

A Discontinuous Galerkin Method for Charged Particle Transport in the Fokker-Planck Limit

Hennink, Aldo

October 08, 2015

This master's thesis is based on the work that was done for the final master's project, in partial fulfilment of the requirements of the Applied Physics track at the Technical University of Delft. It is to be defended publicly on 2015-10-15 at 14:00.

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

Student number:	1513222	
Thesis committee:	Dr. ir. D. Lathouwers,	TU Delft (supervisor)
	Prof. dr. ir. J. L. Kloosterman,	TU Delft
	Dr. N. V. Budko,	TU Delft

Abstract

A numerical scheme is presented for steady-state, mono-energetic charged particle transport in the Fokker-Planck limit. The spatial domain is meshed into elements, each of which has its own angular mesh. The basis is formed by a discontinuous Galerkin method both in space and in angle. The resulting discretisation is fully arbitrarily refinable in space and angle. No assumptions are made with regards to the shape of the domain. This could form the basis for a deterministic charged particle transport code.

A novel approach to spherical diffusion is presented in chapters 2 and 3. It is based on a spherical adaptation of the symmetric interior penalty method. It is developed both for regular discontinuous Galerkin methods and for cell-centred Galerkin finite element methods.

Chapter 4 introduces spatial streaming. The weak form is derived. For spatial basis functions of order p , the numerical solution converges with order $p + 1$ as the spatial mesh is refined. It converges to the exact value. That is, there is no approximation other than the Fokker-Planck limit.

Other original work includes an analysis of several basis functions on the unit sphere (appendix A) and an overview of numerical integration methods on the unit sphere (appendix B).

Preface

Particle transport has long been a field of interest in the Nuclear Energy and Radiation Applications (NERA) section of the Radiation Science and Technology (RST) Department. This was traditionally concentrated on neutron transport for nuclear reactors. In recent years, the research focus has shifted in part to charged particle transport. This is in large part due to the Holland Particle Therapy Centre, the proton therapy centre that is being built in Delft, next to the RST department.

This thesis develops a new deterministic numerical method for charged particle transport in the Fokker-Planck limit. Chapter 1 introduces the mathematical problem. It briefly explains why this is a difficult problem to solve numerically. Chapter 2 describes the theoretical foundation for a Discontinuous Galerkin method for the angular part of the transport problem. It is supported by appendix A, which deals with the properties of the basis functions on the sphere. Appendix B is concerned with spherical integrals. The appendices can be read on their own, and may be of interest to other fields, such as geomathematics. Chapter 3 expands on the previous chapter and provides numerical examples. Chapter 4 introduces spatial streaming. It also deals with linear solvers, and some numerical examples are given.

Though an attempt was made to keep this thesis mostly self-contained, it touches upon many aspects of finite element methods and particle transport, and it doesn't elaborate on the more well-known aspects. Special effort has been made to provide references to publicly available material that is relatively accessible to a non-expert. Nevertheless, this is probably a tough read for someone who's never implemented a finite element code.

Acknowledgement

The calculations in this thesis were performed with the in-house particle transport code ADAPT of the TU Delft, most of which was written by Danny Lathouwers.

I want to thank Danny Lathouwers for his support during my master's project. He helped with many things, but perhaps the most important was that he kept a good overview of the project, separating what's truly important from what it not.

Many thanks also go to the people at NERA in general, for making my stay fun. I will be back in the group soon.

Contents

Abstract	2
Preface	3
Acknowledgement	3
Contents	5
List of symbols	7
1 Introduction	8
1.1 The transport equation	8
1.2 Numerical difficulties in particle therapy	9
2 A DG method for spherical diffusion	13
2.1 Spherical SIPG	14
2.2 The angular mesh	17
2.3 The DG solution space: angular basis functions	20
2.3.1 Conditioning of the mass matrix and orthonormalisation	21
2.4 The ccG solution space	23
2.4.1 Adjustments to the spherical space	27
3 Numerical results of spherical SIP	28
3.1 Implementation	28
3.2 Order of convergence to analytical solution for DG	30
3.3 Anisotropic refinement	33
3.4 Conditioning of the global matrix	33
3.5 Numerical results of ccG SIPG	35
4 A finite element method for Fokker-Planck particle transport	39
4.1 The discrete setting: Riemann upwinding	40
4.2 Solution methods	43

4.2.1	Convergence of the linear solvers	44
4.3	Convergence with mesh refinement	45
4.4	A diffusing proton beam	48
5	Conclusion	51
6	Future work	52
A	Parameterisations of spherical elements	54
A.1	Parameterisations of orthodrome segments	54
A.1.1	Parameterisations via a line segment in \mathbb{R}^3	54
A.1.2	Parameterisation in angle	56
A.2	Parameterisations of spherical triangles	58
A.2.1	Parameterisations via a flat triangle in \mathbb{R}^3	59
A.2.2	A parameterisation in angles	61
A.2.3	A comparison of the Jacobians	65
B	Integrals over spherical elements	67
B.1	Integrals over orthodrome segments	68
B.1.1	Analytical results	68
B.1.2	Numerical integration rules	69
B.2	Integrals over spherical triangles	69
B.2.1	Numerical integration rules	71
	Bibliography	74

List of symbols

Symbol: **Meaning** [*ref. page*]:

f_i	i 'th angular basis function on a patch [20, 40]
$I^{n \times n}$	identity matrix in $\mathbb{R}^{n \times n}$
\mathbf{j}	current density [17]
L_{FP}	Fokker-Planck operator [13]
L_{str}	streaming operator [39]
\mathbf{r}	spatial position (in \mathbb{R}^3) [16]
S	volumetric angular source [8]
\mathbf{V}_i	i 'th vertex of a patch [22, 58, 70]
Y_{lm}	m 'th real spherical harmonic of order l [30]
α	macroscopic transport cross section (Fokker-Planck diffusion constant) [9]
δ_{ij}	Kronecker delta function
Σ_a	macroscopic absorption cross section [8]
φ	angular particle flux [8]
$\boldsymbol{\Omega}$	unit direction vector (solid angle; in \mathbb{S}^2) [8, 14]
\mathbb{S}^{d-1}	$\{\mathbf{q} \in \mathbb{R}^d: \ \mathbf{q}\ _2 = 1\}$ (d -dimensional sphere) [17]
∇_s	spherical gradient [14]
Δ_s	spherical Laplacian (Laplace-Beltrami) [9, 16]
$\ \mathbf{w}\ _p$	$(\sum_i (w_i ^p))^{1/p}$ (p -norm) [14]
$[\cdot]_F$	jump operator across the face F [16]
$\{\cdot\}_F$	average operator across the face F [16]

Chapter 1

Introduction

1.1 The transport equation

In a mono-energetic, time-independent model, the angular particle flux $\varphi = \varphi(\mathbf{r}, \boldsymbol{\Omega})$ depends on the spatial position \mathbf{r} and the (unit) direction vector $\boldsymbol{\Omega} \in \mathbb{S}^2$. The angular flux is determined by an appropriate set of boundary conditions in combination with the linear Boltzmann equation, the most general form of which is given by

$$\boldsymbol{\Omega} \cdot \frac{\partial}{\partial \mathbf{r}} \varphi + \Sigma_a \varphi - Q \varphi = S, \quad (1.1)$$

where $\boldsymbol{\Omega} \cdot \partial\varphi/\partial\mathbf{r}$ is the (spatial) streaming term, $\Sigma_a = \Sigma_a(\mathbf{r})$ is the macroscopic absorption cross section and $S = S(\mathbf{r}, \boldsymbol{\Omega})$ is the angular source. The effects of scattering in an isotopic medium are grouped together in

$$Q \varphi(\mathbf{r}, \boldsymbol{\Omega}) = \Sigma_S \left(\int_{\mathbb{S}^2} \frac{1}{2\pi} p(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) \varphi(\mathbf{r}, \boldsymbol{\Omega}') \, d\boldsymbol{\Omega}' - \varphi(\mathbf{r}, \boldsymbol{\Omega}) \right), \quad (1.2)$$

where $\Sigma_S(\mathbf{r})$ is the macroscopic scattering cross section and $p: [-1, 1] \mapsto \mathbb{R}^+$ is a probability density function. In the case of charged particle transport, the most important form of scattering is due to the Coulomb interactions between the charged particle and the nuclei. As a result, the direction vector undergoes a series of small deviations as the particle travels through the medium.

The linearity of equation 1.1 with 1.2 corresponds physically to the fact that there is no interaction between the particles. This assumption is valid for free charged particles in any solid or liquid, because the number of interactions with the nuclei in the medium always far outweighs the number of encounters with other free charged particles.

Equation 1.2 can be greatly simplified by the Fokker-Planck approximation. This model results from taking the limit of infinitely many collisions, whilst keeping constant the average total angular deviation per distance travelled. The average deviation per collision should therefore be infinitesimal, so that $p(\mu_0) = 0$ for $\mu_0 \neq 1$, and the integral drops out of equation 1.2. The standard derivation of the Fokker-Planck term begins with an expansion of φ and p into spherical harmonics and Legendre polynomials respectively. Subsequently, a Taylor expansion of the moments of $p(\mu_0)$ about $\mu_0 = 1$ is truncated. (Morel, 1981) (Uilkema, 2012) This yields

$$Q \varphi \rightarrow \frac{\alpha}{2} \Delta_s \varphi \equiv L_{FP} \varphi , \quad (1.3)$$

where Δ_s is the spherical Laplacian. It acts on Ω and is usually expressed in spherical coordinates as

$$\Delta_s = \frac{1}{\sin^2 \psi} \frac{\partial^2}{\partial \theta^2} + \frac{1}{\sin \psi} \frac{\partial}{\partial \psi} \left(\sin \psi \frac{\partial}{\partial \psi} \right) , \quad (1.4)$$

where θ and ψ are the polar and the azimuthal angles respectively. For a rigorous examination of the validity and accuracy of equation 1.3, see Borgers and Larsen (1996a). Note that it basically describes a diffusive process for Ω on the unit sphere \mathbb{S}^2 that is driven by many random independent small deviations. This is analogous to the more familiar diffusive process due to Brownian motion. The diffusion constant α is also called the (macroscopic) transport cross section. It can either be measured experimentally, or it can be expressed in terms of $p(\mu_0)$, which in turn can be determined from the first principles of electrodynamics and Lagrangian mechanics. These theoretical models can also be used to find a better fit of the measurement data.

1.2 Numerical difficulties in particle therapy

In particle therapy, a patient is irradiated by particles. Currently they are usually protons. Due to the electric charge, the particles engage in inelastic scatter events, thus slowing down as they travel through the patient. In combination with the energy dependence of the cross sections, this causes most particles to stop at a similar depth, which can be regulated by adjusting the energy of the incoming particles. The average energy that is deposited by a proton is highly localised in a fairly small area, called the Bragg peak. This is why proton therapy is thought to have great potential for the treatment of tumours. They can be irradiated while sparing the surrounding healthy tissues as much as possible. This also increases the demands on the accuracy

of the dose computation by the so-called dose engines; small computational errors can result in a significantly displaced radiation dose, thereby putting the patient at risk.

Unfortunately, efficient and generally applicable numerical methods for determining the radiation dose have not yet been devised. There are currently two main approaches that dose engines can take. The first is a Monte Carlo method. This is both accurate and flexible, but costly. Another disadvantage is that it does not facilitate a perturbation analysis, which is useful for the study of the effects of errors. Monte Carlo methods may still play an important role in the verification of other codes, or as a check of the final treatment plan. With the growth in computer power, Monte Carlo simulations are expected to become routine tools, especially in combination with some simplifying approximations. (Paganetti et al., 2008) (Jia et al., 2012)

Modern clinical treatment planning systems rely on the so-called (Fermi) pencil beam approximation. A simple heuristic derivation can be found in the original paper by Rossi and Greisen (1941), that attributes the reasoning to Fermi. For a more thorough analysis, see Borgers and Larsen (1996b). Consider the half-infinite homogeneous space $r_2 > 0$. There is a perpendicular infinitely narrow incident beam at $\mathbf{r} = \mathbf{0}$. In the absence of absorption, the angular flux is given by

$$\phi(\mathbf{r}, \boldsymbol{\Omega}) = \frac{3}{\pi^2 \alpha^2 r_2^4} \exp\left(\frac{-2}{\alpha} \left(\frac{\Omega_1^2 + \Omega_3^2}{r_2} - 3\frac{r_1 \Omega_1 + r_3 \Omega_3}{r_2^2} + 3\frac{r_1^2 + r_3^2}{r_2^3}\right)\right) \quad (1.5)$$

in the limit $\Omega_2 \gg |\Omega_1|$ and $\Omega_2 \gg |\Omega_3|$. Both Smith (2007) and Borgers and Larsen (1996a) claim that the error introduced by the Fokker-Planck approximation 1.3 is much greater than the error due to the Fermi pencil beam approximation. Nevertheless, the above equation is not easily adapted to inhomogeneous media or more general geometries, which, for obvious reasons, are of great importance in proton therapy planning. Figure 1.1 compares the pencil beam approximation to a Monte Carlo calculation for a typical brain tumour patient. There are clear differences in the predicted dose near the end of the beam. This is mostly due to the bone tissue, which has different material properties from the rest of the body. The pencil beams significantly predict a much lower radiation dose in the brain stem. In general, the pencil beams perform poorly in geometrically complex or highly inhomogeneous regions, but reasonably well otherwise. This motivates the search for an efficient numerical approximation of the Fokker-Planck model in general geometries.

Since Boltzmann equations in the form of 1.1 and 1.2 have many other applications, one may be tempted to use the numerical schemes that were developed there. The spatial part of the problem is meshed into voxels. Two

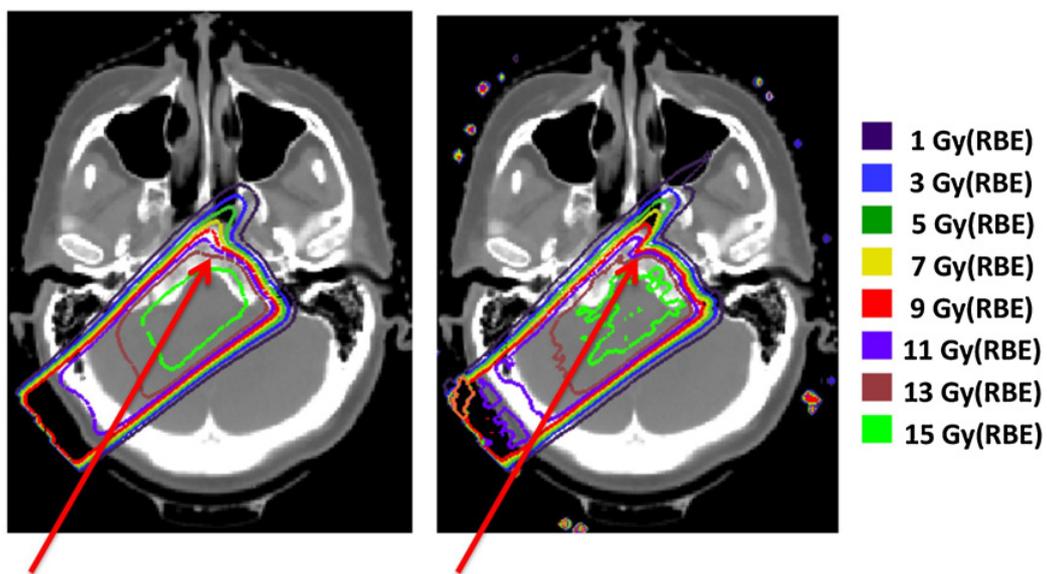


Figure 1.1: Axial view of the expected radiation dose in the brain of a proton therapy patient. On the left are results from a commercially available dose engine (XiO [Computerized Medical Systems]). On the right is the prediction made by a Monte Carlo code that is not commercially available. The red arrow indicates where the discrepancy is largest. It is near the brain stem. *(image reproduced from Paganetti (2012b))*

types of deterministic numerical methods are frequently used to deal with the angular part of the problem: the spherical harmonics expansion and the S_N method. The former is wholly unsuitable for charged particle transport, as the angular particle flux is much too sharply forwardly peaked to be approximated by a truncated series of spherical harmonics. In the S_N method, \mathbb{S}^2 is sampled at N points, called discrete ordinates. Different sets of ordinates and weights have been devised, some of which can deal with anisotropy. (Jarrell, 2010) Unfortunately, there's no simple way to couple adjacent voxels if their S_N -sets are different. From figure 1.1 it should be obvious that certain areas require a different angular discretisation than others.

The next chapters in this thesis describe a fully adaptable deterministic numerical approach. The idea is to use a discontinuous Galerkin method for both the spatial and the angular part of the problem. The resulting numerical scheme can be refined *anisotropically* (i.e.: in Ω -space) or *heterogeneously* (i.e.: in \mathbf{r} -space). It is based on an upwinding principle for the streaming between adjacent voxels. This could form the basis for a dose engine that is both fast and accurate, thereby aiding the treatment planning.

Chapter 2

A DG method for spherical diffusion

In this chapter a numerical scheme for the angular diffusion is derived. For simplicity, consider the time-independent, mono-energetic version of the Fokker-Planck equation in an infinite homogeneous space. That is,

$$\Sigma_a \varphi(\mathbf{\Omega}) - L_{FP} \varphi(\mathbf{\Omega}) = S(\mathbf{\Omega}) , \quad (2.1)$$

where $\mathbf{\Omega}$ lies on the unit sphere and the Fokker-Planck operator is given by

$$L_{FP} = \frac{\alpha}{2} \Delta_s ,$$

where Δ_s is the Laplace-Beltrami operator, given by equation 1.4. The boundary of a sphere is empty, and so the removal term in equation 2.1 is necessary to ensure that there is a solution for all sources S .

Like any finite element method (FEM), the discontinuous Galerkin (DG) and cell-centered Galerkin (ccG) methods for spherical diffusion are set up in two steps. First, the discrete solution vector is mapped to a continuous solution space. This involves a meshing of the physical domain (that is, all solid angles), which is discussed in section 2.2. Section 2.3 explains how one can construct a set of basis functions, each of which has support on a single element in the mesh. These form the basis of the DG method, where each function corresponds to a unique entry in the solution vector, and the solution is a weighted sum of all functions. Section 2.4 details the construction of a ccG solution space. The second step in obtaining a FEM is to find a weak formulation for all functions in the solution space, and to derive a linear system from which the solution vector can be obtained. This will be the subject of section 2.1.

Closely following Pietro (2011), the following notation is used. The dimension of the domain of interest is d . The faces of T are collected in \mathcal{F}_T . \mathcal{T}_F are the elements that boarder a face F . The set of nodes in the mesh is \mathcal{N}_h . For every node $P \in \mathcal{N}_h$, \mathcal{F}_P is the set of all faces that contain P , and \mathcal{T}_P is the set of all elements that contain P . The solution space is V_h . The vector space of degrees of freedom is \mathbb{V}_h .

It shall also be useful to introduce the L^n -norm of a vector with Cartesian components w_i , which is defined as $\|\mathbf{w}\|_n \equiv (\sum_i |w_i|^n)^{1/n}$. The L^2 -norm is denoted by the corresponding non-bold letter, so $w \equiv \|\mathbf{w}\|_2$. Denote by $|\cdot|_d$ the Lebesgue norm on \mathbb{R}^d (that is, the length, area and volume for $d = 1$, $d = 2$ and $d = 3$ respectively).

2.1 Spherical SIPG

By far the most common coordinate systems for a solid angle are spherical, angular or geographical. They reflect the fact that there are only two degrees of freedom in Ω . For the analyses in this thesis, however, it is considerably easier to view a solid angle as a Cartesian coordinate

$$\Omega = [\Omega_1, \Omega_2, \Omega_3]^T \in \mathbb{R}^3$$

subject to the constraint $\|\Omega\|_2 = 1$. The derivative is defined as

$$\frac{\partial}{\partial \Omega} \equiv \sum_{i=1}^3 \mathbf{e}_i \frac{\partial}{\partial \Omega_i},$$

where $\{\mathbf{e}_i\}_{i=1}^3$ are the orthonormalised unit vectors.

Given a solution space V , the weak formulation corresponding to equation 2.1 is

$$\text{Find } u \in V, \text{ such that } b(u, v) = \int_{\mathbb{S}^2} S v \text{ for all } v \in V, \quad (2.2)$$

where $b : (V \times V) \rightarrow \mathbb{R}$ is a bilinear operator that doesn't contain the Laplace-Beltrami operator. It involves only the surface gradient on the sphere, which is defined as a regular Euclidean gradient in \mathbb{R}^3 , minus the component that's normal to the spherical surface. It has the simple Cartesian expression

$$\nabla_s = (I^{3 \times 3} - \Omega \Omega^T) \frac{\partial}{\partial \Omega} \quad (2.3)$$

and satisfies $\Delta_s = \nabla_s \cdot \nabla_s$.

The crucial step in the derivation of a weak form for any elliptic equation is based on integration by parts. The spherical equivalent of this is exactly what one would expect. Let $T \in \mathbb{S}^2$ be an arbitrary spherical surface with outward normal \mathbf{n}_T , tangential to the sphere. For any two scalar functions $v(\boldsymbol{\Omega})$ and $w(\boldsymbol{\Omega})$,

$$\int_T v \Delta_s w = \int_{\partial T} v \mathbf{n}_T \cdot \nabla_s w - \int_T \nabla_s v \cdot \nabla_s w . \quad (2.4)$$

The simplicity is somewhat deceiving, as this is not a direct consequence of integration by parts in a Euclidean space. A similar formula for more general curved surfaces is proved rigorously in Dziuk and Elliott (2013), but it is somewhat technical.

In the particular case of a spherical domain, the symmetry allows for a simpler heuristic derivation. Let $\mathbf{h} : T \mapsto \mathbb{R}^3$ be an arbitrary vector function. Extend its domain to all $\mathbf{q} \in \mathbb{R}^3 \setminus \mathbf{0}$ by letting the function be constant along rays that originate from $\mathbf{0}$. Consider a region

$$U_\epsilon = \{\mathbf{q} \in \mathbb{R}^3 : \mathbf{q}/q \in T \text{ and } 1 - \epsilon < q < 1 + \epsilon\}$$

for some $\epsilon > 0$. U_ϵ is bounded by two surfaces $B^{\pm\epsilon}$ on spheres with radii $1 \pm \epsilon$ and normal vectors $\pm\mathbf{q}/q$, and a surface B^* with a normal vector \mathbf{n}_T that is tangential to \mathbb{S}^2 . Integrate $(1/\epsilon)(\partial/\partial\mathbf{q}) \cdot \mathbf{h}$ over U_ϵ , apply the divergence theorem, let $\epsilon \rightarrow 0$ and note that the contributions of $B^{\pm\epsilon}$ cancel each other out to find

$$\lim_{\epsilon \downarrow 0} \frac{1}{\epsilon} \int_{U_\epsilon} \frac{\partial}{\partial \mathbf{q}} \cdot \mathbf{h} = \lim_{\epsilon \downarrow 0} \frac{1}{\epsilon} \int_{B^*} \mathbf{n}_T \cdot \mathbf{h} .$$

Since U_ϵ has a vanishingly small width 2ϵ in the radial direction, $\mathbf{h}(\mathbf{q})$ can be replaced by $\mathbf{h}(\mathbf{q}/q)$ and thus

$$\frac{1}{\epsilon} 2\epsilon \int_T \frac{\partial}{\partial \mathbf{q}} \cdot \mathbf{h} = \frac{1}{\epsilon} 2\epsilon \int_{\partial T} \mathbf{n}_T \cdot \mathbf{h} .$$

Noting that $(\partial/\partial\mathbf{q}) \cdot \mathbf{h}|_T = \nabla_s \cdot \mathbf{h}$,

$$\int_T \nabla_s \cdot \mathbf{h} = \int_{\partial T} \mathbf{h} \cdot \mathbf{n}_T . \quad (2.5)$$

Some straightforward arithmetic yields

$$(\nabla_s v) \cdot (\nabla_s w) = (I^{3 \times 3} - \boldsymbol{\Omega} \boldsymbol{\Omega}^T) : \left(\frac{\partial v}{\partial \boldsymbol{\Omega}} \left(\frac{\partial w}{\partial \boldsymbol{\Omega}} \right)^T \right)$$

and

$$\nabla_s \cdot (v \nabla_s w) = (I^{3 \times 3} - \boldsymbol{\Omega} \boldsymbol{\Omega}^T) : \left(\left(\frac{\partial v}{\partial \boldsymbol{\Omega}} + v \frac{\partial}{\partial \boldsymbol{\Omega}} \right) \left(\frac{\partial w}{\partial \boldsymbol{\Omega}} \right)^T \right) - 2v \boldsymbol{\Omega} \cdot \frac{\partial w}{\partial \boldsymbol{\Omega}} . \quad (2.6)$$

The Cartesian expression for Δ_s can be found by using the definitions of the spherical coordinates to rewrite equation 1.4, but it's simpler to start with equation 2.3. The result is

$$\Delta_s = (I^{3 \times 3} - \boldsymbol{\Omega} \boldsymbol{\Omega}^T) : \left(\frac{\partial}{\partial \boldsymbol{\Omega}} \left(\frac{\partial}{\partial \boldsymbol{\Omega}} \right)^T \right) - 2 \boldsymbol{\Omega} \cdot \frac{\partial}{\partial \boldsymbol{\Omega}} . \quad (2.7)$$

The last three equations demonstrate explicitly that the product rule holds for the spherical Laplacian:

$$v \Delta_s w = \nabla_s \cdot (v \nabla_s w) - \nabla_s v \cdot \nabla_s w .$$

Combine this with equation 2.5 to arrive at equation 2.4.

The combination of a discontinuous solution space and a spherical Laplace operator suggests a spherical adaptation of an interior penalty method. Due to the similarity between equation 2.4 and its Euclidean counterpart, it is completely analogous to what can be found in standard literature, the only difference being that ∇ is replaced by ∇_s . Derivations of the standard Euclidean case can be found in many places, including Hartmann (2008) and Pietro and Ern (2012), the latter taking a variational approach.

Here only the symmetric interior penalty (SIP) Galerkin method is considered. Let \mathcal{T}_h denote a tessellation of \mathbb{S}^2 with typical length scale h . Possible definitions for h include the diameter of the largest element or face in the mesh. Denote the set of faces in \mathcal{T}_h by \mathcal{F}_h . For every $F \in \mathcal{F}_h$, define a typical length h_F , such as $h_F = \text{diam}(F)$, and choose an arbitrary but fixed ordering of the neighbouring elements T_1 and T_2 . Define a normal vector \mathbf{n}_F of F that points in the direction of T_2 . Define the jump operator and the averaging operator respectively as

$$[[\cdot]]_F \equiv \cdot|_{T_1} - \cdot|_{T_2} \quad \text{and} \quad \{\cdot\}_F \equiv \frac{1}{2} (\cdot|_{T_1} + \cdot|_{T_2}) .$$

On vectors they act component-wise. The SIP bilinear operator is

$$\begin{aligned} b^{\text{SIP}}(u, v) &= \int_{\mathbb{S}^2} \frac{\alpha}{2} \nabla_s u \cdot \nabla_s v - \sum_{F \in \mathcal{F}_h} \int_F \left([[v]] \left\{ \frac{\alpha}{2} \nabla_s u \right\} \cdot \mathbf{n}_F + [[u]] \left\{ \frac{\alpha}{2} \nabla_s v \right\} \cdot \mathbf{n}_F \right) \\ &+ \sum_{F \in \mathcal{F}_h} \int_F \frac{\alpha}{2} \frac{\eta}{h_F} [[u]] [[v]] + \int_{\mathbb{S}^2} \Sigma_a u v , \end{aligned} \quad (2.8)$$

where $\eta > 0$ is the penalty parameter. The numerical solution φ_h is the unique function that satisfies the discrete weak form

$$\text{Find } u \in V_h, \text{ such that } b(u, v) = \int_{\mathbb{S}^2} S v \text{ for all } v \in V_h, \quad (2.9)$$

on an angular mesh \mathcal{T}_h .

Two functionals of the angular flux are of special interest: the scalar flux $\phi \equiv \langle 1, \varphi \rangle$ and the current density

$$\mathbf{j} \equiv \langle \boldsymbol{\Omega}, \varphi \rangle, \quad (2.10)$$

where the inner product is defined as

$$\langle v, w \rangle \equiv \int_{\mathbb{S}^2} v w \quad (2.11)$$

and acts component-wise. A direct consequence of equation 2.4 is that $\langle 1, \Delta_s \varphi \rangle = 0$, so, from equation 2.1, $\phi = (1/\Sigma_a) \langle 1, S \rangle$. This implies that the Fokker-Planck operator preserves the number of particles. A highly desirable property of any discretisation of the Fokker-Planck operator is that this conservative property also holds discretely. This is the case if the DG solution space contains a constant function. Expand φ in equation 2.1 into its eigenfunctions (i.e.: the spherical harmonics), note that $\boldsymbol{\Omega}$ is one of those eigenfunctions, and take the inner product with $\boldsymbol{\Omega}$ to find $\mathbf{j} = \langle \boldsymbol{\Omega}, S \rangle / (\Sigma_a + \alpha)$. The current is conserved if the DG solution space contains all components of $\boldsymbol{\Omega}$.

2.2 The angular mesh

The unit L^n -sphere is the set of all points that satisfy $\|\mathbf{w}\|_n = 1$. The L^2 -sphere that is embedded in \mathbb{R}^d is denoted by \mathbb{S}^{d-1} . If $\mathbf{w} \in \mathbb{R}^3$, then

- the L^1 -sphere is a regular octahedron with vertices at $(\pm 1, 0, 0)$, $(0, \pm 1, 0)$ and $(0, 0, \pm 1)$;
- the L^2 -sphere is \mathbb{S}^2 , which is what is usually meant by a sphere;
- the L^∞ -sphere is a cube with vertices at $(\pm 1, \pm 1, \pm 1)$.

Only the L^2 -sphere is independent of the orientation of the Cartesian axes. It can therefore be used to define the steradian, which is the area of the surface of \mathbb{S}^2 that is subtended by a solid angle.

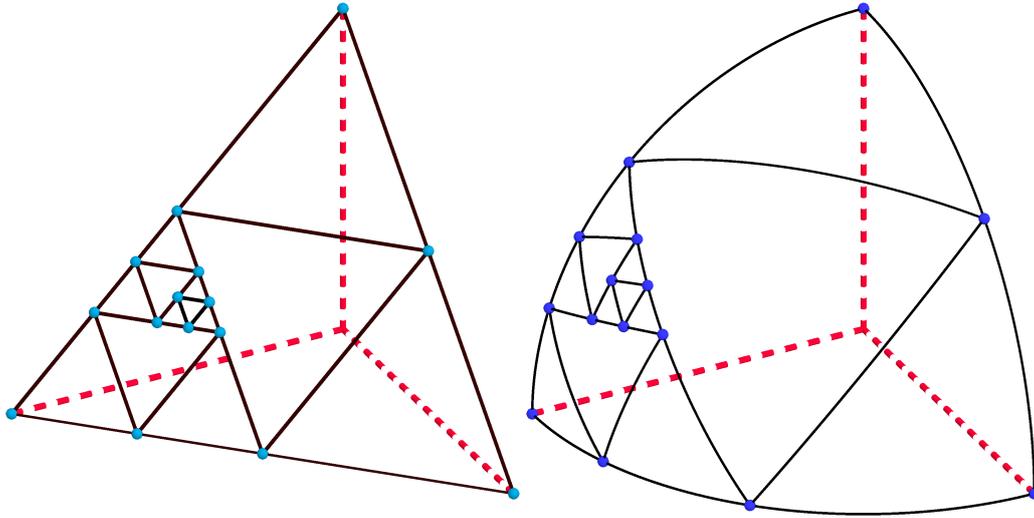


Figure 2.1: Construction of the angular mesh in an arbitrary octant. *Left*: the part of the octahedron (i.e.: the L^1 -sphere) that lies in the octant. The edges are bisected successively to obtain elements of increasing levels. *Right*: the edges and nodes of the L^1 -sphere are projected onto the L^2 -sphere along a line through the origin, to become the boundaries and vertices of the spherical triangles in the angular mesh.

Discretisation of the angular space is equivalent to a tessellation of \mathbb{S}^2 into spherical elements, also called ‘patches’. This is complicated by its non-Euclidean nature. A popular and easy approach is based on the notion that both the L^1 -sphere and the L^∞ -sphere consist of flat surfaces in \mathbb{R}^3 , so that conventional meshing techniques can be applied there. The meshes are subsequently mapped onto \mathbb{S}^2 along straight lines through the origin. This bijective projection is particularly simple due to the fact that

$$\left\| \frac{1}{\|\mathbf{w}\|_n} \mathbf{w} \right\|_n = 1$$

for all vectors \mathbf{w} and all $n > 0$. An important advantage of this class of tessellations is that the meshes are easy to refine locally to any desirable level.

All calculations in this thesis are performed with triangular meshes on the L^1 -sphere that have been mapped onto \mathbb{S}^2 . The resulting patches are spherical triangles. The coarsest mesh consists of the eight octants. Refinement of an element is performed by bisecting the sides of the triangle on the octahedron. This process is illustrated in figure 2.1. The level of an element is the number of times it has been refined. Uniform refinement refers to the case where all

elements are of the same level. This angular discretisation also featured in Jarrell (2010), where it was demonstrated that, for uniform meshes of increasing level, the ratio between the areas of the largest element and the smallest element grows beyond bound.

It may well be that the choice that is made here is suboptimal. For example, one could also refine the elements by bisecting the spherical arcs that form the edge of the spherical triangle. It was shown in Boal et al. (2008) that this yields a quasi-uniform triangulation at all levels. That is, the triangles in a uniform mesh have roughly the same shape and size. An alternative approach would be to map squares or triangles on the L^∞ -sphere onto \mathbb{S}^2 . A great number of other spherical tessellations have been devised, mostly in the field of geophysics. The choice of angular elements also affects the coupling between the elements, and therefore the sparse structure of the global matrix that results from the numerical method. No attempt was made to find the best mesh for charged particle transport.

Though an investigation into this matter would not be without merit, improvements in the accuracy of the calculation are likely to be small for several reasons. FEM in general, and DG methods in particular, tend to work well even on highly irregular meshes. The slight differences between adjacent elements should therefore be of little consequence. Also, the DG method for spherical diffusion was designed with anisotropic refinement in mind. The selection of elements to refine has a much greater impact on the approximating properties of the mesh than the type of tessellation.

If the spatial part of the transport problem is two-dimensional, then the angular flux is symmetric in Ω . Specifically, if $\partial/\partial r_3 = 0$, then $\varphi|_{\Omega_3} = \varphi|_{-\Omega_3}$. One can halve the number of degrees of freedom in the problem by working with a half-sphere $\{\Omega \in \mathbb{S}^2: \Omega_3 > 0\}$ with the homogeneous Neumann boundary condition $\partial\varphi/\partial\Omega_3|_{\Omega_3=0} = 0$. This sort of semi-infinite spatial domains feature much more often in artificial examples, such as those in section 4.4, than in real-life applications, such as dose engines. Two-dimensional models can sometimes preserve qualitatively realistic effects, but the numerical schemes in this thesis were designed specifically for irregular or inhomogeneous spatial meshes, which cannot be reduced to quantitatively accurate problems with two spatial dimensions. An analysis of the use of half-spheres is therefore of limited value, and should not be a driving force in the choice of the spherical mesh and the numerical method.

2.3 The DG solution space: angular basis functions

The solution space of a DG method is the span of the basis functions on all elements. In this thesis, all spherical elements have the same type of basis functions. Their construction should therefore be generic with respect to the size and shape of their element. The basis functions on a spherical element T shall be denoted by $\{f_i\}_{i=1}^p$. It is convenient to represent the basis in the vector form

$$\mathbf{f} = C \mathbf{b} , \quad (2.12)$$

where \mathbf{f} has entries f_i and $C \in \mathbb{R}^{p \times p}$ is a coefficient matrix. The span of the basis is determined by the choice of the functions b_i .

It is easy to conceive of many types of functions on a sphere. In Euclidean geometries the basis functions are typically chosen to be polynomials up to a certain order. Unfortunately, there is no such possibility here, since there are no non-constant linear functions on \mathbb{S}^2 . This is a consequence of the fact that there are no parallel lines in a spherical domain. Two obvious choices for the b_i are studied in this work:

Ω -functions The basis functions are linear in the components of Ω :

$$\mathbf{b} \equiv \begin{bmatrix} 1 \\ \Omega \end{bmatrix} \in \mathbb{R}^4 . \quad (2.13)$$

The spherical gradient is

$$\nabla_s \mathbf{f}^T = (I^{3 \times 3} - \Omega \Omega^T) \frac{\partial \mathbf{b}}{\partial \Omega} C^T . \quad (2.14)$$

octahedron-functions An alternative way to construct the basis functions is to use one of the parameterisations in appendix A.2 that map from a local element \mathcal{K}^{ref} (equation A.6) to an arbitrary T . The basis can be linear locally. That is, given a $\mathbf{k} \in \mathcal{K}^{\text{ref}} \in \mathbb{R}^2$,

$$\mathbf{b} \equiv \begin{bmatrix} 1 \\ \mathbf{k} \end{bmatrix} \in \mathbb{R}^3 . \quad (2.15)$$

In view of its complexity and the non-negligible rounding errors, the angular parameterisation from section A.2.2 is not taken into consideration. That leaves the parameterisations from section A.2.1, where

\mathbf{k} is mapped onto $\boldsymbol{\Omega} \in T$ via an arbitrary intermediate flat triangle \mathcal{Z} with vertices that satisfy equation A.13. The spherical gradient is

$$\begin{aligned} \nabla_s \mathbf{f}^\Gamma &= (I^{3 \times 3} - \boldsymbol{\Omega} \boldsymbol{\Omega}^\top) \frac{\partial \mathbf{f}}{\partial \boldsymbol{\Omega}} \\ &= (I^{3 \times 3} - \boldsymbol{\Omega} \boldsymbol{\Omega}^\top) \frac{\partial \mathbf{z}}{\partial \boldsymbol{\Omega}} \frac{\partial \mathbf{k}}{\partial \mathbf{z}} \frac{\partial \mathbf{b}}{\partial \mathbf{k}} C^\top \\ &= \frac{\partial \mathbf{z}}{\partial \boldsymbol{\Omega}} \frac{\partial \mathbf{k}}{\partial \mathbf{z}} \frac{\partial \mathbf{b}}{\partial \mathbf{k}} C^\top, \end{aligned} \quad (2.16)$$

where the last equality sign follows from equation A.12, and $\partial \mathbf{z} / \partial \boldsymbol{\Omega}$ and $\partial \mathbf{k} / \partial \mathbf{z}$ are given by equations A.11 and A.9 respectively. In the rest of this thesis, all \mathcal{Z} lie on the L^1 -sphere. This is the only choice for which the basis on T can be expressed as a linear combination of the bases on the daughters of T . This is a desirable property when spatial streaming is introduced in chapter 4.

Both types of basis functions have obvious extensions to higher orders. For example, second order functions could be constructed by replacing equation 2.13 or 2.15 by

$$\mathbf{b} \equiv \begin{bmatrix} 1 \\ \boldsymbol{\Omega} \\ \Omega_1 \Omega_1 \\ \Omega_1 \Omega_2 \\ \Omega_1 \Omega_3 \\ \Omega_2 \Omega_2 \\ \Omega_2 \Omega_3 \\ \Omega_3 \Omega_3 \end{bmatrix} \in \mathbb{R}^{10} \quad \text{or} \quad \mathbf{b} \equiv \begin{bmatrix} 1 \\ \mathbf{k} \\ k_1 k_1 \\ k_1 k_2 \\ k_2 k_2 \end{bmatrix} \in \mathbb{R}^6$$

respectively. Both equation 2.14 and 2.16 would remain valid. This option is not further explored here.

2.3.1 Conditioning of the mass matrix and orthonormalisation

In principle, the precise choice of the C in equation 2.12 should not matter, though it should obviously be non-singular in order for the basis functions to be linearly independent. Different matrices will simply lead to different coefficients in the discrete solution vector.

In practice, the linear system resulting from the DG method is not equally easy to solve for all C . Roughly speaking, a better conditioning is generally achieved if no f_i is well approximated by any linear combination of the other basis functions. A more quantitative viewpoint is provided by the mass

matrix $M \in \mathbb{R}^{p \times p}$, defined through its entries $M_{ij} \equiv \langle f_i, f_j \rangle$. Linear solution methods benefit from mass matrices with low condition numbers, as they have to be inverted repeatedly in a linear solver.

Functions that are linear in Ω are discussed in Kópházi and Lathouwers (2015), where it is demonstrated that $C = I^{4 \times 4}$ leads to problematic rounding errors in the case of moderately high refinement. This is to be expected, as there are small patches on which all components of Ω are roughly constant. The proposed solution was to set C such that

$$\begin{aligned} f_i(\mathbf{V}_j) &= \delta_{ij} && \text{for } i = 1, 2, 3, \\ f_4(\mathbf{V}_j) &= 0 && \text{and} \\ f_4\left(\frac{\mathbf{V}_1 + \mathbf{V}_2 + \mathbf{V}_3}{\|\mathbf{V}_1 + \mathbf{V}_2 + \mathbf{V}_3\|_2}\right) &= 1, \end{aligned}$$

where $\{\mathbf{V}_j\}_{j=1}^3$ are the vertices of T . This is a considerable improvement, and yields a workable set of basis functions.

Unfortunately, there is a more fundamental problem that manifests itself in small patches. As the spherical triangles get smaller, they become flat. All four b_i become linear on the triangle, but there can only be three independent linear functions. To ensure a reasonable condition number of M as the elements get smaller, the required coefficients in C can become quite large. Ultimately, this issue is innate in the type of basis functions, and it cannot be resolved by a clever choice of the coefficients. Nevertheless, this notion was found not to be catastrophic for the use the Ω -functions in the numerical tests. (Ref. section 3.4)

There are many reasonable coefficient matrices for functions that are linear on the octahedron. A natural choice would be to let $f_i(\mathbf{V}_j) = \delta_{ij}$ for $i = 1, 2, 3$.

Taking things one step further, the basis functions can be orthonormalised by adjusting C , so that M is the unit matrix. This can be done with a simple stabilized Gram-Schmidt process that is summarized in table 2.1. Orthogonalisation eliminates the cost of inversion for the mass matrices. If the functions are also normalised, the mass matrices don't have to be stored. Some tests were performed to get an indication of the numerical error that is introduced by the orthonormalisation process. The errors are tabulated in 2.2. They are to be compared with the machine epsilon, which is approximately 10^{-16} . The results are encouraging for the basis functions that are linear on the octahedron. For the other functions, orthonormalisation is not recommended. The entries in the coefficient matrix reach values in the order of 10^8 for the smaller elements. The large errors in the mass matrix are a

Table 2.1: A stabilized Gram-Schmidt orthonormalisation process for the set of basis functions $\{f_i\}_{i=1}^p$

1.	$C \leftarrow I^{p \times p}$	# initialise
2.	for $i = 1 \dots p$	
3.	$l \leftarrow \sqrt{\langle f_i, f_i \rangle}$	# length of f_i
4.	for $k = 1 \dots p$	# normalise f_i
5.	$C_{ik} \leftarrow C_{ik}/l$	
6.	end	
7.	for $j = (i + 1) \dots p$	
8.	$\delta \leftarrow \langle f_j, f_i \rangle$	
9.	for $k = 1 \dots p$	# subtract the f_i -component from f_j
10.	$C_{jk} \leftarrow C_{jk} - \delta C_{ik}$	
11.	end	
12.	end	
13.	end	

clear sign of the potentially dubious nature of the basis on highly refined meshes.

2.4 The ccG solution space

It is not clear what the best type of basis functions are for a DG method on \mathbb{S}^2 . Unfortunately, a piecewise constant basis is not an option, as the SIP method from section 2.1 would break down. The non-constant functions are needed to evaluate the derivatives. A finite difference scheme would only provide an approximation at certain points, whilst handling the spatial streaming requires a value for the flux at all $\Omega \in \mathbb{S}^2$. Finite volume methods require an estimation of the gradient on the boundary of the elements. There is no obvious way to do this on the highly non-homogeneous spherical meshes that are necessary for forwardly peaked charged particle transport.

For these reasons, this section explores the possibility of an alternative lowest-order method in the form of a cell centred Galerkin (ccG) method. The unknowns are the values of the flux at the centres of the cells, which are used to estimate both φ and $\nabla_s \varphi$. The numerical solution can be discontinuous at the cell boundaries. It is still based on the SIP bilinear form for discontinuous basis functions, differing only from the DG method in the construction of the solution space from the unknowns.

Since the ccG method is fairly new and not widely used, this section will first give a brief overview how the solution space is constructed in a Euclidean

Table 2.2: Maximum absolute deviation of the mass matrix from the unit matrix on any of the elements after Gram-Schmidt orthonormalisation

refinement level	$\max_T \max_{i,j} M_{ij} - \delta_{ij} $ linear in Ω	linear on L^1 -sphere
0	$2 \cdot 10^{-15}$	$1 \cdot 10^{-16}$
1	$4 \cdot 10^{-15}$	$2 \cdot 10^{-17}$
2	$2 \cdot 10^{-14}$	$2 \cdot 10^{-16}$
3	$1 \cdot 10^{-13}$	$3 \cdot 10^{-16}$
4	$6 \cdot 10^{-13}$	$5 \cdot 10^{-16}$
5	$3 \cdot 10^{-12}$	$1 \cdot 10^{-15}$
6	$5 \cdot 10^{-11}$	$3 \cdot 10^{-15}$
7	$2 \cdot 10^{-10}$	$4 \cdot 10^{-15}$

space of dimension d . The presentation here draws heavily on Pietro (2011), which treats a more general case with anisotropic diffusion and the inclusion of boundary conditions. More details can also be found in Pietro (2010) and Pietro and Ern (2012).

Let $v_h = v_h(\mathbf{x}) \in V_h$ be the numerical approximation of v on a Euclidean mesh \mathcal{T}_h . For the exact constraints on the mesh, see Pietro (2011). The angular mesh described in section 2.2 meets these requirements. The unknowns in the ccG method are the values of v_h in the cell centres \mathbf{x}_T of all the elements $T \in \mathcal{T}_h$. The numerical solution is piecewise linear in \mathbf{x} . On an element T ,

$$v_h(\mathbf{x} \in T) = v_T + \mathbf{G}_T \cdot (\mathbf{x} - \mathbf{x}_T) \quad , \quad (2.17)$$

where \mathbf{x}_T is the centre of an element T and $v_T \equiv v(\mathbf{x}_T)$.

The gradient \mathbf{G}_T is constructed in two steps. First, an approximation of v on every face of the mesh is obtained from the v_T . This is done with the so-called L-construction, originally introduced in the context of finite volume methods as a robust multi-point flux approximation in Aavatsmark et al. (2008). A set $g \subset \mathcal{F}_h$ is called an L-group if

- it contains d faces;
- all faces border a common element;
- all faces share a node that borders this element.

More concisely, the set of all L-groups in the mesh is given by

$$\mathcal{G} \equiv \{g \subset \mathcal{F}_T \cap \mathcal{F}_P, T \in \mathcal{T}_P, P \in \mathcal{N}_h : \text{card}(g) = d\} .$$

An example on a two-dimensional triangular mesh is given in figure 2.2. For every possible L-group $g \in \mathcal{G}$,

- select a primary element T_g , such that $g \subset \mathcal{F}_{T_g}$ (that is, all faces in g border T_g);
- let \mathcal{T}_g be the set of all elements that border a face in g :

$$\mathcal{T}_g \equiv \{T \in \mathcal{T}_h : T \in \mathcal{T}_F, F \in g\} .$$

There are always $d + 1$ such elements;

- define the matrix

$$A_g \equiv \left[\left[\frac{1}{\text{dist}(\mathbf{x}_T, F)} (\mathbf{x}_T - \mathbf{x}_{T_g}) \right]_{g \ni F \subset T_g \cap T}^T \right] \in \mathbb{R}^{d \times d} , \quad (2.18)$$

where

$$\text{dist}(\mathbf{a}, F) = \min_{\mathbf{b} \in F} \|\mathbf{a} - \mathbf{b}\|_2$$

is the distance between F and \mathbf{a} ;

- let \mathcal{G}_F be the set of all L-groups that contain the face F .

For all faces $F \in \mathcal{F}_h$, select a unique L-group $g_F \in \mathcal{G}_F$. It is shown in Agélas et al. (2010) and Pietro (2010) that the best approximating properties are achieved by choosing

$$g_F = \underset{g \in \mathcal{G}_F}{\text{argmin}} \|A_g^{-1}\|_2 . \quad (2.19)$$

For an L-group g , denote by $\xi^g(\mathbf{x})$ the unique function that is linear everywhere on \mathcal{T}_h and that satisfies $\xi^g(\mathbf{x}_T) = v_T$ for all $T \in \mathcal{T}_g$. It is an interpolating function that can be used to estimate an average value of v on F . Specifically, let

$$v_F \equiv \frac{1}{|F|_{d-1}} \int_F \xi^{g_F} \approx \frac{1}{|F|_{d-1}} \int_F v . \quad (2.20)$$

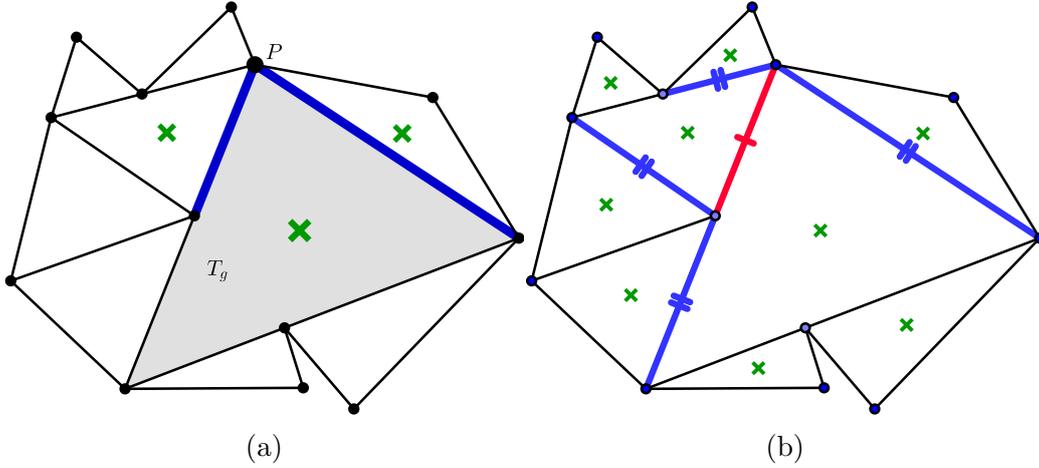


Figure 2.2: *Left:* The two bold, blue faces form an L-group g in a two-dimensional triangular mesh. \mathcal{T}_g is the set of elements whose centre of mass is indicated with a cross. All faces in g border the primary element T_g and a common node P . *Right:* The red face that's indicated with a single dash (\dagger) can form an L-group with any one of the blue faces that are indicated with a double dash (\ddagger).

The ccG method depends critically on the convergence of the v_F . That is,

$$\lim_{h \rightarrow 0} v_F = \frac{1}{|F|_{d-1}} \int_F v .$$

Note how v_F is linear in the unknowns (i.e.: the v_T). The second step in obtaining \mathbf{G}_T is to use the v_F in a gradient reconstruction operator that is reminiscent of Green's formula:

$$\mathbf{G}_T = \frac{1}{|T|_d} \sum_{F \in \mathcal{F}_T} |F|_{d-1} (v_F - v_T) \mathbf{n}_{T,F} . \quad (2.21)$$

It should be noted that the ccG method does not display the conservative properties of DG methods, even though constant functions are part of the solution space. This is because the $\{v_F\}_{F \in \mathcal{F}_h}$ are interpolated values. An equivalent phenomenon can occur in finite volume methods on irregular grids, where the gradient on the surface of an element cannot be determined exactly from the unknowns. If the $\{v_F\}_{F \in \mathcal{F}_h}$ were treated as unknowns, to be determined by solving the global linear system, then the scalar flux would be conserved, as demonstrated in Pietro (2013)

2.4.1 Adjustments to the spherical space

An adaptation of a ccG method on a spherical mesh requires some modifications. Equation 2.17 has an obvious spherical equivalent in

$$\phi_h(\boldsymbol{\Omega} \in T) = \phi_T + \mathbf{G}_T \cdot (\boldsymbol{\Omega} - \boldsymbol{\Omega}_T) , \quad (2.22)$$

where the centre of an element is defined as

$$\boldsymbol{\Omega}_T \equiv \frac{1}{\|\int_T \boldsymbol{\Omega}\|_2} \int_T \boldsymbol{\Omega} , \quad (2.23)$$

As the angular mesh is refined, the spherical elements become flat. As $h \rightarrow 0$, a linearisation of ϕ_h in the components of $\boldsymbol{\Omega}$ should be equivalent to the Euclidean ccG method. This could also be achieved by letting $\boldsymbol{\Omega}_T \equiv \int_T \boldsymbol{\Omega}/|T|_d$, though this would mean that $\boldsymbol{\Omega}_T \notin T$. Both choices are valid, but equation 2.23 is preferred, because it consistently yielded slightly better approximation properties in the numerical test cases, which featured meshes that were refined uniformly, randomly or in a targeted manner.

A similar multitude of options crops up in the choice of the best L-group that is associated with a face. A naive application of equation 2.19 with A_g from equation 2.18 would result in a 2×3 matrix that cannot be inverted. Instead, the L-groups are chosen by projecting all $\boldsymbol{\Omega} \in \mathbb{S}^2$ perpendicularly onto a flat plane $K \in \mathbb{R}^3$, and $\|A_g^{-1}\|_2$ is computed in the Euclidean manner on this plane. K is placed such that it intersects all centres of the elements in \mathcal{T}_g . This provided slightly better approximating behaviour than the other choices that were tested, which included

- placing K such that it intersects the vertices of F and that it is perpendicular to the plane that contains F ;
- projecting $\boldsymbol{\Omega} \in \mathbb{S}^2$ onto K along a straight line through the origin;
- choosing $g_F \in \mathcal{G}_F$ such that $\max_{T \in \mathcal{T}_g} \|\boldsymbol{\Omega} - \boldsymbol{\Omega}_T\|_2$ is minimised. That is, the elements that boarder the L-group are as close to F as possible.

Chapter 3

Numerical results of spherical SIP

This chapter backs up and clarifies the theory from the previous chapter with several numerical examples. The numerical solutions satisfy the discrete weak form 2.9 with the SIP bilinear operator given by 2.8 with $h_F = \text{diam}(F)$ and $\eta = 3$, unless stated otherwise. The computations were performed on several (anisotropic) angular meshes with both types of basis functions from section 2.3 and the ccG solution space from section 2.4. The accuracy could be determined by studying artificial analytical solutions to equation 2.1. The weak form 2.2 is consistent. Sections 3.2 to 3.5 deal with basic properties of the symmetric interior penalty method from chapter 2.

3.1 Implementation

DG methods have been implemented before on non-trivial meshes for many different applications. The same techniques can be used here. The standard approach is to map all basis functions to a degree of freedom (DOF), which is an index in the solution vector $\varphi_h \in \mathbb{R}^N$. Let f_i^T denote the i 'th basis function on element T , and define a bijection $D: (T, i) \leftrightarrow \{1, \dots, N\}$ that maps every degree of freedom to a unique positive integer. D is called the DOF handler. The approximate flux φ_h is expressed in terms of φ_h as

$$\varphi_h(\Omega) = \sum_{T \in \mathcal{T}_h} \sum_i f_i^T(\Omega) [\varphi_h]_{D(T,i)}. \quad (3.1)$$

Forcing

$$b^{\text{SIP}}(f_i^{T_a}, f_j^{T_w}) = \int_{\mathbb{S}^2} S f_j^{T_w} \quad (3.2)$$

to hold for all j and all $T_w \in \mathcal{T}_h$ yields row number $D(T_q, i)$ in the global linear system

$$A_h \boldsymbol{\varphi}_h = \mathbf{S}_h, \quad (3.3)$$

where A_h is the global matrix and \mathbf{S}_h is a forcing term that depends on the source.

The simple stencil in a DG method allows a straightforward construction of the global matrix. A basis function f_i^T is coupled to all functions that have support on either T or its direct neighbours. The linear system can thus be constructed by splitting the weak form into a volumetric and a surface part, looping over all elements and faces, and inserting the local linear system 3.2 into the global system 3.3 at the indices indicated by the degree of freedom handler. This can also be done in a matrix-free manner. That is, the elements of A_h are evaluated at run-time, so that the matrix does not have to be stored explicitly. See Baggag et al. (1999) for details and a comparison of parallel implementations.

The implementation of the ccG method requires a different approach. Classical FEM implementations are based on local element matrices that can be computed without looking up the global degrees of freedom. This concept needs to be abandoned, because the stencil can be non-local and varies from element to element. (See figure 3.1.) The alternative implementation described here is inspired by Pietro and Gratien (2011). A vector quantity $\mathbf{a} \in \mathbb{R}^m$ that depends linearly on N_a unknowns is expressed in terms of a so-called ‘linear combination’ $\{\tau_{\mathbf{a}}, \mathbb{A}\}$, where $\tau_{\mathbf{a}} \in \mathbb{R}^{m \times N_a}$ and $\mathbb{A} \in \mathbb{N}^{N_a}$, such that

$$\mathbf{a} = \tau_{\mathbf{a}} \boldsymbol{\varphi}_h(\mathbb{A}) .$$

\mathbb{A} can be considered a list of global degrees of freedom. The matrix $\tau_{\mathbf{a}}$ indicates how \mathbf{a} depends on the degrees of freedom in \mathbb{A} . A scalar quantity corresponds simply to the case $m = 1$. If $\{\tau_{\mathbf{b}}, \mathbb{B}\}$ represents $\mathbf{b} \in \mathbb{R}^m$, then the scalar product $c = \mathbf{a} \cdot \mathbf{b}$ is represented by $\{\tau_c, (\mathbb{A}, \mathbb{B})\}$ where $\tau_c = (\tau_{\mathbf{a}})^T \tau_{\mathbf{b}}$, so that

$$c = \tau_c : (\boldsymbol{\varphi}_h(\mathbb{A}) \boldsymbol{\varphi}_h^T(\mathbb{B})) .$$

These linear combinations provide a way to express a quantity in terms of $\boldsymbol{\varphi}_h$ when $\boldsymbol{\varphi}_h$ is not yet known.

In the construction of the global matrix, the first step is to associate an L-group to every face F in the mesh, and express the v_F in equation 2.20 in terms of a linear combination. Subsequently, the linear combinations of

the gradients \mathbf{G}_T on all elements T in the mesh are computed according to equation 2.21. Both φ_h and $\nabla_s \varphi_h$ can now be expressed in terms of a linear combination for any arbitrary Ω . The bilinear terms in 2.8 can then be inserted into the global matrix by looping over all elements and all faces. For example, the penalty term on a particular face could have the linear combination $\{A_{\text{loc}}, (\mathbb{I}, \mathbb{J})\}$, which can then be inserted into the global matrix at the appropriate indices by

$$A(\mathbb{I}, \mathbb{J}) \leftarrow A(\mathbb{I}, \mathbb{J}) + A_{\text{loc}} .$$

The source vector \mathbf{S}_h is constructed in a similar manner by looping over all elements.

3.2 Order of convergence to analytical solution for DG

To study the rate of convergence of the spherical SIPG method, consider a single example. Results were similar for other tests. The simplest non-trivial smooth solutions are the eigenfunctions

$$\begin{aligned} \varphi(\Omega) &= Y_{lm}(\Omega) && (\text{s}^{-1}\text{cm}^{-2}) \\ S(\Omega) &= \left(\Sigma_a + \frac{\alpha}{2} l(l+1) \right) \varphi(\Omega) && (\text{s}^{-1}\text{cm}^{-3}), \end{aligned}$$

where Y_{lm} are the real spherical harmonics, explicit expressions of which can be found in many places. The normalisation is such that they have unit energy, so that $\langle Y_{lm}, Y_{l'm'} \rangle = \delta_{ll'} \delta_{mm'}$. Y_{lm} is a polynomial of order l in the components of Ω . Specifically, the source was chosen in such a way that the analytical solution is

$$\begin{aligned} \varphi &= Y_{00} - Y_{21} + Y_{30} && (\text{s}^{-1}\text{cm}^{-2}) \quad (3.4) \\ &= \frac{1}{\sqrt{4\pi}} \left(1 - 15\Omega_1\Omega_2 + \frac{1}{2}\sqrt{7}(\Omega_1(2\Omega_1^2 - 3\Omega_2 - 3\Omega_3^2)) \right) && (\text{s}^{-1}\text{cm}^{-2}) \end{aligned}$$

with $\Sigma_a = 0.10 \text{ cm}^{-1}$ and $\alpha = 0.25 \text{ cm}^{-1}$. The exact Cartesian expression of the spherical harmonics can vary with different authors. The mesh was uniformly refined at various levels. The global error is defined as

$$e \equiv \sqrt{\frac{\langle \varphi_h - \varphi, \varphi_h - \varphi \rangle}{\langle \varphi, \varphi \rangle}} . \quad (3.5)$$

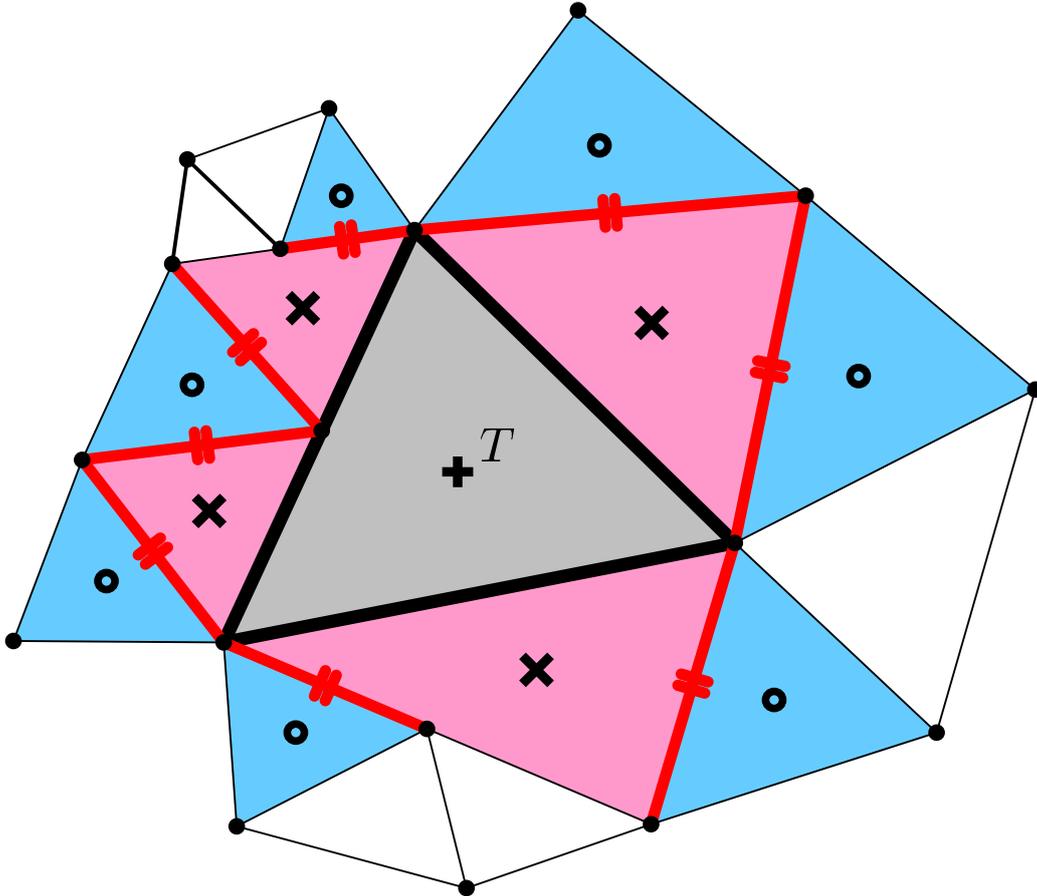


Figure 3.1: The stencil in the ccG method on an unstructured mesh depends on the choice of the L-groups. Consider an arbitrary element T , indicated with '+'. The gradient \mathbf{G}_T on T (equation 2.22) depends on the $\{v_F\}_{F \in \mathcal{F}_T}$ in equation 2.20, which in turn depends on the choice of which L-group to associate with the faces of T . The neighbours of T (red; indicated with \times) are always part of the stencil of T . The red bold faces (indicated with \ddagger) could form an L-group with one of the faces of T . The blue elements (indicated with \circ) border those faces, and could therefore be part of the stencil of T , but are not necessarily so.

Table 3.1: Results of the spherical SIPG method on uniformly refined meshes \mathcal{T}_h with the exact solution 3.4. The error e is given by 3.5. The order of convergence on \mathcal{T}_h is estimated with equation 3.6 and a comparison with the result on \mathcal{T}_{2h} .

The last column lists the order of convergence of a component of the current.

card(\mathcal{T}_h)	<i>basis functions linear in Ω</i>		<i>basis functions linear on octahedron</i>		
	e	order of e	e	order of e	order of $ \langle \Omega_3, \varphi_h \rangle - j_3 $
$8 \cdot 4^0$	$2.331 \cdot 10^{-1}$	-	$2.901 \cdot 10^{-1}$	-	-
$8 \cdot 4^1$	$8.238 \cdot 10^{-2}$	1.50	$1.271 \cdot 10^{-1}$	1.19	5.60
$8 \cdot 4^2$	$2.268 \cdot 10^{-2}$	1.86	$3.890 \cdot 10^{-2}$	1.71	1.30
$8 \cdot 4^3$	$5.944 \cdot 10^{-3}$	1.93	$1.131 \cdot 10^{-2}$	1.78	1.65
$8 \cdot 4^4$	$1.503 \cdot 10^{-3}$	1.98	$2.998 \cdot 10^{-3}$	1.92	2.06
$8 \cdot 4^5$	$3.768 \cdot 10^{-4}$	2.00	$7.638 \cdot 10^{-4}$	1.97	2.06

The results can be found in table 3.1.

Any global quantity $\mathcal{E} = \mathcal{E}(\mathcal{T}_h)$ on a mesh \mathcal{T}_h is said to converge with order n if \mathcal{E}/h^n converges to a nonzero constant as $h \rightarrow 0^+$. In a mesh with d degrees of freedom, h is proportional to $\text{card}(\mathcal{T}_h)^{-1/d}$. If \mathcal{E} is known on two tessellations \mathcal{T}_{h_1} and \mathcal{T}_{h_2} , then the order of convergence can be estimated with

$$n = d \frac{\log(\mathcal{E}(\mathcal{T}_{h_1}) / \mathcal{E}(\mathcal{T}_{h_2}))}{\log(\text{card}(\mathcal{T}_{h_2}) / \text{card}(\mathcal{T}_{h_1}))}, \quad (3.6)$$

where $d = 2$ on a spherical surface. The global error defined by 3.5 behaves as $e \propto h^2$ on highly refined meshes. This was to be expected, as the spherical elements become flat, and the basis functions become linear. In the conventional Euclidean SIPG method, e converges with order $p+1$ if the basis functions are polynomials up to degree p . Due to its physical meaning, the error in the estimate of the angular current density (equation 2.10) deserves special attention. The quantity is conserved for Ω -functions. The error appears to converge roughly with order 2 for the octahedron-functions.

The convergence is slower on coarser meshes for two reasons. First, the solution is insufficiently smooth within the elements. This effect is present in all numerical schemes that require a tessellation of a physical domain. Second, the basis functions are approximately linear only on small elements. The octahedron-functions in particular have a highly irregular shape on large elements, ref. section A.2.3 and figure A.4.

3.3 Anisotropic refinement

One of the strong points of DG methods in general is its performance on irregular meshes. The spherical SIPG method from the previous chapter is well-suited for anisotropic refinement. This is essential for charged particle transport, where the angular flux is strongly forwardly peaked. For illustration, let the source be such that the flux be

$$\begin{aligned} \varphi &= \sum_{j=0}^{14} c_j (\Omega_1)^j && (\text{s}^{-1}\text{cm}^{-2}) && (3.7) \\ c_0 &= 1 && (\text{s}^{-1}\text{cm}^{-2}) \\ c_1 &= -1 && (\text{s}^{-1}\text{cm}^{-2}) \\ c_j &= \frac{(j-2)!}{j!} && (\text{s}^{-1}\text{cm}^{-2}) && \forall j \geq 2, \end{aligned}$$

which is a Taylor expansion of

$$\varphi \approx 1 + (1 - \Omega_1) \log(1 - \Omega_1), \quad (\text{s}^{-1}\text{cm}^{-2})$$

thus ensuring that

$$\frac{\partial^2}{\partial \Omega_1^2} \varphi \approx \frac{1}{1 - \Omega_1}. \quad (\text{s}^{-1}\text{cm}^{-2})$$

which means that the flux has the greatest non-linearity in Ω near $\Omega_1 = 1$, whilst it is roughly linear near $\Omega_1 = -1$. The flux is plotted in figure 3.2. Figure 3.3a displays the error of a numerical simulation with Ω -functions on a mesh where all elements are of the same level. The error is high near $\Omega_1 = 1$, but comparably negligible on the half-sphere $\Omega_1 < 0$. The uniform refinement has obviously resulted in a suboptimal use of the computational memory and work. Figure 3.3b is an improvement. The spherical triangles near the pole $\Omega_1 = -1$ are merged into larger elements. The mesh is more refined near the opposite pole. The error is more uniformly distributed.

3.4 Conditioning of the global matrix

The computational cost depends strongly on the condition number of A_h , which is defined as the ratio between its largest and the smallest eigenvalue. It was evaluated with MatLab's internal routine. Table 3.2 collects the results for various uniformly refined angular meshes for both types of basis functions. The condition number behaves as $\mathcal{O}(h^{-2})$, which is similar to what one would

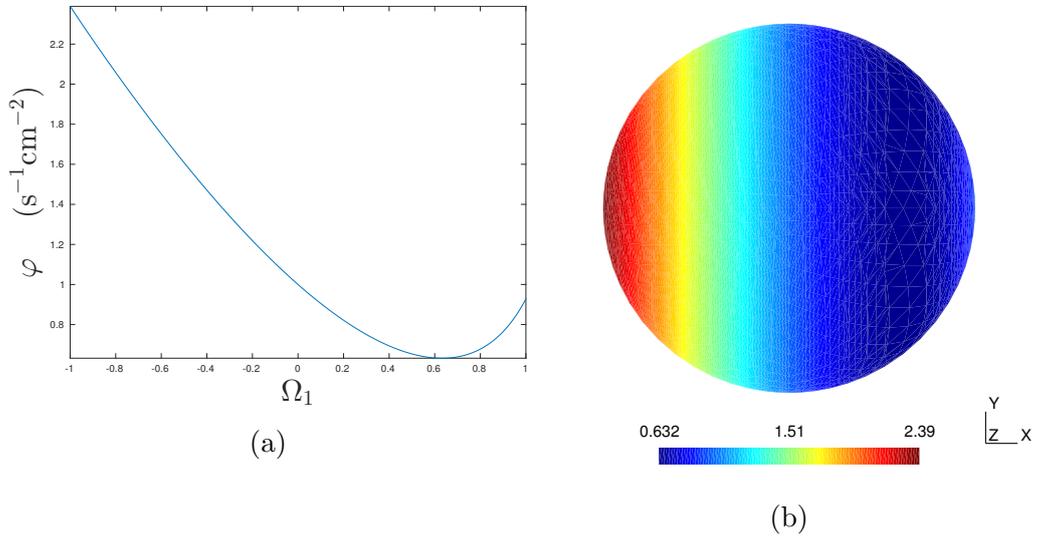
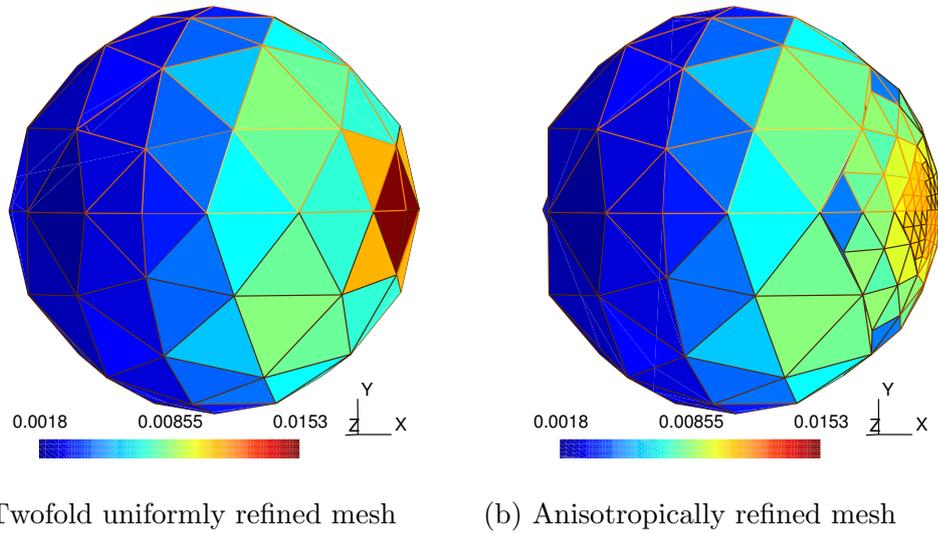


Figure 3.2: Plots of φ from equation 3.7. The image on the right is a ‘heat map’ representation on \mathbb{S}^2 .



(a) Twofold uniformly refined mesh (b) Anisotropically refined mesh

Figure 3.3: Root mean square of the deviation from the exact solution on the elements of the angular mesh. The source is given by equations 2.1 and 3.7. The basis is constructed with the Ω -functions.

Table 3.2: $\text{cond}(A_h)$ in the 2-norm on various uniformly refined meshes for $\alpha/\Sigma_a = 5/2$. The basis functions are constructed as described in sections 2.3 and 2.3.1. They are not orthonormalised. The number of degrees of freedom is $N = 4 \cdot \text{card}(\mathcal{T}_h)$ for Ω -functions and $N = 3 \cdot \text{card}(\mathcal{T}_h)$ for octahedron-functions.

The order of convergence is based on a comparison with the result on a mesh that's one level less refined. It is calculated with equation 3.6, where \mathcal{E} is replaced by the condition number. The negative numbers indicate a divergence as $h \rightarrow 0^+$.

$\text{card}(\mathcal{T}_h)$	<i>basis functions linear in Ω</i>		<i>basis functions linear on octahedron</i>	
	$\text{cond}(A_h)$	order	$\text{cond}(A_h)$	order
$8 \cdot 4^0$	$4.6 \cdot 10^1$	-	$2.2 \cdot 10^1$	-
$8 \cdot 4^1$	$2.0 \cdot 10^2$	-2.09	$8.9 \cdot 10^1$	-1.98
$8 \cdot 4^2$	$7.7 \cdot 10^2$	-1.97	$3.5 \cdot 10^2$	-1.97
$8 \cdot 4^3$	$3.1 \cdot 10^3$	-1.99	$1.4 \cdot 10^3$	-1.99
$8 \cdot 4^4$	$1.2 \cdot 10^4$	-2.00	$5.5 \cdot 10^3$	-2.00
$8 \cdot 4^5$	$4.9 \cdot 10^4$	-2.00	$2.4 \cdot 10^4$	-2.00

expect for a diffusive problem in a Euclidean space. These results can also be seen as a confirmation of the validity of the construction of the Ω -functions, as described in section 2.3.1.

With other factors constant, the condition number varies with the material properties as a function of α/Σ_a . This follows from equation 2.8, as can be verified either by a dimensionless analysis or by noting that, for any $\lambda \in \mathbb{R} \setminus 0$, $\text{cond}(\lambda A_h) = \text{cond}(A_h)$. Figure 3.4 displays numerical results on two uniform meshes of different refinement levels. If $\alpha = 0$, then there is no coupling between the fluxes in adjacent elements, and the matrix is block diagonal. The resulting global matrix is well-conditioned. If $\Sigma_a = 0$, then the continuous problem 2.1 is ill-defined for all sources except those for which $\int_{\mathbb{S}^2} S = 0$. The corresponding discrete Fokker-Planck operator A_h is singular.

3.5 Numerical results of ccG SIPG

The spherical ccG method was tested on several meshes with different sources. Since the necessary amount of work and memory is roughly proportional to the number of degrees of freedom, a comparison with DG methods is possible. The ccG consistently outperforms the DG methods. That is, the error (equation 3.5) per degree of freedom is higher in DG methods. This is prob-

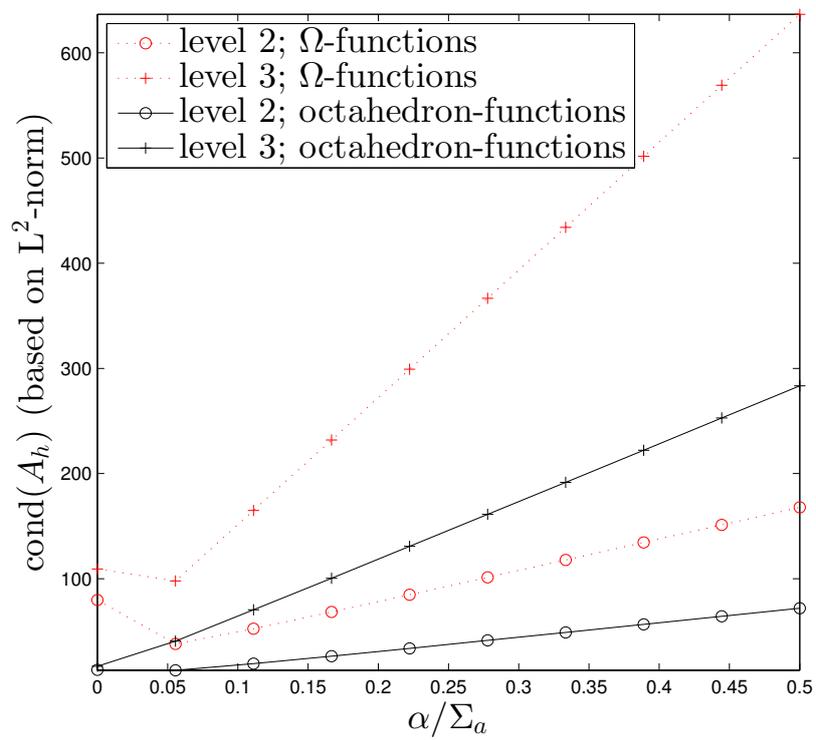


Figure 3.4: Conditioning of A_h as a function of the material properties on uniformly refined angular meshes of various levels and for both types of basis functions

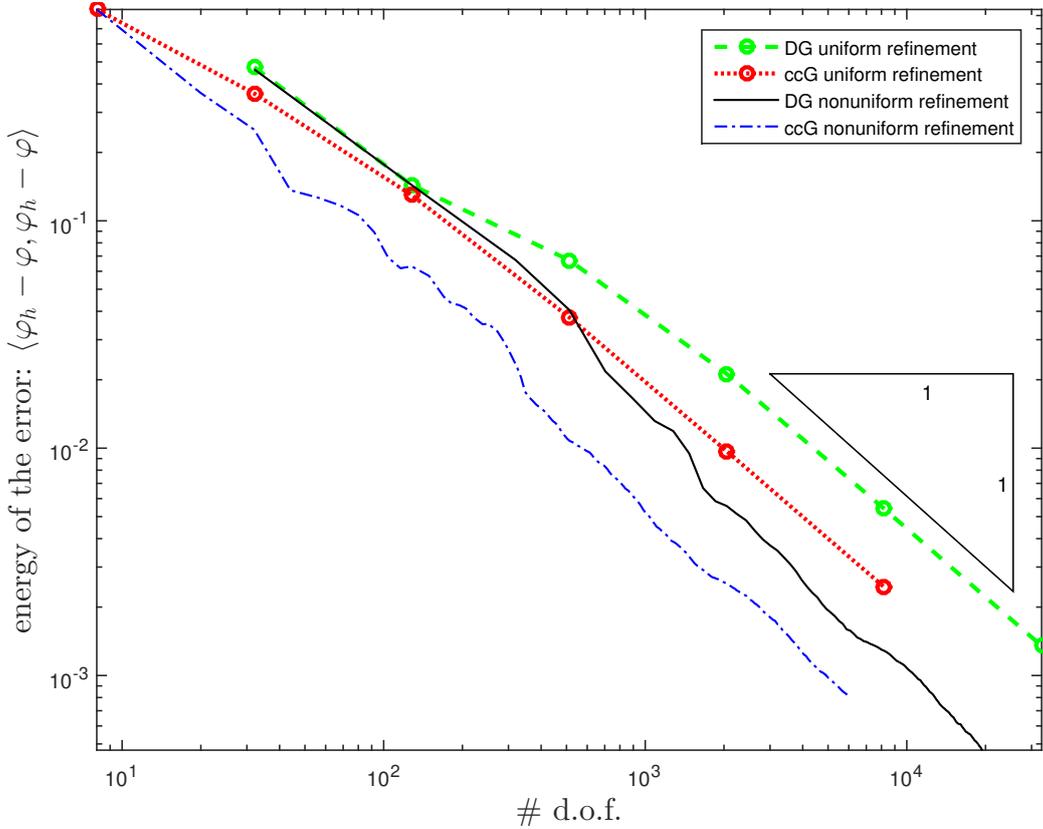


Figure 3.5: Comparison of the errors in ccG and DG methods on various meshes, compared to the number of degrees of freedom that is required to do the calculation. The flux is given by equation 3.8.

ably due to the fact that every unknown in a ccG method is used to estimate both the flux at a point, and the gradient of the flux at the adjacent faces. By contrast, the DG method needs at least three basis functions on an element to estimate the average value and the gradient in two directions.

All this is best illustrated with a single example. Consider the flux

$$\varphi = \frac{1}{4\pi(1-s)} (1 + \tanh(a(\Omega_1 - s))) \quad (\text{s}^{-1}\text{cm}^{-2}). \quad (3.8)$$

If $a \gg 1$, then the flux has a sharp gradient at $\Omega_1 = s$ and the scalar flux is $\phi \approx 1 \text{ s}^{-1}\text{cm}^{-2}$. The computations were performed with $a = 7$, $s = 1/2$, $\alpha = 2 \text{ cm}^{-1}$ and $\sigma_a = 3 \text{ cm}^{-1}$. The results are depicted in figure 3.5. The same figure also illustrates how local refinement is more effective than uniform refinement.

The selection of the elements to refine locally was based on prior knowledge of the solution. After each calculation, the elements that contributed

most to the global error were refined, and the computation was performed again. This is a form of cheating, of course, and in future work it should be done automatically during runtime without knowledge of the exact errors. Nevertheless, the results demonstrate that both the DG and the ccG method perform well on anisotropic meshes. This is an especially important point for the ccG method, because solution space is fairly ‘stiff’. That is, if an entry in the solution vector is perturbed, the resulting perturbation in the numerical solution is not very local. This is due to its non-local stencil.

Chapter 4

A finite element method for Fokker-Planck particle transport

This chapter introduces spatial streaming. The particle flux $\varphi = \varphi(\mathbf{r}, \boldsymbol{\Omega})$ is governed by the time-independent mono-energetic Boltzmann equation in the Fokker-Planck limit in an inhomogeneous space:

$$L_{\text{str}}\varphi + \Sigma_a\varphi - \frac{\alpha}{2}\Delta_s\varphi = S \quad \text{for } \mathbf{r} \in H ; \quad (4.1)$$

$$\varphi = \varphi^D \quad \text{for } \mathbf{r} \in \partial H, \boldsymbol{\Omega} \cdot \mathbf{n}_H < 0 , \quad (4.2)$$

where \mathbf{n}_H is the outward normal of the spatial domain H , and $\boldsymbol{\Omega} \in \mathbb{S}^2$ is the unit direction vector. Most materials of interest, such as biological tissue, are isotropic, which implies that the material properties $\Sigma_a = \Sigma_a(\mathbf{r})$ and $\alpha = \alpha(\mathbf{r})$ do not depend on the direction vector. Both the source and the Dirichlet boundary condition can be anisotropic. That is, $S = S(\mathbf{r}, \boldsymbol{\Omega})$ and $\varphi^D = \varphi^D(\mathbf{r}, \boldsymbol{\Omega})$. The streaming operator is given by

$$L_{\text{str}} \equiv \boldsymbol{\Omega} \cdot \frac{\partial}{\partial \mathbf{r}} .$$

Section 4.1 lays the theoretical foundation of the Riemann upwinding method by deriving the weak form. Section 4.2 explains how the resulting linear system can be solved. Section 4.3 studies the convergence of the numerical solution for different discretisations. A final example of a diffusing pencil beam is given in section 4.4.

4.1 The discrete setting: Riemann upwinding

The discretisations in space and angle are both based on a DG method. Let \mathcal{R} be a mesh of H . All spatial elements $R \in \mathcal{R}$ are assumed to be flat, so that normal vector of a spatial face is independent of the position. Each is assigned its own angular mesh $\mathcal{P}_{[R]}$. Let the components of $\mathbf{g}_{[R]}$ be basis functions on R . To avoid confusion, angular elements are hereinafter referred to as ‘patches’, and denoted with the letter P . The basis functions on P are the components of $\mathbf{f}_{[P]}$. The solution space is a product of the spatial and the angular parts. That is,

$$\varphi(\mathbf{r}, \Omega) \approx \sum_{R \in \mathcal{R}} \boldsymbol{\rho}_{[R]} \cdot \mathbf{g}_{[R]}(\mathbf{r}) \sum_{P \in \mathcal{P}_{[R]}} \boldsymbol{\pi}_{[P]} \cdot \mathbf{f}_{[P]}(\Omega) , \quad (4.3)$$

where the $\boldsymbol{\rho}_{[R]}$ and $\boldsymbol{\pi}_{[P]}$ are solution coefficients. This is simply a convenient notation; in a practical calculation all solution coefficients would be collected in a single solution vector.

To avoid excessive use of indices, the following notation is used. For any quantity \square that’s defined on an element, $\square_{[R]}$ and $\square_{[\tilde{R}]}$ are abbreviated by \square and $\tilde{\square}$ respectively. Similarly, $\diamond \equiv \diamond_{[P]}$ and $\tilde{\diamond} \equiv \diamond_{[\tilde{P}]}$ for any quantity \diamond that depends on the patch. For example, \mathcal{P}_h and $\tilde{\mathcal{P}}_h$ are the angular meshes of R and \tilde{R} respectively, $\tilde{\boldsymbol{\pi}}$ are the solution coefficients on patch P , and \mathbf{g} are the spatial basis functions on element R .

The test functions are \mathbf{g} and \mathbf{f} . They have support on the element R and the patch $P \in \mathcal{P}_h$ respectively. Substitute equation 4.3 into equation 4.1 and multiply by the matrix $\mathbf{g} \mathbf{f}^T$. Integrate over the whole phase-space to find

$$\begin{aligned} & \sum_{\tilde{R} \in \mathcal{R}_q} \int_{\tilde{R}} \mathbf{g} \left(\frac{\partial \tilde{\mathbf{g}}}{\partial \mathbf{r}} \tilde{\boldsymbol{\rho}} \right)^T \sum_{\tilde{P} \in \tilde{\mathcal{P}}_h} \int_{\tilde{P}} \Omega \tilde{\boldsymbol{\pi}}^T \tilde{\mathbf{f}} \mathbf{f}^T \\ & + \int_R \Sigma_a \mathbf{g} \mathbf{g}^T \boldsymbol{\rho} \int_P \boldsymbol{\pi}^T \mathbf{f} \mathbf{f}^T \\ & - \int_R \mathbf{g} \mathbf{g}^T \boldsymbol{\rho} \sum_{\tilde{P} \in \tilde{\mathcal{P}}_h} \int_{\tilde{P}} \tilde{\boldsymbol{\pi}}^T \left(\frac{\alpha}{2} \Delta_s \tilde{\mathbf{f}} \right) \mathbf{f}^T \\ & = \int_R \int_P S \mathbf{g} \mathbf{f}^T . \end{aligned} \quad (4.4)$$

Here \tilde{R} and \tilde{P} are just dummy variables. Some summations in equation 4.3 dropped out because all basis functions have support on only one element or patch. The remaining sums require special attention.

The Fokker-Planck operator is dealt with in the manner that is described in chapter 2. That is, the discontinuities in the angular basis functions necessitate a consistency term, and symmetry and penalty terms can be added without destroying the consistency of the weak form. For any two sets of angular basis functions $\{u_i\}$ and $\{v_i\}$, the SIP bilinear operator is defined component-wise by

$$\begin{aligned} [b_{[P]}(\mathbf{u}, \mathbf{v})]_{ij} &= \int_P \frac{\alpha}{2} \nabla_s u_i \cdot \nabla_s v_i + \sum_{S \in \mathcal{F}_P} \int_S \frac{\alpha}{2} \frac{\eta}{h_F} \llbracket u_i \rrbracket \llbracket v_i \rrbracket + \int_P \Sigma_a u_i v_i \quad (4.5) \\ &\quad - \sum_{S \in \mathcal{F}_P} \int_S \left(\llbracket v_i \rrbracket \left\{ \frac{\alpha}{2} \nabla_s u_i \right\} \cdot \mathbf{n}_{P,S} + \llbracket u_i \rrbracket \left\{ \frac{\alpha}{2} \nabla_s v_i \right\} \cdot \mathbf{n}_{P,S} \right), \end{aligned}$$

where $\mathbf{n}_{P,S}$ is the outward normal vector of patch P at face S , parallel to the surface of the unit sphere. $b(\tilde{\mathbf{f}}, \mathbf{f})$ is non-zero only if the intersection between P and \tilde{P} is non-empty. Note that it also contains the absorption term.

The spatial streaming term in equation 4.4 is separated into a volumetric part and a facial part. In the following, \tilde{R} is the neighbour of R at face F , so that $\mathbf{n}_{R,F} = -\mathbf{n}_{\tilde{R},F}$. Denote by $\{\tilde{P}_i\}_{i=1}^Z$ the set of all patches on the angular mesh of \tilde{R} that overlap with a patch P on the angular mesh of R . Introduce the matrix

$$N_{[R]} \equiv \int_R \mathbf{g}_{[R]} \mathbf{g}_{[R]}^T$$

and the following third order tensors, defined through their components in a Cartesian coordinate system.

$$\begin{aligned} [V_{[R]}]_{ijk} &\equiv \int_R [\mathbf{g}_{[R]}]_i \frac{\partial}{\partial r_j} [\mathbf{g}_{[R]}]_k \\ [W_{[P]}]_{ijk} &\equiv \int_P [\mathbf{f}_{[P]}]_i \Omega_j [\mathbf{f}_{[P]}]_k \end{aligned}$$

A naive approach might start with an integration by parts:

$$\int_R \mathbf{g} \left(\frac{\partial \tilde{\mathbf{g}}}{\partial \mathbf{r}} \tilde{\boldsymbol{\rho}} \right)^T = \sum_{F \in \mathcal{F}_R} \int_F \mathbf{g} \tilde{\mathbf{g}}^T \tilde{\boldsymbol{\rho}} (\mathbf{n}_{F,\tilde{R}})^T - \int_R (\tilde{\boldsymbol{\rho}} \cdot \tilde{\mathbf{g}}) \left(\frac{\partial \mathbf{g}}{\partial \mathbf{r}} \right)^T, \quad (4.6)$$

where \mathcal{F}_R is the set of faces of R . A substitution of equations 4.6 and 4.5 into equation 4.4 would result in

$$\sum_{F \in \mathcal{F}_R} \Upsilon_F - (\boldsymbol{\rho} \cdot V) (\boldsymbol{\pi} \cdot W) - (N \boldsymbol{\rho}) \sum_{\tilde{p} \in \mathcal{P}} \boldsymbol{\pi}_{[\tilde{p}]}^T b_{[\tilde{p}]}(\mathbf{f}_{[\tilde{p}]}, \mathbf{f}_{[P]}) = \int_R \int_P S \mathbf{g} \mathbf{f}^T, \quad (4.7)$$

where \cdot indicates a tensor contraction and

$$\Upsilon_F = \int_F \mathbf{g} \left(\mathbf{g}^T \boldsymbol{\rho} \int_P \mathbf{n}_{F,R}^T \boldsymbol{\Omega} \boldsymbol{\pi}^T \mathbf{f} \mathbf{f}^T + \tilde{\mathbf{g}}^T \tilde{\boldsymbol{\rho}} \sum_{i=1}^Z \left[\int_{\tilde{P}} \mathbf{n}_{F,\tilde{R}}^T \boldsymbol{\Omega} \boldsymbol{\pi}_{[\tilde{P}]}^T \mathbf{f}_{[\tilde{P}]} \mathbf{f}_{[\tilde{P}]}^T \right]_{\tilde{P}=\tilde{P}_i}^T \right)$$

is the surface streaming term. The problem with this is that it does not properly separate the incoming and outgoing fluxes at the faces of the spatial elements. It was derived without full consideration for the relation between the spaces H and \mathbb{S}^2 . Whether a particle in R effects a particle in a neighbouring element depends on its direction.

Equation 4.7 does hold for a corrected version of the surface streaming term, based on an Riemann upwinding scheme that was developed recently by Kópházi and Lathouwers (2015). The weak form of the transport equation is given by equation 4.7 with

$$\Upsilon_F = \int_F \mathbf{g} \sum_{i=1}^Z \left[\mathbf{g}^T \boldsymbol{\rho} \Lambda_i \boldsymbol{\pi} + \mathbf{g}_{[\tilde{P}]}^T \boldsymbol{\rho}_{[\tilde{P}]} \tilde{\Lambda}_i \boldsymbol{\pi}_{[\tilde{P}]} \right]_{\tilde{P}=\tilde{P}_i}^T. \quad (4.8)$$

Here Λ_i and $\tilde{\Lambda}_i$ are matrices that filter out the outgoing or incoming surface streams from R and \tilde{R} respectively. Before providing explicit expressions, define

$$M_{[p_1,p_2]} \equiv \int_{p_1} \mathbf{f}_{[p_1]} \mathbf{f}_{[p_2]}^T \quad (4.9)$$

for any two patches p_1 and p_2 , which aren't necessarily in the same angular mesh. Let

$$A_{[P]} \equiv \int_P \mathbf{f}_{[P]} (\mathbf{n}_{R,F} \cdot \boldsymbol{\Omega}) \mathbf{f}_{[P]}^T \quad \text{and} \quad A_{[\tilde{P}_i]} \equiv \int_{\tilde{P}_i} \mathbf{f}_{[\tilde{P}_i]} (\mathbf{n}_{\tilde{R},F} \cdot \boldsymbol{\Omega}) \mathbf{f}_{[\tilde{P}_i]}^T.$$

Notice the different spatial normal vector at the face F . Determine the unique matrix $Q_{[p]}$ for which

$$A_{[p]} = M_{[p,p]} Q_{[p]} G_{[p]} Q_{[p]}^{-1}, \quad (4.10)$$

where $G_{[p]}$ is a diagonal matrix. This is a diagonalisation based on the generalised eigenvalue problem where $M_{[p,p]}$ is the metric. Split $G_{[p]}$ into $G_{[p]}^+$ and $G_{[p]}^-$, which contain the positive and negative eigenvalues respectively, so that $G_{[p]} = G_{[p]}^+ + G_{[p]}^-$. Write $A_{[p]}^\pm = M_{[p,p]} Q_{[p]} G_{[p]}^\pm Q_{[p]}^{-1}$. Now there are three separate cases to consider.

- $Z = 1$ and the patch \tilde{P}_1 on \tilde{R} is of the same level of refinement as the patch P on R . This means that P and \tilde{P}_1 are basically equivalent.

$$\Lambda_1 = A_{[P]}^+ \quad \text{and} \quad \tilde{\Lambda}_1 = A_{[\tilde{P}_1]}^- = A_{[P]}^- \quad (4.11)$$

- $Z = 1$ and the patch \tilde{P}_1 on \tilde{R} is coarser than the patch P on R . That is, \tilde{P}_1 is a larger patch, at a lower level of refinement. The incoming stream from \tilde{P} is expressed in terms of different angular basis functions from those on P . Transformations from one basis to another are performed by the matrices in equation 4.9.

$$\Lambda_1 = A_{[P]}^+ \quad \text{and} \quad \tilde{\Lambda}_1 = A_{[P]}^- M_{[P,P]}^{-1} M_{[P,\tilde{P}_1]} \quad (4.12)$$

- $Z > 1$ and the patches $\{\tilde{P}_i\}_{i=1}^Z$ on \tilde{R} are of a higher level of refinement than the patch P on R .

$$\begin{aligned} \Lambda_i &= M_{[P,\tilde{P}_i]} M_{[\tilde{P}_i,\tilde{P}_i]}^{-1} A_{[\tilde{P}_i]}^+ M_{[\tilde{P}_i,\tilde{P}_i]}^{-1} M_{[\tilde{P}_i,P]} \\ \tilde{\Lambda}_i &= M_{[P,\tilde{P}_i]} M_{[\tilde{P}_i,\tilde{P}_i]}^{-1} A_{[\tilde{P}_i]}^- \end{aligned} \quad (4.13)$$

The operator Λ_i first projects the flux in P onto the angular basis of \tilde{P}_i , then the outgoing component is selected there, and finally the result is transformed back to P .

Dirichlet boundary conditions of the form 4.2 are implemented by using a ‘shadow element’. If an element R borders ∂H at the face $F = R \cap \partial H$, then a shadow element \tilde{R} is placed on the other side of F . The solution coefficients $\tilde{\rho}$ and $\tilde{\pi}$ on \tilde{R} are set by a Galerkin projection of $\varphi^D(\mathbf{r}, \mathbf{\Omega})$ onto the solution space on \tilde{R} .

4.2 Solution methods

Fortunately, the Riemann upwinding procedure is much less expensive than it first appears, because it is not always necessary to perform the decomposition in equation 4.10. Suppose that $\mathbf{\Omega} \cdot \mathbf{n}_{R,F} > 0$ for all $\mathbf{\Omega} \in P$ for a normal vector $\mathbf{n}_{R,F}$ at face F of spatial element R . This means that there are no incoming particles that stream across F with a direction in P . Thus $A_{[P]}^- = 0$ and $A_{[P]}^+ = A_{[P]}$. Similarly, if $\mathbf{\Omega} \cdot \mathbf{n}_{R,F} < 0$ everywhere on P , then $A_{[P]}^+ = 0$ and $A_{[P]}^- = A_{[P]}$. As the angular mesh becomes more refined, there

Table 4.1: Pseudocode indicating the solution order in a the block Gauss-Seidel method

1.	for $i = 1 \dots 8$	# loop over octants of \mathbb{S}^2
2.	for all $R \in \mathcal{R}^{(i)}$	# loop over ordered set
3.	for all $P \in \mathcal{P}_{[R]}^{(i)}$	# loop over unordered set
4.	<i>Solve for $\boldsymbol{\pi}_{[P]}$ and $\boldsymbol{\rho}_{[R]}$ from equation 4.7 with 4.8, holding all other coefficients constant.</i>	
5.	end	
6.	end	
7.	end	

are continuously fewer patches upon which $\boldsymbol{\Omega} \cdot \mathbf{n}_{R,F}$ changes value, and the Riemann decomposition only has to be performed in a small percentage of cases.

Enforcing equation 4.7 with 4.8 to hold for all $\mathbf{g} \mathbf{f}^T$ on all patches leads to a linear system in the unknown coefficients $\boldsymbol{\pi}$ and $\boldsymbol{\rho}$, which are solved for by iterative means. An obvious and simple method is the block Gauss-Seidel method, where the blocks consist of all unknowns on a patch. The order in which the patches are visited is of great importance. Let \mathbf{v}_i be the centre of the i 'th octant of the unit sphere. Denote by $\mathcal{P}_{[R]}^{(i)}$ all patches in the angular mesh of R that are in the i 'th octant. $\{\mathbf{v}_i\}_{i=1}^8$ are the directions in an S_2 discrete ordinate method. Let $\mathcal{R}^{(i)}$ be a list (that is, an ordered set) of the spatial elements elements, arranged in the order in which one would sweep them for the direction \mathbf{v}_i in an S_N method. The block Gauss-Seidel solver is summarised by the pseudocode in table 4.1. This particular solution order could very probably be improved upon, but it served well enough for the purposes for this thesis.

The convergence can be sped up considerably with a GMRES (generalized minimal residual) or BiCGSTAB (biconjugate gradient stabilized) iterative method that uses the Gauss-Seidel sweep as a preconditioner. An alternative is Richardson iteration. All these methods are detailed in Golub and Loan (1996).

4.2.1 Convergence of the linear solvers

In order to test the efficacy of the linear solvers described in the previous section, several numerical tests were done. In general, both the GMRES and

the BiCGSTAB solvers clearly outperform the Richardson solver in every case. If the number of solver steps is constant, then BiCGSTAB provides a better result than GMRES. This is compensated by the fact that the required computation time per time step is larger. It is not clear from the tests which solver is best.

This is illustrated with a few examples on the domain $H = (0, 1) \times (0, 1) \times (0, 1) \text{ cm}^3$. The spatial mesh was highly irregular and consisted of 63 tetrahedrons of various sizes. The basis consisted of piecewise linear spatial functions and Ω -functions in angle. The material properties were $\Sigma_a = 0.10 \text{ cm}^{-1}$ and $\alpha = 0.25 \text{ cm}^{-1}$. The source is a constant $S = 1/(4\pi) \text{ cm}^{-3}\text{s}^{-1}$ and there are no boundary sources: $\phi^D = 0$.

The speed of convergence depends heavily on the tessellation of the unit sphere. Figure 4.1 displays the convergence for the case where the angular mesh was uniformly refined to level 3. The residuals are normalised by the residual after the first iteration. Figure 4.2 is similar, but here each spatial element was assigned a distinct angular mesh. In each angular mesh, a single element was randomly selected and refined. This process was repeated until all angular meshes had 500 elements in total.¹ Note how the required computing time per time step is lower on uniformly refined meshes. This is because the sum in equation 4.8 always contains only one term. Also, equation 4.11 is easier to compute than equations 4.13 or 4.13. The speed of convergence per solver step is also greater on the uniformly refined mesh. If the spatial elements all have different angular meshes, then it's difficult to see in what order the elements should be visited by the Gauss-Seidel sweeper.

The speed of convergence hardly changes with a changing spatial mesh, though it obviously matters how many spatial elements there are in total.

4.3 Convergence with mesh refinement

When considering the convergence of the numerical solution to the exact particle flux, there are two discretisation to take into account. Let ℓ and h be the typical length scale of the spatial and the angular mesh respectively.

¹This type of highly irregular discretisations can also be used to verify that the method is implemented correctly. If S is chosen such that the exact solution to equation 4.1 is part of the DG solution space, then the numerical solution should equal the exact solution. It is not easy to see how a badly implemented code could yield the right result in several of these consistency tests on randomly refined meshes.

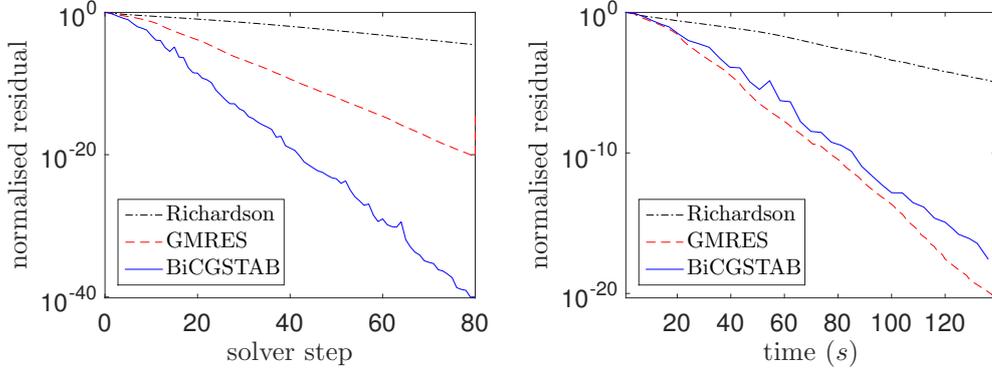


Figure 4.1: Convergence of the solution vector with different numerical solvers. The angular meshes are uniformly refined at level 3.

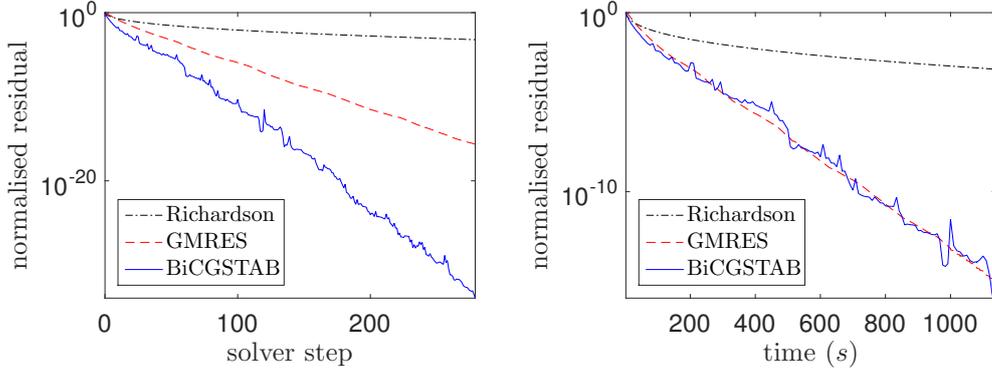


Figure 4.2: Convergence of the solution vector with different numerical solvers. The angular meshes are randomly refined with 500 elements each.

Define a global error

$$e \equiv \sqrt{\frac{\langle \varphi_{\ell h} - \varphi, \varphi_{\ell h} - \varphi \rangle}{\langle \varphi, \varphi \rangle}} . \quad (4.14)$$

The inner product is defined as

$$\langle v, w \rangle \equiv \int_H \int_{S^2} vw .$$

Loosely speaking, if ℓ is in a region such that the error is dominated by the spatial discretisation, and e/ℓ^n approaches a constant as ℓ decreases, then n is the order of convergence. In particular, if the angular dependence of the

Table 4.2: The errors are given by equation 4.14. The order of convergence on \mathcal{R}_ℓ is estimated with equation 3.6 and a comparison with the result on $\mathcal{R}_{2\ell}$.

card(\mathcal{R}_ℓ)	$p = 0$		$p = 1$		$p = 2$	
	e	order	e	order	e	order
8^0	$3.71 \cdot 10^0$	-	$7.92 \cdot 10^{-1}$	-	$6.93 \cdot 10^{-1}$	-
8^1	$1.35 \cdot 10^0$	1.46	$7.18 \cdot 10^{-1}$	1.46	$2.54 \cdot 10^{-1}$	1.45
8^2	$8.37 \cdot 10^{-1}$	0.69	$2.00 \cdot 10^{-1}$	2.76	$3.36 \cdot 10^{-2}$	2.92
8^3	$4.99 \cdot 10^{-1}$	0.75	$4.70 \cdot 10^{-2}$	2.09	$4.13 \cdot 10^{-3}$	3.03
8^4	$2.80 \cdot 10^{-1}$	0.84	$1.10 \cdot 10^{-2}$	2.10	$5.09 \cdot 10^{-4}$	3.02

card(\mathcal{R}_ℓ)	$p = 3$		$p = 4$	
	e	order	e	order
8^0	$6.19 \cdot 10^{-1}$	-	$1.64 \cdot 10^{-1}$	-
8^1	$7.19 \cdot 10^{-2}$	3.11	$1.67 \cdot 10^{-2}$	3.30
8^2	$4.82 \cdot 10^{-3}$	3.90	$5.75 \cdot 10^{-4}$	4.85
8^3	$2.98 \cdot 10^{-4}$	4.01	$1.81 \cdot 10^{-5}$	5.00
8^4	$1.85 \cdot 10^{-5}$	4.00	$5.66 \cdot 10^{-7}$	5.00

exact solution can be expressed exactly in the angular basis functions, then e/ℓ^n approaches a constant as $\ell \rightarrow 0$.

As an example, consider again the domain $H = (0, 1) \times (0, 1) \times (0, 1) \text{ cm}^3$. The source and boundary condition are chosen such that

$$\varphi(\mathbf{r}, \mathbf{\Omega}) = \left(\prod_{i=1}^3 \sin(\pi r_i) \right) (\Omega_1 + 3\Omega_2 + 7\Omega_3)$$

is the exact solution. The angular basis functions are linear in $\mathbf{\Omega}$, so the error in the numerical solution is solely due to the spatial discretisation. The spatial elements are identical cubes. The spatial basis functions are piecewise constant polynomials of order $\leq p$. Table 4.2 lists the errors and orders of convergence for different values of p and on meshes of varying refinement. The numerical solution appears to converge with order $p+1$. The asymptotic speed of convergence sets in more quickly for higher order functions.

In similar tests, it was verified numerically that the angular error is of order h^2 , as was already demonstrated in section 3.2.

4.4 A diffusing proton beam

A more interesting numerical example is given in figure 4.3. It simulates an infinitely narrow incoming proton beam in both a homogeneous and a heterogeneous medium. Note how the spatial mesh is refined locally. The angular meshes are also refined locally, depending on the spatial element. This is illustrated in figure 4.4. The spatial basis is piecewise constant. The angular basis is spanned by Ω -functions. H is bounded by $-\infty < r_3 < \infty$, $r_1 = \pm 0.25$ cm, $r_2 = 0.0$ cm and $r_2 = 1.0$ cm. It is essentially a two-dimensional problem. The volumetric source is $S = 0$ cm⁻³s⁻¹. The boundary condition is given by $\varphi^D(\mathbf{r}, \Omega) = \varphi_0 \delta(r_1)\delta(r_2)\delta(\Omega - \mathbf{e}_2)$ and $\varphi_0 = 1000$ cm⁻²s⁻¹. In the computation of the Galerkin projection of φ^D onto the shadow elements, this boundary source was split evenly over two elements and a total of 8 patches, all of which bordered $\Omega = \mathbf{e}_2$.

The model correctly predicts a non-symmetrical scalar flux. This is in contrast to the Fermi pencil beam approximation, that predicts the same result in both the homogeneous and the heterogeneous case.

The conservative properties of the discretisation were verified numerically. That is, in a domain without absorption, the leakage out of the system equals the sum of the contributions of the volumetric and boundary sources. In other words, the numerical scheme preserves particles.

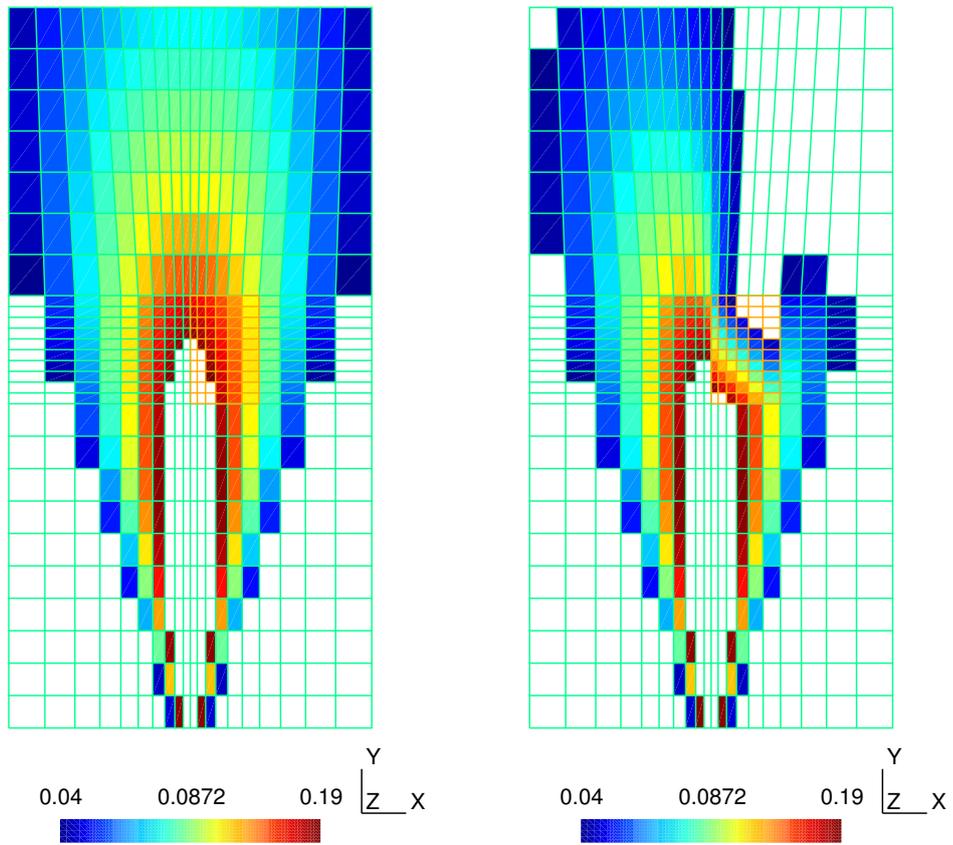
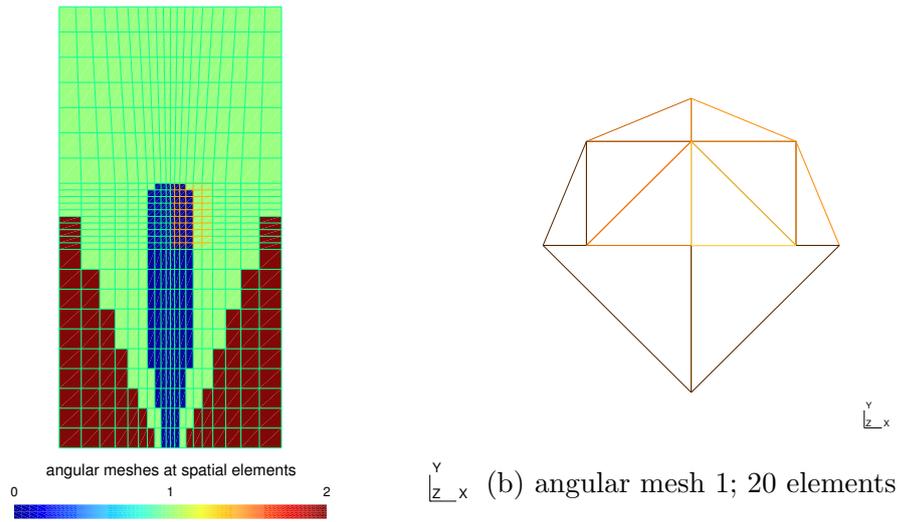


Figure 4.3: Scalar flux ϕ in units of $\text{cm}^{-2}\text{s}^{-1}$ on the spatial domain H . The colour scale is logarithmic. Values that are too high or too low were left out. On the left, the macroscopic cross sections are the same everywhere at $\alpha = 0.25$ cm and $\Sigma_a = 0.0$ cm. On the right, there is a block, located at $0.0 \text{ cm} < r_1 < 0.095 \text{ cm}$ and $0.45 \text{ cm} < r_2 < 0.6 \text{ cm}$, for which $\alpha = 0.25$ cm and $\Sigma_a = 10.0$ cm.



(a) Each spatial element is associated with angular mesh 1, 2 or 3.

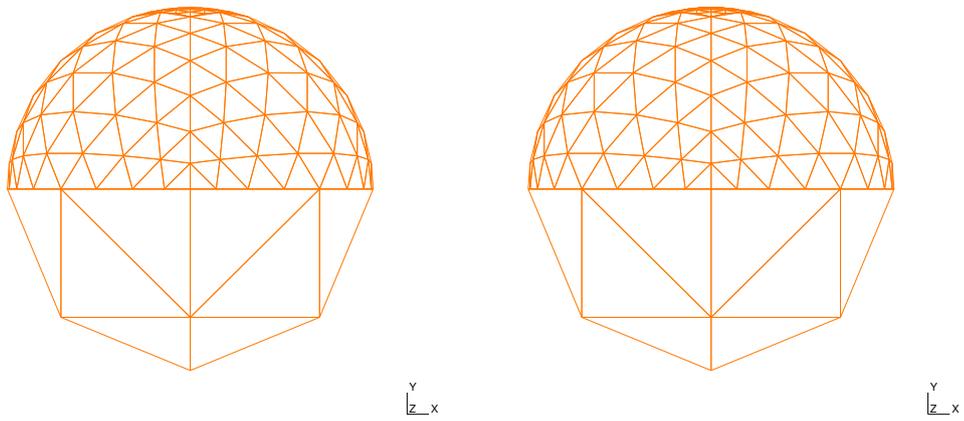


Figure 4.4: Angular meshes corresponding to figure 4.3

Chapter 5

Conclusion

The spherical adaptation of the SIP methods in chapters 2 and 3 is a fairly straightforward extension of the regular Euclidean SIP method. The L^2 -norm of the error (equation 3.5) behaves as h^2 . They perform reliably on the spherical domain, even on irregular tessellations or highly localised angular sources. It is not completely clear what the best basis functions are in a ‘regular’ SIPG method on the sphere. It seems the $\mathbf{\Omega}$ -functions are to be preferred, but only slightly. They have a slightly lower error per degree of freedom. More importantly, they preserve the angular current density.

The ccG method is still not as fully developed as the SIPG methods. For example, it is not fully clear how to choose the penalty parameter. Nevertheless, it already performs a bit better than SIPG in the absence of spatial streaming. It will require a significant coding effort to implement it into the code for spatial streaming, since the implementation is very different from the more common DG methods.

The Riemann upwinding procedure is easily combined with spherical SIP. The L^2 -norm of the error behaves as ℓ^{p+1} for spatial basis functions of order p on a spatial mesh with typical length scale ℓ , provided that the error due to the angular discretisation is negligible.

Chapter 6

Future work

The spherical SIPG method described in chapters 2 and 3 is not likely to be improved upon significantly. A possible exception is the use of higher order basis functions that were mentioned in section 2.3, especially in cases where the flux is fairly smooth in Ω . This is more likely to be the case in neutral particle transport.

There are several possible extensions of the theory in chapters 2 and 3 to other disciplines. For example, the methods could easily be extended to the unit hypersphere \mathbb{S}^N that's embedded in \mathbb{R}^{N+1} . The only adjustment in the derivations is that the '2' in equations 2.6 and 2.7 are replaced by N . The analyses in the appendices carry over in a straightforward manner.

A more physical alternative application can be found in geomathematics. In the simulation of very large weather events and tsunamis, one has to take into account the curvature of the earth. Tsunamis are shock waves, and thus typically upwinded. When they come into contact with the shore, the computational domain can be highly irregular. These properties mean that tsunami predictions could well be fertile research ground for spherical discontinuous Galerkin methods.

Immediate future work should focus on an implementation of an automatic adaptation method. That is, the mesh should be refined automatically during run-time, based on some error measure. This is the only way fully to exploit the flexibility of the numerical scheme, which is its greatest asset.

A significant gain in efficiency is to be expected from a better linear solver for the spatial transport. The Gauss-Seidel sweep could be improved, so that it becomes a better preconditioner. The order in which the spatial elements are visited is now based on the S_2 discrete ordinates sweeper. This is not ideal for spherical elements that are far from the centre of an octant. In principle, every angular element has its own ideal order in which GS sweep

should visit the spatial elements. In practice, some spherical elements could be preconditioned with a multigrid solver, and the sweep order could be based on the elements at a lower refinement level.

Another possible way to speed up the linear solver would be to make use of approximate analytical solutions when available. For example, one could take a Galerkin projection of the simple Fermi pencil beam onto the solution space, and then use the resulting solution vector as a starting point for the iteration.

The weak form as it is derived in section 4.1 could be adapted to more general scattering terms than the bilinear form corresponding to the Fokker-Planck operator. The error in the Fokker-Planck approximation is usually considerably larger than the error in the pencil beam approximation. Though a general scattering term may be significantly different quantitatively, it should be qualitatively similar. The efficacy of the solution methods in chapter 4 should therefore not be greatly impacted.

An easy way to make the calculations more realistic would be to incorporate the energy dependence. The energy could be discretised with a DG method, and upwinded from high to low energy. This has already been implemented into the code that was used for this thesis, but not extensively tested with the numerical methods described here.

Appendix A

Parameterisations of spherical elements

A.1 Parameterisations of orthodrome segments

The boundaries of the elements in the spherical mesh are orthodrome segments. An orthodrome (or ‘great circle’) is the intersection of \mathbb{S}^2 and a plane that passes through the origin. An orthodrome segment \mathcal{C} is a closed, connected subset thereof. It is defined uniquely by its endpoints (‘vertices’) $\{\mathbf{V}_i\}_{i=1}^2$. It is the shortest curve on \mathbb{S}^2 that connects \mathbf{V}_1 to \mathbf{V}_2 .

Parameterisations of orthodrome segments form the basis of both analytical integration and numerical integration techniques. They are bijections between all points $\boldsymbol{\Omega} \in \mathcal{C}$ and all $k \in [-1, 1]$. The rest of this section is devoted to several examples.

A.1.1 Parameterisations via a line segment in \mathbb{R}^3

One way to parameterise \mathcal{C} is to map k onto a line segment $\mathcal{L} \in \mathbb{R}^3$, and then to project all points $\mathbf{l} \in \mathcal{L}$ onto points $\boldsymbol{\Omega} \in \mathcal{C} \subset \mathbb{S}^2$ along a straight line through the origin. Specifically, if $\{\mathbf{r}_i\}_{i=1}^2$ are the endpoints of \mathcal{L} , then

$$\mathbf{l}(k) = \frac{1}{2}(1 - k)\mathbf{r}_1 + \frac{1}{2}(1 + k)\mathbf{r}_2, \quad -1 \leq k \leq 1 \quad (\text{A.1})$$

and $\boldsymbol{\Omega} = \mathbf{l}/l$. Obviously, $\mathbf{V}_1 = \mathbf{r}_1/r_1$ and $\mathbf{V}_2 = \mathbf{r}_2/r_2$.

The Jacobian associated with equation A.1.1 can be found by direct differentiation, similarly to what is done in section A.1.2. Alternatively, the norm of the Jacobian can be found more directly by considering the

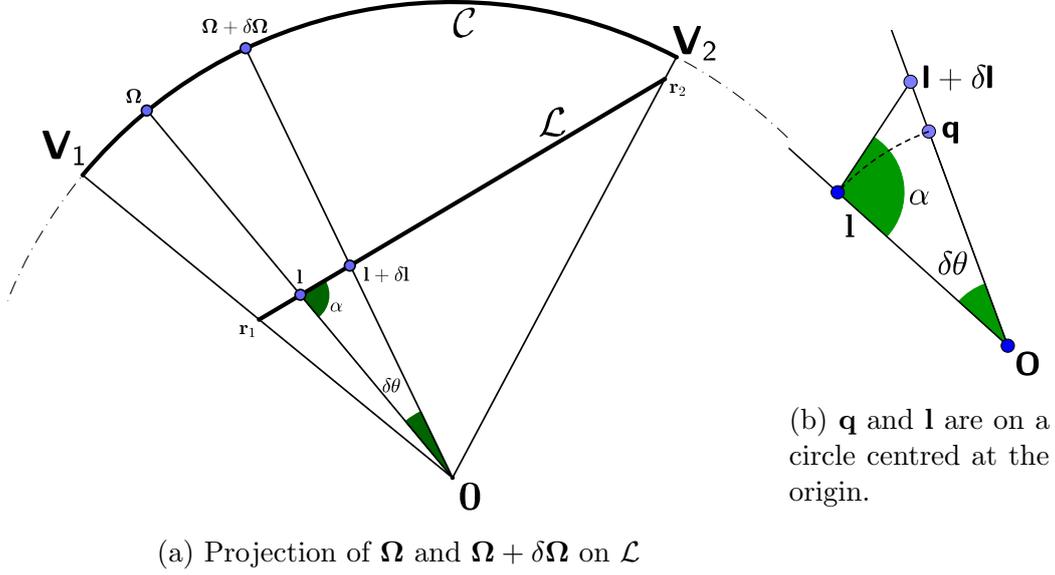


Figure A.1: A schematic overview of the parameterisation of a great circle segment on the unit sphere as treated in section A.1.1. It is not necessary that $r_1, r_2 < 1$; the analysis holds more generally.

pair of displacements

$$\mathbf{l} \longrightarrow \mathbf{l} + \delta\mathbf{l} = \mathbf{l} + \delta l \frac{\mathbf{r}_2 - \mathbf{l}}{\|\mathbf{r}_2 - \mathbf{l}\|_2}$$

$$\Omega \longrightarrow \Omega + \delta\Omega = \frac{1}{\|\mathbf{l} + \delta\mathbf{l}\|_2} (\mathbf{l} + \delta\mathbf{l}) ,$$

as indicated in figure A.1a. Define the point

$$\mathbf{q} \equiv \frac{l}{\|\mathbf{l} + \delta\mathbf{l}\|_2} (\mathbf{l} + \delta\mathbf{l})$$

and the angles

$$\alpha \equiv \angle \mathbf{0} \mathbf{l} \mathbf{r}_2 = \arccos \left(\frac{-\mathbf{l} \cdot (\mathbf{r}_2 - \mathbf{l})}{l \|\mathbf{r}_2 - \mathbf{l}\|_2} \right)$$

and

$$\delta\theta \equiv \angle \mathbf{l} \mathbf{0} \mathbf{q} = \arccos \left(\frac{\mathbf{l} \cdot \mathbf{q}}{lq} \right) ,$$

as indicated in figure A.1b. The sine law for the triangle spanned by $(\mathbf{q} - \mathbf{l})$ and $\delta \mathbf{l}$ yields

$$\frac{\delta l}{\sin\left(\frac{1}{2}(\pi - \delta\theta)\right)} = \frac{\|\mathbf{q} - \mathbf{l}\|_2}{\sin(\alpha + \delta\theta)}.$$

Noting that $0 < \alpha < \pi$ implies $\sin(\arccos(\alpha)) = \sqrt{1 - \alpha^2}$, and that $\|\mathbf{q} - \mathbf{l}\|_2 = l\delta\theta + \mathcal{O}(\delta\theta^3)$ and $\|\delta\boldsymbol{\Omega}\|_2 = \delta\theta + \mathcal{O}(\delta\theta^3)$, the last equation can be rearranged to

$$\lim_{\delta\theta \rightarrow 0} \frac{\|\delta\boldsymbol{\Omega}\|_2}{\delta l} = \left\| \frac{d\boldsymbol{\Omega}}{dl} \right\|_2 = \frac{\sqrt{l^2(\mathbf{r}_2 - \mathbf{l})^2 - [\mathbf{l} \cdot (\mathbf{r}_2 - \mathbf{l})]^2}}{l^2 \|\mathbf{r}_2 - \mathbf{l}\|_2}. \quad (\text{A.2})$$

A.1.2 Parameterisation in angle

It is also possible to map linearly from $[-1, 1]$ to the angle between $\boldsymbol{\Omega}$ and \mathbf{V}_1 , which is the subject of this subsection. Due to its applicability later on, the analysis will be slightly more general than it has to be for the parameterisation of a great circle segment. Let \mathbf{W}_1 and \mathbf{W}_2 be two distinct points on an arbitrary circle in \mathbb{R}^3 with centre \mathbf{c} , such that $\mathbf{W}_1 - \mathbf{c} \neq \mathbf{c} - \mathbf{W}_2$ (ref. figure A.2). A normal vector of the plane of the circle is given by

$$\mathbf{d} = \frac{(\mathbf{W}_1 - \mathbf{c}) \times (\mathbf{W}_2 - \mathbf{c})}{\|(\mathbf{W}_1 - \mathbf{c}) \times (\mathbf{W}_2 - \mathbf{c})\|_2}.$$

Since $\mathbf{d} \times (\mathbf{W}_1 - \mathbf{c})$ and $\mathbf{W}_1 - \mathbf{c}$ are both parallel to the plane of the circle and are perpendicular, any point \mathbf{w} on the circle can be decomposed as

$$\mathbf{w} - \mathbf{c} = x_1 (\mathbf{W}_1 - \mathbf{c}) + x_2 (\mathbf{d} \times (\mathbf{W}_1 - \mathbf{c})),$$

where $x_1 \equiv (\mathbf{w} - \mathbf{c}) \cdot (\mathbf{W}_1 - \mathbf{c})$ and $x_2 \equiv (\mathbf{w} - \mathbf{c}) \cdot (\mathbf{d} \times (\mathbf{W}_1 - \mathbf{c}))$. The vector $\mathbf{d} \times (\mathbf{W}_1 - \mathbf{c})$ has length $|\mathbf{W}_1 - \mathbf{c}|$, so, by Pythagoras's theorem, \mathbf{w} lies on the circle if and only if $x_1^2 + x_2^2 = 1$. From this fact, it can be seen that

$$\mathbf{w}(t) = \mathbf{c} + (\mathbf{W}_1 - \mathbf{c}) \cos((t - t_0)\chi) + \mathbf{d} \times (\mathbf{W}_1 - \mathbf{c}) \sin((t - t_0)\chi), \quad t_0 \leq t \leq t_1 \quad (\text{A.3})$$

is a segment of the circle for all $\chi > 0$. The velocity can be found either algebraically or geometrically and is given by

$$\frac{d\mathbf{w}}{dt} = \chi \mathbf{d} \times (\mathbf{w}(t) - \mathbf{c}). \quad (\text{A.4})$$

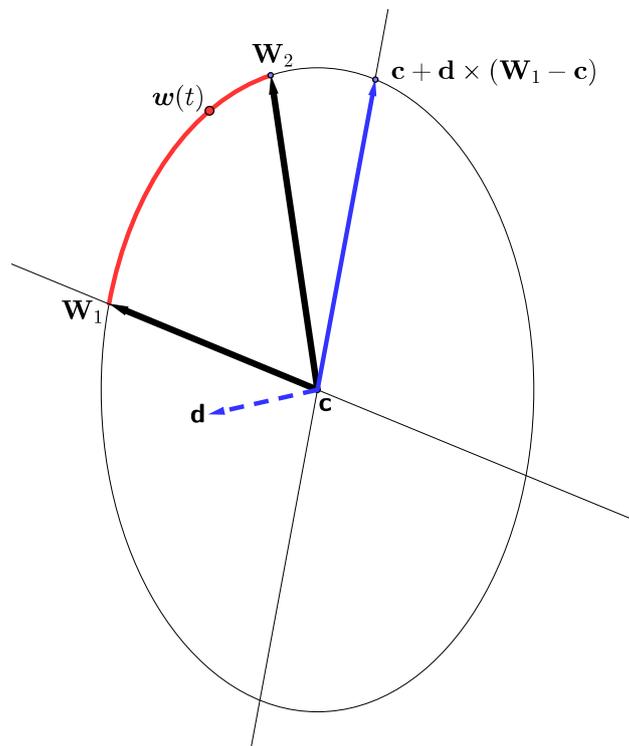


Figure A.2: A schematic overview of the parameterisation of an arbitrary circle segment in \mathbb{R}^3 as treated in section A.1.2. Here \mathbf{d} is perpendicular to the plane of the circle and points into the paper.

This parametrisation is *natural*: the speed $\|\mathrm{d}\mathbf{w}/\mathrm{d}t\|_2 = \chi \|\mathbf{W}_1 - \mathbf{c}\|_2$ is constant. As t increases from t_0 to t_1 , $\mathbf{w}(t)$ starts at \mathbf{W}_1 and traverses an angle $\chi(t_1 - t_0)$ in the direction of \mathbf{W}_2 . A parametrisation of the shortest circle segment that connects \mathbf{W}_1 to \mathbf{W}_2 is given by equation A.3, with the extra condition

$$\chi(t_1 - t_0) = \arccos\left(\frac{(\mathbf{W}_1 - \mathbf{c}) \cdot (\mathbf{W}_2 - \mathbf{c})}{\|\mathbf{W}_1 - \mathbf{c}\|_2^2}\right).$$

In the special case of the great circle segment \mathcal{C} , \mathbf{c} is at the origin and $\mathbf{W}_1 = \mathbf{V}_1$ and $\mathbf{W}_2 = \mathbf{V}_2$ are on the unit sphere. Thus, a bijection between $k \in [-1, 1]$ and $\boldsymbol{\Omega} \in \mathcal{C}$ is

$$\boldsymbol{\Omega}(k) = \mathbf{V}_1 \cos\left(\frac{k+1}{2}\gamma\right) + \frac{(\mathbf{V}_1 \times \mathbf{V}_2) \times \mathbf{V}_1}{\|\mathbf{V}_1 \times \mathbf{V}_2\|_2} \sin\left(\frac{k+1}{2}\gamma\right), \quad -1 \leq k \leq 1, \quad (\text{A.5})$$

where

$$\gamma \equiv \arccos(\mathbf{V}_1 \cdot \mathbf{V}_2).$$

A.2 Parameterisations of spherical triangles

This section details several parameterisations of spherical triangles. A spherical triangle T is defined uniquely by its vertices $\{\mathbf{V}_p\}_{p=1}^3$ as the closed set of all $\boldsymbol{\Omega} \in \mathbb{S}^2$ that satisfy

$$\text{sign}(\mathbf{V}_m \cdot (\mathbf{V}_n \times \mathbf{V}_k)) = \text{sign}(\boldsymbol{\Omega} \cdot (\mathbf{V}_n \times \mathbf{V}_k))$$

for all permutations of $\{m, n, k\}$. It is the smallest part of \mathbb{S}^2 whose boundary consists of the three orthodrome segments that intersect \mathbf{V}_1 and \mathbf{V}_2 , \mathbf{V}_2 and \mathbf{V}_3 , and \mathbf{V}_3 and \mathbf{V}_1 respectively. The parameterisation is a bijection between T and the reference triangle

$$\mathcal{K}^{\text{ref}} \equiv \{\mathbf{k} \in \mathbb{R}^2 : k_1 \geq 0, k_2 \geq 0, k_1 + k_2 \leq 1\}. \quad (\text{A.6})$$

As is common in finite element methods, the independent variables are referred to as ‘local’ variables, and \mathcal{K}^{ref} is called the local element. It is the same for all spherical triangles.

A.2.1 Parameterisations via a flat triangle in \mathbb{R}^3

A family of parameterisations map \mathcal{K}^{ref} onto T via an intermediate flat triangle in \mathbb{R}^3 . Specifically, let $\mathcal{Z} \in \mathbb{R}^3$ be the flat triangle with vertices \mathbf{Z}_1 , \mathbf{Z}_2 and \mathbf{Z}_3 . A linear mapping between $\mathbf{k} = [k_1, k_2]^T \in \mathcal{K}^{\text{ref}}$ and $\mathbf{z} \in \mathcal{Z}$ is given by

$$\begin{aligned} \mathbf{z} &= \mathbf{Z}_1 + k_1 (\mathbf{Z}_2 - \mathbf{Z}_1) + k_2 (\mathbf{Z}_3 - \mathbf{Z}_1) \\ &= \mathbf{Z}_1 + D \mathbf{k} , \end{aligned} \tag{A.7}$$

where

$$D \equiv ((\mathbf{Z}_2 - \mathbf{Z}_1), (\mathbf{Z}_3 - \mathbf{Z}_1)) \in \mathbb{R}^{3 \times 2} .$$

Before specifying the \mathbf{Z}_i , some general consequences of equation A.7 are derived. The derivative with respect to \mathbf{k} is given by

$$\frac{\partial \mathbf{z}}{\partial \mathbf{k}} = D^T .$$

Note that there are three equations in A.7, though \mathcal{K}^{ref} has only two degrees of freedom. There are therefore several non-equivalent ways to solve for \mathbf{k} , the most convenient of which is to left-multiply by a left hand side inverse $E \in \mathbb{R}^{2 \times 3}$ of D . (That is, $ED = I^{2 \times 2}$.) Such a matrix can always be readily constructed by noting that \mathcal{Z} is non-singular, so that $D^T D$ is invertible¹, and setting $E = (D^T D)^{-1} D^T$. This yields

$$\mathbf{k} = (D^T D)^{-1} D^T (\mathbf{z} - \mathbf{Z}_1) . \tag{A.8}$$

The advantage of this particular representation is that the derivative with respect to \mathbf{z} becomes straightforward:

$$\frac{\partial \mathbf{k}}{\partial \mathbf{z}} = D (D^T D)^{-1} . \tag{A.9}$$

Interestingly, this is not a unique solution because E isn't unique. Given an infinitesimal displacement $d\mathbf{z}$, there are infinitely many $\partial \mathbf{k} / \partial \mathbf{z}$ with which the resulting displacement $d\mathbf{k} = (\partial \mathbf{k} / \partial \mathbf{z})^T d\mathbf{z}$ can be computed correctly. Algebraically, this is because D doesn't have full row rank, which in turn results

¹If, for some vector \mathbf{w} , $D^T D \mathbf{w} = \mathbf{0}$, then $0 = \mathbf{w}^T D^T D \mathbf{w} = \|D \mathbf{w}\|_2$, which implies $D \mathbf{w} = \mathbf{0}$. Due to the non-singularity of \mathcal{Z} , $(\mathbf{Z}_2 - \mathbf{Z}_1)$ and $(\mathbf{Z}_3 - \mathbf{Z}_1)$ are non-parallel, and D has full column rank. Therefore, 0 is not an eigenvalue of D , and so $\mathbf{w} = \mathbf{0}$, proving that 0 is not an eigenvalue of $D^T D$.

from the fact that equation A.7 is overdetermined if \mathbf{k} is the unknown. From a geometrical point of view it is a consequence of the constraint $d\mathbf{z} \in \mathcal{Z}$.

A mapping from the points $\mathbf{z} \in \mathcal{Z}$ to the points $\boldsymbol{\Omega} \in T$ should project the straight lines in $\partial\mathcal{Z}$ to the orthodrome segments in ∂T . The only simple bijection that does this is a projection along a line through the origin:

$$\boldsymbol{\Omega} = \frac{1}{z} \mathbf{z} , \quad (\text{A.10})$$

for which

$$\begin{aligned} \frac{\partial \boldsymbol{\Omega}}{\partial \mathbf{z}} &= \frac{1}{z} \frac{\partial \mathbf{z}}{\partial \mathbf{z}} + \frac{\partial z}{\partial \mathbf{z}} \frac{\partial(1/z)}{\partial z} \mathbf{z}^T \\ &= z^{-1} (I^{3 \times 3} - \boldsymbol{\Omega} \boldsymbol{\Omega}^T) . \end{aligned}$$

Since $\det(\partial \boldsymbol{\Omega} / \partial \mathbf{z}) = z^{-1}(1 - \boldsymbol{\Omega}^T \boldsymbol{\Omega})$, this matrix is singular, and $\partial \mathbf{z} / \partial \boldsymbol{\Omega}$ cannot be obtained by the usual method of inversion of the Jacobian. That is,

$$\frac{\partial \mathbf{z}}{\partial \boldsymbol{\Omega}} \frac{\partial \boldsymbol{\Omega}}{\partial \mathbf{z}} \neq I^{3 \times 3} \neq \frac{\partial \boldsymbol{\Omega}}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial \boldsymbol{\Omega}} ,$$

which is a consequence of the fact that $\boldsymbol{\Omega}$ and \mathbf{z} are constrained. Equation A.10 is inverted instead to obtain

$$\mathbf{z} = \frac{\mathbf{Z}_i \cdot \mathbf{n}}{\boldsymbol{\Omega} \cdot \mathbf{n}} \boldsymbol{\Omega} ,$$

where \mathbf{n} is a normal vector of \mathcal{Z} and $i \in \{1, 2, 3\}$ is arbitrary. It follows that

$$\frac{\partial \mathbf{z}}{\partial \boldsymbol{\Omega}} = \frac{\mathbf{Z}_i \cdot \mathbf{n}}{\boldsymbol{\Omega} \cdot \mathbf{n}} \left(I^{3 \times 3} - \frac{1}{\boldsymbol{\Omega} \cdot \mathbf{n}} \mathbf{n} \boldsymbol{\Omega}^T \right) . \quad (\text{A.11})$$

Note that

$$\boldsymbol{\Omega}^T \frac{\partial \mathbf{z}}{\partial \boldsymbol{\Omega}} = \mathbf{0}^T , \quad (\text{A.12})$$

as one would expect geometrically. The derivatives are related by

$$\frac{\partial \boldsymbol{\Omega}}{\partial \mathbf{k}} = \frac{\partial \mathbf{z}}{\partial \mathbf{k}} \frac{\partial \boldsymbol{\Omega}}{\partial \mathbf{z}} \quad \text{and} \quad \frac{\partial \mathbf{k}}{\partial \boldsymbol{\Omega}} = \frac{\partial \mathbf{z}}{\partial \boldsymbol{\Omega}} \frac{\partial \mathbf{k}}{\partial \mathbf{z}} .$$

If the vertices of T are $\{\mathbf{V}_p\}_{p=1}^3$, then the vertices of \mathcal{Z} should obviously satisfy

$$\mathbf{V}_i = \mathbf{Z}_i / Z_i . \quad (\text{A.13})$$

Two specific choices for the \mathbf{Z}_i are mentioned here.

- For the definition angular basis functions in section 2.3,

$$\mathbf{Z}_i = \frac{1}{\|\mathbf{V}_i\|_1} \mathbf{V}_i, \quad (\text{A.14})$$

in which case equation A.11 simplifies to

$$\frac{\partial \mathbf{z}}{\partial \Omega} = z (I^{3 \times 3} - \text{sign}(\mathbf{z}) \mathbf{z}^T),$$

where the sign function acts component-wise. If the vertices are all in the same octant (that is, $\text{sign}(\mathbf{V}_i)$ is independent of i), then \mathcal{Z} lies on the L1-sphere (i.e.: the octahedron).

- A more natural choice is

$$\mathbf{Z}_i = \mathbf{V}_i. \quad (\text{A.15})$$

The resulting parameterisation forms the basis for several widely applied numerical integration techniques.

A.2.2 A parameterisation in angles

It is also possible to parameterise a spherical triangle via two angles. Concretely, the idea is to divide T into circle segments, all of which are an intersection between T and a plane that is parallel to the great circle that intersects \mathbf{V}_1 and \mathbf{V}_3 . That is, all circle segments lie in planes with a common normal vector. The line from $\mathbf{k} = [0, 0]^T$ to $\mathbf{k} = [0, 1]^T$ is mapped to the great circle segment that runs from \mathbf{V}_1 to \mathbf{V}_3 , and $\mathbf{k} = [1, 0]^T$ is mapped to \mathbf{V}_2 . Lines of constant k_1 are mapped to circle segments on T that run from $\mathbf{P}_1(k_1)$ to $\mathbf{P}_2(k_1)$.

Before working out the details, it is convenient to introduce some definitions. Without loss of generality, it is assumed that \mathbf{V}_1 , \mathbf{V}_2 and \mathbf{V}_3 are positively orientated:²

$$(\mathbf{V}_1 \times \mathbf{V}_2) \cdot \mathbf{V}_3 > 0. \quad (\text{A.16})$$

Denote the normals of the planes of the great circles by

$$\begin{aligned} \mathbf{n} &= \mathbf{V}_3 \times \mathbf{V}_1 / \|\mathbf{V}_3 \times \mathbf{V}_1\|_2 \\ \mathbf{m} &= \mathbf{V}_3 \times \mathbf{V}_2 / \|\mathbf{V}_3 \times \mathbf{V}_2\|_2 \\ \mathbf{q} &= \mathbf{V}_1 \times \mathbf{V}_2 / \|\mathbf{V}_1 \times \mathbf{V}_2\|_2 \end{aligned}$$

²For the results in this thesis, the labels \mathbf{V}_1 , \mathbf{V}_2 and \mathbf{V}_3 were assigned to the vertices such that equation A.16 holds and that \mathbf{V}_1 and \mathbf{V}_3 are as far apart as possible. This generally leads to the ‘smoothest’ parameterisation, because the norm of the Jacobian doesn’t vary on curves of constant k_1 , ref. section A.2.3. This is not essential though.

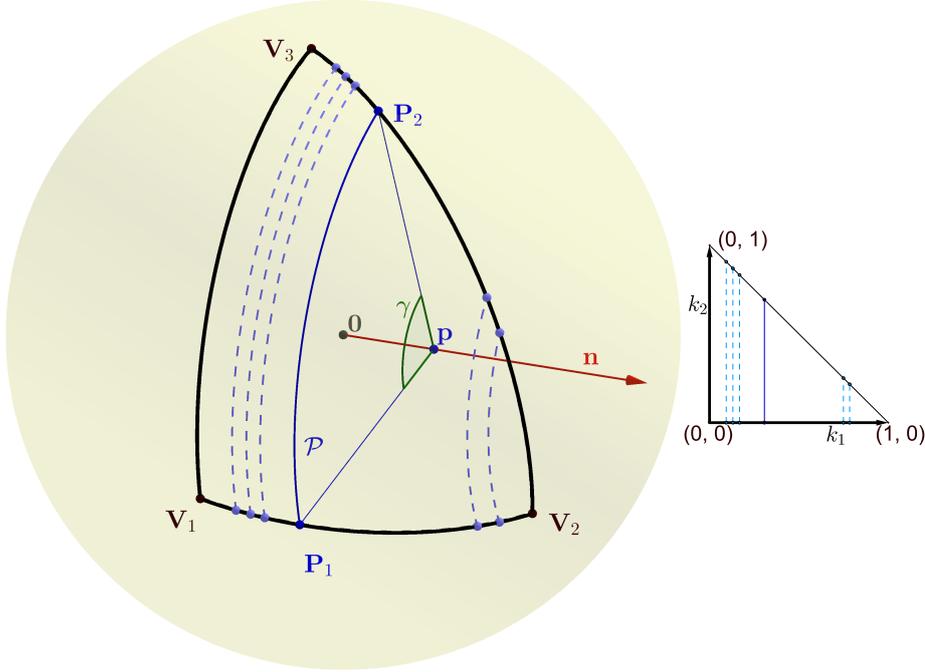


Figure A.3: An overview of the parameterisation of a spherical triangle. *Left*: The vector \mathbf{n} is normal to both the plane that intersects $\{\mathbf{V}_1, \mathbf{V}_3, \mathbf{0}\}$ and the plane that intersects $\{\mathbf{P}_1, \mathbf{P}_2, \mathbf{p}\}$. The circle segment \mathcal{P} is centred at \mathbf{p} and subtends an angle γ . By letting \mathbf{P}_1 run from \mathbf{V}_1 to \mathbf{V}_2 , \mathcal{P} traces the spherical triangle. *Right*: projection of \mathcal{P} -curves on \mathcal{K}^{ref} for different values of \mathbf{P}_1

and the angles of the great circle segments as

$$\begin{aligned}\alpha &= \arccos(\mathbf{V}_3 \cdot \mathbf{V}_2) \\ \beta &= \arccos(\mathbf{V}_1 \cdot \mathbf{V}_2).\end{aligned}$$

In order to find an explicit expression for the parametrisation $\boldsymbol{\Omega}(k_1, k_2)$ of T , first let \mathbf{P}_1 run over the arc from \mathbf{V}_1 to \mathbf{V}_2 :

$$\mathbf{P}_1(k_1) = \mathbf{V}_1 \cos(k_1\beta) + \mathbf{q} \times \mathbf{V}_1 \sin(k_1\beta) \quad 0 \leq k_1 \leq 1. \quad (\text{A.17})$$

If k_2 is varied with k_1 constant, the circle segment

$$\mathcal{P}(c) = \{\boldsymbol{\Omega}(k_1, k_2) \in T : k_1 = c \text{ and } 0 \leq k_2 \leq 1 - c\}$$

is in a plane with normal vector \mathbf{n} . It runs from $\mathbf{P}_1(k_1)$ to its endpoint

$$\mathbf{P}_2(k_1) = \boldsymbol{\Omega}(k_1, k_2 = 1 - k_1) \quad (\text{A.18})$$

on the great circle between \mathbf{V}_3 and \mathbf{V}_2 . This is illustrated in figure A.3. Let \mathbf{p} be the centre of \mathcal{P} . It is the component of $\mathbf{P}_1(k_1)$ in the \mathbf{n} -direction:

$$\mathbf{p}(k_1) = p(k_1) \mathbf{n} = (\mathbf{P}_1(k_1) \cdot \mathbf{n}) \mathbf{n}. \quad (\text{A.19})$$

From equation A.3, $\mathbf{\Omega}$ is found to be

$$\begin{aligned} \mathbf{\Omega}(\mathbf{k}) &= \mathbf{p}(k_1) + \cos\left(\frac{k_2}{1-k_1}\gamma(k_1)\right) (\mathbf{P}_1(k_1) - \mathbf{p}(k_1)) \\ &\quad + \sin\left(\frac{k_2}{1-k_1}\gamma(k_1)\right) \mathbf{P}_1(k_1) \times \mathbf{n}, \end{aligned} \quad (\text{A.20})$$

where

$$\gamma(k_1) = \arccos\left(\frac{(\mathbf{P}_1(k_1) - \mathbf{p}(k_1)) \cdot (\mathbf{P}_2(k_1) - \mathbf{p}(k_1))}{(\mathbf{P}_1(k_1) - \mathbf{p}(k_1))^2}\right) \quad (\text{A.21})$$

is the angle between $(\mathbf{P}_1 - \mathbf{p})$ and $(\mathbf{P}_2 - \mathbf{p})$. Since \mathbf{P}_2 lies on the boundary of T between \mathbf{V}_3 and \mathbf{V}_2 , $\gamma(k_1)$ could in principle be found by isolating it from

$$\mathbf{m} \cdot \mathbf{\Omega}|_{k_2=1-k_1} = 0,$$

but the process is somewhat messy. A simpler way forward is to consider the parametrisation of the great circle segment between \mathbf{V}_3 and \mathbf{V}_2 :

$$\mathbf{s}(y) = \mathbf{V}_3 \cos(y\alpha) + \mathbf{m} \times \mathbf{V}_3 \sin(y\alpha) \quad 0 \leq y \leq 1. \quad (\text{A.22})$$

Define y^* implicitly by

$$\mathbf{s}(y^*) = \mathbf{P}_2(k_1). \quad (\text{A.23})$$

Combine equations A.18 and A.23, take the inner product with \mathbf{n} and use equation A.19 to find

$$p = \mathbf{n} \cdot (\mathbf{m} \times \mathbf{V}_3) \sin(y^*\alpha), \quad (\text{A.24})$$

which is solved by

$$y^* = \frac{1}{\alpha} \arcsin\left(\frac{p}{\mathbf{n} \cdot (\mathbf{m} \times \mathbf{V}_3)}\right). \quad (\text{A.25})$$

Note that $\mathbf{n} \cdot (\mathbf{m} \times \mathbf{V}_3)$ is strictly positive due to the positive orientation of \mathbf{V}_3 , \mathbf{V}_1 and \mathbf{V}_2 . Equations A.17, A.20, A.21, A.22, A.23 and A.25 now form a bijection between (k_1, k_2) and the spherical element.

The derivative with respect to k_2 is fairly straightforward. Recall that $\boldsymbol{\Omega}|_{k_1}$ lies on a circle segment with centre $\mathbf{p}(k_1)$, so that, from equation A.4,

$$\frac{\partial \boldsymbol{\Omega}}{\partial k_2} = \frac{\gamma(k_1)}{1-k_2} (\boldsymbol{\Omega}(k_1, k_2) - \mathbf{p}(k_1)) \times \mathbf{n} = \frac{\gamma(k_1)}{1-k_2} \boldsymbol{\Omega}(k_1, k_2) \times \mathbf{n}.$$

Several intermediate steps are required to obtain an explicit expression for $\partial \boldsymbol{\Omega} / \partial k_1$. From equation A.4,

$$\frac{d\mathbf{P}_1}{dk_1} = \beta \mathbf{q} \times \mathbf{P}_1(k_1), \quad (\text{A.26})$$

with which the derivative of equation A.19 can be evaluated as

$$\frac{d\mathbf{p}}{dk_1} = \beta \cos(k_1 \beta) ((\mathbf{q} \times \mathbf{V}_1) \cdot \mathbf{n}) \mathbf{n}. \quad (\text{A.27})$$

The derivative of equation A.24 is

$$\alpha \frac{dy^*}{dk_1} \cos(y^* \alpha) = \frac{dp/dk_1}{\mathbf{n} \cdot (\mathbf{m} \times \hat{\mathbf{V}}_3)}.$$

This can be used to simplify the derivative of equation A.23, yielding

$$\frac{d\mathbf{P}_2}{dk_1} = \frac{dF/dk_1}{\mathbf{n} \cdot (\mathbf{m} \times \hat{\mathbf{V}}_3)} (\mathbf{m} \times \hat{\mathbf{V}}_3 - \tan(\alpha y^*) \mathbf{V}_3). \quad (\text{A.28})$$

For any two vectors $\mathbf{v}(z)$ and $\mathbf{w}(z)$ of equal length $v = w$, it can be shown by direct computation that

$$\frac{d}{dz} \left(\frac{\mathbf{v} \cdot \mathbf{w}}{v^2} \right) = v^{-2} \left(\mathbf{w} \cdot \frac{d\mathbf{v}}{dz} + \mathbf{v} \cdot \frac{d\mathbf{w}}{dz} \right) - 2v^{-4} \left(\mathbf{v} \cdot \frac{d\mathbf{v}}{dz} \right) (\mathbf{v} \cdot \mathbf{w}).$$

Taking the cosine of equation A.21, differentiating implicitly with respect to k_1 , using the vector identity above and rearranging results in

$$\begin{aligned} \sin(\gamma) \frac{d\gamma}{dk_1} = & 2 |\mathbf{P}_1 - \mathbf{p}|^{-4} \left((\mathbf{P}_1 - \mathbf{p}) \cdot \left(\frac{d\mathbf{P}_1}{dk_1} - \frac{d\mathbf{p}}{dk_1} \right) \right) (\mathbf{P}_1 - \mathbf{p}) \cdot (\mathbf{P}_2 - \mathbf{p}) \\ & - |\mathbf{P}_1 - \mathbf{p}|^{-2} \left((\mathbf{P}_1 + \mathbf{P}_2 - 2\mathbf{p}) \cdot \frac{d\mathbf{p}}{dk_1} + (\mathbf{P}_2 - \mathbf{p}) \cdot \frac{d\mathbf{P}_1}{dk_1} + (\mathbf{P}_1 - \mathbf{p}) \cdot \frac{d\mathbf{P}_2}{dk_1} \right). \end{aligned} \quad (\text{A.29})$$

Finally, differentiate equation A.20 to obtain

$$\begin{aligned} \frac{\partial \boldsymbol{\Omega}}{\partial k_1} = & \frac{\partial \mathbf{p}}{\partial k_1} + \frac{k_2}{1-k_1} \left(\frac{\gamma}{1-k_1} + \frac{d\gamma}{dk_1} \right) \boldsymbol{\Omega} \times \mathbf{n} \\ & + \left(\frac{d\mathbf{P}_1}{dk_1} - \frac{d\mathbf{p}}{dk_1} \right) \cos \left(\frac{k_2}{1-k_1} \gamma \right) + \frac{d\mathbf{P}_1}{dk_1} \times \mathbf{n} \sin \left(\frac{k_2}{1-k_1} \gamma \right). \end{aligned}$$

A.2.3 A comparison of the Jacobians

It is a well-known fact that there are no bijections $\mathcal{K}^{\text{ref}} \leftrightarrow T$ that preserve the area. (Robbin and Salamon, 2013). An infinitesimal area $dk_1 dk_2$ on \mathcal{K}^{ref} is mapped onto a surface with area $|\mathbf{J}| dk_1 dk_2$ on T . The Jacobian is defined as

$$\mathbf{J}(\mathbf{k}) \equiv \frac{\partial \Omega}{\partial k_1} \times \frac{\partial \Omega}{\partial k_2} . \quad (\text{A.30})$$

In other words, the norm of the Jacobian cannot be constant.

This doesn't mean that all parameterisations deform \mathcal{K}^{ref} equally. The norm of the Jacobian is a simple measure for the 'skewness'. It is plotted in figure A.4 for the three parameterisations mentioned in this section. For large spherical elements, the angular parameterisation is considerably smoother than the parameterisations of section A.2.1. The mapping via a flat plane with vertices \mathbf{V}_i is always smoother than the mapping via a flat plane on the octahedron. As the spherical triangles get smaller, they become flat, and the Jacobians become constant. For the angular parameterisation, the norm of the Jacobian doesn't depend on k_2 .

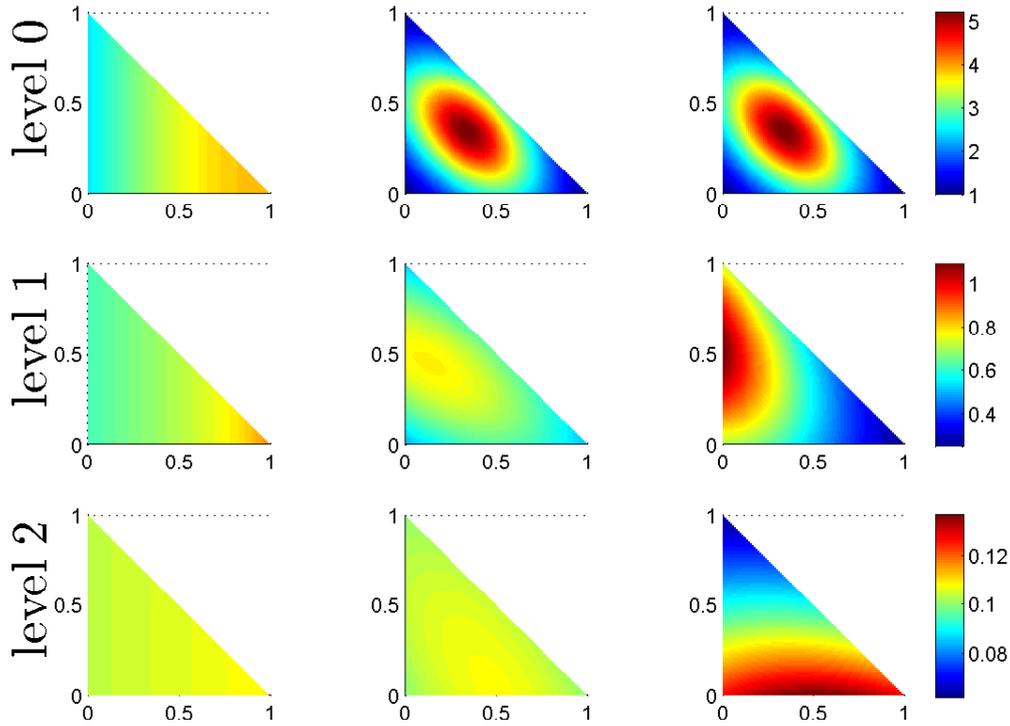


Figure A.4: Values of $|\mathbf{J}(\mathbf{k})|$ on the reference triangle \mathcal{K}^{ref} at various levels of the angular mesh described in section 2.2. k_1 and k_2 are on the horizontal and vertical axes respectively.

- First column: angular parameterisations (section A.2.2)
- Second column: parameterisations via flat triangle (section A.2.2); vertices of \mathcal{Z} are \mathbf{V}_i (equation A.15)
- Third column: parameterisations via flat triangle (section A.2.2); vertices of \mathcal{Z} are $\mathbf{V}_i/||\mathbf{V}_i||_1$ (equation A.14)

Note that at level 0, the spherical triangle is an octant of \mathbb{S}^2 , so equations A.14 and A.15 are equivalent. The results for level 1 and 2 are not unique.

Appendix B

Integrals over spherical elements

In order to implement a DG method in the solid angle, functions need to be integrated over angular elements (‘patches’) and their boundaries. It is conceptually easiest to view the set of all possible solid angles as a spherical surface \mathbb{S}^2 in \mathbb{R}^3 . The boundary of an element consists of three segments of great circles on the unit sphere. Due to the complicated shape, there appears to be no analytical expression for the integrals, except in some simple cases. This chapter details how the integrals can be computed numerically.

The usual approach in finite element methods is to transform the domain of integration into a standard ‘local’ reference element. This necessitates a suitable parametrisation of the region of interest. The integration is performed with a quadrature set on the local element. To obtain an integration rule, one can map the local abscissa to a ‘global’ point in the physical space and multiply the weights with the norm of the Jacobian of the bijection. The numerical schemes differ from each other due to different choices of the parameterisation and the local quadrature set.

If the global elements are Euclidean, there exists an affine mapping to the local element. That is, the Jacobian is constant and the transformation doesn’t affect the shape of the function. Polynomials on a global element are also polynomials on the local element. There are optimised local quadrature sets that integrate specific functions analytically, resulting in simple and efficient schemes. Concrete lists of abscissa and weights can be found in many places. A good general reference is Solin et al. (2003).

Parameterisations of spherical triangles cannot be linear. Numerical integration is therefore considerably less straightforward, and typically more expensive.

B.1 Integrals over orthodrome segments

This section deals with integrals over curves in Ω -space. If $\mathbf{V}_i \in \mathbb{S}^2$ and $\mathbf{V}_1 \neq \pm\mathbf{V}_2$, then the integrals of interest are of the form

$$\int_{\mathcal{C}} f(\Omega) \, d\Omega, \quad (\text{B.1})$$

where $f : \mathcal{C} \mapsto \mathbb{R}$ is an arbitrary function and $\mathcal{C} \in \mathbb{S}^2$ is the shortest orthodrome segment that connects \mathbf{V}_1 to \mathbf{V}_2 .

B.1.1 Analytical results

Some progress can be made in the evaluation of the integral B.1 without resorting to approximating methods. From equation A.5,

$$I \equiv \int_{\mathcal{C}} f(\Omega) \, d\Omega = \int_{t=0}^{\gamma} f(\Omega(t)) \, dt ,$$

where $\gamma = \arccos(\mathbf{V}_1 \cdot \mathbf{V}_2)$ and

$$\Omega(t) = \mathbf{V}_1 \cos(t) + \mathbf{p} \sin(t) ,$$

where \mathbf{p} is known. Letting $x = \cos(t)$, this can be rewritten to

$$I = \int_{\mathbf{V}_1 \cdot \mathbf{V}_2}^1 \frac{f(\Omega(x))}{\sqrt{1-x^2}} \, dx \quad (\text{B.2})$$

and

$$\Omega(x) = x \mathbf{V}_1 + \sqrt{1-x^2} \mathbf{p} .$$

If f is a polynomial in the components of Ω , then I can be evaluated analytically. It suffices to consider an arbitrary non-constant monomial $f(\Omega) = (\Omega_1)^m (\Omega_2)^n (\Omega_3)^q$. Expanding all terms in equation B.2 leads to an expression of the form

$$I = \sum_j c_j \int_{\mathbf{V}_1 \cdot \mathbf{V}_2}^1 x^{r_j} (1-x^2)^{s_j/2} \, dx .$$

The coefficients c_j depend on m , n , q , and on the components of \mathbf{V}_1 and \mathbf{p} . The corresponding r_j and s_j are non-negative integers that are bounded by $r_j + s_j \leq m + n + q - 1$. If s_j is even, the j 'th integral is a polynomial in x of order $r_j + s_j$, and it can be evaluated analytically with an appropriate Gaussian quadrature set. If s_j is odd, the integral can always be expressed in terms of x , $\sqrt{1-x^2}$ and $\arcsin(x)$.

Other functions pose greater difficulties analytically, but may still yield nicely to a more generic numerical method, which is the subject of the remainder of this section.

B.1.2 Numerical integration rules

Given a local quadrature set in $[-1, 1]$, consisting of the abscissas $\{t_i\}_{i=1}^N$ and the corresponding weights $\{w_i\}_{i=1}^N$, the parameterisations of section A.1 can be used for numerical integration over spherical arcs:

$$\int_{\mathcal{C}} f(\boldsymbol{\Omega}) d\boldsymbol{\Omega} = \int_{-1}^1 f(\boldsymbol{\Omega}(k)) \left\| \frac{d\boldsymbol{\Omega}}{dk} \right\|_2 dk \approx \sum_{i=1}^N w_i \left[f(\boldsymbol{\Omega}(k)) \left\| \frac{d\boldsymbol{\Omega}}{dk} \right\|_2 \right]_{k=t_i} .$$

The local quadrature set considered here is always Gaussian. More accurate estimates can be obtained by increasing N . At some point, the error in the approximation is dominated by the machine epsilon, and it becomes pointless to use higher quadrature orders.

Some tests were performed to determine the accuracy of the numerical scheme. The analytical results from the previous section provided useful reference values, because they can easily be evaluated to arbitrary precision. As one would expect, the different parameterisations vary somewhat in their accuracy. Risking a statement of the obvious, it must be emphasised that both the location of the vertices and the type of function affect the accuracy greatly. Nevertheless, some general observations can be made.

- The quadrature set tends to perform better when the norm of the Jacobian of the transformation doesn't vary much over the interval. With parameterisations via a straight line (section A.1.1), this is achieved by letting the straight line segment run from \mathbf{V}_1 to \mathbf{V}_2 . Only the angular parameterisation (section A.1.2), has a constant weight factor. This yields the best results.
- The numerical error due to the limited machine precision is roughly the same for all methods.
- Small spherical arcs require fewer quadrature points than larger ones.

If f is a polynomial of third order in the components of $\boldsymbol{\Omega}$, and the length of \mathcal{C} is at most $\pi/2$, then at most 8 quadrature points with an angular parameterisation on a double precision machine are needed to ensure that the error is mostly due to rounding errors.

B.2 Integrals over spherical triangles

This section deals with the numerical evaluation of integrals of the form

$$\int_T f(\boldsymbol{\Omega}) d\boldsymbol{\Omega} , \tag{B.3}$$

where $f : T \mapsto \mathbb{R}$ is an arbitrary function and $T \in \mathbb{S}^2$ is a spherical triangle with vertices \mathbf{V}_1 , \mathbf{V}_2 and \mathbf{V}_3 .

Sometimes integrals need to be carried out over the whole Ω -space, for example when calculating the scalar flux, or when determining the energy of the difference of two functions. Due to the wide variety of applications, the more general problem of efficient integration over an n -dimensional sphere has recently attracted considerable attention from mathematicians. (Folland, 2001; Hesse and Womersley, 2012; Othmani, 2011) Their methods might be applicable to the current problem as well. A more obvious approach in a practical particle transport code is to sum over the integrals over the spherical elements. Alternatively, Lebedev quadrature sets are designed to integrate polynomials in Ω over \mathbb{S}^2 analytically, and they are also suitable for numerical integration of other functions.

Integrals over a part of the unit sphere are far less well studied. Efficient numerical integration rules have been found for spherical caps $\{\Omega \in \mathbb{S}^2 : \Omega \cdot \mathbf{v} \geq \cos(\mu)\}$ (Hesse and Womersley, 2012), but it is not obvious how they would translate to T , the problem being the complicated shape of the boundary. Integrals of multivariate trigonometric polynomials over triangular elements on the unit hypersphere \mathbb{S}^n are studied in Beckmann et al. (2012). This comes closest to making some analytical progress in the evaluation of integrals over spherical triangles, but the set of functions under consideration there is quite limited. Arbitrary functions on spherical triangles in \mathbb{S}^2 have received very little attention, though an important exception from the field of chemical physics can be found in Bardhan et al. (2007), which also deals with generalised spherical triangles.

It is worth mentioning that constant functions can be integrated analytically over T , since the area can be found from a special case of Girard's formula:

$$\text{sum of interior angles of } T = \pi + \text{area} .$$

Unfortunately, the computation of the angles necessarily involves divisions by $\|\mathbf{V}_i \times \mathbf{V}_j\|_2$ for $i \neq j$, which results in appreciable numerical errors if the area is small. This practical issue appears not to be mentioned in most places that cite Girard's formula. It is of no great concern however, since smaller spherical triangles can be dealt with quite well by the numerical methods described in the remainder of this section.

For arbitrary functions, the best numerical method for evaluating B.3 might be an adaptive one. Contrary to most other methods, this has the benefit of providing some control over the error, and a means to ascertain that it dropped below a certain threshold. The efficiency of this approach

depends on how well one can direct the refinement effort automatically. Boal and Sayas (2004) use Richardson extrapolation to accelerate the convergence and aid the error estimate. For the implementation of a spherical finite element method, however, multiple functions need to be integrated over the same spherical element. It is probably cheaper to use a single quadrature set for all functions, rather than to integrate them adaptively one by one.

By sampling random points uniformly on the smallest spherical cap that contains T , a Monte Carlo rejection method can be used to estimate the integrals. This is highly expensive, and so the practicality is limited to the validation of the implementation of any of the other methods.

B.2.1 Numerical integration rules

Given a local quadrature set on \mathcal{K}^{ref} (equation A.6) with abscissa $\{\mathbf{t}_i\}_{i=1}^N$ and weights $\{w_i\}_{i=1}^N$, the parameterisations of section A.2 can be used for a numerical evaluation of B.3.

$$\int_T f(\boldsymbol{\Omega}) \, d\boldsymbol{\Omega} = \int_{\mathcal{K}^{\text{ref}}} f(\boldsymbol{\Omega}(\mathbf{k})) \|\mathbf{J}(\mathbf{k})\|_2 \, d\mathbf{k} \approx \sum_{i=1}^N w_i \left[f(\boldsymbol{\Omega}(\mathbf{k})) \|\mathbf{J}(\mathbf{k})\|_2 \right]_{\mathbf{k}=\mathbf{t}_i}, \quad (\text{B.4})$$

where the Jacobian \mathbf{J} is given by equation A.30. A great variety of quadrature sets on \mathcal{K}^{ref} can be found in finite element literature, e.g. Solin et al. (2003) and Sunder and Cookson (1985). In combination with the three parameterisations considered here, the number of distinct integration rules is quite large.

Previous work

Seemingly independently, various authors have come up with their own combination of a parameterisation and a local quadrature set. Most employ the parameterisation via a flat triangle with vertices \mathbf{V}_i . Freeden (2010) recommends a three- or four-point quadrature set, but this lacks the necessary accuracy for a finite element implementation. Kópházi and Lathouwers (2015) subdivide T into smaller spherical triangles, and uses a 34-point quadrature set on each of them, resulting in relative errors of roughly 10^{-12} .

In Jarrell (2010) the vertices of the intermediate flat triangle are located on the unit L1-sphere, and the equivalent representations of the derivatives are derived in a more complicated geometrical fashion. The local quadrature set is constructed by taking a tensor product of two (presumably Gaussian) one-dimensional quadrature sets. The number of quadrature points can thus

be increased easily, so that high accuracy can always be achieved, irrespective of f .

An entirely different parameterisation is introduced in Bardhan et al. (2007). It was designed to be applicable to generalised spherical triangles, where the boundaries are segments of small circles. No mention is made of its accuracy.

Some general observations

As with the integration over orthodrome segments, several general observations can be made, despite the fact that the accuracy and the cost obviously depend heavily on both T and f .

- With increasing N , the estimation will converge more quickly if the Jacobian doesn't vary much on \mathcal{K}^{ref} .
- In the case of parameterisations via an intermediate flat triangle (section A.2.1), small spherical triangles require fewer quadrature points to achieve a certain accuracy. This is a direct consequence of the previous point.
- The parameterisation via a triangle on the L1-sphere should not be used for integration. It is inferior in two ways to the case where the intermediate triangle has vertices \mathbf{V}_i . First, the Jacobian is considerably less smooth (ref. figure A.4). Second, the numerical error due to the finite machine precision was observed to be greater, especially for smaller spherical elements.

The method was used in Jarrell and Adams (2011), where high precision was required. The error could probably have been lowered by placing the vertices of the intermediate triangle at the \mathbf{V}_i .

- As T becomes smaller, the limited precision starts to play a major factor in the angular parameterisation of section A.2.2. (This is caused by the divisions by $(1 - k_1)$.) For large triangles, however, the rounding error is comparable to that of the parameterisations of section A.2.1, and the smooth Jacobian effects a fast convergence.
- For a given N , the type of local quadrature set has a great impact on the accuracy. It has not become clear which choice is best.

For convenience, all computations in this work were performed with a product quadrature, derived from one-dimensional Gaussian quadrature sets. This makes it easy to ensure that the final error is dominated by the machine

precision. Large spherical triangles were parameterised with the angular mapping of section A.2.2, while the smaller ones were done with the method of section A.2.1, where the intermediate plane had vertices $\{\mathbf{V}_i\}_{i=1}^3$.

Bibliography

- I. Aavatsmark, G. T. Eigestad, B. T. Mallison, and J. M. Nordbotten. A compact multipoint flux approximation method with improved robustness. *Numerical Methods for Partial Differential Equations*, 24(5):1329–1360, 2008. ISSN 1098-2426.
- Léo Agélas, Daniele A. Di Pietro, and Jérôme Droniou. The g method for heterogeneous anisotropic diffusion on general meshes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 44(4):597–625, 6 2010.
- Abdelkader Baggag, Harold Atkins, and David Keyes. Parallel implementation of the discontinuous galerkin method. Technical Report No. NAS1-97046, National Aeronautics and Space Administration, 1999.
- Jaydeep P. Bardhan, Michael D. Altma, David J. Willis, Shaun M. Lippow, Bruce Tidor, and Jacob K. White. Numerical integration techniques for curved-element discretizations of molecule-solvent interfaces. *The Journal of Chemical Physics*, 127(1):014701+, 2007.
- J. Beckmann, H. N. Mhaskar, and J. Prestin. Quadrature formulas for integration of multivariate trigonometric polynomials on spherical triangles. *GEM - International Journal on Geomathematics*, 3(1):119–138, 2012. ISSN 1869-2672.
- N. Boal, V. Domnguez, and F-J. Sayas. Asymptotic properties of some triangulations of the sphere. *Journal of Computational and Applied Mathematics*, 211(1):11 – 22, 2008. ISSN 0377-0427.
- Natalia Boal and Francisco-Javier Sayas. Adaptive numerical integration on spherical triangles. *Monografías del Seminario Matemático García de Galdeano*, (31):61–69, 2004.
- Christoph Borgers and Edward W. Larsen. On the accuracy of the fokker-planck and fermi pencil beam equations for charged particle transport. *Medical Physics*, 23, 1996a.

- Christoph Borgers and Edward W. Larsen. Asymptotic derivation of the fermi pencil-beam approximation. *Nuclear Science and Engineering*, 123: 343–357, 1996b.
- Gerhard Dziuk and Charles M. Elliott. Finite element methods for surface pdes. *Acta Numerica*, 22:289–396, 5 2013. ISSN 1474-0508.
- Gerald B. Folland. How to integrate a polynomial over a sphere. *The American Mathematical Monthly*, 108(5):pp. 446–448, May 2001. ISSN 00029890.
- W. Freeden. *Handbook of Geomathematics*. Springer reference. Springer, 2010. ISBN 9783642015458.
- Irene Gamba, Maria Pia Gualdani, Richard W Sharp, et al. An adaptable discontinuous galerkin scheme for the wigner-fokker-planck equation. *Communications in Mathematical Sciences*, 7(3):635–664, 2009.
- Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, third edition, 1996.
- Ralf Hartmann. Numerical analysis of higher order discontinuous galerkin finite element methods. *VKI Lecture Series 2008-08*, October 2008.
- Kerstin Hesse and Robert S. Womersley. Numerical integration with polynomial exactness over a spherical cap. *Advances in Computational Mathematics*, 36(3):451–483, 2012. ISSN 1019-7168.
- Joshua Jarrell. *An Adaptive Angular Discretization Method for Neutral-Particle Transport in Three-Dimensional Geometries*. PhD thesis, Texas A&M University, December 2010.
- Joshua J Jarrell and Marvin L Adams. discrete-ordinates quadrature sets based on linear discontinuous finite elements. In *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering*, 3133 TAMU, College Station, TX 77843-3133, May 2011.
- Xun Jia, Todd Pawlicki, Kevin T. Murphy, and Arno J. Mundt. Proton therapy dose calculations on gpu: advances and challenges. *Translational Cancer Research*, 1(3), 2012. ISSN 2218-676X.
- József Kópházi and Danny Lathouwers. A space-angle dgfem approach for the boltzmann radiation transport equation with local angular refinement. *Journal of Computational Physics*, 297:637 – 668, 2015. ISSN 0021-9991.

- Chohong Min, Frédéric Gibou, and Hector D. Ceniceros. A supra-convergent finite difference scheme for the variable coefficient poisson equation on non-graded grids. *Journal of Computational Physics*, 2006.
- J. E. Morel. Fokker-planck calculations using standard discrete ordinates transport codes. *Nuclear Science and Engineering*, 79:340–356, June 1981.
- J. E. Morel. An improved fokker-planck angular differencing scheme. *Nuclear Science and Engineering*, 89:131–136, September 1984.
- Yamen Othmani. Polynomial integration over the unit sphere. *Applied Mathematics Letters*, 24(7):1260 – 1264, 2011. ISSN 0893-9659.
- Harald Paganetti. *Proton Therapy Physics*. series in medical and biomedical engineering. CRC Press, Taylor & Francis Group, 2012a. ISBN 978-1-4398-3644-6.
- Harald Paganetti. Range uncertainties in proton therapy and the role of monte carlo simulations. *Physics in medicine and biology*, 57(11):R99117, June 2012b. ISSN 0031-9155.
- Harald Paganetti, Hongyu Jiang, Katia Parodi, Roelf Slopsema, and Martijn Engelsman. Clinical implementation of full monte carlo dose calculation in proton beam therapy. *Physics in Medicine and Biology*, 53(17):4825, 2008.
- D. A. Di Pietro. On the conservativity of cell centered galerkin methods. *Comptes rendus - Mathematique*, 351(3-4):155–159, 2013.
- Daniele A. Di Pietro. Cell centered galerkin methods. *Comptes Rendus Mathematique*, 348(12):31 – 34, 2010. ISSN 1631-073X.
- Daniele A. Di Pietro and Alexandre Ern. Mathematical aspects of discontinuous Galerkin methods, 2012. ISSN 1154-483X.
- Daniele A. Di Pietro and Jean-Marc Gratien. Lowest order methods for diffusive problems on general meshes: A unified approach to definition and implementation. In *Finite Volumes for Complex Applications VI Problems & Perspectives*, pages 803–819. Springer, February 2011.
- Daniele Antonio Di Pietro. Cell centered Galerkin methods for diffusive problems. *ESAIM: Mathematical Modelling and Numerical Analysis*, 46(1):111–144, August 2011.

- Joel W. Robbin and Dietmar A. Salamon. Introduction to differential geometry, September 2013.
- Bruno Rossi and Kenneth Greisen. Cosmic-ray theory. *Reviews of Modern Physics*, 13:241–309, 1941.
- Paul H. Smith. Analysis of moment-preserving methods for charged particle transport. Master’s thesis, University of New Mexico, 2007.
- P. Solin, K. Segeth, and I. Dolezel. *Higher-Order Finite Element Methods*. Studies in Advanced Mathematics. Taylor & Francis, 2003. ISBN 9780203488041.
- K. Sham Sunder and R. A. Cookson. Integration points for triangles and tetrahedrons obtained from the gaussian quadrature points for a line. *Computers & Structures*, 21(5):881 – 885, 1985. ISSN 0045-7949.
- Alex Szalay, Jim Gray, Gyorgy Fekete, Peter Kunszt, Peter Kukol, and Ani Thakar. Indexing the sphere with the hierarchical triangular mesh. Technical Report MSR-TR-2005-123, Microsoft Research, September 2005.
- Sander Bastiaan Uilkema. Proton therapy planning using the S_N method with the fokker-planck approximation. Master’s thesis, Delft University of Technology, October 2012.
- P. van Slingerland and C. Vuik. Spectral two-level deflation for dg: a preconditioner for cg that does not need symmetry, September 2011. Delft University of Technology, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft Institute of Applied Mathematics; internal report ISSN 1389-6520.