\mathbf{x}, \boldsymbol{\mu})^2 \ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}\mathbf{x}\right)^{\frac{1}{2}} \\
&= \left(\int_{\Omega}\int_{\mathbb{S}}\beta(\mathbf{x})|w_h(\mathbf{x}, \boldsymbol{\mu}')|^2 \ \mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}\mathbf{x}\right)^{\frac{1}{2}}\left(\int_{\Omega}\int_{\mathbb{S}}\beta(\mathbf{x})|v_h(\mathbf{x}, \boldsymbol{\mu})|^2 \ \mathrm{d}\boldsymbol{\mu}\,\mathrm{d}\mathbf{x}\right)^{\frac{1}{2}} \\
&= M(w_h, w_h)^{\frac{1}{2}} M(v_h, v_h)^{\frac{1}{2}}.
\end{aligned}$$

□

Remark. *As a consequence of Lemma 5.2.2, we get that*

$$0 \leq S(v_h, v_h) \leq M(v_h, v_h)$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$.

Remark. *The relationship between* S *and* M *above is a restatement of Lemma 3.3.1 in the mono-energetic setting.*

The properties of symmetry and positive-semidefiniteness of S and M are inherited by the linear algebra quantities $\mathbf{S}$ and $\mathbf{M}$. However, the bilinear form $S - \omega M$ (or equivalently the matrix $\mathbf{S} - \omega\mathbf{M}$) is generally symmetric indefinite; that is, there exist $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ such that

$$S(w_h, w_h) - \omega M(w_h, w_h) < 0 < S(v_h, v_h) - \omega M(v_h, v_h).$$

This means it cannot necessarily induce a (semi)norm, although we have the following bound for all $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ and $\omega \in [0, 1]$ by naïve applications of the triangle and Cauchy-Schwarz inequalities:

$$S(w_h, v_h) - \omega M(w_h, v_h) \leq (1 + \omega)\, M(w_h, w_h)^{\frac{1}{2}} M(v_h, v_h)^{\frac{1}{2}}.$$

However, we will require a stronger "Cauchy-Schwarz-like" bound for the bilinear form $S(\cdot, \cdot) - \omega M(\cdot, \cdot)$. We shall work around this problem by invoking the following theorem by Horn and Johnson [51] regarding the simultaneous diagonalisation of two Hermitian positive-semidefinite matrices by congruence. We remark that a matrix $\mathbf{M} = (m_{ij})_{i,j=1}^{n} \in \mathbb{C}^{n\times n}$ is Hermitian if $m_{ij} = \bar{m}_{ji}$ for all $1 \leq i, j \leq n$; equivalently, $\mathbf{M}$ is Hermitian if $\mathbf{M} = \bar{\mathbf{M}}^{\top}$.

Theorem 5.2.3 (Thm. 7.6.4b)**.** *If* $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n\times n}$ *are Hermitian positive-semidefinite matrices and* $\operatorname{rank}(\mathbf{A}) = r$*, then there exists a nonsingular* $\mathbf{S} \in \mathbb{C}^{n\times n}$ *such that*

$$\mathbf{A} = \mathbf{S}(\mathbf{I}_r \oplus \mathbf{0}_{n-r})\mathbf{S}^*,$$
$$\mathbf{B} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^*,$$

where $\oplus$ *denotes the direct sum of matrices and* $\mathbf{\Lambda}$ *is a non-negative diagonal matrix with* $\operatorname{rank}(\mathbf{\Lambda}) = \operatorname{rank}(\mathbf{B})$.

We shall now resolve the problem of indefiniteness of $S - \omega M$ by proving the following "Cauchy-Schwarz-like" inequality.

Lemma 5.2.4. *For any* $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ *and* $\omega \in [0, 1]$*, we have that*

$$|S(w_h, v_h) - \omega M(w_h, v_h)| \leq r(\omega) M(w_h, w_h)^{\frac{1}{2}} M(v_h, v_h)^{\frac{1}{2}},$$

where $r(\omega) = \max\{\omega, 1 - \omega\}$.

Proof. We remark that it is sufficient to prove that

$$|\mathbf{v}^*(\mathbf{S} - \omega\mathbf{M})\mathbf{w}| \leq r(\omega)(\mathbf{w}^*\mathbf{M}\mathbf{w})^{\frac{1}{2}}(\mathbf{v}^*\mathbf{M}\mathbf{v})^{\frac{1}{2}}$$

for all $\mathbf{w}, \mathbf{v} \in \mathbb{R}^N$, owing to the connection between bilinear forms and matrix quantities. However, it shall be convenient to instead prove the inequality above for all $\mathbf{w}, \mathbf{v} \in \mathbb{C}^N$. Since $\mathbf{S}$ and $\mathbf{M}$ are both real, symmetric and positive-semidefinite, Theorem 5.2.3 can be applied. By that theorem, there exists a nonsingular $\mathbf{R} \in \mathbb{C}^{N\times N}$ such that

$$\begin{aligned}\mathbf{S} &= \mathbf{R}\mathbf{\Lambda}\mathbf{R}^*,\\ \mathbf{M} &= \mathbf{R}(\mathbf{I}_r \oplus \mathbf{0}_{N-r})\mathbf{R}^*\end{aligned}$$

where $r = \text{rank}\mathbf{M}$ and $\mathbf{\Lambda}$ is a non-negative diagonal matrix with $\text{rank}\mathbf{\Lambda} = \text{rank}\mathbf{S}$. Introducing the matrix $\mathbf{W} = \mathbf{R}^{-*} \in \mathbb{C}^{N\times N}$, we equivalently have

$$\begin{aligned}\mathbf{W}^*\mathbf{S}\mathbf{W} &= \mathbf{\Lambda},\\ \mathbf{W}^*\mathbf{M}\mathbf{W} &= \mathbf{I}_r \oplus \mathbf{0}_{N-r}.\end{aligned}$$

Let $\{\mathbf{w}_i\}_{i=1}^N \in \mathbb{C}^N$ denote the columns of $\mathbf{W}$. The above equalities show that this set of vectors is both $\mathbf{S}$- and $\mathbf{M}$-orthogonal, in the sense that $\mathbf{w}_j^*\mathbf{S}\mathbf{w}_i = \mathbf{w}_j^*\mathbf{M}\mathbf{w}_i = 0$ for $i \neq j$. Considering only the diagonal entries of $\mathbf{\Lambda}$ and $\mathbf{I}_r \oplus \mathbf{0}_{N-r}$, we have the following for $1 \leq i \leq N$:

$$\begin{aligned}\mathbf{w}_i^*\mathbf{S}\mathbf{w}_i &= \lambda_i,\\ \mathbf{w}_i^*\mathbf{M}\mathbf{w}_i &= \begin{cases}1 & 1 \leq i \leq r,\\ 0 & r+1 \leq i \leq N.\end{cases}\end{aligned}$$

By Lemma 5.2.2, we have that $0 \leq \lambda_i \leq 1$ for $1 \leq i \leq r$ and $\lambda_i = 0$ for $r+1 \leq i \leq N$. We can therefore write $\mathbf{\Lambda} = \tilde{\mathbf{\Lambda}} \oplus \mathbf{0}_{N-r}$, where $\tilde{\mathbf{\Lambda}} \in \mathbb{R}^{r\times r}$ is a non-negative diagonal matrix.

Now consider the product $\mathbf{M}\mathbf{W}\mathbf{\Lambda}$:

$$\begin{aligned}\mathbf{M}\mathbf{W}\mathbf{\Lambda} &= \mathbf{W}^{-*}(\mathbf{I}_r \oplus \mathbf{0}_{N-r})(\tilde{\mathbf{\Lambda}} \oplus \mathbf{0}_{N-r})\\ &= \mathbf{W}^{-*}\left((\mathbf{I}_r\tilde{\mathbf{\Lambda}}) \oplus (\mathbf{0}_{N-r}^2)\right)\\ &= \mathbf{W}^{-*}\left(\tilde{\mathbf{\Lambda}} \oplus \mathbf{0}_{N-r}\right)\\ &= \mathbf{W}^{-*}\mathbf{\Lambda}\\ &= \mathbf{S}\mathbf{W}.\end{aligned}$$

It follows that the vectors $\{\mathbf{w}_i\}_{i=1}^N \subset \mathbb{C}^N$ are generalised eigenvectors of the generalised eigenvalue problem

$$\mathbf{S}\mathbf{w}_i = \lambda_i\mathbf{M}\mathbf{w}_i$$

with corresponding generalised eigenvalues on the main diagonal of $\mathbf{\Lambda}$. Moreover, since $\mathbf{W}$ is nonsingular, these eigenvectors span $\mathbb{C}^N$. If we have $\mathbf{w}_i^*\mathbf{M}\mathbf{w}_i > 0$, then λ_i is given by

$$\lambda_i = \frac{\mathbf{w}_i^*\mathbf{S}\mathbf{w}_i}{\mathbf{w}_i^*\mathbf{M}\mathbf{w}_i} \in [0,1].$$

The statement that $\lambda_i \in [0,1]$ follows from Lemma 5.2.2. If, on the other hand, we have $\mathbf{w}_i^*\mathbf{M}\mathbf{w}_i = 0$, we can choose to define $\lambda_i = 0$ since we also have $\mathbf{w}_i^*\mathbf{S}\mathbf{w}_i = 0$. Therefore, the diagonal entries of $\mathbf{\Lambda}$ lie in the interval $[0,1]$.

We are now ready to prove the statement above. Since $\{\mathbf{w}_i\}_{i=1}^N$ spans $\mathbb{C}^N$, we can expand $\mathbf{w}$ and $\mathbf{v}$ in the generalised eigenvector basis:

$$\mathbf{w} = \sum_{i=1}^N \alpha_i \mathbf{w}_i,$$
$$\mathbf{v} = \sum_{i=1}^N \beta_i \mathbf{w}_i,$$

where $\alpha_i, \beta_i \in \mathbb{C}$ for $1 \leq i \leq N$. Evaluating the indefinite bilinear form directly, we have

$$\begin{aligned}
\mathbf{v}^*(\mathbf{S}-\omega\mathbf{M})\mathbf{w} &= \left(\sum_{j=1}^N \beta_j \mathbf{w}_j\right)^* (\mathbf{S}-\omega\mathbf{M}) \left(\sum_{i=1}^N \alpha_i \mathbf{w}_i\right) \\
&= \sum_{i,j=1}^N \alpha_i \beta_j^* (\mathbf{w}_j^*\mathbf{S}\mathbf{w}_i - \omega \mathbf{w}_j^*\mathbf{M}\mathbf{w}_i) \\
&= \sum_{i=1}^N \alpha_i \beta_i^* (\mathbf{w}_i^*\mathbf{S}\mathbf{w}_i - \omega \mathbf{w}_i^*\mathbf{M}\mathbf{w}_i) \\
&= \sum_{i=1}^N \alpha_i \beta_i^* (\lambda_i - \omega) \mathbf{w}_i^*\mathbf{M}\mathbf{w}_i.
\end{aligned}$$

We have used the fact that $\{\mathbf{w}_i\}_{i=1}^N$ are simultaneously $\mathbf{S}$- and $\mathbf{M}$-orthogonal and satisfy the aforementioned generalised eigenvalue problem. In view of obtaining the bound in the statement of the lemma, we have

$$\begin{aligned}
|\mathbf{v}^*(\mathbf{S}-\omega\mathbf{M})\mathbf{w}| &= \left|\sum_{i=1}^N \alpha_i \beta_i^* (\lambda_i - \omega) \mathbf{w}_i^*\mathbf{M}\mathbf{w}_i\right| \\
&\leq \sum_{i=1}^N |\alpha_i|\ |\beta_i|\ |\lambda_i - \omega|\ \mathbf{w}_i^*\mathbf{M}\mathbf{w}_i \\
&\leq \left(\max_{i=1}^N |\lambda_i - \omega|\right) \sum_{i=1}^N |\alpha_i|\ |\beta_i|\ \mathbf{w}_i^*\mathbf{M}\mathbf{w}_i \\
&\leq \left(\max_{0\leq\lambda\leq 1} |\lambda - \omega|\right) \left(\sum_{i=1}^N |\alpha_i|^2\ \mathbf{w}_i^*\mathbf{M}\mathbf{w}_i\right)^{\frac{1}{2}} \left(\sum_{i=1}^N |\beta_i|^2\ \mathbf{w}_i^*\mathbf{M}\mathbf{w}_i\right)^{\frac{1}{2}}
\end{aligned}$$

$$
\begin{aligned}
&= \max\{\omega, 1-\omega\} \left(\sum_{i,j=1}^{N} \alpha_i \alpha_j^* \ \mathbf{w}_j^* \mathbf{M} \mathbf{w}_i \right)^{\frac{1}{2}} \left(\sum_{i,j=1}^{N} \beta_i \beta_j^* \ \mathbf{w}_j^* \mathbf{M} \mathbf{w}_i \right)^{\frac{1}{2}} \\
&= \max\{\omega, 1-\omega\} \left(\left(\sum_{j=1}^{N} \alpha_j \mathbf{w}_j \right)^* \mathbf{M} \left(\sum_{i=1}^{N} \alpha_i \mathbf{w}_i \right) \right)^{\frac{1}{2}} \cdot \\
&\quad \left(\left(\sum_{j=1}^{N} \beta_j \mathbf{w}_j \right)^* \mathbf{M} \left(\sum_{i=1}^{N} \beta_i \mathbf{w}_i \right) \right)^{\frac{1}{2}} \\
&= \max\{\omega, 1-\omega\} (\mathbf{w}^* \mathbf{M} \mathbf{w})^{\frac{1}{2}} (\mathbf{v}^* \mathbf{M} \mathbf{v})^{\frac{1}{2}}.
\end{aligned}
$$

Here, we invoke the triangle inequality, the Cauchy-Schwarz inequality, the fact that each $\lambda_i \in [0,1]$ and the fact that the vectors $\{\mathbf{w}_i\}_{i=1}^N$ are $\mathbf{M}$-orthogonal. □

One more lemma is needed that relates the bilinear form M back to the natural norm $|||\cdot|||_{DG(\omega)}$ used in the proof of convergence of the modified source iteration method.

Lemma 5.2.5. *For any $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$, $\omega \in [0,1]$ and $c < 1$, we have*

$$
M(v_h, v_h) \leq \frac{c}{1-\omega c} |||v_h|||^2_{DG(\omega)}.
$$

Proof. The proof closely follows that of Lemma 5.2.1. Recalling the definition of $\tilde{c}$ in (5.20), we have

$$
\begin{aligned}
M(v_h, v_h) &= \int_{\mathbb{S}} \int_{\Omega} \beta |v_h|^2 \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \\
&= \int_{\mathbb{S}} \int_{\Omega} \frac{\beta}{\alpha + (1-\omega)\beta} (\alpha + (1-\omega)\beta) |v_h|^2 \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \\
&= \int_{\mathbb{S}} \int_{\Omega} \frac{\tilde{c}}{1-\omega\tilde{c}} (\alpha + (1-\omega)\beta) |v_h|^2 \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \\
&\leq \frac{c}{1-\omega c} \int_{\mathbb{S}} \int_{\Omega} (\alpha + (1-\omega)\beta) |v_h|^2 \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \\
&\leq \frac{c}{1-\omega c} |||v_h|||^2_{DG(\omega)}.
\end{aligned}
$$

□

We are finally ready to prove *a priori* and *a posteriori* error estimates for the modified source iteration by invoking Banach's fixed point theorem.

Theorem 5.2.6 (Convergence of MSI)**.** *The map $F : \mathcal{V}_{\Omega,\mathbb{S}} \to \mathcal{V}_{\Omega,\mathbb{S}}$ defined, for any $w_h \in \mathcal{V}_{\Omega,\mathbb{S}}$, $c < 1$ and $\omega \in \left[0, \frac{1}{2c}\right)$ as the solution to the variational problem*

$$
T(F(w_h), v_h) - \omega M(F(w_h), v_h) = S(w_h, v_h) - \omega M(w_h, v_h) + \ell(v_h) \tag{5.21}
$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ admits a unique fixed point $u_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ provided that $q < 1$, where

$$
q = \frac{r(\omega)c}{1-\omega c} = \frac{c \ \max\{\omega, 1-\omega\}}{1-\omega c}.
$$

Moreover, the sequence $\{u_h^{(n)}\}_{n\geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S}}$ defined by $u_h^{(n+1)} = F(u_h^{(n)})$ for $n \geq 0$ converges to u_h for any choice of $u_h^{(0)}$. We also have the following error reduction

formula, a priori *error estimate and* a posteriori *error estimate for the* $|||\cdot|||_{DG(\omega)}$*-norm solver error:*

$$
\begin{aligned}
|||u_h^{(n+1)} - u_h|||_{DG(\omega)} &\leq q\ |||u_h^{(n)} - u_h|||_{DG(\omega)}, \\
|||u_h^{(n+1)} - u_h|||_{DG(\omega)} &\leq \frac{q^n}{1-q}\ |||u_h^{(1)} - u_h^{(0)}|||_{DG(\omega)}, \\
|||u_h^{(n+1)} - u_h|||_{DG(\omega)} &\leq \frac{q}{1-q}|||u_h^{(n+1)} - u_h^{(n)}|||_{DG(\omega)}.
\end{aligned}
$$

Proof. We first remark that the mapping F is well-posed in the sense that the variational problem (5.21) for $F(w_h)$ is well-posed for any $w_h \in \mathcal{V}_{\Omega,\mathbb{S}}$. Let $w_1, w_2 \in \mathcal{V}_{\Omega,\mathbb{S}}$. We have

$$
\begin{aligned}
|||F(w_1) - F(w_2)|||^2_{DG(\omega)} &= T(F(w_1) - F(w_2), F(w_1) - F(w_2)) \\
&\quad - \omega M(F(w_1) - F(w_2), F(w_1) - F(w_2)) \\
&= S(w_1 - w_2, F(w_1) - F(w_2)) \\
&\quad - \omega M(w_1 - w_2, F(w_1) - F(w_2)).
\end{aligned}
$$

Invoking Lemmas 5.2.4 and 5.2.5, we get

$$
\begin{aligned}
|||F(w_1) - F(w_2)|||^2_{DG(\omega)} &\leq r(\omega)\ M(w_1 - w_2, w_1 - w_2)^{\frac{1}{2}} \cdot \\
&\qquad M(F(w_1) - F(w_2), F(w_1) - F(w_2))^{\frac{1}{2}} \\
&\leq \frac{r(\omega)c}{1-\omega c}\ |||w_1 - w_2|||_{DG(\omega)}|||F(w_1) - F(w_2)|||_{DG(\omega)}.
\end{aligned}
$$

On rearrangement, we obtain the following contractive property of F for all $w_1, w_2 \in \mathcal{V}_{\Omega,\mathbb{S}}$:

$$
|||F(w_1) - F(w_2)|||_{DG(\omega)} \leq \frac{r(\omega)c}{1-\omega c}\ |||w_1 - w_2|||_{DG(\omega)}.
$$

In order for F to be contractive, we require that $q = \frac{r(\omega)c}{1-\omega c} < 1$. For any given $0 \leq c < 1$, this is achieved if $0 \leq \omega < \frac{1}{2c}$. Under these assumptions, we have a contraction mapping on $\mathcal{V}_{\Omega,\mathbb{S}}$. Since $(\mathcal{V}_{\Omega,\mathbb{S}}, |||\cdot|||_{DG(\omega)})$ is a non-empty and complete metric space, Banach's fixed point theorem implies that F has a unique fixed point $u_h \in \mathcal{V}_{\Omega,\mathbb{S}}$, and that the sequence $\{u_h^{(n)}\}_{n\geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S}}$ defined by $u_h^{(n+1)} = F(u_h^{(n)})$ for $n \geq 0$ converges to u_h for any choice of $u_h^{(0)} \in \mathcal{V}_{\Omega,\mathbb{S}}$.

The proofs of the three error bounds are straightforward. The error reduction inequality is proven by the definition of the fixed point u_h and the relationship between consecutive terms in the sequence $\{u_h^{(n)}\}_{n\geq 0}$:

$$
|||u_h^{(n+1)} - u_h|||_{DG(\omega)} = |||F(u_h^{(n)}) - F(u_h)|||_{DG(\omega)} \leq q\ |||u_h^{(n)} - u_h|||_{DG(\omega)}.
$$

Applying the triangle inequality after one application of the error reduction inequality yields

$$
\begin{aligned}
|||u_h^{(n+1)} - u_h|||_{DG(\omega)} &\leq q\ |||u_h^{(n)} - u_h^{(n+1)} + u_h^{(n+1)} - u_h|||_{DG(\omega)} \\
&\leq q\left(|||u_h^{(n)} - u_h^{(n-1)}|||_{DG(\omega)} + |||u_h^{(n+1)} - u_h|||_{DG(\omega)}\right).
\end{aligned}
$$

The *a posteriori* error estimate follows on rearrangement. The *a priori* error estimate follows from applying the error reduction estimate n times, followed by one application of the *a posteriori* error estimate:

$$\begin{aligned}|||u_h^{(n+1)} - u_h|||_{DG(\omega)} &\leq q^n \; |||u_h^{(1)} - u_h|||_{DG(\omega)} \\ &\leq \frac{q^n}{1-q} \; |||u_h^{(1)} - u_h^{(0)}|||_{DG(\omega)}.\end{aligned}$$

□

We have proven that the family of methods (5.14) discretised using discontinuous Galerkin finite elements in the space-angle setting is convergent for all $\omega \in \left[0, \frac{1}{2c}\right)$, where $c < 1$ denotes the global scattering ratio. This was achieved by showing that the map between successive fluence approximations is a contraction with factor $q = \frac{r(\omega)c}{1-\omega c}$. For fixed $c < 1$, the contraction factor assumes its minimum value of $\frac{c}{2-c}$ at $\omega = \frac{1}{2}$. In contrast, when $\omega = 0$, (5.14) reduces to the classical source iteration method, and the contraction factor assumes the value of c - this agrees with the classical result in the infinite-medium setting [1].

By using Lemma 5.2.1 in conjunction with Theorem 5.2.6, it is possible to derive *a priori* and *a posteriori* error estimates in $|||\cdot|||_{DG(\omega)}$ norms for values of ω different from those employed in the iterative scheme. However, such bounds may lose sharpness by exploiting norm-equivalence. We instead focus on deriving a computable *a posteriori* error estimate in the $|||\cdot|||_{DG}$-norm, which we earlier remarked is identical to the $|||\cdot|||_{DG(1)}$-norm.

Theorem 5.2.7. *Let $\{u_h^{(n)}\}_{n\geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S}}$ be constructed as in Theorem 5.2.6 with a fixed value of $\omega \in \left[0, \frac{1}{2c}\right)$ and assume that the global scattering ratio $c < 1$. At the n^{th} modified source iteration, the DG-energy norm of the solver error $u_h^{(n)} - u_h$ satisfies*

$$|||u_h^{(n)} - u_h|||_{DG} \leq r(\omega)\sqrt{\frac{c}{1-\omega c}} \; ||\beta^{\frac{1}{2}}(u_h^{(n)} - u_h^{(n-1)})||_{L^2(\mathcal{D})}.$$

Proof. Letting $e_h^{(n)} = u_h^{(n)} - u_h$, we have

$$\begin{aligned}|||e_h^{(n)}|||_{DG}^2 &\leq T(e_h^{(n)}, e_h^{(n)}) - S(e_h^{(n)}, e_h^{(n)}) \\ &= \left[T(e_h^{(n)}, e_h^{(n)}) - \omega M(e_h^{(n)}, e_h^{(n)})\right] - \left[S(e_h^{(n)}, e_h^{(n)}) - \omega M(e_h^{(n)}, e_h^{(n)})\right] \\ &= \left[S(e_h^{(n-1)}, e_h^{(n)}) - \omega M(e_h^{(n-1)}, e_h^{(n)})\right] - \left[S(e_h^{(n)}, e_h^{(n)}) - \omega M(e_h^{(n)}, e_h^{(n)})\right] \\ &= S(u_h^{(n-1)} - u_h^{(n)}, e_h^{(n)}) - \omega M(u_h^{(n-1)} - u_h^{(n)}, e_h^{(n)}) \\ &\leq r(\omega) \; M(u_h^{(n-1)} - u_h^{(n)}, u_h^{(n-1)} - u_h^{(n)})^{\frac{1}{2}} M(e_h^{(n)}, e_h^{(n)})^{\frac{1}{2}} \\ &\leq r(\omega)\sqrt{\frac{c}{1-\omega c}} \; M(u_h^{(n-1)} - u_h^{(n)}, u_h^{(n-1)} - u_h^{(n)})^{\frac{1}{2}} |||e_h^{(n)}|||_{DG},\end{aligned}$$

where we have invoked Lemma 5.2.5. The *a posteriori* error bound is proven on rearrangement. □

Finally, we shall present the following residual-based *a posteriori* error estimate which is valid for any approximation of the solution of the discrete problem.

Theorem 5.2.8 (DG-energy norm *a posteriori* error bound, mono-energetic version)**.** *Define an inner product* $(\cdot,\cdot)_{L^2_w(\mathcal{D})} : \mathcal{V}_{\Omega,\mathbb{S}} \times \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ *and associated norm* $||\cdot||_{L^2_w(\mathcal{D})} : \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ *for all* $v_h, w_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ *by*

$$(w_h, v_h)_{L^2_w(\mathcal{D})} = \int_{\mathbb{S}}\int_{\Omega} \alpha(\mathbf{x}) w_h(\mathbf{x},\boldsymbol{\mu}) v_h(\mathbf{x},\boldsymbol{\mu})\, d\mathbf{x}d\boldsymbol{\mu},$$
$$||v_h||_{L^2_w(\mathcal{D})} = \sqrt{(v_h, v_h)_{L^2_w(\mathcal{D})}}.$$

Let $u_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ *be the exact solution to the variational problem*

$$T(u_h, v_h) = S(u_h, v_h) + \ell(v_h)$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$*, and* $\hat{u}_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ *denote any approximation of* u_h*. Then we have*

$$|||u_h - \hat{u}_h|||_{DG} \leq ||r_h||_{L^2_w(\mathcal{D})},$$

where $r_h = r_h(\hat{u}_h) \in \mathcal{V}_{\Omega,\mathbb{S}}$ *denotes the unique solution to the following variational problem for all* $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$*:*

$$(r_h(\hat{u}_h), v_h)_{L^2_w(\mathcal{D})} = \ell(v_h) - \left(T(\hat{u}_h, v_h) - S(\hat{u}_h, v_h)\right).$$

Proof. We have

$$\begin{aligned}
|||u_h - \hat{u}_h|||^2_{DG} &\leq T(u_h - \hat{u}_h, u_h - \hat{u}_h) - S(u_h - \hat{u}_h, u_h - \hat{u}_h) \\
&= \ell(u_h - \hat{u}_h) - (T(\hat{u}_h, u_h - \hat{u}_h) - S(\hat{u}_h, u_h - \hat{u}_h)) \\
&=: R(u_h - \hat{u}_h),
\end{aligned}$$

where $R : \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ denotes the *residual functional*. By the Riesz Representation Theorem, there exists a unique $r_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ satisfying

$$(r_h, v_h)_{L^2_w(\mathcal{D})} = R(v_h) \tag{5.22}$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$. Therefore, we have

$$\begin{aligned}
|||u_h - \hat{u}_h|||^2_{DG} &\leq R(u_h - \hat{u}_h) \\
&= (r_h, u_h - \hat{u}_h)_{L^2_w(\mathcal{D})} \\
&\leq ||r_h||_{L^2_w(\mathcal{D})} ||u_h - \hat{u}_h||_{L^2_w(\mathcal{D})} \\
&\leq ||r_h||_{L^2_w(\mathcal{D})} |||u_h - \hat{u}_h|||_{DG},
\end{aligned}$$

where we remark that $||v_h||_{L^2_w(\mathcal{D})} = ||\alpha^{\frac{1}{2}} v_h||_{L^2(\mathcal{D})} \leq |||v_h|||_{DG}$ for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$. Dividing both sides by $|||u_h - \hat{u}_h|||_{DG}$ retrieves the desired bound. □

We conclude this section by showing how the *a posteriori* error estimate can be evaluated in a linear algebraic setting. We first denote by $\mathbf{r}, \hat{\mathbf{u}} \in \mathbb{R}^N$, $N = \dim \mathcal{V}_{\Omega,\mathbb{S}}$, the vector of expansion coefficients of $r_h, u_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ in the basis $\{\phi_i\}_{i=1}^N$:

$$r_h = \sum_{j=1}^{N} (\mathbf{r})_j \phi_j, \tag{5.23}$$

$$\hat{u}_h = \sum_{j=1}^{N} (\hat{\mathbf{u}})_j \phi_j. \tag{5.24}$$

We note that the coefficient vector $\hat{\mathbf{u}}$ is assumed to be known. Inserting (5.23) and (5.24) into (5.22) and setting $v_h = \phi_i$ for $1 \leq i \leq N$ yields the following linear system of equations for the vector $\mathbf{r}$:

$$\mathbf{M}\mathbf{r} = \mathbf{f} - (\mathbf{T} - \mathbf{S})\hat{\mathbf{u}} =: \hat{\mathbf{r}},$$

where $\hat{\mathbf{r}} \in \mathbb{R}^N$ denotes the true residual vector induced by the approximation $\hat{\mathbf{u}}$ and the entries of the mass matrix $\mathbf{M} \in \mathbb{R}^{N \times N}$ are given by

$$(\mathbf{M})_{ij} = (\phi_j, \phi_i)_{L^2_w(\mathcal{D})}. \tag{5.25}$$

Moreover, the *a posteriori* error estimate can be written as

$$||r_h||_{L^2_w(\mathcal{D})} = \sqrt{(r_h, r_h)_{L^2_w(\mathcal{D})}} = \sqrt{\mathbf{r}^\top \mathbf{M} \mathbf{r}}.$$

Putting everything together, the *a posteriori* estimate can therefore be evaluated as

$$||r_h||_{L^2_w(\mathcal{D})} = \sqrt{\hat{\mathbf{r}}^\top \mathbf{M}^{-1} \hat{\mathbf{r}}}. \tag{5.26}$$

Remark. *The* a posteriori *error estimate in the statement of Theorem 5.2.8 is in fact not sharp. Indeed, a sharper estimate can be employed by replacing the $(\cdot,\cdot)_{L^2_w(\mathcal{D})}$-inner product with an inner product $(\cdot,\cdot)_{DG}$ induced by the DG-energy norm, defined for all $w_h, v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ by the polarisation identity*

$$(w_h, v_h)_{DG} = \frac{1}{4}\left(|||w_h + v_h|||^2_{DG} - |||w_h - v_h|||^2_{DG} \right).$$

However, the mass matrix induced by the inner product $(\cdot,\cdot)_{DG}$ has a slightly denser structure than the mass matrix induced by the inner product $(\cdot,\cdot)_{L^2_w(\mathcal{D})}$. In the latter case, the mass matrix is block-diagonal, with each on-diagonal block matrix corresponding to a space-angle-energy element. In the former case, however, the mass matrix has additional off-diagonal blocks due to the additional face terms present between neighbouring spatial elements. Therefore, the application of the mass matrix inverse cannot be performed separately for each space-angle-energy element.

Remark. *The residual vector $\hat{\mathbf{r}}$ and mass matrix $\mathbf{M}$ can be partitioned into blocks:*

$$\hat{\mathbf{r}} = \begin{pmatrix} \hat{\mathbf{r}}_1 \\ \hat{\mathbf{r}}_2 \\ \vdots \\ \hat{\mathbf{r}}_{|\mathcal{T}_{\Omega,\mathbb{S}}|} \end{pmatrix}, \qquad \begin{pmatrix} \mathbf{M}_1 & & & \\ & \mathbf{M}_2 & & \\ & & \ddots & \\ & & & \mathbf{M}_{|\mathcal{T}_{\Omega,\mathbb{S}}|} \end{pmatrix},$$

where, for each $1 \leq k \leq |\mathcal{T}_{\Omega,\mathbb{S}}|$, $\hat{\mathbf{r}}_k$ *and* $\mathbf{M}_k$ *respectively correspond to the local residual vector and local space-angle mass matrix on a unique space-angle element pair* $\kappa_\Omega \times \kappa_\mathbb{S} \in \mathcal{T}_{\Omega,\mathbb{S}}$. *By constructing an appropriate bijection from* $\{1, 2, \ldots, |\mathcal{T}_{\Omega,\mathbb{S}}|\}$ *to the elements of* $\mathcal{T}_{\Omega,\mathbb{S}}$, *the* a posteriori *error estimate in (5.26) can be written as*

$$|||u_h - \hat{u}_h|||_{DG} \leq \sqrt{\sum_{\kappa \in \mathcal{T}_{\Omega,\mathbb{S}}} \hat{\mathbf{r}}_\kappa^\top \mathbf{M}_\kappa \hat{\mathbf{r}}_\kappa}.$$

That is, the a posteriori *error estimate proven in Theorem 5.2.8 can be used to compute local solver error estimators for each element in the space-angle mesh.*

5.3 Spectral Properties of Modified Source Iteration

Having shown that the family of modified source iteration methods converges for selected parameter values and has computable *a posteriori* error estimates, we turn our attention to the spectral properties of the iteration (5.18) by means of analysing the iteration matrix $\mathbf{G}_\omega \in \mathbb{C}^{N\times N}$ defined by

$$\mathbf{G}_\omega = (\mathbf{T} - \omega\mathbf{M})^{-1} (\mathbf{S} - \omega\mathbf{M}) . \tag{5.27}$$

For a matrix $\mathbf{G} \in \mathbb{C}^{N\times N}$, we denote its spectrum by $\sigma(\mathbf{G})$ and define it as the subset of $\mathbb{C}$ containing the eigenvalues of $\mathbf{G}$; that is,

$$\sigma(\mathbf{G}) = \left\{\lambda \in \mathbb{C} : \mathbf{G}\mathbf{v} = \lambda\mathbf{v} \text{ for some } \mathbf{v} \in \mathbb{C}^N \setminus \{\mathbf{0}\}\right\} . \tag{5.28}$$

We similarly denote the spectral radius of $\mathbf{G}$ by $\rho(\mathbf{G})$ and define it as the largest absolute value of any eigenvalue of $\mathbf{G}$; that is,

$$\rho(\mathbf{G}) = \max\{|\lambda| : \lambda \in \sigma(\mathbf{G})\} . \tag{5.29}$$

We will specifically identify subsets of the complex plane containing the spectrum of the iteration matrix (5.27), as well as its spectral radius, given as functions of the relaxation parameter ω. The motivation for doing this is to better understand the convergence of over-relaxed variants of source iteration and Krylov subspace methods with multiple-transport-sweep preconditioners.

Note that the matrix $\mathbf{T} - \omega\mathbf{M}$ is invertible since the corresponding bilinear form $T - \omega M$ is coercive in the $||\cdot||_{DG(\omega)}$-norm. In the following work, we shall assume that the scattering kernel θ satisfies Mercer's condition and that the mono-energetic LBTE is discretised using discontinuous Galerkin finite elements in space and angle.

5.3.1 Spectrum of $\mathbf{G}_\theta$

We commence with the following straightforward result on complex numbers, which will be used in the spectral analysis.

Lemma 5.3.1. *Let $\alpha, \beta, \gamma, \rho \in \mathbb{R}$ with $\beta > 0$, $\rho \geq 0$ and $\left|\frac{\alpha}{\beta}\right| \leq \rho$.*

1. *If $\alpha \geq 0$, then*
$$\left|\frac{\alpha}{\beta + i\gamma} - \frac{\rho}{2}\right| \leq \frac{\rho}{2}.$$
2. *If $\alpha \leq 0$, then*
$$\left|\frac{\alpha}{\beta + i\gamma} + \frac{\rho}{2}\right| \leq \frac{\rho}{2}.$$

Proof. We consider each case separately.

1. We have $0 \leq \alpha \leq \rho\beta$ and
$$\begin{aligned}\left|\frac{\alpha}{\beta + i\gamma} - \frac{\rho}{2}\right|^2 &= \left(\frac{\alpha}{\beta + i\gamma} - \frac{\rho}{2}\right)\left(\frac{\alpha}{\beta - i\gamma} - \frac{\rho}{2}\right) \\ &= \frac{\alpha^2}{\beta^2 + \gamma^2} - \frac{\rho\alpha}{2} \cdot \frac{2\beta}{\beta^2 + \gamma^2} + \frac{\rho^2}{4} \\ &\leq \frac{\rho^2}{4}.\end{aligned}$$
2. We have $0 \leq -\alpha \leq \rho\beta$ and
$$\begin{aligned}\left|\frac{\alpha}{\beta + i\gamma} + \frac{\rho}{2}\right|^2 &= \left(\frac{\alpha}{\beta + i\gamma} + \frac{\rho}{2}\right)\left(\frac{\alpha}{\beta - i\gamma} + \frac{\rho}{2}\right) \\ &= \frac{\alpha^2}{\beta^2 + \gamma^2} + \frac{\rho\alpha}{2} \cdot \frac{2\beta}{\beta^2 + \gamma^2} + \frac{\rho^2}{4} \\ &\leq \frac{\rho^2}{4}.\end{aligned}$$

□

Lemma 5.3.2. *Every eigenvalue $\lambda_k \in \sigma(\mathbf{G}_\omega)$ can be written in the form*
$$\lambda_k = \frac{\alpha_k}{\beta_k + i\gamma_k} \tag{5.30}$$
where $\alpha_k, \beta_k, \gamma_k \in \mathbb{R}$ are constants depending on the real and imaginary parts of a corresponding eigenvector $\mathbf{v}_k$.

Proof. Let $\lambda_k \in \sigma(\mathbf{G}_\omega)$ denote an eigenvalue of $\mathbf{G}_\omega$ and $\mathbf{v}_k \in \mathbb{C}^N$ a corresponding eigenvector. We have
$$(\mathbf{T} - \omega\mathbf{M})^{-1} (\mathbf{S} - \omega\mathbf{M}) \mathbf{v}_k = \lambda_k \mathbf{v}_k.$$
Multiplying both sides by $\mathbf{v}_k^* (\mathbf{T} - \omega\mathbf{M})$ and rearranging, we get
$$\lambda_k = \frac{\mathbf{v}_k^* (\mathbf{S} - \omega\mathbf{M}) \mathbf{v}_k}{\mathbf{v}_k^* (\mathbf{T} - \omega\mathbf{M}) \mathbf{v}_k}.$$
Introducing $\mathbf{x}_k, \mathbf{y}_k \in \mathbb{R}^N$ such that $\mathbf{v}_k = \mathbf{x}_k + i\mathbf{y}_k$ and recalling that $\mathbf{S}$ and $\mathbf{M}$ (defined in (5.11) and (5.16) respectively) are Hermitian, the result of the lemma is readily shown:
$$\mathbf{v}_k^* (\mathbf{S} - \omega\mathbf{M}) \mathbf{v}_k = \underbrace{\mathbf{x}_k^\top (\mathbf{S} - \omega\mathbf{M}) \mathbf{x}_k + \mathbf{y}_k^\top (\mathbf{S} - \omega\mathbf{M}) \mathbf{y}_k}_{=:\alpha_k},$$
$$\begin{aligned}\mathbf{v}_k^* (\mathbf{T} - \omega\mathbf{M}) \mathbf{v}_k = &\underbrace{\mathbf{x}_k^\top (\mathbf{T} - \omega\mathbf{M}) \mathbf{x}_k + \mathbf{y}_k^\top (\mathbf{T} - \omega\mathbf{M}) \mathbf{y}_k}_{=:\beta_k} \\ &+ i \underbrace{\left[\mathbf{x}_k^\top \mathbf{T} \mathbf{y}_k - \mathbf{y}_k^\top \mathbf{T} \mathbf{x}_k\right]}_{=:\gamma_k}.\end{aligned}$$

□

Remark. *The division by the term* $\mathbf{v}_k^* (\mathbf{T} - \omega\mathbf{M}) \mathbf{v}_k$ *is reasonable since the real part of this term is equal to* $||v_1||^2_{DG(\omega)} + ||v_2||^2_{DG(\omega)}$ *for some* $v_1, v_2 \in \mathcal{V}_{\Omega,\mathbb{S}}$*, with* v_1 *(resp.* v_2*) the finite element function corresponding to* $\mathbf{x}$ *(resp.* $\mathbf{y}$*) - at least one of these* $DG(\omega)$*-norm terms must be non-zero.*

Theorem 5.3.3 (Bounding discs of MSI spectrum)**.** *The spectrum of the modified source iteration matrix* $\mathbf{G}_\omega$ *satisfies*

$$\sigma(\mathbf{G}_\omega) \subseteq D\left(\frac{c(1-\omega)}{2(1-c\omega)}, \frac{c(1-\omega)}{2(1-c\omega)}\right) \cup D\left(-\frac{c\omega}{2(1-c\omega)}, \frac{c\omega}{2(1-c\omega)}\right),$$

where the sets $D(a, s)$ *are defined for* $a \in \mathbb{C}$ *and* $s \geq 0$ *by*

$$D(a, s) = \{z \in \mathbb{C} : |z - a| \leq s\}.$$

In particular, we have

$$\sigma(\mathbf{G}_\omega) \subseteq D\left(\frac{c(1-2\omega)}{2(1-c\omega)}, \frac{c}{2(1-c\omega)}\right) \subseteq D\left(0, \frac{c\ r(\omega)}{1-c\omega}\right)$$

and

$$\rho(\mathbf{G}_\omega) \leq \frac{c\ r(\omega)}{1-c\omega},$$

where $r(\omega) = \max\{\omega, 1-\omega\}$.

Proof. Letting $\lambda_k \in \sigma(\mathbf{G}_\omega)$ and using the definitions of $\alpha_k, \beta_k, \gamma_k \in \mathbb{R}$ and $\mathbf{x}_k, \mathbf{y}_k \in \mathbb{C}^N$ in Lemma 5.3.2, it suffices to prove that $\beta_k > 0$ and $|\alpha_k| \leq \rho\beta_k$ for some $\rho \geq 0$, which will depend on the sign of α_k. In fact, we automatically have that $\beta_k > 0$ since

$$\beta_k = \mathbf{x}_k^\top (\mathbf{T} - \omega\mathbf{M}) \mathbf{x}_k + \mathbf{y}_k^\top (\mathbf{T} - \omega\mathbf{M}) \mathbf{y}_k,$$

and the right-hand-side above can be expressed as

$$T(x, x) - \omega M(x, x) + T(y, y) - \omega M(y, y) = |||x|||^2_{DG(\omega)} + |||y|||^2_{DG(\omega)} > 0$$

for some $x, y \in \mathcal{V}_{\Omega,\mathbb{S}} \setminus \{0\}$.

Lemmas 5.2.2 and 5.2.5 will be translated from the language of bilinear forms to the language of vector-matrix-vector products and used to prove that $|\alpha_k| \leq \rho\beta_k$ for each of the cases $\alpha_k \geq 0$ and $\alpha_k < 0$.

1. If $\alpha_k \geq 0$, we have

$$\begin{aligned}
0 &\leq \alpha_k \\
&= \mathbf{x}_k^\top (\mathbf{S} - \omega\mathbf{M}) \mathbf{x}_k + \mathbf{y}_k^\top (\mathbf{S} - \omega\mathbf{M}) \mathbf{y}_k \\
&\leq (1-\omega)\ \mathbf{x}_k^\top \mathbf{M} \mathbf{x}_k + (1-\omega)\ \mathbf{y}_k^\top \mathbf{M} \mathbf{y}_k \\
&\leq \frac{c(1-\omega)}{1-c\omega}\ \mathbf{x}_k^\top (\mathbf{T} - \omega\mathbf{M}) \mathbf{x}_k + \frac{c(1-\omega)}{1-c\omega}\ \mathbf{y}_k^\top (\mathbf{T} - \omega\mathbf{M}) \mathbf{y}_k \\
&= \frac{c(1-\omega)}{1-c\omega} \beta_k.
\end{aligned}$$

Here, we have used Lemma 5.2.2 for the first inequality and Lemma 5.2.5 for the second inequality. Invoking Lemma 5.3.1 with $\rho = \frac{c(1-\omega)}{1-c\omega}$ shows that

$$\lambda_k \in D\left(\frac{c(1-\omega)}{2(1-c\omega)}, \frac{c(1-\omega)}{2(1-c\omega)}\right).$$

2. If $\alpha_k \leq 0$, we have

$$\begin{aligned}
0 &\leq -\alpha_k \\
&= -\mathbf{x}_k^\top (\mathbf{S} - \omega\mathbf{M})\,\mathbf{x}_k - \mathbf{y}_k^\top (\mathbf{S} - \omega\mathbf{M})\,\mathbf{y}_k \\
&\leq \omega\ \mathbf{x}_k^\top \mathbf{M}\mathbf{x}_k + \omega\ \mathbf{y}_k^\top \mathbf{M}\mathbf{y}_k \\
&\leq \frac{c\omega}{1-c\omega}\ \mathbf{x}_k^\top (\mathbf{T} - \omega\mathbf{M})\,\mathbf{x}_k + \frac{c\omega}{1-c\omega}\ \mathbf{y}_k^\top (\mathbf{T} - \omega\mathbf{M})\,\mathbf{y}_k \\
&= \frac{c\omega}{1-c\omega}\beta_k.
\end{aligned}$$

Here, we have used Lemma 5.2.2 for the first inequality and Lemma 5.2.5 for the second inequality. Invoking Lemma 5.3.1 with $\rho = \frac{c\omega}{1-c\omega}$ shows that

$$\lambda_k \in D\left(-\frac{c\omega}{2(1-c\omega)}, \frac{c\omega}{2(1-c\omega)}\right).$$

Since each λ_k lies in one of these two discs, $\sigma(\mathbf{G}_\omega)$ is contained in their union. The final two results follow by straightforward geometric considerations. □

We can draw some similarities to the previous *a priori* convergence analysis of the modified source iteration method. The contraction factor $\frac{c\ r(\omega)}{1-c\omega}$ appearing in the error reduction bound in Theorem 5.2.6 is precisely the (upper bound on the) spectral radius of $\mathbf{G}_\omega$. We remarked that the choice $\omega = \frac{1}{2}$ is optimal in the sense of minimising the contraction factor - in light of Theorem 5.3.3, we now recognise that this choice "centralises" the spectrum of the iteration matrix at the origin in the complex plane. This is shown in Figure 5.1.

5.3.2 Related linear solvers

Successive over-relaxation (SOR) In light of Theorem 5.3.3, we can make a direct comparison between the modified source iteration (5.18) and the following relaxed variant of the classical source iteration method:

$$\mathbf{u}^{(n+1)} = \mathbf{u}^{(n)} + \omega\mathbf{T}^{-1}\left(\mathbf{f} - (\mathbf{T} - \mathbf{S})\mathbf{u}^{(n)}\right). \tag{5.31}$$

The iteration (5.31) can be thought of as the specific application of the preconditioned Richardson iteration (5.4) recast in a linear algebraic form. Here, $0 < \omega < 2$ is an over-relaxation parameter. When $\omega < 1$, the method is *under-relaxed*; when $\omega > 1$, the method is *over-relaxed*; and when $\omega = 1$, (5.31) reduces to the source iteration method (5.13). We will henceforth denote by ω_{MSI} the relaxation parameter used in

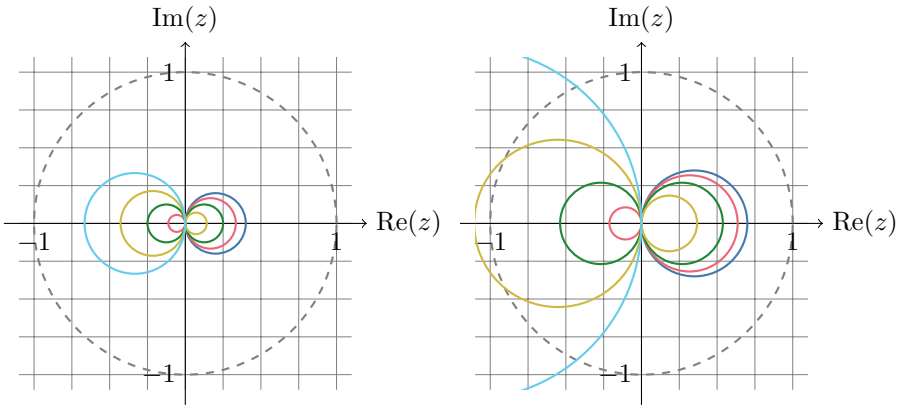

Figure 5.1: Bounding discs for the modified source iteration method with $c = 0.4$ (left) and $c = 0.7$ (right) for different values of $\theta \in [0, 1]$. Dashed grey circle shows the boundary of $D(0, 1)$. Blue: $\theta = 0$. Red: $\theta = 0.25$. Green: $\theta = 0.5$. Yellow: $\theta = 0.75$. Cyan: $\theta = 1$.

the modified source iteration method (5.18) and by ω_{SOR} the relaxation parameter used in the successively-over-relaxed source iteration method (5.31) to avoid confusion.

The family of iteration matrices $\mathbf{G}(\omega_{SOR})$ corresponding to the iteration (5.31) is defined by

$$\mathbf{G}(\omega_{SOR}) = (1 - \omega_{SOR})\mathbf{I} + \omega_{SOR}\mathbf{G},$$

where $\mathbf{G}$ is the iteration matrix for the classical source iteration (i.e. $\omega_{SOR} = 1, \omega_{MSI} = 0$) and, by Theorem 5.3.3, satisfies

$$\sigma(\mathbf{G}) \subseteq D\left(\frac{c}{2}, \frac{c}{2}\right).$$

Through the relationship between $\mathbf{G}(\omega_{SOR})$ and $\mathbf{G}$, it can be shown that the spectrum of the relaxed iteration matrix is also contained in a disc in the complex plane:

$$\sigma(\mathbf{G}(\omega_{SOR})) \subseteq D\left(1 - \omega_{SOR} + \frac{c\omega_{SOR}}{2}, \frac{c\omega_{SOR}}{2}\right),$$

from which it can be deduced that

$$\rho(\mathbf{G}(\omega_{SOR})) \leq \left|1 - \omega_{SOR} + \frac{c\omega_{SOR}}{2}\right| + \frac{c\omega_{SOR}}{2}.$$

Table 5.1 shows a comparison between the spectral properties of the modified and relaxed versions of source iteration. The results of the relaxed version of source iteration are consistent with the analysis of Wang [100], where the theoretical optimal choice of ω_{SOR} was also found to be close to $\frac{2}{2-c}$. In that work, however, a scheme based on different spatial and angular discretisations was analysed, and the result proven there was explicit with respect to the optical thickness ε of the medium. We recall the definition of ε in (5.1).

The most striking similarity is that the optimal choices of ω_{MSI} and ω_{SOR} minimising the spectral radius of the iteration matrices of both methods yield the same upper bound

of $\frac{c}{2-c}$. In fact, by substituting

$$\omega_{SOR} = \frac{1}{1 - c\omega_{MSI}}$$

into the second column of Table 5.1, the bounding discs, spectral radii, convergence criteria and optimal parameter choices between the modified and relaxed source iterations all agree. This establishes the modified source iteration as a type of over-relaxation method.

	Modified SI ($\omega = \omega_{MSI}$)	Relaxed SI ($\omega = \omega_{SOR}$)
Iteration matrix	$(\mathbf{T} - \omega\mathbf{M})^{-1}(\mathbf{S} - \omega\mathbf{M})$	$(1-\omega)\mathbf{I} + \omega\mathbf{T}^{-1}\mathbf{S}$
Bounding disc	$D\left(\frac{c(1-2\omega)}{2(1-c\omega)}, \frac{c}{2(1-c\omega)}\right)$	$D\left(1-\omega+\frac{c\omega}{2}, \frac{c\omega}{2}\right)$
Spectral radius	$\frac{c}{1-c\omega}\left(\left\lvert\omega - \frac{1}{2}\right\rvert + \frac{1}{2}\right)$	$\left\lvert 1-\omega+\frac{c\omega}{2}\right\rvert + \frac{c\omega}{2}$
Convergence	$0 < \omega < \frac{1}{2c}$	$0 < \omega < 2$
Optimal parameter choice	$\omega = \frac{1}{2}$	$\omega = \frac{2}{2-c}$
Optimal spectral radius	$\frac{c}{2-c}$	$\frac{c}{2-c}$

Table 5.1: Comparison of spectral properties of modified source iteration (5.18) and relaxed source iteration (5.31).

The modified source iteration method enjoys a couple of benefits with respect to the over-relaxed source iteration method. Firstly, the optimal choice of the parameter ω_{MSI} is independent of the global scattering ratio c (defined in (5.19)), as opposed to the parameter ω_{SOR} whose optimal choice depends on c. This is useful since we no longer need to have accurate approximations of c in order to implement the version of the method with (theoretically) the most rapid convergence. Moreover, c may not be representative of the local scattering ratio $\tilde{c}$ (defined in (5.20)) everywhere in the spatial domain if the medium is heterogeneous - the modified source iteration method naturally respects variation in the local scattering ratio.

Secondly, the analysis of the modified source iteration method readily yields *a posteriori* error estimates on the solver error, which are not so straightforward for the over-relaxed source iteration method. This is because one needs to prove sufficiently-sharp continuity bounds for the terms involving the bilinear form $(1-\omega_{SOR})T(\cdot,\cdot)+\omega_{SOR}S(\cdot,\cdot)$ with respect to some $DG(\omega)$-norm. In the modified source iteration case, this bound was provided by Lemma 5.2.4 - however, the relationship between the (Hermitian positive-semidefinite) bilinear forms S and M was instrumental in the proof of that lemma.

Generalised Minimum Residual (GMRES) The generalised minimum residual method (GMRES), developed by Saad and Schultz [84], is one of the most widely-used Krylov subspace methods for large, sparse non-symmetric systems of equations. Applied to the linear system

$$\mathbf{Ax} = \mathbf{b}$$

with an initial estimate $\mathbf{x}_0$, GMRES constructs a sequence of Krylov subspaces $\mathcal{K}_n$ defined by

$$\mathcal{K}_n = \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots, \mathbf{A}^{n-1}\mathbf{r}_0\},$$

where $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$ is the initial residual error. At the n^{th} iteration, the corresponding approximation $\mathbf{x}_n$ to $\mathbf{x}$ is selected to be the minimiser of the Euclidean norm of the residual error $\mathbf{r}_n = \mathbf{b} - \mathbf{A}\mathbf{x}_n$ from $\mathbf{x}_0 + \mathcal{K}_n$; that is, $\mathbf{x}_n$ is the unique solution to the following minimisation problem:

$$\mathbf{x}_n = \underset{\mathbf{y} \in \mathbf{x}_0 + \mathcal{K}_n}{\arg\min} ||\mathbf{b} - \mathbf{A}\mathbf{y}||_2. \tag{5.32}$$

In practice, the Arnoldi iteration is employed to construct an orthonormal basis $\{\mathbf{v}_i\}_{i=1}^n$ for $\mathcal{K}_n$. At the n^{th} step, the new basis vector $\mathbf{v}_n$ is obtained by orthonormalising $\mathbf{A}\mathbf{v}_{n-1}$ against all previous basis vectors $\{\mathbf{v}_i\}_{i=1}^{n-1}$. This process additionally returns an $(n+1)$-by-n upper Hessenberg matrix $\mathbf{H}_n$, which is used in the solution of the least-squares problem (5.32). The least-squares problem can be written in the form

$$\mathbf{y}_n = \underset{\mathbf{y} \in \mathbb{R}^n}{\arg\min} ||\beta \mathbf{e}_1 - \mathbf{H}_n \mathbf{y}||_2, \tag{5.33}$$

where $\beta = ||\mathbf{r}_0||_2$ and $\mathbf{e}_1 = (1, 0, \ldots, 0)^\top$. This problem (of size n) can be solved using repeated applications of plane rotations and additionally returns the Euclidean norm of the residual error $||\mathbf{r}_n||_2$ as a by-product, which may be used to terminate the GMRES iteration early if the linear system is solved to a sufficient accuracy. Writing the matrix $\mathbf{V}_n = (\mathbf{v}_1, \ldots, \mathbf{v}_n)$, the n^{th} approximation to the solution of the linear system is constructed as

$$\mathbf{x}_n = \mathbf{x}_0 + \mathbf{V}_n \mathbf{y}_n.$$

GMRES is an example of a direct linear solver since, in exact arithmetic, the true solution to the linear system of equations is returned after N iterations, where N denotes the number of unknown variables to solve for. However, since the sequence of approximations $\mathbf{x}_n$ are chosen to minimise the residual error in the n^{th} Krylov subspace, it is often sufficient to terminate the iterative process when the residual error is smaller than some accepted tolerance. One may also be forced to terminate or restart GMRES when the dimension of the Krylov subspace has grown large enough that one may not store the full basis.

The convergence of GMRES in the worst-case scenario is well-known - for any non-increasing sequence of non-negative real numbers $(r_n)_{n=0}^N$ with $r_N = 0$, one may find a N-by-N linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ for which $(r_n)_{n=0}^N$ is precisely the sequence of residual errors $r_n = ||\mathbf{b} - \mathbf{A}\mathbf{x}_n||$ obtained from the GMRES algorithm [46]. In particular, one can construct a problem for which the sequence of residual errors is non-decreasing for the first $N-1$ iterations and convergence is only reached on the last iteration.

Outside of such pathological cases, the convergence of GMRES can be characterised by a number of different properties of the coefficient matrix $\mathbf{A}$ - an interesting review of

some techniques for obtaining bounds on the sequence of residual norm errors is given in [41]. We shall focus on the simplest convergence result based on the eigenvalues of $\mathbf{A}$.

Proposition 5.3.4 ([83], Prop. 6.32)**.** *Assume that* $\mathbf{A}$ *is a diagonalisable matrix and let* $\mathbf{A} = \mathbf{X}\mathbf{\Lambda}\mathbf{X}^{-1}$ *where* $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$ *is the diagonal matrix of eigenvalues of* $\mathbf{A}$*. Define*

$$\varepsilon^{(m)} = \min_{p \in \mathbb{P}_m, p(0)=1} \max_{i=1,\ldots,N} |p(\lambda_i)|,$$

where $\mathbb{P}_m$ *denotes the space of polynomial functions on* $\mathbb{C}$ *of degree* $m \geq 0$*. Then, the residual norm achieved by the* m^{th} *step of GMRES satisfies the inequality*

$$||\mathbf{r}_m||_2 \leq \kappa_2(\mathbf{X})\varepsilon^{(m)}||\mathbf{r}_0||_2,$$

where $\kappa_2(\mathbf{X}) = ||\mathbf{X}||_2||\mathbf{X}^{-1}||_2$ *denotes the 2-norm condition number of* $\mathbf{X}$*.*

If the spectrum of $\mathbf{A}$ is known to satisfy

$$\sigma(\mathbf{A}) \subseteq \{z \in \mathbb{C} : |z - a| \leq r\} =: D(a, r)$$

for some $a \in \mathbb{C}$ and $r \geq 0$, the residual error bound given in Proposition 5.3.4 can be made explicit with respect to a and r through manipulation of the quantity $\varepsilon^{(m)}$:

$$\begin{aligned} ||\mathbf{r}_m||_2 &\leq \kappa_2(\mathbf{X})||\mathbf{r}_0||_2 \min_{p \in \mathbb{P}_m, p(0)=1} \max_{i=1,\ldots,N} |p(\lambda_i)| \\ &\leq \kappa_2(\mathbf{X})||\mathbf{r}_0||_2 \min_{p \in \mathbb{P}_m, p(0)=1} \max_{\lambda \in D(a,r)} |p(\lambda)| \\ &\leq \kappa_2(\mathbf{X})||\mathbf{r}_0||_2 \max_{\lambda \in D(a,r)} \left| \left(1 - \frac{z}{a}\right)^m \right| \\ &= \kappa_2(\mathbf{X})||\mathbf{r}_0||_2 \left(\frac{r}{|a|}\right)^m. \end{aligned} \tag{5.34}$$

By itself, GMRES may not be rapidly convergent, or may even stagnate. However, one may employ preconditioning operations to accelerate the convergence of GMRES. Loosely speaking, a preconditioner $\mathbf{P}^{-1} \approx \mathbf{A}^{-1}$ is an operation chosen such that the condition number of $\mathbf{P}^{-1}\mathbf{A}$ is smaller than that of $\mathbf{A}$. GMRES may be implemented with left, right or split-preconditioning [83]; for generality, we shall describe the case with split-preconditioning. Assume that $\mathbf{P} \approx \mathbf{A}$ is (the inverse of) a preconditioner for the matrix $\mathbf{A}$ which may be factored as $\mathbf{P} = \mathbf{P}_L\mathbf{P}_R$. Split-preconditioned GMRES attempts to solve the linear system

$$\mathbf{P}_L^{-1}\mathbf{A}\mathbf{P}_R^{-1}\mathbf{u} = \mathbf{P}_L^{-1}\mathbf{b}$$

for the auxiliary variable $\mathbf{u}$, followed by a transformation back to the original solution variable $\mathbf{x}$ via

$$\mathbf{x} = \mathbf{P}_R^{-1}\mathbf{u}.$$

At the n^{th} step, split-preconditioned GMRES will attempt to minimise the left-preconditioned residual

$$||\mathbf{P}_L^{-1}\mathbf{b} - \mathbf{P}_L^{-1}\mathbf{A}\mathbf{P}_R^{-1}\mathbf{u}_n||_2 = ||\mathbf{P}_L^{-1}\left(\mathbf{b} - \mathbf{A}\mathbf{x}_n\right)||_2.$$

Left- and right-preconditioned GMRES can be thought of as special cases with $\mathbf{P}_R = \mathbf{I}$ and $\mathbf{P}_L = \mathbf{I}$ respectively.

By careful selection of the left- and right-preconditioners, we can exploit the split-preconditioned GMRES method to terminate once the residual-based *a posteriori* error estimate given in the statement of Theorem 5.2.8 is smaller than a given tolerance, without explicitly computing the residual error vector, for the case of the discretised LBTE system

$$(\mathbf{T} - \mathbf{S})\,\mathbf{u} = \mathbf{f}.$$

Theorem 5.3.5. *Let* $\mathbf{T}$, $\mathbf{S}$ *and* $\mathbf{f}$ *be as in (5.10), (5.11) and (5.12) respectively, and let* $\mathbf{P} \approx \mathbf{T} - \mathbf{S}$ *denote any preconditioner for the matrix* $\mathbf{T} - \mathbf{S}$. *Let* $\mathbf{M}$ *be as in (5.25), and let* $\mathbf{L}$ *denote the (lower-triangular) matrix in the Cholesky decomposition* $\mathbf{M} = \mathbf{L}\mathbf{L}^\top$. *Finally, for any* $\mathbf{A} \in \mathbb{C}^{N\times N}$, $\mathbf{g} \in \mathbb{C}^N$ *and* $\texttt{TOL} > 0$, *let* $\texttt{gmres}(\mathbf{A}, \mathbf{g}, \texttt{TOL})$ *denote an implementation of the GMRES algorithm returning an approximation* $\hat{\mathbf{u}} \in \mathbb{C}^N$ *to the true solution* $\mathbf{u} \in \mathbb{C}^N$ *of the linear system*

$$\mathbf{A}\mathbf{u} = \mathbf{g}$$

satisfying $||\mathbf{g} - \mathbf{A}\hat{\mathbf{u}}||_2 \leq \texttt{TOL}$. *The following function calls*

$$\begin{aligned}
&\hat{\mathbf{z}} \leftarrow \texttt{gmres}(\mathbf{L}^{-1}(\mathbf{T} - \mathbf{S})\mathbf{P}^{-1}\mathbf{L}, \mathbf{L}^{-1}\mathbf{f}, \texttt{TOL}),\\
&\hat{\mathbf{u}} \leftarrow \mathbf{P}^{-1}\mathbf{L}\hat{\mathbf{z}}
\end{aligned}$$

generate an approximate solution $\hat{u}_h \approx u_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ *to the solution of the (discrete) variational problem (5.17) satisfying*

$$|||u_h - \hat{u}_h|||_{DG} \leq \texttt{TOL}.$$

Proof. Consider the following splitting of the preconditioner $\mathbf{P}$:

$$\mathbf{P} = \underbrace{\mathbf{L}}_{=:\mathbf{P}_L} \cdot \underbrace{\mathbf{L}^{-1}\mathbf{P}}_{=:\mathbf{P}_R}.$$

The sequence of calls then assumes the form

$$\begin{aligned}
&\hat{\mathbf{z}} \leftarrow \texttt{gmres}(\mathbf{P}_L^{-1}(\mathbf{T} - \mathbf{S})\mathbf{P}_R^{-1}, \mathbf{P}_L^{-1}\mathbf{f}, \texttt{TOL}),\\
&\hat{\mathbf{u}} \leftarrow \mathbf{P}_R^{-1}\hat{\mathbf{z}},
\end{aligned}$$

which generates a vector $\hat{\mathbf{u}}$ satisfying $||\mathbf{L}^{-1}\hat{\mathbf{r}}||_2 \leq \texttt{TOL}$, where $\hat{\mathbf{r}} = \mathbf{f} - (\mathbf{T} - \mathbf{S})\hat{\mathbf{u}}$ denotes the residual vector induced by $\hat{\mathbf{u}}$. But we have

$$\begin{aligned}
||\mathbf{L}^{-1}\hat{\mathbf{r}}||_2 &= \sqrt{\hat{\mathbf{r}}^*\mathbf{L}^{-*}\mathbf{L}^{-1}\hat{\mathbf{r}}}\\
&= \sqrt{\hat{\mathbf{r}}^*\mathbf{M}^{-1}\hat{\mathbf{r}}}\\
&= ||r_h||_{L^2_w(\mathcal{D})},
\end{aligned}$$

where we have used the fact that $\mathbf{M}^{-1} = \mathbf{L}^{-*}\mathbf{L}^{-1}$ and the definition of $||r_h||_{L^2_w(\mathcal{D})}$ in (5.26). Using Theorem 5.2.8, we therefore have

$$|||u_h - \hat{u}_h|||_{DG} \leq ||r_h||_{L^2_w(\mathcal{D})} = ||\mathbf{L}^{-1}\hat{\mathbf{r}}||_2 \leq \texttt{TOL}.$$

□

Remark. *The result of Theorem 5.3.5 applies to both mono-energetic and poly-energetic problems. In the latter case, the* a posteriori *error estimate is provided by Theorem 5.5.3 in Chapter 5.5.1.*

Remark. *The result of Theorem 5.3.5 still holds when the function* `GMRES` *is used with restarting.*

5.3.3 Transport-Based Preconditioners

We shall now discuss preconditioning techniques for Krylov subspace solvers based on transport sweeps [77]. Specifically, we shall consider the model problem

$$(\mathbf{I} - \mathbf{G})\,\mathbf{x} = \mathbf{b},$$

where $\mathbf{G}$ is an iteration matrix associated with the (convergent) stationary iterative method

$$\mathbf{x}_{n+1} = \mathbf{G}\mathbf{x}_n + \mathbf{b}.$$

We shall assume that $\sigma(\mathbf{G}) \subseteq D(a, r)$ is known *a priori* for some $a \in \mathbb{C}$ and $r \geq 0$. Using this assumption, we shall construct preconditioners for the matrix $\mathbf{I} - \mathbf{G}$ as polynomial functions of $\mathbf{G}$:

$$\mathbf{P}_n^{-1} = \sum_{k=0}^{n-1} a_k^{(n)} \mathbf{G}^k,$$

where the coefficients $\{a_k^{(n)}\}_{k=0}^{n-1} \subset \mathbb{C}$ are to be determined. The preconditioner $\mathbf{P}_n^{-1}$ may be applied on the left:

$$\mathbf{P}_n^{-1}\,(\mathbf{I} - \mathbf{G})\,\mathbf{x} = \mathbf{P}_n^{-1}\mathbf{b},$$

or on the right:

$$\begin{aligned}(\mathbf{I} - \mathbf{G})\,\mathbf{P}_n^{-1}\mathbf{u} &= \mathbf{f},\\ \mathbf{x} &= \mathbf{P}_n^{-1}\mathbf{u},\end{aligned}$$

in the context of Krylov subspace methods. The spectral properties of the left- and right-preconditioned systems are identical, and in practice both approaches tend to share similar convergence properties. For simplicity, we shall first study the left-preconditioned case.

The following theorem provides a family of preconditioners based on *a priori* knowledge of the iteration matrix $\mathbf{G}$.

Theorem 5.3.6. *Let* $\mathbf{G}$ *be a matrix satisfying*

$$\sigma(\mathbf{G}) \subseteq D(a, s)$$

where $a \in \mathbb{R} \setminus \{1\}$ *and* $s \geq 0$. *The family of polynomial preconditioners* $\mathbf{P}_n^{-1}$ *for the matrix* $\mathbf{I} - \mathbf{G}$ *of the form*

$$\mathbf{P}_n^{-1} = \sum_{k=0}^{n-1} a_k^{(n)} \mathbf{G}^k$$

with coefficients $\{a_k^{(n)}\}_{k=0}^{n-1} \subset \mathbb{R}$ *given by*

$$a_k^{(n)} = \frac{1}{(1-a)^n} \sum_{j=k+1}^{n} \binom{n}{j} (-a)^{n-j}$$

satisfies

$$\sigma(\mathbf{I} - \mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})) \subseteq D\left(0, \left(\frac{s}{1-a}\right)^n\right).$$

Moreover, if $\mathbf{G} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$ *is diagonalisable and* $s < 1 - a$, *then*

$$\kappa_2(\mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})) \leq \kappa_2(\mathbf{V})^2 \cdot \frac{1 + \left(\frac{s}{1-a}\right)^n}{1 - \left(\frac{s}{1-a}\right)^n}, \tag{5.35}$$

where $\kappa_2(\mathbf{V})$ *denotes the 2-norm condition number of* $\mathbf{V}$.

Proof. Consider the family of maps $f_n : \mathbb{C} \to \mathbb{C}$ defined by

$$f_n(z) = r_n \left(\frac{z-a}{s}\right)^n.$$

Note that f_n maps $D(a, s)$ into $D(0, r_n)$; in particular, it maps $\sigma(\mathbf{G})$ into $D(0, r_n)$.

Our objective is to select the coefficients $\{a_k^{(n)}\}_{k=0}^{n-1}$ such that $\mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})$ has eigenvalues clustered about 1; or equivalently, that $\mathbf{I} - \mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})$ has eigenvalues clustered about zero[1]. We shall achieve this by balancing the coefficients of $\{\mathbf{G}^k\}_{k=0}^n$ in the equation

$$\mathbf{I} - \mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G}) = f_n(\mathbf{G}) := r_n \left(\frac{\mathbf{G} - a\mathbf{I}}{s}\right)^n$$

and solving the resulting linear system for the coefficients $\{a_k^{(n)}\}_{k=0}^{n-1}$ as well as the maximum distance r_n between 1 and any eigenvalue of $\mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})$. For simplicity, we shall introduce $R_n = r_n s^{-n}$ and rewrite the equation above as

$$\left(\sum_{k=0}^{n-1} a_k^{(n)} \mathbf{G}^k\right)(\mathbf{I} - \mathbf{G}) + R_n (\mathbf{G} - a\mathbf{I})^n = \mathbf{I}.$$

Balancing the $\mathbf{I} = \mathbf{G}^0$ and $\mathbf{G}^n$ terms first, we obtain

$$\begin{aligned} a_0^{(n)} + (-a)^n R_n &= 1, \\ -a_{n-1}^{(n)} + R_n &= 0. \end{aligned}$$

[1] We recognise that the matrix $\mathbf{I} - \mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})$ is the iteration matrix corresponding to a preconditioned Richardson iteration with preconditioner $\mathbf{P}_n^{-1}$.

Balancing the remaining $\mathbf{G}^j$ terms for $1 \leq j \leq n-1$, we obtain

$$a_j^{(n)} - a_{j-1}^{(n)} + \binom{n}{j}(-a)^{n-j} R_n = 0.$$

We arrive at a system of $n+1$ linear equations in $n+1$ unknowns, which can be written in the following form:

$$\begin{pmatrix} 1 & & & & \binom{n}{0}(-a)^n \\ -1 & 1 & & & \binom{n}{1}(-a)^{n-1} \\ & -1 & 1 & & \binom{n}{2}(-a)^{n-2} \\ & & \ddots & \ddots & \vdots \\ & & & -1 & 1 & \binom{n}{n-1}(-a)^1 \\ & & & & -1 & \binom{n}{n}(-a)^0 \end{pmatrix} \begin{pmatrix} a_0^{(n)} \\ a_1^{(n)} \\ a_2^{(n)} \\ \vdots \\ a_{n-1}^{(n)} \\ R_n \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}.$$

This system can be solved straightforwardly using Gaussian elimination. Performing forward substitution on this system yields

$$\begin{pmatrix} 1 & & & & \sum_{j=0}^{0} \binom{n}{j}(-a)^{n-j} \\ & 1 & & & \sum_{j=0}^{1} \binom{n}{j}(-a)^{n-j} \\ & & 1 & & \sum_{j=0}^{2} \binom{n}{j}(-a)^{n-j} \\ & & & \ddots & \vdots \\ & & & 1 & \sum_{j=0}^{n-1} \binom{n}{j}(-a)^{n-j} \\ & & & & \sum_{j=0}^{n} \binom{n}{j}(-a)^{n-j} \end{pmatrix} \begin{pmatrix} a_0^{(n)} \\ a_1^{(n)} \\ a_2^{(n)} \\ \vdots \\ a_{n-1}^{(n)} \\ R_n \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \\ 1 \end{pmatrix}.$$

The last equation gives us

$$\begin{aligned} R_n &= \left(\sum_{j=0}^{n} \binom{n}{j}(-a)^{n-j} \right)^{-1} = \frac{1}{(1-a)^n}, \\ \Rightarrow r_n &= \left(\frac{s}{1-a} \right)^n. \end{aligned}$$

Completing the back-substitution gives us the remaining values of $\{a_k^{(n)}\}_{k=0}^{n-1}$:

$$\begin{aligned} a_k^{(n)} &= 1 - R_n \sum_{j=0}^{k} \binom{n}{j} (-a)^{n-j} \\ &= \frac{1}{(1-a)^n} \left(\sum_{j=0}^{n} \binom{n}{j} (-a)^{n-j} - \sum_{j=0}^{k} \binom{n}{j} (-a)^{n-j} \right) \\ &= \frac{1}{(1-a)^n} \sum_{j=k+1}^{n} \binom{n}{j} (-a)^{n-j} \end{aligned}$$

To show that any eigenvalue of $\mathbf{I} - \mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})$ lies in $D(0, r_n)$, it suffices to select any eigenpair $(\lambda_i, \mathbf{v}_i)$ of $\mathbf{G}$ and substitute into the balance equation:

$$\begin{aligned} \left(\mathbf{I} - \mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})\right) \mathbf{v}_i &= f_n(\mathbf{G})\mathbf{v}_i \\ &= \frac{r_n}{s^n} \sum_{k=0}^{n} \binom{n}{k} (-a)^{n-k} \mathbf{G}^k \mathbf{v}_i \\ &= \frac{r_n}{s^n} \sum_{k=0}^{n} \binom{n}{k} (-a)^{n-k} \lambda_i^k \mathbf{v}_i \\ &= r_n \left(\frac{\lambda_i - a}{s} \right)^n \mathbf{v}_i \\ &= f_n(\lambda_i)\mathbf{v}_i. \end{aligned}$$

Observe that $\lambda_i \in D(a, s)$ and f_n maps $D(a, s)$ into $D(0, r_n)$. Finally, assuming that $\mathbf{G}$ is diagonalisable as $\mathbf{G} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$, we immediately get that $\mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})$ is also diagonalisable since

$$\mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G}) = \mathbf{I} - f_n(\mathbf{G}) = \mathbf{V}\left(\mathbf{I} - f_n(\mathbf{D})\right)\mathbf{V}^{-1}.$$

Moreover, if $s < 1 - a$, we have that $r_n < 1$ and so

$$\begin{aligned} \kappa_2(\mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})) &\leq \kappa_2(\mathbf{V})^2 \cdot \frac{\max_{\lambda_i \in \sigma(\mathbf{G})} |1 - f_n(\lambda_i)|}{\min_{\lambda_i \in \sigma(\mathbf{G})} |1 - f_n(\lambda_i)|} \\ &\leq \kappa_2(\mathbf{V})^2 \cdot \frac{\max_{\lambda \in D(a,s)} |1 - f_n(\lambda)|}{\min_{\lambda \in D(a,s)} |1 - f_n(\lambda)|} \\ &= \kappa_2(\mathbf{V})^2 \cdot \frac{\max_{f \in D(0,r_n)} |1 - f|}{\min_{f \in D(0,r_n)} |1 - f|} \\ &= \kappa_2(\mathbf{V})^2 \cdot \frac{1 + \left(\frac{s}{1-a}\right)^n}{1 - \left(\frac{s}{1-a}\right)^n}. \end{aligned}$$

□

Theorem 5.3.6 gives a method of selecting a family of preconditioners for the matrix $\mathbf{I} - \mathbf{G}$ as a polynomial in $\mathbf{G}$ given *a priori* spectral information about $\mathbf{G}$. The proof also establishes a method to implement the action of the preconditioned matrix $\mathbf{P}_n^{-1}(\mathbf{I} - \mathbf{G})$

on a vector $\mathbf{v}$ without the explicit construction of $\mathbf{P}_n^{-1}$:

$$\begin{aligned}\mathbf{P}_n^{-1}(\mathbf{I}-\mathbf{G})\mathbf{v} &= \mathbf{v} - r_n\left(\frac{\mathbf{G}-a\mathbf{I}}{s}\right)^n\mathbf{v}\\ &= \mathbf{v} - \left(\frac{s}{1-a}\right)^n\left(\frac{\mathbf{G}-a\mathbf{I}}{s}\right)^n\mathbf{v}\\ &= \mathbf{v} - \left(\frac{1}{1-a}\right)^n(\mathbf{G}-a\mathbf{I})^n\mathbf{v}.\end{aligned}$$

In particular, the coefficients $\{a_l^{(n)}\}_{k=0}^{n-1}$ are only required to compute the action of $\mathbf{P}_n^{-1}$ on a vector. When implemented within a GMRES method, this computation is only performed once - either before the start of the Arnoldi iteration (in the case of left-preconditioning) or after the Arnoldi iteration has terminated (in the case of right-preconditioning).

The choice of the preconditioner seeks to cluster the eigenvalues of the preconditioned matrix about 1 and only requires an estimate of the centre a of the spectral disc $D(a,s) \supseteq \sigma(\mathbf{G})$, which may be difficult to estimate if no other information about $\mathbf{G}$ is known. If the corresponding stationary iterative method with iteration matrix $\mathbf{G}$ is convergent, then we must have $\rho(\mathbf{G}) < 1$ and so an immediate choice of the spectral disc is given by $D(0,1) \supseteq \sigma(\mathbf{G})$. This yields the family of preconditioners based on the Neumann series for $(\mathbf{I}-\mathbf{G})^{-1}$:

$$\mathbf{P}_n^{-1} = \sum_{k=0}^{n-1}\mathbf{G}^k \approx \sum_{k=0}^{\infty}\mathbf{G}^k = (\mathbf{I}-\mathbf{G})^{-1}.$$

Moreover, if one can find a bounding disc $D(a,s) \supseteq \sigma(\mathbf{G})$ with $\frac{s}{1-a} < 1$, then it is expected that the preconditioner $\mathbf{P}_n^{-1}$ improves as $n \to \infty$, in the sense that iterative methods for the linear system $(\mathbf{I}-\mathbf{G})\mathbf{x} = \mathbf{b}$ employing $\mathbf{P}_n^{-1}$ as a preconditioner converge more rapidly for large n.

Since Theorem 5.3.3 gives us a bounding disc on the spectrum of $\mathbf{G}_\omega$, the modified source iteration matrix defined in (5.27), the following corollary holds.

Corollary 5.3.6.1. *The selection of the preconditioner $\mathbf{P}_n^{-1}$ in Theorem 5.3.6 for the case $\mathbf{G} = \mathbf{G}_\omega$ (assuming that $\mathbf{G}_\omega$ is diagonalisable) given by*

$$\mathbf{G}_\omega = (\mathbf{T}-\omega\mathbf{M})^{-1}(\mathbf{S}-\omega\mathbf{M})$$

yields the following spectrum and condition number bounds of the preconditioned matrix (independent of the parameter ω):

$$\begin{aligned}\sigma(\mathbf{I}-\mathbf{P}_n^{-1}(\mathbf{I}-\mathbf{G}_\omega)) &\subseteq D\left(0,\left(\frac{c}{2-c}\right)^n\right),\\ \kappa_2(\mathbf{P}_n^{-1}(\mathbf{I}-\mathbf{G}_\omega)) &\leq \kappa_2(\mathbf{V})^2\cdot\frac{1+\left(\frac{c}{2-c}\right)^n}{1-\left(\frac{c}{2-c}\right)^n}.\end{aligned}$$

Proof. From Theorem 5.3.3, we have that

$$\sigma(\mathbf{G}_\omega) \subseteq D\left(\frac{c(1-2\omega)}{2(1-c\omega)}, \frac{c}{2(1-c\omega)}\right) =: D(a,s),$$

from which Theorem 5.3.6 can be applied, noting that

$$\frac{s}{1-a} = \frac{\frac{c}{2(1-c\omega)}}{1-\frac{c(1-2\omega)}{2(1-c\omega)}} = \frac{c}{2-c} < 1.$$

□

Corollary 5.3.6.1 suggests that the family of preconditioners $\mathbf{P}_n^{-1}$ based on the iteration matrix of the modified source iteration method is able to cluster eigenvalues of the preconditioned matrix $\mathbf{P}_n^{-1}(\mathbf{I}-\mathbf{G}_\omega)$ into a disc centred at 1 of radius $\left(\frac{c}{2-c}\right)^n$. It is possible that using a different polynomial mapping function f_n in the proof of Theorem 5.3.6 is able to more tightly cluster eigenvalues about 1; however, analytically describing the image of the original bounding disc under such a mapping is difficult. Tighter clustering, and thus better preconditioners, may be achievable if sharper bounding discs can be found; this will be addressed shortly.

Since the choice of ω does not make a significant difference in the statement of Corollary 5.3.6.1, we shall take the case $\omega = 0$, where $\mathbf{P}_n^{-1}$ are preconditioners based on standard transport/source-iteration sweeps. The system of equations we actually want to precondition is

$$(\mathbf{T}-\mathbf{S})\,\mathbf{u} = \mathbf{f}.$$

The left-preconditioners for $\mathbf{T}-\mathbf{S}$ we shall consider are given by

$$\mathbf{P}_n^{-1} = \sum_{k=0}^{n-1} a_k^{(n)} \left(\mathbf{T}^{-1}\mathbf{S}\right)^k \mathbf{T}^{-1}, \tag{5.36}$$

and the right-preconditioners are given by

$$\mathbf{P}_n^{-1} = \mathbf{T}^{-1} \sum_{k=0}^{n-1} a_k^{(n)} \left(\mathbf{S}\mathbf{T}^{-1}\right)^k. \tag{5.37}$$

In each case, the coefficients $\{a_k^{(n)}\}_{k=0}^{n-1}$ are given in the statement of Theorem 5.3.6 for a given (estimate of the) spectral disc centre a. One estimate for a is given in the proof of Corollary 5.3.6.1 - specifically, by the choice $a = \frac{c}{2}$, where c denotes the global scattering ratio and the spectral radius estimate for the classical source-iteration operator.

Heuristic refinement of the source-iteration spectral disc centre

A more accurate estimate of the spectral disc centre for the classical source-iteration operator is given by $a = \frac{c_l c}{2}$, where c_l denotes the following parameter:

$$c_l = 1 - \frac{1-e^{-\lambda L}}{\lambda L}. \tag{5.38}$$

Here, L denotes a characteristic length-scale (for example, the size of the spatial domain or the mesh size parameter h) and $\lambda = \min_{\mathbf{x}\in\Omega}(\alpha(\mathbf{x}) + \beta(\mathbf{x}))$ denotes the least value

of the macroscopic total cross-section. We will provide a motivation for the use of this parameter using probabilistic arguments.

Consider a particle at position $\mathbf{x} \in \Omega$ travelling in direction $\boldsymbol{\mu} \in \mathbb{S}$ has just undergone a scattering event. In order for the particle to remain in the system, two conditions must hold:

- The particle must travel a distance s along $\boldsymbol{\mu}$ such that $\mathbf{x} + s\boldsymbol{\mu} \in \Omega$; that is, the next (potential) interaction with the medium must occur inside the spatial domain;
- The particle must not be absorbed as a result of interacting with the medium.

The scattering ratio c addresses the probability of the second event occuring. We shall attempt to address the first event.

Let Ω be contained within a d-dimensional box $\mathcal{B}^d$ with side length h, understood as the characteristic length-scale of Ω. Without loss of generality, we shall consider $\Omega \subseteq \mathcal{B}^d = \left[-\frac{h}{2}, \frac{h}{2}\right]^d$. We will define three independent random variables for a particle's initial position, initial trajectory and track length between scattering events by

$$\begin{aligned} X &\sim U\left(\mathcal{B}^d\right), \\ \Theta &\sim U\left(\mathbb{S}\right), \\ S &\sim \text{Exp}\left(\lambda\right). \end{aligned}$$

Here, $\lambda = \alpha + \beta$ denotes the macroscopic total cross-section of the medium - for simplicity, we have assumed that the medium is homogeneous so that λ is constant. Since X and Θ are independent uniform random variables, we have that their joint density function is given by $f_{X,\Theta}(\mathbf{x}, \boldsymbol{\mu}) = f_X(\mathbf{x}) f_\Theta(\boldsymbol{\mu}) = \frac{1}{h^d |\mathbb{S}|}$ for $\mathbf{x} \in \mathcal{B}^d$ and $\boldsymbol{\mu} \in \mathbb{S}$.

Suppose a particle travels in a direction $\boldsymbol{\mu} = (\mu_i)_{i=1}^d \in \mathbb{S}$ (with $||\boldsymbol{\mu}||_2 = 1$) from an initial position $\mathbf{x} \in \mathcal{B}^d$; see Figure 5.2. The maximum distance s that the particle can travel along this trajectory before it hits the boundary of $\mathcal{B}^d$ is given by $\min\{s_i\}_{i=1}^d$, where s_i is the largest track length satisfying

$$|x_i + s_i \mu_i| \leq \frac{h}{2}.$$

It can be shown that

$$s_i = \frac{h}{2|\mu_i|} - \frac{x_i}{\mu_i},$$

so the maximum track length s a particle with initial position $\mathbf{x}$ can travel in a direction $\boldsymbol{\mu}$ and still remain inside $\mathcal{B}^d$ is given by

$$s_{max} = \min \left\{ \frac{h}{2|\mu_i|} - \frac{x_i}{\mu_i} \right\}_{i=1}^d.$$

Next, we shall consider the probability that the track length does not exceed s_{max} for a given $(\mathbf{x}, \boldsymbol{\mu})$ pair. Since $S \sim \text{Exp}\left(\lambda\right)$, we know that the probability density function

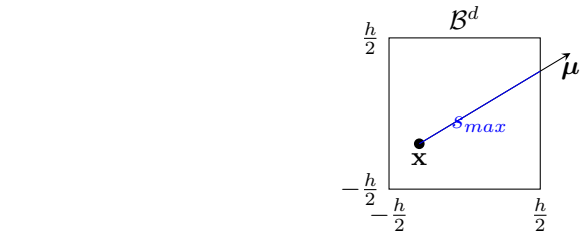

Figure 5.2: Schematic diagram showing the physical interpretation of s_{max} for a given choice of initial position $\mathbf{x}$, direction $\boldsymbol{\mu}$ and bounding box side length h in the two-dimensional case ($d = 2$).

for the random track length variable is given by

$$f_S(s) = \lambda \exp(-\lambda s).$$

To determine the probability that the track length does not exceed s_{max}, one integrates f_S over all s satisfying $s \leq s_{max}$. This probability is given by

$$\begin{aligned} P(S \leq s_{max}) &= \int_0^{s_{max}} f_S(s) \ \mathrm{d}s \\ &= \int_0^{s_{max}} \lambda \exp(-\lambda s) \ \mathrm{d}s \\ &= 1 - \exp(-\lambda s_{max}). \end{aligned}$$

For each $d \in \mathbb{N}$, we define the probability $c_l^{(d)} = P\left(X - S\Theta \in \mathcal{B}^d\right)$. That is, $c_l^{(d)}$ denotes the probability that a particle with a random initial position in $\mathcal{B}^d$ and a random trajectory in $\mathbb{S}$ will not leave $\mathcal{B}^d$ before its next interaction with the medium. By conditioning on the initial position and trajectory, we have:

$$\begin{aligned} c_l^{(d)} &= \int_{\mathbb{S}} \int_{\mathcal{B}^d} P\left(X - S\Theta \in \mathcal{B}^d | X = \mathbf{x}, \Theta = \boldsymbol{\mu}\right) f_{X,\Theta}(\mathbf{x}, \boldsymbol{\mu}) \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \\ &= \frac{1}{h^d |\mathbb{S}|} \int_{\mathbb{S}} \int_{\mathcal{B}^d} P\left(\mathbf{x} - S\boldsymbol{\mu} \in \mathcal{B}^d\right) \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \\ &= \frac{1}{h^d |\mathbb{S}|} \int_{\mathbb{S}} \int_{\mathcal{B}^d} P\left(S \leq \min\left\{ \frac{h}{2|\mu_i|} - \frac{x_i}{\mu_i} \right\}_{i=1}^{d}\right) \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu} \\ &= 1 - \frac{1}{h^d |\mathbb{S}|} \int_{\mathbb{S}} \int_{\mathcal{B}^d} \exp\left(-\lambda \min\left\{ \frac{h}{2|\mu_i|} - \frac{x_i}{\mu_i} \right\}_{i=1}^{d}\right) \ \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{\mu}. \end{aligned}$$

The right-hand-side of this expression above is very cumbersome to integrate for $d > 1$, so we shall instead treat the special case $d = 1$. The probability $c_l^{(1)}$ may be

obtained in closed form by

$$\begin{aligned} c_l^{(1)} &= 1 - \frac{1}{h|\mathbb{S}|}\int_{\mathbb{S}}\int_{\mathcal{B}^1} \exp\left(-\lambda\left(\frac{h}{2|\mu|} - \frac{x}{\mu}\right)\right)\,\mathrm{d}x\,\mathrm{d}\mu \\ &= 1 - \frac{1}{2h}\int_{-h/2}^{h/2}\left[\exp\left(-\lambda\left(\frac{h}{2} - x\right)\right) + \exp\left(-\lambda\left(\frac{h}{2} + x\right)\right)\right]\,\mathrm{d}x \\ &= 1 - \frac{1}{h}\exp\left(-\frac{\lambda h}{2}\right)\int_{-h/2}^{h/2}\cosh\left(\lambda x\right)\,\mathrm{d}x \\ &= 1 - \frac{1}{\lambda h}\exp\left(-\frac{\lambda h}{2}\right)\left[\sinh(\lambda x)\right]_{-h/2}^{h/2} \\ &= 1 - \frac{1 - e^{-\lambda h}}{\lambda h}. \end{aligned} \tag{5.39}$$

The leakage ratio $c_l^{(d)}$ attempts to approximate the proportion of particles which do not escape the bounding box between two consecutive iterations with the medium. The dimensionless quantity λh, which we will refer to as the *cell aspect ratio* or *optical thickness*, denotes the number of mean-track-lengths required to traverse the width of the bounding box. The assumptions of uniformly-distributed initial positions and trajectories are made for the idealised case of a constant fluence.

Figure 5.3 shows the dependence of $c_l^{(1)}$ on λh. The system is leaky when $c_l^{(1)}$ is small (compared to the size of the domain) - this occurs when λh is also small. This matches our intuition of what we mean by a "leaky" system - particles in a leaky system undergo very few interactions with the medium to escape the spatial domain.

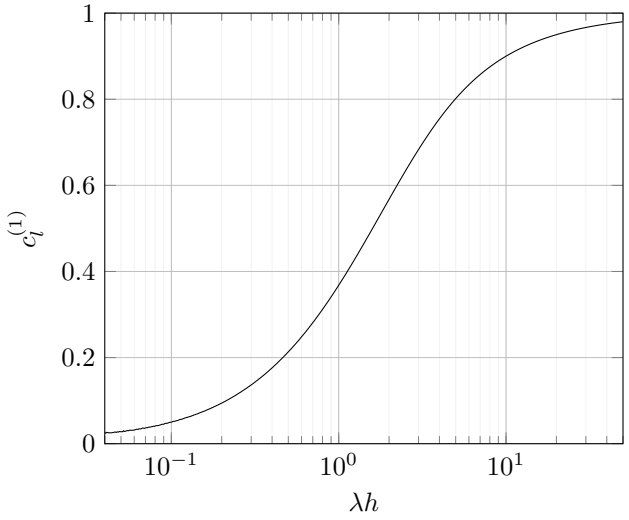

Figure 5.3: Plot of leakage ratio $c_l^{(1)}$ against the dimensionless quantity λh.

We now relate this result to the problem of finding a refined upper bound $\bar{c}$ for the spectral radius ρ of the standard source iteration operator, for which we have previously only obtained an upper bound of c. As discussed earlier, a particle can only remain in the system after a transport sweep if it does not escape through the domain or be removed via an absorption process inside the domain. The quantity $c_l^{(d)}$ describes the probability of the former process occurring and c describes the probability of the latter

process happening. The product

$$\bar{c} = c_l c \tag{5.40}$$

therefore represents a heuristic refinement of c that attempts to cater for the possibility of fluence leakage out of the spatial domain, where we have defined c_l by

$$c_l = \max_{\mathbf{x}\in\Omega} \left(1 - \frac{1 - \exp\left(-(\alpha(\mathbf{x}) + \beta(\mathbf{x}))h\right)}{(\alpha(\mathbf{x}) + \beta(\mathbf{x}))h}\right) \tag{5.41}$$

as a generalisation to mono-energetic problems with non-constant-coefficient data.

By replacing c with $\bar{c}$ in any of the *a posteriori* error estimates outlined in Chapter 5, we can obtain new (and potentially sharper) *a posteriori* error indicators. For example, by making such a replacement in the result of Theorem 5.2.7, we get the following error indicator for the DGFEM-energy norm error incurred by the modified source iteration:

$$|||u_h^{(n)} - u_h|||_{DG} \lessapprox r(\omega)\sqrt{\frac{\bar{c}}{1-\bar{c}}}||\beta^{\frac{1}{2}}(u_h^{(n)} - u_h^{(n-1)})||_{L^2(\mathcal{D})}. \tag{5.42}$$

We will conclude by briefly considering the evaluation of $c_l^{(d)}$ for $d > 1$, which we earlier remarked requires the evaluation of a cumbersome integral. We may alternatively take a stochastic approach to approximating $c_l^{(d)}$ for any λ, h and d by sampling the random variable $Y = X + S\Theta$ and checking whether it lies within the box $\mathcal{B}^d$. Algorithm 7 can be used to estimate $c_l^{(d)}$.

Algorithm 7 Estimation of the leakage parameter $c_l^{(d)}$.

Fix number of samples N
$c_l^{(d)} \leftarrow 0$
for $n = 1, \ldots, N$ **do**
▷ *Sample from uniform distribution on* $\left(-\frac{h}{2}, \frac{h}{2}\right)^d$ *- here,* $U\left(\left(-\frac{h}{2}, \frac{h}{2}\right)\right)$ *denotes the uniform distribution on* $\left(-\frac{h}{2}, \frac{h}{2}\right)$
    Sample $\{x_i\}_{i=1}^d$ from $U\left(\left(-\frac{h}{2}, \frac{h}{2}\right)\right)$ and construct $\mathbf{x} = (x_i)_{i=1}^d$
▷ *Sample from uniform distribution on* $\mathbb{S}$ *- here,* $N(0,1)$ *denotes the normal distribution with mean 0 and variance 1*
    Sample $\{\mu_i\}_{i=1}^d$ from $N(0,1)$ and construct $\boldsymbol{\mu} = \left(\frac{\mu_i}{\sqrt{\sum_{j=1}^d \mu_j^2}}\right)_{i=1}^d$
▷ *Sample from exponential distribution with rate parameter* λ *- here,* $U((0,1))$ *denotes the uniform distribution on* $(0,1)$
    Sample t from $U(0,1)$ and construct $s = -\frac{\log t}{\lambda}$
    $\mathbf{y} \leftarrow \mathbf{x} + s\boldsymbol{\mu}$
    if $||\mathbf{y}||_\infty < \frac{h}{2}$ **then**
        $c_l^{(d)} \leftarrow c_l^{(d)} + 1$
    end if
end for
$c_l^{(d)} \leftarrow \frac{c_l^{(d)}}{N}$

5.4 Mono-Energetic Numerical Experiments

5.4.1 Rayleigh Scattering

This test problem seeks to compare the source iteration, modified source iteration and right-preconditioned GMRES methods applied to the linear system resulting from a DGFEM discretisation of the two-dimensional constant-coefficient mono-energetic LBTE on a space-angle domain $\mathcal{D} = (0, L)^2 \times \mathbb{S}$:

$$\boldsymbol{\mu} \cdot \nabla_{\mathbf{x}} u(\mathbf{x}, \boldsymbol{\mu}) + \lambda u(\mathbf{x}, \boldsymbol{\mu}) = c\lambda \int_{\mathbb{S}} \theta(\boldsymbol{\mu} \cdot \boldsymbol{\mu}') u(\mathbf{x}, \boldsymbol{\mu}') \ \mathrm{d}\boldsymbol{\mu}' + f(\mathbf{x}, \boldsymbol{\mu}) \quad \text{in } \mathcal{D},$$
$$u(\mathbf{x}, \boldsymbol{\mu}) = g(\mathbf{x}, \boldsymbol{\mu}) \quad \text{on } \partial\mathcal{D},$$

for some $\lambda, c, L > 0$. Here, the forcing data is chosen to be $f = g = 1$ and the differential scattering cross-section is selected to be the following function of the deflection cosine $\cos\varphi = \boldsymbol{\mu} \cdot \boldsymbol{\mu}'$:

$$\theta(\cos\varphi) = \frac{1 + \cos^2\varphi}{3\pi}.$$

Notice that the integral of $\theta(\cos\varphi)$ over the angular domain $\mathbb{S} = S^1$ is equal to 1.

The scattering ratio c, the macroscopic total cross-section λ and the length-scale L of the spatial domain are parameters which we will vary. The triplet of parameters (c, λ, L) will affect the rate of convergence of each iterative method, as well as the effectivities of the *a posteriori* error estimates presented in Theorems 5.2.7 and 5.2.8. Henceforth, we shall take $c \in \{\frac{1}{10}, \frac{3}{10}, \frac{5}{10}, \frac{7}{10}, \frac{9}{10}\}$, $\lambda \in \{\frac{1}{10}, 1, 10\}$ and $L \in \{\frac{1}{5}, 2, 20\}$.

A coarse discretisation of the space-angle domain is implemented. Rather than opting to discretise the angular domain as in Chapter 3.2.2, we instead discretise the angular domain into 16 equally-spaced discrete ordinate directions, which we remark is equivalent to a piecewise-constant discretisation on a uniform mesh of the unit circle. The spatial discretisation is performed by first constructing a triangular mesh the reference domain $(0, 1)^2$ using 312 triangular elements, and then scaling the vertices of the resulting mesh by the factor L. The resulting meshes are equipped with a discontinuous piecewise-linear spatial finite element space.

We stress that the coarseness of the spatial and angular discretisations, as well as the specification of a different angular mesh, do not invalidate the convergence results proven earlier since no assumptions on the space-angle mesh were made. Indeed, all of the results presented in this chapter are independent of discretisation parameters - this is reflected in numerical tests not reported here.

For each (c, λ, L)-triplet, we will find the exact solution u_h to the discrete equations by directly computing the coefficient vector $\mathbf{u} = (\mathbf{T} - \mathbf{S})^{-1}\mathbf{f}$, as well as sequences of approximations $\{u_h^{(n)}\}_{n \geq 0}$ generated by the following linear solvers:

- **Source iteration (SI):** starting from the initial guess $u_h^{(0)} = 0$, the iteration (5.9) is used to construct the sequence of approximations $\{u_h^{(n)}\}_{n \geq 0}$. Furthermore, we

will also compute the DG-energy norm errors $|||u_h - u_h^{(n)}|||_{DG}$ as well as the *a posteriori* error estimates given in Theorem 5.2.7 with parameter $\omega = 0$.

- **Modified source iteration (MSI(ω)):** starting from the initial guess $u_h^{(0)} = 0$, the iteration (5.17) is used to construct the sequence of approximations $\{u_h^{(n)}\}_{n\geq 0}$ with the choice of relaxation parameter $\omega \in \{\frac{1}{10}, \frac{3}{10}, \frac{5}{10}, \frac{7}{10}, \frac{9}{10}\}$. Furthermore, we will also compute the DG-energy norm errors $|||u_h - u_h^{(n)}|||_{DG}$ as well as the *a posteriori* error estimates given in Theorem 5.2.7.

- **Right-preconditioned GMRES (RPGMRES-T(n)):** starting from the initial guess $u_h^{(0)} = 0$, GMRES is used to construct the sequence of approximations $\{u_h^{(n)}\}_{n\geq 0}$, with right-preconditioners $\mathbf{P}_n^{-1}$ defined in (5.36) for $n \in \{1, 2, 3\}$. The right-preconditioned GMRES method will be executed as in Theorem 5.3.5. Furthermore, we will also compute the DG-energy norm errors $|||u_h - u_h^{(n)}|||_{DG}$ as well as the *a posteriori* error estimates given in Theorem 5.2.8 - the latter is a by-product of the GMRES implementation. While it is expected that left-, right- and split-preconditioning strategies all share similar convergence properties [83], only the right-preconditioning strategy ensures that the correct norm of the residual vector is computed for the purposes of *a posteriori* error estimation.

The number of iterations will be taken as a surrogate for the total CPU time taken for all algorithms to converge to the specified tolerance. It has been observed that a single iteration of SI and MSI(ω) takes approximately the same amount of CPU time, owing to the similarity of the actions of the transport operators $\mathbf{T}^{-1}$ and $(\mathbf{T} - \omega\mathbf{M})^{-1}$ and the scattering operations $\mathbf{S}$ and $\mathbf{S} - \omega\mathbf{M}$. A single iteration of RPGMRES-T(1) is slightly more expensive than a single iteration of SI due to:

- the additional actions of the Cholesky factors $\mathbf{L}$ and $\mathbf{L}^{-1}$ on vectors, and
- an orthogonalisation step at each iteration of RPGMRES-T(1).

However, it has been observed that the total CPU time taken to perform these actions is comparable to the total time taken to perform the action of $\mathbf{T}^{-1}$ and significantly less than the CPU time to perform the action of $\mathbf{S}$ for moderately-size problems. Finally, a single iteration of RPGMRES-T(n) requires n evaluations of the actions of $\mathbf{T}^{-1}$ and $\mathbf{S}$, and so we have that the CPU time taken for a single iteration of RPGMRES-T(n) is approximately n times the CPU time taken for a single iteration of RPGMRES-T(1).

Finally, we will define the effectivity of an *a posteriori* error estimator as follows. Let u_h be the exact solution of the discrete problem and $\hat{u}_h$ an approximation to u_h obtained by premature termination of any of the above solvers. Assume that an *a posteriori* error estimate $\mathcal{E}(\hat{u}_h)$ of the DG-energy norm solver error $|||u_h - \hat{u}_h|||_{DG}$ can be computed. The *effectivity* of the error estimate $\mathcal{E}(\hat{u}_h)$ is defined to be the ratio $\mathcal{E}(\hat{u}_h)/|||u_h - \hat{u}_h|||_{DG}$ and can be interpreted as a measure of how much the *a posteriori* error estimate over- or

under-estimates the true solver error. The closer the effectivity of the *a posteriori* error estimate is to 1, the better the estimate is; since we have provided guaranteed upper bounds on the DG-energy norm error, the effectivity of the *a posteriori* error estimates presented below should all have effectivities greater than or equal to 1.

Test A: SI vs MSI(ω) Figures 5.4 and 5.6 display the convergence behaviours of source iteration and modified source iteration applied to the benchmark problem with $(\lambda, L) = (10, 20)$ and $(\lambda, L) = (1, 2)$ respectively. Each plot is divided into the cases depending on the scattering ratio $c \in \{\frac{1}{10}, \frac{3}{10}, \frac{5}{10}, \frac{7}{10}, \frac{9}{10}\}$. Both figures show that modified source iteration converges faster than standard source iteration for any choice of the relaxation parameter $0 < \omega \leq \frac{1}{2}$, and that the fastest convergence of modified source iteration for ω varying over this range is consistently attained when $\omega = \frac{1}{2}$. A similar result was predicted in the previous discussion of Theorem 5.2.6 - we recall that the choice $\omega = \frac{1}{2}$ minimises the contraction factor appearing in that theorem.

However, Figures 5.4 and 5.6 also show that the behaviour of modified source iteration for $\omega > \frac{1}{2}$ is highly dependent on the parameter triplet (c, λ, L). Specifically, Figure 5.4 shows that MSI($\frac{9}{10}$) diverges for $c \in \{\frac{7}{10}, \frac{9}{10}\}$ and MSI($\frac{7}{10}$) diverges for $c = \frac{9}{10}$ for the parameter choice $(\lambda, L) = (10, 20)$; however, Figure 5.6 shows that both MSI($\frac{7}{10}$) and MSI($\frac{9}{10}$) converge for all tested values of c for the parameter choice $(\lambda, L) = (1, 2)$. While the precise dependence of the convergence of modified source iteration on the parameters λ and L will not be discussed here, it should be noted that Theorem 5.2.6 does not guarantee that MSI(ω) will converge for any $0 < c < 1$ when $\omega > \frac{1}{2}$.

Figures 5.4 and 5.6 demonstrate that MSI(ω) can converge faster with $\omega > \frac{1}{2}$ than $\omega = \frac{1}{2}$ for some choice of c, λ and L, but that the latter choice of ω yields the fastest convergence rate for which MSI(ω) is guaranteed to converge consistently for all choices of c, λ and L. While we have not investigated why this behaviour occurs, we believe that a further study of the eigenvalues of the modified source iteration operator $\mathbf{G}_\omega = (\mathbf{T} - \omega\mathbf{M})^{-1}(\mathbf{S} - \omega\mathbf{M})$ may prove insightful. From Theorem 5.3.3, we deduced that the spectral radius of $\mathbf{G}_\omega$ is bounded above by a function of ω and c, which can be used to predict when MSI(ω) will converge. A finer analysis may show that (an upper bound for) the spectral radius of $\mathbf{G}_\omega$ may also depend on λ and L, which may allow us to relax the conditions on ω, c, λ and L for which MSI(ω) is guaranteed to converge.

Figures 5.5 and 5.7 show the effectivities of the *a posteriori* error estimates employed by source iteration and modified source iteration. We remark that the *a posteriori* error estimates for the solver error induced by modified source iteration in Theorem 5.2.7 have effectivities close to 1 for the range of test problems studied and for $0 \leq \omega \leq 1$. However, the effectivity of the error estimate in Theorem 5.2.7 deteriorates for $\omega > \frac{1}{2}$. It is important to note that, in some cases, the effectivity drops suddenly after a certain number of iterations - this is an artifact of the true solver error dropping to machine

precision.

The effectivities of the *a posteriori* error estimates studied here are also sensitive to the triplet of parameters (c, λ, L). Qualitatively, the effectivity of the error estimate in Theorem 5.2.7 becomes larger both as $c \to 1$ (as can be seen in Figure 5.7) and as $\lambda, L \to 0$ - the latter case will be studied in Test C.

Test B: SI vs RPGMRES Figure 5.8 and 5.11 display the convergence behaviours of source iteration and right-preconditioned GMRES applied to the benchmark problem with $(\lambda, L) = (10, 20)$ and $(\lambda, L) = (1, 2)$ respectively. The preconditioners $\mathbf{P}_n^{-1}$ for $n \in \{1, 2, 3\}$ employed for the right-preconditioned GMRES are based on multiple transport sweeps and are defined as in Theorem 5.3.6 with the choice of spectral centre $a = 0$. Each plot is divided into the cases depending on the scattering ratio $c \in \{\frac{1}{10}, \frac{3}{10}, \frac{5}{10}, \frac{7}{10}, \frac{9}{10}\}$. Both figures show that GMRES converges faster than standard source iteration, and that the convergence rate of multiple-transport-preconditioned GMRES improves as the number of transport sweeps n increases. This result was predicted in the discussion of Theorem 5.9. We also observe more rapid convergence of both source iteration and right-preconditioned GMRES upon reducing the magnitudes of λ and L - this will be investigated shortly.

It is worth discussing the storage requirements of source iteration, modified source iteration and GMRES. At the k^{th} step of source iteration or modified source iteration, a constant number of vectors of storage (independent of k) are required to generate the iterate $u_h^{(k)}$. In contrast, GMRES requires the storage of k Krylov vectors in order to generate the iterate $u_h^{(k)}$. Therefore, while each step of RPGMRES-T(n) takes approximately the same amount of CPU time to perform, it requires the storage of an extra vector. For very large problems with highly-resolved spatial and angular grids, the number of GMRES steps that can be taken may be significantly limited by the amount of available storage.

With this in mind, one might choose the number of transport sweeps n per GMRES iteration to be large, as this results in a large reduction in the *a posteriori* solver error estimate (and the DG-energy norm solver error) per additional vector of storage. However, the total number of transport sweeps required to reduce the *a posteriori* error estimate below a given user-defined tolerance actually increases with n. This is shown in Figure 5.9, which is a rescaled version of Figure 5.8 in which the x-axis is scaled by the number of transport sweeps applied at each iteration.

Another approach to mitigate the demanding memory constraints is to perform RPGMRES-T(n) with restarting [83]. Given a restart length of k, the storage for k Krylov vectors is pre-allocated and RPGMRES-T(n) is ran until all Krylov vectors have been specified. One then constructs the approximate solution after k (inner) iterations and uses it as an initial guess for another k iterations of RPGMRES-T(n). In practice,

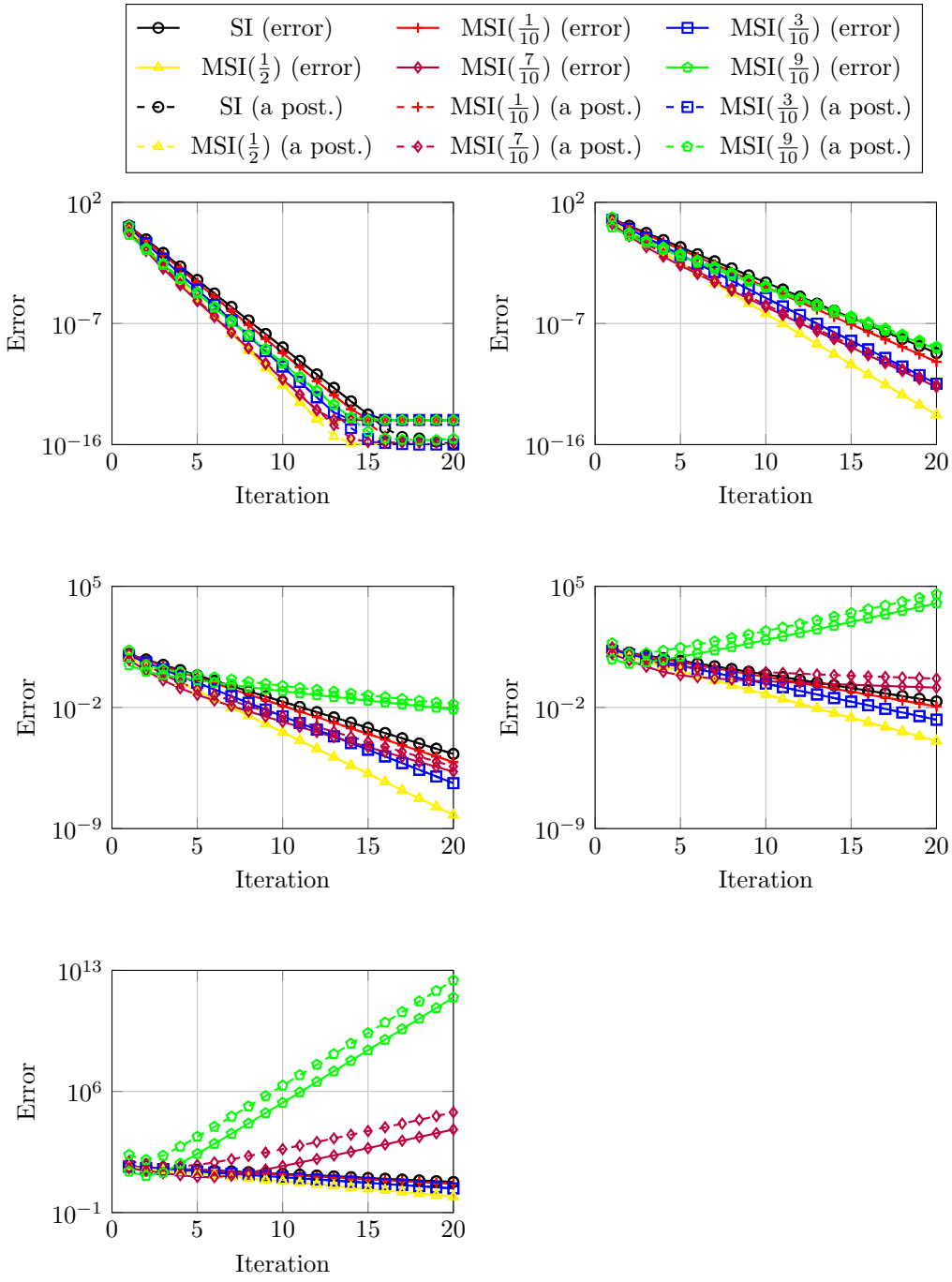

Figure 5.4: Convergence histories of source iteration and modified source iteration for a number of choices of relaxation parameters applied to the Rayleigh scattering problem for a range of scattering ratios. The macroscopic total cross-section and spatial domain length-scale are chosen to be $\lambda = 10$ and $L = 20$ respectively. Solid line: DG-energy norm error. Dashed line: *a posteriori* solver error estimate. Top-left: $c = \frac{1}{10}$. Top-right: $c = \frac{3}{10}$. Middle-left: $c = \frac{5}{10}$. Middle-right: $c = \frac{7}{10}$. Bottom-left: $c = \frac{9}{10}$.

good choices of k depend on the size of the linear system as well as the amount of memory available. Specifically, one should attempt to maximise the number of stored Krylov vectors as it is known that restarted GMRES is prone to *stagnation* [83], a phe-

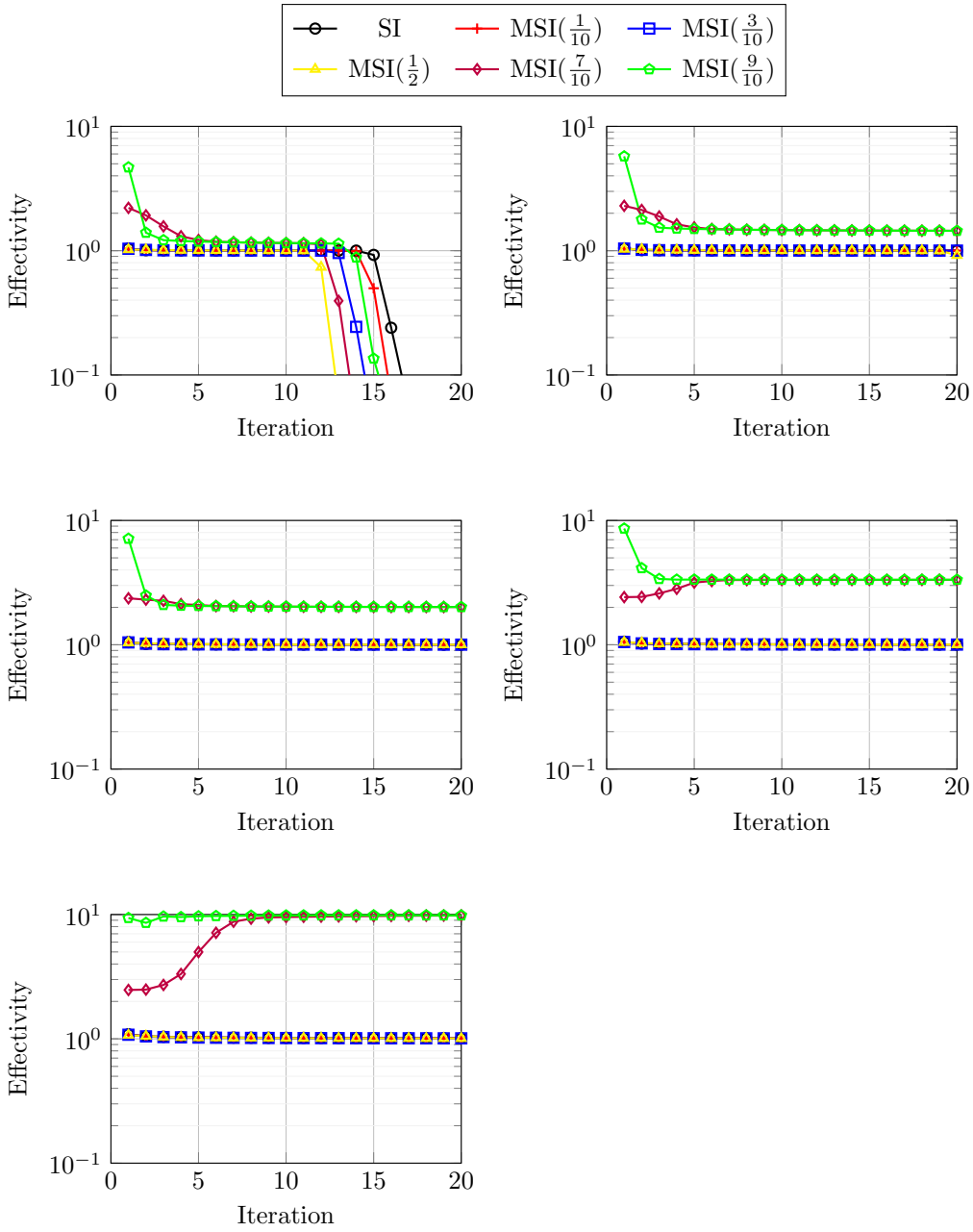

Figure 5.5: Effectivities of the *a posteriori* error estimates for source iteration and modified source iteration for a number of choices of relaxation parameters applied to the Rayleigh scattering problem for a range of scattering ratios. The macroscopic total cross-section and spatial domain length-scale are chosen to be $\lambda = 10$ and $L = 20$ respectively. Top-left: $c = \frac{1}{10}$. Top-right: $c = \frac{3}{10}$. Middle-left: $c = \frac{5}{10}$. Middle-right: $c = \frac{7}{10}$. Bottom-left: $c = \frac{9}{10}$.

nomenon describing the apparent slow initial convergence of GMRES after each restart. Stagnation can be alleviated with selecting a good preconditioner. While we have not investigated RPGMRES-T(n) with restarting, it is expected that stagnation is likely to be problematic only for small n.

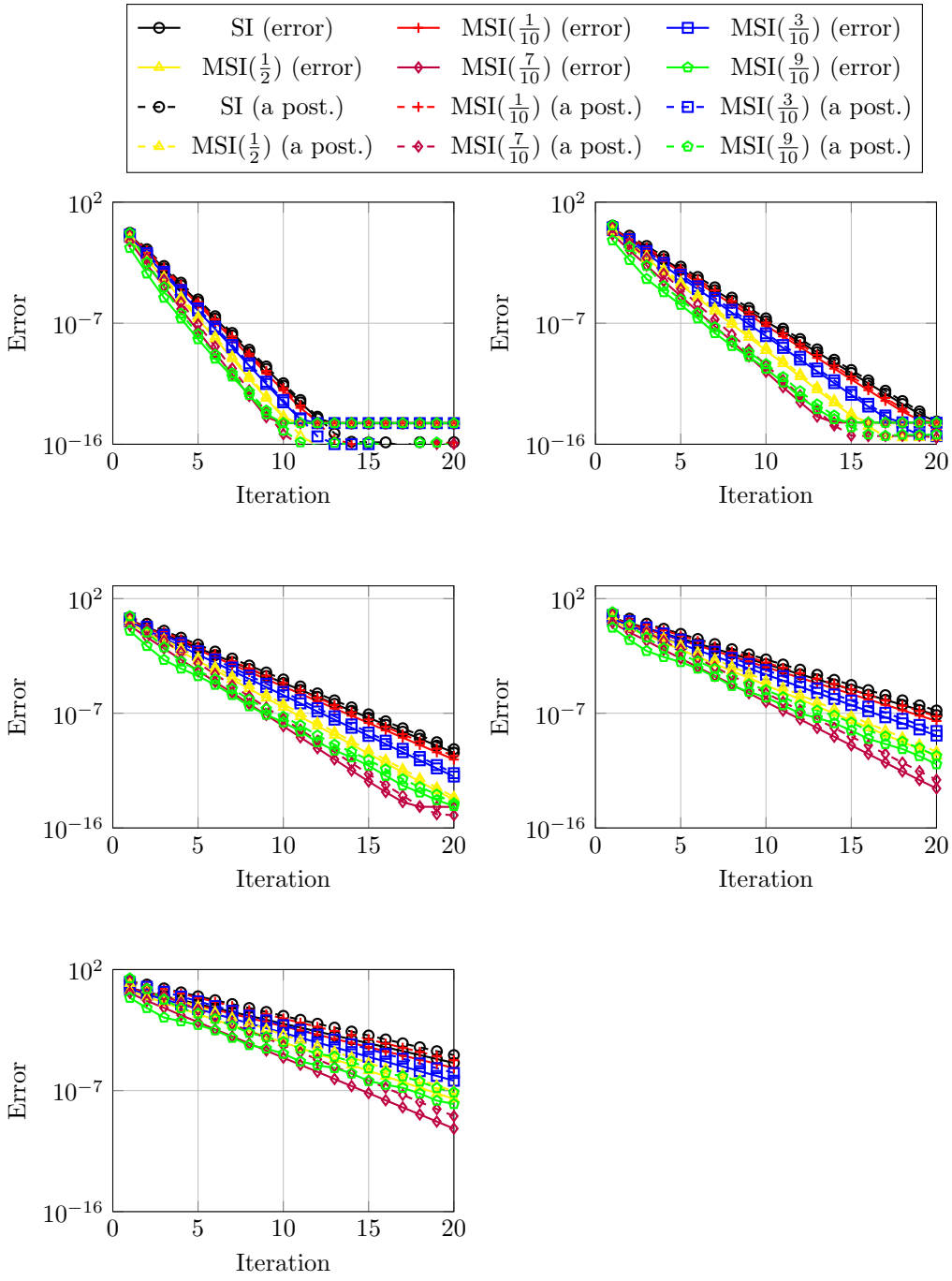

Figure 5.6: Convergence histories of source iteration and modified source iteration for a number of choices of relaxation parameters applied to the Rayleigh scattering problem for a range of scattering ratios. The macroscopic total cross-section and spatial domain length-scale are chosen to be $\lambda = 1$ and $L = 2$ respectively. Solid line: DG-energy norm error. Dashed line: *a posteriori* solver error estimate. Top-left: $c = \frac{1}{10}$. Top-right: $c = \frac{3}{10}$. Middle-left: $c = \frac{5}{10}$. Middle-right: $c = \frac{7}{10}$. Bottom-left: $c = \frac{9}{10}$.

Figures 5.10 and 5.12 show the effectivities of the *a posteriori* error estimates employed by source iteration and right-preconditioned GMRES. The effectivities of the source iteration *a posteriori* error estimates are the same as those displayed in Figures

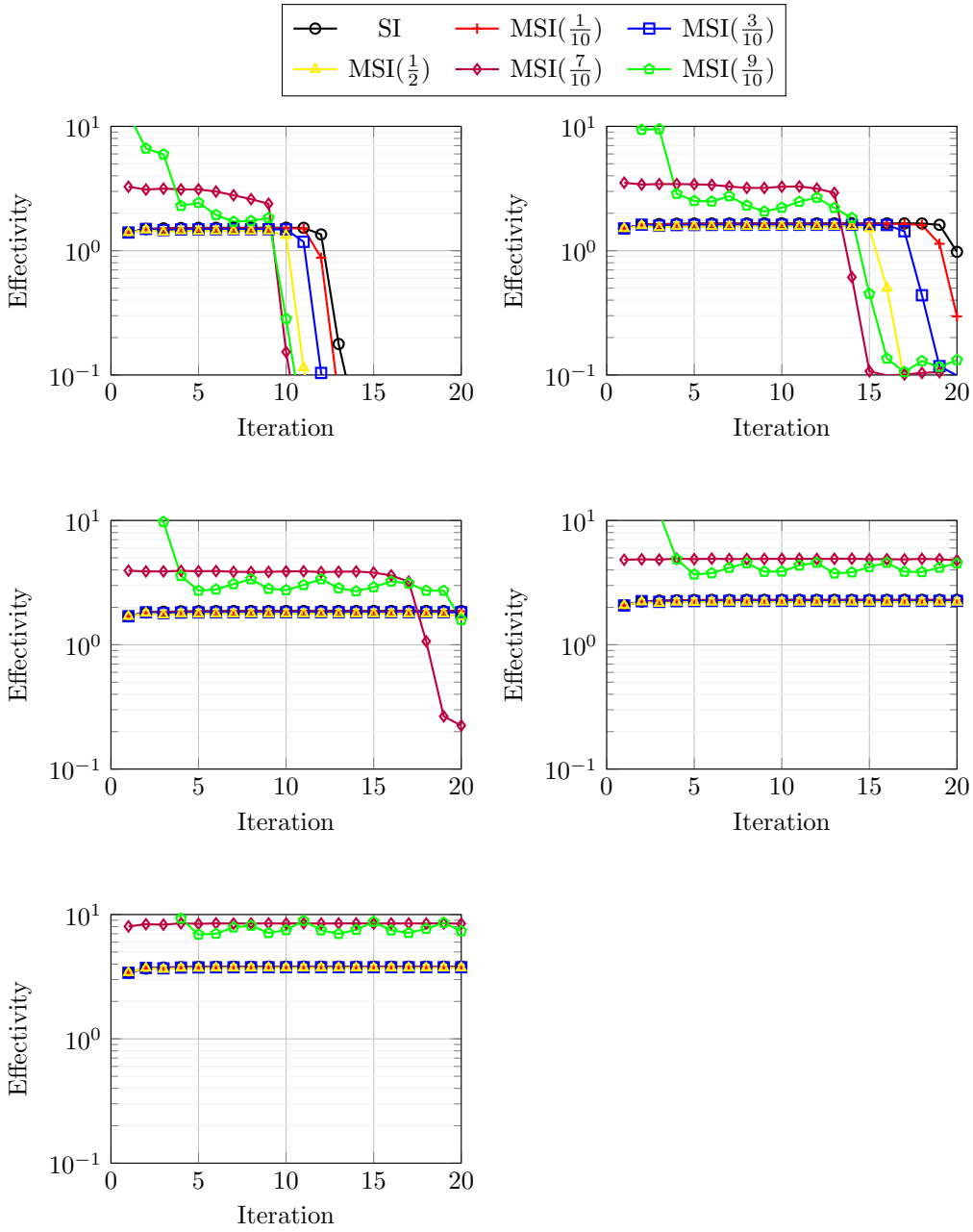

Figure 5.7: Effectivities of the *a posteriori* error estimates for source iteration and modified source iteration for a number of choices of relaxation parameters applied to the Rayleigh scattering problem for a range of scattering ratios. The macroscopic total cross-section and spatial domain length-scale are chosen to be $\lambda = 1$ and $L = 2$ respectively. Top-left: $c = \frac{1}{10}$. Top-right: $c = \frac{3}{10}$. Middle-left: $c = \frac{5}{10}$. Middle-right: $c = \frac{7}{10}$. Bottom-left: $c = \frac{9}{10}$.

5.5 and 5.7. It is clear that the effectivities of the *a posteriori* error estimates employed by the right-preconditioned GMRES method are slightly worse than those for both source iteration and modified source iteration, but are still reasonably close to 1 for the range of test problems studied. It is worth noting that the *a posteriori* error estimate in

Theorem 5.2.8 applies to more general settings than the error estimate in Theorem 5.2.7. We also expect that the *a posteriori* error estimate employed by the GMRES method is not sharp, as remarked after the proof of Theorem 5.2.8. The non-sharpness of the GMRES *a posteriori* error estimate worsens as both $c \to 1$ and $\lambda, L \to 0$ as before.

Test C: Effect of λ and L Figure 5.13 shows the behaviour of source iteration, modified source iteration using the optimal parameter choice $\omega = \frac{1}{2}$, and right-preconditioned GMRES with a single-sweep preconditioner for the model problem for each pair of parameters (λ, L) with $\lambda \in \{\frac{1}{10}, 1, 10\}$ and $L \in \{\frac{1}{5}, 2, 20\}$. The choice of scattering ratio is not relevant for the subsequent study, but is kept constant at $c = \frac{7}{10}$ for the results presented below.

The convergence of all three methods is fastest when both $\lambda = \frac{1}{10}$ and $L = \frac{1}{5}$ and slowest when both $\lambda = 10$ and $L = 20$. We also notice that the convergence of all three methods are similar for the following sets of parameter configurations:

- When $(\lambda, L) = (\frac{1}{10}, 2)$ and $(\lambda, L) = (1, \frac{1}{5})$, the DG-energy norm solver error of source iteration approaches machine precision after around 15 iterations. Moreover, the solver error of modified source iteration approaches machine precision after around 12 iterations, and the solver error of right-preconditioned GMRES is approximately the same order of magnitude after 6 iterations.

- When $(\lambda, L) = (\frac{1}{10}, 20)$, $(\lambda, L) = (1, 2)$ and $(\lambda, L) = (10, \frac{1}{5})$, the DG-energy norm solver error of source iteration has roughly decreased by the same factor over 20 iterations, as has the solver error of modified source iteration, and the solver error of right-preconditioned GMRES is approximately the same order of magnitude after 10 iterations.

- When $(\lambda, L) = (1, 20)$ and $(\lambda, L) = (10, 2)$, the DG-energy norm solver error of source iteration has roughly decreased by the same factor over 20 iterations, as have the solver error of modified source iteration and right-preconditioned GMRES.

In other words, the convergence rate of source iteration (for a fixed value of the scattering ratio) appears to be dependent on the dimensionless quantity λL; this is also true for modified source iteration and right-preconditioned GMRES. The quantity λL may be interpreted as the number of scattering events that a radiative particle is expected to undergo as it travels a distance L through the domain, assuming that the mean free path length (the average distance travelled by a particle between scattering interatctions) is λ^{-1}. When λL is small, all three methods converge rapidly, and when λL is large, all three methods converge slowly. The convergence plots for fixed values of λL in Figure 5.13 are all qualitatively similar to each other.

For all pairs of parameters (λ, L) tested, RPGMRES-T(1) converges faster than MSI($\frac{1}{2}$), which converges faster than SI. While the results are not presented here, it

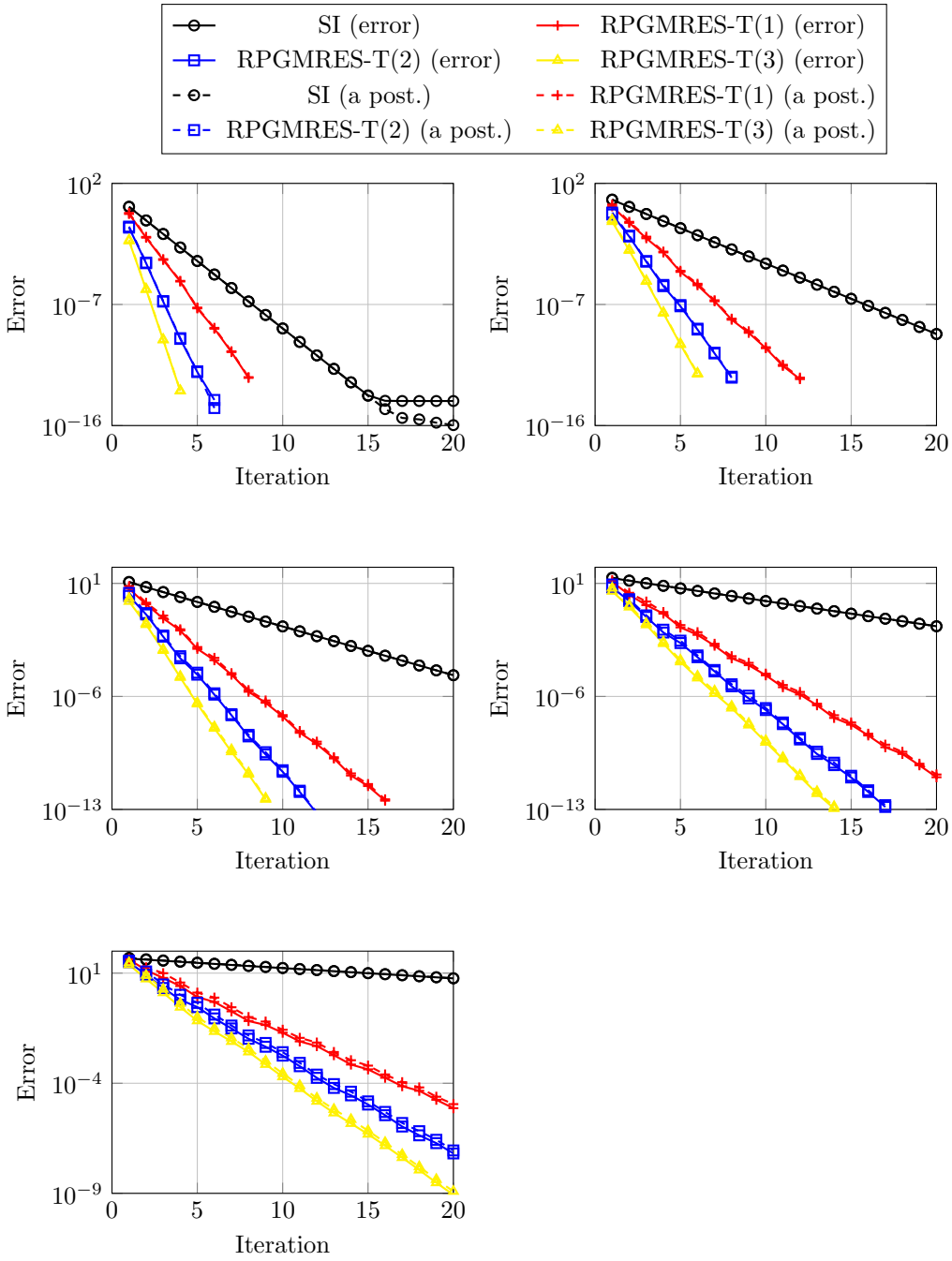

Figure 5.8: Convergence histories of source iteration and right-preconditioned GMRES for a number of choices of transport-based preconditioners applied to the Rayleigh scattering problem for a range of scattering ratios. The macroscopic total cross-section and spatial domain length-scale are chosen to be $\lambda = 10$ and $L = 20$ respectively. Solid line: DG-energy norm error. Dashed line: *a posteriori* solver error estimate. Top-left: $c = \frac{1}{10}$. Top-right: $c = \frac{3}{10}$. Middle-left: $c = \frac{5}{10}$. Middle-right: $c = \frac{7}{10}$. Bottom-left: $c = \frac{9}{10}$.

is expected that the behaviours of RPGMRES-T(n) for $n \geq 2$, as well as MSI(ω) for $\omega \neq \frac{1}{2}$, are consistent with Tests A and B. In the case of right-preconditioned GMRES,

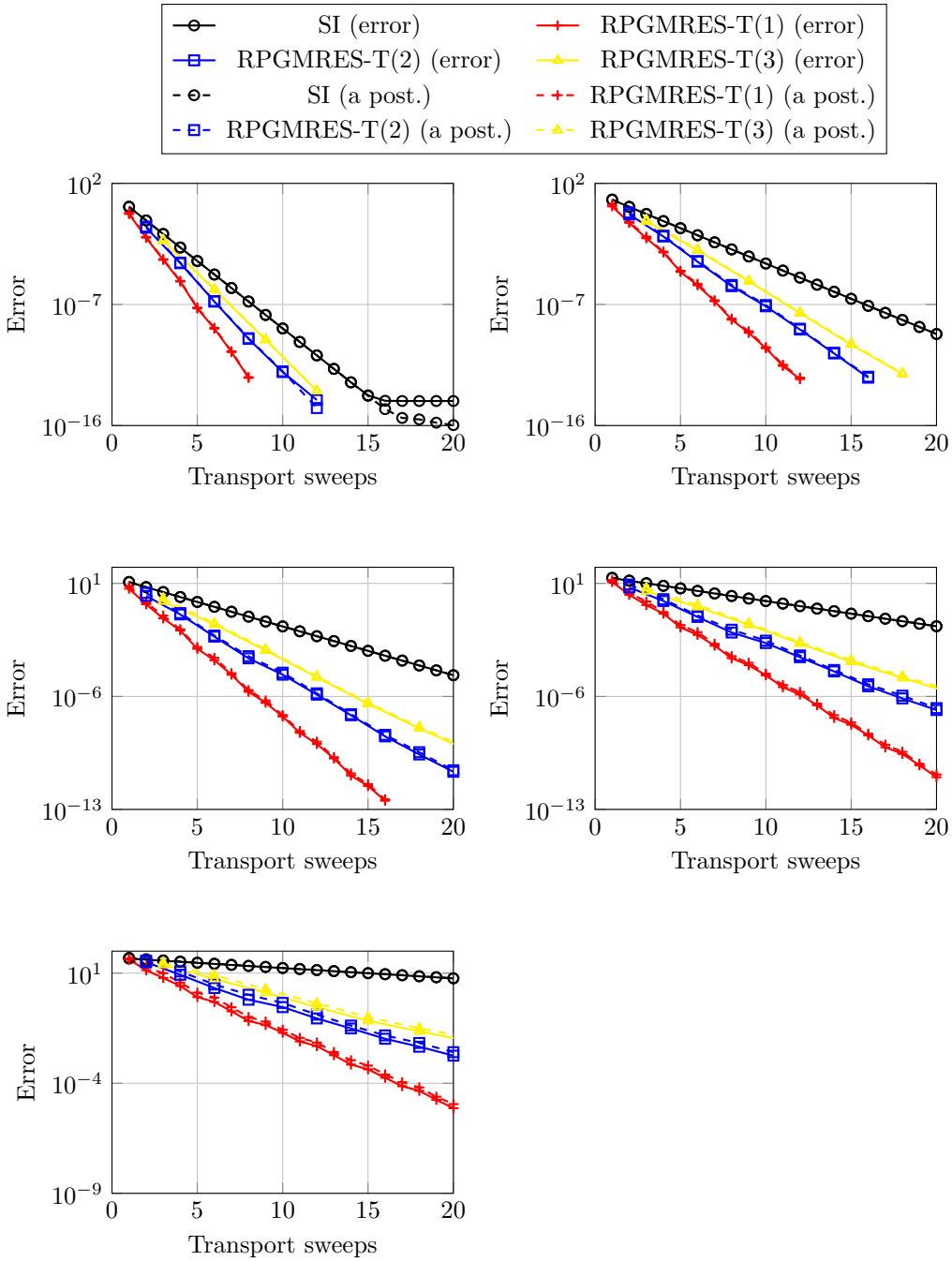

Figure 5.9: Figure 5.8 rescaled by the number of applications of the transport operator. Solid line: DG-energy norm error. Dashed line: *a posteriori* solver error estimate. Top-left: $c = \frac{1}{10}$. Top-right: $c = \frac{3}{10}$. Middle-left: $c = \frac{5}{10}$. Middle-right: $c = \frac{7}{10}$. Bottom-left: $c = \frac{9}{10}$.

we expect that RPGMRES-T(n) converges faster than RPGMRES-T(1) for $n \geq 2$ and all pairs (λ, L). In the case of modified source iteration, we expect that, for all pairs (λ, L), the fastest rate of convergence of MSI(ω) is attained at $\omega = \omega^*$ for some $\frac{1}{2} \leq \omega^* \leq 1$. However, it was observed in Test B that this optimal relaxation parameter is likely to be highly dependent on λ and L.

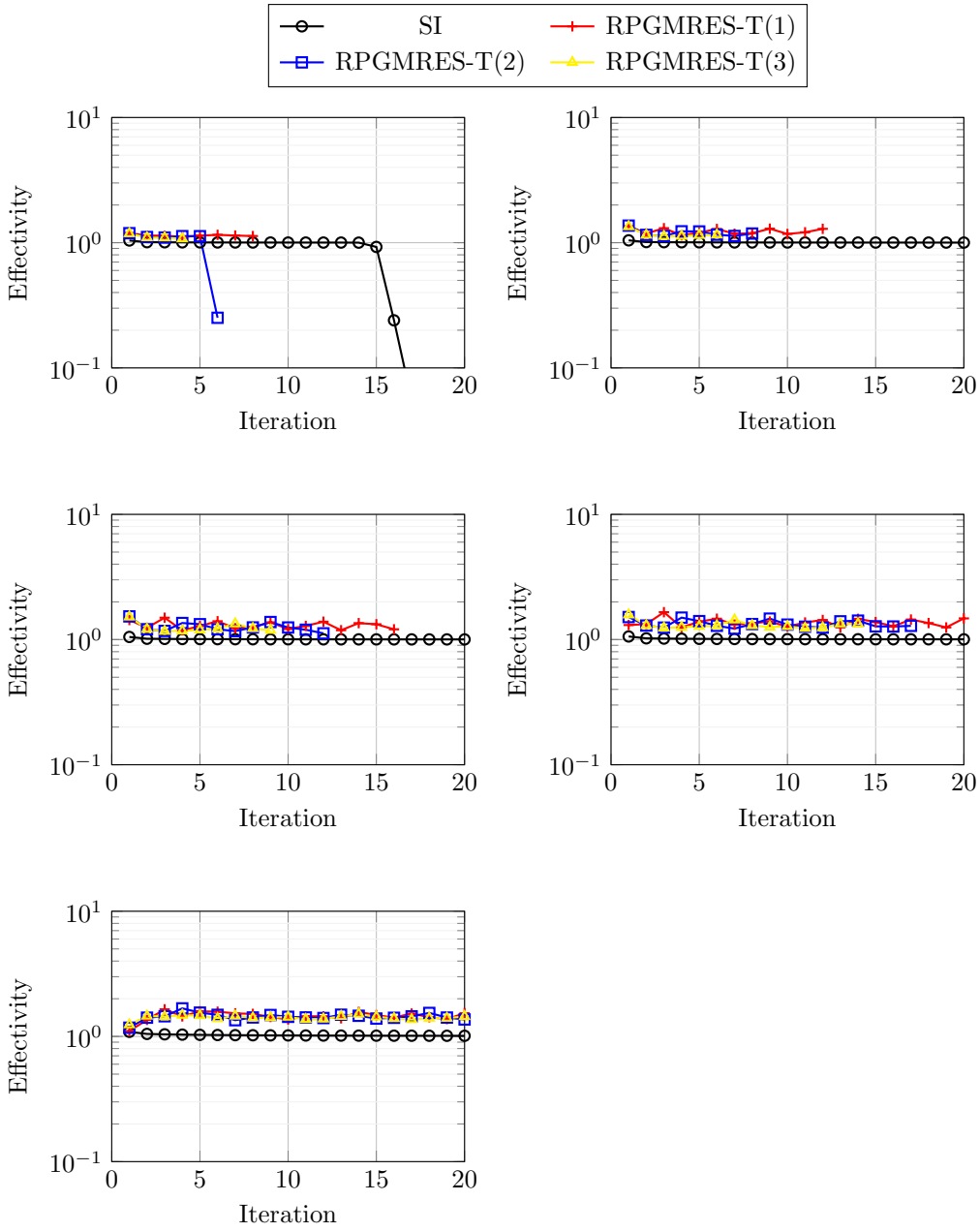

Figure 5.10: Effectivities of the *a posteriori* error estimates for source iteration and right-preconditioned GMRES for a number of choices of transport-based preconditioners applied to the Rayleigh scattering problem for a range of scattering ratios. The macroscopic total cross-section and spatial domain length-scale are chosen to be $\lambda = 10$ and $L = 20$ respectively. Top-left: $c = \frac{1}{10}$. Top-right: $c = \frac{3}{10}$. Middle-left: $c = \frac{5}{10}$. Middle-right: $c = \frac{7}{10}$. Bottom-left: $c = \frac{9}{10}$.

Figure 5.14 shows the effectivities of the *a posteriori* solver error estimates employed by the source iteration, modified source iteration and right-preconditioned GMRES methods. As was seen in Figure 5.13, the effectivity plots corresponding to model problems with similar values of λL display similar behaviours. In particular, we see that effectivites of all tested *a posteriori* error estimates are close to 1 whenever λL is large,

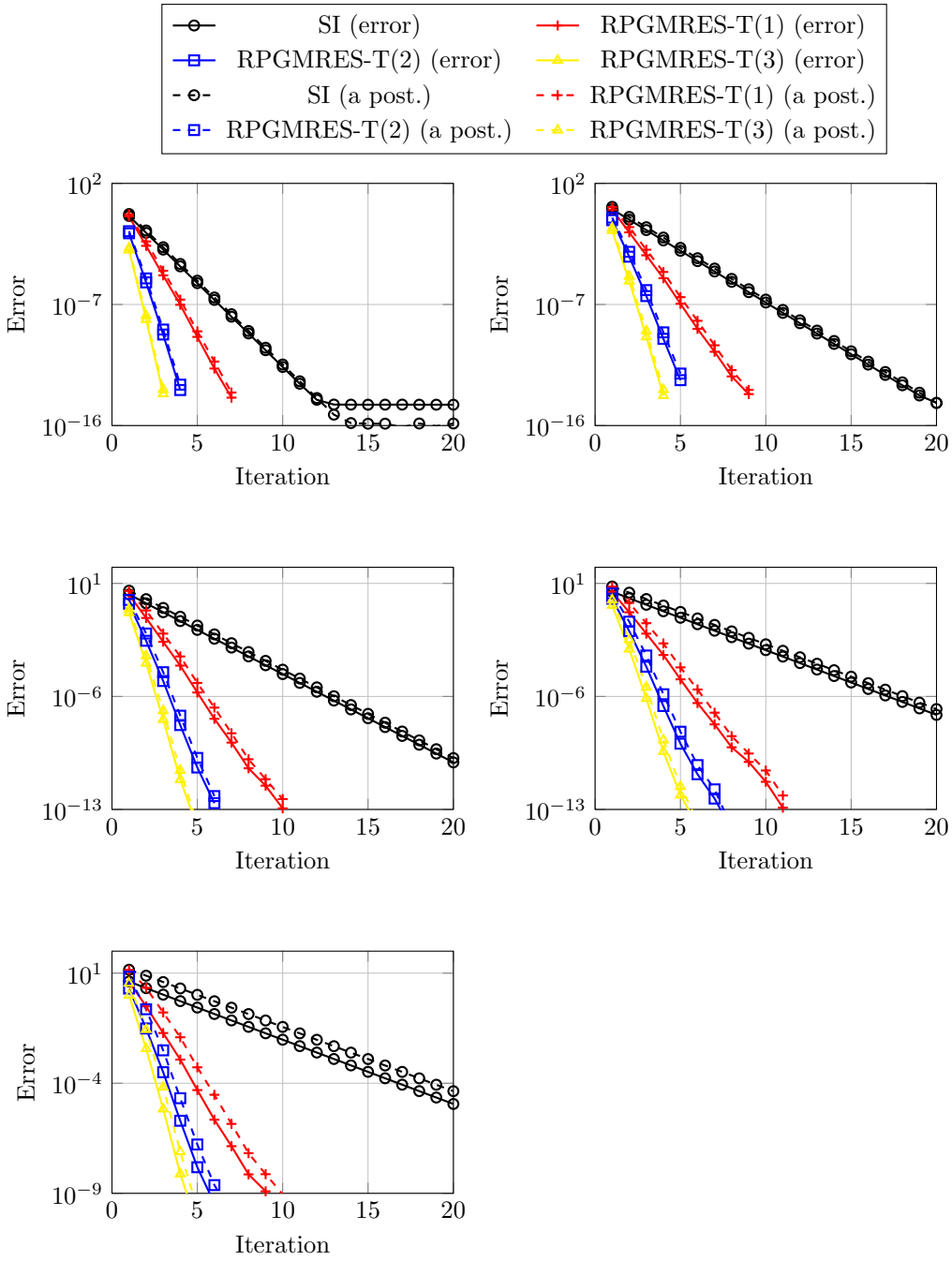

Figure 5.11: Convergence histories of source iteration and right-preconditioned GMRES for a number of choices of transport-based preconditioners applied to the Rayleigh scattering problem for a range of scattering ratios. The macroscopic total cross-section and spatial domain length-scale are chosen to be $\lambda = 1$ and $L = 2$ respectively. Solid line: DG-energy norm error. Dashed line: *a posteriori* solver error estimate. Top-left: $c = \frac{1}{10}$. Top-right: $c = \frac{3}{10}$. Middle-left: $c = \frac{5}{10}$. Middle-right: $c = \frac{7}{10}$. Bottom-left: $c = \frac{9}{10}$.

and that these effectivities deteriorate when $\lambda L \to 0$.

Since the meshes employed in these experiments are essentially scaled versions of each other, the spatial mesh-size parameter h scales with L. Therefore, we can instead

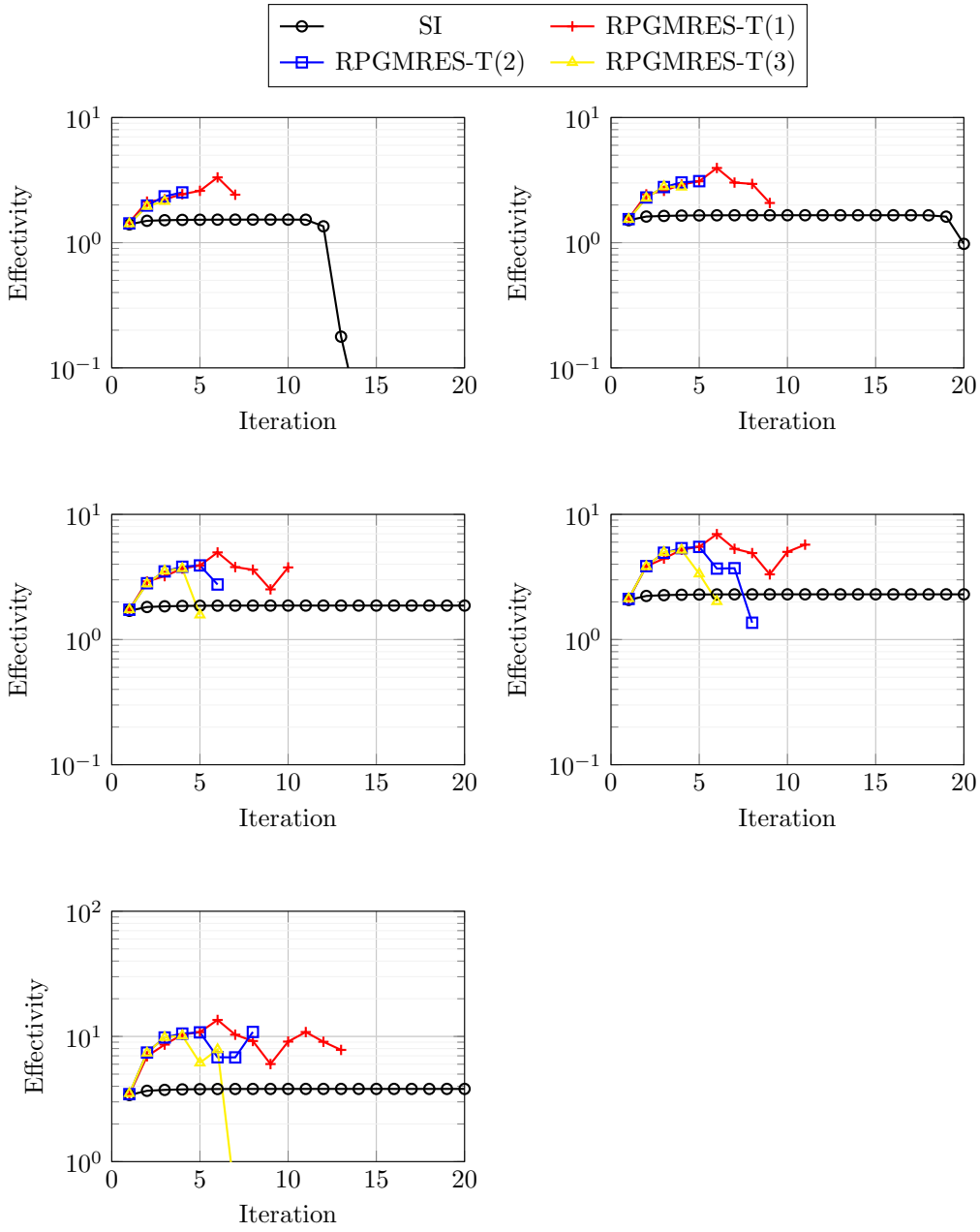

Figure 5.12: Effectivities of the *a posteriori* error estimates for source iteration and right-preconditioned GMRES for a number of choices of transport-based preconditioners applied to the Rayleigh scattering problem for a range of scattering ratios. The macroscopic total cross-section and spatial domain length-scale are chosen to be $\lambda = 1$ and $L = 2$ respectively. Top-left: $c = \frac{1}{10}$. Top-right: $c = \frac{3}{10}$. Middle-left: $c = \frac{5}{10}$. Middle-right: $c = \frac{7}{10}$. Bottom-left: $c = \frac{9}{10}$.

consider the dependence of the qualitative behaviour of the tested iterative solvers on λh rather than λL. By (5.1), we recognise that the behaviours of the tested iterative solvers is dependent on the optical thickness (or cell aspect ratio) of the medium, defined in (5.1).

One may ask whether the rate of convergence of source iteration, modified source

iteration and right-preconditioned GMRES can be predicted using *a priori* knowledge of c, λ and L. A partial answer to this question for both source iteration and modified source iteration was given in the statements of Theorems 5.5.1 and 5.2.6, namely in the contraction factor q. For source iteration, this contraction factor was found to be $q = c$, and for modified source iteration with $\omega = \frac{1}{2}$ was found to be $q = \frac{c}{2-c}$. However, this fails to address the apparent dependence of the contraction factor on λL. A useful heuristic is developed in Chapter 5.3.3, in which the scattering ratio may be multiplied by the leakage ratio c_l appearing in (5.38) to improve the predictions of the convergence rates of these methods.

Test D: Effect of spectral centre approximation for RPGMRES-T(n) Figure 5.15 shows the behaviour of right-preconditioned GMRES using a number of n-sweep preconditioners for the model probem with scattering ratio $c = \frac{7}{10}$ for each of the parameters (λ, L) with $\lambda \in \{\frac{1}{10}, 1, 10\}$ and $L \in \{\frac{1}{5}, 2, 20\}$. As before, the choice of scattering ratio is not relevant for the subsequent study. The n-sweep preconditioners employed are based on (5.37) with the coefficients $\{a_k^{(n)}\}_{k=0}^{n-1}$ chosen as in Theorem 5.3.6 for the following choices of the spectral centre a:

- $a = 0$ - this corresponds to the first two terms of the Neumann expansion of the matrix $(\mathbf{T} - \mathbf{S})^{-1} = \mathbf{T}^{-1}(\mathbf{I} - \mathbf{S}\mathbf{T}^{-1})^{-1}$;
- $a = \frac{c}{2}$ - this corresponds to the preconditioner in Theorem 5.3.6 using the theoretically-predicted optimal selection of a (as in Corollary 5.3.6.1);
- $a = \frac{c_l c}{2}$ - this corresponds to a correction to the theoretically-predicted optimal selection of a that attempts to take into consideration the effect of the dimensionless quantity λL through the leakage ratio (5.38).

Since the choice $n = 1$ essentially yields a preconditioner $\mathbf{P}^{-1}$ that is a constant rescaling of $\mathbf{T}^{-1}$, the choice of the spectral centre approximation a has no tangible effect on the convergence of RPGMRES-T(1); in fact, RPGMRES-T(1) generates the same sequence of approximate solutions (for a given initial guess) regardless of the choice of a. We therefore must take $n \geq 2$ to observe changes in the behaviour of RPGMRES-T(n) for different values of a. In view of minimising the CPU time taken per iteration of RPGMRES-T(n), we select $n = 2$, although similar qualitative behaviour is expected for larger values of n. In practice, one may want to take large values of n if only a small number of Krylov vectors can be stored; these considerations are discussed in Test B.

As was seen earlier, the convergence of all three methods is dependent on the dimensionless quantity λL, with the fastest convergence achieved when λL is large and the slowest convergence achieveds when λL is small. This can be seen in Figure 5.15. Moreover, for large values of λL, the GMRES methods employing two-sweep preconditioners based on the spectral centre estimates $a \in \{\frac{c}{2}, \frac{c_l c}{2}\}$ converge slightly faster than those

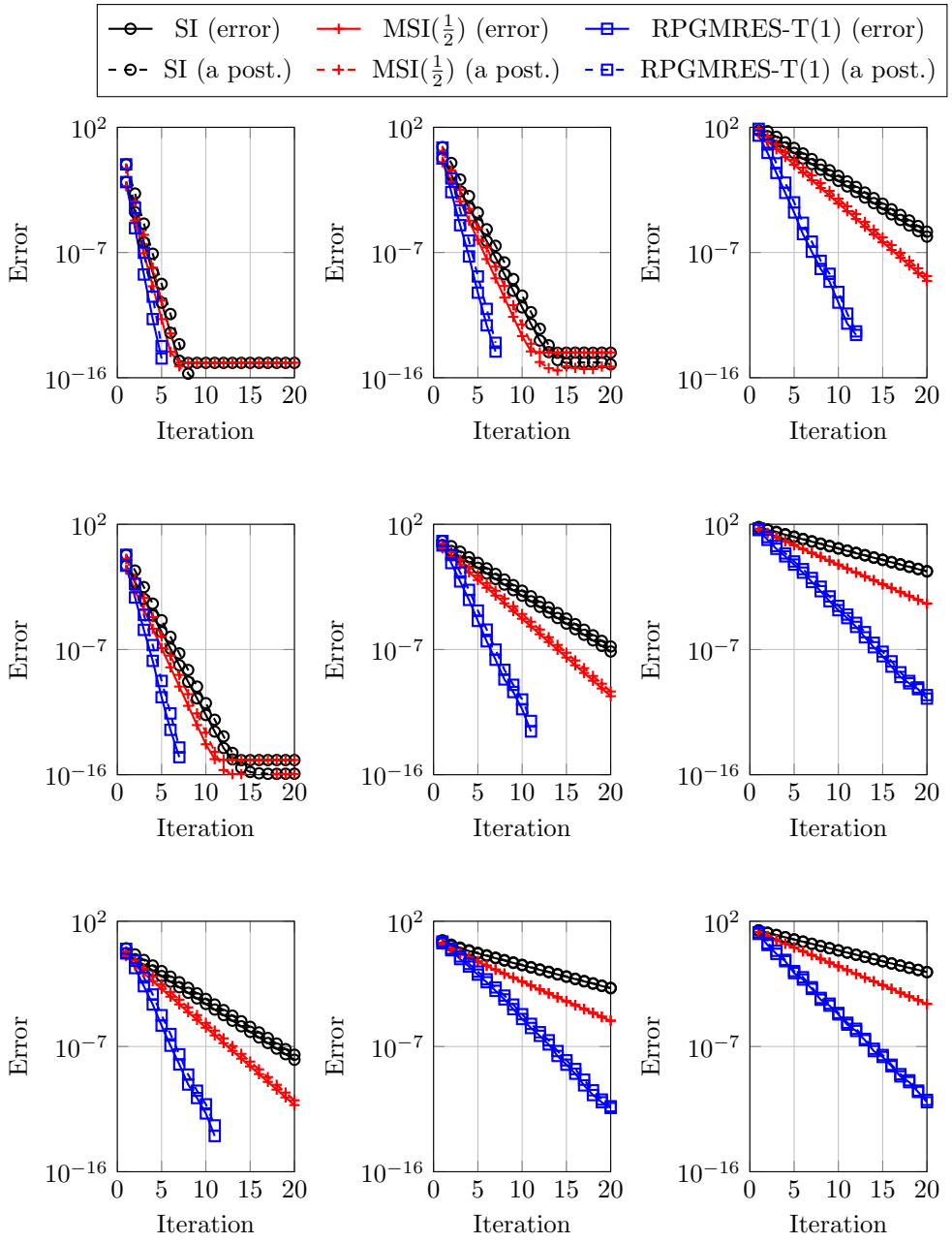

Figure 5.13: Convergence histories of SI, MSI($\frac{1}{2}$) and RPGMRES-T(1) applied to the Rayleigh scattering problem for $c = \frac{7}{10}$, $\lambda \in \{\frac{1}{10}, 1, 10\}$ and $L \in \{\frac{1}{5}, 2, 20\}$. Solid line: DG-energy norm error. Dashed line: *a posteriori* solver error estimate. Top row: $\lambda = \frac{1}{10}$. Middle row: $\lambda = 1$. Bottom row: $\lambda = 10$. Left column: $L = \frac{1}{5}$. Middle column: $L = 2$. Right column: $L = 20$.

employing preconditioners based on the estimate $a = 0$. It is important to remark that the computational cost of RPGMRES-T(2) at each iteration is roughly identical for all values of $a = 0$ under the assumption that transport sweeps are much more expensive than forming linear combinations of small numbers of vectors.

As λL becomes small, the convergence of RPGMRES-T(2) with spectral centre es-

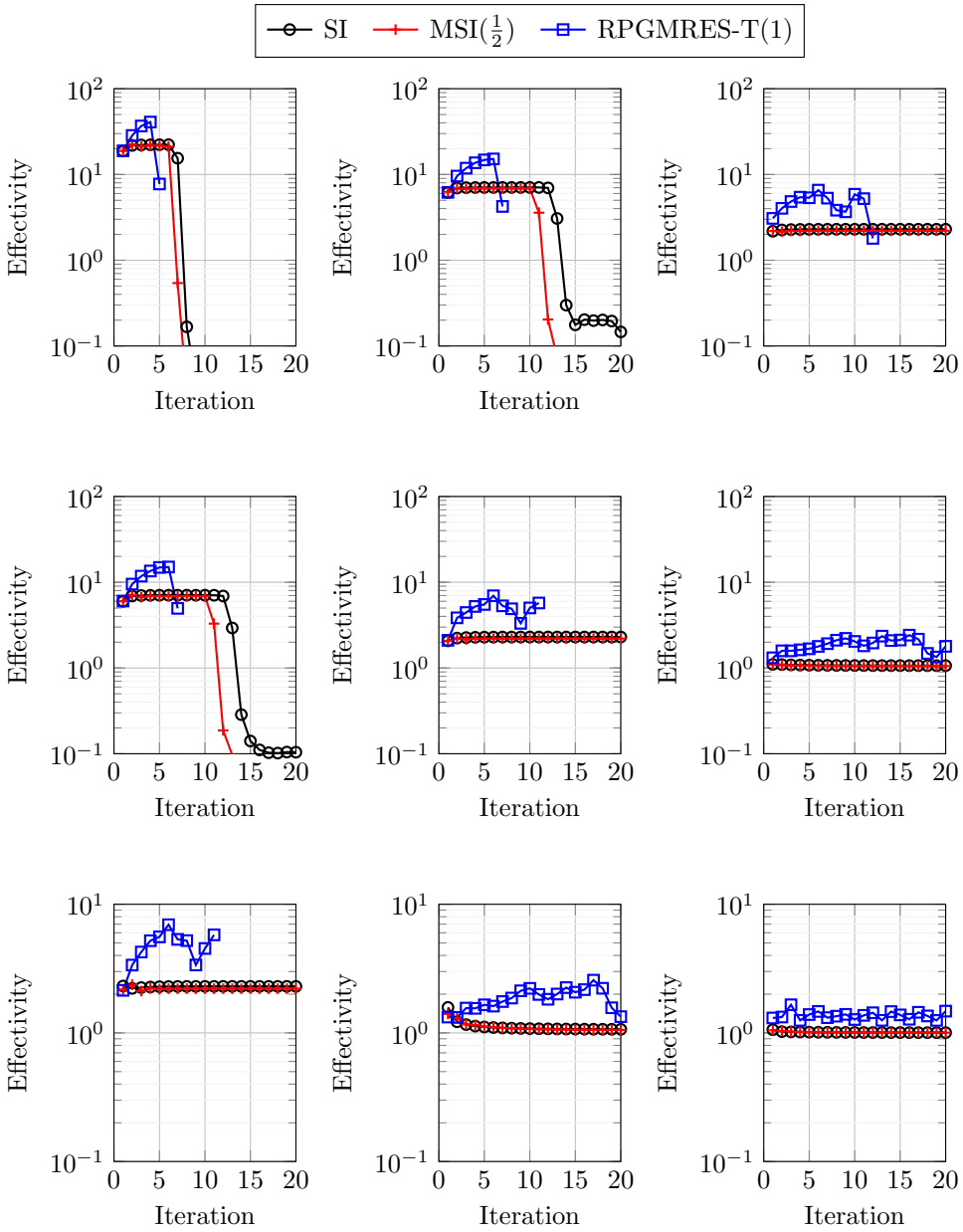

Figure 5.14: Effectivities of the *a posteriori* error estimates for SI, MSI($\frac{1}{2}$) and RPGMRES-T(1) applied to the Rayleigh scattering problem for $c = \frac{7}{10}$, $\lambda \in \{\frac{1}{10}, 1, 10\}$ and $L \in \{\frac{1}{5}, 2, 20\}$. Top row: $\lambda = \frac{1}{10}$. Middle row: $\lambda = 1$. Bottom row: $\lambda = 10$. Left column: $L = \frac{1}{5}$. Middle column: $L = 2$. Right column: $L = 20$.

timate $a = \frac{c}{2}$ becomes slower than the same method with estimate $a = 0$. This is because the eigenvalues of the matrix $\mathbf{S}\mathbf{T}^{-1}$ (which shares the same eigenvalues as $\mathbf{T}^{-1}\mathbf{S}$) are much closer to 0 than to $\frac{c}{2}$. The RPGMRES-T(2) method with spectral estimate $a = \frac{c_I c}{2}$ displays similar rates of convergence as the same method with $a = 0$. It is therefore recommended that the preconditioners based on Theorem 5.3.6 to be used in RPGMRES-T(2) should be based on the spectral centre estimates $a = 0$ or $a = \frac{c_I c}{2}$, with the latter being the optimal choice when λL is large.

Figure 5.16 shows the effectivities of the *a posteriori* solver error estimates employed by RPGMRES-T(2) for each tested value of $a \in \{0, \frac{c}{2}, \frac{c_I c}{2}\}$, $\lambda \in \{\frac{1}{10}, 1, 10\}$ and $L \in \{\frac{1}{5}, 2, 20\}$. As before, the effectivities of the *a posteriori* error estimates are close to 1 for large values of λL and deteriorate as $\lambda L \to 0$.

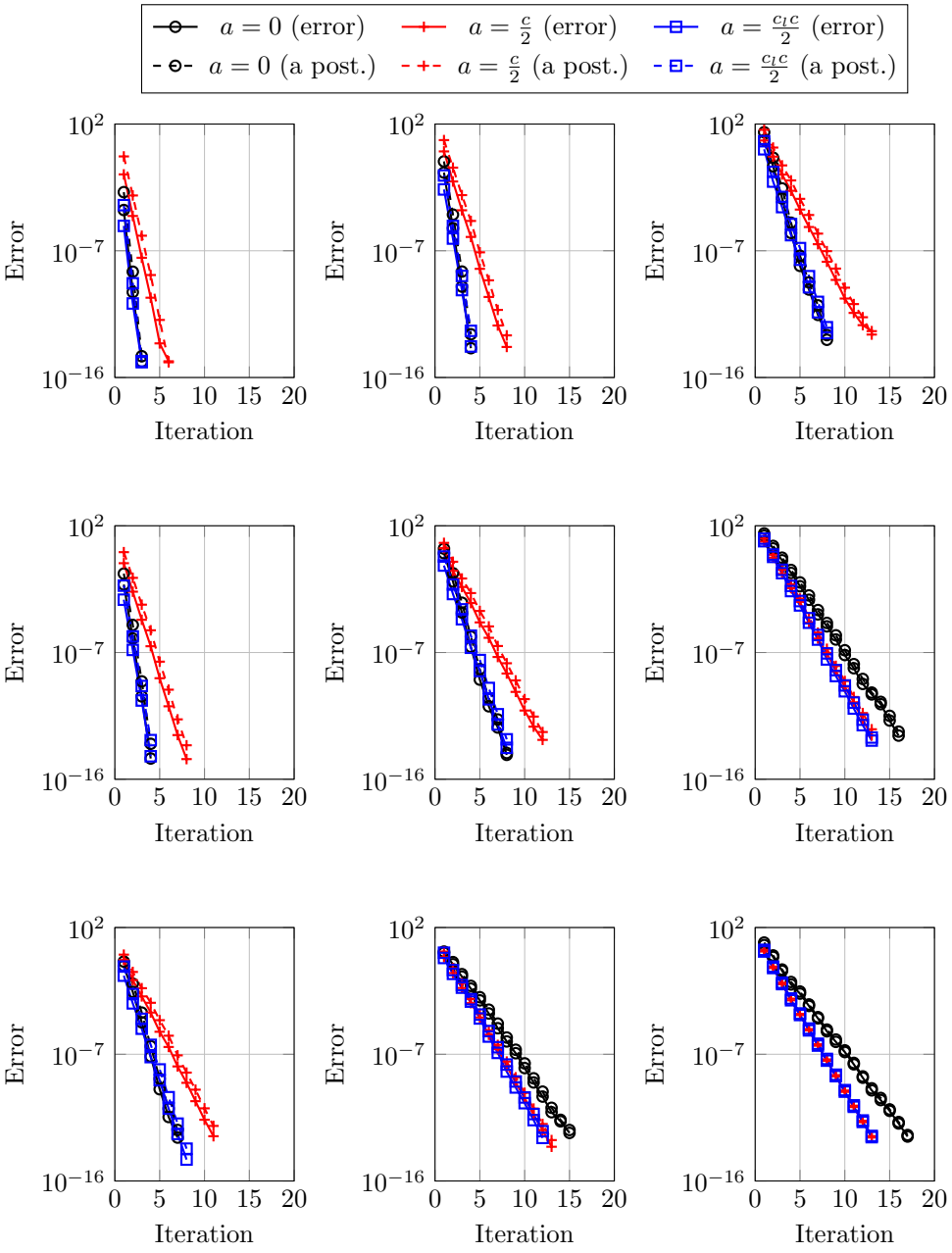

Figure 5.15: Convergence histories of RPGMRES-T(2) applied to the Rayleigh scattering problem for $c = \frac{7}{10}$, $\lambda \in \{\frac{1}{10}, 1, 10\}$ and $L \in \{\frac{1}{5}, 2, 20\}$ using different estimates of the spectral centre a. Solid line: DG-energy norm error. Dashed line: *a posteriori* solver error estimate. Top row: $\lambda = \frac{1}{10}$. Middle row: $\lambda = 1$. Bottom row: $\lambda = 10$. Left column: $L = \frac{1}{5}$. Middle column: $L = 2$. Right column: $L = 20$.

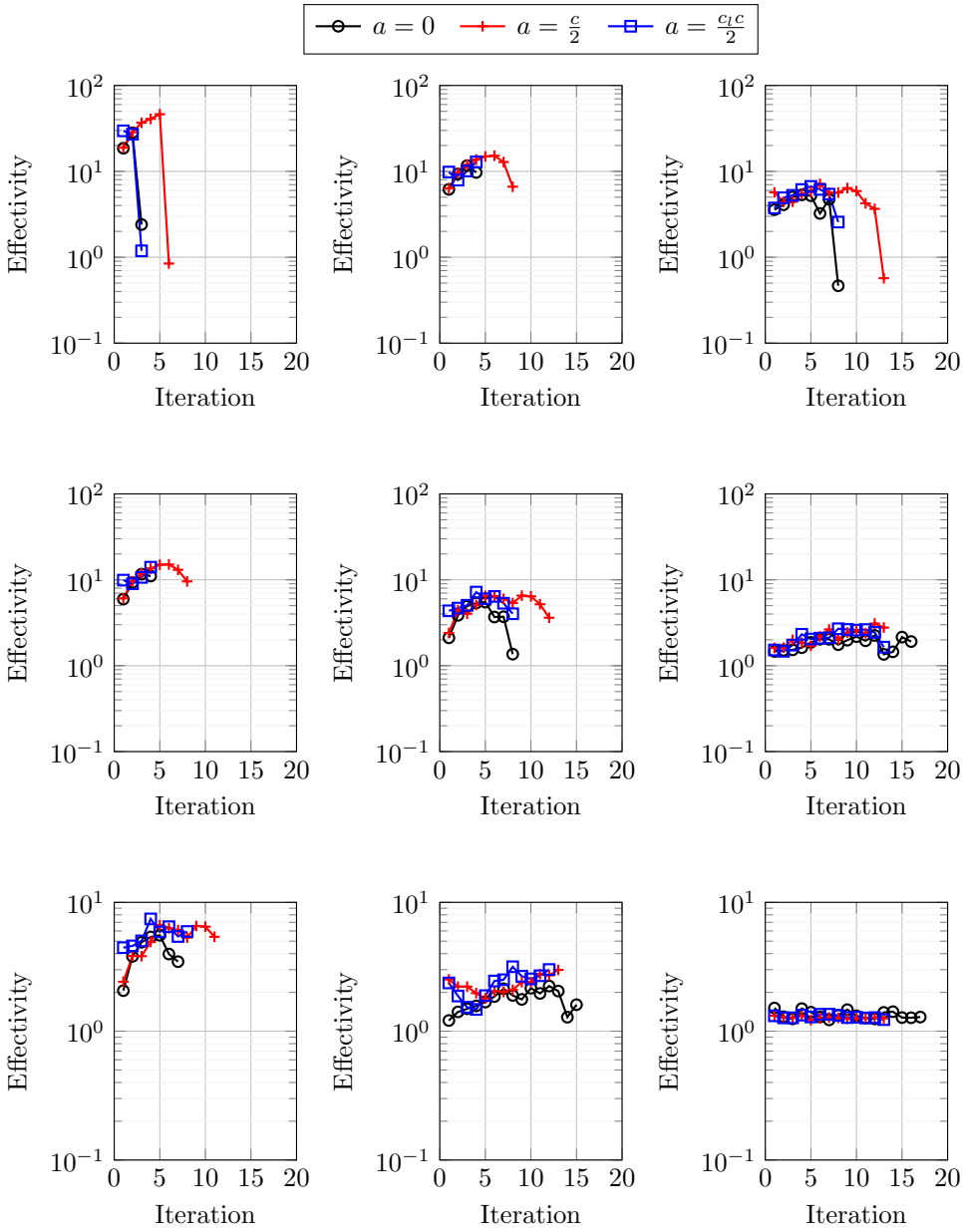

Figure 5.16: Effectivities of the *a posteriori* error estimates for SI, MSI($\frac{1}{2}$) and RPGMRES-T(1) applied to the Rayleigh scattering problem for $c = \frac{7}{10}$, $\lambda \in \{\frac{1}{10}, 1, 10\}$ and $L \in \{\frac{1}{5}, 2, 20\}$. Top row: $\lambda = \frac{1}{10}$. Middle row: $\lambda = 1$. Bottom row: $\lambda = 10$. Left column: $L = \frac{1}{5}$. Middle column: $L = 2$. Right column: $L = 20$.

5.5 Further Extensions

5.5.1 Error analysis for poly-energetic source iteration

While this chapter has largely focussed on the convergence of iterative methods applied to the mono-energetic LBTE, some of our prevoius results can be extended to the poly-energetic setting. Since the scattering bilinear form $S(\cdot, \cdot)$ is no longer symmetric, we do not consider prescribing a poly-energetic analogue of the modified source iteration

method of Chapter 5.2. However, we shall prove the convergence of the classical source iteration method applied to the discrete poly-energetic problem; this will be achieved through the introduction of a poly-energetic notion of the *scattering ratio* which characterises the rate of convergence of source iteration [1].

For simplicity of presentation, we shall restate the poly-energetic DGFEM problem: find $u_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ such that

$$T(u_h, v_h) = S(u_h, v_h) + \ell(v_h) \tag{5.43}$$

Rather than proving convergence in the DG-energy norm (3.38) defined in Chapter 3.3, we will instead prove convergence in a norm $|||\cdot|||_T : \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \times \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \to \mathbb{R}$ defined by

$$\begin{aligned}|||v|||_T^2 = &||\sqrt{\alpha+\beta}v||_{L^2(\mathcal{D})}^2 \\ &+ \frac{1}{2}\int_{\mathbb{Y}}\int_{\mathbb{S}}\sum_{\kappa_\Omega\in\mathcal{T}_\Omega}\left(||v^+ - v^-||^2_{\partial_-\kappa_\Omega(\boldsymbol{\mu})\setminus\partial\Omega} + ||v^+||^2_{\partial\kappa_\Omega(\boldsymbol{\mu})\cap\partial\Omega}\right)\,\mathrm{d}\boldsymbol{\mu}\,\mathrm{d}E.\end{aligned}$$

This is the natural norm in which to measure coercivity of the bilinear form $T : \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \times \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \to \mathbb{R}$; indeed, for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$, we have

$$|||v_h|||_T^2 = T(v_h, v_h).$$

In fact, the definition of $|||\cdot|||_T$ can be extended to include the broken space $\mathcal{G}(\mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}})$ (defined in (3.16)) in its domain, though we shall not need this fact for the forthcoming analysis.

For the proof of the following theorem, we shall restate (3.7) from Chapter 3.1, a coefficient derived from the differential scattering cross-section defined by

$$\gamma(\mathbf{x}, \boldsymbol{\mu}, E) = \int_{\mathbb{Y}}\int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu}' \to \boldsymbol{\mu}, E' \to E)\,\,\mathrm{d}\boldsymbol{\mu}'\,\mathrm{d}E$$

and assert that, under the assumption that the medium is angularly isotropic, we have $\gamma(\mathbf{x}, \boldsymbol{\mu}, E) = \gamma(\mathbf{x}, E)$.

Theorem 5.5.1 (Poly-energetic source iteration)**.** *The map $F : \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \to \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ defined for any $w_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ as the solution to the variational problem*

$$T(F(w_h), v_h) = S(w_h, v_h) + \ell(v_h)$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ admits a unique fixed point $u_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ provided that $q = \sqrt{q_\beta q_\gamma} < 1$, where

$$\begin{aligned}q_\beta &= \operatorname*{ess\,sup}_{\mathbf{x}\in\Omega, E\in\mathbb{Y}}\left(\frac{\beta(\mathbf{x}, E)}{\alpha(\mathbf{x}, E) + \beta(\mathbf{x}, E)}\right), \\ q_\gamma &= \operatorname*{ess\,sup}_{\mathbf{x}\in\Omega, E\in\mathbb{Y}}\left(\frac{\gamma(\mathbf{x}, E)}{\alpha(\mathbf{x}, E) + \beta(\mathbf{x}, E)}\right).\end{aligned}$$

Moreover, the sequence $\{u_h^{(n)}\}_{n\geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ defined by $u_h^{(n+1)} = F(u_h^{(n)})$ for $n \geq 0$ converges to u_h for any choice of $u_h^{(0)}$. We also have the following error reduction

estimate, a priori *error estimate and* a posteriori *error estimate for the* $|||\cdot|||_T$*-norm solver error:*

$$
\begin{aligned}
|||u_h^{(n+1)} - u_h|||_T &\leq q\ |||u_h^{(n)} - u_h|||_T, \\
|||u_h^{(n+1)} - u_h|||_T &\leq \frac{q^n}{1-q}\ |||u_h^{(1)} - u_h^{(0)}|||_T, \\
|||u_h^{(n+1)} - u_h|||_T &\leq \frac{q}{1-q} |||u_h^{(n+1)} - u_h^{(n)}|||_T.
\end{aligned}
$$

Proof. We first remark that the mapping F is well-posed. Let $w_1, w_2 \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$. We have

$$
\begin{aligned}
|||F(w_1) - F(w_2)|||_T^2 &= T(F(w_1) - F(w_2), F(w_1) - F(w_2)) \\
&= S(w_1 - w_2, F(w_1) - F(w_2)) \\
&\leq ||\beta^{\frac{1}{2}}(w_1 - w_2)||_{L^2(\mathcal{D})} ||\gamma^{\frac{1}{2}}(F(w_1) - F(w_2))||_{L^2(\mathcal{D})},
\end{aligned}
$$

where we have used Lemma 3.3.1 from Chapter 3.3 to bound the bilinear form $S(\cdot,\cdot)$ from above. Noting that

$$
\begin{aligned}
\beta(\mathbf{x}, E) &\leq \underbrace{\operatorname*{ess\,sup}_{\mathbf{z}\in\Omega, E'\in\mathbb{Y}} \left(\frac{\beta(\mathbf{z}, E')}{\alpha(\mathbf{z}, E') + \beta(\mathbf{z}, E')} \right)}_{=:q_\beta} (\alpha(\mathbf{x}, E) + \beta(\mathbf{x}, E)), \\
\gamma(\mathbf{x}, E) &\leq \underbrace{\operatorname*{ess\,sup}_{\mathbf{z}\in\Omega, E'\in\mathbb{Y}} \left(\frac{\gamma(\mathbf{z}, E')}{\alpha(\mathbf{z}, E') + \beta(\mathbf{z}, E')} \right)}_{=:q_\gamma} (\alpha(\mathbf{x}, E) + \beta(\mathbf{x}, E)),
\end{aligned}
$$

we have

$$
\begin{aligned}
|||F(w_1) - F(w_2)|||_T^2 &\leq ||\beta^{\frac{1}{2}}(w_1 - w_2)||_{L^2(\mathcal{D})} ||\gamma^{\frac{1}{2}}(F(w_1) - F(w_2))||_{L^2(\mathcal{D})} \\
&\leq \sqrt{q_\beta q_\gamma}\ ||(\alpha+\beta)^{\frac{1}{2}}(w_1 - w_2)||_{L^2(\mathcal{D})} \cdot \\
&\qquad ||(\alpha+\beta)^{\frac{1}{2}}(F(w_1) - F(w_2))||_{L^2(\mathcal{D})} \\
&\leq \sqrt{q_\beta q_\gamma}\ |||w_1 - w_2|||_T |||F(w_1) - F(w_2))|||_T.
\end{aligned}
$$

Dividing both sides by $|||F(w_1) - F(w_2))|||_T$ and defining $q = \sqrt{q_\beta q_\gamma}$, we get

$$
|||F(w_1) - F(w_2)|||_T \leq q\ |||w_1 - w_2|||_T.
$$

Therefore, we have a contraction mapping on $\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ provided that $q < 1$. Since $(\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}, |||\cdot|||_T)$ is a non-empty and complete metric space, Banach's fixed point theorem implies that F admits a unique fixed point $u_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$, and that the sequence $\{u_h^{(n)}\}_{n\geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ defined by $u_h^{(n+1)} = F(u_h^{(n)})$ for $n \geq 0$ converges to u_h for any choice of $u_h^{(0)} \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$.

The proofs of the three error bounds are straightforward. The error reduction inequality is proven by the definition of the fixed point u_h and the relationship between consecutive terms in the sequence $\{u_h^{(n)}\}_{n\geq 0}$:

$$
|||u_h^{(n+1)} - u_h|||_T = |||F(u_h^{(n)}) - F(u_h)|||_T \leq q\ |||u_h^{(n)} - u_h|||_T.
$$

Applying the triangle inequality after one application of the error reduction inequality yields

$$\begin{aligned}|||u_h^{(n+1)} - u_h|||_T &\leq q\ |||u_h^{(n)} - u_h^{(n+1)} + u_h^{(n+1)} - u_h|||_T \\ &\leq q\left(|||u_h^{(n)} - u_h^{(n+1)}|||_T + |||u_h^{(n+1)} - u_h|||_T\right).\end{aligned}$$

The *a posteriori* estimate follows on rearrangment. The *a priori* estimate follows from applying the error reduction estimate n times, followed by one application of the *a posteriori* error estimate:

$$\begin{aligned}|||u_h^{(n+1)} - u_h|||_T &\leq q^n\ |||u_h^{(1)} - u_h|||_T \\ &\leq \frac{q^n}{1-q}\ |||u_h^{(1)} - u_h^{(0)}|||_T.\end{aligned}$$

□

Theorem 5.5.1 states that the poly-energetic LBTE method discretised using discontinuous Galerkin finite element methods in the space-angle-energy setting is convergent provided $q = \sqrt{q_\beta q_\gamma} < 1$, where q_β and q_γ are constants that depend only on the total absorption cross-section $\alpha(\mathbf{x}, E)$ and the differential scattering cross-section $\theta(\mathbf{x}, \boldsymbol{\mu}' \cdot \boldsymbol{\mu}, E' \rightarrow E)$. Therefore, q plays an analogous role to the so-called *(global) scattering ratio* c found in the analysis of the (non-discretised and infinite-medium) mono-energetic LBTE [1], and can be thought of as an extension of the scattering ratio to the poly-energetic setting. By considering the DGFEM-discretised mono-energetic LBTE as a DGFEM-discretised poly-energetic problem with the energetic dependence of u, α and θ dropped, one can show that the contraction factor q in the analysis above simplifies to

$$q = \operatorname*{ess\,sup}_{\mathbf{x}\in\Omega}\left(\frac{\beta(\mathbf{x})}{\alpha(\mathbf{x}) + \beta(\mathbf{x})}\right),$$

which agrees with the classical result in the infinite-medium setting [1].

We shall now turn our attention to the derivation of *a posteriori* error estimates for DG-energy norm solver errors. Recall that the definition of $|||v_h|||_{DG}^2$ in (3.38) includes a weighted norm of the form $||\bar{\alpha}^{\frac{1}{2}} v_h||_{L^2(\mathcal{D})}^2$, where

$$\bar{\alpha}(\mathbf{x}, E) = \alpha(\mathbf{x}, E) + \frac{1}{2}(\beta(\mathbf{x}, E) - \gamma(\mathbf{x}, E)). \tag{5.44}$$

As such, we have $||\bar{\alpha}^{\frac{1}{2}} v_h||_{L^2(\mathcal{D})}^2 \leq |||v_h|||_{DG}^2$ for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$. The following theorem provides an *a posteriori* error estimate for the DG-energy norm error rather than the $|||\cdot|||_T$-norm error.

Theorem 5.5.2. *Let $\{u_h^{(n)}\}_{n\geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ be constructed as in Theorem 5.5.1. At the n^{th} source iteration, the DG-energy norm of the solver error $u_h^{(n)} - u_h$ satisfies*

$$|||u_h^{(n)} - u_h|||_{DG} \leq r_\gamma^{\frac{1}{2}}\ ||\beta^{\frac{1}{2}}(u_h^{(n)} - u_h^{(n-1)})||_{L^2(\mathcal{D})},$$

where

$$r_\gamma = \operatorname*{ess\,sup}_{\mathbf{z}\in\Omega, E'\in\mathbb{Y}}\left(\frac{\gamma(\mathbf{z}, E')}{\alpha(\mathbf{z}, E') + \frac{1}{2}(\beta(\mathbf{z}, E') - \gamma(\mathbf{z}, E'))}\right).$$

Proof. Letting $e_h^{(n)} = u_h^{(n)} - u_h$, we have

$$\begin{aligned}
|||e_h^{(n)}|||_{DG}^2 &\leq T(e_h^{(n)}, e_h^{(n)}) - S(e_h^{(n)}, e_h^{(n)}) \\
&= S(e_h^{(n-1)}, e_h^{(n)}) - S(e_h^{(n)}, e_h^{(n)}) \\
&= S(u_h^{(n-1)} - u_h^{(n)}, e_h^{(n)}) \\
&\leq ||\beta^{\frac{1}{2}}(u_h^{(n-1)} - u_h^{(n)})||_{L^2(\mathcal{D})} ||\gamma^{\frac{1}{2}} e_h^{(n)}||_{L^2(\mathcal{D})}.
\end{aligned}$$

Writing $\bar{\alpha}$ as in (5.44) and noting that

$$\gamma(\mathbf{x}, E) \leq \underbrace{\operatorname*{ess\,sup}_{\mathbf{z}\in\Omega, E'\in\mathbb{Y}} \left(\frac{\gamma(\mathbf{z}, E')}{\bar{\alpha}(\mathbf{z}, E')} \right)}_{=:r_\gamma} \bar{\alpha}(\mathbf{x}, E),$$

we have

$$\begin{aligned}
|||e_h^{(n)}|||_{DG}^2 &\leq r_\gamma^{\frac{1}{2}} ||\beta^{\frac{1}{2}}(u_h^{(n-1)} - u_h^{(n)})||_{L^2(\mathcal{D})} ||\bar{\alpha}^{\frac{1}{2}} e_h^{(n)}||_{L^2(\mathcal{D})} \\
&\leq r_\gamma^{\frac{1}{2}} ||\beta^{\frac{1}{2}}(u_h^{(n-1)} - u_h^{(n)})||_{L^2(\mathcal{D})} |||e_h^{(n)}|||_{DG}.
\end{aligned}$$

The *a posteriori* error bound is proved on rearrangement. □

From the perspective of designing linear solvers for the discretised poly-energetic LBTE, it is useful to have a computable *a posteriori* error estimator to bound the error $u_h - \hat{u}_h$ between the exact solution u_h of the discrete equations and a computed approximation $\hat{u}_h$ of u_h. However, the *a posteriori* error estimate presented in Theorem 5.5.2 is insufficient in two ways:

- The evaluation of the error estimate requires knowledge of q which may be expensive to compute, since it requires the solution of two maximisation problems over the space-energy domain;
- The error estimate in the theorem above is only valid for sequences of approximate solutions generated by source iteration.

To this end, we shall present a computable DG-energy norm *a posteriori* error bound, based on the residual of the linear system of equations, that is valid for any approximation of the solution of the discrete problem.

Theorem 5.5.3 (DG-energy norm *a posteriori* error bound, poly-energetic version)**.** *Let $\bar{\alpha}$ be as in (5.44) and define an inner product $(\cdot,\cdot)_{L_w^2(\mathcal{D})} : \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \times \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \to \mathbb{R}$ and associated norm $||\cdot||_{L_w^2(\mathcal{D})} : \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}} \to \mathbb{R}$ for all $v_h, w_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ by*

$$\begin{aligned}
(w_h, v_h)_{L_w^2(\mathcal{D})} &= \int_{\mathbb{Y}} \int_{\mathbb{S}} \int_{\Omega} \bar{\alpha}(\mathbf{x}, E) w_h(\mathbf{x}, \boldsymbol{\mu}, E) v_h(\mathbf{x}, \boldsymbol{\mu}, E) \, d\mathbf{x} d\boldsymbol{\mu} dE, \\
||v_h||_{L_w^2(\mathcal{D})} &= \sqrt{(v_h, v_h)_{L_w^2(\mathcal{D})}}.
\end{aligned}$$

Let $u_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ be the exact solution to the variational problem

$$T(u_h, v_h) = S(u_h, v_h) + \ell(v_h)$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$*, and* $\hat{u}_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ *denote an approximation of* u_h*. Then we have*

$$|||u_h - \hat{u}_h|||_{DG} \leq ||r_h||_{L^2_w(\mathcal{D})},$$

where $r_h = r_h(\hat{u}_h) \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ *denotes the unique solution to the following variational problem for all* $v_h \in \mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$*:*

$$(r_h(\hat{u}_h), v_h)_{L^2_w(\mathcal{D})} = \ell(v_h) - (T(\hat{u}_h, v_h) - S(\hat{u}_h, v_h)).$$

The proof of Theorem 5.5.3 follows identical steps as in the proof of Theorem 5.2.8 with the exception of the slightly different definition of the inner product $(\cdot,\cdot)_{L^2_w(\mathcal{D})}$. This inner product differs from the one employed in the mono-energetic version in two ways:

- the inner product accepts arguments from $\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ rather than $\mathcal{V}_{\Omega,\mathbb{S}}$;
- the inner product is defined using the weight function $\bar{\alpha} = \alpha + \frac{1}{2}(\beta - \gamma)$ rather than α.

The *a posteriori* error estimate in Theorem 5.5.3 may also be implemented in a similar manner as the estimate in Theorem 5.2.8 by using a linear algebra representation of the residual vector. The result is similar to (5.26):

$$|||u_h - \hat{u}_h|||_{DG} \leq \sqrt{\hat{\mathbf{r}}^\top \mathbf{M}^{-1} \hat{\mathbf{r}}},$$

where $\hat{\mathbf{r}} = \mathbf{f} - (\mathbf{T} - \mathbf{S})\hat{\mathbf{u}}$ denotes the residual vector induced by the approximate solution $\hat{u}$ expanded in a basis of $\mathcal{V}_{\Omega,\mathbb{S},\mathbb{Y}}$ and $\mathbf{M}$ denotes a weighted mass matrix associated with the inner product $(\cdot,\cdot)_{L^2_w(\mathcal{D})}$.

Remark. *As was seen in the mono-energetic setting, one may compute the* a posteriori *error estimate above as*

$$|||u_h - \hat{u}_h|||_{DG} \leq \sqrt{\sum_{\kappa \in \mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}} \hat{\mathbf{r}}_\kappa^\top \mathbf{M}_\kappa \hat{\mathbf{r}}_\kappa},$$

where $\hat{\mathbf{r}}_\kappa$ *and* $\mathbf{M}_\kappa$ *denote, respectively, the local residual vector and local space-angle-energy mass matrix on each space-angle-energy element* $\kappa \in \mathcal{T}_{\Omega,\mathbb{S},\mathbb{Y}}$*.*

5.5.2 Iterative methods for DOG implementations

In this chapter, we have primarily discussed iterative solvers for systems of equations arising from DGFEM discretisations of the mono-energetic linear Boltzmann transport equations. In Chapter 5.1, we indicated a preference to derive linear solvers for the original discrete problems of Chapter 3.2 rather than the discrete ordinates Galerkin (DOG) implementations of Chapter 3.4. It was remarked in this chapter that the DOG implementations generally resulted in a sparser matrix representation of the discretised

transport operator as a result of a judicious choice of angular basis functions. Specifically, we decomposed the solution $u_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ and test function $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ as

$$u_h(\mathbf{x},\boldsymbol{\mu}) = \sum_{\kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}} \sum_{i=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}} u^i_{\kappa_{\mathbb{S}}}(\mathbf{x}) \varphi^i_{\kappa_{\mathbb{S}}}(\boldsymbol{\mu}), \tag{5.45}$$

$$v_h(\mathbf{x},\boldsymbol{\mu}) = \sum_{\kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}} \sum_{i=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}} v^i_{\kappa_{\mathbb{S}}}(\mathbf{x}) \varphi^i_{\kappa_{\mathbb{S}}}(\boldsymbol{\mu}), \tag{5.46}$$

where each $\varphi^i_{\kappa_{\mathbb{S}}} \in \mathcal{V}_{\mathbb{S}}$ for $\kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}$ and $1 \leq i \leq (q_{\kappa_{\mathbb{S}}}+1)^{d-1}$ are the angular basis functions outlined in Chapter 3.4.2. Henceforth, we shall always associate any space-angle finite element function $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ with the set of functions $\{v^i_{\kappa_{\mathbb{S}}} : \kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}, 1 \leq i \leq (q_{\kappa_{\mathbb{S}}}+1)^{d-1}\}$ in (5.46). We shall highlight some minor modifications to the analysis of linear solvers that suggest that the convergence results presented in Chapter 5.2 can be extended to mono-energetic DOG schemes. For simplicity of presentation, we shall only study the development of source iteration methods for mono-energetic DOG schemes.

Discretisation

In Chapter 3.4.2, it was remarked that the DOG implementation for mono-energetic problems resulted in a linear system with a block structure; we shall recast this system of equations in a variational setting here. Recall that the DOG scheme for mono-energetic problems introduces quadrature schemes $\{(\boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}}, \omega^i_{\kappa_{\mathbb{S}}})\}_{i=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}}$ for each angular element $\kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}$. These quadrature schemes are used to introduce a basis $\{\varphi^i_{\kappa_{\mathbb{S}}} : \kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}, 1 \leq i \leq (q_{\kappa_{\mathbb{S}}}+1)^{d-1}\}$ of the angular finite element space $\mathcal{V}_{\mathbb{S}}$, as well as the family of bilinear forms $\hat{T}^i_{\kappa_{\mathbb{S}}}, \hat{S}^{j,i}_{\kappa'_{\mathbb{S}},\kappa_{\mathbb{S}}} : \mathcal{V}_\Omega \times \mathcal{V}_\Omega \to \mathbb{R}$ and linear functionals $\hat{\ell}^i_{\kappa_{\mathbb{S}}} : \mathcal{V}_\Omega \to \mathbb{R}$ defined for all $w_h, v_h \in \mathcal{V}_\Omega$ by

$$\begin{aligned}
\tilde{T}^i_{\kappa_{\mathbb{S}}}(w_h, v_h) = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Bigg(& \int_{\kappa_\Omega} \left(-w_h \boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}} \cdot \nabla_{\mathbf{x}} v_h + (\alpha(\mathbf{x}, \boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}}) + \beta(\mathbf{x}, \boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}})) w_h v_h\right) \, \mathrm{d}\mathbf{x} \\
& + \int_{\partial_+ \kappa_\Omega(\boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}})} |\boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}} \cdot \mathbf{n}_{\kappa_\Omega}| w_h^+ v_h^+ \, \mathrm{d}s \\
& - \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}}) \setminus \partial\Omega} |\boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}} \cdot \mathbf{n}_{\kappa_\Omega}| w_h^- v_h^+ \, \mathrm{d}s \Bigg), \\
\tilde{S}^{j,i}_{\kappa'_{\mathbb{S}},\kappa_{\mathbb{S}}}(w_h, v_h) = & \int_\Omega \beta^{j,i}_{\kappa'_{\mathbb{S}},\kappa_{\mathbb{S}}}(\mathbf{x}) w_h(\mathbf{x}) v_h(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \\
\tilde{\ell}^i_{\kappa_{\mathbb{S}}}(v_h) = \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \Bigg(& \int_{\kappa_\Omega} f(\mathbf{x}, \boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}}) v_h \, \mathrm{d}\mathbf{x} \\
& + \int_{\partial_- \kappa_\Omega(\boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}}) \cap \partial\Omega} |\boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}} \cdot \mathbf{n}_{\kappa_\Omega}| g(\mathbf{x}, \boldsymbol{\mu}^i_{\kappa_{\mathbb{S}}}) v_h^+ \, \mathrm{d}s \Bigg),
\end{aligned}$$

where $\beta^{j,i}_{\kappa'_{\mathbb{S}},\kappa_{\mathbb{S}}}(\mathbf{x})$ is defined for all $\kappa_{\mathbb{S}}, \kappa'_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}$, $1 \leq i \leq (q_{\kappa_{\mathbb{S}}}+1)^{d-1}$ and $1 \leq j \leq (q_{\kappa'_{\mathbb{S}}}+1)^{d-1}$ by

$$\beta^{j,i}_{\kappa'_{\mathbb{S}},\kappa_{\mathbb{S}}}(\mathbf{x}) = \int_{\mathbb{S}} \int_{\mathbb{S}} \theta(\mathbf{x}, \boldsymbol{\mu} \cdot \boldsymbol{\mu}') \varphi^i_{\kappa_{\mathbb{S}}}(\boldsymbol{\mu}) \varphi^j_{\kappa'_{\mathbb{S}}}(\boldsymbol{\mu}') \, \mathrm{d}\boldsymbol{\mu}' \, \mathrm{d}\boldsymbol{\mu}.$$

For any $w_h, v_h \in \mathcal{V}_\Omega$, we recognise that $\tilde{T}^i_{\kappa_{\mathbb{S}}}(w_h, v_h)$ and $\tilde{\ell}^i_{\kappa_{\mathbb{S}}}(v_h)$ denote approximations of $T(w_h \varphi^i_{\kappa_{\mathbb{S}}}, v_h \varphi^i_{\kappa_{\mathbb{S}}})$ and $\ell(v_h \varphi^i_{\kappa_{\mathbb{S}}})$ (with $T(\cdot,\cdot)$ and $\ell(\cdot)$ defined in (5.6) and (5.8)

respectively) in which the angular integrals are replaced with the previously-defined quadrature scheme. On the other hand, we have that $\tilde{S}^{j,i}_{\kappa'_{\mathbb{S}},\kappa_{\mathbb{S}}}(w_h, v_h) = S(w_h\varphi^j_{\kappa'_{\mathbb{S}}}, v_h\varphi^i_{\kappa_{\mathbb{S}}})$ for all $w_h, v_h \in \mathcal{V}_\Omega$.

The mono-energetic DOG implementation reads as follows: for each $\kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}$ and $i \leq i \leq (q_{\kappa_{\mathbb{S}}}+1)^{d-1}$, find $u^i_{\kappa_{\mathbb{S}}} \in \mathcal{V}_\Omega$ such that

$$\omega^i_{\kappa_{\mathbb{S}}}\tilde{T}^i_{\kappa_{\mathbb{S}}}(u^i_{\kappa_{\mathbb{S}}}, v^i_{\kappa_{\mathbb{S}}}) = \sum_{\kappa'_{\mathbb{S}}\in\mathcal{T}_{\mathbb{S}}} \sum_{j=1}^{(q_{\kappa'_{\mathbb{S}}}+1)^{d-1}} \tilde{S}^{j,i}_{\kappa'_{\mathbb{S}},\kappa_{\mathbb{S}}}(u^j_{\kappa'_{\mathbb{S}}}, v^i_{\kappa_{\mathbb{S}}}) + \omega^i_{\kappa_{\mathbb{S}}}\tilde{\ell}^i_{\kappa_{\mathbb{S}}}(v^i_{\kappa_{\mathbb{S}}}) \tag{5.47}$$

for all $v^i_{\kappa_{\mathbb{S}}} \in \mathcal{V}_\Omega$. We note that the matrix form outlined in Chapter 3.4.2 is recovered upon selection of an appropriate basis of $\mathcal{V}_\Omega$.

The mono-energetic DOG scheme can be more compactly written in the following manner. Associating each $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ with the set $\{v^i_{\kappa_{\mathbb{S}}} : \kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}, 1 \leq i \leq (q_{\kappa_{\mathbb{S}}}+1)^{d-1}\} \subset \mathcal{V}_\Omega$ and introducing the following bilinear forms $\tilde{T}, \tilde{S} : \mathcal{V}_{\Omega,\mathbb{S}} \times \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ and linear functional $\tilde{\ell} : \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$:

$$\begin{aligned}
\tilde{T}(w_h, v_h) &= \sum_{\kappa_{\mathbb{S}}\in\mathcal{T}_{\mathbb{S}}} \sum_{i=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}} \omega^i_{\kappa_{\mathbb{S}}}\tilde{T}^i_{\kappa_{\mathbb{S}}}(w^i_{\kappa_{\mathbb{S}}}, v^i_{\kappa_{\mathbb{S}}}),\\
\tilde{S}(w_h, v_h) &= \sum_{\kappa'_{\mathbb{S}}\in\mathcal{T}_{\mathbb{S}}} \sum_{j=1}^{(q_{\kappa'_{\mathbb{S}}}+1)^{d-1}} \sum_{\kappa_{\mathbb{S}}\in\mathcal{T}_{\mathbb{S}}} \sum_{i=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}} \tilde{S}^{j,i}_{\kappa'_{\mathbb{S}},\kappa_{\mathbb{S}}}(w^j_{\kappa'_{\mathbb{S}}}, v^i_{\kappa_{\mathbb{S}}}),\\
\tilde{\ell}(v_h) &= \sum_{\kappa_{\mathbb{S}}\in\mathcal{T}_{\mathbb{S}}} \sum_{i=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}} \omega^i_{\kappa_{\mathbb{S}}}\tilde{\ell}^i_{\kappa_{\mathbb{S}}}(v^i_{\kappa_{\mathbb{S}}}),
\end{aligned}$$

the mono-energetic DOG scheme reads as follows: find $u_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ such that

$$\tilde{T}(u_h, v_h) = \tilde{S}(u_h, v_h) + \tilde{\ell}(v_h) \tag{5.48}$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$. Notice that we may replace $\tilde{S}(u_h, v_h)$ with $S(u_h, v_h)$ since

$$\begin{aligned}
\tilde{S}(u_h, v_h) &= \sum_{\kappa'_{\mathbb{S}}\in\mathcal{T}_{\mathbb{S}}} \sum_{j=1}^{(q_{\kappa'_{\mathbb{S}}}+1)^{d-1}} \sum_{\kappa_{\mathbb{S}}\in\mathcal{T}_{\mathbb{S}}} \sum_{i=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}} \tilde{S}^{j,i}_{\kappa'_{\mathbb{S}},\kappa_{\mathbb{S}}}(u^j_{\kappa'_{\mathbb{S}}}, v^i_{\kappa_{\mathbb{S}}})\\
&= \sum_{\kappa'_{\mathbb{S}}\in\mathcal{T}_{\mathbb{S}}} \sum_{j=1}^{(q_{\kappa'_{\mathbb{S}}}+1)^{d-1}} \sum_{\kappa_{\mathbb{S}}\in\mathcal{T}_{\mathbb{S}}} \sum_{i=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}} S(u^j_{\kappa'_{\mathbb{S}}}\varphi^j_{\kappa'_{\mathbb{S}}}, v^i_{\kappa_{\mathbb{S}}}\varphi^i_{\kappa_{\mathbb{S}}})\\
&= S(u_h, v_h).
\end{aligned}$$

It is straightforward to derive a source iteration method for (5.48): for a given $u^{(0)}_h \in \mathcal{V}_{\Omega,\mathbb{S}}$, find $\{u_h\}_{n\geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S}}$ such that

$$\tilde{T}(u^{(n+1)}_h, v_h) = \tilde{S}(u^{(n)}_h, v_h) + \tilde{\ell}(v_h) \tag{5.49}$$

for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$ and $n \geq 0$.

Analysis

For the analysis of (5.49), it shall be useful to introduce two norms $|||\cdot|||_T : \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$ and $|||\cdot|||_{\tilde{T}} : \mathcal{V}_{\Omega,\mathbb{S}} \to \mathbb{R}$:

$$
\begin{aligned}
|||v|||_T^2 &= ||\sqrt{\alpha+\beta}v||_{L^2(\mathcal{D})}^2 \\
&\quad + \frac{1}{2}\int_{\mathbb{S}} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} \left(||v^+ - v^-||_{\partial_- \kappa_\Omega(\boldsymbol{\mu})\setminus\partial\Omega}^2 + ||v^+||_{\partial_{\kappa_\Omega}(\boldsymbol{\mu})\cap\partial\Omega}^2 \right) \mathrm{d}\boldsymbol{\mu}, \\
|||v_h|||_{\tilde{T}}^2 &= \sum_{\kappa_{\mathbb{S}} \in \mathcal{T}_{\mathbb{S}}} \sum_{i=1}^{(q_{\kappa_{\mathbb{S}}}+1)^{d-1}} \omega_{\kappa_{\mathbb{S}}}^i \Bigg(||\sqrt{\alpha+\beta}v_{\kappa_{\mathbb{S}}}^i||_{L^2(\Omega)}^2 \\
&\quad + \frac{1}{2} \sum_{\kappa_\Omega \in \mathcal{T}_\Omega} ||(v_{\kappa_{\mathbb{S}}}^i)^+ - (v_{\kappa_{\mathbb{S}}}^i)^-||_{\partial_- \kappa_\Omega(\boldsymbol{\mu}_{\kappa_{\mathbb{S}}}^i)\setminus\partial\Omega}^2 + ||(v_{\kappa_{\mathbb{S}}}^i)^+||_{\partial_- \kappa_\Omega(\boldsymbol{\mu}_{\kappa_{\mathbb{S}}}^i)\cap\partial\Omega}^2 \Bigg).
\end{aligned}
$$

Note that $|||\cdot|||_{\tilde{T}}$ is the natural norm in which to measure coercivity of the bilinear form $\tilde{T}$; moreover, we have the following identity for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$:

$$
|||v_h|||_{\tilde{T}}^2 = \tilde{T}(v_h, v_h).
$$

The norm $|||\cdot|||_{\tilde{T}}$ may be interpreted as an approximation of the norm $|||\cdot|||_T$ using the previously-defined angular quadrature scheme. As such, we have that $|||v_h|||_{\tilde{T}} \approx |||v_h|||_T$ for all $v_h \in \mathcal{V}_{\Omega,\mathbb{S}}$. Denoting by u_h the exact solution to the DOG scheme (5.48) and $\{u_h^{(n)}\}_{n\geq 0} \subset \mathcal{V}_{\Omega,\mathbb{S}}$ the sequence of approximate solutions generated by the iteration (5.49), we have that

$$
\begin{aligned}
|||u_h^{(n+1)} - u_h|||_{\tilde{T}}^2 &= \tilde{T}(u_h^{(n+1)} - u_h, u_h^{(n+1)} - u_h) \\
&= \tilde{S}(u_h^{(n)} - u_h, u_h^{(n+1)} - u_h) \\
&= S(u_h^{(n)} - u_h, u_h^{(n+1)} - u_h) \\
&\leq c\ |||u_h^{(n)} - u_h|||_T |||u_h^{(n+1)} - u_h|||_T \\
&\approx c\ |||u_h^{(n)} - u_h|||_{\tilde{T}} |||u_h^{(n+1)} - u_h|||_{\tilde{T}}.
\end{aligned}
$$

We point out that, since we were able to introduce the bilinear form $S(\cdot,\cdot)$, Lemmas 5.2.4 5.2.5 were invoked (with the choice of relaxation parameter $\omega = 0$) in order to obtain the first bound on the solver error $|||u_h^{(n+1)} - u_h|||_{\tilde{T}}$. The result above may be rearranged to yield

$$
|||u_h^{(n+1)} - u_h|||_{\tilde{T}} \lessapprox c\ |||u_h^{(n)} - u_h|||_{\tilde{T}}. \tag{5.50}
$$

Note that (5.50) is insufficient to prove that the iteration (5.49) is convergent since we have not bounded the error incurred by replacing the norm $|||\cdot|||_T$ with $|||\cdot|||_{\tilde{T}}$.

5.6 Summary

As we have seen in Chapter 3, the mono- and poly-energetic forms of the time-independent linear Boltzmann transport equation can be discretised using discontinuous Galerkin finite elements in the spatial, angular and energetic domains, and that

the resulting scheme (when solved exactly) is convergent with optimal order in the DG-energy norm error. However, the linear systems arising from such discretisations are typically large and sparse, so iterative solution methods are more suitable than direct methods. The most common iterative method employed in radiation transport codes, called *source iteration*, has been shown to be convergent when applied to model (continuum) mono-energetic problems.

In this chapter, we started by verifying two important properties of source iteration. Firstly, we showed (under relatively general assumptions) that the convergence properties of source iteration in the continuum case are maintained when the iterative method is applied to the discrete equations arising from a fully-discontinuous Galerkin finite element discretisation of the LBTE. Secondly, we extended the convergence result to poly-energetic problems by proving a contractive property about source iteration in the poly-energetic case. The resulting contraction factor is analogous to the *scattering ratio* for mono-energetic problems. While not presented here, we have observed that the mesh size parameter h, as well as the polynomial degree of approximation p, have little influence on the convergence rate of source iteration for poly- or mono-energetic problems.

We also derived computable *a posteriori* error estimates for the DG-energy norm solver error $|||u_h - \hat{u}_h|||_{DG}$, where u_h denotes the exact solution to the DGFEM approximation of the LBTE and $\hat{u}_h \approx u_h$ denotes an approximation formed by terminating a linear solver for the discrete equations prematurely. One of these *a posteriori* error estimates is applicable only for approximate solutions generated by source iteration; the other estimate applies to approximate solutions generated by more general linear solvers and is based on the residual vector corresponding to the linear system. While the effect of finite element discretisation parameters have not been presented here, we have observed in previous experiments that they only slightly change the computed values of the *a posteriori* error estimates.

We looked at a modification of source iteration in the mono-energetic case and again proved a convergence result and *a posteriori* DG-energy norm solver error estimate. We found that the modified source iteration method converges slightly faster than standard source iteration, and drew comparisons between the so-called *modified source iteration* with successively over-relaxed treatments of source iteration. These comparisons included a study on the spectral properties of the discrete iteration operators. We also showed how the generalised minimal residual (GMRES) method can be applied to the DGFEM-discretised LBTE. By a slight modification of the linear system, we saw that GMRES could compute a residual-based *a posteriori* DG-energy norm solver error estimate at each iteration as a byproduct. Moreover, the implementation is compatible with any (right-)preconditioner; we focussed on families of preconditioners based on employing standard transport sweeps.

We numerically studied a range of iterative methods applied to a model mono-energetic test problem in two spatial dimensions. We validated that, under certain conditions, the modified source iteration method and right-preconditioned GMRES methods (using transport-based preconditioners) are both more rapidly convergent than source iteration. More specifically, we identified two parameters that control the rate of convergence of each method, namely the scattering ratio c and the number of mean free paths required to travel across the spatial domain λL. We remark that the latter parameter was used as a surrogate for the cell aspect ratio ε defined in (5.1) and appeared as an important quantity of interest in the analysis of Chapter 5.3.3. We also saw how these parameters affect the effectivities of the *a posteriori* solver error estimates employed by each method.

From the numerical results presented above, it is clear that right-preconditioned GMRES using transport-sweep-based preconditioners offers consistently-faster convergence rates than either of the stationary iterative methods. In particular, the solver RPGMRES-T(1) is only slightly more computationally expensive per step than standard source iteration. To see this, note that a single step of source iteration requires the computations of the actions of $\mathbf{T}^{-1}$ and $\mathbf{S}$ on a vector, while a single step of RPGMRES-T(1) additionally requires the computations of the actions of $\mathbf{L}$ and $\mathbf{L}^{-1}$ on a vector, as well as an orthogonalisation step. However, the actions of $\mathbf{L}$ and $\mathbf{L}^{-1}$ are relatively cheap due to their block-diagonal structure. In practice, the computation of the action of $\mathbf{S}$ on a vector is the dominating cost (in terms of CPU time) in a single step of both methods, and so each iteration of RPGMRES-T(1) takes only slightly more CPU time than that of source iteration. The rapid convergence of RPGMRES-T(1) compared to source iteration means that RPGMRES-T(1) generally takes less CPU time overall to achieve a given solver tolerance.

For $n > 1$, each step of RPGMRES-T(n) requires the repeated actions of $\mathbf{T}^{-1}$ and $\mathbf{S}$ on a vector n times. Moreover, we have demonstrated that the solver error after n steps of RPGMRES-T(1) is generally smaller than one step of RPGMRES-T(1). Therefore, RPGMRES-T(1) is always preferred over RPGMRES-T(n) if one is able to store a large Krylov basis. However, RPGMRES-T(n) offers a reasonable compromise between rapid convergence rates and reasonable storage requirements when only a few Krylov vectors can be stored.

If memory constraints are severe enough to rule out RPGMRES-T(n), then the modified source iteration method MSI(ω) is preferred over standard source iteration. It was found that a single iteration of both methods took approximately the same amount of CPU time, and MSI(ω) can be incorporated into existing source iteration codes in a relatively straightforward manner. We recommend the parameter choice $\omega = \frac{1}{2}$ since this guarantees the convergence of MSI(ω) in every example provided. While choices of ω greater than $\frac{1}{2}$ can offer faster convergence rates of MSI(ω), their convergence is no

longer guaranteed, particularly if either the scattering ratio c or the optical thickness ε is large.

We also generalised the transport-sweep preconditioner employed in RPGMRES-T(n) to additionally incorporate information about the spectral centre of the source-iteration operator $\mathbf{T}^{-1}\mathbf{S}$; i.e. the centre of the complex disc which we proved bounded the eigenvalues of the source-iteration operator. Our numerical experiments suggest that such preconditioners can yield improved convergence rates of RPGMRES-T(n) over preconditioners employing standard transport sweeps provided that the spectral centre is sufficiently well-approximated. However, our experiments also suggest that this improvement is relatively small. Given the sensitivity of the convergence behaviour of RPGMRES-T(n) on the quality of the approximation of the spectral centre, it may be advisable to employ standard sweep-based (i.e. truncated Neumann) preconditioners.

Chapter 6

Conclusions

In this thesis we have developed high-order discontinuous Galerkin finite element methods (DGFEMs) for the numerical approximation of the mono- and poly-energetic forms of the linear Boltzmann transport equation (LBTE). Our development employed DGFEMs in each of the spatial, angular and energetic domains. It was shown that the resulting scheme can be efficiently implemented into many multigroup discrete-ordinates codes for radiation transport. The remaining work focussed on the fast assembly and solution of the resulting equations.

In Chapter 3, we introduced families of discretisations for each of the spatial, angular and energetic domains, and specified spaces of discontinuous piecewise-polynomial functions in each case. These function spaces were necessary to develop a full space-angle-energy discretisation of the poly-energetic linear Boltzmann transport equation using the discontinuous Galerkin finite element method. In particular, the angular and energetic function spaces introduced permitted an implementation of the resulting method in a multigroup discrete-ordinates-like fashion. We have demonstrated that the order of accuracy of the resulting DGFEM is optimal with respect to the space-angle-energy mesh-size parameter h and the global polynomial degree of approximation p; i.e. that the DG-energy norm error in the computed DGFEM approximation scales like $O(h^{p+1/2})$ and that the $L^2(\mathcal{D})$ norm error scales like $O(h^{p+1})$.

In Chapter 4, we investigated the assembly of the linear system arising from a DGFEM discretisation of the constant-coefficient first-order linear transport equation on arbitrary polytopic meshes. In particular, we studied the assembly of the system matrix using both standard quadrature-based procedures and novel quadrature-free-based procedures based on the fast numerical integration of homogeneous functions on polytopes. A quadrature-free assembly method was developed that assembles local matrix contributions using a single loop over mesh faces, as opposed to two separate loops over elements and faces. An analysis of the floating-point operation count of the resulting method was performed and compared to a corresponding quadrature-based method, un-

der relatively general assumptions on the volume quadrature scheme employed on each element. The analysis revealed that the quadrature-free-based assembly algorithm could outcompete the quadrature-based algorithm; in particular, a mesh-dependent parameter was identified that partially characterised the performance improvement in switching to a quadrature-free-based approach. This was verified through numerical examples.

In Chapter 5, we studied the classical source iteration method for the solution of the linear Boltzmann transport equation. Our reason for this was to verify that the convergence properties of source iteration applied to the continuum problem are retained when a full DGFEM discretisation in the space-angle-energy domain is performed. We also obtained a more general convergence result for poly-energetic source iteration. We paid particular attention to the derivation of *a posteriori* solver error estimates which exploited the variational framework of the DGFEM problem. We then focussed on mono-energetic problems and introduced a new basic iterative solver, coined the "modified source iteration" method, which generalises standard source iteration via the introduction of a tailorable parameter. We also discussed the application of the generalised minimal residual (GMRES) method to problems in radiation transport. We described a framework in which standard implementations of GMRES may be exploited to incorporate *a posteriori* solver error estimation. The convergence properties of source iteration, modified source iteration and transport-preconditioned GMRES were demonstrated with numerical examples. It was observed that transport-sweep-preconditioned GMRES almost always outperformed standard source iteration in terms of the total CPU time taken to achieve a given user-specified solver tolerance, provided that one has sufficient storage for the Krylov vectors. Moreover, it was found that employing multiple transport sweeps per GMRES step offered a good compromise between memory and total CPU time. In cases where even a small number of Krylov vectors cannot be stored, we have shown that the modified source iteration (for specific choices of the tailorable parameter) often converges faster than standard source iteration.

6.1 Further work

The work considered in this thesis suggests several topics of further research interest which we shall briefly discuss.

6.1.1 Improved linear solvers

A number of extensions to the linear solvers presented in Chapter 5 can be made. Most notably, a comparison of the aforementioned methods against the widely-employed *diffusion-synthetic acceleration* (DSA) method would be highly valuable. For mono-energetic problems, it is known that DSA converges more rapidly than source iteration, provided that the scattering kernel is not too highly-peaked and that the DSA equations

are discretised "consistently" with the LBTE [89, 103]. A proof of convergence of DSA employing functional-analytic methods (as was performed in Chapter 5.5.1 for source iteration) would be highly valuable, particularly if it could be extended to poly-energetic problems. It would also be useful to employ DSA-based (or approximate DSA-based) preconditioners for the right-preconditioned GMRES method. Such preconditioners may accelerate the convergence of GMRES (and thus minimise the size of the stored Krylov basis) for test problems where standard transport-based preconditioners converge most slowly. Finally, the variational framework employed in the DGFEM discretisation naturally lends itself to the development of multigrid-based preconditioners. This may be especially useful for problems with highly-peaked scattering kernels, where it has been observed (for discrete ordinate calculations) that "angular multigrid" methods can outperform DSA in terms of convergence rates [1].

6.1.2 Further applications of quadrature-free methods

The primary application of quadrature-free methods employed in this work was for the assembly of the matrix arising from the discontinuous Galerkin discretisation of the constant-coefficient first-order linear transport equation. While quadrature-free methods have also been applied to second-order elliptic problems [6], the approach outlined earlier can also be used for discontinuous Galerkin discretisations of more general problems [28]. We have also not addressed the problem of assembling integral terms involving more general non-polynomial functions. Outside of a few special cases [69], integrands involving products and compositions of homogeneous and non-homogeneous functions cannot be integrated using quadrature-free techniques, and one typically resorts to standard quadrature schemes to integrate such functions. An alternative approach might be to approximate such integrands with a linear combination of homogeneous functions and then invoke quadrature-free arguments to exactly integrate the resulting (approximate) integral. It is also of practical interest to investigate code-optimisation strategies that may speed up the general quadrature-free assembly outlined earlier. While our approach allows for the assembly of the system matrix by looping over mesh faces with little regard to the order of faces visited, it is likely that looping over the boundary faces of each element may offer improvements in assembly time. This is because one may then assemble volume-like contributions once per element rather than once per face, and meshes typically have fewer elements than faces.

6.1.3 Functional error control

We have cast the numerical approximation of the linear Boltzmann transport equation within a fully variational framework, which allows for both greater flexibility in the finite element spaces/meshes used to discretise the equation and functional error control.

Provided that one can define an appropriate "goal" functional $J(\cdot)$, one may use dual-weighted residual techniques to formulate and solve a dual problem. The approximate primal and dual solutions may then be used to quantify the difference between $J(u)$, the goal functional evaluated at the exact solution (which we typically have no access to), and $J(u_h)$, the goal functional evaluated at the approximate DGFEM solution (which we can solve for). Moreover, such techniques often provide *error indicators* - additional information about the element-wise contributions to the functional error. This information can be exploited within a "goal-oriented" adaptive mesh refinement algorithm [49, 52]. The variational framework of finite element methods can also accommodate the approximation of contributions to functional error estimates arising from sources other than discretisation. For example, one could incorporate error indicators associated with both the "discrete ordinates Galerkin" implementation of Chapter 3.4 and any of the approximate linear solvers described in Chapter 5.

6.1.4 Medical physics applications

The work presented in this thesis is geared towards the analysis of discontinuous Galerkin discretisations of the linear Boltzmann transport equation and its numerical solution. However, the motivation for this work was the application of such techniques to problems in radiotherapy treatment planning. In the most complex problem considered in this work, we studied the convergence of our DGFEM approximation applied to a model problem consisting of Compton-scattering photons travelling through a 2D slab of water - there are many more realistic and complex problems we have yet to consider. For example, we have not yet applied our methods to problems with strong spatial heterogeneities in the material coefficients and have also not incorporated other scattering and absorption processes into our physics model. In order to show that our deterministic approach can compete with current "gold-standard" Monte Carlo methods, it is necessary to benchmark our method against more demanding problems of interest to the medical physics community; cf. [61, 71, 101]. We have also focussed primarily on photon scattering/absorption physics, although our analysis also applies to more general radiative particle physics. One extension to our model is to additionally couple photon and electron fluences via a coupled system of linear Boltzmann transport equations - this requires further study of the scattering/absorption physics of electrons. In view of obtaining accurate radiative dose estimates, we have also yet to implement "dose delivery" functionals within a dual-weighted residual framework - although photon sources are often used in radiotherapy, it is the electrons liberated from Compton scattering that actually deliver a radiative dose to the medium [11].

Bibliography

[1] Adams, M. L., and Larsen, E. W. Fast iterative methods for discrete-ordinates particle transport calculations. *Progress in Nuclear Energy 40*, 1 (2002), 3–159.

[2] Agostinelli, S., Allison, J., Amako, K. a., Apostolakis, J., Araujo, H., Arce, P., Asai, M., Axen, D., Banerjee, S., Barrand, G., et al. GEANT4—a simulation toolkit. *Nuclear instruments and methods in physics research section A: Accelerators, Spectrometers, Detectors and Associated Equipment 506*, 3 (2003), 250–303.

[3] Ahrens, C. D. Lagrange discrete ordinates: a new angular discretization for the three-dimensional linear Boltzmann equation. *Nuclear Science and Engineering 180*, 3 (2015), 273–285.

[4] Alcouffe, R. E. Diffusion synthetic acceleration methods for the diamond-differenced discrete-ordinates equations. *Nuclear Science and Engineering 64*, 2 (1977), 344–355.

[5] Antonietti, P., Houston, P., and Smears, I. A note on optimal spectral bounds for nonoverlapping domain decomposition preconditioners for *hp*–version discontinuous Galerkin methods. *International Journal of Numerical Analysis and Modeling 13*, 4 (2016), 513–524.

[6] Antonietti, P. F., Houston, P., and Pennesi, G. Fast numerical integration on polytopic meshes with applications to discontinuous Galerkin finite element methods. *Journal of Scientific Computing* (2018), 1–32.

[7] Asadzadeh, M. Analysis of a fully discrete scheme for neutron transport in two-dimensional geometry. *SIAM Journal on Numerical Analysis 23*, 3 (1986), 543–561.

[8] Askew, J. A characteristics formulation of the neutron transport equation in complicated geometries. Tech. rep., United Kingdom Atomic Energy Authority, 1972.

[9] Badal, A., Kyprianou, I., Badano, A., and Sempau, J. Monte Carlo simulation of a realistic anatomical phantom described by triangle meshes: application to prostate brachytherapy imaging. *Radiotherapy and Oncology 86*, 1 (2008), 99–103.

[10] Bakkali, J. E., and Bardouni, T. E. Validation of Monte Carlo Geant4 code for a 6 MV Varian linac. *Journal of King Saud University-Science 29*, 1 (2017), 106–113.

[11] Bedford, J. L. Calculation of absorbed dose in radiotherapy by solution of the linear Boltzmann transport equations. *Physics in Medicine and Biology* (2018).

[12] Beentjes, C. H. Quadrature on a spherical surface. *Working note available on the website http://people. maths. ox. ac. uk/beentjes/Essays* (2015).

[13] Beirão da Veiga, L., Brezzi, F., Cangiani, A., Manzini, G., Marini, L. D., and Russo, A. Basic principles of virtual element methods. *Mathematical Models and Methods in Applied Sciences 23*, 01 (2013), 199–214.

[14] Beirão da Veiga, L., Brezzi, F., Marini, L. D., and Russo, A. The hitchhiker's guide to the virtual element method. *Mathematical Models and Methods in Applied Sciences 24*, 08 (2014), 1541–1573.

[15] Bell, G. I., and Glasstone, S. Nuclear reactor theory. Tech. rep., US Atomic Energy Commission, Washington, DC (United States), 1970.

[16] Benedetto, M. F., Berrone, S., Borio, A., Pieraccini, S., and Scialo, S. Order preserving SUPG stabilization for the virtual element formulation of advection–diffusion problems. *Computer Methods in Applied Mechanics and Engineering 311* (2016), 18–40.

[17] Bennison, T. *Adaptive discontinuous Galerkin methods for the neutron transport equation*. PhD thesis, University of Nottingham, 2014.

[18] Berger, M. J. XCOM: Photon cross section database. *http://physics. nist. gov/xcom* (1999).

[19] Berger, M. J., and Seltzer, S. M. Stopping powers and ranges of electrons and positrons. *Unknown* (1982).

[20] Berrone, S., Borio, A., and Manzini, G. SUPG stabilization for the nonconforming virtual element method for advection–diffusion–reaction equations. *Computer Methods in Applied Mechanics and Engineering 340* (2018), 500–529.

[21] Boellaard, R., Essers, M., Van Herk, M., and Mijnheer, B. J. New method to obtain the midplane dose using portal in vivo dosimetry. *International Journal of Radiation Oncology* Biology* Physics 41*, 2 (1998), 465–474.

[22] Börgers, C. Complexity of Monte Carlo and deterministic dose-calculation methods. *Physics in Medicine & Biology 43*, 3 (1998), 517.

[23] Brezzi, F., and Marini, L. Virtual element and discontinuous Galerkin methods. In *Recent developments in discontinuous Galerkin finite element methods for partial differential equations*. Springer, 2014, pp. 209–221.

[24] Buluç, A., Meyerhenke, H., Safro, I., Sanders, P., and Schulz, C. Recent advances in graph partitioning. In *Algorithm Engineering*. Springer, 2016, pp. 117–158.

[25] Bünger, J., Sarna, N., and Torrilhon, M. Stable boundary conditions and discretization for PN equations. *arXiv preprint arXiv:2004.02497* (2020).

[26] Cancer Research UK. Cancer incidence for all cancers combined. https://www.cancerresearchuk.org/health-professional/cancer-statistics/incidence/all-cancers-combined. Accessed: 2022-11-16.

[27] Cangiani, A., Dong, Z., Georgoulis, E. H., and Houston, P. hp-version discontinuous Galerkin methods for advection-diffusion-reaction problems on polytopic meshes. *ESAIM: Mathematical Modelling and Numerical Analysis 50*, 3 (2016), 699–725.

[28] Cangiani, A., Dong, Z., Georgoulis, E. H., and Houston, P. *hp-version discontinuous Galerkin methods on polygonal and polyhedral meshes*. SpringerBriefs in Mathematics. Springer International Publishing, Berlin, 2017.

[29] Cangiani, A., Georgoulis, E. H., and Houston, P. hp-version discontinuous Galerkin methods on polygonal and polyhedral meshes. *Mathematical Models and Methods in Applied Sciences 24*, 10 (2014), 2009–2041.

[30] Cangiani, A., Manzini, G., and Sutton, O. J. Conforming and nonconforming virtual element methods for elliptic problems. *IMA Journal of Numerical Analysis 37*, 3 (2017), 1317–1354.

[31] Chazelle, B. Triangulating a simple polygon in linear time. *Discrete & Computational Geometry 6*, 3 (1991), 485–524.

[32] Chen, J.-Y., Kincaid, D. R., and Young, D. M. Generalizations and modifications of the GMRES iterative method. *Numerical Algorithms 21*, 1 (1999), 119–146.

[33] Chin, E. B., Lasserre, J. B., and Sukumar, N. Numerical integration of homogeneous functions on convex and nonconvex polygons and polyhedra. *Computational Mechanics 56*, 6 (2015), 967–981.

[34] Chin, E. B., and Sukumar, N. An efficient method to integrate polynomials over polytopes and curved solids. *Computer Aided Geometric Design 82* (2020), 101914.

[35] Ciarlet, P. G., Kesavan, S., Ranjan, A., and Vanninathan, M. *Lectures on the finite element method*, vol. 49. Tata Institute of Fundamental Research Bombay, 1975.

[36] Collaboration, G., et al. Physics reference manual. *Version: Geant4 9*, 0 (2016).

[37] Currell, F., and Villagomez-Bernabe, B. Physical and chemical processes for gold nanoparticles and ionising radiation in medical contexts. *Gold Nanoparticles for Physics, Chemistry and Biology; World Scientific: Singapore* (2017), 509–536.

[38] Di Pietro, D. A., and Ern, A. *Mathematical aspects of discontinuous Galerkin methods*, vol. 69. Springer Science & Business Media, 2011.

[39] Duffy, M. G. Quadrature over a pyramid or cube of integrands with a singularity at a vertex. *SIAM Journal on Numerical Analysis 19*, 6 (1982), 1260–1262.

[40] Egger, H., and Schlottbom, M. A class of Galerkin schemes for time-dependent radiative transfer. *SIAM Journal on Numerical Analysis 54*, 6 (2016), 3577–3599.

[41] Embree, M. How descriptive are GMRES convergence bounds? *arXiv preprint arXiv:2209.01231* (2022).

[42] Ern, A., and Guermond, J.-L. *Theory and practice of finite elements*, vol. 159. Springer, 2004.

[43] Führer, C., and Kanschat, G. A posteriori error control in radiative transfer. *Computing 58*, 4 (1997), 317–334.

[44] Gasparo, M., Papini, A., and Pasquali, A. Some properties of GMRES in Hilbert spaces. *Numerical Functional Analysis and Optimization 29*, 11-12 (2008), 1276–1285.

[45] Gifford, K. A., Horton Jr, J. L., Wareing, T. A., Failla, G., and Mourtada, F. Comparison of a finite-element multigroup discrete-ordinates code with Monte Carlo for radiotherapy calculations. *Physics in Medicine & Biology 51*, 9 (2006), 2253.

[46] Greenbaum, A., Pták, V., and Strakoš, Z. e. k. Any nonincreasing convergence curve is possible for GMRES. *Siam Journal on Matrix Analysis and Applications 17*, 3 (1996), 465–469.

[47] Günnel, A., Herzog, R., and Sachs, E. A note on preconditioners and scalar products in Krylov subspace methods for self-adjoint problems in Hilbert space. *Electron. Trans. Numer. Anal 41* (2014), 13–20.

[48] Hall, E., Houston, P., and Murphy, S. hp-Adaptive discontinuous Galerkin methods for neutron transport criticality problems. *SIAM Journal on Scientific Computing 39*, 5 (2017), B916–B942.

[49] Hartmann, R., and Houston, P. Goal-oriented a posteriori error estimation for multiple target functionals. In *Hyperbolic problems: theory, numerics, applications*. Springer, 2003, pp. 579–588.

[50] Hensel, H., Iza-Teran, R., and Siedow, N. Deterministic model for dose calculation in photon radiotherapy. *Physics in Medicine & Biology 51*, 3 (2006), 675.

[51] Horn, R. A., and Johnson, C. R. *Matrix analysis*. Cambridge University press, 2012.

[52] Houston, P. Adjoint error estimation and adaptivity for hyperbolic problems. In *Handbook of Numerical Analysis*, vol. 18. Elsevier, 2017, pp. 233–261.

[53] Houston, P., Hubbard, M. E., Radley, T. J., Sutton, O. J., and Widdowson, R. S. J. Efficient High-Order Space-Angle-Energy Polytopic Discontinuous Galerkin Finite Element Methods for Linear Boltzmann Transport, 2023.

[54] Houston, P., Schwab, C., and Süli, E. Discontinuous hp-finite element methods for advection-diffusion-reaction problems. *SIAM Journal on Numerical Analysis 39*, 6 (2002), 2133–2163.

[55] Hubbell, J. H., Gimm, H. A., and Ø verbø, I. Pair, triplet, and total atomic cross sections (and mass attenuation coefficients) for 1 MeV-100 GeV photons in elements Z= 1 to 100. *Journal of Physical and Chemical Reference Data 9*, 4 (1980), 1023–1148.

[56] Jaffray, D. A., and Gospodarowicz, M. K. Radiation Therapy for Cancer. In *Disease Control Priorities: Cancer*, H. Gelband, P. Jha, R. Sankaranarayanan, and S. Horton, Eds., vol. 3. The World Bank, 2015, ch. 14, pp. 239–247.

[57] Johnson, C., and Pitkäranta, J. Convergence of a fully discrete scheme for two-dimensional neutron transport. *SIAM Journal on Numerical Analysis 20*, 5 (1983), 951–966.

[58] Kanschat, G. *Parallel and adaptive Galerkin methods for radiative transfer problems*. PhD thesis, 1996.

[59] Karypis, G., and Kumar, V. METIS: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices.

[60] Kavanagh, B., Ding, M., Schefter, T., Stuhr, K., and Newman, F. The dosimetric effect of inhomogeneity correction in dynamic conformal arc stereotactic body radiation therapy for lung tumors. *Journal of Applied Clinical Medical Physics 7*, 2 (2006), 58–63.

[61] Kawrakow, I., and Fippel, M. Investigation of variance reduction techniques for Monte Carlo photon dose calculation using XVMC. *Physics in Medicine & Biology 45*, 8 (2000), 2163.

[62] Kim, C. H., Choi, S. H., Jeong, J. H., Lee, C., and Chung, M. S. HDRK-Man: a whole-body voxel model based on high-resolution color slice images of a Korean adult male cadaver. *Physics in Medicine & Biology 53*, 15 (2008), 4093.

[63] Kirby, R. C. From functional analysis to iterative methods. *SIAM Review 52*, 2 (2010), 269–293.

[64] Klein, O., and Nishina, Y. Über die Streuung von Strahlung durch freie Elektronen nach der neuen relativistischen Quantendynamik von Dirac. *Zeitschrift für Physik 52*, 11-12 (Nov. 1929), 853–868.

[65] Knepley, M. G., Rupp, K., and Terrel, A. R. Finite element integration with quadrature on the GPU. *arXiv preprint arXiv:1607.04245* (2016).

[66] Koch, R., Krebs, W., Wittig, S., and Viskanta, R. Discrete ordinates quadrature schemes for multidimensional radiative transfer. *Journal of Quantitative Spectroscopy and Radiative Transfer 53*, 4 (1995), 353–372.

[67] Kópházi, J., and Lathouwers, D. A space–angle DGFEM approach for the Boltzmann radiation transport equation with local angular refinement. *Journal of Computational Physics 297* (2015), 637–668.

[68] Kulikowska, T. An introduction to the neutron transport Phenomena. Tech. rep., 2001.

[69] Lasserre, J. Integration and homogeneous functions. *Proceedings of the American Mathematical Society 127*, 3 (1999), 813–818.

[70] Lewis, E. E., and Miller, W. F. Computational methods of neutron transport.

[71] Lewis, R., Ryde, S., Seaby, A., Hancock, D., and Evans, C. Use of Monte Carlo computation in benchmarking radiotherapy treatment planning system algorithms. *Physics in Medicine & Biology 45*, 7 (2000), 1755.

[72] Møller, C. Zur theorie des durchgangs schneller elektronen durch materie. *Annalen der Physik 406*, 5 (1932), 531–585.

[73] Mousavi, S., Xiao, H., and Sukumar, N. Generalized Gaussian quadrature rules on arbitrary polygons. *International Journal for Numerical Methods in Engineering 82*, 1 (2010), 99–113.

[74] Murphy, S. *Methods for solving discontinuous-Galerkin finite element equations with application to neutron transport.* PhD thesis, École Doctorale Mathématiques, Informatique et Télécommunications (Toulouse), 2015.

[75] NHS Inform. Radiotherapy. https://www.nhsinform.scot/tests-and-treatments/non-surgical-procedures/radiotherapy. Accessed: 2022-11-16.

[76] Papanikolaou, N., Battista, J. J., Boyer, A. L., Kappas, C., Klein, E., Mackie, T. R., Sharpe, M., and Van Dyk, J. Tissue inhomogeneity corrections for megavoltage photon beams. *AAPM Task Group 65* (2004), 1–142.

[77] Patton, B. W., and Holloway, J. P. Application of preconditioned GMRES to the numerical solution of the neutron transport equation. *Annals of Nuclear Energy 29*, 2 (2002), 109–136.

[78] Peterson, T. E. A note on the convergence of the discontinuous Galerkin method for a scalar hyperbolic equation. *SIAM Journal on Numerical Analysis 28*, 1 (1991), 133–140.

[79] Pomraning, G. The Fokker-Planck Operator as an Asymptotic Limit. *Mathematical Models and Methods in Applied Sciences 02*, 01 (Mar. 1992), 21–36.

[80] Ramirez, J. V., Chen, F., Nicolucci, P., and Baffa, O. Dosimetry of small radiation field in inhomogeneous medium using alanine/EPR minidosimeters and PENELOPE Monte Carlo simulation. *Radiation Measurements 46*, 9 (2011), 941–944.

[81] Reed, W. H., and Hill, T. R. Triangular mesh methods for the neutron transport equation. Tech. rep., Los Alamos Scientific Lab., N. Mex.(USA), 1973.

[82] Renken, J. H. Legendre Polynomial Expansion for the Klein-Nishina Formula. *Journal of Applied Physics 38*, 12 (1967), 4925–4927.

[83] Saad, Y. *Iterative methods for sparse linear systems.* SIAM, 2003.

[84] Saad, Y., and Schultz, M. H. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing 7*, 3 (1986), 856–869.

[85] Salvat, F., and Fernández-Varea, J. M. Overview of physical interaction models for photon and electron transport used in Monte Carlo codes. *Metrologia 46*, 2 (2009), S112.

[86] Schoenberg, I. Positive definite functions on spheres. *Duke Math. J 1* (1988), 172.

[87] Schötzau, D., Schwab, C., Wihler, T., and Wirz, M. Exponential convergence of hp-DGFEM for elliptic problems in polyhedral domains. In *Spectral and High Order Methods for Partial Differential Equations-ICOSAHOM 2012*. Springer, 2014, pp. 57–73.

[88] Simon, C. P., Blume, L., et al. *Mathematics for economists*, vol. 7. Norton New York, 1994.

[89] Southworth, B. S., Holec, M., and Haut, T. S. Diffusion synthetic acceleration for heterogeneous domains, compatible with voids. *Nuclear Science and Engineering 195*, 2 (2021), 119–136.

[90] Stacey, W. M. *Nuclear reactor physics*. John Wiley & Sons, 2018.

[91] Storm, L., and Israel, H. I. Photon cross sections from 1 keV to 100 MeV for elements Z= 1 to Z= 100. *Atomic Data and Nuclear Data Tables 7*, 6 (1970), 565–681.

[92] Stroud, A. Approximate Calculation of Multiple Integrals. Prentice-Hall Series in Automatic Computation.

[93] Sukumar, N., and Tabarraei, A. Conforming polygonal finite elements. *International Journal for Numerical Methods in Engineering 61*, 12 (2004), 2045–2066.

[94] Takeuchi, K. A numerical method for solving the neutron transport equation in finite cylindrical geometry. *Journal of Nuclear Science and Technology 6*, 8 (1969), 466–473.

[95] Talischi, C., Paulino, G. H., Pereira, A., and Menezes, I. F. PolyMesher: a general-purpose mesh generator for polygonal elements written in Matlab. *Structural and Multidisciplinary Optimization 45*, 3 (2012), 309–328.

[96] Tarjan, R. Depth-first search and linear graph algorithms. *SIAM Journal on Computing 1*, 2 (1972), 146–160.

[97] Thurgood, C., Pollard, A., and Becker, H. The TN quadrature set for the discrete ordinates method. *Journal of Heat Transfer 117*, 4 (1995), 1068–1070.

[98] Toro, E. F. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer Science & Business Media, 2013.

[99] Vassiliev, O. N., Wareing, T. A., McGhee, J., Failla, G., Salehpour, M. R., and Mourtada, F. Validation of a new grid-based Boltzmann equation solver for dose calculation in radiotherapy with photon beams. *Physics in Medicine & Biology 55*, 3 (2010), 581.

[100] Wang, D. Enhancing lpCMFD Acceleration with Successive Overrelaxation for Neutron Transport Source Iteration. *Nuclear Science and Engineering 195*, 1 (2021), 1–12.

[101] Wang, L., Lovelock, M., and Chui, C.-S. Experimental verification of a CT-based Monte Carlo dose-calculation method in heterogeneous phantoms. *Medical Physics 26*, 12 (1999), 2626–2634.

[102] Wang, Y. *Adaptive mesh refinement solution techniques for the multigroup SN transport equation using a higher-order discontinuous finite element method.* Texas A&M University, 2009.

[103] Warsa, J. S., Wareing, T. A., and Morel, J. E. Fully consistent diffusion synthetic acceleration of linear discontinuous SN transport discretizations on unstructured tetrahedral meshes. *Nuclear Science and Engineering 141*, 3 (2002), 236–251.

[104] Wihler, T. P., Frauenfelder, P., and Schwab, C. Exponential convergence of the hp-DGFEM for diffusion problems. *Computers & Mathematics with Applications 46*, 1 (2003), 183–205.